Delivering Expressive And Personalized Fingertip Tactile Cues

Eric Young
University of Pennsylvania

Follow this and additional works at: https://repository.upenn.edu/edissertations

Part of the Mechanical Engineering Commons

Recommended Citation
Young, Eric, "Delivering Expressive And Personalized Fingertip Tactile Cues" (2020). Publicly Accessible Penn Dissertations. 4143.
https://repository.upenn.edu/edissertations/4143

This paper is posted at ScholarlyCommons. https://repository.upenn.edu/edissertations/4143
For more information, please contact repository@pobox.upenn.edu.
Delivering Expressive And Personalized Fingertip Tactile Cues

Abstract
Wearable haptic devices have seen growing interest in recent years, but providing realistic tactile feedback is not a challenge that is soon to be solved. Daily interactions with physical objects elicit complex sensations at the fingertips. Furthermore, human fingertips exhibit a broad range of physical dimensions and perceptive abilities, adding increased complexity to the task of simulating haptic interactions in a compelling manner. However, as the applications of wearable haptic feedback grow, concerns of wearability and generalizability often persuade tactile device designers to simplify the complexities associated with rendering realistic haptic sensations. As such, wearable devices tend to be optimized for particular uses and average users, rendering only the most salient dimensions of tactile feedback for a given task and assuming all users interpret the feedback in a similar fashion. We propose that providing more realistic haptic feedback will require in-depth examinations of higher-dimensional tactile cues and personalization of these cues for individual users. In this thesis, we aim to provide hardware and software-based solutions for rendering more expressive and personalized tactile cues to the fingertip. We first explore the idea of rendering six-degree-of-freedom (6-DOF) tactile fingertip feedback via a wearable device, such that any possible fingertip interaction with a flat surface can be simulated. We highlight the potential of parallel continuum manipulators (PCMs) to meet the requirements of such a device, and we refine the design of a PCM for providing fingertip tactile cues. We construct a manually actuated prototype to validate the concept, and then continue to develop a motorized version, named the Fingertip Puppeteer, or Fuppeteer for short. Various error reduction techniques are presented, and the resulting device is evaluated by analyzing system responses to step inputs, measuring forces rendered to a biomimetic finger sensor, and comparing intended sensations to perceived sensations of twenty-four participants in a human-subject study. Once the functionality of the Fuppeteer is validated, we begin to explore how the device can be used to broaden our understanding of higher-dimensional tactile feedback. One such application is using the 6-DOF device to simulate different lower-dimensional devices. We evaluate 1-, 3-, and 6-DOF tactile feedback during shape discrimination and mass discrimination in a virtual environment, also comparing to interactions with real objects. Results from 20 naive study participants show that higher-dimensional tactile feedback may indeed allow completion of a wider range of virtual tasks, but that feedback dimensionality surprisingly does not greatly affect the exploratory techniques employed by the user. To address alternative approaches to improving tactile rendering in scenarios where low-dimensional tactile feedback is appropriate, we then explore the idea of personalizing feedback for a particular user. We present two generalizable software-based approaches to personalize an existing data-driven haptic rendering algorithm for fingertips of different sizes. We evaluate our algorithms in the rendering of pre-recorded tactile sensations onto rubber casts of six different fingertips as well as onto the real fingertips of 13 human participants, all via a 3-DOF wearable device. Results show that both personalization approaches significantly reduced force error magnitudes and improved realism ratings.
First Advisor
Katherine J. Kuchenbecker

Keywords
haptics, personalization, tactile feedback, wearable devices

Subject Categories
Mechanical Engineering

This dissertation is available at ScholarlyCommons: https://repository.upenn.edu/edissertations/4143
DELIVERING EXPRESSIVE AND PERSONALIZED FINGERTIP TACTILE CUES

Eric M. Young

A DISSERTATION

in

Mechanical Engineering and Applied Mechanics

Presented to the Faculties of the University of Pennsylvania

in

Partial Fulfillment of the Requirements for the

Degree of Doctor of Philosophy

2020

Katherine J. Kuchenbecker, Supervisor of Dissertation
Adjunct Professor of Mechanical Engineering and Applied Mechanics

Jennifer Lukes, Graduate Group Chairperson
Professor of Mechanical Engineering and Applied Mechanics

Dissertation Committee:
Mark Yim, Professor of Mechanical Engineering and Applied Mechanics
Michelle J. Johnson, Associate Professor of Physical Medicine and Rehabilitation
Katherine J. Kuchenbecker, Adjunct Professor of Mechanical Engineering and Applied Mechanics
Acknowledgments

First I would like to thank my PhD supervisor, Dr. Katherine J. Kuchenbecker, for encouraging me to explore questions I found interesting, offering insight when those questions proved challenging, and fostering a welcoming research environment. The past five years have been a wonderful experience, and I am grateful to Katherine for sharing her knowledge of and passion for engineering, teaching, and research. I would also like to thank my thesis committee, Dr. Mark Yim and Dr. Michelle J. Johnson, for sharing their expertise, particularly as it relates to this work.

I am fortunate to have had the opportunity to work with many wonderful researchers during my studies, both in the Penn Haptics Lab and in the Haptic Intelligence Department at the Max Planck Institute for Intelligent Systems. My peers have consistently been willing to share their expertise and ideas, making these labs ideal environments in which to learn. I am also grateful to have truly enjoyed my time in both labs, and thank all past and current members for the many friendly chats, laughs, and adventures outside of work. I hope these friendships continue long after I graduate.

Finally, I would like to thank my family for their constant support and encouragement. Unjustified confidence and sarcasm are only ever a phone call away, and I do not admit enough how much I appreciate both.
ABSTRACT
DELIVERING EXPRESSIVE AND PERSONALIZED FINGERTIP TACTILE CUES
Eric M. Young
Katherine J. Kuchenbecker

Wearable haptic devices have seen growing interest in recent years, but providing realistic tactile feedback is not a challenge that is soon to be solved. Daily interactions with physical objects elicit complex sensations at the fingertips. Furthermore, human fingertips exhibit a broad range of physical dimensions and perceptive abilities, adding increased complexity to the task of simulating haptic interactions in a compelling manner. However, as the applications of wearable haptic feedback grow, concerns of wearability and generalizability often persuade tactile device designers to simplify the complexities associated with rendering realistic haptic sensations. As such, wearable devices tend to be optimized for particular uses and average users, rendering only the most salient dimensions of tactile feedback for a given task and assuming all users interpret the feedback in a similar fashion. We propose that providing more realistic haptic feedback will require in-depth examinations of higher-dimensional tactile cues and personalization of these cues for individual users. In this thesis, we aim to provide hardware and software-based solutions for rendering more expressive and personalized tactile cues to the fingertip.

We first explore the idea of rendering six-degree-of-freedom (6-DOF) tactile fingertip feedback via a wearable device, such that any possible fingertip interaction with a flat surface can be simulated. We highlight the potential of parallel continuum manipulators (PCMs) to meet the requirements of such a device, and we refine the design of a PCM for providing fingertip tactile cues. We construct a manually actuated prototype to validate the concept, and then continue to develop a motorized version, named the Fingertip Puppeteer, or Fuppeteer for short. Various error reduction techniques are presented, and the resulting device is evaluated by analyzing system responses to step inputs, measuring forces rendered to a biomimetic finger sensor, and comparing intended sensations to perceived sensations of twenty-four participants in a human-subject study.

Once the functionality of the Fuppeteer is validated, we begin to explore how the device can be used to broaden our understanding of higher-dimensional tactile
feedback. One such application is using the 6-DOF device to simulate different lower-dimensional devices. We evaluate 1-, 3-, and 6-DOF tactile feedback during shape discrimination and mass discrimination in a virtual environment, also comparing to interactions with real objects. Results from 20 naive study participants show that higher-dimensional tactile feedback may indeed allow completion of a wider range of virtual tasks, but that feedback dimensionality surprisingly does not greatly affect the exploratory techniques employed by the user.

To address alternative approaches to improving tactile rendering in scenarios where low-dimensional tactile feedback is appropriate, we then explore the idea of personalizing feedback for a particular user. We present two generalizable software-based approaches to personalize an existing data-driven haptic rendering algorithm for fingertips of different sizes. We evaluate our algorithms in the rendering of pre-recorded tactile sensations onto rubber casts of six different fingertips as well as onto the real fingertips of 13 human participants, all via a 3-DOF wearable device. Results show that both personalization approaches significantly reduced force error magnitudes and improved realism ratings.
# Contents

Acknowledgments .................. iii

Abstract ........................... iv

List of Tables ..................... x

List of Figures ..................... xvii

1 Introduction ..................... 1
  1.1 Motivation .................. 2
  1.2 Thesis Overview ............ 5

2 A Wearable Device for Delivering 6-DOF Fingertip Haptic Cues .... 8
  2.1 Related Work ................ 9
    2.1.1 Parallel Continuum Manipulators ....... 10
  2.2 Design and Simulation Parameters .......... 12
    2.2.1 Design Space Parameterization ....... 12
    2.2.2 Calculation of Simulator Inputs ....... 13
  2.3 Design Refinement ............ 14
    2.3.1 Phase I of Design Refinement ..... 15
    2.3.2 Phase II of Design Refinement ...... 16
  2.4 Manually Actuated Prototype ........ 17
    2.4.1 Prototype Design and Setup .......... 18
    2.4.2 Simulation Validation .............. 20
    2.4.3 Evaluation of Displayable Tactile Cues .... 22
  2.5 Motorized Prototype Design .... 23
    2.5.1 System Design .............. 26
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.6</td>
<td>Simulated Performance</td>
<td>30</td>
</tr>
<tr>
<td>2.6.1</td>
<td>Inverse Kinematics in Discrete Workspace</td>
<td>31</td>
</tr>
<tr>
<td>2.6.2</td>
<td>Inverse Kinematics in Continuous Workspace</td>
<td>35</td>
</tr>
<tr>
<td>2.7</td>
<td>Experimental Performance</td>
<td>37</td>
</tr>
<tr>
<td>2.7.1</td>
<td>Preliminary Evaluation</td>
<td>37</td>
</tr>
<tr>
<td>2.7.2</td>
<td>Error Reduction via Measured Jacobians</td>
<td>39</td>
</tr>
<tr>
<td>2.7.3</td>
<td>Self-calibration</td>
<td>40</td>
</tr>
<tr>
<td>2.8</td>
<td>User Study on Perception of Rendered Location and Shear Direction</td>
<td>42</td>
</tr>
<tr>
<td>2.8.1</td>
<td>Delivered Cues</td>
<td>42</td>
</tr>
<tr>
<td>2.8.2</td>
<td>Setup</td>
<td>44</td>
</tr>
<tr>
<td>2.8.3</td>
<td>Results And Discussion</td>
<td>47</td>
</tr>
<tr>
<td>2.9</td>
<td>Limitations and Future Considerations</td>
<td>52</td>
</tr>
<tr>
<td>2.10</td>
<td>Summary</td>
<td>53</td>
</tr>
<tr>
<td>3</td>
<td>Vari-Dimensional Tactile Fingertip Feedback</td>
<td>55</td>
</tr>
<tr>
<td>3.1</td>
<td>Related Work</td>
<td>57</td>
</tr>
<tr>
<td>3.2</td>
<td>System Integration</td>
<td>58</td>
</tr>
<tr>
<td>3.2.1</td>
<td>Counterweight</td>
<td>58</td>
</tr>
<tr>
<td>3.2.2</td>
<td>Device Positioning</td>
<td>60</td>
</tr>
<tr>
<td>3.2.3</td>
<td>Finger Positioning</td>
<td>61</td>
</tr>
<tr>
<td>3.2.4</td>
<td>Sensing and Communication</td>
<td>61</td>
</tr>
<tr>
<td>3.3</td>
<td>Human-subject Study</td>
<td>62</td>
</tr>
<tr>
<td>3.3.1</td>
<td>Haptic Rendering Methods</td>
<td>62</td>
</tr>
<tr>
<td>3.3.2</td>
<td>Hidden Surface Task</td>
<td>64</td>
</tr>
<tr>
<td>3.3.3</td>
<td>Slider Task</td>
<td>66</td>
</tr>
<tr>
<td>3.3.4</td>
<td>Procedure</td>
<td>69</td>
</tr>
<tr>
<td>3.4</td>
<td>Results</td>
<td>71</td>
</tr>
<tr>
<td>3.4.1</td>
<td>Performance Metrics</td>
<td>71</td>
</tr>
<tr>
<td>3.4.2</td>
<td>Exploratory Motions</td>
<td>74</td>
</tr>
<tr>
<td>3.4.3</td>
<td>Qualitative Feedback</td>
<td>76</td>
</tr>
<tr>
<td>3.5</td>
<td>Discussion</td>
<td>79</td>
</tr>
<tr>
<td>3.6</td>
<td>Limitations and Future Considerations</td>
<td>83</td>
</tr>
<tr>
<td>3.7</td>
<td>Summary</td>
<td>84</td>
</tr>
</tbody>
</table>
## 4 Rendering Personalized Remotely Sensed Cues

4.1 Related Work ........................................... 88
4.2 Fingertip Device and Sensor Design .................. 90
4.3 Approaches for Personalization ....................... 91
   4.3.1 Data-driven personalization ..................... 92
   4.3.2 Geometric personalization ...................... 93
4.4 Objective Evaluation .................................. 96
4.5 Human-subject Evaluation ............................. 99
4.6 Discussion ............................................. 100
4.7 Limitations and Future Considerations ............... 102
4.8 Summary .............................................. 103

## 5 Conclusion

5.1 Summary of Contributions .............................. 106
5.2 Conclusions and Future Directions .................... 107

Bibliography ............................................. 110
List of Tables

2.1 Design parameters of the 12 configurations that were able to deliver all 1001 Phase II desired tactile cues. We opted to construct and evaluate configuration 4 in Section 2.4 ........................................ 18
2.2 Accuracy of the simulation in predicting the position and orientation of the contact surface. ................................................................. 20
2.3 Response characteristics during steps from the zero pose to nine locations around the fingertip. Means and standard errors of peak-to-peak amplitudes and five-percent settling times over 10 trials are reported at each location. .......................................................... 30
2.4 Maximum Oscillation Amplitude (mm) ........................................ 31
2.5 Five Percent Settling Time (s) .................................................... 31
2.6 Wire Slip (mm) .......................................................................... 31
2.7 Lookup table discretization .......................................................... 33
2.8 Errors of simulating 1,000 unreachable poses using various shrink factors to estimate leg lengths. Means and standard deviations are reported. Moving forward, we implement a shrink factor of one as this value leads to the highest simulation success rate (0.96), the smallest position error (0.83 mm), and average angular error (7.41 degrees). .......................... 35
2.9 Errors of simulating 1,000 poses in the continuous workspace using four interpolation methods. Means and standard deviations are reported. Moving forward, we implement cubic interpolation, which had the smallest mean position error (0.22 mm) and the smallest mean angular error (1.03 degrees). ....................................................... 37
3.1 Success rates for each combination of rendering method and task. A total of 80 trials were completed for each combination. A † indicates performance significantly better than chance ($p < 0.05$), as computed via a one-tailed one-sample t-test with Bonferroni correction.

3.2 Mean subjective ratings for each rendering method. Realism for each task was rated on a numeric scale from very unrealistic (1) to very realistic (5), and overall comfort was rated from very uncomfortable (1) to very comfortable (5).
List of Figures

1.1 Examples of three real-world interactions (upper panels) in which different dimensions of tactile feedback are experienced at different locations on the index fingertip (lower panels). In the left example of pushing a button, only normal force is felt, and it is applied near the tip of the finger. Holding a heavy cup brings in translational shear in addition to normal force, with contact happening on the flat area of the fingerpad. The right example consists of pinching a card between the thumb and the side of the index finger while the card is being pulled away (black arrow); this scenario causes normal force, translational shear, and rotational shear all collocated on the radial side of the index finger. .................................................. 3

2.1 A general design of a PCM for fingertip haptic applications. Coordinate frames and fixed variables are shown in black, while parameters that describe the PCM design, and must be refined, are shown in red. 12

2.2 Location of 13 desired contact positions tested in Phase I of the design refinement process. Note that the fingernail faces into the page. 15

2.3 Prototype instrumented with a fingertip tactile sensor, a six-axis force/torque sensor, and magnetic motion tracking. ................. 19

2.4 Simulated configuration and actual configuration for the same set of leg lengths. ............................................................... 21

2.5 Depiction of all 15 tactile cues, broken into normal (red) and tangential (black) components. The arrows shown have an average magnitude of 0.63 newtons .................................................. 22
2.6 BioTac data for tactile cues with a normal force in the -Z direction. Each circle includes the 19 BioTac electrodes, with red representing a bulging portion and yellow representing compression. The middle circle has no shear force, and the surrounding images are positioned in the same direction as the applied shear force. 

2.7 The Fuppeteer uses six DC motors and six Nitinol legs to control the position and orientation of a mobile platform in a 6-D workspace surrounding the user’s fingertip.

2.8 One wire-driving mechanism. The Nitinol wire is compressed between two Norprene rollers, one of which is actuated to control the wire length.

2.9 Responses to a step input both (a) without trajectory constraints and (b) with constraints of 50 mm/s and 50 mm/s$^2$. All six wire lengths were increased from 0 mm to 50 mm with the end-effector attached.

2.10 Nine destinations of step inputs, labeled in black letters, used to evaluate end-effector vibrations. The finger coordinates associated with location G are shown in red. The depicted finger is pointing in the −X direction, and the fingernail is in the +Z direction.

2.11 Response following a step from the zero pose to location F with and without velocity and acceleration constraints. End-effector position is presented as distance from the zero position.

2.12 All shear cues at a single fingertip location defined by $(r, \theta, \phi) = (6 \text{ mm}, 45 \text{ deg.}, 90 \text{ deg.})$. Each black dot represents a reachable shear cue, or a cue for which the simulation successfully estimated the required leg lengths, whereas each × indicates an unreachable shear pose. A boundary around reachable shear configurations is shown in gray. The red dot is an unreachable shear cue, and the blue dot shows the nearest point on the boundary. A similar boundary of reachable shear configurations is defined at each of the 245 fingertip locations.

2.13 Error in mm observed when driving the physical device to 1,000 randomly selected poses. The depicted spherical finger has a radius of 8 mm, but poses range between 6 mm and 10 mm from the fingertip center, as given by Table 2.7. Dots are shown at desired positions.
2.14 Contact surface poses before (a) and after (b) performing error reduction. Poses are represented by frames to capture position and orientation, with desired frames in black and measured frames in brighter colors. Blue lines link desired poses to the corresponding measured poses. We highlight an example desired pose and the corresponding measured pose before and after error reduction.

2.15 Calibration errors for 100 random offsets from the zero pose. The magnitude of positional and angular error is reported, with 1 mm and 5 degrees given equal weight.

2.16 The user study setup, including a visual guide (a) for subjects that shows the possible contact positions (letters) and shear directions (colors). The bottom images show (b) the setup for a preliminary example cue and (c) the setup with the device blocked from the user's sight.

2.17 Measured forces and moments for each of the 63 user study cues when delivered to a biomimetic finger attached to a Nano17. Colors for cues match those shown in Fig. 2.16(a), with black arrows representing location cues without shear force. For visualization, depicted forces (black, red, yellow, blue, and green arrows) have a magnitude of 2 N/mm and range from 0.1 N to 2.1 N. Depicted moments (purple and gray arrows) have a magnitude of 0.5 N·mm/mm and range from 0.5 N·mm to 4.5 N·mm.

2.18 Measured force in the +Z direction as contact is made with a biomimetic finger at location F.

2.19 Confusion matrix for commanded and perceived contact locations across all 24 subjects. Note that rounding can cause entries in a row not to sum exactly to unity.

2.20 Confusion matrices for shear cues across all 24 subjects. Only cues for which the location was perceived correctly are included. Labels are two-part combinations, with the letter signifying commanded location and the arrow signifying shear direction.

2.21 Accuracy of perceived locations and shear directions throughout the study, averaged across all participants.

2.22 Accuracy as a function of fingertip size.
3.1 Examples of the visual scene (top) and corresponding tactile cues (bottom) during virtual shape discrimination (left) and mass discrimination (right) tasks. In these examples, the wearable device is rendering 6-DOF tactile feedback to the user’s index finger; the platform orients itself to be tangent to the surface, contacts the appropriate position on the fingertip with varying normal force, and simultaneously applies tangential and rotational shear caused by the interaction taking place. ................................................................. 56

3.2 A labeled diagram of the 6-DOF tactile fingertip feedback device that we adapted for use in the human-subject study described in this chapter. ................................................................................................................. 59

3.3 The workspace for the human-subject study (not drawn to scale). The counterweight pulley system is located above the middle of the back wall of the box, and it pulls upward on the haptic device to reduce the weight that must be supported by the participant. .... 60

3.4 Using two adjustable straps to secure the fingertip gives us independent control of the fingertip’s left/right position and the strap’s tightness. Fingers of different sizes and curvatures can be positioned such that the fingertip is in a consistent location. ...................... 61

3.5 Example tactile cues that can be presented by our 1-, 3-, and 6-DOF rendering methods. Our 1-DOF method can provide contact and normal force at one fingertip location, the 3-DOF method can render contact and normal force at any fingertip location, and our 6-DOF method can provide contact, normal force, and shear force at any fingertip location. .......................... 63

3.6 The six quadric surfaces presented in the hidden surface task and the corresponding coefficients that are substituted into Eq. (3.3.1). The shorthand names for each surface are given in parenthesis and represent the curvatures in the front/back and left/right directions. Identical shapes were 3D-printed for the real-world haptic rendering method. All virtual and real surfaces are positioned such that the point at \((x, y, z) = (0 \text{ mm}, 0 \text{ mm}, 0 \text{ mm})\) is located at the same position. .......................... 65
3.7 In the slider task, the relationship between virtual distance \( d_V \) and haptic distance \( d_H \) is given by a piecewise function with four segments. 

a) When the virtual finger is far from the disk, no haptic feedback is provided. 
b) When the virtual finger presses into the disk, but a mass-dependent penetration threshold is not met, haptic normal displacement proportional to penetration is provided and the disk remains static. 
c) When the penetration threshold is exceeded, haptic normal displacement remains proportional to penetration and the disk moves with the finger. 
d) Once the virtual fingertip center penetrates the disk, maximum haptic normal displacement is rendered.

3.8 Side-by-side images of the VR and real-world setups during the human-subject study. The hidden surface task is depicted in (a) and (b), and the slider task is shown in (c) and (d). Subfigures (a) and (c) both show training scenarios in VR and physical setups for the 1-DOF, 3-DOF, or 6-DOF tactile rendering methods, whereas (b) and (d) show non-training trials and physical setups for real-world rendering. Note that the real-world haptic rendering of the slider task in (d) has no VR environment.

3.9 Example virtual interactions and the corresponding platform poses rendered via the 1-, 3-, and 6-DOF methods. The 1-DOF platform maintains a steady orientation relative to the finger during all interactions, whereas the 3-DOF and 6-DOF platforms orient themselves to match the slope of the surface. Note that the 6-DOF method also renders shear cues to the fingertip; although these shear forces can be felt, they are difficult to see in these frames.

3.10 A sequence of frames depicting a complete hidden surface test trial, with 6-DOF tactile feedback. The user first lifts the device from the calibration holder. After touching the indicated start location, the user explores the surface. Once the participant believes they know which surface is present, they select the surface from the six options, using an Oculus controller in their left hand. Finally, the participant is guided back to the calibration holder. This cycle repeats until all trials in the task are complete.
3.11 Confusion matrices of the responses given in the hidden surface task for each rendering method. Each confusion matrix represents 80 trials, but because trial types were randomized, the number of trials within each row are not equal. ........................................... 74

3.12 Confusion matrices of the responses given in the slider task for each rendering method. Each confusion matrix represents 80 trials, but because trial types were randomized, the number of trials within each row are not equal. ........................................... 74

3.13 Example trajectories observed during the two tasks. On the left, we show examples of (a) horizontal exploration and (b) poking of surface three with the 3-DOF method. On the right, we show examples of (c) more vigorous shaking and (d) precise, gentle pushing of the light slider with the 6-DOF method. ........................................... 76

3.14 Horizontal and vertical speeds and accelerations during the two tasks, organized by rendering method. ........................................... 77

3.15 Vertical speeds for each participant during the hidden surface task (left) and horizontal accelerations for each participant during the slider task (right) across all trials. ........................................... 78

4.1 Problem: the same end-effector configuration elicits different sensations for fingertips of different size, including users with (a) small, (b) medium, and (c) large fingertips. Our personalization methods aim to account for the user’s fingertip shape so as to always elicit the desired haptic sensation. ........................................... 87

4.2 Our 3-DOF fingertip device modified to house a device-mounted sensor (DS) on the mobile platform. This sensor is needed only during the data collection phase. The device is shown being worn by (a) our fingertip sensor (FS) and (b) a human user. The FS is suspended in the device such that any force exerted by the mobile platform is fully transferred into the sensor. ........................................... 88

4.3 Our fingertip-shaped sensor (FS) consists of a 3D-printed rigid core attached to an ATI Nano17 force/torque sensor on one side and embedded in a molded VytaFlex rubber fingertip-like cast on the other. 91
4.4 Block diagrams of all three discussed rendering approaches. The adjustment for data-driven personalization is given by $\hat{\mu}_D$ and the adjustment for geometric personalization is given by $\hat{\mu}_G$. In our implementation, $n = 8$, $f, f_*, d_*, m_*, \hat{m}_*, \hat{m}_* \in \mathbb{R}^6$, and $m_*, \hat{m}_* \in \mathbb{R}^3$. 91

4.5 2D device schematic (front view) of a representative configuration. The blue segment represents the static platform of the device, the green segment the mobile platform, the red disk the fingertip-shaped sensor (FS), and the yellow disk the subject’s fingertip. To match the same deformation (i.e., $d_U = d_F$), the mobile platform needs to be moved up from its position in (a). 95

4.6 The process of creating a cast of a human fingertip. We first make a mold out of Kulzer’s Flexitime Correct Flow; then we insert a rigid core and fill the void with Smooth-On VytaFlex 30 to make the fingertip cast. The rigid core is later attached to a Nano17, as in Fig. 4.3(b), creating a sensor that outputs a six-element vector of the forces and torques acting on the fingertip. 97

4.7 Objective evaluation results. Box plots of rendering errors for each combination of fingertip cast and rendering approach. For each target cue, the error is defined as the magnitude of the difference between the force component of the target sensation, $f$, and the force rendered to the test fingertip. Asterisks denote differences that are statistically significant, with $p < 0.0001$. 98

4.8 Human-subject study results. (a) Ratings for each rendering method as a function of measured finger radius. For each participant, the mean ratings for all three rendering methods are marked at the corresponding finger radius, with bars representing standard deviations. The vertical dashed line at 7.5 mm shows the FS radius. The two asterisks give the means and standard deviations of ratings for the conditions with strong visual contact and no haptic contact (V1/H0), and no visual contact with strong haptic contact (V0/H1) across all participants. (b) Boxplots for all ratings of the three rendering methods across all trials and finger sizes. Asterisks denote differences that are statistically different, with $p < 0.0001$. 101
Our daily lives are increasingly filled with virtual or augmented interactions. We shop for clothes from our computers. We communicate through texts and video calls. We live virtual lives through handheld controllers. The visual aspects of these interactions have become quite convincing, with high-resolution displays and immersive 3D environments becoming less and less distinguishable from genuine real-world visuals. However, the physical aspect of these virtual interactions is still lacking. As vibrotactile technology improves, devices increasingly provide haptic interaction to some extent; tablets provide virtual clicks to mimic physical buttons, phones announce incoming calls and messages with vibration patterns, and game controllers rumble with the terrain or an accelerated heart rate. However, the representation of haptic interactions is largely limited to vibrotactile feedback that is used to inform the user, rather than render authentic physical contacts.

Furthermore, tactile cues have traditionally been rendered without consideration of diversity among users. Virtual button-clicks on a given tablet or phone are identical for everyone, ringtones come pre-programmed, and video game controllers deliver the same vibration pattern regardless of who is holding the controller, or how they are holding it. Some level of customization may be available with respect to duration or signal strength, but the process of rendering a desired cue is typically optimized for an average user.

Meanwhile, the importance of meaningful tactile experiences has become increasingly apparent in areas such as teleoperation, education, and gaming. As the applications of rendering haptic stimuli are growing, so are the requirements of
haptic interfaces, in that the intended audience is becoming more diverse and the intended interactions are becoming more complex. For the past few decades there has been a consistent effort to improve the range and realism of haptic cues, yet typical haptic rendering continues to be one-size-fits-all and low-dimensional. To render increasingly realistic haptic cues, we must focus on both expressiveness and personalization.

1.1 Motivation

Daily physical interactions, such as flipping a switch or inserting a key into a lock, often involve complex tactile cues at the fingertips (Fig. 1.1). These soft, distal parts of the human hand are densely enervated with mechanoreceptors that enable precise perception of diverse object properties as well as competent control of contact [31]. Detailed studies of grasping and manipulation demonstrate that common tasks elicit complex haptic sensations on the fingertips. Kamakura et al. [35] classified grips that humans use to grasp common objects, finding that different objects consistently elicited contact with different parts of the index finger, including the volar, radial, and distal aspects. Atzori and Müller [2] presented the Ninapro database, which included over 20 grasping movements that again elicited normal force and shear force at multiple locations on the index finger. Feix et al. [18] compiled existing human grasp taxonomies into a single GRASP taxonomy, which divided grasps into power, intermediate, and precision categories. Power grasps tended to require the volar aspect of the index finger, intermediate grasps tended to require the radial aspect of the index finger, and precision grasps often required the distal-most tip of the index finger. Figure 1.1 illustrates a small sample of common interactions discussed in these references, also depicting the corresponding forces that act at different locations on the index finger.

Yet the haptic feedback provided by mainstream handheld devices, such as smartphones or virtual reality controllers, is typically limited to vibration [12]; these common haptic rendering approaches can catch the user’s attention and communicate the intensity of the event taking place, but they cannot apply the same contact conditions to the skin as most of the original interactions being simulated. Even within academic research, the importance of reducing the weight and encumbrance of wearable haptic interfaces means that cutting-edge devices are well-suited
Figure 1.1: Examples of three real-world interactions (upper panels) in which different dimensions of tactile feedback are experienced at different locations on the index fingertip (lower panels). In the left example of pushing a button, only normal force is felt, and it is applied near the tip of the finger. Holding a heavy cup brings in translational shear in addition to normal force, with contact happening on the flat area of the fingerpad. The right example consists of pinching a card between the thumb and the side of the index finger while the card is being pulled away (black arrow); this scenario causes normal force, translational shear, and rotational shear all collocated on the radial side of the index finger.

to render only particular haptic interactions, rather than a more general set that ideally approaches all haptic interactions. To the best of our knowledge, no existing wearable haptic device can render all three of the common interactions portrayed in Fig. 1.1 even somewhat authentically. This thesis aims to address the lack of expressiveness demonstrated by existing wearable fingertip haptic device through the mechanical design of a novel 6-DOF fingertip device.

In addition to bounds on expressiveness, mainstream and lab-based haptic devices are also limited by the difficulty in designing fingertip interfaces that fit all users. Many studies have highlighted large differences in the fingertip’s size across the human population. Dandekar et al. [15] assert that the human index fingertip varies between 16 and 20 mm in width, while Johnson and Blackstone [33] registered a width of 20.3 ± 2.4 mm (mean ± standard deviation) for the same finger
across participants. Gender, race, and age all play a role in causing these differences [13, 54, 27]. Moreover, the five fingertips on a single hand vary greatly in size and shape. These physical differences between fingertips have the potential to distort the perception of haptic feedback provided by fingertip haptic devices. Indeed, touch is highly sensitive to variations in contact stimulation, such as small changes of the contact force during static [87] and dynamic touch [24]. However, traditionally, haptic rendering algorithms have been designed without accounting for finger size, incorrectly assuming that the provided sensations would be (roughly) consistent across users.

A few approaches partially take this important variable into account by implementing closed-loop control via a force or pressure sensor. For example, the wearable fingertip device of Prattichizzo et al. [66] used three force-sensing resistors (FSRs) to register the contact sensations applied by the end-effector onto the finger skin. By closing a force-control loop around these sensor readings, this device was able to adapt to fingertips of different shapes after contact was initiated. Leonardis et al. [47] placed an OptoForce three-axis optical force sensor on the end-effector of their wearable fingertip device. However, they used it only to characterize the force-indentation response of the device, and they did not look into personalization for individual users. Others have used open-loop force output to render appropriate sensations to differently sized fingertips. For example, Solazzi et al. [74] presented a 3-DOF device that uses two position-controlled actuators to locate a platform around the fingertip and one force-controlled voice coil actuator to render forces into the fingertip. The current commanded to the voice coil was proportional to the force acting on a finger avatar moving in a virtual environment. Similarly, Khurshid et al. [37] rendered one-dimensional haptic feedback to the operator’s fingertip via a voice coil whose current was proportional to the force experienced at the robot’s side.

Although both force-feedback approaches address personalization, each comes with limitations. Closed-loop force control inevitably makes the overall system bulkier, as a sensor must be included on the device. Furthermore, force control enters into play only after the end-effector contacts the skin, and it is thus not able to control the position and orientation of the end-effector at the moment of contact, nor the precise timing of initial contact. Finally, both approaches require an interpretable desired force that some tactile sensors, such as the SynTouch Bio-
Tac, do not directly provide [86]. Given the high variability of human fingertip characteristics and the human tactile sensitivity to contact sensations, we strongly believe the unique characteristics of each fingertip should be taken into account when designing fingertip interfaces and rendering algorithms. This thesis proposes two methods to overcome these limitations and personalize the haptic rendering algorithm of a fingertip haptic device for fingertips of different size.

In this thesis, we present our work towards rendering more expressive and personalized fingertip tactile cues. First we present a six-degree-of-freedom (6-DOF) wearable fingertip device, followed by a framework for using this device to evaluate how tactile fingertip feedback dimensionality affects user performance and exploratory motions. We then turn our focus to expressing individualized fingertip cues through two generalizable software-based adaptations of an existing system.

1.2 Thesis Overview

Following this general introduction to tactile fingertip feedback and wearable devices, we now outline the chapters of this thesis, which aim to provide tested hardware and software-based solutions for rendering more expressive and personalized tactile cues.

In Chapter 2, we first investigate the potential of displaying fingertip haptic sensations with a 6-DOF parallel continuum manipulator (PCM) that mounts to the user’s index finger and moves a contact platform around the fingertip. We define the design space of 6-DOF parallel continuum manipulators and outline a process for refining such a device for fingertip haptic applications. Following extensive simulation, we construct a manually actuated prototype of one suitable design and demonstrate the range of deliverable fingertip tactile cues, including a normal force into the finger and shear forces tangent to the finger at three extreme points on the boundary of the fingertip. After verifying the potential of this design, we present a motorized version named the Fingertip Puppeteer, or Fuppeteer for short, that is capable of controlling the position and orientation of a flat platform, such that any combination of normal and shear force can be delivered at any location on any human fingertip. After creating a six-dimensional lookup table and adjusting simulated inputs using measured Jacobians, we show that the device can make contact with all parts of the fingertip with a mean error of 1.42 mm. Finally we present re-
results from a human-subject study in which twenty-four users discerned nine evenly distributed contact locations with an average accuracy of 80.5%. Translational and rotational shear cues were identified reasonably well near the center of the fingertip and more poorly around the edges.

After demonstrating our 6-DOF fingertip tactile device in Chapter 2, we continue by investigating how the expressiveness of cutaneous fingertip feedback affects user movements and virtual object recognition in Chapter 3. We present a system in which the Fuppeteer is combined with motion tracking, a head-mounted display, and novel contact-rendering algorithms to enable a user to tactiley explore immersive virtual environments. We then evaluate rudimentary 1-DOF, moderate 3-DOF, and complex 6-DOF tactile feedback during shape discrimination and mass discrimination, also comparing to interactions with real objects. Results from 20 naive study participants show that higher-dimensional tactile feedback may indeed allow completion of a wider range of virtual tasks, but that feedback dimensionality surprisingly does not greatly affect the exploratory techniques employed by the user.

Although Chapter 3 demonstrates a benefit of higher-dimensional tactile feedback in rendering a assortment of interactions with virtual objects, it also highlights that in some tasks, 6-DOF tactile cues are not needed, and the associated weight and encumbrance are therefore disadvantageous. This argument is further supported by additional research showing that some tactile interactions can be effectively rendered using relatively few degrees of freedom. In Chapter 4, we leverage this insight to focus on improving lower-dimensional tactile feedback through personalization. This chapter starts with an existing data-driven haptic rendering algorithm that ignores fingertip size, and it then develops two generalizable software-based approaches to personalize this algorithm for fingertips of different sizes using either additional data or geometry. We evaluate our algorithms in the rendering of pre-recorded tactile sensations onto rubber casts of six different fingertips as well as onto the real fingertips of 13 human participants. Results on the casts show that both approaches significantly improve performance, reducing force error magnitudes by an average of 78% with respect to the standard non-personalized rendering technique. Congruent results were obtained for real fingertips, with subjects rating each of the two personalized rendering techniques significantly better than the standard non-personalized method.
Finally, Chapter 5 concludes this thesis by summarizing my contributions and presenting possible continuations of this work.
Chapter 2

A Wearable Device for Delivering 6-DOF Fingertip Haptic Cues

The field of fingertip haptics is growing rapidly. The past decade alone has seen the rise of using electrovibration [3, 39] and the squeeze film effect [92] to simulate texture and friction on touch screens, the invention of ways to use lasers [44] and focused ultrasonic waves [91, 8, 28] to display tactile sensations to the unadorned hand in mid-air, and many new designs and applications of wearable fingertip robotic devices. Even when we restrict our discussion to delivering haptic cues to the fingertip, we see applications ranging from surgery [50, 58] and education [80, 62, 34], to better understanding how humans perceive and respond to fingertip tactile sensations [71, 69]. Widespread excitement surrounding virtual reality has further fueled the need for wearable fingertip haptic devices that enhance user presence in virtual environments [4], as well as simpler hardware designed for reaching the masses, such as devices made by Foldaway Haptics [7], Tactical Haptics [1], and Go Touch VR [84]. However, no existing wearable device can move a contact surface around the fingertip with six degrees of freedom (6 DOF), thus providing the entire set of fingertip cues a person may experience when interacting with a real surface.

High-dimensional cues may be beneficial for more complex interactions with virtual objects. For example, picking up a water bottle by pinching its base requires rotational shear, translational shear and normal forces at whichever part of each fingertip is making contact. Moving along the surface of the fingertip is similar to applying shear in the same direction locally, but for larger-scale shear and slip,
unique degrees of freedom are required, so that the orientation of the contact platform can be controlled independently of its position. To realistically simulate all possible ways in which a fingertip may interact with a surface, we argue that a fingertip haptic device must be able to control the position and orientation of a contact surface in six-dimensional (6D) space. To demonstrate this requirement, we can relate the required six degrees of freedom (DOFs) to the tactile sensations we would expect to experience: one DOF for how hard we press into the surface (making contact and normal force), two DOFs for rotating the finger such that a different location on the fingertip contacts the surface, two DOFs for sliding the finger along the surface (translational shear), and one DOF for twisting the finger on the surface (rotational shear).

This chapter presents the Fuppeteer (short for Fingertip Puppeteer), which is a motorized 6-DOF wearable fingertip haptic device that is based on a parallel continuum manipulator (PCM), as opposed to rigid links and discrete joints. We first present a manually actuated prototype to defend our proposal, followed by the design and evaluation of a motorized version. This chapter presents work that has been published in the Proceedings of the IEEE World Haptics Conference in 2017 [97] and the IEEE Transactions on Haptics in 2019 [98].

2.1 Related Work

Wearable one-degree-of-freedom (1-DOF) haptic devices have been presented for rendering contact location [68], tangential skin stretch [77] and normal indentation into the fingerpad [84]. Some researchers have combined multiple 1-DOF devices into one system to render more complex interactions, such as using one voice coil on each finger to enhance object manipulation [45], using up to 12 asymmetric actuators to render 6D directional cues to the hand [14], or using a single motor to move an electrode array around the finger to present both skin stretch and pressure distribution with high spatial resolution [95]. This idea has also been extended to the design of various haptic hand exoskeletons and gloves that apply 1D haptic feedback to each fingertip, e.g., [78, 49, 23].

The field of wearable haptics has also fostered the development of devices with more degrees of freedom. Two-DOF fingertip devices have provided 1D kinesthetic force feedback with 1D vibrotactile feedback [11], 1D tangential shear with 1D
normal displacement [88, 82, 51], and 2D tangential shear [22, 81, 41, 85]. Two
2-DOF tangential shear devices have been worn on the index finger and thumb to
successfully communicate 5-DOF directional cues [25]. Three-DOF fingertip devices
have demonstrated the ability to move a contact element along a spherical surface
around the fingertip and translate in the direction normal to the sphere [75, 20], tilt
a contact surface in place with 2 DOF and then extend into the volar side of the
fingertip [63, 4], or provide 3-DOF feedback that couples orientation and position in
a more complex manner [66, 46]. At least one design has offered 3-DOF translation
of a contact element with a fixed orientation to render more precise shear cues [72],
and another has presented 3-DOF translation via asymmetric vibrations from three
orthogonal actuators [38]. We refer the reader to a review by Pacchierotti et al. [59]
for more information on the breadth of wearable haptic systems for the fingertip
and hand.

All six DOF have been represented in smaller subsets, so the limiting factor
to providing 6-DOF fingertip feedback seems to be the additional volume, weight,
and complexity of combining all six degrees of freedom in one device. We propose
parallel continuum manipulators as a means of providing six degrees of freedom in
a design that is sufficiently compact and lightweight to be mounted directly to the
user’s finger.

2.1.1 Parallel Continuum Manipulators

A PCM consists of six parallel, compliant legs that pass through a fixed base plat-
form and attach to the end-effector, or distal platform. The end-effector’s pose can
be controlled by independently adjusting the six leg lengths (the length of wire be-
tween the two platforms) via six motors located below the base platform, allowing
the mobile portion of the robot to remain lightweight. The early work in modeling
and evaluating the potential of PCMs has come primarily from the REACH Lab at
the University of Tennessee, Knoxville. Bryson, Till and Rucker have shown that
Cosserat rod theory is an effective modeling tool for PCMs, due to the elastic nature
of the parallel legs [6, 70, 79]. This group has also demonstrated that independently
changing the length of each of the six parallel legs modifies the position and ori-
entation of the end-effector platform with positional errors of less than 3% of the
average leg length [6]. Since the math behind modeling PCMs has been thoroughly
discussed in published papers, we will highlight the essential concepts and direct the reader to [6, 70] for a more thorough explanation.

Cosserat rod theory allows the state of an entire compliant rod to be modeled given the position $p \in \mathbb{R}^3$, orientation $R \in SO(3)$, internal force $n \in \mathbb{R}^3$ and internal moment $m \in \mathbb{R}^3$ at one end of the rod, along with any external forces $f \in \mathbb{R}^3$ and moments $l \in \mathbb{R}^3$ distributed along the rod. When we combine six compliant rods for a PCM, we mechanically define the position and orientation of each rod at the base platform, leaving $n_x, n_y, n_z, m_x, m_y, m_z$ as six unknowns at the base of each rod. To solve the forward kinematics, we must also solve for the position and orientation of the end-effector (six unknowns), while for the inverse kinematics we must solve for the length of each rod (also six unknowns). This results in 42 unknowns for an entire PCM for either forward or inverse kinematics. At static equilibrium, the sum of the forces and the sum of the moments exerted on the distal platform by the six rods must equal zero, giving six constraints. The position and orientation of each rod are fixed with respect to the end-effector platform, giving an additional six constraints per rod. Altogether the setup gives 42 constraints, allowing all 42 unknowns to be found.

Early numerical computation methods described by Bryson and Rucker showed both forward and inverse kinematics could be solved by guessing the unknowns, integrating along the rods following Cosserat rod mechanics, finding the residual error, and repeating the process until a desired tolerance was met [6]. Recently, Till, Bryson and Rucker improved upon the technique: by using a more efficiently calculated Jacobian, implementing an adaptive Levenberg-Marquardt algorithm to update guesses, and implementing the code in C++, the group was able to improve speeds from roughly 1 Hz to several kHz [79]. The C++ code developed by Till, Bryson and Rucker was adapted for use in the simulations discussed in this chapter.

The advantages of PCMs, including being strong, dexterous, and compact, suggest there is a large potential for adapting them to haptics applications, but implementing this unique design comes with unique challenges. Thin, smooth wires must be extended and retracted quickly without slipping, the end-effector’s stiffness depends on configuration, and a balance must be found between movement speed and the unwanted oscillations that are inherent to a compliant device. We first validated the potential of using a PCM as a fingertip haptic device via simulation and a manually actuated prototype, and then we constructed a motorized version
Figure 2.1: A general design of a PCM for fingertip haptic applications. Coordinate frames and fixed variables are shown in black, while parameters that describe the PCM design, and must be refined, are shown in red.

for use in human perceptual studies.

2.2 Design and Simulation Parameters

2.2.1 Design Space Parameterization

We first define the parameters that could vary in a general parallel continuum manipulator design. Fig. 2.1 shows an implementation of a PCM as a fingertip haptic device, with the manipulator located behind the finger. A rigid contact surface attached to the distal platform reaches around the finger to make contact at the desired fingertip location. Superscripts denote the frame, such as the finger (f), base (b), end-effector (e), or contact surface (c) frames, and subscripts identify the measurement. As discussed in Chapter 1, the region of interest on the fingertip is the entire section with positive \( x^f \) and \( z^f \) coordinates in Fig. 2.1, which we model as a quadrant of a sphere with radius \( r_f \). Although this geometric simplification appears to work well for comparing designs against each other, we expect that displaying accurate tactile sensations will require an improved finger model and
potentially individualized calibration to account for variance in finger shapes and sizes.

Twelve coordinates are needed to fully define the setup: six coordinates describing the pose of the base \((b)\) with respect to the finger frame \((f)\), and six coordinates describing the pose of the contact point \((c)\) with respect to the end-effector frame \((e)\). Since we want to be able to deliver the same tactile cues to the left and right sides of the fingertip, we require the device to be positioned such that it is symmetric over the \(x^f z^f\)-plane of the finger, reducing our final parameters to \(\{x_b^f, z_b^f, \theta_y^b, \theta_z^b\}\) for defining the base’s pose in the finger frame and \(\{x_c^e, z_c^e, \theta_{cy}^e\}\) for defining the contact surface’s pose in the end-effector frame. Fig. 2.1 also depicts the three parameters of leg radius, \(r_l\), the radius of the circle on which the holes for the legs are located, \(r_h\), and the larger of the two angles between consecutive holes, \(\theta_{major}\), resulting in a total of 10 parameters to refine:

\[
\{x_b^f, z_b^f, \theta_y^b, \theta_z^b, x_c^e, z_c^e, \theta_{cy}^e, \theta_{cy}^e, r_l, r_h, \theta_{major}\}
\]

Fig. 2.1 shows that we assume both the distal and proximal platforms to be radially symmetric, with sets of guide holes repeating every \(120^\circ\) on a circular arc. The leg guide holes are symmetric about the \(x^f z^f\)-plane for only two unique orientations of the base, which we define as \(\theta_z^b = \pm 30^\circ\). To minimize the weight and size of the device, we set the maximum hole pattern radius, \(r_h\), to 16 mm, or about twice the radius of a finger. Additionally, no two consecutive holes for legs were allowed to be less than \(10^\circ\) apart, to maintain the structural integrity of the end-effector platform.

### 2.2.2 Calculation of Simulator Inputs

To evaluate fingertip PCM designs, we first define the set of tactile cues we want to be able to display to the fingertip. Each desired cue is represented by a point of contact on the fingertip and a force vector associated with both normal and shear force. The position and orientation of the contact surface required for the desired haptic cue are denoted \(p^f_c\) and \(R^f_c\) respectively, where \(p^m_n\) and \(R^m_n\) denote the position and orientation of frame \(m\) in frame \(n\).

In order to determine if each desired tactile cue is displayable, we must convert \(p^f_c\) and \(R^f_c\) to the corresponding position and orientation of the end-effector with
respect to the base, $p_b^e$ and $R_b^e$, as these are the values required by the PCM simulator. Using the frames shown in Fig. 2.1, we see that for any desired robot configuration the required orientation and position of the end-effector with respect to the base are given by:

$$R_b^e = (R_b^f)^T R_e^f (R_e^c)^T \ (2.2.1)$$

$$p_b^e = (R_b^f)^T (p_c^f - p_b^f) - R_b^e p_c^e \ (2.2.2)$$

where $R_b^f$, $R_e^c$, $p_b^f$, and $p_c^e$ are given by design parameters.

To simulate the fingertip interaction force associated with the desired tactile cue, it must similarly be represented in the base frame as $F_b$. Given our defined $F^f$, we see:

$$F_b = R_b^f F^f \ (2.2.3)$$

$$M_b = (R_b^e p_e^c) \times F_b \ (2.2.4)$$

Equations (2.2.1)-(2.2.4) give us the four inputs needed to run the simulation: $R_b^e$, $p_b^e$, $F_b$, and $M_b$. If the simulation converges to a solution and no legs interfere with the virtual finger, the design is deemed capable of displaying the desired cue. If a maximum number of iterations, $i_{\text{max}} = 200$, is reached, we record the desired tactile cue as unreachable and the simulation continues on to evaluate the next desired tactile stimulus.

### 2.3 Design Refinement

Due to the high dimensionality of the design space, the refinement process was partitioned into two phases. Phase I involved simulating many designs with a sparse set of desired tactile cues to quickly determine appropriate bounds for parameters. In order to iterate through as many designs as possible, Phase I assessed only the unloaded workspace of each design. Phase II involved simulating the top-tier designs from Phase I on a more encompassing set of desired tactile cues, including both normal and tangential loads.
2.3.1 Phase I of Design Refinement

To begin Phase I, we defined an initial design space $D_0$ as all combinations of parameters chosen from partitioned subsets within our initial guesses for parameter bounds. Due to symmetry of the finger and device about the $x'z'$-plane, each design within $D_0$ was simulated over the 13 desired contact positions shown in Fig. 2.2. After all of the designs in $D_0$ were simulated, the results were analyzed with respect to each parameter. If the success rate with respect to parameter $k_m$ appeared to be limited by parameter $k_n$, then the bounds of parameter $k_n$ were expanded. The adjusted parameter bounds formed a new design space $D_1$, and the process was repeated until the optimal parameters were largely contained within, rather than on, the parameter bounds.

The initial design space $D_0$ included 26,244 designs. The best design in $D_0$ was able to reach only 8 of the 13 desired locations, or just over 60%. Additionally, simulation of $D_0$ showed that leg radius, $r_l$, had little effect on the success rate of a given design; holding all other parameters constant, changing the leg radius had zero effect in 97.6% of the designs, and at most resulted in one more or one fewer desired configuration being reached. Thus the leg radius was held constant at 0.0635 mm until Phase II, where the addition of desired loads was expected to cause greater disparity between different leg radii. The results with respect to other parameters were more evenly dispersed, and the parameters for the following design space were adjusted to position the parameters of the best configurations within parameter
bounds.

After five iterations of simulating and adjusting the design space $D$, over 180,000 robot configurations had been tested. The final design space, $D_5$, contained 16 designs that reached all 13 positions and 270 designs that reached 12 positions, giving 286 “top-tier” designs that could reach at least 90% of the Phase I desired workspace. The success of the designs in $D_5$ shows that using the simulation to refine the design space allowed us to obtain designs much more suited to fingertip haptics applications than those defined by our intuitive parameter guesses in $D_0$.

### 2.3.2 Phase II of Design Refinement

For Phase II the simulation was expanded to include 91 desired fingertip contact positions by reducing $\alpha$ and $\beta$ in Fig. 2.2 from 30 degrees to 10 degrees. Three normal forces (0 N, 0.5 N, 1 N) were tested at every position, and every normal force (except zero normal force) was simulated with five shear force values consisting of 0 N and a maximum shear force in each of four directions tangent to the finger. Thus Phase II simulated a total of 1001 tactile stimuli for each design: 91 desired contact positions, each at 11 loading conditions. Maximum shear force, $F_s$, is given by $F_s = \mu_s F_N$, where $\mu_s$ is the coefficient of static friction between the contact surface and the finger and $F_N$ is normal force.

The coefficient $\mu_s$ depends on the contact surface material, which had not yet been decided. We expected to use a smooth and common material, avoiding textured, sticky, or slippery materials, such that the device could realistically simulate contact with a wide range of virtual surfaces. For simulation, $\mu_s$ was assigned to be 1, which is within the range of expected values for various materials sliding on a human finger, including rubber, steel, and acrylic [48]. In the future, adjusting the coefficient of friction may allow different tactile sensations to be more effectively rendered. For example, a lower-friction contact surface may be ideal for rendering interactions with large amounts of skin slip, whereas a higher-friction contact surface would allow more shear force to be rendered at a given normal force.

Fingertips are deformable, so the inclusion of force requires the desired contact position to be updated as well. Combining the methods of Gulati and Srinivasan [26] and Serina et al. [73] led us to approximate displacement into the finger as the
positive root of:

\[ F_N = (0.05 \text{ N/mm}^3)x_N^3 + (0.2 \text{ N/mm}^2)x_N^2 + (0.002 \text{ N/mm})x_N \]  

(2.3.1)

The work of Wiertlewski and Hayward [89] and Nakazawa et al. [53] led us to approximate displacement in the tangential direction as:

\[ x_s = \frac{F_s}{0.5 \text{ N/mm}} \]  

(2.3.2)

We use Equations (2.3.1) and (2.3.2) to update the desired position of the contact surface, then reevaluate Equations (2.2.2), (2.2.3), and (2.2.4) to give the inputs needed for the simulation.

For Phase II the leg radius variable was reintroduced, as we expected the inclusion of forces and moments in the simulation to cause more variation in the results with respect to leg properties. The 286 top-tier designs from Phase I were tested with different leg radii until the optimal designs were contained within the range of tested values. As expected, not all of the 286 designs were able to display all of the desired tactile cues in Phase II. However, 12 designs had success rates of 100 percent, indicating strong potential for future application to fingertip haptics. Table 2.1 shows the parameters of designs able to reach all 1001 of the desired poses in Phase II. Note that while parameters are clustered within a range of values, there is still reasonable variation in nearly all parameters, giving an assortment of optimal designs for fingertip haptic applications. Although the refinement process did not lead to a clear single selection of design parameters, the variety gives us more flexibility in choosing parameters that may facilitate a specific application or mechanical design process.

### 2.4 Manually Actuated Prototype

Prior to designing a motorized prototype, we constructed a preliminary manually actuated prototype to both validate the simulation and offer physical evidence that the refined design is capable of displaying the full range of desired tactile cues.
Table 2.1: Design parameters of the 12 configurations that were able to deliver all 1001 Phase II desired tactile cues. We opted to construct and evaluate configuration 4 in Section 2.4.

<table>
<thead>
<tr>
<th>Config. Num.</th>
<th>$x^f_b$</th>
<th>$z^f_b$</th>
<th>$\theta^h_{y}$</th>
<th>$\theta^h_{z}$</th>
<th>$x^e_c$</th>
<th>$z^e_c$</th>
<th>$\theta^e_{c,y}$</th>
<th>$r_l$</th>
<th>$\theta_{maj}$</th>
<th>$r_h$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>-32</td>
<td>30</td>
<td>-30</td>
<td>14</td>
<td>-26</td>
<td>60</td>
<td>0.1143</td>
<td>110</td>
<td>12</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>-32</td>
<td>30</td>
<td>-30</td>
<td>14</td>
<td>-26</td>
<td>60</td>
<td>0.1905</td>
<td>110</td>
<td>12</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>-38</td>
<td>30</td>
<td>-30</td>
<td>14</td>
<td>-32</td>
<td>60</td>
<td>0.1143</td>
<td>110</td>
<td>16</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>-38</td>
<td>30</td>
<td>-30</td>
<td>14</td>
<td>-32</td>
<td>60</td>
<td>0.1905</td>
<td>110</td>
<td>16</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>-44</td>
<td>40</td>
<td>-30</td>
<td>14</td>
<td>-32</td>
<td>60</td>
<td>0.1143</td>
<td>110</td>
<td>16</td>
</tr>
<tr>
<td>6</td>
<td>-6</td>
<td>-44</td>
<td>30</td>
<td>-30</td>
<td>14</td>
<td>-38</td>
<td>60</td>
<td>0.1143</td>
<td>110</td>
<td>16</td>
</tr>
<tr>
<td>7</td>
<td>-12</td>
<td>-38</td>
<td>20</td>
<td>-30</td>
<td>20</td>
<td>-32</td>
<td>70</td>
<td>0.1905</td>
<td>110</td>
<td>16</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>-38</td>
<td>30</td>
<td>-30</td>
<td>14</td>
<td>-32</td>
<td>60</td>
<td>0.1524</td>
<td>110</td>
<td>16</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>-44</td>
<td>40</td>
<td>-30</td>
<td>14</td>
<td>-32</td>
<td>60</td>
<td>0.1524</td>
<td>110</td>
<td>16</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>-32</td>
<td>30</td>
<td>-30</td>
<td>14</td>
<td>-26</td>
<td>60</td>
<td>0.1524</td>
<td>110</td>
<td>16</td>
</tr>
<tr>
<td>11</td>
<td>-12</td>
<td>-38</td>
<td>20</td>
<td>-30</td>
<td>20</td>
<td>-32</td>
<td>70</td>
<td>0.1524</td>
<td>110</td>
<td>16</td>
</tr>
<tr>
<td>12</td>
<td>-6</td>
<td>-44</td>
<td>30</td>
<td>-30</td>
<td>14</td>
<td>-38</td>
<td>60</td>
<td>0.1524</td>
<td>110</td>
<td>16</td>
</tr>
</tbody>
</table>

2.4.1 Prototype Design and Setup

As discussed in Section IV, the final output of the design refinement process was 12 sets of optimal design parameters. To build a prototype, we first had to choose one parameter set to implement. Some parameters were chosen to ease the construction process, such as having the largest value for $r_h$ and $r_l$, while others were chosen to roughly represent the middle of their ranges. This process led us to choose configuration 4, shown in Table 2.1, as the basis for our prototype.

Fig. 2.3 shows the constructed prototype. The majority of the prototype was laser-cut out of 1.5-mm-thick acrylic, allowing the end-effector to be lightweight and compact. The legs are made of Nitinol and have a radius of 0.1905 mm, Young’s modulus of 30.0 GPa, and shear modulus of 11.5 GPa. Note that while Nitinol is commonly used for its interesting shape-memory properties, we are using Nitinol purely for its superelasticity. The distal and proximal platforms have guide holes that are spaced in alternating increments of $10^\circ$ and $110^\circ$ on a circle of radius 16 mm. As shown in the figure, one end of every leg is rigidly attached to the distal platform. The legs pass freely through the base and are then fastened to independent guides.
behind the base platform. The legs are held in place at both ends via 000-120 bolts, which prevent any changes in the legs’ orientations and positions relative to their housings at both ends. The guides are free to slide along rigid rails independent of each other and can be secured in place via 0-80 bolts to ease the process of measuring the distal platform’s position and orientation. The guide rails have laser-cut lines spaced 1 mm apart to assist positioning, but inaccuracies in manually extending and retracting the wires leave the overall positional error of each leg at roughly 1 mm. Feedback from encoders allows us to reduce this positional error in the motorized version presented in Section 2.5.

After selecting the design parameters and designing the overall device, the construction process was simple and straightforward. The highlighted design parameters allowed us to create a sufficiently strong and lightweight end-effector out of laser-cut acrylic and 3D-printed ABS, but we expect that using stronger materials and more precise machining or cutting techniques may allow more compact designs in the future, as described in Section 2.9.

For experimental validation we used a SynTouch BioTac sensor to mimic the size and mechanical properties of a human fingertip, an ATI Nano17 six-axis force/torque sensor to measure forces and torques applied to the BioTac, and a TrakSTAR 3D electromagnetic tracking system to measure the position and orientation of the
Table 2.2: Accuracy of the simulation in predicting the position and orientation of the contact surface.

<table>
<thead>
<tr>
<th>Simulated Position (mm)</th>
<th>Measured Position (mm)</th>
<th>Positional Error (mm)</th>
<th>Simulated Normal Direction</th>
<th>Measured Normal Direction</th>
<th>Angular Error (deg)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(7, 0, 0)</td>
<td>(6.4, -1.6, -2.5)</td>
<td>(-0.6, -1.6, -2.5)</td>
<td>(-1, 0, 0)</td>
<td>(-1.0, -0.0, 0.1)</td>
<td>4.6</td>
</tr>
<tr>
<td>(0, 7, 0)</td>
<td>(5.5, 7.4, -3.9)</td>
<td>(5.5, 0.4, -3.9)</td>
<td>(0, -1, 0)</td>
<td>(0.1, -1.0, 0.0)</td>
<td>3.3</td>
</tr>
<tr>
<td>(0, -7, 0)</td>
<td>(6.2, -6.9, -0.8)</td>
<td>(6.2, 0.1, -0.8)</td>
<td>(0, 1, 0)</td>
<td>(-0.1, 1.0, 0.2)</td>
<td>9.5</td>
</tr>
<tr>
<td>(0, 0, -7)</td>
<td>(6.4, 0.7, -7.5)</td>
<td>(6.4, 0.7, -0.5)</td>
<td>(0, 0, 1)</td>
<td>(0.1, -0.1, 1.0)</td>
<td>5.6</td>
</tr>
</tbody>
</table>

end-effector relative to the BioTac. Fig. 2.3 shows the complete setup.

2.4.2 Simulation Validation

Once we had a manually driven prototype, we first evaluated the accuracy of the simulation used for design refinement. We selected four points along the boundary of the fingertip and simulated the chosen design to find the leg lengths needed to drive the contact surface to each point, at an orientation tangent to the skin. We then manually set the leg lengths of our prototype to the values output by the simulation. The TrakSTAR allows us to measure the position and orientation of the end-effector with respect to the BioTac fingertip, so that we can calculate the error between the simulated and measured values. Our results are shown in Table 2.2.

Looking at the first row of Table 2.2, we see that when the desired contact is at the tip of the finger, the positional error between the simulation and the prototype is relatively small, with a magnitude of 3 mm. The other three poses have slightly larger positional errors, with the prototype’s contact surface consistently having an offset in the +X direction. Fig. 2.4 shows both the simulated model making contact with the base of the finger and the physical prototype after we manually set the leg lengths to the predicted values. We notice that the outer-most legs in the simulation appear to bend more than their counterparts on the physical prototype, contributing to the measurement discrepancy.

Potential sources of error include the relatively low resolution of the prototype’s leg lengths (±1 mm) and inaccuracies in the legs’ material properties, but it is more likely a result of the manner in which the legs are attached to the robot.
Our simulation assumes that both ends of each leg are rigidly attached such that their position and orientation at each platform is fixed. However, the base of each leg is truly fixed to a guide, which can be more than 100 mm away from the base platform. The distance between the base platform and the guide gives a segment of rod with nontrivial length that is free to rotate about its centerline, but which our model assumes is rigid. Additionally, both ends of the legs are secured via 000-120 bolts. The process of tightening a bolt into each leg may have induced a nontrivial internal moment. If present, this moment would propagate down the entire leg and cause the leg to bend in unpredictable ways, especially at these extreme poses. For the motorized prototype presented in Section 2.5, we adjust the simulation and our design such that only one end of each leg is rigidly fixed, in order to minimize any discrepancy between simulated and actualized internal moment for each leg.

Despite this difference between the simulation and the prototype, the prototype’s contact surface was in the same orientation and general position for all four tested poses, suggesting that although the simulation does not perfectly predict the contact surface’s pose, it can provide a good first guess for a control loop.
2.4.3 Evaluation of Displayable Tactile Cues

We also used the setup in Fig. 2.3 to validate that the refined design is able to display the full range of desired tactile cues. As discussed in Section 2.3, the set of desired contact positions on the finger forms a spherical octant, the boundaries of which can be described by three points on the fingertip: one at the end of the finger, one at the bottom, and one at the side, as shown in Fig. 2.5. To haptically present the interactions discussed in Chapter 1 and shown in Fig. 1.1, we want to render normal force and shear force in any direction at each of these three points.

Once the device makes contact with the fingertip, the application of normal and shear force to the finger requires changes in leg lengths less than 1 mm, which is finer than the stated resolution of our manually driven prototype. We do not attempt to convince the reader that this manual position control is sufficient for displaying reliable tactile cues, but rather that the device is physically capable of delivering all desired tactile cues, suggesting that a motorized version would be equally capable once a proper control scheme is developed. To this end, for each of the desired tactile cues, we viewed live measurements of the force vector exerted on the BioTac fingertip (via the Nano17) and manually adjusted the leg lengths of each rod until the approximate desired force vector was achieved. We then locked the legs into place and recorded data from the BioTac, Nano17, and TrakSTAR. Fig. 2.5 shows all
results from the Nano17, and Fig. 2.6 shows example results from the BioTac. The forces in Fig. 2.5 are broken into normal and tangential components, as determined by the normal vector of the contact surface. These results demonstrate that the prototype is able to display normal force into the finger and shear force tangent to the finger in all directions at each of the three tested locations. This preliminary work provided sufficient motivation for the development of a motorized 6-DOF parallel continuum manipulator for rendering fingertip haptic cues.

2.5 Motorized Prototype Design

The mechanical design of our motorized device, shown in Fig. 2.7, satisfies the following key design requirements: the entire fingertip is exposed so that it can make contact with the end-effector, the six Nitinol wires can be independently controlled, and each wire has the necessary range of motion. During the design refinement process, simulation showed that each leg needed to extend up to 150 mm beyond its starting length in order to deliver all of the desired tactile cues. To accommodate the relatively high stiffness of the Nitinol wires and the range of motion required for each leg, one end of each Nitinol wire is rigidly attached to the end-effector,
Figure 2.7: The Fuppeteer uses six DC motors and six Nitinol legs to control the position and orientation of a mobile platform in a 6-D workspace surrounding the user’s fingertip.

and the other is compressed between two Norprene rollers, as shown in Fig. 2.8. In simulation, we now represent the Nitinol wire as fully constrained at the end-effector and free to rotate about the axial direction at the rollers, since each wire can spin as the rollers move if an internal moment exists. The device uses six Faulhaber 1524SR brushed DC motors, each with a 22:1 gearhead reduction ratio and a 16 CPR encoder. Each motor has a maximum continuous torque of 2.9 mNm and a maximum continuous current of 0.28 A. The rollers on the motor axles have an outer radius of 6.35 mm, giving a final continuous force output of 10 N, well above the 1 N shown to be necessary for each leg to deliver the desired tactile cues in simulation. A laser-cut housing holds the six motors and driving mechanisms. The motorized device has a total mass of 360 g; the six motors are 240 g, the housing is 110 g, and the mobile end-effector is only 10 g. The housing and motors fit within a volume of dimension 55 mm \times 100 mm \times 100 mm. The mass and volume of the complete device are admittedly large for a fingertip device, but the goal of this design was to explore the potential of a PCM-based approach for delivering fingertip cues. Additionally, the mobile portion of the device accounts for under 3% of the total mass, and there is room for greatly reducing the overall mass with an
Figure 2.8: One wire-driving mechanism. The Nitinol wire is compressed between two Norprene rollers, one of which is actuated to control the wire length.

optimized design and more compact motors, as discussed in Section 2.9.

When evaluating our manually actuated prototype, we noticed that the Nitinol wires often deflected between their respective actuation points and the base platform, which in turn caused errors in the leg lengths and likely affected the internal forces and moments of the Nitinol legs as well. The motorized device remedies this problem by passing the Nitinol through stainless-steel tubing with an inner diameter roughly 0.2 mm larger than the diameter of the Nitinol, thus preventing deflections while causing negligible sliding friction.

During early testing of the motorized device, we noticed a drift in the measured pose of the contact surface after repeated motions. We determined two causes of this drift: the Norprene rollers slipping on the shaft collars, and the Nitinol legs slipping relative to the Norprene rollers. To prevent the Norprene rollers from slipping on the shaft collars, we 3D-printed slightly oversized shaft collars and applied an adhesive to hold the rollers in place. To prevent the Nitinol wires from slipping relative to the Norprene rollers, we decreased the distance between the rollers, as described below, thus increasing the compression force acting on the Nitinol wires.

We expected the required motor current to increase as the distance between the
rollers decreased, due to higher rolling resistance. Since the theoretical continuous force output of 10 N was well above our desired output of 1 N, we felt comfortable using up to half of the motor’s rated current of 0.28 A to overcome roller friction, and reserving the other 0.14 A for delivering continuous forces. We incrementally decreased the distance between the rollers in steps of 0.25 mm, beginning with the original distance of 11.5 mm, and measured the motor current needed to move a non-loaded Nitinol wire 50 mm at 50 mm/s across 30 trials in each direction. We continued this process until the average current exceeded 0.14 A, resulting in a selected distance of 10.75 mm. We also measured the force which the Nitinol wire could exert in the final setup by commanding the motor to a constant position, and then slowly adding weight until slip occurred. An average force of 1.5 N was required to induce slip, which is greater than our previously stated maximum expected load of 1 N.

2.5.1 System Design

An inner control loop runs in C++ at about 750 Hz and is responsible for driving each Nitinol wire to a desired position. It uses a Sensoray 626 PCI I/O Board to read the six encoders, converts the quadrature encoder positions to wire lengths, feeds the resulting values to independent manually-tuned PD controllers given by the equation

\[ i(t) = K_p e(t) + K_d \frac{de(t)}{dt}, \]  

with proportional gains \( K_p \) of 150 A/m and derivative gains \( K_d \) of 2 A/s/m, where \( i(t) \) is the commanded motor current in amps and \( e(t) \) is the wire position error in meters. An ACCES I/O board sends the desired motor currents to current amplifiers that drive the six motors. The end-effector’s position and orientation are tracked using a trakSTAR mid-range transmitter and a model 130 sensor that is embedded into the end-effector. The higher level computation of desired leg lengths is handled in MATLAB. With all of the discussed sensor readings and computations, the MATLAB outer loop runs at just over 50 Hz.
Figure 2.9: Responses to a step input both (a) without trajectory constraints and (b) with constraints of 50 mm/s and 50 mm/s\(^2\). All six wire lengths were increased from 0 mm to 50 mm with the end-effector attached.

Step Responses

Figure 2.9(a) shows commanded and measured wire positions following a step input of 50 mm for all of the six wires. Without velocity or acceleration constraints, we see a maximum wire velocity of roughly 400 mm/s, limited by the supplied voltage of 20 V. Although the wire positions track commanded positions with minimal overshoot and reasonable settling times, these results show the response of only the individual wires, not the end-effector. In general, the combined stiffness of all six wires varies inversely with wire lengths, meaning fast movements near the boundary of the fingertip tend to cause the end-effector to oscillate after wire motion has stopped. We measured the oscillation peak-to-peak amplitudes and settling times following unconstrained step inputs from the zero pose, located 20 mm in front of the fingertip, to the nine fingertip locations shown in Fig. 2.10.

We found that without velocity and acceleration constraints, the vibrations can be quite large; over ten trials, we measured a mean maximum oscillation peak-to-peak amplitude of 35.66 mm with a standard error of 0.28 mm at position F. Left uncorrected, these end-effector oscillations may lead to undesired vibrotactile sensations. Furthermore, the maximum oscillation amplitudes are larger than typical finger widths [15, 33], meaning contact may occur anywhere on the fingertip. In the presence of these end-effector vibrations, contact location depends on when contact
Figure 2.10: Nine destinations of step inputs, labeled in black letters, used to evaluate end-effector vibrations. The finger coordinates associated with location G are shown in red. The depicted finger is pointing in the $-X$ direction, and the fingernail is in the $+Z$ direction.

is made during the oscillation period, preventing the device from rendering contact at a precise location. The settling time, defined as the time after which the error stays below 5 percent of the step distance, was also greatest at location F, as the system required more than the allotted five seconds to stay within 5 percent of steady-state. Sample data for position F is shown in Fig. 2.11, and details of the responses at all nine locations are given in Table 2.3.

**Trajectory Constraints**

We hypothesized that introducing velocity and acceleration constraints on the commanded wire lengths would reduce end-effector oscillations and wire slip, but would require longer trajectory times. This trade-off suggested the need to test a broad set of velocity and acceleration constraints in order to determine reasonable limits. We recorded system responses under velocity limits of $\{25, 50, 100, 200, 300, \infty\}$ mm/s and acceleration limits of $\{25, 50, 100, 250, 500, \infty\}$ mm/s$^2$. Since we observed previously that steps to position F induced the largest end-effector oscillations, each constraint combination was evaluated over ten trials between the zero pose and posi-
Figure 2.11: Response following a step from the zero pose to location F with and without velocity and acceleration constraints. End-effector position is presented as distance from the zero position.

As expected, we see that more restrictive trajectory constraints tend to minimize oscillation amplitudes. Including trajectory constraints has a similar effect on settling time, as reduced oscillations allow the system to reach steady-state more quickly. However, constraints that are too restrictive cause the settling time to increase, because the ideal trajectory takes longer to complete. Wire slip appears to be independent of trajectory constraints, although having no constraints at all leads to slightly more slip. While there is clearly an overall benefit to imposing some trajectory constraints, these results suggest a trade-off between the speed at which the contact surface can reach a desired pose and the magnitude of lingering vibrations, which may impact the tactile cue itself. Although these measurements provide insight into the best way to control the device, more work would be needed to understand the effect of oscillation amplitudes and settling times on user perception of tactile cues delivered by the device. For the experimental results presented in this chapter, we are more interested in accuracy rather than the speed at which cues are delivered, so we chose to implement relatively conservative constraints of...
Table 2.3: Response characteristics during steps from the zero pose to nine locations around the fingertip. Means and standard errors of peak-to-peak amplitudes and five-percent settling times over 10 trials are reported at each location.

<table>
<thead>
<tr>
<th>Location Label</th>
<th>Max Oscillation Amplitude (mm)</th>
<th>Settling Time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>8.15 ± 0.05</td>
<td>0.39 ± 0.01</td>
</tr>
<tr>
<td>B</td>
<td>15.78 ± 0.06</td>
<td>3.51 ± 0.11</td>
</tr>
<tr>
<td>C</td>
<td>4.07 ± 0.06</td>
<td>0.25 ± 0.01</td>
</tr>
<tr>
<td>D</td>
<td>3.48 ± 0.01</td>
<td>0.20 ± 0.00</td>
</tr>
<tr>
<td>E</td>
<td>0.05 ± 0.00</td>
<td>0.19 ± 0.00</td>
</tr>
<tr>
<td>F</td>
<td>35.66 ± 0.28</td>
<td>&gt; 5</td>
</tr>
<tr>
<td>G</td>
<td>1.99 ± 0.06</td>
<td>0.27 ± 0.01</td>
</tr>
<tr>
<td>H</td>
<td>18.15 ± 0.21</td>
<td>2.07 ± 0.04</td>
</tr>
<tr>
<td>I</td>
<td>7.86 ± 0.18</td>
<td>0.54 ± 0.04</td>
</tr>
</tbody>
</table>

50 mm/s and 50 mm/s². These constraints, highlighted in yellow, resulted in peak-to-peak oscillations of 0.50 mm and five-percent settling times of 1.11 s.

2.6 Simulated Performance

The control scheme for our device relies on position-based inverse kinematics. We first determine a desired position and orientation for the contact surface and then use inverse kinematics to estimate the leg lengths required to move the contact surface to the desired pose. To perform the requisite inverse kinematics calculations, we use an adapted version of the simulator developed by Till et al. [79]. The simulator itself is described in far greater detail in [79] and Section 2.1.1 of this chapter. As was previously discussed, forward and inverse kinematics have similar complexity, unlike rigid robots.

Small changes in pose can be computed at rates over 1 kHz [79], since using the previous solution as the new initial guess yields relatively small starting error; larger changes in pose can take a significant amount of time to compute. Inherent to this process is a trade-off between computation time and precision. Increasing the number of iterations allowed in the inverse kinematics calculation increases the likelihood of a solution being found, but it also allows the simulation to search for a longer duration even if there is no solution. For the results presented in this section, we allowed 300 iterations per computation. Increasing the iteration limit to 1,000 resulted in twice the computation time with identical performance on a test set of
Table 2.4: Maximum Oscillation Amplitude (mm)

<table>
<thead>
<tr>
<th>Max. Vel. (m/s)</th>
<th>Max. Accel. (mm/s)²</th>
<th>25</th>
<th>50</th>
<th>100</th>
<th>250</th>
<th>500</th>
<th>∞</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>0.57±0.07</td>
<td>0.42±0.04</td>
<td>0.50±0.08</td>
<td>0.62±0.06</td>
<td>0.62±0.04</td>
<td>0.66±0.06</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>0.59±0.06</td>
<td>0.50±0.06</td>
<td>0.66±0.08</td>
<td>0.50±0.09</td>
<td>0.61±0.07</td>
<td>0.84±0.07</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>×</td>
<td>0.55±0.05</td>
<td>0.86±0.07</td>
<td>1.15±0.14</td>
<td>1.79±0.04</td>
<td>1.59±0.13</td>
<td></td>
</tr>
<tr>
<td>200</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>2.40±0.20</td>
<td>3.44±0.16</td>
<td></td>
</tr>
<tr>
<td>300</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>8.23±0.13</td>
<td></td>
</tr>
<tr>
<td>∞</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>35.66±0.28</td>
<td></td>
</tr>
</tbody>
</table>

Table 2.5: Five Percent Settling Time (s)

<table>
<thead>
<tr>
<th>Max. Vel. (m/s)</th>
<th>Max. Accel. (mm/s)²</th>
<th>25</th>
<th>50</th>
<th>100</th>
<th>250</th>
<th>500</th>
<th>∞</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>1.69±0.00</td>
<td>1.39±0.02</td>
<td>1.19±0.01</td>
<td>1.13±0.00</td>
<td>1.13±0.01</td>
<td>1.19±0.06</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>1.54±0.01</td>
<td>1.11±0.00</td>
<td>0.89±0.01</td>
<td>0.70±0.01</td>
<td>0.60±0.00</td>
<td>0.58±0.04</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>×</td>
<td>1.12±0.04</td>
<td>0.89±0.03</td>
<td>0.60±0.03</td>
<td>0.64±0.00</td>
<td>0.34±0.03</td>
<td></td>
</tr>
<tr>
<td>200</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>&gt; 5</td>
</tr>
<tr>
<td>300</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>&gt; 5</td>
</tr>
<tr>
<td>∞</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>&gt; 5</td>
</tr>
</tbody>
</table>

Table 2.6: Wire Slip (mm)

<table>
<thead>
<tr>
<th>Max. Vel. (m/s)</th>
<th>Max. Accel. (mm/s)²</th>
<th>25</th>
<th>50</th>
<th>100</th>
<th>250</th>
<th>500</th>
<th>∞</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>0.26±0.07</td>
<td>0.26±0.07</td>
<td>0.28±0.06</td>
<td>0.26±0.05</td>
<td>0.26±0.06</td>
<td>0.24±0.05</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>0.28±0.06</td>
<td>0.29±0.07</td>
<td>0.30±0.07</td>
<td>0.30±0.07</td>
<td>0.29±0.07</td>
<td>0.29±0.07</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>×</td>
<td>0.25±0.06</td>
<td>0.28±0.06</td>
<td>0.26±0.06</td>
<td>0.30±0.08</td>
<td>0.30±0.08</td>
<td></td>
</tr>
<tr>
<td>200</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>0.27±0.06</td>
<td>0.28±0.07</td>
</tr>
<tr>
<td>300</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>0.26±0.06</td>
</tr>
<tr>
<td>∞</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>0.39±0.04</td>
</tr>
</tbody>
</table>

100 calculations.

2.6.1 Inverse Kinematics in Discrete Workspace

The time required to solve the inverse kinematics depends on the required change in leg lengths and the proximity of the configuration to a singularity. When using this device for fingertip haptic applications, we expect the leg lengths to require large displacements to keep up with quick finger motions. We also want to use the entire workspace of the device, including portions near singularities. Thus we developed
an inverse kinematics lookup table to speed up the process. Some computation is still needed to map the discrete lookup table to the continuous workspace, but the interpolation methods we will discuss shortly can all be performed much more quickly than the original inverse kinematics computation.

**Workspace Discretization**

To reach every desired tactile cue, the range of motion required for each leg varies between 60 mm and 150 mm. If each input is discretized into 1 mm increments and every combination of leg lengths within the boundaries is simulated, there would be over $1.166 \times 10^{12}$ poses to simulate, which is unachievable. Instead, we look at the subset of the workspace we expect to be relevant. We first define six finger coordinates to better describe the workspace relative to the tactile cues we want to deliver, as shown in Fig. 2.10. The first three coordinates $(r, \theta, \phi)$ describe the position of the contact platform relative to the finger in spherical coordinates. We assume the platform is normal to the finger at the contact position, such that $\theta$ and $\phi$ define the location on a sphere of radius $r$. The variable $r$ allows us to simulate contact with fingertips of different sizes or to apply varying levels of normal displacement into a single fingertip. The final three coordinates $(u, v, \alpha)$ define the planar translation of the contact surface and rotation of the surface about its normal vector. We discretized the workspace along our new coordinates $(r, \theta, \phi, u, v, \alpha)$ into the 55,125 configurations given in Table 2.7, consisting of 245 fingertip locations and 225 shear displacements at each fingertip location. We recorded the leg lengths needed for each successful simulation. Note that the workspace and coordinate system shown in Fig. 2.10 are symmetric about the XZ-plane. Our device is also symmetric about the XZ-plane, meaning we need to simulate only one half of the workspace. We chose to focus on the half with non-negative Y coordinates, which is identically equal to the portion of the workspace defined by $\theta \geq 0$.

**Interpolation of Unsuccessful IK Calculations**

While all contact positions on the fingertip were reachable, roughly 20 percent of all shear configurations were unreachable, resulting in an incomplete lookup table. In general, the reachable shear configurations at each fingertip contact position formed a three-dimensional polyhedron. The ultimate goal was to use our lookup
Table 2.7: Lookup table discretization

<table>
<thead>
<tr>
<th>Coordinate</th>
<th>( r ) (mm)</th>
<th>( \theta ) (deg.)</th>
<th>( \phi ) (deg.)</th>
<th>( u ) (mm)</th>
<th>( v ) (mm)</th>
<th>( \alpha ) (deg.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min. Values</td>
<td>6</td>
<td>0</td>
<td>0</td>
<td>-4</td>
<td>-4</td>
<td>-20</td>
</tr>
<tr>
<td>Max. Values</td>
<td>10</td>
<td>90</td>
<td>90</td>
<td>4</td>
<td>4</td>
<td>20</td>
</tr>
<tr>
<td>Step Size</td>
<td>1</td>
<td>15</td>
<td>15</td>
<td>2</td>
<td>2</td>
<td>5</td>
</tr>
</tbody>
</table>

Table to interpolate leg lengths in the continuous workspace of the device, but for poses far away from this polyhedron of reachable configurations, we expected interpolation methods to be unreliable. Thus we first defined the boundary of reachable configurations for each fingertip location and mapped missing elements of our lookup table to these boundaries so that we were certain later interpolation methods would yield reasonable values.

The grid seen in Fig. 2.12 represents the three-dimensional set of shear cues that were evaluated at one fingertip location given by \( r = 6 \) mm, \( \theta = 45 \) degrees, and \( \phi = 90 \) degrees. Reachable shear configurations are depicted by solid black dots, while unreachable shear configurations are marked by \( \times \). The boundary of the 3-dimensional set of reachable shear configurations is shown in a transparent gray shading. If an unreachable pose was within this gray boundary, which rarely occurred, we linearly interpolated the leg lengths at the unreachable shear coordinates. If an unreachable pose was outside of the gray boundary, we first calculated the nearest point on the boundary and then linearly interpolated the leg lengths.

**Boundary Calculation**

Note that the boundary of the 3D polyhedron of reachable shear configurations is not clearly defined; we can select the convex hull as the boundary, or a more compact boundary. A boundary that is too loose may lead to larger interpolation errors, while a more compact boundary may unnecessarily restrict the region of deliverable shear cues. The looseness of a boundary is described by a shrink factor, which is related to the size of alpha shapes used to make the boundary. The process is described in greater detail in [17], but for our purposes it suffices to say a larger shrink factor leads to a more compact boundary. In order to determine the ideal
Figure 2.12: All shear cues at a single fingertip location defined by \((r, \theta, \phi) = (6 \text{ mm}, 45 \text{ deg.}, 90 \text{ deg.})\). Each black dot represents a reachable shear cue, or a cue for which the simulation successfully estimated the required leg lengths, whereas each \(\times\) indicates an unreachable shear pose. A boundary around reachable shear configurations is shown in gray. The red dot is an unreachable shear cue, and the blue dot shows the nearest point on the boundary. A similar boundary of reachable shear configurations is defined at each of the 245 fingertip locations.

shrink factor, we selected a large set of random unreachable poses, interpolated the required leg lengths using a variety of shrink factors, plugged the leg lengths into the forward kinematics simulator to obtain the predicted pose of the contact surface, and finally computed the errors between the desired contact surface pose and the estimated poses. The results are given in Table 2.8. We see that the using a shrink factor of one leads to the highest simulation success rate (0.96), the smallest position error (0.83 mm), and average angular error (7.41 degrees). We thus chose this setting.

We also note that the mean errors depend on the scaling used for positional and angular coordinates. Ideally, we would scale coordinates based on expected perception, such that a unit of translational shear feels similar in magnitude to a unit of rotational shear. Although interesting, finding the best equivalence setting is beyond the scope of this chapter. As a preliminary effort to scale position and orientation appropriately, we note that a 1 mm arc along the surface of a finger with
Table 2.8: Errors of simulating 1,000 unreachable poses using various shrink factors to estimate leg lengths. Means and standard deviations are reported. Moving forward, we implement a shrink factor of one as this value leads to the highest simulation success rate (0.96), the smallest position error (0.83 mm), and average angular error (7.41 degrees).

<table>
<thead>
<tr>
<th>Shrink Factor</th>
<th>Simulation Success Rate</th>
<th>Position Error (mm)</th>
<th>Angular Error (deg.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.94</td>
<td>1.08 ± 2.13</td>
<td>7.50 ± 6.78</td>
</tr>
<tr>
<td>0.5</td>
<td>0.94</td>
<td>0.90 ± 1.46</td>
<td>7.30 ± 5.91</td>
</tr>
<tr>
<td>0.8</td>
<td>0.95</td>
<td>0.85 ± 1.39</td>
<td>7.33 ± 5.69</td>
</tr>
<tr>
<td>1</td>
<td>0.96</td>
<td>0.83 ± 1.37</td>
<td>7.41 ± 5.62</td>
</tr>
</tbody>
</table>

diameter 20 mm corresponds to a rotation of 5.7 degrees about the finger center, or approximately 5 degrees. The results given in Table 2.8 were obtained by giving 1 mm of translational shear the same weight as 5 degrees of rotational shear. If we instead give 1 mm equal weight to 1 degree, we get a success rate of 0.94, a mean position error of 2.92 mm, and a mean angular error of 2.74 degrees. As expected, these translational errors are larger than and these angular errors are smaller than the results in Table 2.8. We will discuss methods to reduce errors of the physical device in the next section, but the scaling used when creating the lookup table should be carefully chosen to reflect the relative importance of shear directions for the intended use of the device.

2.6.2 Inverse Kinematics in Continuous Workspace

When using the real device, we will want the contact surface to achieve positions and orientations in the continuous workspace of the device, rather than only poses that are exactly in the lookup table; thus we must interpolate between entries of the lookup table. We evaluated different interpolation methods on 1,000 random poses, but this time the poses were selected from the entire continuous workspace of interest around the finger (Table 2.7). For each pose, we used a variety of interpolation methods on our lookup table to estimate the required leg lengths, and then we passed each set of leg lengths through the forward kinematics simulator and recorded the error between the resulting position and orientation and the desired position and orientation.
Interpolation Methods

Some of the interpolation methods we implemented ignored the structure of the lookup table and instead estimated the leg lengths as a weighted average of a set of K nearest neighbors (KNN). For the KNN method, the estimated leg lengths were given by

\[
\hat{L}_q = \sum_{n=1}^{K} L_n w_n \tag{2.6.1}
\]

and

\[
w_n = \frac{1}{\frac{D_i}{\sum_{j=1}^{K} \frac{1}{D_j}}}, \tag{2.6.2}
\]

where \(\hat{L}_q\) is a vector of six interpolated leg lengths for the query point, \(L_n\) and \(w_n\) are the leg length vector and weight associated with the \(n^{th}\) neighbor, and \(D_i\) is the \(L_2\) norm between the \(i^{th}\) neighbor and the query point, with 5 degrees given equal weight as 1 mm. We evaluated distance metrics for \(\alpha\) values of 0.5, 1 and 2, but the effect of \(\alpha\) was much smaller than the effect of \(K\), so we present results for only \(\alpha = 1\).

We also evaluated linear and cubic interpolation methods, both of which take into account the gridded structure of the lookup table. Linear interpolation fits the function

\[
f(r, \theta, \phi, u, v, \alpha) = \sum_{i=0}^{1} \sum_{j=0}^{1} \cdots \sum_{n=0}^{1} a_{ijklmn} r^i \theta^j \phi^k u^l v^m \alpha^n \tag{2.6.3}
\]

to the 64 vertices of the \(2 \times 2 \times 2 \times 2 \times 2 \times 2\) grid around the query point, while cubic interpolation fits the function

\[
f(r, \theta, \phi, u, v, \alpha) = \sum_{i=0}^{3} \sum_{j=0}^{3} \cdots \sum_{n=0}^{3} a_{ijklmn} r^i \theta^j \phi^k u^l v^m \alpha^n \tag{2.6.4}
\]

to the 4,096 vertices of a \(4 \times 4 \times 4 \times 4 \times 4 \times 4\) grid around the query point, with the query point being as close to the center as the workspace boundaries allow.
Table 2.9: Errors of simulating 1,000 poses in the continuous workspace using four interpolation methods. Means and standard deviations are reported. Moving forward, we implement cubic interpolation, which had the smallest mean position error (0.22 mm) and the smallest mean angular error (1.03 degrees).

<table>
<thead>
<tr>
<th>Interpolation Method</th>
<th>Simulation Success Rate</th>
<th>Position Error (mm)</th>
<th>Angular Error (deg.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>4-NN</td>
<td>0.995</td>
<td>0.90 ± 0.45</td>
<td>5.72 ± 2.68</td>
</tr>
<tr>
<td>8-NN</td>
<td>0.995</td>
<td>0.86 ± 0.41</td>
<td>5.67 ± 2.63</td>
</tr>
<tr>
<td>16-NN</td>
<td>0.995</td>
<td>0.92 ± 0.42</td>
<td>5.59 ± 2.62</td>
</tr>
<tr>
<td>Linear</td>
<td>0.996</td>
<td>0.39 ± 0.45</td>
<td>1.37 ± 2.36</td>
</tr>
<tr>
<td>Cubic</td>
<td>0.997</td>
<td>0.22 ± 0.63</td>
<td>1.03 ± 2.66</td>
</tr>
</tbody>
</table>

Interpolation Results

Table 2.9 shows the success rate, mean position error, and mean angular error for 1,000 simulated poses in the continuous workspace using linear interpolation, cubic interpolation, and K-nearest neighbors with 4, 8, and 16 neighbors weighted by the inverse $L_2$ norm. We see that all methods have simulation success rates of over 99%. The best performing method was cubic interpolation, which had the smallest mean position error at 0.22 mm and the smallest mean angular error at 1.03 degrees; we thus use this method. The leg lengths can be computed via cubic interpolation at over 150 Hz.

2.7 Experimental Performance

Whereas the previous section focused on how to minimize the simulated error, this section evaluates the possible sources of error associated with the physical device and presents methods for their reduction.

2.7.1 Preliminary Evaluation

As a preliminary measure of how accurately the physical device could deliver a desired tactile cue via our inverse kinematics procedure, we randomly selected 1,000 desired poses in the continuous workspace, computed the required leg lengths via the procedure outlined in Section 2.6, and commanded the physical device to the desired leg lengths. We measured the error for each pose using the trakSTAR,
Figure 2.13: Error in mm observed when driving the physical device to 1,000 randomly selected poses. The depicted spherical finger has a radius of 8 mm, but poses range between 6 mm and 10 mm from the fingertip center, as given by Table 2.7. Dots are shown at desired positions.

which has a reported RMS accuracy of 1.4 mm and 0.5 degrees. Figure 2.13 shows the physical device’s positional error at each of the 1,000 desired poses. The mean positional error was 3.1 mm with a standard deviation of 1.1 mm, and the mean angular error was 6.3 degrees with a standard deviation of 4.8 degrees. Figure 2.13 shows that the error depends on the target location in the workspace. Although over 50 percent of the workspace results in errors of less than 3 mm, the device appears to be more prone to errors near the workspace boundary, where errors up to roughly 12 mm were measured.

Overall, we see that discrepancies exist between the simulation and the physical device. We expect some error due to imperfections in the Nitinol legs, steady-state error caused by friction, and incomplete loading models, such as ignoring the weight of the end-effector and attached trakSTAR sensor cable. We want to minimize the error of the physical device, particularly for the poses near the boundary of the workspace that showed errors larger than 5 mm.
2.7.2 Error Reduction via Measured Jacobians

We developed a method of reducing the physical device’s error by measuring the Jacobian at a pose and adjusting the leg lengths as needed to take a step toward the desired pose. To compute the Jacobian, we first command the device to the desired pose and record the pose measured by a trakSTAR sensor on the end-effector. We then extend a single leg by 2 mm and record the change in pose as the corresponding column of the Jacobian matrix. The first three elements of this vector are the difference in position, and the final three elements represent the change in orientation, given as the axis of rotation multiplied by the magnitude of rotation. Once each leg has been individually extended and the entire Jacobian has been computed, we update the leg lengths by

\[ \Delta \vec{l} = \alpha J^T \vec{e}, \]  

(2.7.1)

where \( \Delta \vec{l} \) is the change in leg lengths, alpha is a scalar multiplier that we set to 0.2, \( J \) is the Jacobian, and \( \vec{e} \) is a unit vector comprised of three position components and three components corresponding to the axis-angle representation of the angular error. Similar to Section 2.6.1, we use units of 1 mm and 5 degrees to represent positional and angular error, respectively. This calculation has the effect of applying a virtual force and torque to the end-effector and is computationally stable for any numerical Jacobian [93]. We repeat this process until the end-effector reaches a steady-state pose, which we define as moving less than 0.2 mm from the previous pose.

This procedure admittedly takes some time to perform, making it impractical to implement for every entry in the lookup table. At present we have corrected the leg lengths for the set of 63 poses presented in the human-subject study. Figure 2.14 shows the contact surface poses both before and after using the measured Jacobian to update leg lengths. The original mean errors and standard deviations for all 63 poses were 3.6 \( \pm \) 1.8 mm and 6.5 \( \pm \) 5.4 degrees. Additionally, we see in Fig. 2.14(a) that in general the true poses have a y-coordinate greater than the desired y-coordinate, which we believe is likely caused by the trakSTAR sensor cable lightly pulling the end-effector in the +y direction. This error is not symmetric about the XZ-plane and further motivates the need for error reduction via the
measured Jacobian.

After updating the leg lengths via the measured Jacobians, the mean errors were $1.4 \pm 0.9$ mm and $7.5 \pm 5.5$ degrees, or a 60% reduction in positional error and a 15% increase in angular error. Note that the updated poses had approximately 1 mm of positional error for every 5 degrees in angular error, matching the scaling for error vector $\vec{e}$ of Eq. (2.7.1). Although the angular error increased slightly, the measured Jacobian allowed us to reduce the overall error with respect to this scaling. Angular errors were still over 20 degrees for some of the updated poses at $(r, \theta, \phi) = (6, \pm 90, 90)$, suggesting that these poses are not truly reachable by the device.

The updated mean positional error plus one standard deviation is approximately equal to two-point discrimination thresholds on the fingertip, which Won et al. measured to be between 1.9 mm and 2.5 mm, depending on sex and indentor sharpness [94]. We do expect that these errors could cause users to perceive contact at locations that are slightly different than the desired locations. However, the smallest distance between two of the nine contact locations used for the human-subject study presented in Section 2.8 is 4.1 mm for a 12 mm diameter fingertip. Since the distance between labeled locations is greater than the mean measured errors, we expect contact rendered at these nine locations to be distinguishable.

### 2.7.3 Self-calibration

We have mentioned efforts to design the roller mechanisms such that slip of the Nitinol wires is minimal. The results reported thus far largely avoid any issues with slip by collecting data in small samples before error accumulates. However when using this device continuously for longer periods of time, such as when allowing the user to freely explore a virtual environment or performing the user study presented in this chapter, a means of measuring and accounting for slip is necessary. We address slip accumulation through a self-calibration procedure. The procedure begins by driving the contact surface to the zero pose. We then measure the position and orientation of the end-effector and the housing of the device using two trakSTAR sensors, and we estimate the leg lengths via the inverse kinematics simulation. Finally, since the estimated lengths should all be zero, we offset our leg length measurements by the estimated lengths. As long as this procedure is performed when leg length error is
Figure 2.14: Contact surface poses before (a) and after (b) performing error reduction. Poses are represented by frames to capture position and orientation, with desired frames in black and measured frames in brighter colors. Blue lines link desired poses to the corresponding measured poses. We highlight an example desired pose and the corresponding measured pose before and after error reduction.

less than roughly 10 mm per leg, the procedure can finish in a few seconds.

To evaluate our calibration procedure, we offset the leg lengths from zero by a random vector of magnitude less than 20 mm and measured the error of the contact surface before and after calibration. To include position and orientation errors, we calculated a non-dimensional error as the norm of a four-dimensional error vector that included positional error along all three axes and the angular error about the Euler axis between the desired orientation and the measured orientation. Units of mm and 5 degrees were used, such that 1 mm error again carries as much weight as 5 degrees error. The results from 100 calibration tests are seen in Fig. 2.15. Every pose had a smaller error after calibration, and we see from the line of best fit that error was reduced on average by about 85%. Even with extremely large pre-calibration errors of 30 error units, we can run the calibration procedure twice to bring the error to less than one error unit.

At present, the entire procedure is run in the same thread as the rest of the code. However, in the future we hope to measure the trakSTAR readings during
normal use of the device and feed these readings to another thread to calculate the inverse kinematics, such that the error can continuously be estimated and corrected as the device is used.

2.8 User Study on Perception of Rendered Location and Shear Direction

A significant preliminary evaluation metric for our 6-DOF fingertip haptic device is how well it can deliver tactile cues to all parts of a user’s fingertip. A motion-tracking system adequately measures the accuracy and consistency with which the device positions and orients the contact surface, but only user feedback can show how effective the device ultimately is in delivering tactile cues.

2.8.1 Delivered Cues

We delivered the same set of cues to participants regardless of finger shape or size. We view the ability to adapt to unique fingertips as one of the potential advantages of a 6-DOF device, and we wanted to determine whether we need to account for fingertip shape and size when controlling the device. We defined nine contact
Figure 2.16: The user study setup, including a visual guide (a) for subjects that shows the possible contact positions (letters) and shear directions (colors). The bottom images show (b) the setup for a preliminary example cue and (c) the setup with the device blocked from the user’s sight.

Locations across the surface of the fingertip, labeled A through I in Fig. 2.16(a). These locations were purposely selected to include positions near the center of the fingertip, which we expected to require relatively small deflections in the Nitinol wires and thus be delivered with little error, and positions on the boundary of the fingertip, such as the side, front, and bottom, where we expected larger deflections in the Nitinol wires and greater error. At each location, we also defined six shear directions, which are depicted as colored arrows in Fig. 2.16(a). These shear directions included four 4 mm translations tangent to the fingertip surface at the given location (shown in green, yellow, red and blue), and two 20 degree rotations about the vector normal to the fingertip at the given location (shown in purple and white). These cues were selected to represent the 6-DOF capabilities of the device,
as two DOF are required to locate the contact position on the fingertip, one DOF is required for indentation into the fingertip, and three DOF are required for the described shear displacements.

The position errors of all cues delivered in the user study are shown in Fig. 2.14(b). As was previously reported, these 63 cues had a mean error of $1.4 \pm 0.9$ mm and $7.5 \pm 5.5$ degrees. We measured the positional repeatability of the cues by driving the end-effector to each of the nine locations over 10 trials, finding the centroid measured position for each location, and then computing the distance between the centroids and each individual trial. Over all 90 trials, the mean distance from the centroids was only $0.3$ mm with a standard deviation of $0.2$ mm. We also measured the forces delivered to a SynTouch BioTac sensor attached to a ATI Nano17 force/torque sensor. The BioTac was used only as a biomimetic fingertip, while all forces and moments were recorded by the Nano17. The measured cues are shown in Fig. 2.17. To display rotational shear cues in an intuitive manner, we subtracted the cross product of the position vector and the measured force vector from the measured moment, and we projected the result onto the normal vector. The magnitude of deliverable force seems to be limited at the edges and bottom of the fingertip, whereas cues near the center of the finger have greater magnitudes and are more directionally distinct from each other. These measurements neglect slip, which may provide additional directional information to the user.

Finally, we evaluated the transient force to ensure that our trajectory constraints eliminate unwanted vibrations that could affect perception. As shown in Fig. 2.18, although the force decays slightly as the soft fingertip deforms, there are no noticeable oscillations in force when contact is made.

### 2.8.2 Setup

Nineteen males and five females aged between 18 and 33 participated in the study, all of whom reported full sensory capabilities in their right index finger. Each participant heard an overview of the study, gave informed consent, and completed a demographic survey. We took pictures of the top and side of each subject’s right index fingertip on gridded paper, to make sure the width was within the range for which our device was calibrated (12 mm to 24 mm) and to later analyze the relationship between fingertip size and perception of delivered cues. We measured
Figure 2.17: Measured forces and moments for each of the 63 user study cues when delivered to a biomimetic finger attached to a Nano17. Colors for cues match those shown in Fig. 2.16(a), with black arrows representing location cues without shear force. For visualization, depicted forces (black, red, yellow, blue, and green arrows) have a magnitude of 2N/mm and range from 0.1N to 2.1N. Depicted moments (purple and gray arrows) have a magnitude of 0.5N-mm/mm and range from 0.5N-mm to 4.5N-mm.

a mean fingertip width of 16.3 mm with a standard deviation of 1.2 mm, and no fingertip widths were beyond the allowable range. Although we eventually want to use this device to provide haptic feedback to a user who is freely exploring a virtual environment, the purpose of this study was simply to show what cues can be delivered and how accurately cues are perceived. For this reason the device and the user’s right index finger were both held stationary. The subject’s wrist and forearm rested on the table, and the index finger was guided into position and strapped into place for the duration of the study, as shown in Fig. 2.16. An example trial was delivered to make sure the procedure was clearly understood, and then the device was hidden from view. Pink noise, a less abrasive alternative to white noise, was played via headphones to ensure responses were based solely on tactile perception.

For each trial in the study, we first drove the contact surface to a lettered location on a sphere of radius 12 mm, to avoid making contact with the subject’s finger. The
contact surface was then driven to the same lettered location on a sphere of radius 6 mm, with the goal of delivering a normal force into the fingertip. A voice prompt was played through the headphones asking the participant to verbally indicate the letter at which contact was felt, or that no contact was felt, and to state his or her confidence on a scale from one (not at all confident) to ten (extremely confident). The contact surface was then commanded to shift either 4 mm or 20 degrees in the direction of one of the six colored arrows. A second verbal prompt was played that asked the subject to say the color of the arrow in which either shear force or motion was perceived, or that no change was noticed, and give his or her confidence. The device re-calibrated its pose after every shear cue to ensure consistency across users regardless of the order in which cues were presented.

Each subject completed 54 trials, with each trial consisting of one location cue and one shear cue. The order of cues was random for each subject. After finishing all 54 trials, the participant completed a post-survey that asked which cues were delivered well and which cues were delivered poorly, also providing space for any additional comments and/or suggestions. The study took roughly 30 minutes, including a break offered after half of the trials were completed, and participants were not compensated in any way. Study procedures were approved by the Penn IRB under protocol number 831368.

Figure 2.18: Measured force in the +Z direction as contact is made with a biomimetic finger at location F.
2.8.3 Results And Discussion

Across all delivered cues, subjects perceived 80.5% of commanded location cues accurately and reported a mean confidence of 8.48 out of 10. Figure 2.19 presents a confusion matrix of commanded and perceived contact locations across all 24 subjects. Each location was delivered six times for each subject, so every row in the confusion matrix represents all 144 times a position cue was delivered.

On the whole, we see that subjects were able to accurately detect the location of contact. The lowest true positive rates were observed at locations B and H, at rates of 0.69 and 0.59 respectively. Interestingly, we previously measured similar positional and angular errors at locations B and H as at the other seven locations, as shown in Fig. 2.14, so we have no reason to believe the contact itself is any less accurate. Furthermore, a review of tactile perception at the fingertip provided by Lederman and Klatzky shows that two-point discrimination thresholds (∼2–4 mm) and point-localization thresholds (∼1–2 mm) are typically smaller than the distance between locations B and C, which is 4.1 mm for a 12 mm diameter fingertip. However, participants in our study were not provided with training trials in which contact was made at each location; thus users had to rely on an internal
mapping between the visual geometric model and the tactile sensations they felt. Responses from surveys taken after the study suggest that some users had difficulty discerning locations B and H from locations C and G respectively, possibly due to the geometric ambiguity of neither \( \theta \) nor \( \phi \) equaling 0 or ±90 degrees at locations B, C, G, and H. Indeed, the confusion matrix shows that over 80% of cues rendered at locations B and H were either perceived correctly or as neighbor locations C and G, respectively. Subjects were most accurate at D, with some subjects noting that contact with their fingernail made the location more clear. We see that accuracy generally declined closer to the sides of the fingertip, which is the same trend we observed in the original position error data shown in Figure 2.13. Our results suggest the sides of the fingertip may be less densely innervated with fast-adapting type I (FA-I) and slowly-adapting type I (SA-I) afferents, which are likely responsible for providing spatial details of skin deformation [32], but additional examination of the relationship between fingertip location and discrimination thresholds is needed to support or refute this hypothesis.

Figure 2.20 shows nine confusion matrices, with each matrix depicting the commanded and perceived shear cues at a single commanded position. We include responses only if the location was correctly reported, since the shear direction definitions depend on the location. Each shear cue was delivered once to each participant, so every row represents 24 delivered cues, minus any cues for which the commanded and perceived locations differed. Subjects perceived 54.4% of commanded shear cues accurately, reporting a mean confidence of 7.20 out of 10. We grouped directions into up/down, left/right, and rotations, and then we analyzed the results using two-way repeated measures ANOVA. We found that there was a significant effect of both location \((F(8, 40) = 4.69, p < 0.001)\) and direction \((F(1, 5) = 296.3, p < 0.001)\) on accuracy.

We see that shear cues are generally less accurately perceived than location cues. Accuracy is highest at location E, where all six shear directions have true positive rates over 0.75, and accuracy at E is significantly higher than accuracy at A \((p = 0.04)\) and F \((p < 0.01)\). Overall we see that commanded and perceived directions align more at inner locations such as C, D, E and G than at outer locations such as A, B, H and I. Birznieks et al. [5] previously reported that nearly all FA-I, SA-I, and SA-II afferents located throughout the fingertip respond preferentially to specific directions of force [5], with SA-II being particularly responsive to planar
Figure 2.20: Confusion matrices for shear cues across all 24 subjects. Only cues for which the location was perceived correctly are included. Labels are two-part combinations, with the letter signifying commanded location and the arrow signifying shear direction.
skin stretch [32]. The degradation of shear direction discrimination on the sides of the fingertip may result from a more sparse population of mechanoreceptors in these areas, but again more work is needed to examine how location on the fingertip affects various discrimination thresholds. The force measurements shown in Fig. 2.5 also indicate rendered shear cues were smaller at these outer locations, suggesting that the lower shear cue identification rate at outer locations likely results from rendering error rather than perception error.

Shear cues in the red (down) and green (up) directions were perceived significantly more accurately than in the blue (left) and yellow (right) directions ($p = 0.01$). Birzieks et al. argued that the preferred directions of FA-I, SA-I, and SA-II afferents are distributed among all angular directions, but not uniformly. The finding that users distinguish the red and green directions more accurately than the blue and yellow directions in our study may suggest mechanoreceptors generally respond more strongly to shear forces in these directions. However, other studies have offered contrasting data, with different works showing improved direction discrimination for shear displacements in the radial/volar directions [83] and the proximal/distal directions [16]. The inconsistent findings and small significance values suggest our results may not be sufficiently explained by an effect of shear direction on the threshold for direction discrimination. As such, we believe our observed effect of shear direction on perception accuracy is more likely due to the Nitinol wires being more closely aligned with the red and green directions, leading to increased stiffness and greater ability to deliver force in those directions.

Many users also noted that it was difficult to feel shear cues given at location D due to the device making contact with their fingernail rather than the fingertip itself. However, the responses were relatively accurate at D, with only the rotation cues being perceived with less than 50% accuracy. Johansson found that SA-II afferents with Ruffini endings were responsible for high sensitivity near the proximal or lateral borders of the fingernail, noting that different cells responded more strongly to different directions of pressure applied to the fingernail itself [30]. Although the lack of mechanoreceptors within the fingernail itself may make directional cues to the fingernail feel distant or unclear, we found that users’ shear direction discrimination was not hindered when forces were applied to the fingernail rather than the skin.

Figure 2.21 depicts the trends in accuracy over the course of the study. We see that there is roughly a 10% increase in accuracy over the course of the study for
both location and shear cues. The effect of cue number on accuracy is significant for location \((p = 0.01)\), and we see a non-significant trend for shear \((p = 0.09)\). A few subjects also wrote that they believed they would have performed better if they had experienced all possible cues before starting the study, indicating the perception of improved performance as experience with the device increases.

Finally, we show the relationship between accuracy and fingertip size in Fig. 2.22. We define fingertip size as the sum of the maximum width and maximum height of the fingertip between the tip and the distal interphalangeal (DIP) joint, and we look at overall accuracy of both location and shear cues combined. The effect of fingertip size on accuracy is significant \((p < 0.01)\), with larger fingers resulting in better performance. Users with larger fingertips may be assisted by increased separation between the nine contact locations. For example, the arc length between locations B and C is 4.1 mm on a spherical fingertip of diameter 12 mm, whereas the same arc length is 6.9 mm on a fingertip of diameter 20 mm. However, previous studies have demonstrated that tactile spatial acuity improves with decreasing finger size, likely because SA-I afferents with Merkel endings, which are sensitive to static forces, are more densely packed in smaller fingers [64]. Since the contact surface was commanded along identical trajectories for each participant, we also expect the platform to push deeper into larger fingers, resulting in larger normal forces.
relationship between finger size and accuracy suggests these larger and more widely separated contact forces were more salient for the participants.

### 2.9 Limitations and Future Considerations

Chapter 3 focuses on implementing the 6-DOF Fuppeteer for virtual interactions rather than pursuing further improvements to the design. Here we present a few considerations for future designs of similar high-dimensional parallel continuum manipulators for haptic applications.

First, we noted in Section 2.5 that the Fuppeteer is admittedly heavy (360 g) and bulky (550 cm$^3$) for a fingertip device, encumbering the user and preventing similar devices from being worn on multiple fingers. In this thesis, we are more interested in evaluating the potential of a PCM to render higher-dimensional fingertip cues than optimizing the design to be as lightweight and compact as possible. However, future work could continue the procedure discussed in this chapter to create a more wearable 6-DOF fingertip device. For example, upgrading to Maxon ECX 6 motors can theoretically reduce the motor mass by 70% while still meeting our design requirement of applying 1 N to each of the six wires. It may also be possible to offload the motors to a remote location, such as a backpack, and run the six lightweight wires from the motors to a base platform located at the finger. Accurately controlling the wire lengths in such a setup would require either a defined path length,
e.g. a flexible Bowden cable design, or path length measurement, e.g. passing the wires through a lightweight exoskeleton and measuring joint angles to compensate for hand and arm motions.

Future designs may also benefit from implementing a haptic cost function to help select an optimal set of design parameters. In this chapter, our design refinement process yielded many potential sets of design parameters, and we selected the final design based on overall size and ease of construction. Although we demonstrated that this final design could effectively render 6-DOF cues near the center of the fingertip, our quantitative and qualitative results showed reduced performance near the sides of the fingertip. Instead of selecting a final design based on geometric constraints, we could have achieved a design that is better suited for rendering 6-DOF shear cues at the boundaries of the fingertip by optimizing for maximum deliverable shear at boundary locations or developing a similar haptic cost function.

Finally, the human-subject study results depicted perception accuracy as a function of fingertip location and shear direction. Our quantitative measurements suggest that device performance may be the most influential factor in perception accuracy, since maximum position errors and minimum rendered forces were measured near the sides of the fingertip, where perception accuracy was lowest. However, the relationship between fingertip location and tactile perception deserves more careful analysis. We provided a small glimpse into literature related to two-point discrimination and contact perception thresholds, but the prior work we have seen considers the fingertip as a whole. Finding which locations on the fingertip are most sensitive to various tactile cues would allow us to target those fingertip locations and render more easily perceived sensations, possibly reducing the required workspace of our device.

2.10 Summary

Overall, the results from our human-subject study show that the Fuppeteer can deliver meaningful 6-DOF haptic cues to a user’s fingertip. Over 80% of locations and 54% of shear cues were perceived correctly, although these rates depend on the location of the cue, the user’s familiarity with the device, and finger size. The high rates with which users perceived contact locations and the low measured mean error of 1.4 mm for poses used in the human-subject study suggest that our device
is capable of delivering small contact forces across its entire workspace, although accuracy is better near the center of the fingertip (positions C, D, E, and G) than at the workspace boundaries (positions A and I). The user responses to shear cues suggest that the device is also able to deliver larger stimuli near the center of the finger, but that performance again drops when trying to deliver larger forces near the boundary of the fingertip. While this trend is to be expected with an inherently compliant PCM design, it may be possible to improve the delivery of large forces at the boundaries of the workspace by using closed-loop force control. As is, our PCM design seems to be best suited for delivering a light contact around the entire fingertip, and a 6-DOF combination of contact and shear force near the middle of the fingertip.

The effect of finger size on accuracy reinforces the notion that we need to adjust the device’s behavior for the size and shape of the user’s fingertip, as people with larger fingers tended to perform better in our study. One possible solution is to model the user’s fingertip, in order to describe a personalized relationship between contact surface position and tactile cue. Alternatively, the contact surface could be equipped with a sensing element to either help create this fingertip model beforehand or provide feedback for a closed-loop control scheme. This limitation provides motivation for the personalization approaches we present in Chapter 4.

A key limitation of this study was that the subjects’ hands were kept stationary. In the next chapter, we present a human-subject study with a modified version of Puppeteer that allows unconstrained user motions.
Chapter 3

Vari-Dimensional Tactile Fingertip Feedback

In Chapter 2, we presented the Fingertip Puppeteer, or Fuppeteer, as a potential solution for a higher-dimensional wearable haptic device. As was previously noted, this device has the potential to render the entire 6-DOF set of possible interactions between a stationary index finger and a flat surface described in Chapter 1. The Fuppeteer can advantageously also simulate any other fingertip device that contacts the finger with a flat surface, since the ranges of motion for all such devices fall within its range of motion. We thus decided to use this unique device to explore how the expressiveness of cutaneous fingertip feedback affects user movements and virtual object recognition. This device was characterized from an engineering standpoint and used in a perceptual validation study where it remained stationary on a table and delivered tactile cues to the user’s fingertip (passive touch) in Chapter 2, whereas in this chapter, we integrate the device into a complete haptic rendering system.

As shown in Fig. 3.1, we are purposefully focusing on rich tactile cues and omitting the grounded forces and torques that physically oppose finger movements during real contact in order to achieve a large workspace while limiting system cost and user encumbrance; this popular strategy is sometimes called sensory subtraction [67]. Importantly, even our complex 6-DOF consideration of finger-surface interactions completely disregards the local shape and material properties of the object being contacted, which would require infinite dimensions for proper portrayal.
Figure 3.1: Examples of the visual scene (top) and corresponding tactile cues (bottom) during virtual shape discrimination (left) and mass discrimination (right) tasks. In these examples, the wearable device is rendering 6-DOF tactile feedback to the user’s index finger; the platform orients itself to be tangent to the surface, contacts the appropriate position on the fingertip with varying normal force, and simultaneously applies tangential and rotational shear caused by the interaction taking place.

Instead, we aim to represent any physical contact with the surface’s local tangent plane as a first-order approximation, in much the same way that computer graphics models often represent surfaces as polygon meshes. We are curious to discover whether this proposed 6-DOF approximation of tactile contact offers any benefits over the lower-dimensional tactile rendering approaches that have been previously been explored, as well as how strongly various dimensions of tactile feedback affect human motions during virtual object interaction.

After analyzing related work (Sec. 3.1), we present our approach to creating a novel integrated system that can render 6-DOF fingertip tactile cues to a user exploring an immersive virtual environment (Sec. 3.2). This setup is capable of recreating a wide range of dynamic tactile interactions that have previously been achieved by individual devices published in the wearable haptics literature. We utilize this single system to simulate systems with fewer degrees of freedom in order to examine how the number of degrees of freedom of a device (which dictates the expressiveness of the cues it can deliver) affects the user (Sec. 3.3).

The results of our study (Sec. 3.4) show that the perceptual performance of users is boosted by more complex rendering when the additional DOFs are relevant to the task at hand, in some cases approaching performance associated with real physical interactions. However, even this sophisticated system with 6-DOF tactile cues falls short of imitating the perceived realism associated with physical object interactions. Furthermore, users did not seem to adjust their exploratory movements
based on the available tactile feedback DOFs, raising questions about whether naive individuals can take advantage of richer contact-based cues with minimal training and interaction time. We discuss our results and reflect on the limitations of our approach (Sec. 3.5) to enable other researchers to learn from this investigation.

3.1 Related Work

Several studies have demonstrated that providing cutaneous and kinesthetic force feedback to the fingertip can improve curvature discrimination [66, 19, 9] or the perception of edges and texture [20], compared to providing kinesthetic feedback alone. Cutaneous feedback alone has been shown to reduce grasping forces during manipulation tasks [46, 56], or allow participants to find and follow a virtual surface [75], but that the provision of kinesthetic feedback often further improves performance [75, 56]. Augmenting kinesthetic feedback with cutaneous skin stretch or slip can enhance the perception of friction [77] or reduce the force applied during a manipulation task [85]. Even in the absence of kinesthetic force feedback, skin stretch can allow participants to distinguish weight [51, 72] or directional cues [22, 81, 41].

A few wearable devices have rendered kinesthetic feedback to the finger itself, rather than the entire limb, by grounding the device on the base of the finger or on the hand. Benko et al. showed that the combination of kinesthetic and cutaneous feedback in this manner allowed the realistic presentation of virtual haptic interactions such as localizing targets or tracing lines [4]. Chinello et al. found that cutaneous feedback alone yielded better performance than no haptic feedback during robotic palpation and VR exploration tasks, and that the addition of kinesthetic feedback further improved performance in these tasks [9].

Even from this brief summary of selected previous studies, it is clear that the addition of cutaneous feedback can improve performance in many tasks. However, these past works considered cutaneous feedback as a binary variable that is either present or not present, rather than considering and comparing the different types of cutaneous feedback that can be provided. In this chapter, we use the 6-DOF Fuppeteer to look more closely at how different dimensions of cutaneous feedback affect user performance to inform the design of future devices and the creation of future fingertip tactile feedback systems.
Despite its novelty and appeal, a few important limitations of the Fuppeteer were previously identified in Chapter 2. Due to the compliance of the parallel continuum manipulator design, some end-effector motions result in sustained oscillations. However, we demonstrated that trajectory constraints minimized these vibrations, resulting in no noticeable oscillatory forces once the platform contacts the user’s fingertip; we implement the same trajectory constraints in this chapter. Although the Fuppeteer is position controlled, the device’s compliance also means that forces acting on the end-effector cause some amount of deformation, resulting in a complex relationship between configuration and stiffness. However, it was shown that feed-forward pose commands could be used to render recognizable contact and shear cues at all locations on the fingertip. Although more work is needed to understand the complexities of 6-DOF tactile cues presented by the Fuppeteer, we can nonetheless use this device to render meaningful tactile feedback and answer our research questions. In Chapter 2 we also noted that end-effector pose error tended to accumulate over time, and we presented a quick calibration routine through which accumulated error could be eliminated.

3.2 System Integration

As described below, a few key improvements were necessary to adapt the original Fuppeteer design for use in a virtual environment.

3.2.1 Counterweight

For the study discussed in this chapter, we ask users to hold the device while exploring a virtual environment for approximately 60 minutes. However, the mass of the entire device with its six motors is 0.45 kg. When developing the study, we observed user fatigue before all of the trials were complete. We reduce user fatigue via a 1.5 kg counterweight attached to the device’s hand-held base platform via a three-pulley system that has a 4:1 reduction ratio. The pulley system reduces the acceleration of the counterweight itself, limiting the effect of the counterweight’s inertia on the force applied to the device, and preventing slack in the counterweight line unless the device is accelerated upward at over about 39 m/s² (four times gravitational acceleration). The counterweight cable is tied to the device at a point
Figure 3.2: A labeled diagram of the 6-DOF tactile fingertip feedback device that we adapted for use in the human-subject study described in this chapter.

located above the device’s center of mass when the finger is held flat. We chose this attachment location to minimize the moment about the device’s center of mass, but we note that there is a small moment pushing the finger back to the horizontal orientation used for calibration. A maximum moment about the center of mass of approximately 0.2 Nm is applied by the counterweight when the finger points straight up or down.

The device’s entire workspace and a schematic of the pulley system are shown in Fig. 3.3. The counterweight force always includes a vertical component that reduces the effective weight of the device, but it also includes a horizontal component, since the device is not always located directly beneath the point where the cable exits the last pulley. The highest ratio of horizontal component to vertical component in the counterweight force occurs when the device is positioned in an upper, front corner of the workspace, where a static device feels a vertical force of 3.6 N and a horizontal force of 0.8 N. Although the non-vertical component of the counterweight force may affect the user’s motion, we empirically found the benefit of reducing the
**Figure 3.3:** The workspace for the human-subject study (not drawn to scale). The counterweight pulley system is located above the middle of the back wall of the box, and it pulls upward on the haptic device to reduce the weight that must be supported by the participant.

effective weight to greatly exceed the drawback of slightly pulling the device in the horizontal direction.

### 3.2.2 Device Positioning

The calibration routine described in Chapter 2 to eliminate accumulated error requires a designated calibration pose. To ensure that error is consistently minimized throughout our human-subject study, we calibrate the device before each trial. During the majority of our study, the user is wearing a headset and is thus unable to see the physical world. We provide virtual cues to aid the subject in reaching the calibration pose, but small errors in our tracking or in the subject’s trajectory can make this task difficult. To facilitate reaching the calibration pose, we designed a holder that physically funnels the device to the correct position and orientation, as shown at the bottom of Fig. 3.2.
3.2.3 Finger Positioning

Another important improvement to the mechanical design of the selected device was the inclusion of a two-degree-of-freedom mechanism to position the fingertip. In Chapter 2, one strap was approximately located at the distal interphalangeal (DIP) joint of the index finger, and a second strap was located on the proximal phalanx. Although these two straps allowed the device to be quickly donned and doffed, the fingertips of different subjects were held in different positions, depending on the curvature and size of the individual finger. We previously noted that the size of the user’s fingertip had a significant effect on their perception of tactile cues, which suggests that fingertip position would also have a significant effect on perception with this device. Our new design replaces the distal strap with two narrow straps, each of which can be shortened or lengthened via two independent screw mechanisms. This setup beneficially allows independent adjustment of the overall tightness and the lateral position of the fingertip, as shown in Fig. 3.4.

3.2.4 Sensing and Communication

We use an NDI 3D Guidance trakSTAR electromagnetic tracking system to measure the position and orientation of the hand-held portion of the device, and we then apply a constant transform to estimate the fingertip pose. A MATLAB program, which maintains the virtual environment parameters, uses the fingertip pose to compute the desired tactile interaction and the six associated rod lengths that are
required to render this cue to the user.

We created a visual representation of our virtual environment in Unity, which is displayed to the user via an Oculus Quest VR headset. We pair the Quest with a Bluetooth ESP32 module, allowing fingertip pose to be transmitted to the Quest such that a virtual hand is visually presented at the same position and orientation as the user’s actual hand. This wireless communication channel also allows MATLAB to receive user inputs from the Oculus Touch controller.

### 3.3 Human-subject Study

We designed a human-subject study in which a user wears the haptic device while completing two virtual tasks focused on shape discrimination and mass discrimination. Unlike the human-subject study in Chapter 2, here participants are encouraged to dynamically explore the environment. The user completes several trials of each task while the device provides tactile feedback to the right index finger using one of four rendering methods. Following a within-subjects design, each user completed one block of trials for each of the four rendering methods. Block order was assigned by a balanced Latin square for every four participants, in order to minimize order effects.

#### 3.3.1 Haptic Rendering Methods

Three rendering methods simulate tactile feedback that would be provided by an \( n \)-DOF fingertip device, where \( n \in \{1, 3, 6\} \), and one rendering method allows the user to interact with real objects. Our specific definitions for the three \( n \)-DOF haptic rendering methods are depicted in Fig. 3.5. Our ultimate goal is to compare the subjects’ performance and motion when each of these rendering methods is implemented, to better motivate what types of fingertip devices are best suited to represent diverse haptic interactions.

Our 1-DOF controller simulates a device that moves a flat surface with fixed orientation to make contact with a single location on the fingertip and then apply variable normal force, similar to the VRTouch by Go Touch VR [84]. In this study, the 1-DOF contact location is defined by rotating 30 degrees from the volar aspect, or bottom, of the fingertip toward the distal tip, near location D from Chapter 2. In
Figure 3.5: Example tactile cues that can be presented by our 1-, 3-, and 6-DOF rendering methods. Our 1-DOF method can provide contact and normal force at one fingertip location, the 3-DOF method can render contact and normal force at any fingertip location, and our 6-DOF method can provide contact, normal force, and shear force at any fingertip location.

general, we expect this 1-DOF rendering method to provide a natural representation of contact and penetration depth, especially when the pose of the user’s hand would cause virtual contact to occur near the selected tactile feedback point.

Our 3-DOF controller simulates a device that rotates a flat surface around the finger center and then translates the surface into the finger to apply variable normal force. This 3-DOF rendering method can similarly be used to display contact and penetration depth, but at different orientations and locations on the fingertip. Given the sensitivity of the fingerpad, the contact location is relatively easy to perceive. We expect this method to allow better understanding of a surface’s local slope compared to the 1-DOF method.

Our 6-DOF controller builds upon the 3-DOF controller by including three-dimensional shear cues. Once contact is made with the fingertip, the flat surface can then translate up to 4 mm in any direction tangent to the fingertip surface, and it can rotate up to 20 degrees in either direction about the vector normal to the fingertip surface. The friction between the skin and the platform causes translational and rotation traction that are difficult to see by eye but are relatively easy to feel.
For the trials with real-world haptics, we remove the end-effector of the device such that no contact is made. We then have the participant interact with a physical model of the virtual environment to complete the trial. Whereas the 1-, 3- and 6-DOF rendering methods provide only tactile feedback, the real rendering method also provides kinesthetic feedback (grounded forces and torques) because the finger contacts real objects.

### 3.3.2 Hidden Surface Task

In the hidden surface task, participants explore and try to recognize smoothly curving shapes defined by the equation

\[ z = ax^2 + by^2, \]  

(3.3.1)

where the positive z-direction is up and \( a \) and \( b \) are assigned different values for each of six predetermined surfaces. In each trial, the subject explores a randomly selected surface while receiving tactile feedback from one of the four rendering methods. Images of all six possible surfaces, as shown in Fig. 3.6, are visible on the back wall of the virtual environment. Once the participant believes they know which surface is present, they use an Oculus controller in their left hand to select the corresponding image.

#### Hidden Surface Motivation

These shapes were inspired by Kappers et al. [36], who asked participants to identify a wide range of three-dimensional quadric surfaces. Kirkpatrick and Douglas [40] used five of these surfaces to evaluate point force devices, and Kuchenbecker et al. [42] used a similar set of five surfaces to evaluate how a passive tactile display device affected three-dimensional shape recognition.

Our six surfaces and their corresponding \( a \) and \( b \) values are shown in Fig. 3.6. Using the definition of curvedness in [36], our five non-flat surfaces have a curvedness of \( 4 \, \text{m}^{-1} \), which is within the range of curvedness for their presented shapes. Each of the quadric surfaces spans \( x \in [-200 \, \text{mm}, 200 \, \text{mm}] \) and \( y \in [-150 \, \text{mm}, 150 \, \text{mm}] \), making them similar in size to the 200 mm diameter surfaces presented in [36]. Furthermore, it was shown in [40] that size only accounted for modest changes in
Figure 3.6: The six quadric surfaces presented in the hidden surface task and the corresponding coefficients that are substituted into Eq. (3.3.1). The shorthand names for each surface are given in parenthesis and represent the curvatures in the front/back and left/right directions. Identical shapes were 3D-printed for the real-world haptic rendering method. All virtual and real surfaces are positioned such that the point at \((x, y, z) = (0 \text{ mm}, 0 \text{ mm}, 0 \text{ mm})\) is located at the same position.

the time needed to recognize the surface.

Hidden Surface Haptics

For our 1-DOF, 3-DOF, and 6-DOF haptic rendering methods, we first compute the distance between the virtual fingertip center and the nearest point on the virtual surface, \(d_V\). We define \(d_H\) as the haptic distance from the real fingertip center to the device’s platform, and we set the pose of the end-effector such that \(d_H\) equals \(d_V\). The value of \(d_H\) is restricted to a range of 4 mm to 12 mm, which allowed contact to be made and broken for all fingertip sizes observed in this study. For the 3-DOF and 6-DOF rendering methods, we also assign the orientation of the end-effector such that the flat contact surface is tangent to the virtual surface at the nearest point.

Our 6-DOF rendering method includes three-dimensional shear cues. When the finger makes contact with and moves tangent to the surface, the end-effector moves in the opposite direction so to render translational shear. When the finger rotates about the surface’s normal vector, the end-effector rotates about the same vector in the opposite direction, such that rotational shear is presented. We reduce end-effector translational shear by a factor of five relative to finger motion, in order to
make the sensation smoother and haptically more pleasant. We also have the end-
effector pose creep toward the current finger pose through a discrete 1 Hz low-pass
filter so that we can continue to present shear during longer motions.

For our real-world rendering method, we remove the end-effector and place a 3D-
printed surface in the same position as the virtual surface, allowing the participant
to receive natural haptic feedback as the environment is explored. In these tasks,
the participant continues to wear the headset so that the physical surface is not
visible. An additional face shield was used to prevent viewing of the surface or the
device through gaps around the headset.

3.3.3 Slider Task

In the slider task, participants must determine whether the mass of a disk is less
than or greater than the mass of a reference disk, which is provided for the subject
to interact with at their own discretion. The light, reference, and heavy sliders
have masses of 13 g, 18 g, and 23 g respectively, which pilot studies suggested are
difficult, but possible, to discern. Each disk has a diameter of 50 mm and a height
of 10 mm. Subjects are not allowed to pick up the disks, but rather they must
estimate the mass by pressing their index finger into the disk and sliding it on the
table anywhere in the workspace. To give a response, the participant must slide
the disk to a region marked ‘Light’, or a region marked ‘Heavy’, both of which are
approximately 300 mm from the starting disk location.

Slider Motivation

Many previous studies have explored how effectively wearable fingertip devices can
render the mass of an object [51, 52, 21, 76, 72, 10]. In most of these studies,
participants pinched virtual objects between their thumb and index finger while
haptic feedback was provided to both fingers. We present a similar scenario in which
participants slide a disk on a flat table, such that we must provide haptic feedback to
only the index finger. Although this scenario is independent of gravity and does not
present the weight of a held object, we still render the inertial forces associated with
accelerating the object. Exploratory motions with acceleration peaks were required
in [21] and were observed without prompting in [76] for conditions in which only
skin deformation was rendered. We note that the forces measured in this study were
much smaller than the weights of the objects, since the measured accelerations were less than 9.81 m/s$^2$. Thus we expected to users to experience a reduced ability to distinguish masses than would be observed if the object was held.

**Slider Haptics**

For our 1-DOF, 3-DOF, and 6-DOF haptic rendering methods, the virtual disk slides on the table only when the virtual fingertip presses into the top of the disk by a required penetration depth, $p$. This condition follows from the assumption that a real disk moves only when static friction between the finger and the disk is greater than static friction between the disk and the table, as determined by the mass, finger force, and coefficients of friction. Moving our virtual disks requires a penetration depth, $p$, of 2 mm for the light disk, 4 mm for the reference disk, and 6 mm for the heavy disk. Once the required penetration depth is met, the virtual slider moves with the finger.

Similar to the hidden surface task, we define $d_V$ as the distance between the top surface of the slider and the virtual fingertip center, and we define $d_H$ as the distance between the haptic contact surface and the real fingertip center. Fig. 3.7 shows the relationship between $d_H$ and $d_V$ for the slider task. This piecewise mapping ensures that no haptic feedback is given before the virtual finger touches the disk and that every participant receives the same 2 mm range of normal displacement between first contact and maximum required virtual penetration depth of 6 mm. For every rendering method, the participant has to press farther into heavier disks and feel larger normal forces in order to move the disk.

Our 6-DOF controller also provides feedback related to the inertia of the disk. We pass the disk’s acceleration through a discrete 5 Hz low-pass filter, to reduce noise associated with our finger pose measurements, and we multiply by the disk’s mass to estimate applied force. For the trials with real disks, participants receive some amount of kinesthetic force feedback in addition to tactile skin stretch feedback. Suchoski et al. reported that the Weber Fraction, or the just noticeable difference in proportion to the reference value, of virtual mass was approximately three times greater for skin deformation feedback than for kinesthetic force feedback [76]. To account for this ratio, we scaled the difference between the virtual masses and the reference mass by a factor of three. After implementing this scaling in a
\[6 < d_V \]
\[(6 - p) < d_V \leq 6\]
\[0 \leq d_V \leq (6 - p)\]
\[d_V < 0\]

\[d_H = 12\]
\[d_H = 6 - \frac{1}{3}(6 - d_V)\]
\[d_H = 6 - \frac{1}{3}(6 - d_V)\]
\[d_H = 4\]

Figure 3.7: In the slider task, the relationship between virtual distance \(d_V\) and haptic distance \(d_H\) is given by a piecewise function with four segments. a) When the virtual finger is far from the disk, no haptic feedback is provided. b) When the virtual finger presses into the disk, but a mass-dependent penetration threshold is not met, haptic normal displacement proportional to penetration is provided and the disk remains static. c) When the penetration threshold is exceeded, haptic normal displacement remains proportional to penetration and the disk moves with the finger. d) Once the virtual fingertip center penetrates the disk, maximum haptic normal displacement is rendered.

In a pilot study, we found that a shear displacement to force ratio of 40 mm/N resulted in the task being difficult, but possible. This ratio of skin stretch to force is much higher than was reported in [76] and [72], but as was discussed in Sec. 3.1, the Fuppeteer’s end-effector is not perfectly rigid. Our effective stiffness accounts for fingertip stiffness and end-effector stiffness in series, resulting in a lower effective stiffness (0.025 N/mm) than for the fingertip shear stiffness alone, which Nakazawa et al. report as approximately 0.2 N/mm when contact force is 1 N [53].

For our natural haptic feedback controller, we provide physical disks for the participant to interact with. Similar to the hidden surface task, we have the subject wear the device, but we remove the end-effector. It is difficult to align a virtual slider with a real slider well enough that it can be effectively manipulated, so we do not have the participant wear the VR headset. We 3D-printed disks with different amounts of infill, such that they are externally identical but have the specified
masses of 13 g, 18 g, and 23 g for the light, reference, and heavy disks respectively. We also attached a layer of 0.3 mm Teflon to the base of each disk to reduce friction, since our virtual representation of this environment did not include friction.

### 3.3.4 Procedure

Each subject first gave informed consent and filled out an initial survey. Twenty people participated in the study, which was approved by the local university’s ethics council. Participation took between 60 and 90 minutes; individuals not employed by our institution were paid a nominal hourly rate for their time. Our participant pool included thirteen females and seven males, with ages between 25 and 56 (mean of 32, standard deviation of 8.6). Their experience with haptic devices included ‘None’ (3 users), ‘Limited’ (8), and ‘Moderate’ (9), and their experience with VR included ‘None’ (2), ‘Limited’ (9), ‘Moderate’ (8), and ‘Extensive’ (1).

The device was worn on the subject’s right index finger across all trials. A visual representation of the subject’s hand was depicted in the virtual environment and followed the user’s real motion. Both tasks took place in a virtual box with a width of 400 mm, depth of 300 mm, and height of 300 mm. We did not physically restrict the user’s motion, but if the user moved beyond the boundaries, the penetrated wall turned red.

The study was broken into four blocks, each associated with one of the four rendering methods. Block order was assigned by a balanced Latin square for every four participants. We are interested in observing how participants naturally use these rendering methods to try to complete the tasks, rather than training participants to employ new exploratory techniques in order to succeed in each specific task. As such, participants were told that each rendering method was unique, but they were not given specific details as to how the methods differed.

Within each block, the participant first donned the Fingertip Puppeteer on their right index finger, and the experiment administrator adjusted the straps such that the fingertip was securely held in the proper location. The subject then put on the VR headset and began the hidden surface task, consisting of two training trials, in which the surface was shown to the participant, and four test trials, in which the surface was invisible. After the hidden surface task, subjects completed the slider task, which again consisted of two training trials and four test trials.
Surface types and slider masses were randomized, although the two training types were not allowed to be the same. Before each trial, the device was automatically calibrated using a trakSTAR sensor embedded in the end-effector, as is described in [98]. Examples of virtual and physical setups for both tasks are shown in Fig. 3.8. Figure 3.9 depicts virtual interactions and corresponding platform poses while a user explores a hidden surface with each of the three tactile rendering methods, and Fig. 3.10 shows frames from a complete hidden surface test trial.

We recorded the duration, accuracy, and motion trajectory associated with each trial. Following completion of the slider task, the participant removed the headset and then fingertip device. They then completed a survey specific to the most recent rendering method, with five-point scales for the realism of each task and the overall comfort of the rendering method. Two free-response questions allowed participants to note any aspects of the rendering method that worked well or poorly. After all four blocks had been completed, subjects were able to provide additional comments in a closing free-response question.
Figure 3.9: Example virtual interactions and the corresponding platform poses rendered via the 1-, 3-, and 6-DOF methods. The 1-DOF platform maintains a steady orientation relative to the finger during all interactions, whereas the 3-DOF and 6-DOF platforms orient themselves to match the slope of the surface. Note that the 6-DOF method also renders shear cues to the fingertip; although these shear forces can be felt, they are difficult to see in these frames.

3.4 Results

We focused our analysis of the study results on performance metrics, motion metrics, and qualitative feedback.

3.4.1 Performance Metrics

Success rates and completion times for each rendering method are shown in Table 3.1. We computed the accuracy for each of the twenty participants within each task and rendering method as an interval variable, and we performed one-tailed
Figure 3.10: A sequence of frames depicting a complete hidden surface test trial, with 6-DOF tactile feedback. The user first lifts the device from the calibration holder. After touching the indicated start location, the user explores the surface. Once the participant believes they know which surface is present, they select the surface from the six options, using an Oculus controller in their left hand. Finally, the participant is guided back to the calibration holder. This cycle repeats until all trials in the task are complete.

one-sample t-tests to compare the responses to chance, with Bonferroni correction. All four rendering methods significantly outperformed chance (16.7%) in the hidden surface task, whereas only the 6-DOF and real rendering methods significantly outperformed chance (50%) in the slider task, with $p < 0.05$.

We also compiled the participants’ responses into confusion matrices for each rendering method during the hidden surface task (Fig. 3.11) and the slider task (Fig. 3.12). Perfect identification would yield a diagonal confusion matrix; the diagonal is almost perfect for real rendering on the surface identification task but
Table 3.1: Success rates for each combination of rendering method and task. A total of 80 trials were completed for each combination. A † indicates performance significantly better than chance ($p < 0.05$), as computed via a one-tailed one-sample t-test with Bonferroni correction.

<table>
<thead>
<tr>
<th>Rendering Method</th>
<th>Surface Accuracy</th>
<th>Time (s)</th>
<th>Slider Accuracy</th>
<th>Time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-DOF</td>
<td>53%†</td>
<td>43.3</td>
<td>51%</td>
<td>21.7</td>
</tr>
<tr>
<td>3-DOF</td>
<td>64%†</td>
<td>38.2</td>
<td>48%</td>
<td>25.8</td>
</tr>
<tr>
<td>6-DOF</td>
<td>60%†</td>
<td>37.2</td>
<td>64%†</td>
<td>24.3</td>
</tr>
<tr>
<td>Real</td>
<td>99%†</td>
<td>20.7</td>
<td>66%†</td>
<td>17.6</td>
</tr>
</tbody>
</table>

not for any of the other confusion matrices. We note that 1-DOF tactile rendering resulted in 14 surfaces incorrectly labeled as the flat surface (20.9% of non-flat surfaces), whereas the 6-DOF method yielded the next highest number of flat surface false-positives at three (4.8% of non-flat surfaces).

For further analysis, we conducted a two-way ANOVA to examine the effect of task and rendering method on success rate. We found a statistically significant interaction between task and rendering method ($F = 4.49, p < 0.01$). We then conducted a one-way ANOVA within each task and found a significant effect of rendering method on success rate for both the hidden surface task ($F = 14.55, p < 0.001$) and the slider task ($F = 2.78, p = 0.047$). In the hidden surface task, Tukey’s post-hoc analysis yielded significant differences between the real rendering method and each of the other three methods ($p < 0.001$), but no statistically significant differences were found between the 1- and 3-DOF methods ($p = 0.48$), the 1- and 6-DOF methods ($p = 0.89$) or the 3- and 6-DOF methods ($p = 0.89$).

In the slider task, Tukey’s post-hoc analysis yielded no statistically significant differences between rendering methods. The 1- and 3-DOF ($p = 0.96$), and the 6-DOF and real ($p = 0.99$) were the pairings least likely to be significantly different, followed by the 1- and 6-DOF ($p = 0.39$), 1-DOF and real ($p = 0.22$), 3- and 6-DOF ($p = 0.17$), and 3-DOF and real ($p = 0.09$).

The mean ratings from qualitative surveys administered after the completion of each rendering method are provided in Table 3.2. Using the Kruskal-Wallis test, we found that hidden surface task realism, slider task realism, and overall comfort were each significantly affected by rendering method ($p < 0.01$). Post-hoc analysis showed that the real rendering method yielded significantly higher
hidden surface realism ratings than each of the other three rendering methods, and significantly higher slider realism and overall comfort ratings than the 1-DOF rendering method ($p < 0.05$). There is a trend indicating the 3-DOF rendering method may yield higher hidden surface realism ratings ($p = 0.07$) and overall comfort ratings ($p = 0.07$) than the 1-DOF method, and there is a trend that the real method may yield higher slider realism ratings than the 3-DOF ($p = 0.06$) and 6-DOF ($p = 0.07$) methods.

### 3.4.2 Exploratory Motions

In general, we observed a wide variety of exploratory techniques during the human-subject study. In the hidden surface task, some participants implemented sweeping horizontal motions (Fig. 3.13(a)), whereas other participants instead favored vertical poking motions (Fig. 3.13(b)). Similarly, in the slider task some subjects vigorously shook the slider side-to-side (Fig. 3.13(c)), whereas others made more precise, slow
Table 3.2: Mean subjective ratings for each rendering method. Realism for each task was rated on a numeric scale from very unrealistic (1) to very realistic (5), and overall comfort was rated from very uncomfortable (1) to very comfortable (5).

<table>
<thead>
<tr>
<th>Rendering Method</th>
<th>Surface Realism</th>
<th>Slider Realism</th>
<th>Overall Comfort</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-DOF</td>
<td>2.95</td>
<td>2.95</td>
<td>3.00</td>
</tr>
<tr>
<td>3-DOF</td>
<td>3.95</td>
<td>3.30</td>
<td>3.65</td>
</tr>
<tr>
<td>6-DOF</td>
<td>3.70</td>
<td>3.35</td>
<td>3.50</td>
</tr>
<tr>
<td>Real</td>
<td>4.85</td>
<td>4.2</td>
<td>3.95</td>
</tr>
</tbody>
</table>

We measured the total horizontal and vertical distances travelled by the user’s finger, which we normalized with respect to trial duration, resulting in average horizontal and vertical speeds for each trial. We also computed the horizontal and vertical accelerations by passing the cubic interpolation of the position data through a 5 Hz low-pass filter and taking the second derivative. The horizontal and vertical speeds and accelerations across all trials are shown in Fig. 3.14. Finally, to highlight the variation between participants, we show the vertical speeds and horizontal accelerations for each participant during the hidden surface and slider tasks, respectively, in Fig. 3.15.

We performed a one-way repeated measures ANOVA with Tukey post-hoc analysis on the motion measurements within each task. In the hidden surface task, we found a significant effect of rendering method on horizontal speeds ($F = 11.21, p < 0.001$); the real rendering method prompted larger horizontal speeds than each of the other three methods ($p < 0.01$). In the slider task, we found a significant effect of rendering method on vertical speeds ($F = 11.35, p < 0.001$) and that the real rendering method resulted in smaller vertical speeds than each of the other methods ($p < 0.01$). No other individual comparisons were statistically significant.

We also looked at the relationship between exploratory motions and performance, but we found no significant differences between the speeds or accelerations implemented during successful trials versus unsuccessful trials. Across all trials for both tasks, we measured overall speeds (mean ± standard deviation) of $0.109 ± 0.050$ m/s during successful trials and $0.110 ± 0.046$ m/s during unsuccessful trials. Looking more closely at slider task trials with 6-DOF tactile feedback,
during which horizontal acceleration directly affects the amount of tactile shear rendered to the user, we measured horizontal accelerations of $0.798 \pm 0.524 \text{ m/s}^2$ during successful trials and $0.858 \pm 0.506 \text{ m/s}^2$ during unsuccessful trials.

### 3.4.3 Qualitative Feedback

Answers to the free-response questions largely upheld the quantitative results. Seventeen comments specified that the hidden surface task was well-rendered by the 3-DOF and 6-DOF tactile methods, with five comments directly mentioning the ability to feel the ‘slope’ or ‘curvature’ of the surfaces. Two positive comments for the 6-DOF method noted the ‘continuous’ feel of the surfaces (P2, P5), whereas three negative comments for the 1-DOF method noted the absence of a ‘continuous’ feeling (P2, P5, P6). For the 6-DOF rendering, five participants stated that
the slider task felt somewhat easier, with comments such as “I had the feeling I was now able to better determine... what disc was heavier, but only slightly” (P12). Overall, participants consistently noted that they struggled during the slider task, with every rendering method (including real haptic feedback) receiving from four to eight comments related to slider task difficulty.

Free-response answers also provide additional insight into components of the study that were not directly measured. Two subjects disliked the virtual finger being able to penetrate the virtual surface: “I didn’t like that the virtual hand was able to go through the virtual hidden surface...” (P1) and “the finger crossed the surface visually” (P5). A third user noted a shortcoming of the system was that there was “no modeling of infinite resistance like for real surfaces” (P6).

Despite our efforts to minimize weight supported by the user, some participants still commented on the device’s mass. Two participants described discomfort caused by weight: “The device is a bit heavy for me” (P4) and “the weight was a bit too high” (P5). Three participants noted that the weight of the device may have affected performance (P2, P6, P19), and one attributed blame to the counterweight itself: “I think that the floating counterweight made it harder for me to distinguish how hard I was pushing on the slider” (P17).

While administering the study, we observed multiple subjects who spent more
Figure 3.15: Vertical speeds for each participant during the hidden surface task (left) and horizontal accelerations for each participant during the slider task (right) across all trials.

Time exploring the edges of the hidden surfaces when interacting with the real physical surfaces (real haptic rendering) than when the surfaces were virtually rendered by the 1-, 3-, or 6-DOF methods. A few participants lamented that edges were not rendered for virtual surfaces: “The edges of the surfaces were not very clear.” (P18), and “I couldn’t feel the edges of the surfaces for the hidden surface task” (P13). Previous works have noted that slowly-adapting type I (SA-I) [32] and fast-adapting type I (FA-I) [31] afferents in the fingerpad are sensitive to edge contours, possibly providing more salient information to the user. In this experiment, we propose two additional reasons users may choose to explore edges. First, the most extreme
heights of these quadric surface are located at the edges, making points along the edges ideal candidates for exploration in order to identify a surface based on height discrimination. Secondly, each edge has one defined coordinate in the horizontal plane, leaving the user to estimate only two spatial dimensions of their fingertip rather than three. Since the six surfaces are distinguishable by any edge, exploring the edges effectively reduces this shape discrimination task from three-dimensional to two-dimensional.

Although these critical comments may be more interesting to the reader and better motivate design decisions for future tactile fingertip feedback systems, we also received many written and verbal comments expressing an overall pleasant experience with the tested system, some of which we highlight here: “Good resolution of force.” (P6, 1-DOF), “I really enjoyed the hidden surface task” (P4, 6-DOF), and “I really like the detail where the device swivels around the tip of the finger!” (P17, closing).

3.5 Discussion

In the hidden surface task, we recorded higher mean success rates, median completion times, and mean realism ratings for the 3- and 6-DOF rendering methods than for the 1-DOF method, which fits with prior literature that suggests haptic curvature perception is dominated by local surface orientation [65, 90]. Non-flat surfaces were also incorrectly labeled flat over four times as frequently with the 1-DOF method than the other methods. However, all four of the rendering methods yielded surface identification rates better than chance, and none of the pairwise differences between success rates of the three tactile-only rendering methods were statistically significant. These findings suggest that there is minimal benefit from providing higher-dimensional tactile feedback in the absence of kinesthetic force feedback for this particular task. The large extent of the surfaces may have enabled reasonably accurate recognition simply through recognition of the spatial regions that cause fingertip contact, the cue that is provided even by the simplest 1-DOF rendering scheme. The real rendering method significantly outperformed the three tactile-only rendering methods, suggesting that kinesthetic force feedback, which was presented in each of the past works cited as motivation [36, 40, 42, 65, 90], is superior to tactile feedback alone for this shape discrimination task.
In the slider task, success rates for the 1-DOF and 3-DOF rendering methods were approximately equal to chance (50%), suggesting that participants did not notice any difference between the required penetration depths. The 6-DOF and rendering method resulted in success rates greater than chance, showing that the relationship between shear skin stretch and inertia was more easily perceived by users. However, subjects still had difficulty during the slider task with the 6-DOF and real rendering methods, as indicated by multiple free responses and success rates of less than 70% for all conditions. As mentioned in Section 3.3.3, we expected this task to be more difficult than a mass discrimination task where users are allowed to hold the objects and feel weight in addition to inertia. It’s possible that mass discrimination was further impeded by the mass of the device itself, which was large relative to the mass of the virtual and real disks. The similarity of the success rates for the 6-DOF and real rendering methods in the slider task suggests that our shear representation was reasonably realistic, despite the incomplete modeling of end-effector stiffness as a function of configuration.

Interestingly, exploration techniques were not heavily influenced by rendering method but instead seem to depend more strongly on individual preference. For example, P13 consistently applied relatively large horizontal accelerations to the slider, even when no tactile representation of inertia was provided, whereas P11 consistently applied small horizontal accelerations to the slider, even when skin stretch proportional to acceleration was rendered. Although participants were only able to perform mass discrimination at rates above chance when inertia was presented through tactile shear, the presence of this shear did not elicit larger accelerations or any other change in exploratory motion. Naive users seemingly expected the tactile fingertip feedback to accommodate their exploratory tendencies, and as such, subjects did not adjust their motions to match the provided tactile sensations.

Success rates were also independent of exploratory motion, as users implemented nearly identical overall fingertip speeds during successful and unsuccessful trials. If system latency hindered the haptic rendering, we would expect performance to degrade when users move more quickly. Instead, performance is consistent regardless of user speed, suggesting participants are able to intuitively understand and compensate for any delays between virtual interaction and tactile feedback. During the 6-DOF rendering of the slider task, when tactile shear is directly related to user acceleration, we still do not find a significant effect of acceleration on performance.
It is possible that during these interactions in which we expect large accelerations to yield better results, latency has a more negative impact and thus offsets the expected benefits of high-acceleration exploratory techniques.

Even when real tactile and kinesthetic feedback was provided, we measured significant differences in exploratory motion only with respect to horizontal speeds during the hidden surface task and vertical speeds during the slider task. The difference in vertical speeds during the slider task is expected, since the finger is unable to penetrate the real disks, which remained at a constant height. A similar rationale may explain the increase in horizontal speed during the hidden surface task, as participants may focus on horizontal exploration when kinesthetic feedback limits vertical exploration. However, the increase in horizontal speed with real feedback may also be attributed to greater comfort or confidence while exploring the surface, or less concern regarding the bandwidth of the system. Multiple users commented that the weight of the device influenced their perception, suggesting that higher velocities and accelerations might have been measured if the device was lighter.

Although we were surprised that users did not adjust exploratory techniques depending on the haptic rendering method, we should note that most of the implemented strategies can indeed provide enough information to successfully complete the tasks. The two most common exploratory techniques for the hidden surface task, as highlighted in Figure 3.13, were horizontal exploration and vertical poking. Through horizontal exploration, a user can first position their finger at an arbitrary height, and then by carefully noting the horizontal locations at which contact is made or broken, find the contour line at which the surface has the specified elevation. For the six surfaces presented in this study, the contour lines for each surface are unique at any height other than 0 m, allowing identification. Through vertical poking, users can instead determine the height of the surface above particular locations on the table, creating a mental 3D point cloud that can be used to identify the surface. A particularly effective strategy may be to locate the center of the surface, then poke one location left or right of the center and one location behind or in front of the center to determine the curvature in each direction, allowing the surface to be identified.

The most common exploratory techniques for the slider task, again shown in Figure 3.13, were vigorously shaking the slider and gently pushing the slider. Although
larger accelerations yield larger absolute differences in shear forces for sliders with
different masses, smaller accelerations still elicit different shear forces and may even
reduce noise associated with the inertia of the device itself. For real disks, precise
motions may also allow more careful observation of how much force is required to
overcome static friction, which users may have expected in the virtual task as well.

However, some exploratory techniques offered no information that could be used
to complete the task, in both the virtual and real environments. For example, some
participants poked downward on the slider rather than moving it, suggesting either
confusion between the physical properties of mass and stiffness, or an expectation
for virtual mass to be haptically rendered as virtual stiffness. Interestingly, two
participants poked the real sliders at least once. The perceptual confusion between
mass and stiffness provides additional insight to the findings of Schorr et al. [72],
who previously reported that a change in virtual mass was sometimes perceived as
a change in stiffness or friction.

Answers to the free-response questions provide deeper understanding of the
strengths and shortcomings of the tactile rendering methods and our overall system.
Free-responses that commented on discomfort caused by the virtual finger passing
through the virtual surface imply a desire for, at a minimum, visual kinesthetic
feedback, which is also often called pseudo-haptics [43, 29]. One comment noting
the ‘lack of infinite resistance’ reaffirmed the request for kinesthetic force feedback.
Other featured comments suggested potential improvements related to reducing the
mass of the device or enabling the system to render edges as well as faces.

Importantly, the study instructions provided no information as to how the ren-
dering methods differed, or how each rendering method could be used to complete
the tasks. This approach allowed us to observe naive performance, but it also proved
challenging for some participants who never discovered exploration techniques that
yielded positive results. If the tactile rendering methods were explained, or if sug-
gested exploratory techniques were provided, users would most likely be better able
to discern which techniques work best for a particular task and rendering method,
rather than consistently trying the same technique. Alternatively, more exposure
time and feedback about answer correctness would most likely have enabled users
to discover better techniques on their own but would have increased study duration
and fatigue.
3.6 Limitations and Future Considerations

Our user study was limited to two virtual tasks. Although we selected two tasks expected to elicit different fingertip sensations, such a limited set of environments clearly fails to represent all haptic fingertip interactions. A clear continuation of this work is to incorporate additional virtual interactions. For example, rotational shear provides no intuitive benefit in completing the hidden surface task or the slider task, but rotational skin stretch on the fingertip may come into play for transferring objects, such as the third example in Fig. 1.1, or estimating a pinched object’s center of mass. A related possible continuation is providing a wider range of vari-dimensional tactile feedback. In the slider task, we observed performance better than chance with only 6-DOF tactile feedback or real haptic feedback. However, we hypothesize that the two DOFs related to translations shear were more beneficial to this task than the two DOFs related to fingertip contact location. A 3-DOF device that can render translational shear and variable normal displacement at a single location on the finger may yield similar performance to the 6-DOF device, yet require only three actuators. Furthermore, our mass discrimination task required users to distinguish disks based on inertia rather than weight, and we noted that the forces associated with our measured accelerations were much smaller than the weights of the disks. A similar task in which users are asked to slide a disk up a vertical wall would allow both weight and acceleration to contribute to the haptic sensation. Exploring additional interactions and subsets of 6-DOF tactile cues may further improve understanding of how tactile feedback dimensionality affects user performance and exploration during common interactions.

Using the 6-DOF device to render virtual interactions also highlighted potential improvements to the device design. Although the rolling mechanisms used to actuate each wire work reasonably well, the virtual interactions presented in this chapter highlight the potential benefit of finding more efficient methods to account for or eliminate wire slip. The calibration process was successfully completed after each trial, but navigating to the calibration pose via virtual visual cues was a time-consuming and nontrivial task, distracting from the primary goals of this research. Reducing the reliance on this calibration procedure would also allow longer virtual interactions to be rendered with this device. One solution may be to measure the wire positions directly, allowing a control loop to minimize wire position errors di-
rectly rather than roller orientation errors. Alternatively, an improved design may be able to reduce slip such that significant error no longer accumulates. Etching notches into the wires, similar to a rack and pinion design, may allow more precise positioning, although we expect this would change the mechanical properties of each wire. Coating the wires in a flexible, high-friction substance or selecting a different roller material may instead reduce slip with minimal effect on the mechanical properties used in the simulation.

The results of this chapter add additional value to the discussion in Section 2.9 related to potential ways to reduce the mass and volume of the Fuppeteer. A few user comments and general observation during the interactive virtual study suggested that the weight of the device affected exploration and performance, despite the counterweight effectively reducing user fatigue. In addition to making the device more wearable and allowing prolonged haptic interactions, redesigning the 6-DOF device with an emphasis on being lightweight and compact may allow users to interact more freely with virtual objects, eliciting more natural exploratory motions. We also noted in Section 3.2 that the counterweight attachment point induced a small moment on the device, pushing it toward a horizontal orientation. Moving this attachment point to the center of mass or otherwise reducing the counterweight-induced moment would also allow users to more naturally explore virtual environments.

3.7 Summary

We integrated the 6-DOF fingertip tactile device presented in Chapter 2 into an immersive VR system with haptic feedback, and we programmed it to render feedback as would be provided by devices with varying degrees of freedom. We presented rendering algorithms for 1-, 3-, and 6-DOF tactile cues during a shape discrimination and a mass discrimination task. Results from 20 naive study participants show that higher-dimensional tactile feedback may indeed allow completion of a wider range of virtual tasks, but that feedback dimensionality surprisingly does not greatly affect the exploratory techniques employed by the user. Furthermore, the shape discrimination task demonstrated that for some interactions, there is only marginal benefit in providing higher-dimensional tactile feedback. This finding is further supported in Chapter 4 and motivates the continued use of fingertip tac-
tile devices with fewer degrees of freedom in particular applications. In order to improve performance of such devices with fewer DOF, in the next chapter we explore software-based approaches to personalization, rather than increased tactile expressiveness.
Chapter 4

Rendering Personalized Remotely Sensed Cues

In Chapter 1, we presented motivation for the our goal of providing personalized fingertip tactile feedback. Given the variability of human fingertip characteristics and the human tactile sensitivity to contact sensations, we strongly believe the unique characteristics of each fingertip should be taken into account when designing fingertip interfaces and rendering algorithms (see Fig. 4.1). The goal of personalization is further motivated by Chapter 2, where we noted that the user’s finger size influenced their ability to accurately perceive haptic cues that were rendered the same way for everyone. This finding suggests the need to account for differences in users’ physical finger characteristics, particularly if we want users to perceive cues in a similar manner. Yet haptic rendering algorithms traditionally do not account for differences between users, incorrectly assuming that sensations are consistent for everyone [63, 46, 9].

We have demonstrated that fingertip devices with more degrees of freedom, such as the Fuppeteer, offer unique advantages. In Chapter 3 we noted that higher-dimensional tactile feedback allows completion of a wider range of virtual tasks. In another previous work, we presented a framework for estimating the minimum number of actuators required to render a particular tactile interaction, without sacrificing haptic rendering quality [61]. Although the details of this work are not included in this thesis, we showed that in some cases, a reduction in the number of actuators used to render a tactile interaction will result in substantially greater
haptic rendering error.

However, we have also seen that not every haptic experience can be improved through higher-dimensional tactile rendering. In Chapter 3, we found minimal benefit in providing higher-dimensional tactile fingertip feedback compared to lower-dimensional feedback during the presented shape discrimination task. Similarly, in our previous work [96] we quantitatively confirmed that there is no substantial benefit in rendering some tactile interactions with more than one or two degrees of freedom. Improved rendering of such interactions cannot be achieved through increased expressiveness of the tactile device and instead require an alternative approach, prompting us to focus our first attempts at personalizing fingertip cues on devices with fewer degrees of freedom. Thus to develop and evaluate novel approaches toward providing personalized tactile rendering, we turn our attention to a 3-DOF wearable device. We note that the software-based solutions developed in this chapter can be extended to the 6-DOF device demonstrated in Chapters 2 and 3, and we believe this would be a worthwhile path for continued work.

This chapter proposes two methods to personalize the haptic rendering algorithm of a fingertip haptic device for fingertips of different size. Each algorithm has unique benefits and drawbacks; they both modify a known data-driven rendering algorithm presented in Sec. 4.1, changing its core characteristics to account for the geometry of a target fingertip, and they are compatible with any mechanical design of a fingertip device and any fingertip sensor. We present our system in Sec. 4.2 and our personalization algorithms in Sec. 4.3. We evaluate our algorithms in the rendering of pre-recorded tactile sensations onto rubber casts of six different
Figure 4.2: Our 3-DOF fingertip device modified to house a device-mounted sensor (DS) on the mobile platform. This sensor is needed only during the data collection phase. The device is shown being worn by (a) our fingertip sensor (FS) and (b) a human user. The FS is suspended in the device such that any force exerted by the mobile platform is fully transferred into the sensor.

4.1 Related Work

We start from the standard data-driven haptic rendering algorithm presented by Pacchierotti et al. [57]. Although this algorithm ignores the viscoelastic components of the fingertip response, prior work showed that the interaction sequences generated by the algorithm received high ratings from participants. They used a setup similar to ours, with a 3-DOF fingertip device and a fingertip-like BioTac sensor. First, the...
BioTac sensor is placed inside the fingertip device in the same way a human user would wear it (similar to Fig. 4.2(a)). Then, the end-effector of the device is moved to a wide range of configurations, and the effect of each of these configurations is measured by the sensor. This data collection process generates a lookup table, expressed with a mapping function

\[ \mu(f_s) = m_s, \quad (4.1.1) \]

where \( m_s \) is a vector of motor inputs (three, in this case) commanded during the data collection process and \( f_s \) is the corresponding fingertip sensor outputs (twenty, in the case of the BioTac, and six for our FS).

To render a new tactile sensation \( f \) registered by the sensor in a remote environment, Pacchierotti et al. [57] searched the lookup table to find the \( n \) most similar sensor outputs measured during data collection, \( \tilde{f}_s \). Let us define this search function as

\[ \nu_n(f) = [f_{s,1} \cdots f_{s,n}]^T = \tilde{f}_s. \quad (4.1.2) \]

They then mapped each element of \( \tilde{f}_s \) back to its corresponding motor inputs, using the lookup table,

\[ \mu_n(\tilde{f}_s) = \mu_n \begin{bmatrix} f_{s,1} \\ \vdots \\ f_{s,n} \end{bmatrix} = \begin{bmatrix} \mu(f_{s,1}) \\ \vdots \\ \mu(f_{s,n}) \end{bmatrix} = \begin{bmatrix} m_{s,1} \\ \vdots \\ m_{s,n} \end{bmatrix} = \tilde{m}_s. \quad (4.1.3) \]

Finally, the \( n \) motor inputs \( \tilde{m}_s \) were averaged into one, weighting each input \( m_{s,i} \) according to the distance between the original sensation \( f \) and the one elicited by \( m_{s,i} \) during data collection, \( f_{s,i} \) [57],

\[ \varphi_n(\tilde{m}_s) = \varphi_n \left( \begin{bmatrix} m_{s,1} & \cdots & m_{s,n} \end{bmatrix}^T \right) = \tilde{m}. \quad (4.1.4) \]

where \( \tilde{m} \) are the final motor commands, expected to recreate \( f \) on the user’s fingertip. The final mapping, from a generic sensor sensation \( f \) to the device’s motor inputs \( \tilde{m} \), is

\[ \Lambda_n(f) = \varphi_n(\mu_n(\nu_n(f))) = \varphi_n(\mu_n(\tilde{f}_s)) = \varphi_n(\tilde{m}_s) = \tilde{m}. \quad (4.1.5) \]
Although this algorithm showed good performance [57] and was also employed in a robot-assisted palpation task [58], it has one major drawback; it assumes that the user’s fingertip is shaped exactly like the sensor used during the data collection, i.e., the BioTac in their case. As such, the more the user’s fingertip differs from the BioTac, the worse the algorithm performs. A representative example of this behavior is shown in Fig. 4.1. Even small errors in relative position can lead to differences in perception, due to the high sensitivity of some mechanoreceptors to local spatial discontinuities. The fingertip is densely populated with both FA-I afferents with Meissner endings and SA-I afferents with Merkel endings, which are sensitive to high-frequency and low-frequency local skin deformations, respectively [31].

4.2 Fingertip Device and Sensor Design

For this work, we chose a three-degree-of-freedom (3-DOF) fingertip device similar to that of [58, 57, 55] and shown in Figs. 4.1 and 4.2. It is composed of a static platform grounded on the back of the fingertip and a moving platform located opposite the fingertip and acting as the end-effector. The two platforms are connected by three cables enclosed by three springs, each of which is driven by a servo motor (HS-55) mounted to the static platform. The device is able to provide the sensations of making and breaking contact with slanted surfaces as well as pressing on them. We also built a modified version of this device that is able to house a device-mounted sensor (DS), in our case a Nano17 force/torque sensor, on the mobile platform (see Fig. 4.2). This additional sensor is required for our data-driven personalization approach, which will be discussed in Section 4.3.1.

Similar to [58, 57], we use a fingertip-shaped sensor (FS) to record remote and rendered interactions. It consists of a 3D-printed rigid core attached to an ATI Nano17 sensor on one side and embedded in a VytaFlex rubber fingertip cast on the other (see Fig. 4.3). The tip of our FS is a quarter sphere with a radius of 7.5 mm; it registers six-dimensional force/torque readings. Although existing sensors offer greater sensing richness, our evaluation presented in Sec. 4.4 required us to create a fingertip sensor with custom geometry, leading us to choose this sensor setup instead. Even though we present and test our two personalization approaches using the above haptic system, both are compatible with any other similar fingertip device.
4.3 Approaches for Personalization

We aim to personalize the lookup table described above, so as to account for the unique shape of each user’s fingertip. We propose two approaches, each with its benefits and drawbacks. Fig. 4.4 shows the block diagrams summarizing the standard non-personalized approach and our two adjustments.

and sensor.
4.3.1 Data-driven personalization

We start by using our haptic setup to run an FS-centered data collection, similar to what we described in Sec. 4.1. The FS is placed inside the device, and DS is placed on the mobile platform, as shown in Fig. 4.2(a). Then, we move the end-effector of the device to a wide range of configurations, and the effect of each of these configurations is registered on the two sensors (FS and DS). We can now define mapping a function similar to eq. (4.1.1),

$$\gamma(f_*) = d_*$$,  \hspace{1cm} (4.3.1)

where $f_* \in \mathbb{R}^6$ is an FS output and $d_* \in \mathbb{R}^6$ is the corresponding DS output, both measured for each motor configuration $m_* \in \mathbb{R}^3$.

To enable personalization, we carry out a second, user-centered data collection routine. We remove the FS and ask the target user to wear the device, as shown in Fig. 4.2(b). Then, we move the end-effector of the device to a second wide range of configurations, and we record the effect of each of these configurations using the DS. To spare the user’s time, the user-centered data collection can include a smaller set of configurations than the FS-centered data collection. From this second data collection, we define our user personalization function as

$$\alpha_U(d_{*,U}) = m_{*,U}$$, \hspace{1cm} (4.3.2)

where $m_{*,U}$ is a vector of motor inputs and $d_{*,U}$ is the corresponding sensation registered by the DS, for a particular user, $U$.

Now we have all the information needed to adjust the lookup table defined by eq. (4.3.1) for our target fingertip. To do so, for each DS sensation registered during the FS-centered data collection $d_*$, we find the $n$ nearest DS sensations observed during the user-centered data collection $\tilde{d}_{*,U}$, i.e.,

$$\rho_n(d_*) = \left( [d_{*,U,1} \ldots d_{*,U,n}]^T \right)^T = \tilde{d}_{*,U}$$, \hspace{1cm} (4.3.3)
and then we retrieve the corresponding motor inputs, defining

$$\alpha_{U,n}(\tilde{d}_{s,U}) = \begin{bmatrix} \alpha_{U}(d_{s,U,1}) \\ \vdots \\ \alpha_{U}(d_{s,U,n}) \end{bmatrix} = \begin{bmatrix} m_{s,U,1} \\ \vdots \\ m_{s,U,n} \end{bmatrix} = \tilde{m}_{s,U}$$

(4.3.4)

as the function that maps a set of $n$ DS sensations registered during the user-centered data collection $\tilde{d}_{s,U}$ to the set of $n$ motor inputs $\tilde{m}_{s,U}$ that elicited them. We now find the weighted average of these $n$ adjusted motor inputs, as in eq. (4.1.4),

$$\phi_n(\tilde{m}_{s,U}) = \phi_n \left( \begin{bmatrix} m_{s,U,1} & \cdots & m_{s,U,n} \end{bmatrix}^T \right) = \hat{m}_s.$$  

(4.3.5)

Using this process, we can map every FS sensation observed during the FS-centered calibration, $f_s$, to a personalized motor command $\hat{m}_s$. The complete adjustment for our data-driven approach is

$$\hat{\mu}_D(f_s) = \phi_n(\alpha_{U,n}(\rho_n(\gamma(f_s)))) = \phi_n(\alpha_{U,n}(\rho_n(d_s))) = \phi_n(\alpha_{U,n}(\tilde{d}_{s,U})) = \phi_n(\tilde{m}_{s,U}) = \hat{m}_s,$$

(4.3.6)

which can be substituted for $\mu(\cdot)$ in eq. (4.1.3) to give the full rendering algorithm with data-driven personalization. In this chapter, we consider $n = 8$, as it was proven to provide the best rendering performance [57].

This first approach for rendering personalized haptic cues is quite general and easy to use, as it is completely data-driven and does not require any additional knowledge about the device or the user’s fingertip. All the information this approach needs is retrieved during two data collection processes; we expect this approach to inherently account for the stiffness and elasticity of the fingertip to render similar forces and/or pressure distributions to different fingertips. On the other hand, this approach requires that a rich haptic sensor be placed in the moving platform during data collection and that each user undergoes one data collection routine.

### 4.3.2 Geometric personalization

Our second approach uses the geometry of the target fingertip to personalize the FS-centered lookup table. This approach’s objective is to adjust the behavior of
the mobile platform so as to replicate on the user the deformation generated on the FS's rubber surface during data collection. Similar to Sec. 4.3.1, we define a new mapping function
\[ \tilde{\mu}_G(f_\star) = \varepsilon(\mu(f_\star)) = \varepsilon(m_\star) = \hat{\mathbf{m}}_\star, \] (4.3.7)
where \( \varepsilon(\cdot) \) adjusts each motor input registered during the FS-centered data collection. The complexity of \( \varepsilon(\cdot) \) is arbitrary and depends on the information available about the target finger. Here we employ a simple 1-DOF model of the fingertip, which we simply define by a sphere of radius \( r \). As discussed in Sec. 5.2, future work may consider more complex geometric approximations.

We start by running an FS-centered data collection, similar to Secs. 4.1 and 4.3.1, with the FS placed inside the device as in Fig. 4.1(a). No additional device-mounted sensor is needed for this approach. Let us consider a representative configuration of the device, with the mobile platform tilted upward on the left and pressing on the sensor. Fig. 4.5a shows a 2D front view of the device in this configuration during data collection, while Fig. 4.5b shows a 2D front view of the device in the target personalized configuration when worn by the user. Let \( S_0 \) be the reference frame fixed on the centroid of the static platform and \((x_0, y_0, z_0)\) be its unit vectors. All the following positions are referenced with respect to \( S_0 \) unless otherwise stated. Similarly, let \( S_1 \), with unit vectors \((x_1, y_1, z_1)\), be the reference frame fixed on the centroid of the mobile platform and let \( \mathbf{p}_c \) be the position of this centroid. We approximate the FS with a sphere of radius \( r_F = 7.5 \text{ mm} \), centered at \( \mathbf{p}_F \). For each device configuration, our objective is to adjust the position of the platform relative to the user’s fingertip so as to render to the user (Fig. 4.5b) the same absolute deformation applied to the FS (Fig. 4.5a). Let \( \hat{S}_1 \) be the reference frame fixed on the adjusted mobile platform’s centroid, and let \( \hat{\mathbf{p}}_c \) be the position of this centroid. We approximate the user’s fingertip with a sphere of radius \( r_U \), centered at \( \mathbf{p}_U \). We assume that the FS and user fingertip are positioned such that \( \mathbf{p}_F = \mathbf{p}_U \). In the representative case of Fig. 4.5, the platform needs to be moved up to ensure a match in the target deformation because \( r_U < r_F \), but the following procedure can be applied to a fingertip of any size.

Since our data collection provides a lookup table of sensations \( f_\star \) elicited by certain motor actions \( m_\star \), we first convert each \( m_\star \) to the corresponding mobile platform position and orientation, given by \( \mathbf{p}_c \) and \( S_1 \), respectively. To convert
Device worn by the FS sensor during data collection.

Device worn by a user with a fingertip smaller than the FS.

Figure 4.5: 2D device schematic (front view) of a representative configuration. The blue segment represents the static platform of the device, the green segment the mobile platform, the red disk the fingertip-shaped sensor (FS), and the yellow disk the subject’s fingertip. To match the same deformation (i.e., $d_U = d_F$), the mobile platform needs to be moved up from its position in (a).

between motor commands and mobile platform pose, we assume the three cables remain parallel to the $z_0$ axis and define the contact surface as the plane that intersects the bases of these three cables. However, the equations in this section hold for any device, so long as the forward and inverse kinematics can be computed.

Next, we calculate the point $p_o$ at which the undeformed surface of the FS is parallel to our mobile platform,

$$p_o = p_F - r_F \cdot z_1. \quad (4.3.8)$$

We are then able to estimate how much and in which direction the mobile platform deforms the sensor,

$$d_F = p_c - p_o. \quad (4.3.9)$$

The magnitude of this deformation is zero when the platform barely touches the sensor, and it increases as the platform applies more pressure. To recreate the same deformation as was rendered to the FS, we define our desired finger deformation $d_U = d_F$, and we estimate the orientation and position of the centroid of the adjusted platform,

$$\hat{S}_1 = S_0, \quad \hat{p}_c = p_U - r_U \cdot \hat{z}_1 + d_U. \quad (4.3.10)$$

Finally, we convert our adjusted mobile platform orientation and position into
adjusted motor inputs \( \hat{m}_s \). By applying this procedure to each element \( m_s \) registered during the data collection, we can generate a new personalized lookup table.

This second approach for rendering personalized haptic cues is rather fast and easy to implement, as it does not require any additional sensor on the mobile platform, and more importantly it does not require the user to carry out any data collection process. Moreover, it is quite flexible, as it can be implemented with arbitrarily complex fingertip models. However, of course, this approach is only as good as the fingertip approximation chosen and the model of the device employed. This preliminary fingertip approximation is entirely geometric in nature, and thus it does not account for the stiffness of the fingertip, which almost certainly varies between a sensor and a user. We expect this approach to be most beneficial for light contact, where relatively small fingertip deformations are required.

4.4 Objective Evaluation

We first quantitatively evaluated the performance of our personalization approaches on six sensorized rubber fingertips.

We built two fingertip-like casts in addition to the one we used for the FS, following the same procedure presented in Sec. 4.2. One is 13% smaller than the FS and has a spherical tip with radius 6.5 mm, and the other is 13% larger than the FS and has a spherical tip with radius 8.5 mm. We also constructed three casts of the authors’ fingertips using a slightly different procedure that is shown in Fig. 4.6. We approximate each human cast with a sphere of radius \( r \), defined as the distance from the middle of the rigid core to the bottom of the fingertip. We measured radii of 6.5 mm, 8 mm, and 10 mm for the three casts of human fingertips. Although they have mechanical properties that are not identical to those of human fingertips, we believe these rubber fingertips can provide a preliminary and objective evaluation of how our personalization approaches affect rendering error. Finally, each of these six rubber fingertips can be attached to a Nano17 sensor (similar to Fig. 4.3b) and inserted into our fingertip device (similar to Fig. 4.2(a)).

Next, we built an FS-centered lookup table, as described in Sec. 4.3. To do so, we placed the FS inside the device and the DS below the mobile platform, as shown in Fig. 4.2(a). Then, we moved the motors to \( 13^3 \times (2,197) \) different configurations \( m_s \) that were evenly distributed throughout the entire motor space of the device. Each
Figure 4.6: The process of creating a cast of a human fingertip. We first make a mold out of Kulzer’s Flexitime Correct Flow; then we insert a rigid core and fill the void with Smooth-On VytaFlex 30 to make the fingertip cast. The rigid core is later attached to a Nano17, as in Fig. 4.3(b), creating a sensor that outputs a six-element vector of the forces and torques acting on the fingertip.

...
Figure 4.7: Objective evaluation results. Box plots of rendering errors for each combination of fingertip cast and rendering approach. For each target cue, the error is defined as the magnitude of the difference between the force component of the target sensation, \( f \), and the force rendered to the test fingertip. Asterisks denote differences that are statistically significant, with \( p < 0.0001 \).

Cues rendered to the FS using the standard method had a median force error magnitude of 0.17 N. Rendering the target sensations \( f \) to the FS requires no personalization, as the FS-centered lookup table is already built for the FS. Therefore, this error represents our baseline (i.e., the lowest error we can expect even with ideal personalization). The rendering error associated with the standard approach was significantly smaller for the FS than for all other fingertips, as expected. The errors associated with each of the six cast fingertips and each of three rendering approaches are shown in Fig. 4.7.

For all of the tested cast types, there was a significant effect of the rendering approach (Friedman statistical test with \( p < 0.0001 \) for all cast types). The post-hoc analysis showed that the data-driven and geometric personalizations significantly reduced the error compared to the standard approach for all of the fingertip casts other than the FS (Wilcoxon matched pairs signed rank tests with \( p < 0.0001 \) while Bonferroni correction required \( p < 0.0028 \)). Furthermore, we compared the two personalization methods across all conditions, and we found the force error magnitude of the data-driven approach was significantly smaller than that of the geometric
one (Wilcoxon matched pairs signed rank test, $W = 22682, p = 0.0076$). The difference was especially large for the cast of the large human fingertip, for which the geometric personalization had much higher errors than in other conditions.

### 4.5 Human-subject Evaluation

We also performed a human-subject study. The experimental setup was nearly identical to that presented in Sec. 4.4 and was inspired by [57], in which participants rated the similarity between tactile interactions they saw and tactile interactions they felt. To generate the user-centered lookup table, we asked the participants to wear the fingertip device on their right index finger, as in Fig. 4.2(b). For the sake of their comfort, we reduced the set of motor configurations $m_{s,U}$ to $5^3$ (125), which takes roughly five minutes to collect. Subjects were isolated from external noise through a pair of headphones, and their vision of their fingertip in the device was blocked by a wooden panel. Then, we placed a second device with the FS inside, as in Fig. 4.2(a), right in front of them. During each trial, we simultaneously rendered one sensation to the subject’s hidden finger (H) and one to the visible FS (V). Each sensation was maintained until the user rated the similarity of the cue they felt on their fingertip to the one they saw on the FS. After that, both devices returned to a neutral pose with no contact. The rating was given on a number pad that ranged from 1 to 9, where a score of 1 meant “very different” and a score of 9 meant “very similar”.

We selected ten target sensations from the original 100 in Sec. 4.4 that delivered forces of less than 1 N to the FS, so as to evaluate our methods on light-contact interactions that are more difficult to render. Each cue applied on the visible FS was rendered using the original motor commands associated with the target sensation, while each cue applied on the hidden fingertip was rendered by feeding the target sensation into one of three rendering approaches and using the adjusted motor commands. We rendered each of the ten sensations twice using each of the three rendering approaches, for a total of 60 comparisons per subject. In addition to these, we also asked the subjects to compare two cues in which we rendered a very strong contact on the FS (all motors pulling the platform up) and no contact at all on the user’s fingertip (V1/H0), and two cues with no contact on the FS and a very strong contact at the fingertip (V0/H1). These four comparisons are useful to
understand how users judge sensations that are clearly different from each other. Eight females and five males with ages between 24 and 40 (mean of 31, standard deviation of 6.1) participated in this study, which was approved by the Penn IRB under protocol 834168. All subjects gave informed consent. One participant did the experiment twice due to a calibration error in the first run; we report only the second data set.

In Fig. 4.8b, we see that across all trials, the similarity rating was significantly affected by the rendering approach (Friedman statistical test: \( n = 260, Q(3) = 118.7, p < 0.0001 \)). The mean ratings for the standard, data-driven, and geometric methods were 3.7, 6.6, and 6.3, respectively. A post-hoc analysis further showed that ratings were significantly lower for the standard approach compared to the other two (Wilcoxon matched-pairs signed-ranked test: \( p < 0.0001 \)), while the data-driven and geometric approaches did not significantly differ. In Fig. 4.8a we see that the performance of the standard approach tends to degrade as the difference between the radius of the subject’s finger and FS (7.5 mm) increases. The standard approach performs worst for small fingertips, for which some trials resulted in a clear contact on the FS and no contact at all on the user’s fingertip.

4.6 Discussion

Both evaluations confirm that the performance of the standard rendering algorithm significantly degrades as the fingertip starts to differ from the sensor FS used for building the lookup table.

As expected, the standard technique of Sec. 4.1 renders contacts on the FS very well, with a median error of less than 0.2 N, confirming the results of [60]; this technique also shows good performance in rendering contacts on human fingertips that are sized similar to the FS. In these cases, rendering personalized cues provides no benefit over the standard method. However, as the difference between the target fingertip and the FS increases, the performance of the standard method degrades. The standard approach performed significantly worse for all the rubber fingertips other than the FS, with measured force errors as large as 5 N for the largest cast fingertip. Similarly, human subjects with fingers larger or smaller than the FS gave lower ratings. We note that the standard approach performs worse for smaller fingers than for larger fingers, since smaller fingers are often not contacted at all,
Figure 4.8: Human-subject study results. (a) Ratings for each rendering method as a function of measured finger radius. For each participant, the mean ratings for all three rendering methods are marked at the corresponding finger radius, with bars representing standard deviations. The vertical dashed line at 7.5 mm shows the FS radius. The two asterisks give the means and standard deviations of ratings for the conditions with strong visual contact and no haptic contact (V1/H0), and no visual contact with strong haptic contact (V0/H1) across all participants. (b) Boxplots for all ratings of the three rendering methods across all trials and finger sizes. Asterisks denote differences that are statistically different, with \( p < 0.0001 \).
causing several subjects to give all 20 standard trials a rating of one.

The results also show the overall viability and effectiveness of our personalization approaches. For our rubber fingertips other than the FS, both personalization approaches are able to significantly reduce the rendering error. For the human subjects, we see in Fig. 4.8b that both the data-driven and geometric personalization methods were rated significantly higher than the standard approach across trials. Interestingly, the two personalization approaches achieve similar reductions in force errors.

However, our results also suggest that there are areas of improvement, especially for our geometric personalization. For the large rubber human fingertip, the geometric personalization leads to significantly larger errors than the data-driven one. We also notice that the geometric personalization method tends to receive lower ratings from subjects with larger fingertips. These results suggest that the true geometry of these fingertips may not be well represented by our one-dimensional model. Moreover, a geometric model cannot capture biomechanical variations, such as the elasticity and anisotropy of the skin. For human fingertips that are larger than the FS, we expect the geometric approach to reduce fingertip deformation. However, if these fingers are also less stiff than the FS, then the smaller deformations may result in unrealistically small forces. Thus, updating our fingertip model to account for finger mechanics may improve our personalization approach. These mechanical characteristics are inherently taken into account by the data-driven personalization, as it focuses on the contact forces applied by the end-effector. As such, the data-driven approach receives consistently high ratings regardless of finger radius, with all participants giving it mean ratings greater than five.

4.7 Limitations and Future Considerations

Each of our two personalization methods has room for further improvements. Although our geometric method was limited to a 1D representation of finger size, the same process can be applied to arbitrarily complex fingertip models. We expect that rendering errors could be further reduced using a higher-dimensional geometric model of the fingertip, such as a mesh generated by a 3D-scanner. A model that includes biomechanical properties, such as fingerpad stiffness, could yield improved performance as well. Future work, possibly following the procedures provided in this
chapter, is needed to better understand how the type and complexity of fingertip model effects rendering error for a geometric personalization approach.

There are also opportunities for continued examination of the data-driven personalization approach. Of course, expanding to different fingertip devices and device-mounted sensors may be beneficial to specific applications. It may be particularly useful to explore using internal feedback from the servo motors themselves, such that no sensor needs to be mounted to the mobile platform during data collection. Furthermore, certain devices may require the DS to be reconfigured or the types of collected data to be redefined. The 3-DOF device in this chapter tilts the end-effector such that each motor command elicits a unique combination force and torque on the mobile platform. For a device that enables the center of the end-effector to apply pure normal force at different locations on the fingertip, such as the 6-DOF device presented in Chapter 2, the relationship between motor commands and DS values is not one-to-one; we would expect the DS to read the same value when the end-effector center renders 1N of normal force to different locations on the fingertip. If the motor command required to elicit a particular DS value is ambiguous, it may be possible to expand the data to include an estimate of force direction or location, using device kinematics, such that there is again a one-to-one relationship between motor command and force with respect to the mobile platform’s frame.

4.8 Summary

While wearable and fingertip haptic devices are gaining increasing popularity, little work has been dedicated to the important issue of adapting devices and algorithms to fingertips of different sizes and shapes. This chapter presented the first effort in devising personalized haptic rendering techniques for fingertip haptics. We presented two personalization techniques, the first of which is purely data driven. It requires an additional sensor on the platform during the data collection phase, and it requires each user to undergo a data collection procedure. However, it does not need any knowledge about the fingertip or device characteristics. The second technique aims at matching the deformation generated on the FS sensor during data collection, for which a geometric model of the fingertip is needed. However, it does not require any additional sensor, and it does not require the user to undergo a data
collection routine. Overall, our results suggest that both approaches can effectively personalize tactile interactions rendered by a haptic device to fingertips of different shapes and sizes. Although we employed a specific fingertip device and sensor to present and evaluate our personalization techniques, the same approaches can be applied to any other similar haptic system.
Chapter 5

Conclusion

Wearable haptic devices have seen growing interest in recent years and promise to remain essential in creating a technologically enhanced future. Although prior work had demonstrated highly wearable designs that are well suited to render particular interactions, to reach the full potential of haptic technology, designers must utilize humans’ complex and diverse sense of touch. We propose that providing more expressive and personalized tactile fingertip feedback will be core components of building haptic systems that convincingly simulate everyday physical interactions for a wide range of people.

This thesis began with the design and evaluation of a wearable fingertip device that is capable of rendering six-dimensional tactile feedback to the fingertip. More expressive cues can be rendered with the presented device than with existing wearable devices, including combinations of normal and shear force at locations encompassing the entire fingertip. We demonstrated that users donning the device were able to discern nine evenly distributed contact locations, as well as translational and rotational shear cues near the center of the fingertip.

This 6-DOF device was then integrated into a system that enabled a user to tactiley explore immersive virtual environments. We evaluated the effect of tactile dimensionality on user performance during shape discrimination and mass discrimination in a virtual environment, also comparing to interactions with real objects. Results showed that higher-dimensional tactile feedback may indeed allow completion of a wider range of virtual tasks, but that feedback dimensionality surprisingly does not greatly affect the exploratory techniques employed by the user.
To address alternative approaches to improving tactile rendering in scenarios where low-dimensional tactile feedback is appropriate, we then present two software-based approaches to personalize tactile cues for fingertips of different sizes. Results show that both personalization approaches significantly reduced force error magnitudes and improved realism ratings.

The designs and methods presented in this thesis demonstrate important steps toward providing natural, compelling haptic feedback to users regardless of physical characteristics. We summarize our contributions in Section 5.1 and possible avenues of continued work towards providing more expressive and personalized fingertip tactile cues in Section 5.2.

5.1 Summary of Contributions

The primary contributions of this thesis are as follows:

- The idea of using a parallel continuum manipulator (PCM) to create a fingertip haptic device and a process to refine the design for fingertip haptic applications. This process resulted in design parameters that were verified to enable a manually actuated prototype to render a desired set of fingertip tactile cues.

- A motorized 6-DOF wearable fingertip tactile device, named the Fingertip Puppeteer. This ungrounded device is the first that is capable of controlling the position and orientation of a flat platform, such that any combination of normal and shear force can be delivered at any location on any human fingertip. We present design, control, and error reduction techniques, along with validation of contact location and shear rendering by human subjects.

- A system that allows a participant to explore virtual environments while feeling fingertip tactile feedback of variable dimensions, along with a framework for rendering 1-DOF, 3-DOF and 6-DOF fingertip tactile cues during virtual interactions. This system allows analysis of the effect of tactile fingertip feedback dimensionality on perceptual performance and exploratory motions.

- Demonstration that higher-dimensional tactile feedback may indeed allow completion of a wider range of virtual tasks, but that feedback dimension-
ality surprisingly does not greatly affect the exploratory techniques employed by the user.

- Two software-based approaches to personalizing a haptic rendering algorithm for fingertips of different sizes using either additional data or geometry, resulting in reduced force-rendering errors and improved user-perceived realism.

5.2 Conclusions and Future Directions

This thesis discusses a hardware-based approach to rendering more expressive fingertip tactile cues and a system through which the affect of tactile feedback dimensionality on user performance can be observed. A software-based approach to rendering personalized fingertip tactile cues is also presented, allowing fingertip sensations to be more realistically rendered to a variety of users. More research will be needed to improve the rendering of higher-dimensional tactile cues and enable a broader range of haptic devices to accommodate variation between users.

One possible branch of continued work relates to the design of the Fingertip Puppeteer. We expect that future mechanical designs could reduce mass, primarily through the use of more compact motors, or improve comfort, through a more ergonomic handle design. Alternative designs of the roller mechanism may also reduce the frequency at which the device needs to be calibrated, or eliminate wire slip altogether, which would allow the device to render longer, more immersive interactions. Finally, although we demonstrated that the position controlled Puppeteer could render recognizable contact locations and shear directions reasonably well in Chapter 2, it is unable to apply a specified amount of force. Future work includes adapting the device to apply a specified force, rather than displacement, by either considering the mechanical properties of the fingertip and modeling the end-effector stiffness as a function of configuration, or implementing closed-loop feedback. Closed-loop feedback of contact, force, position, and/or orientation may also lead to improved rendering of displacements.

Continued research could also expand the virtual reality vari-dimensional tactile feedback system to include a larger set of rendering methods, such as providing shear skin stretch at a single location on the fingertip, and observing performance in additional tasks. Furthermore, study instructions in Chapter 3 provided no in-
formation as to how the rendering methods differed, or how each rendering method could be used to complete the tasks. This approach allowed us to observe naive performance, but it also proved challenging for some participants who never discovered exploration techniques that yielded positive results. If the tactile rendering methods were explained, or if suggested exploratory techniques were provided, users would most likely be better able to discern which techniques work best for a particular task and rendering method, rather than consistently trying the same technique.

With respect to the software-based approaches to rendering personalized fingertip tactile cues presented in Chapter 4, a clear continuation would be to apply the method to an assortment of sensors and fingertip devices. Additionally, our data-driven and geometric personalization approaches have many parameters that can be further explored. For the geometric personalization, performance may be improved by generating higher-dimensional geometric models of the user’s fingertip, such as generating a point cloud from a 3D scan, or using a mechanical model that details the relationship between force and deformation. For the data-driven personalization, future work could examine the effect of weighting functions, number of nearest neighbors, and lookup table size on the rendering error. It would be interesting to extend such personalization approaches to also consider differences in the relative positioning of the fingertip with respect to the device end-effector, or to even consider differences in perception between users.

There may also be benefits in combining elements from each personalization method. For example, coupling a contact or force sensor on the mobile platform with knowledge of the device kinematics may allow a geometric model of the fingertip to be generated and extracted for external use. This model could be used to personalize a virtual representation of a fingertip, or even motivate a personalized mechanical design of a wearable device.

Finally, a more direct extension of this thesis would be to combine our methods for rendering expressive and personalized fingertip tactile cues. The Fingertip Puppeteer could first be used to render the remotely sensed cues of Chapter 4 with six degrees of freedom. We expect that such an extension would not be trivial, since rendering shear cues is a path-dependent process that requires an end-effector trajectory rather than a single end-effector pose. Whereas in Chapter 4 we mapped each remote sensor value to a single vector of motor values, to accommodate shear cues we would need to map each remote sensor value to at least two vectors of motor
values, associated with initial contact and then shear displacement. Once rendering remotely-sensed 6-DOF cues is established, then the software-based approaches of Chapter 4 could be used to personalize these cues, to address the relationship between finger size and accuracy discussed in Chapter 2.
Bibliography


[27] Sheik N. Imrhan, M. D. Sarder, and Nabeel Mandahawi, *Hand anthropometry in Bangladeshis living in America and comparisons with other populations*, Ergonomics **52** (2009), no. 8, 987–998.


[74] Massimiliano Solazzi, Antonio Frisoli, and Massimo Bergamasco, *Design of a cutaneous fingertip display for improving haptic exploration of virtual objects,*


[82] Dzmitry Tsetserukou, Shotaro Hosokawa, and Kazuhiko Terashima, Link-Touch: A wearable haptic device with five-bar linkage mechanism for presenta-


