

# **Redirected Touching**

Luv Kohli

A dissertation submitted to the faculty of the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Computer Science.

Chapel Hill  
2013

Approved by:

Frederick P. Brooks Jr.

Mary C. Whitton

Robert W. Lindeman

Russell M. Taylor II

Mark Hollins

©2013  
Luv Kohli  
ALL RIGHTS RESERVED

## ABSTRACT

LUV KOHLI: Redirected Touching  
(Under the direction of Frederick P. Brooks, Jr.)

In immersive virtual environments, virtual objects cannot be touched. One solution is to use *passive haptics*—physical props to which virtual objects are registered. The result is compelling; when a user reaches out with a virtual hand to touch a virtual object, her real hand touches and feels a real object. However, for every virtual object to be touched, there must be an analogous physical prop. In the limit, an entire real-world infrastructure would need to be built and changed whenever a virtual scene is changed.

Virtual objects and passive haptics have historically been mapped one-to-one. I demonstrate that the mapping need not be one-to-one. One can make a single passive real object provide useful haptic feedback for many virtual objects by exploiting human perception. I developed and investigated three categories of such techniques:

1. *Move the virtual world* to align different virtual objects in turn with the same real object
2. *Move a virtual object* into alignment with a real object
3. *Map real hand motion to different virtual hand motion*, e.g., when the real hand traces a real object, the virtual hand traces a differently shaped virtual object.

The first two techniques were investigated for feasibility, and the third was explored more deeply. The first technique (Redirected Passive Haptics) enables users to touch multiple instances of a virtual object, with haptic feedback provided by a single real object. The second technique (The Haptic Hand) attaches a larger-than-hand virtual user interface to the non-dominant hand, mapping the currently relevant part of the interface onto the palm.

The third technique (Redirected Touching) warps virtual space to map many differently shaped virtual objects onto a single real object, introducing a discrepancy between real and virtual hand

motions. Two studies investigated the technique's effect on task performance and its potential for use in aircraft cockpit procedures training. Users adapt rather quickly to real-virtual discrepancy, and after adaptation, users perform no worse with discrepant virtual objects than with one-to-one virtual objects. Redirected Touching shows promise for training and entertainment applications.

To my parents, Om and Pushp, and to my brother, Manu.

## ACKNOWLEDGMENTS

While working on this dissertation, I have been helped by many people. I am grateful for their support. Though I cannot possibly express the depth of my appreciation in the few short sentences below, I would like to acknowledge those that have helped me complete this work.

I first thank my doctoral committee:

- Fred Brooks, for his guidance, patience, encouragement, and understanding. I could not ask for a better advisor. He always found time to meet with me whenever I needed his advice (even late at night), and he pushed me when I most needed it. His advice on research, writing, and life has forever shaped the way I think.
- Mary Whitton, for entrusting me with implementing her idea for *The Haptic Hand*; working on it gave me a new appreciation for virtual environments research and started me on the path to completing the rest of this work. Also, she always found time to help me when deadlines were fast approaching.
- Rob Lindeman, for his friendship throughout the years and for introducing me to virtual reality (it's all his fault).
- Russ Taylor, for helping me with the various technical problems I encountered during my work in the lab, enabling me to benefit from his expertise.
- Mark Hollins, for guiding my research in ways that made it more efficient, helping me avoid falling into research rabbit holes.

I am grateful for the support and funding that I have received from different agencies and programs during my time at UNC: the NIH National Institute of Biomedical Imaging and Bioengineering, the NIH National Center for Research Resources, the Office of Naval Research, the Link Foundation, the North Carolina Space Grant, and Electronic Arts. I thank CyberGlove Systems for letting me use their virtual hand model (©2013 CyberGlove Systems).

The faculty and staff in the Department of Computer Science are outstanding. I would particularly like to thank:

- John Thomas, for building most of the physical infrastructure required for my research. Every time I went to him with a research need, he had an idea for how to address it quickly and efficiently, even when my requirements changed after he already built something for me. I would not have been able to do any of my experiments without John.
- David Harrison and Herman Towles, for letting me borrow so much of their equipment and bailing me out whenever I had last-minute equipment failures.
- Henry Fuchs, Gary Bishop, Anselmo Lastra, Steve Weiss, and Tim Quigg, for giving me so much advice about research, graduate school, and more over the years.
- Missy Wood and Kelli Gaskill, for being so kind and caring, and for always helping me find a solution to whatever strange problem I had in the lab.

I am also indebted to the many friends and colleagues who have spent their time meeting with me, giving me advice, teaching me, encouraging me, and otherwise supporting me:

- Anatole Lécuyer, for discussing Redirected Touching with me in its very early stages, and for encouraging me to move forward with it.
- Rohit Saboo, for giving me the space-warping code that provided the foundation for my code. Without it, I would not have been able to investigate Redirected Touching.
- Anish Chandak, for helping me with FMOD during my first Redirected Touching study.
- Kan Anant and Tracy McSheery at PhaseSpace, Inc. for their excellent support whenever I had difficulties getting tracking to work.
- Rob Teather, for helping me understand ISO 9241-9.
- Chris Wiesen, for his pragmatic explanations of statistics.
- Lt. Col. Paul Holst, USAF, for his interest in my research, and for his help in getting the Link Foundation Fellowship in Advanced Simulation and Training.

- All of the previous members of Effective Virtual Environments (EVE) whose research created giant shoulders upon which my work could rest.
- Chris VanderKnyff and Chris Oates, the other two members of the EVEIL trio. My work is built upon their technical expertise.
- Paul Zimmons, for his continued encouragement, for his sense of humor, and for drawing Chicken Luv, Happy Luv, Messed-up Luv, Angry Luv, and Frightened Luv, some of which became unofficial EVE mascots.
- Eric Burns and Sharif Razzaque, for their friendship, and for always being willing to help me figure out my problems even when they had very busy schedules.
- Jeremy Wendt, for helping me get through grad school by being a model of perseverance.
- Alexis Chan, for believing in me.
- Tabitha Peck, for helping me get through my first time teaching a course.
- Jason Jerald, Jeff Feasel, Jess Martin, Rick Skarbez, Ky Waegel, and Zhen Shao, for helping me work through my research ideas and giving me valuable feedback.
- Sachin Patil, for always being willing to be a guinea pig for my research.
- Ariele Sieling, for encouraging me and for proofreading my dissertation.
- Elise Carmichael, for helping me recruit participants, and for helping me find Waldo.
- Peter Lincoln, for giving me his pan-tilt unit code for my first Redirected Touching study. In addition to Peter, Ryan Schubert, Miranda Steed, Sean and Marie Curtis, Chris and Anna Berg, and everyone else in the barn-cs group helped bring sanity to long weeks of stressful work.
- Matt Nealon, for drawing art for my dissertation.
- Tammy Shenkman, for being so supportive, particularly during the weeks leading up to my defense when I had hardly any time for anything but work.
- Evan Suma, for being a sounding board whenever I had gripes about research.

- Mark Bolas and Haruo Noma, for giving me the opportunity to work in their respective labs.
- Matt Brown and Tony Marinello, for being outstanding mentors during my internships at Electronic Arts.

Finally, and most importantly, I thank my parents, Om and Pushp Kohli, and my brother, Manu. Their love and kindness have guided me throughout my life. I cannot adequately express how grateful I am for their support; without them, I would be lost.

# TABLE OF CONTENTS

<b>LIST OF TABLES</b> .....	<b>xv</b>
<b>LIST OF FIGURES</b> .....	<b>xvi</b>
<b>1 What and Why</b> .....	<b>1</b>
1.1 Thesis statement .....	4
1.2 Technique 1: <i>Redirected Passive Haptics</i> .....	4
1.2.1 Summary of results .....	4
1.3 Technique 2: <i>The Haptic Hand</i> .....	5
1.3.1 Summary of results .....	6
1.4 Technique 3: <i>Redirected Touching</i> .....	6
1.4.1 Summary of results .....	7
<b>2 Redirected Passive Haptics</b> .....	<b>9</b>
2.1 Moving the virtual world .....	10
2.2 Proof of concept .....	13
2.3 Informal evaluation.....	15
<b>3 The Haptic Hand</b> .....	<b>17</b>
3.1 Bimanual interaction and indirect manipulation .....	18
3.1.1 Exploiting proprioception .....	19
3.2 The idea: <i>The Haptic Hand</i> .....	20
3.2.1 The interface panel .....	20
3.2.2 Interacting with the virtual environment using hand gestures .....	20
3.2.3 Widget activation .....	21

3.2.4	Widget manipulation .....	23
3.2.5	Calibration .....	23
3.2.6	Manipulation issues .....	24
3.3	Usability study .....	24
3.3.1	Materials and methods .....	25
3.3.2	Study design .....	25
3.4	Results .....	27
3.5	Future work.....	30
3.6	Insights from mapping a virtual object onto the non-dominant hand .....	31
<b>4</b>	<b>Redirected Touching.....</b>	<b>33</b>
4.1	Warping virtual space .....	34
4.2	Preliminary exploration.....	38
4.3	Implementation changes .....	40
4.3.1	Tracking .....	40
4.3.2	Automating space warping: thin-plate splines .....	41
4.4	Further exploration .....	42
4.5	Quickly set-up passive haptics .....	44
4.5.1	Finger tracking and calibration .....	44
4.5.2	Determination of physical geometry .....	45
4.5.3	Discrepant objects .....	46
<b>5</b>	<b>Redirected Touching: Task Performance .....</b>	<b>47</b>
5.1	Research questions .....	48
5.2	System .....	49
5.3	Warping space .....	50
5.3.1	Choosing correspondences for space-warping .....	50
5.3.2	Hand orientation.....	51
5.4	Measuring task performance .....	51

5.5	Non-inferiority testing .....	53
5.5.1	Indifference zone: throughput .....	53
5.5.2	Indifference zone: error rate .....	54
5.5.3	Movement time .....	54
5.6	Study .....	54
5.6.1	Participants .....	54
5.6.2	Notation .....	54
5.6.3	Angle range .....	55
5.6.4	Adaptation .....	55
5.6.5	Target distances and widths .....	55
5.6.6	Trials and counterbalancing .....	56
5.6.7	Procedure .....	56
5.7	Results .....	58
5.7.1	Spatial outliers .....	58
5.7.2	Learning effect and analysis .....	58
5.7.3	Throughput .....	59
5.7.4	Error rate .....	60
5.7.5	Movement time .....	61
5.8	Discussion .....	63
5.8.1	Caveats .....	63
5.8.2	Computing throughput with 2D endpoint deviation .....	64
5.8.3	Participant comments and other observations .....	65
5.9	Revisiting adaptation .....	66
5.10	Exploratory detectability data .....	75
5.11	Summary .....	76
<b>6</b>	<b>Redirected Touching: Training and Adaptation</b> .....	<b>77</b>
6.1	Study .....	78

6.1.1	Non-inferiority indifference zones .....	80
6.1.2	Apparatus .....	80
6.1.3	Participants .....	81
6.1.4	Study procedure .....	81
6.2	Results: task performance .....	82
6.2.1	Posttest vs. Pretest .....	84
6.2.2	Non-inferior task performance .....	86
6.3	Measuring adaptation by path variability .....	88
6.4	Results: adaptation .....	91
6.5	Discussion .....	92
6.5.1	Participant comments: discrepancy detected? .....	93
6.5.2	Other participant comments .....	94
6.6	Summary and conclusions .....	94
<b>7</b>	<b>Conclusions and Future Work .....</b>	<b>96</b>
7.1	Revisiting the thesis statement .....	96
7.2	Future work .....	97
7.2.1	Training .....	97
7.2.2	Discrepancy detectability .....	97
7.2.3	Space-warping .....	98
7.2.3.1	Automatic geometry determination and correspondence generation	98
7.2.3.2	Multiple objects .....	98
7.2.3.3	Moving rigid objects .....	99
7.2.3.4	Whole-hand interaction .....	99
7.2.3.5	Adding active haptics .....	99
7.3	Implementing Redirected Touching with off-the-shelf hardware .....	100
<b>A</b>	<b>Questionnaires .....</b>	<b>101</b>
A.1	The Haptic Hand .....	101

A.2 Redirected Touching: Task Performance .....	108
A.3 Redirected Touching: Training and Adaptation .....	110
<b>B Target Ring Details .....</b>	<b>112</b>
<b>BIBLIOGRAPHY .....</b>	<b>113</b>

## LIST OF TABLES

5.1	Differences between discrepant and one-to-one throughputs (bps).....	60
5.2	Differences between discrepant and one-to-one error rates .....	61
5.3	Differences between discrepant and one-to-one movement times (ms).....	62
5.4	Differences between discrepant and one-to-one throughputs (bps) computed using 2D endpoint deviation .....	65
5.5	<i>t</i> -test results for path variability differences between first and second exposures to the largest-target-width ring in each condition .....	73
6.1	ANOVA results for mean throughput differences .....	85
6.2	ANOVA results for mean error rate differences .....	85
6.3	ANOVA results for mean movement time differences .....	85

## LIST OF FIGURES

1.1	A virtual hand penetrates a virtual object because there is no real object to touch. . . . .	1
1.2	Passive haptics (images from Insko [2001]) . . . . .	2
1.3	Mapping many different virtual objects onto one physical object . . . . .	3
1.4	A VE with five differently colored virtual pedestals. The real pedestal and tracked space are superimposed. . . . .	5
1.5	A user touches the one cylindrical object intended to provide haptic feedback. The foam walls mark the tracked space boundaries. . . . .	5
1.6	A user interacts with a virtual interface panel by touching his non-dominant hand. . . . .	5
1.7	A user touches a virtual board that is oriented differently than the real board providing passive-haptic feedback. . . . .	7
2.1	Five virtual easels (upper left) and five real easels (upper right) that provide haptic feedback, enabling a user to draw effectively on the virtual easels. Mr. Unsavory Character seeks to disrupt the task and steals four of the real easels (lower left). Can the task still be done with only one real easel (lower right)? (Artwork by Matt Nealon) . . . . .	9
2.2	The virtual world is moved so that each virtual easel is in turn aligned with the real easel. (Artwork by Matt Nealon) . . . . .	10
2.3	When the virtual world is rotated, real and virtual objects become misaligned. . . . .	11
2.4	Rotating the virtual world around the user's head . . . . .	12
2.5	Difficulties with object alignment . . . . .	12
2.6	A VE with five differently colored virtual pedestals. The real pedestal and tracked space are superimposed. . . . .	13
2.7	A user touches the one cylindrical object intended to provide haptic feedback. The foam walls mark the tracked space boundaries. . . . .	13
2.8	Step-by-step method to align virtual objects with real object: (t1) User guided towards rotation point, equidistant from real and virtual objects; (t2) user reaches rotation point; (t3) user turns head, virtual world rotates imperceptibly; (t4) real and virtual objects aligned . . . . .	14
2.9	A transporter beam (left) and droid (right) . . . . .	15
2.10	One user's real (left) and virtual (right) paths . . . . .	16

3.1	A virtual easel and a real easel that provides haptic feedback (upper left). Mr. Unsavory Character shoots the real easel with a shrink ray (upper right, lower left), yielding a real easel that is smaller than the virtual easel (lower right). Can a virtual drawing task still be done using the small easel for haptic feedback? (Artwork by Matt Nealon) .....	17
3.2	A user interacts with a virtual interface panel by touching his non-dominant hand. ...	19
3.3	A sample interface panel with buttons and sliders .....	21
3.4	Interface panel manipulation: (a) the panel is not attached to the NDH; (b) the NDH is closed and intersects the panel to highlight the desired widget; (c) the NDH is opened to snap the desired widget to the center of the palm; (d) the DH is used to manipulate the active widget; (e) the NDH is closed and moved to a new desired widget; (f) the NDH is opened and the new widget is snapped and activated. ....	22
3.5	The NDH (left) and DH (right) with trackers attached. The white circles indicate where the NDH trackers are located underneath the sports bandage: one on the fingertip to monitor whether the palm is open or closed, and one on the back of the hand to determine its pose. ....	23
3.6	The interface panel and environment for the matching task .....	26
3.7	User-reported ratings of ease of interaction on a scale from 1 (extremely easy) to 7 (extremely difficult) .....	28
4.1	A virtual easel and a real easel that provides haptic feedback (upper left). Mr. Unsavory Character sees he must try something new (upper right), so he angrily bends the real easel (lower left), yielding a real easel that no longer matches the shape of the virtual easel (lower right). Can a virtual drawing task still be done despite the real-virtual shape discrepancy? (Artwork by Matt Nealon) .....	33
4.2	A flat virtual easel and a curved real easel, viewed from above. When the real hand moves 10cm to touch the real easel, the virtual hand must move 15cm to touch the virtual easel. ....	36
4.3	Horizontal slices of unwarped (a) and warped (b) spaces and objects (viewed from above) .....	37
4.4	A user touches a flat square table .....	38
4.5	Differently shaped virtual objects for which a square-top real table (Figure 4.4) serves as haptic feedback .....	39
4.6	Horizontal slices of unwarped space and real square table (left) and warped space and tapered virtual table (Figure 4.5d), viewed from above (not drawn to scale). Note that the warped space is not smooth as in Figure 4.3b. ....	40

4.7	PhaseSpace® IMPULSE™ tracking cameras mounted on an aluminum frame. Inset shows tracking LEDs mounted onto an index finger using a self-adhesive athletic bandage. ....	41
4.8	Correspondences between real (left) and virtual (right) geometries .....	42
4.9	Eight different deformed virtual objects mapped onto the square real table (Figure 4.4)	43
4.10	A low-cost two-faced passive-haptic foam board .....	45
4.11	Finger calibration .....	45
4.12	A user (wearing an NVIS nVisor SX60 HMD) touches a corner of the physical foam board. The virtual finger position is warped to touch the corners of three different virtual objects. The insets show the distances between the virtual finger and the superimposed real finger (red). ....	46
5.1	Lockheed Martin® C-130 Hercules® cockpit .....	47
5.2	System constructed for space-warping studies. Inset shows the finger tracking ring. ....	49
5.3	The multi-directional tapping task. Overlaid arrows show the target sequence. ....	52
5.4	Left: front view of real board at 18°; Middle: virtual board at 18°; Right: real board at 0°. Left and middle panels make up a one-to-one condition, and right and middle panels a discrepant condition .....	55
5.5	A user touches a virtual board that is oriented differently than the real board .....	57
5.6	A learning effect is seen in the throughput data, most evident between the first three and last eight blocks of target rings .....	59
5.7	Throughput by virtual angle. Error bars are 95% confidence intervals. ....	60
5.8	Error rate by virtual angle. Error bars are 95% confidence intervals.....	61
5.9	Movement time by virtual angle. Error bars are 95% confidence intervals.....	62
5.10	Throughput computed using 2D endpoint deviation. Error bars are 95% confidence intervals.....	65
5.11	One participant’s motion paths for a target ring in the (18, 18) condition .....	67
5.12	One participant’s motion paths for a target ring in the (0, 18) condition (first exposure to largest target width) .....	68
5.13	One participant’s motion paths for a target ring in the (0, 18) condition (second exposure to largest target width) .....	70

5.14	Path variability by path sample number (page 69) for each participant's first and second exposures to the largest-target-width rings in condition (18, 18). The numbers in each plot are the participant number and the total path variability for the first and second exposures. ....	71
5.15	Path variability by path sample number (page 69) for each participant's first and second exposures to the largest-target-width rings in condition (0, 18). The numbers in each plot are the participant number and the total path variability for the first and second exposures. ....	72
5.16	Path variability across all participants for one-to-one and discrepant conditions, first and second exposures to largest-target-width rings. Error bars are 95% confidence intervals. ....	74
5.17	Throughput across all participants for one-to-one and discrepant conditions, first and second exposures to largest-target-width rings. Error bars are 95% confidence intervals. ....	74
5.18	Exploratory discrepancy detectability data ....	75
6.1	A user sees an 18° monitor in an HMD (upper left) while touching a 0° touchscreen monitor (upper right); Lower left: touchscreen oriented at 18°; Lower right: touchscreen at 0° ....	79
6.2	Mean throughput across all participants for $R$ , $V_{1-1}$ , and $V_d$ conditions. Error bars are 95% confidence intervals. ....	83
6.3	Mean error rate across all participants for $R$ , $V_{1-1}$ , and $V_d$ conditions. Error bars are 95% confidence intervals. ....	83
6.4	Mean movement time across all participants for $R$ , $V_{1-1}$ , and $V_d$ conditions. Error bars are 95% confidence intervals. ....	84
6.5	Mean throughput differences between pairs of conditions. Error bars are one-tailed 95% confidence intervals. Shaded area represents non-inferiority indifference zone. ....	86
6.6	Mean error rate differences between pairs of conditions. Error bars are one-tailed 95% confidence intervals. Shaded area represents non-inferiority indifference zone. ....	87
6.7	Mean movement time differences between pairs of conditions. Error bars are one-tailed 95% confidence intervals. Shaded area represents non-inferiority indifference zone. ....	87
6.8	A $V_d$ participant's motion paths, first Training target ring. ....	89
6.9	A $V_d$ Training target ring just after the first break. ....	89
6.10	A $V_d$ participant's motion paths ....	90

6.11	Mean path variability across all participants for $R$ , $V_{1-1}$ , and $V_d$ conditions. Error bars are 95% confidence intervals. ....	91
6.12	Mean path variability differences between pairs of conditions. Error bars are one-tailed 95% confidence intervals. Shaded area represents non-inferiority indifference zone. ....	92
7.1	Upper left: a user's hand is tracked by the Leap Motion™ Controller while he touches the corner of a physical object onto which virtual objects will map; Upper right: horizontal slice of warped space, viewed from above (real object: black; virtual: blue); Bottom half: two differently shaped virtual objects mapped onto the same real object .....	100

## CHAPTER 1

# What and Why

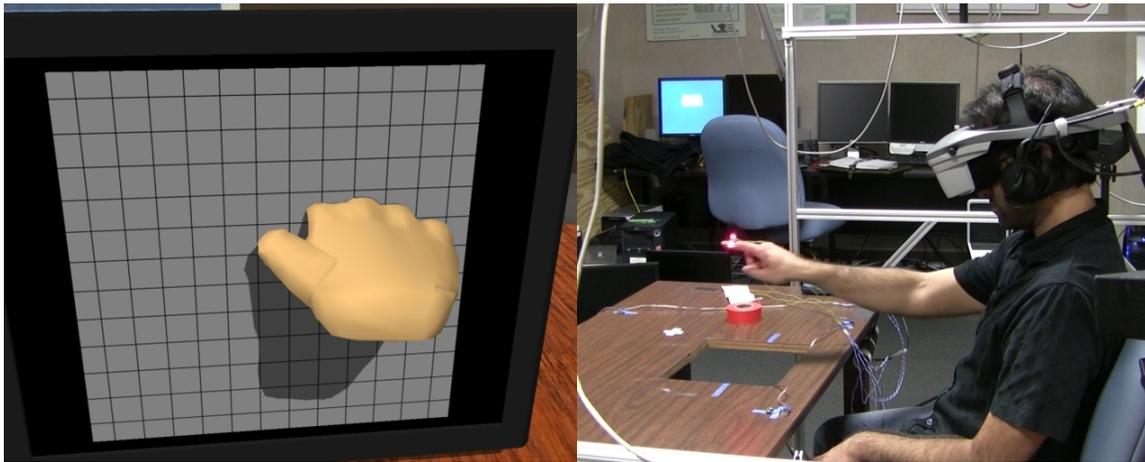


Figure 1.1: A virtual hand penetrates a virtual object because there is no real object to touch.

Imagine a world in which the only physical constraints on your movement are your own body and the ground on which you stand. In some ways, this world might be desirable; you could walk from room to room very efficiently, not needing to plan your path around obstacles such as walls. In other ways, life would be very difficult; you would not be able to pick up a glass to take a sip, sit down on a chair, or lie down on a bed. Your body would go right through these objects. Many virtual environments (VEs) behave much like this imagined world: you can move through a beautifully rendered environment, but you cannot touch anything. Experiences such as that in Figure 1.1 destroy the illusion upon which the usefulness of VEs depends.

There are two broad classes of haptic (touch) feedback solutions in VEs: *active-haptic* feedback (or *active haptics*), and *passive-haptic* feedback (or *passive haptics*). Active haptics are computer-controlled devices that supply tactile or force feedback to the user through electric, electronic, or mechanical means [Lindeman, 1999]. Examples include force feedback devices such as the Argonne

Remote Manipulator, Sensable Technologies, Inc.® PHANTOM®, and Novint Falcon®; vibrotactile pager motors; and pin arrays that can be used to approximate different shapes. Active haptics can be programmed to provide compelling haptic feedback for a variety of virtual objects. However, they are often expensive and have a limited working volume. Low-cost active-haptic devices, such as vibrotactile pager motors, can be used wirelessly to provide a large working volume, but the resulting haptic feedback is not precise enough for many applications.

I focus on passive haptics—physical props to which virtual objects are registered (Figure 1.2). Traditional passive haptics use a one-to-one mapping between real and virtual objects. In a head-mounted display (HMD), users see a VE filled with virtual objects. When a user touches a virtual object with her virtual hand, her real hand touches the corresponding real object. This technique is compelling because the user experiences a real sense of having touched something. However, to enable a user to touch everything in the VE, there must be a physical mock-up of every virtual object. In the limit, an entire real-world infrastructure would need to be built.

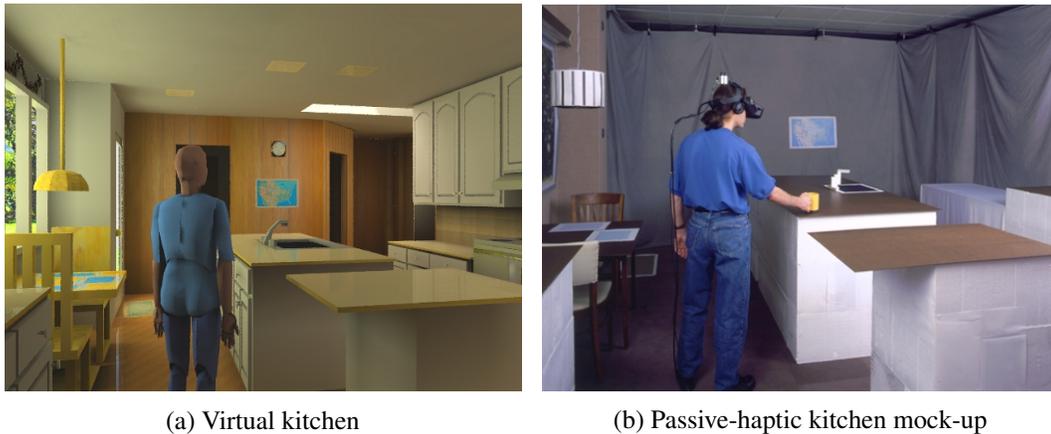


Figure 1.2: Passive haptics (images from Insko [2001])

Insko found that compared to no haptics, using even low-fidelity passive haptics (such as real Styrofoam walls in place of virtual brick walls) led to significantly increased presence (loosely, the sense of being in an environment) and spatial knowledge training transfer [Insko, 2001]. Using low-fidelity mock-ups makes an environment designer’s job easier, but passive haptics impose constraints on changes to the environment: if there is a need to change the position, size, or shape of a virtual object, its corresponding physical object must also change accordingly. Depending on the complexity of the change, it can be time-consuming to update passive haptics to match the changing VE.

Now imagine that one physical object could effectively represent many different virtual objects (Figure 1.3). Fewer mock-ups would be needed to provide a rich haptic experience, and existing mock-ups would be reusable rather than obsolete. If the mapping between virtual and real objects could be changed quickly by modifying the VE, then less time need be spent on updating physical mock-ups.

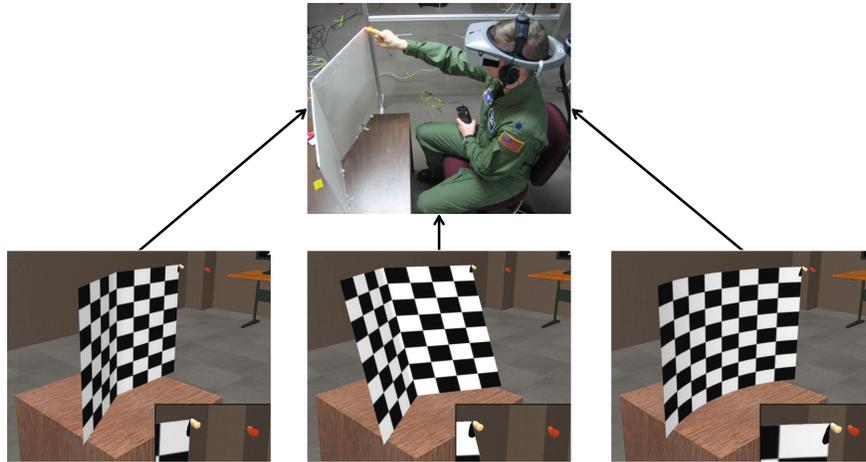


Figure 1.3: Mapping many different virtual objects onto one physical object

The goal of this dissertation is to demonstrate that the mapping between passive haptics and virtual objects need not be one-to-one, and that one passive real object can be made to provide useful haptic feedback for many virtual objects by exploiting the peculiarities of human perception. This dissertation develops and investigates three techniques that change the mapping between real and virtual objects. The first two techniques are investigated for feasibility, and the third is explored more deeply to evaluate its effect on task performance and its potential use in training applications. Ideally, these techniques will enable users to physically interact with many virtual objects without detracting from their experience. The techniques fall into three categories:

1. *Move the virtual world* to align different virtual objects with the same real object
2. *Move a virtual object* into alignment with a real object
3. *Map real hand motion to different virtual hand motion*, e.g., when the real hand traces a real object's surface, the virtual hand traces a differently shaped virtual object's surface.

## 1.1 Thesis statement

**Part I:** *In a virtual environment, computer-controlled variable mappings between real and virtual objects can enable a single passive real object to provide haptic feedback for different instances and kinds of virtual objects.*

**Part II:** *By warping virtual space, one can incorporate a non-zero amount of real-virtual shape discrepancy without impacting task performance.*

I will demonstrate Part I using three techniques I developed: *Redirected Passive Haptics*, *The Haptic Hand*, and *Redirected Touching*. I will demonstrate Part II using *Redirected Touching*.

## 1.2 Technique 1: *Redirected Passive Haptics*

The first technique (Chapter 2) supports the general idea of mapping more than one virtual object onto the same physical object. It demonstrates that, for well-planned scenarios, a VE with spatially separated instances of a virtual object can be remapped onto the real world such that only one passive real object provides effective haptic feedback for several virtual objects. This technique uses *Redirected Walking* [Razzaque, 2005] to map different parts of the VE onto the same real space.

### 1.2.1 Summary of results

This technique was developed as a course project. In an informal demonstration, eight users walked around a VE (8.26m x 13.12m) in a much smaller tracked space (4.13m x 6.56m) and touched five virtual pedestals (Figure 1.4), which were actually the same object in the real world (Figure 1.5). None of the users reported detecting that there was only one physical object, or that the world was rotating around them at a different rate from their head rotation. Even those users who had previously heard an explanation of how the system worked expressed surprise that they were able to touch all five virtual pedestals without detecting the manipulation.

In its original form, this technique does not enable unconstrained free exploration with haptics. The scenario was contrived, but it was developed to demonstrate that it is possible to map more than one virtual object onto a single physical object. Generalized mapping of virtual objects onto physical objects in large-scale VEs has since been further investigated by others [Steinicke et al., 2008b].

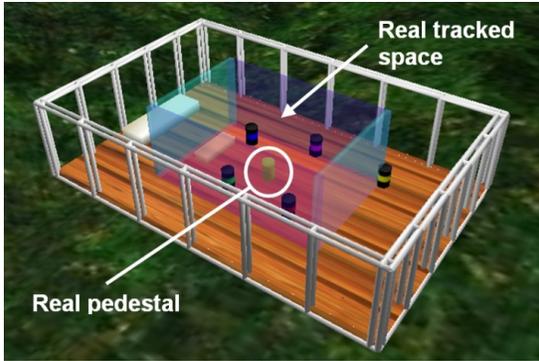


Figure 1.4: A VE with five differently colored virtual pedestals. The real pedestal and tracked space are superimposed.



Figure 1.5: A user touches the one cylindrical object intended to provide haptic feedback. The foam walls mark the tracked space boundaries.

### 1.3 Technique 2: *The Haptic Hand*

Even when there are no physical objects to provide haptic feedback in a VE, one haptic device is always available: the user's body. The second technique (Chapter 3) enables interaction with many virtual widgets (buttons, sliders) on a plane fitted to the non-dominant palm; users receive haptic feedback from a virtual interface panel by touching their own hands (Figure 1.6). Of particular relevance here is the result that users can interact effectively with a virtual object shaped differently from its physical counterpart.

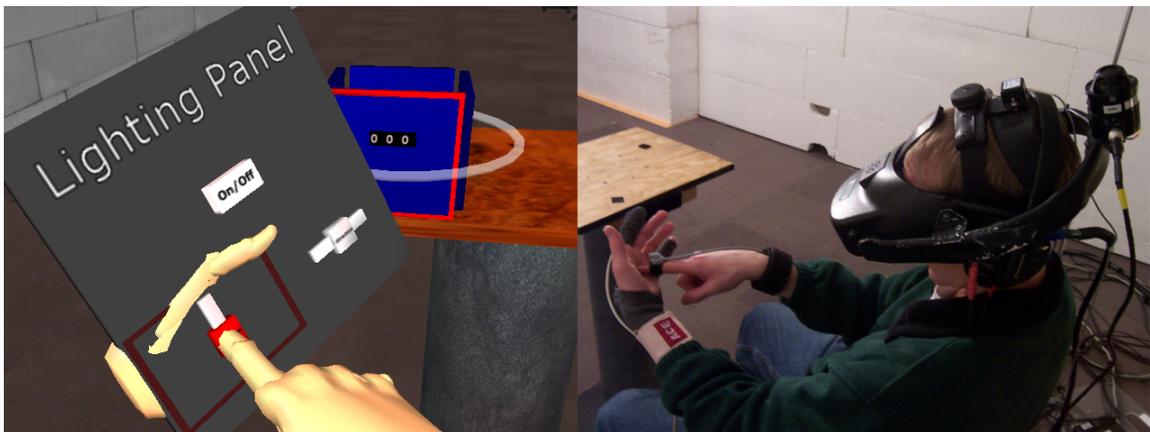


Figure 1.6: A user interacts with a virtual interface panel by touching his non-dominant hand.

### 1.3.1 Summary of results

For this technique, I created a system to present a user interface on a virtual plane fitted to the user's non-dominant hand (NDH). The user manipulates widgets (buttons and sliders) with the dominant hand. Because the virtual interface was larger than the palm, gestures to switch between active widgets were needed. These gestures enabled the NDH to provide haptic feedback for several different widgets, despite their being at different locations on the virtual interface panel. A feasibility study was run to determine the technique's promise for future development. Eight users were asked to rate ease of interaction via several questions arranged on Likert scales from 1 (extremely easy) to 7 (extremely difficult). Users were also asked to provide comments about their experience with the system.

All but two participants rated widget activation as easy or extremely easy. The remaining two rated it neutral. Universally, participants found buttons to be easier to use than sliders. Half of the participants rated sliders to be moderately easy, easy, or extremely easy. Two participants rated sliders neutral, and two rated them moderately difficult. Participant data indicate that the technique is viable and warrants further study, but care must be taken to improve performance with sliders. However, the technique was not further developed for this dissertation.

## 1.4 Technique 3: *Redirected Touching*

Realistic and technology-rich simulator training is often inaccessible to deployed armed forces. Deployable virtual training systems can help maintain combat readiness in the field [Vincenzi et al., 2008]. Haptic feedback enables users to learn about their environments through touch, but it is typically difficult to deploy. The third technique (Chapters 4–6) warps virtual space to enable low-cost, quickly set-up passive haptics to provide coarse haptic feedback that can be repurposed for different training scenarios (e.g., different aircraft with *differently shaped* and instrumented cockpits). When a user moves her real hand along some path (e.g., on the surface of a real object) in the real world, this technique moves her virtual hand along a different path (e.g., on the surface of a virtual object). Because vision usually dominates proprioception, users can be made to believe that the virtual hand is touching the *seen* shape, rather than the *felt* shape.

This technique, *Redirected Touching*, is not intended as a replacement for full flight simulators, but as a mechanism for enabling continued training while deployed, particularly for mission familiarity and emergency procedures, and for retraining on updated models of familiar aircraft.

Warping virtual space may have unintended effects on task performance, which is important for training applications. I investigated Redirected Touching deeply to determine its effect on task performance.

### 1.4.1 Summary of results

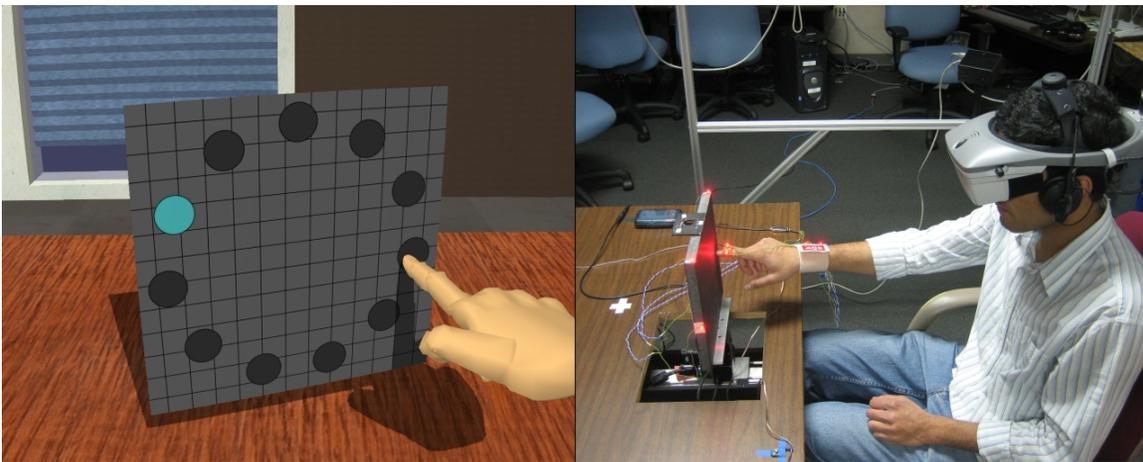


Figure 1.7: A user touches a virtual board that is oriented differently than the real board providing passive-haptic feedback

To make the scope of this research manageable, I focused on angular discrepancy between real and virtual objects. In two separate studies, participants were presented with a vertically oriented 10" x 10" virtual surface upon which they performed a rapid-aiming task (Figure 1.7). In the first study, angular discrepancies between seen and felt surfaces of up to 18° around the vertical axis (~4cm positional discrepancy at the left and right surface edges) yielded acceptable task performance relative to a one-to-one mapping. Furthermore, a pilot study of six research group members suggested that 18° was near the discrepancy detection threshold—12° discrepancies were usually undetected, and 24° discrepancies were usually detected.

The second study investigated whether participants could adapt to and train in a discrepant VE. Participants were divided into three groups: Real, in which participants did a rapid-aiming task on a monitor oriented at 18° in the real world; Virtual One-to-One, in which participants did

the task while wearing an HMD, with both real and virtual monitors oriented at 18°; and Virtual Discrepant, in which the real monitor was at 0° and the virtual monitor was at 18°. Results indicate that training in either virtual condition was not as effective as in the real condition. However, there was clear evidence of adaptation to the discrepancy. Participants were slower in the virtual discrepant condition than in the virtual one-to-one condition, but after adaptation, were no worse in error rate and throughput (a standard metric combining speed and accuracy).

## CHAPTER 2

# Redirected Passive Haptics<sup>1</sup>

Figure 2.1 shows a VE with five virtual easels. Suppose each easel has a paper pad on it, and you are tasked with drawing something precisely on each one. Drawing is easier with haptic feedback, which can be provided by putting five appropriately registered real easels into the tracked space.

Now suppose that Mr. Unsavory Character comes along and steals four of the real easels. Despite this setback, your virtual drawing task must not go unfinished. However, you need haptic feedback to complete your task. How might haptic feedback be reintroduced for the four missing easels?

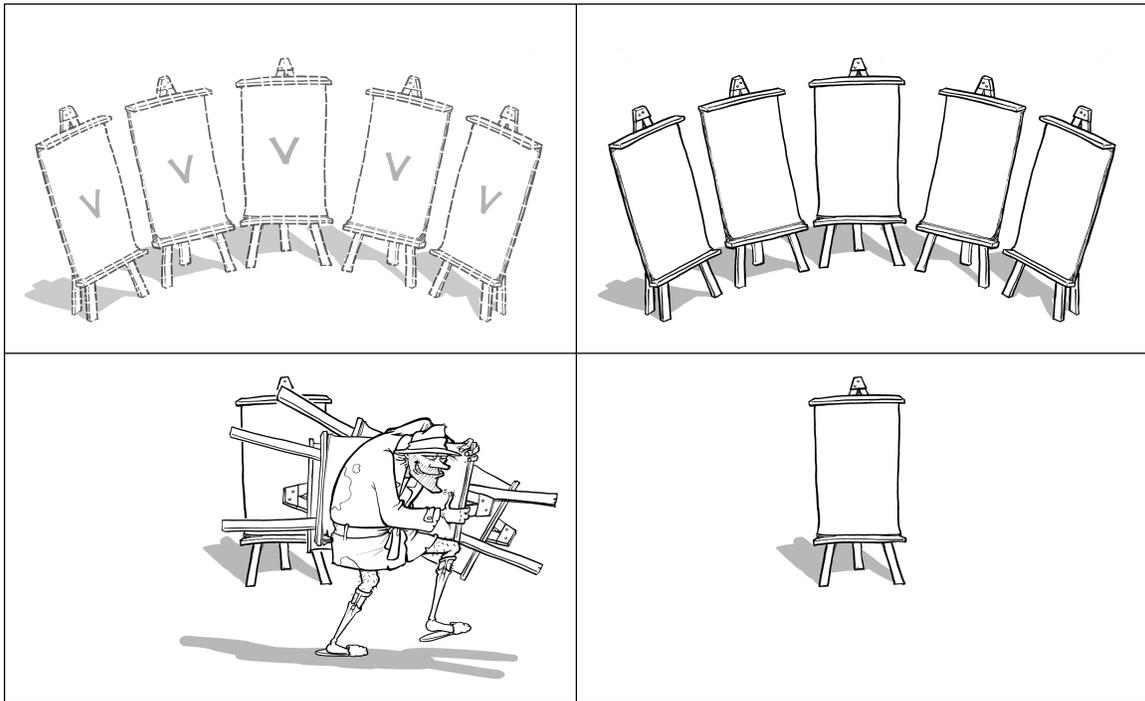


Figure 2.1: Five virtual easels (upper left) and five real easels (upper right) that provide haptic feedback, enabling a user to draw effectively on the virtual easels. Mr. Unsavory Character seeks to disrupt the task and steals four of the real easels (lower left). Can the task still be done with only one real easel (lower right)? (Artwork by Matt Nealon)

<sup>1</sup>Portions of this chapter were previously published elsewhere [Kohli et al., 2005].

Typically, passive haptics have been mapped one-to-one to their virtual counterparts. However, passive-haptic displays do not actually require a strict one-to-one mapping from positions in the VE to positions in the real world. They do require that a virtual object intended to provide haptic feedback map to a position in the real world that contains a real object. Likewise, every open space in the VE must map to open space in the real world. These mappings could be many-to-many, opening up the possibility that passive haptics could be used even if real and virtual world positions became decoupled. In the example above, is there some way to map each virtual easel onto the single remaining real easel?

One possibility is to **move the virtual world** to align the different virtual easels with the real easel (Figure 2.2). Different areas of the VE are mapped onto the real environment as needed. In this chapter, I present *Redirected Passive Haptics*, a technique I developed to explore this idea.

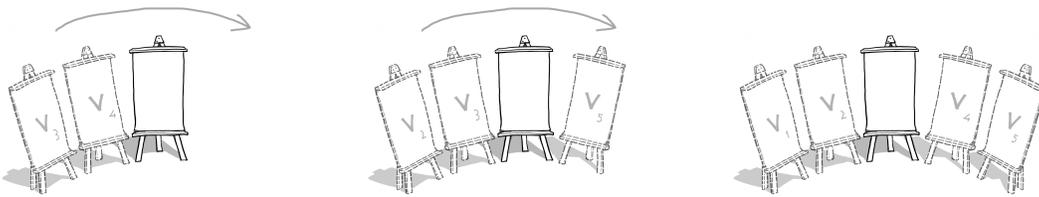


Figure 2.2: The virtual world is moved so that each virtual easel is in turn aligned with the real easel. (Artwork by Matt Nealon)

## 2.1 Moving the virtual world

The virtual world could be moved while the user remains stationary. For example, once drawing on the first virtual easel is complete, the virtual world could translate and rotate until the second virtual easel is aligned with the real one. However, this sort of manipulation might be expected to lead to disconcerting self-motion perception. A large discrepancy between a user’s vestibular sense (“I am stationary”) and visual sense (“the world is moving around me or I am moving through the world”) often results in discomfort and even simulator sickness [Pausch et al., 1992]. Ideally, the virtual world would move without these adverse effects.

Razzaque developed Redirected Walking, a technique for enabling users to explore VEs that are larger than the available tracked space [Razzaque, 2005]. It dynamically maps different areas

of the VE onto a smaller real space by injecting a discrepancy between a user’s real-world head rotation and virtual head rotation. During high-frequency head turns, the vestibular system is more sensitive than the visual system [Duh et al., 2004], and the injected discrepant visual rotation can be quite large and go undetected. Even when a user is not turning her head, small amounts of virtual-head rotation can be injected without her noticing. The injected rotational discrepancies cause users to walk along real-world paths that are different from their virtual paths, enabling navigation in larger-than-tracked-space VEs. One of the goals of Redirected Walking is to enable large-VE exploration without increasing the likelihood of simulator sickness. It seemed a reasonable choice for moving the virtual world. Nevertheless, Redirected Walking presents some difficulties in the context of passive-haptic feedback.

Passive haptics require positions in the VE to remain coupled with positions in the real world. Redirected Walking, by its nature, breaks this coupling by rotating the virtual world with respect to the real world. Suppose a real object and a virtual object are aligned. When Redirected Walking rotates the virtual world about the user’s head, the two objects become misaligned (Figure 2.3).

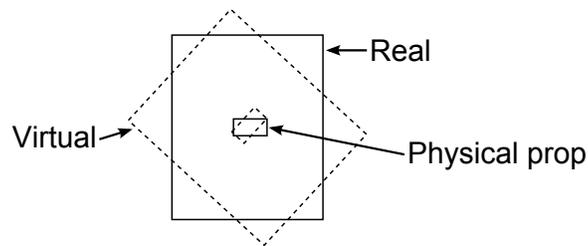


Figure 2.3: When the virtual world is rotated, real and virtual objects become misaligned.

Suppose the five virtual easels and the single real easel in Figure 2.1 are arranged such that they lie on a circular arc and that a user is standing at the center of the circle (Figure 2.4a). Assume the user remains stationary except for turning her head back and forth. As she turns her head, the virtual world is rotated such that one of the virtual easels aligns with the real easel (Figures 2.4b–c). The user can then approach the aligned easel and draw on it (Figure 2.4d).

Now suppose the user wanted to draw on the next virtual easel. If she remains in front of the current easel and turns her head, no amount of rotation about her head will align the real easel with the next virtual easel (Figures 2.4e–f). She would first need to return to the original circle center and then turn her head.

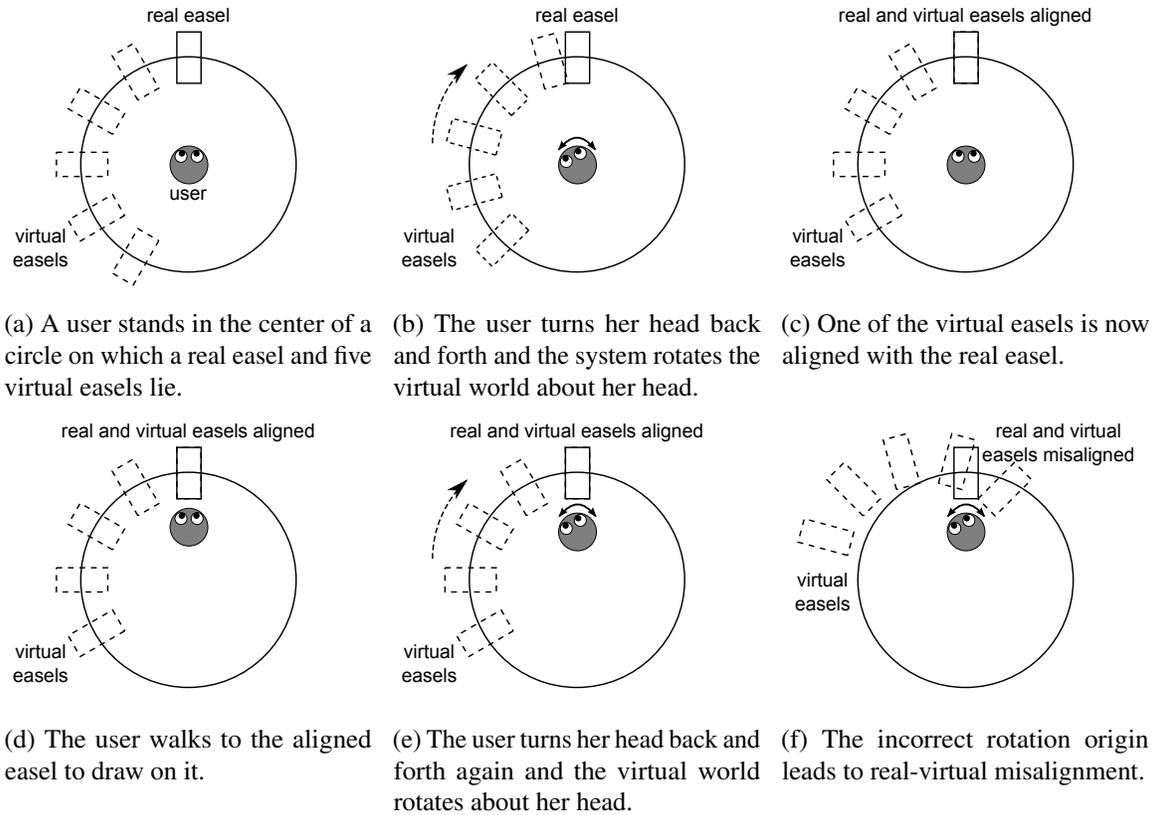


Figure 2.4: Rotating the virtual world around the user's head

Thus far, the easels have all been oriented towards the center of the original circle. Suppose instead that the easels all faced in the same direction (Figure 2.5a). Aligning objects via virtual-world rotation is now much more difficult. Even though the easels still lie on the same circle, rotation about the circle center can achieve only positional alignment, and not rotational alignment (Figure 2.5b).

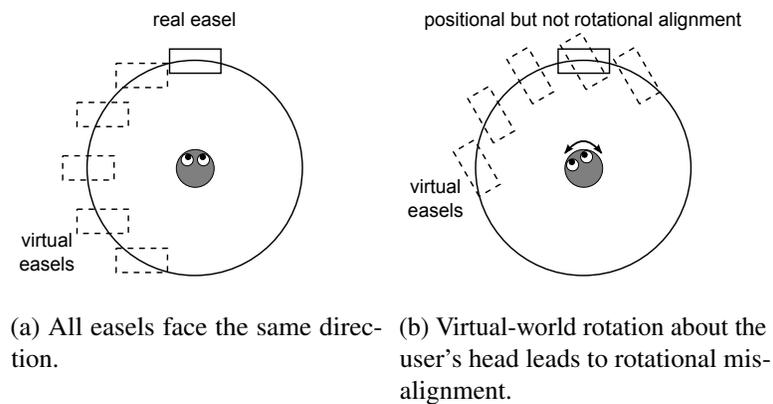


Figure 2.5: Difficulties with object alignment

All of this discussion has ignored the fact that users do not typically remain perfectly stationary, and that there may be other real and virtual objects in the environment, dramatically increasing the number of constraints imposed. It is theoretically possible to find a sequence of rotations about different points in the environment that would eventually align another virtual object onto the same real object. The problem can be cast as a motion-planning problem:

Given a user's start pose, desired end pose near a virtual object, and sets of real and virtual objects, determine a real- and virtual-collision-free path between the start and end poses that aligns the desired virtual object with a real object; the virtual world can be rotated about the user's head as she walks along the path.

Assume such a path is found for a given environment. If the path were to be followed by a robot, one could command the robot to follow the path. A human user, however, is less likely to follow commands exactly. Users typically want to explore more freely; even if a user is somehow enticed to follow the predetermined path, she is unlikely to do so perfectly. It is not clear under what circumstances all constraints can be satisfied. To test the basic concept, I simplified the problem.

## 2.2 Proof of concept

I created a VE of a room that was empty except for five differently colored cylindrical pedestals, chosen for their rotational symmetry (Figure 2.6). The real world was empty except for one cylindrical object intended to provide haptic feedback for all five virtual pedestals (Figure 2.7).

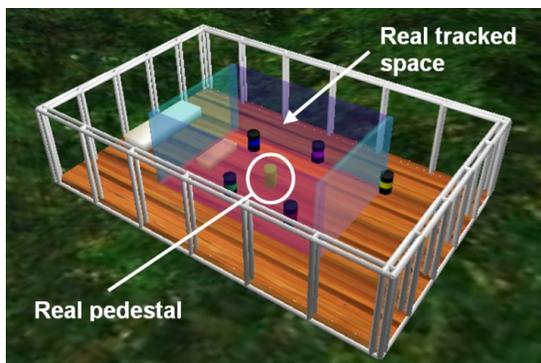


Figure 2.6: A VE with five differently colored virtual pedestals. The real pedestal and tracked space are superimposed.



Figure 2.7: A user touches the one cylindrical object intended to provide haptic feedback. The foam walls mark the tracked space boundaries.

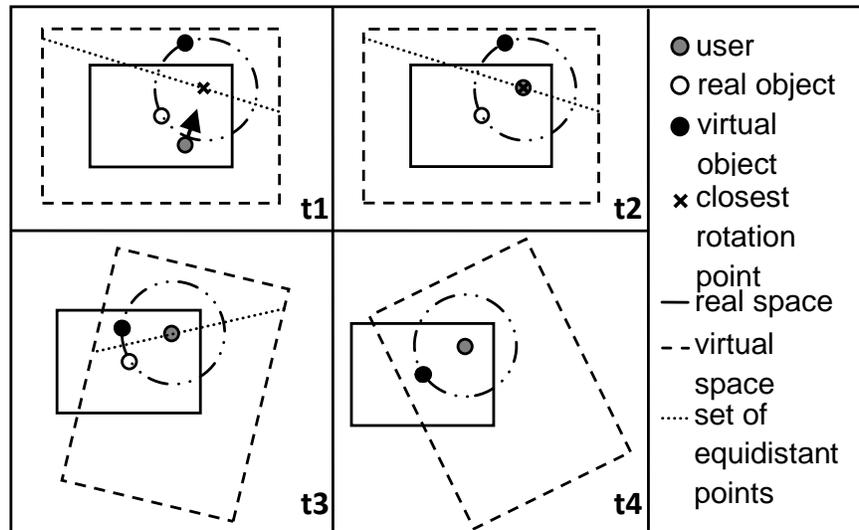


Figure 2.8: Step-by-step method to align virtual objects with real object: (t1) User guided towards rotation point, equidistant from real and virtual objects; (t2) user reaches rotation point; (t3) user turns head, virtual world rotates imperceptibly; (t4) real and virtual objects aligned

To bring a virtual object into alignment with its real counterpart by using just rotation (as in Redirected Walking), the rotation point must be equidistant from the real and virtual objects (so they lie on a circle centered at the rotation point, Figure 2.8).

The user must move to one of these rotation points and turn her head (more imperceptible redirection is possible when the user's head is turning). I needed a task that required users to do so. For a class project, Dorian Miller made a game called *Virtual Reality Quest*; it extended Redirected Walking by adding a distracting task. The game introduced a dragon avatar that, through its actions, guided the user through a predetermined path in the VE. At waypoints along the path, the dragon would appear with a fire in its mouth, and fly back and forth in front of the user. The user was required to douse the flame with her hand before moving on. As the dragon flew back and forth, the user had to turn her head, thus enabling more redirection.

I used a similar distracting task for my VE. Users were told that droids had stolen a precious gem collection. To recover the gems, users needed to activate and destroy all the droids. Users activated a droid by walking into its transporter beam (Figure 2.9, left). The beams were placed such that they were in the plane of potential rotation points. When a user reached the beam, a droid appeared, and the user jabbed at it with a hand-held weapon until it was destroyed. The droid traced out a sinusoidal path that required the user to look back and forth several times (Figure 2.9, right).

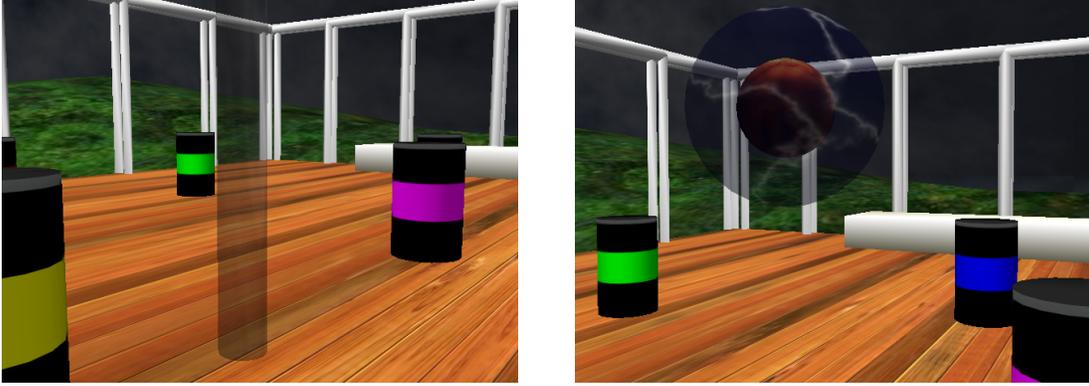


Figure 2.9: A transporter beam (left) and droid (right)

When a user turned her head during the distracting task, the system rotated the virtual world in the opposite direction by a rate ( $\omega_{VE}$ ) based on the head's angular velocity ( $\omega_{head}$ ) and the current angle between the real and virtual pedestals ( $\theta_{RV}$ ).  $\omega_{VE}$  was computed as:

$$\omega_{VE} = \omega_{head} * -c * (0.25 + 0.75 * \sin(\frac{\theta_{RV}}{2})) \quad (2.1)$$

$c$  is a predefined constant where  $1 + c$  represents the maximum gain applied to apparent virtual head rotation;  $c$  was set to 0.45 (e.g., if  $\omega_{head} = 60^\circ/sec$  and  $\theta_{RV} = 180^\circ$ , then  $\omega_{VE} = -27^\circ/sec$ , leading to an apparent virtual head angular velocity of  $87^\circ/sec$ ). This value for  $c$  was chosen based on an informal investigation of how much rotational gain is undetectable.<sup>2</sup> The highest magnitude rotational gain is applied when  $\theta_{RV} = 180^\circ$ .

Users inevitably did not turn precisely about the ideal rotation point. The system tolerated error below a certain threshold, but if it became too large, the user was directed to another transporter beam around which the virtual world could be rotated again to more closely align the real and virtual objects.

### 2.3 Informal evaluation

As part of a class project demonstration, eight users walked around a VE (~8m x ~13m) in a tracked space one-quarter that size and touched five virtual pedestals, which were actually the same physical object in the real world. Users expressed surprise that they were able to touch all five virtual

<sup>2</sup>Steinicke et al. [2008a] present a more rigorous exploration of undetectable gain thresholds for Redirected Walking.

pedestals without detecting the visual manipulation—even those users who had previously heard an explanation of how the system worked. Figure 2.10 shows one user’s paths in the real and virtual environments.

Despite the contrived scenario, this technique demonstrates that it is possible to map multiple virtual objects onto a single real object by moving the virtual world. However, generalizing the technique requires significant additional research. A more complex VE would require much more sophisticated user route planning. Adequate route planning may even be impossible for dense environments, especially if objects of different shapes and sizes are used. Others have extended this work with a framework for interacting with less-constrained virtual objects [Steinicke et al., 2008b], but developing a practical and general technique is ongoing work. Peck has further explored using distractors to enhance Redirected Walking [Peck et al., 2008; Peck, 2010].

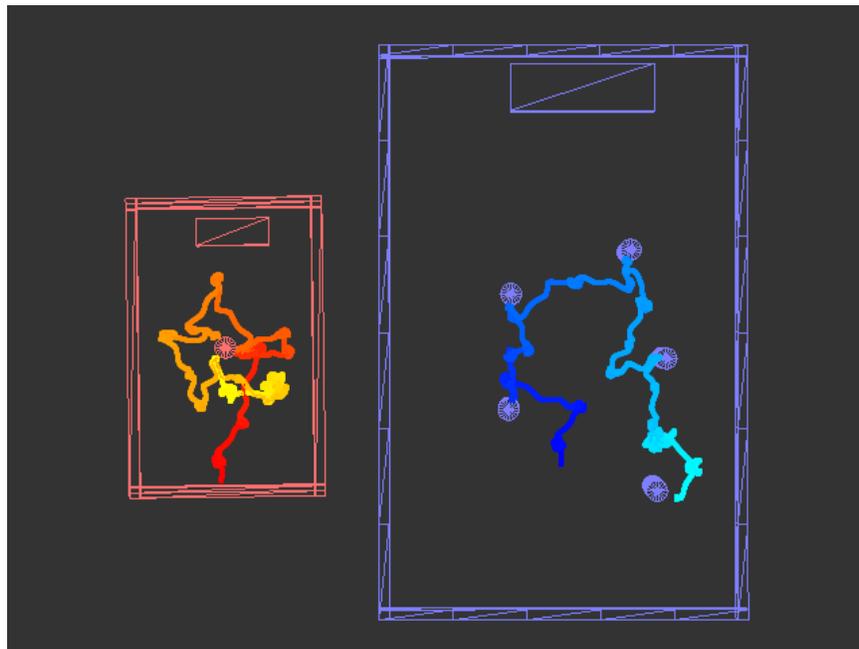


Figure 2.10: One user’s real (left) and virtual (right) paths

## CHAPTER 3

# The Haptic Hand<sup>1</sup>

Suppose you are once again tasked with drawing on a virtual easel, as in Chapter 2. This time, you have a single real easel and a single virtual easel that properly map onto each other (Figure 3.1). Mr. Unsavory Character returns to wreak havoc again. He shoots the real easel with his shrink ray, making the easel half its original size, and leaves as unceremoniously as he arrived. Needing to draw on the entire virtual easel, how might you do it given the real-virtual size discrepancy?

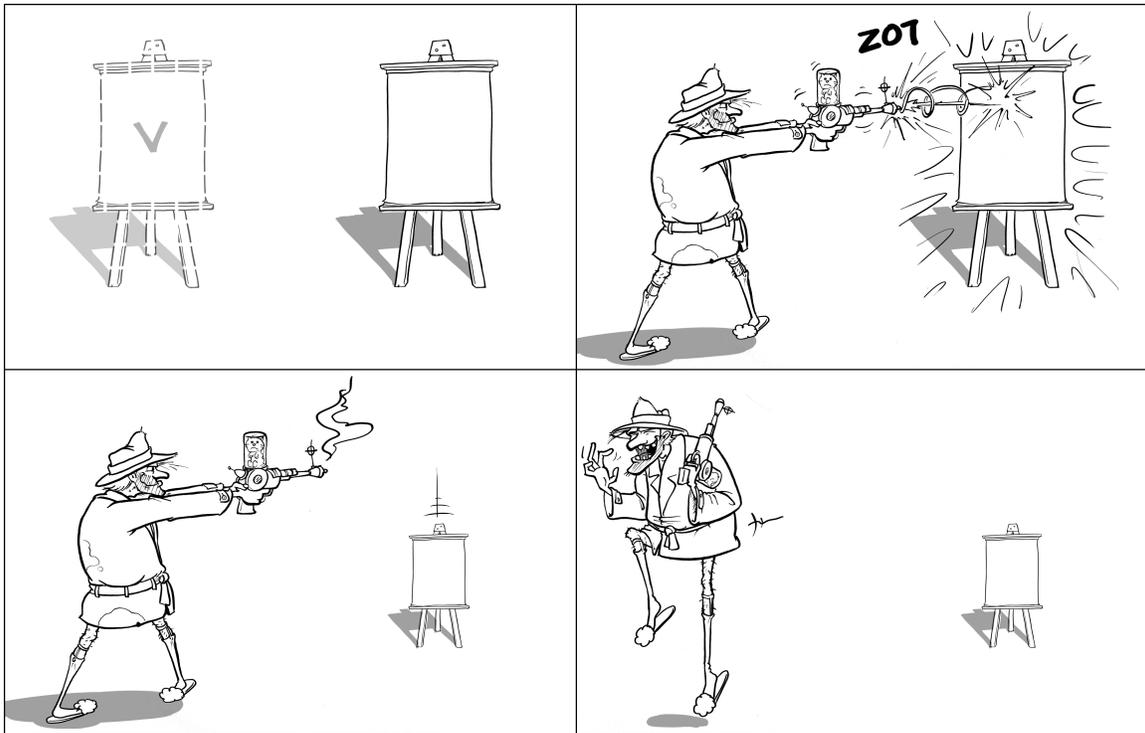


Figure 3.1: A virtual easel and a real easel that provides haptic feedback (upper left). Mr. Unsavory Character shoots the real easel with a shrink ray (upper right, lower left), yielding a real easel that is smaller than the virtual easel (lower right). Can a virtual drawing task still be done using the small easel for haptic feedback? (Artwork by Matt Nealon)

<sup>1</sup>Portions of this chapter were previously published elsewhere [Kohli and Whitton, 2005].

One possibility is to move the virtual easel to align different parts of it with the real easel. In this chapter, I explore the idea of **moving a virtual object** to align it with a real object, in the context of a virtual widget interface; I present *The Haptic Hand*, an interaction technique that maps different parts of a larger-than-hand virtual interface to the non-dominant hand.

### 3.1 Bimanual interaction and indirect manipulation

Guiard studied the distribution of work between the dominant and the non-dominant hands and classified tasks as *unimanual* (e.g., one-handed throwing), *bimanual symmetric* (identical actions performed by each hand), and *bimanual asymmetric* (both hands perform closely coordinated but different actions) [Guiard, 1987]. Studies have shown that two-handed interaction techniques can provide significant advantages over one-handed techniques [Buxton and Myers, 1986; Balakrishnan and Kurtenbach, 1999]. These observations have been used in the development of user interfaces for VEs [Cutler et al., 1997; Hinckley et al., 1997b, 1998; Lindeman et al., 1999]. The general idea is to provide a user interface that is held and/or coarsely adjusted by the non-dominant hand (NDH), and manipulated more finely by the dominant hand (DH). However, the lack of haptic feedback makes this fine manipulation difficult in VEs, much as it does in the real world: writing on a sheet of paper held in the air is much more difficult than when the paper rests on a desk.

In some VEs it is necessary to provide the user with a control for precisely manipulating VE parameters. For example, when interactively designing a VE, one may want to adjust an object's color or height. This function is often provided through widgets (e.g., buttons, dials, and sliders) on a virtual hand-held interface panel. Research has shown that using physical props for haptic feedback helps users interact with VEs more effectively. A common physical prop is a tracked tablet or paddle, held in the NDH [Poupyrev et al., 1998; Schmalstieg et al., 1999; Coquillart and Wesche, 1999; Lindeman et al., 1999, 2001]. In the VE, a virtual tablet aligned with the real one presents the user with an interface panel of widgets. The user can move the tablet with the NDH as necessary and interact with the widgets using the DH.

It is impractical to carry a physical paddle in cases where both hands are required to perform some task in a VE [Lindeman et al., 2001]. What should be done with the paddle when the NDH is otherwise occupied? One can hang it down by one's side, but that can be cumbersome. Also, using

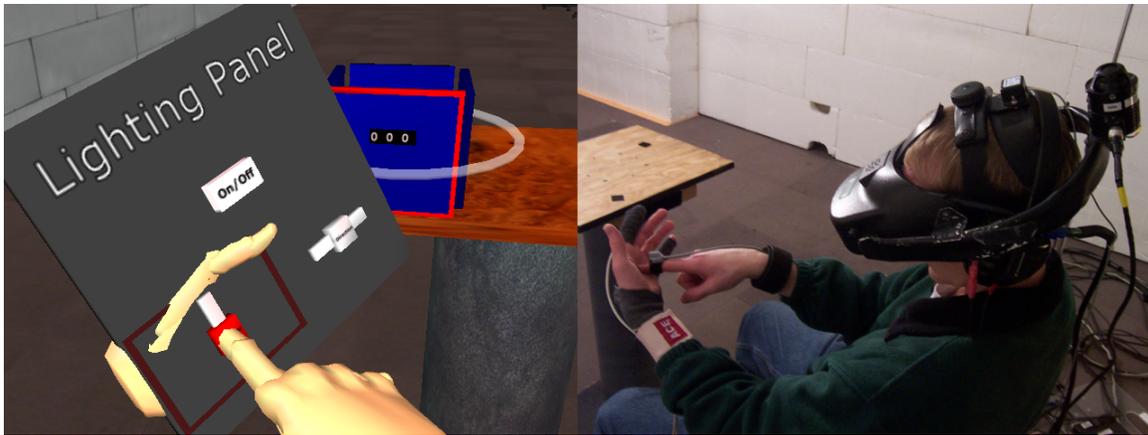


Figure 3.2: A user interacts with a virtual interface panel by touching his non-dominant hand.

multiple props for different tasks increases the number of objects that must be tracked, and there must be a sensible place to store these props if the user needs to switch between them often.

### 3.1.1 Exploiting proprioception

Proprioception is our sense of the position, orientation, and movement of our limbs, joints, and muscles (e.g., one can accurately touch one's nose with one's eyes shut). Research suggests that, independent of visual feedback, proprioception gives humans a good sense of where their hands are relative to one another [Hinckley et al., 1997a].<sup>2</sup> This idea has been used in the development of user interfaces for immersive VEs [Mine et al., 1997]. I developed a technique, *The Haptic Hand*, that exploits proprioception; it uses the NDH itself to provide haptic feedback for a portable 2D user interface in an immersive VE (Figure 3.2).

Implementing this technique presented several challenges:

1. Properly tracking a user's hands is difficult; the relative position of the two hands is critical. Because people have varying hand sizes and shapes, trackers cannot be mounted in exactly the same way on every person. This variability necessitates additional calibration.
2. It is difficult to provide an accurate representation of a user's hands in a general-purpose VE without explicit per-person measurement and calibration.

---

<sup>2</sup>In the presence of visual-proprioceptive discrepancy, vision usually dominates. This idea is discussed in Chapter 4.

3. The palm is non-planar. Dragging an object along the palm may not feel smooth enough to adequately emulate a planar surface.

I conducted an initial usability study to determine the feasibility of the technique. Results from the study were sufficiently positive to warrant further development.

## **3.2 The idea: *The Haptic Hand***

I created a system in which a user interface is presented to the user on a plane fitted to the NDH palm. The interface panel contains widgets that are used to manipulate and modify objects in the VE. Touching the NDH palm with the DH gives both hands haptic feedback when manipulating these widgets: the DH senses that it has touched the NDH, and the NDH senses where on the palm it was touched. Although the study reported here does not include a direct comparison with other techniques, it is my hypothesis (to be tested in future work) that this method is easier to use than a system with no haptic feedback and less cumbersome than systems that use physical paddles. My goal for this work was to demonstrate feasibility and to collect subjective, qualitative data about usability.

### **3.2.1 The interface panel**

To test using the NDH to provide haptic feedback, I designed the interface panel to include widgets that control both discrete and continuous variables: buttons (discrete) and sliders (continuous) (Figure 3.3). In its initial state, the panel floats in the VE; it can easily be initialized to appear in a location conveniently within the user's reach.

### **3.2.2 Interacting with the virtual environment using hand gestures**

The system assumes that, at a minimum, there are virtual representations (avatars) of the user's two hands. The NDH model must support at least two poses: open and closed. The DH model must support a pose with a pointing index finger. In this system, these three are the only supported hand poses. The user interacts with the interface panel using two simple hand gestures: opening and closing the NDH. These gestures are used for grabbing, holding, and releasing the interface panel; activating widgets; and manipulating other objects in the environment.

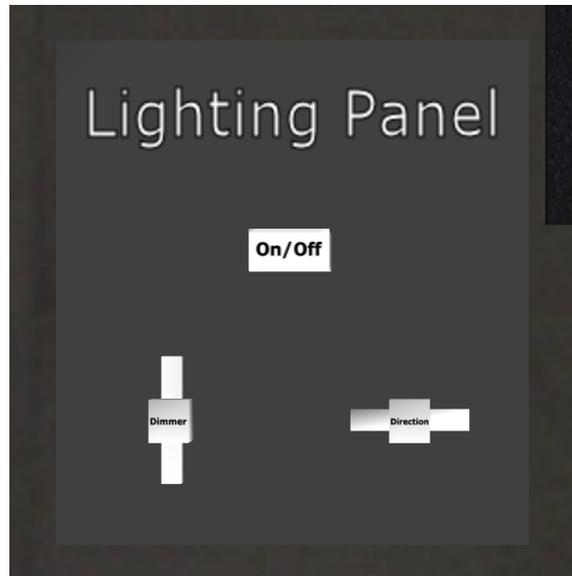


Figure 3.3: A sample interface panel with buttons and sliders

### 3.2.3 Widget activation

One of the challenges of using the NDH for haptic feedback as opposed to a fixed-size (large) physical paddle is the limited amount of real estate available on the palm. Unless tracking is very precise, many widgets cannot be active simultaneously. Consequently, we need a way to select which widget is currently active.

In my initial implementation, which proved unsatisfactory, the panel was initially placed in the VE at some location near the user, and widgets were activated by moving the open NDH behind the desired widget. Once activated, the user could manipulate the widget with the DH. Closing the NDH while intersecting the panel enabled the user to grab the panel and move and rotate it as desired to position it more comfortably. However, this particular implementation suffered from precisely the same problem that I intended to tackle: because the interface panel had no real-world physical manifestation, there was no way for the user to steady the NDH. It was therefore extremely difficult and tiresome to keep a widget active; the NDH would drift in and out of the widget's activation area.

Given this experience, I took a new approach: the panel is snapped to the NDH so that the panel and NDH move and rotate together in the VE. An activated widget remains active until explicitly deactivated by the user, instead of being affected by the hand's natural tendency to drift when nothing

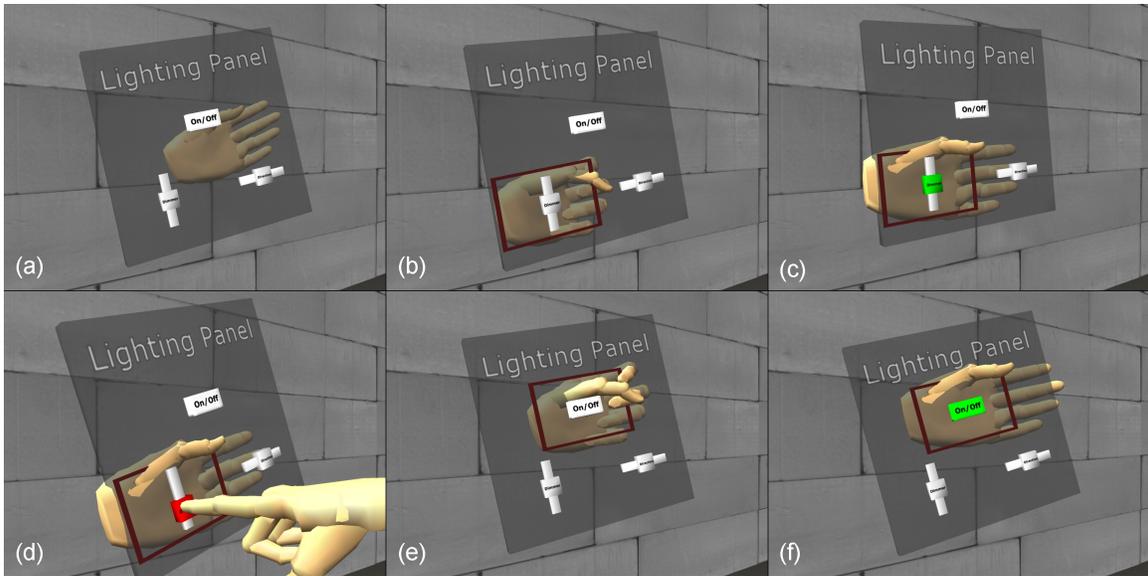


Figure 3.4: Interface panel manipulation: (a) the panel is not attached to the NDH; (b) the NDH is closed and intersects the panel to highlight the desired widget; (c) the NDH is opened to snap the desired widget to the center of the palm; (d) the DH is used to manipulate the active widget; (e) the NDH is closed and moved to a new desired widget; (f) the NDH is opened and the new widget is snapped and activated.

steadies it. To implement this approach, I reversed the effect the two hand gestures have on the panel and widgets (Figure 3.4).

Initially, the panel is not attached to the NDH, and the two are not intersecting. To select which widget to activate, the user must close her NDH and move it so that it intersects the panel. At this point, the widget that is closest to the NDH is highlighted via a surrounding red rectangle. If the NDH is opened while a widget is highlighted, the panel snaps to the NDH such that the highlighted widget is centered on and oriented with the palm. The highlighted widget changes to a second color to indicate that it has been activated and can be manipulated with the DH. The panel now moves and rotates with the NDH.

To activate a different widget the user closes the NDH, selects the desired widget, and opens the NDH, causing the new widget to snap to the center of the palm. The user can let go of the panel by closing the NDH and moving it away from the panel.

### 3.2.4 Widget manipulation

Once a widget has been activated, it is manipulated by bringing the tip of the DH index finger in contact with the widget on the NDH palm. An active widget will change to a third color when being manipulated (Figure 3.4d). Buttons are pressed and released using discrete, ballistic motions, whereas sliders are changed by continuous dragging along the NDH palm's surface.

### 3.2.5 Calibration

To snap the interface panel to the palm of the user's NDH, we must know where the surface of the palm is. The user's hands are tracked with Polhemus Fastrak® magnetic trackers. Because the palm surface must be unobstructed, a tracker is rigidly mounted on the back of the user's NDH using stretchable sports bandages (Figure 3.5).

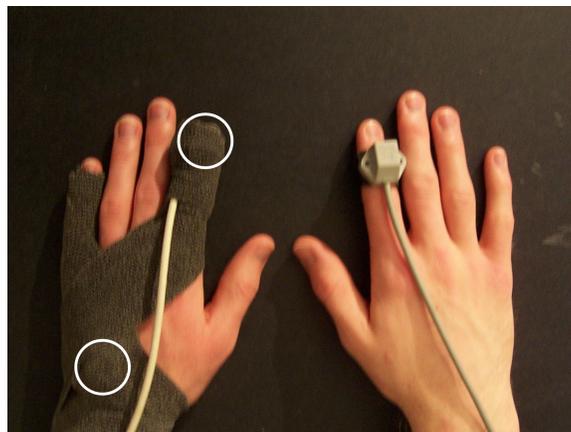


Figure 3.5: The NDH (left) and DH (right) with trackers attached. The white circles indicate where the NDH trackers are located underneath the sports bandage: one on the fingertip to monitor whether the palm is open or closed, and one on the back of the hand to determine its pose.

Because different people's hands are different in shape and size, the tracker mounts differently relative to the palm surface for each person. To calibrate the user's palm relative to the NDH tracker, the user's hand is placed flat in a fixed pose on a pedestal. The transform between the NDH tracker and the fixed pose is determined and then applied to all subsequent tracker readings. The DH index-finger tracker is placed 2cm from the fingertip.

### **3.2.6 Manipulation issues**

I determined when the user was manipulating a widget by detecting collisions between the user's virtual DH fingertip and the widget. This technique usually worked well. However, since the surface of the palm is not truly planar, and the user can make any number of small motions that affect her palm shape, the approximated plane and the surface of the NDH palm do not always coincide. As a result, the system may not always know when the DH fingertip is physically touching the NDH palm. One consequence of this is that lifting the DH finger off the NDH palm does not immediately deactivate widgets. This is not a problem for buttons because they require only ballistic input and have two discrete states: pressed or released. Sliders, however, are continuous input devices; lifting the DH finger off of the NDH palm may leave the slider still active until the virtual finger and slider are no longer colliding. In this situation, the user once again has no way to steady the DH finger, and the slider value wanders from the intended value very easily.

To alleviate this problem, I implemented a slider locking mechanism. If the user holds the slider's value for at least 200ms, the value is locked. If the user removes the virtual finger from the slider within the next 500ms, the value of the slider snaps back to the locked value. If the virtual finger does not leave the slider within those 500ms, then the slider value will not snap back. During those 500ms, the slider behaves normally (i.e., values are updated). These time thresholds were chosen by trial and error, but they seem to work acceptably in practice. This slider locking mechanism is similar to techniques proposed for solving the same problem for touch-sensitive tablets. In particular, Buxton, et al. discuss keeping a short FIFO queue of tracking samples [Buxton et al., 1985]. When the user lifts her finger off of the tablet, the oldest sample in the queue is used, and the queue is cleared. The queue's length is determined through experimentation.

## **3.3 Usability study**

I ran a small usability study to test the idea of using the NDH as haptic feedback for user interfaces in immersive VEs. The goal was to determine if the technique is worth pursuing further. As such, I did not run a rigorous study comparing the technique with a physical paddle interface, nor did I make extensive performance measurements. I was more interested in finding out how difficult

it is to learn to use the interface, and whether the technique enables both discrete and continuous interaction.

### **3.3.1 Materials and methods**

#### **Participants**

Eight people participated (six male, two female, seven right-handed) and were paid \$5 each for their time. Participants were a mix of graduate and undergraduate students of various majors. Three participants had never been in a VE, three had been in a VE occasionally, and two had been in a VE many times.

#### **Equipment**

In addition to the Polhemus Fastrak® magnetic hand trackers mentioned, participants wore a Virtual Research Systems V8 HMD. The head was tracked by a 3rdTech™ HiBall™-3000 wide-area tracker. Participants were immersed in the VE for approximately 15 minutes and were seated.

### **3.3.2 Study design**

The study consisted of a training session and a matching task. Before entering the VE, participants were shown the environment and interface panel on a desktop display and were told how to interact with the panel and with objects in the environment. The training session in the VE enabled participants to familiarize themselves with these interactions while immersed. Once participants felt comfortable, they were given a matching task to perform.

#### **Training session**

The interface panel used for the training session controlled a red light in the environment (Figure 3.3). A button labeled “On/Off” toggled the light, and sliders labeled “Dimmer” and “Direction” controlled the light’s intensity and the lateral direction it came from, respectively. The training environment also included a four-sided object (essentially a cube without the top and bottom faces) sitting atop a pedestal in front of the participant. Each cube face was selectable by the DH fingertip. The cube was surrounded by a ring that served as a handle to rotate the cube about

its vertical axis. Intersecting the ring with a closed NDH and moving around the ring with the intersecting hand rotated the cube. This action required the participant to let go of the interface panel.

### Matching task

The participant's goal in the matching task was to make two cubes look alike (Figure 3.6). The right cube was the reference cube; each face was a different shade of blue, and had a different number from 0 to 100 on it. The interface panel had two sliders ("Number" and "Color") and two buttons ("Reset Face" and "Done"). The "Number" and "Color" sliders adjusted the number and color of the currently selected face, respectively. The "Reset Face" button reset the currently selected face to a default shade of blue and the number 0. Upon pressing the "Done" button, participants received audio cues indicating whether their attempt was successful.

Participants were told that the numbers on the two cubes had to be exactly the same, but that the colors had to be only "close." A fairly large threshold was used to determine if colors were matched,

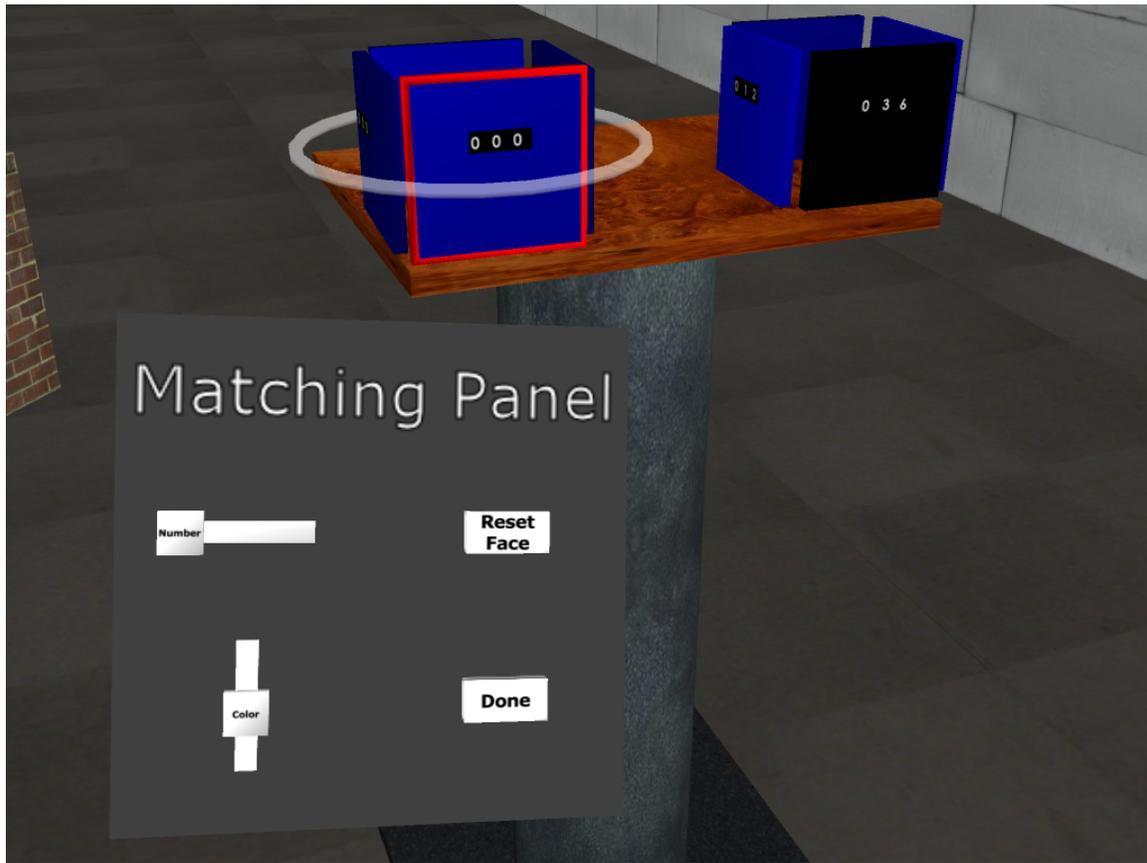


Figure 3.6: The interface panel and environment for the matching task

to avoid potential frustration from performing a perceptual task. However, I did want both horizontal and vertical sliders to test both dimensions of the NDH palm surface plane.

When the left cube was rotated, the right cube rotated in synchrony, so the participant always saw the two corresponding faces that were to be matched. Participants had to rotate the left cube several times to get to all four sides. This task required participants to discard and reacquire the panel several times and, consequently, it enabled me to query the participants on the difficulty of these actions.

## **Measures**

After participants had completed both the training and matching tasks, they were orally interviewed with several open-ended questions. These questions were designed to elicit comments on fatigue, ease of learning to use the panel, ease of manipulating widgets, and ease of discarding and reacquiring the panel as needed for their task.

Following the interview, participants were given a written exit questionnaire with questions arranged on Likert scales (Appendix A.1). These questions asked participants to rate ease of use of the different user interface components on a scale from 1 (extremely easy) to 7 (extremely difficult) and complemented the open-ended comments received during the interview.

## **3.4 Results**

Overall, participants reported the interface intuitive to use after becoming familiar with it over the 10-15 minutes they used it for the training session and the matching task.

### **Using the widgets**

Figure 3.7 shows how participants rated the ease of interacting with the system. All but two participants rated widget activation as moderately easy, easy, or extremely easy. The remaining two rated it neutral. Universally, participants found buttons to be easier to use than sliders. Half of the participants rated sliders to be moderately easy, easy, or extremely easy. Two participants rated sliders neutral, and two rated them moderately difficult. Those who rated sliders neutral had not been told about the slider locking mechanism due to experimenter error, and I observed that the two who

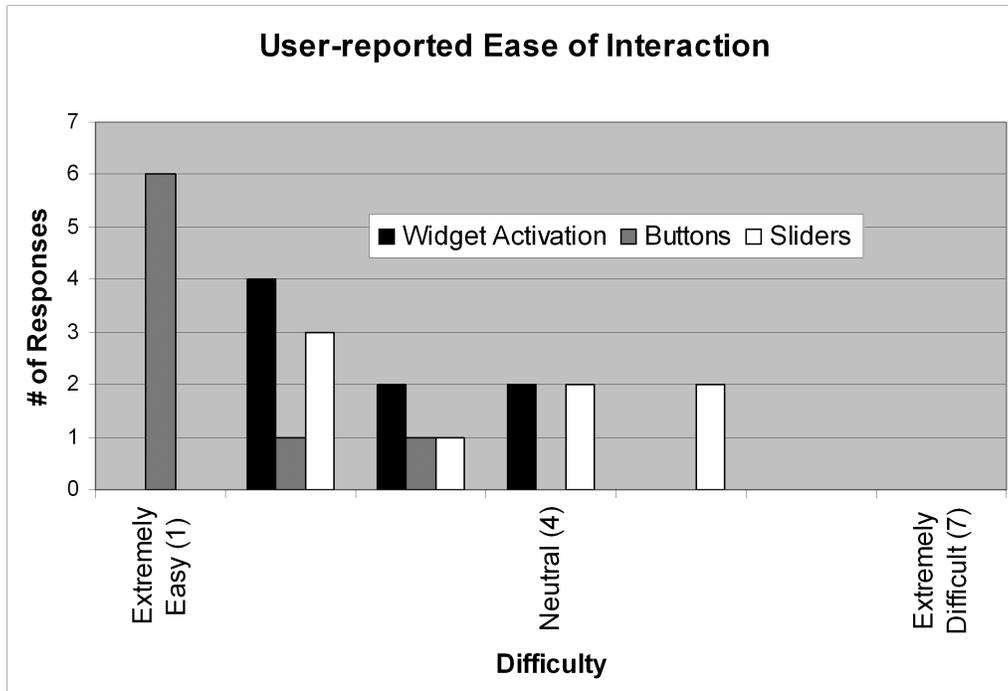


Figure 3.7: User-reported ratings of ease of interaction on a scale from 1 (extremely easy) to 7 (extremely difficult)

rated sliders as moderately difficult did not hold the NDH palm flat when open, leading to the issues discussed in Section 3.2.6.

The numeric ratings were supported by user comments in the interviews:

*At first I found myself not actually touching my hand, but once I realized it's a lot easier to go all the way down and touch [my NDH], it got a lot better. It made it much quicker. . . instead of [slowly] moving my hand [(DH finger)] until I got close enough, I could just put my hand [(DH finger)] down and control [the widgets].*

*[Touching the NDH] was fine; . . . it made it easier to actually comprehend the contact with the widgets.*

*Selecting the various widgets, that was really easy. . .*

*[For] buttons . . . it was fine, it was good to have a stop, but with the sliders, your hand isn't perfectly smooth, so you can hit something [that restricts your movement].*

*. . . eventually my [DH] finger ran out of room to go, especially when [trying to] go all the way to 99. . . you're all the way over here on the palm, right near the [NDH] fingers, so that gets pretty difficult.*

*During the color section, I would touch the bottom of the [slider], and my finger was too low [and off the bottom of the palm]. I had to go a little bit higher to [use the slider].*

## Interacting with the panel and cubes

After some practice, none of the participants had any difficulty discarding or reacquiring the interface panel. All participants felt that moving and rotating the interface panel when it was attached to the NDH was easy. Several participants found rotating the cube to be difficult until they adjusted their chair to a more comfortable position.

*No trouble getting rid of the panel... [but] the panel was just a little too big to leave in the scene and yet still have a view of the two cubes.*

*Once I got used to the interface, it was pretty simple.*

*I actually liked [moving and rotating the panel]; that was extremely handy. The cube was difficult to rotate. ... I found that if I grabbed [the cube] from above as opposed to from under, I could control it better.*

*[Moving and rotating the panel] was just like moving my hand. When I was trying to [rotate the cube] all the way around, I had a tendency to let my palm slide off of the ring.*

## Fatigue

A participant whose chair was improperly positioned in the real environment reported fatigue from having to stretch out his arm to rotate the cube. Otherwise, none of the participants reported having felt fatigued as compared to everyday arm and hand usage while seated.

## Suggested improvements

Participants suggested improvements for the next iteration of the system:

*I liked the locking behavior on the sliders. It was just a little bit odd to see [a slider] snap back, but it was better to have it snap back than for my finger to move the slider off the originally intended position. Some visual feedback that you've locked the slider might be nice though.*

*The sliders were a little too sensitive, I thought. ... If it had worked more consistently with my hand I think it would have been a little easier—maybe I didn't hold my hand flat enough.*

*I think the size of the widget panel was somewhat restrictive. ... If the widget panel was a cube on your hand that you could rotate. ... then you wouldn't have half your view blocked by this big panel. I wanted to be able to look at the cubes clearly while manipulating the panel, and I felt a little bit like it was one or the other [because of the low field of view].*

### 3.5 Future work

I have presented The Haptic Hand, a user interface technique for VEs that makes use of the NDH for haptic feedback. The technique enables users to map different parts of a larger-than-hand virtual interface onto the NDH by moving the virtual interface. An initial usability test indicates that the technique is viable and warrants further development.

There are several directions for future research. A more performance- and precision-oriented study directly comparing the technique with systems using physical paddles would provide valuable information on the class of tasks suitable for the technique.

A contact switch or pressure sensor on the DH fingertip would provide better information about when the fingertip and NDH palm are in contact, lessening the dependence on a slider locking mechanism. Also, more precise hand-size calibration techniques may significantly increase the system's effectiveness. For example, given more data on a user's hand size, the widgets could be scaled such that the user's DH finger will not move off the surface of the NDH palm (e.g., when setting extreme slider values).

Most current head-mounted displays considerably restrict the field of view. When the interface panel is large, the restricted field of view often causes the panel to obstruct other objects in the environment. This makes it necessary for the user to look back and forth between the panel and the objects they are manipulating with widgets (the same is true for physical paddle interfaces). I have several ideas:

- When a widget is activated, do not display any of the rest of the interface panel—display only the hand and currently snapped widget. Upon widget deactivation, the entire panel will be visible again.
- Use transparency to give a view of both the panel and the environment.
- Map the interface to a cylinder (or truncated cone) instead, and place it around the user's non-dominant forearm.

This last idea about the cylinder has several potential advantages. Since the forearm provides a larger surface than the palm, all of the widgets can be active simultaneously. The length of the forearm

will also enable longer dragging operations (e.g., sliders with much longer tracks). Additionally, the forearm (essentially) is a single rigid body, so it can be tracked with a single rigidly mounted tracker, calibrated to determine the forearm's primary axis of rotation.

More recently, researchers have projected user interfaces onto the real non-dominant hand [Mistry et al., 2009] or other parts of the body [Harrison et al., 2010] and have created user interfaces that use the Microsoft® Kinect® or other depth cameras to detect when a user is touching her non-dominant hand with her dominant index finger [Gustafson et al., 2011; Dezfuli et al., 2012].

### **3.6 Insights from mapping a virtual object onto the non-dominant hand**

This study yielded several interesting insights:

1. In general, when exploring an environment through touch, our hands serve as haptic input devices, relaying to us the haptic output provided by the environment. With this technique, the NDH served not only as a haptic input device to take in information from the environment, but as a haptic output device as well, providing haptic feedback to the dominant index finger.
2. The NDH acted as a haptic input device in an additional sense: when moving the dominant index finger along the NDH surface, the user is keenly aware of the index finger's position in relation to the NDH surface's bounds.
3. Insight #2 implies that this technique increases the amount of information a user receives, compared to a tablet-based interface. The user knows not only *that* the DH is touching the NDH, but also *where* the DH is touching the NDH. Consequently, after making contact with a slider the user can turn to look away from the interface and continue to use the slider properly by feeling where the DH is touching the NDH. I did not explicitly study this idea, but anecdotally, several participants were able to interact with the user interface in exactly this manner. This behavior was particularly useful because of the HMD's small field of view; by dropping the hand out of the field of view, users could better see the virtual cubes.
4. Many different virtual widgets were mapped onto the same real surface: the NDH. Although button interaction was more often rated as easy than slider interaction, moving the virtual

interface to map different parts of it onto the same real surface did not generally pose a problem for users.

5. A curved surface (the NDH) was used to approximate a planar surface (the user interface panel). Users were still able to interact with the user interface panel, despite the shape discrepancy between the virtual surface and the physical surface.

That users *can* interact with shape-discrepant real and virtual objects (insight #5) is the subject of the remainder of this dissertation.

## CHAPTER 4

# Redirected Touching<sup>1</sup>

Mr. Unsavory Character returns to find that all of his clever attempts to derail the virtual drawing task by stealing (Chapter 2) and shrinking (Chapter 3) easels have been thwarted. Seeing a new real-virtual easel pair, he angrily picks up the real easel and bends it so that it is curved (Figure 4.1). Snickering, he slinks away. Is it still possible to draw effectively on the virtual easel even though the real easel is now a different shape?

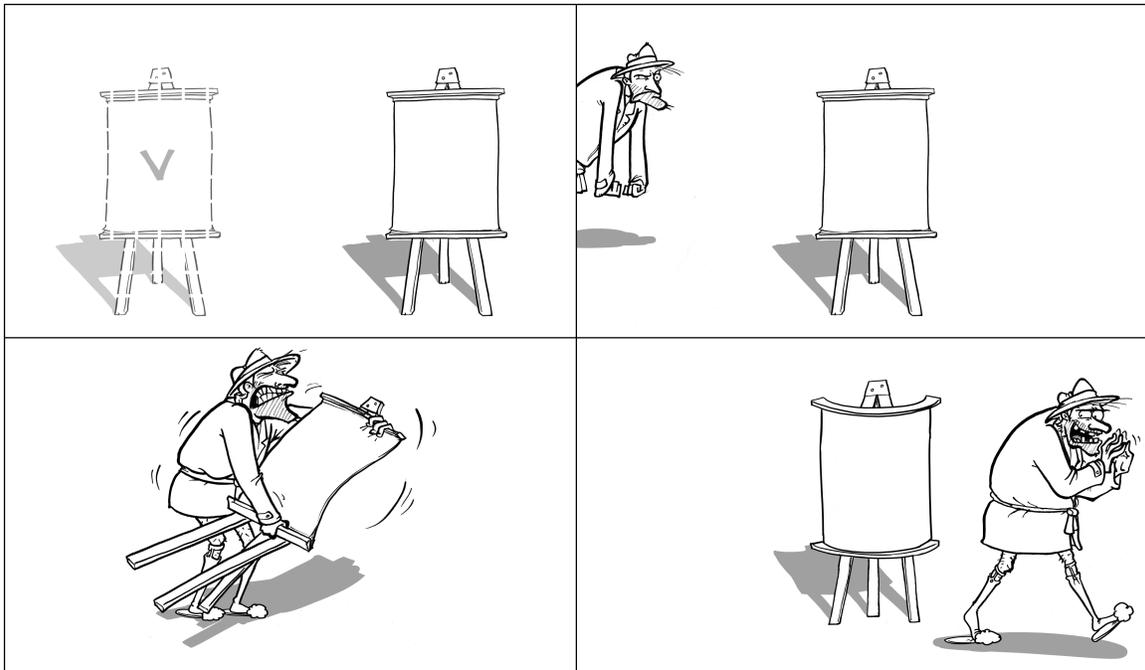


Figure 4.1: A virtual easel and a real easel that provides haptic feedback (upper left). Mr. Unsavory Character sees he must try something new (upper right), so he angrily bends the real easel (lower left), yielding a real easel that no longer matches the shape of the virtual easel (lower right). Can a virtual drawing task still be done despite the real-virtual shape discrepancy? (Artwork by Matt Nealon)

<sup>1</sup>Portions of this chapter were previously published elsewhere [Kohli, 2009, 2010].

It turns out that it is. Researchers have mapped differently shaped virtual objects onto real objects mechanically [McNeely, 1993]: when a user reaches for a virtual object, a robotic arm places a real object in front of the user’s hand. One such robot has a shape-approximation device with several corners and curved and flat edges [Tachi et al., 1994]. While impressive, these active-haptic displays are expensive and require sophisticated control mechanisms; miscalculations, mechanical failures, bugs, and latency could be dangerous.

I instead introduce and develop *Redirected Touching*, a perception-based technique that maps any of many differently shaped virtual objects onto a single passive real object (and vice versa) by **mapping real hand motion to different virtual hand motion**. This mapping is accomplished by warping virtual space.

## 4.1 Warping virtual space

*It was six men of Indostan  
To learning much inclined,  
Who went to see the Elephant  
(Though all of them were blind),  
That each by observation  
Might satisfy his mind.*

*The First approached the Elephant,  
And happening to fall  
Against his broad and sturdy side,  
At once began to bawl:  
“God bless me!—but the Elephant  
Is very like a wall!”*

*The Second, feeling of the tusk,  
Cried, “Ho, what have we here,  
So very round and smooth and sharp?  
To me ’tis mighty clear  
This wonder of an Elephant  
Is very like a SPEAR!”*

...  
*And so these men of Indostan  
Disputed loud and long,  
Each in his own opinion  
Exceeding stiff and strong,  
Though each was partly in the right,  
And all were in the wrong!*

**The Blind Men and the Elephant**  
John Godfrey Saxe (1816–1887)

In the above poem, each of six blind men believes the elephant to be something different depending on which part he touched: the side becomes a wall, the tusk a spear, the trunk a snake, the knee a tree, the ear a fan, and the tail a rope. All of the men are disastrously incomplete in their understanding, but being blind they must base their judgments on what they feel.

Suppose a sighted man were wearing a head-mounted display while standing beside the elephant. If the HMD showed him a wall where he touched the elephant's side, would the man perceive a wall? What if the virtual wall were instead a virtual school bus? Would the man then believe he touched a bus? The situation is not unlike that of the blind men; the virtual imagery presented to the sighted man is analogous to the imagery presented to the blind men by their minds.

Research has shown that when senses conflict, vision usually dominates. A subject moving his hand along a straight surface while wearing distorting glasses feels the straight surface as curved [J. J. Gibson, 1933]. Subjects holding an object through a cloth while viewing the same object through a distorting lens believed the object was more similar to the distorted visual image they saw, rather than the shape that they felt [Rock and Victor, 1964]. Lécuyer et al. had participants push with their thumb on a piston mounted on a passive isometric input device. Simultaneously, participants were visually shown a virtual spring that compressed as force was applied to the piston. Even though the piston did not physically move, spring stiffness perception was influenced by the virtual spring [Lécuyer et al., 2000].

Visual dominance is not always complete; during sensory conflict, sensory signals are weighted by their reliability [Helbig and Ernst, 2007]. When mixed-reality users are presented with real haptic and virtual cube-shaped objects with discrepant edge curvatures, they perceive the curvature to be intermediate [Kitahara et al., 2010].

Redirected Touching leverages visual dominance to enable real-virtual object discrepancies to go unnoticed. I now return to the easel example. How might the flat virtual easel map onto the real curved easel? Suppose that a user's hand is already touching the upper-left corner of the real easel. For consistency, the virtual hand must be shown touching the upper-left corner of the virtual easel. Now suppose the user's hand moves along the top edge of the real easel, to the upper-right corner. The hand's real-world path is curved, matching the real easel's curve. The virtual hand cannot move along the same curved path because it would leave the surface of the virtual easel. Instead, the

virtual hand must be shown moving straight along the top edge of the virtual easel. There must be a discrepancy between the real and virtual hand motions.

What about when the user's real hand is *not* touching the surface? What should the virtual hand do in those situations? Because the real easel is convex towards the user, its center is closer to the user than is the virtual easel's center. Assume that the real hand starts 10cm in front of the real easel's center and approaches it (Figure 4.2). The virtual hand starts in the same location, but is 15cm from the virtual easel's center. For the user to feel the virtual easel, the virtual hand must touch the virtual easel at the same time the real hand touches the real easel. When the real hand moves forward 10cm, the virtual hand must move forward 15cm—again a discrepancy between the real and virtual hand motions.

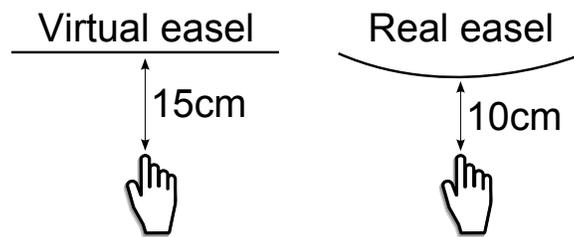


Figure 4.2: A flat virtual easel and a curved real easel, viewed from above. When the real hand moves 10cm to touch the real easel, the virtual hand must move 15cm to touch the virtual easel.

Discrepant stimuli are sometimes introduced in VEs to useful effect. For example, in Redirected Walking (from which my technique borrows its name), a discrepancy is injected between a user's real-world head rotation and the virtual head rotation [Razzaque, 2005]. This discrepancy enables exploration of larger-than-tracked-space VEs (see Chapter 2).

Burns, in his MACBETH technique, employed visual dominance over proprioception to address the problem of missing haptic feedback in VEs [Burns, 2007; Burns et al., 2007]. By clamping the virtual hand's motion to virtual surfaces [Lindeman et al., 2001], Burns made users believe that the virtual hand did not penetrate virtual objects, when in fact the real hand penetrated the space the virtual objects apparently occupied. This real-virtual hand-position discrepancy was removed over time by introducing a velocity distortion along the user's hand motion vector. For passive haptics, the real and virtual hands must reach the real and virtual objects simultaneously. To satisfy this constraint, real-virtual hand discrepancies must be *introduced*. I introduce these discrepancies

by warping virtual space. I call the case where real and virtual objects are the same *one-to-one*; otherwise, *discrepant*.

Ideally, any discrepancies introduced would be undetectable and moreover have no subconscious effect on a user's ability to interact effectively with the VE. If a user were to move her hand through a VE and there were suddenly a 10cm real-virtual hand discrepancy where previously none existed, she would be disconcerted; hands do not teleport. Introduced discrepancies should vary smoothly to be plausible.

Figure 4.3 shows top-down views of a square real tabletop and a trapezoidal virtual tabletop. The space around the real tabletop is rectilinear, whereas the space around the virtual tabletop is warped. This pair of spaces form the mapping between the real and virtual spaces. If a user moves her real hand along the left edge of the real table, her virtual hand moves along the angled left edge of the virtual table. If her real hand moves through real space, her virtual hand moves smoothly according to the warped space.

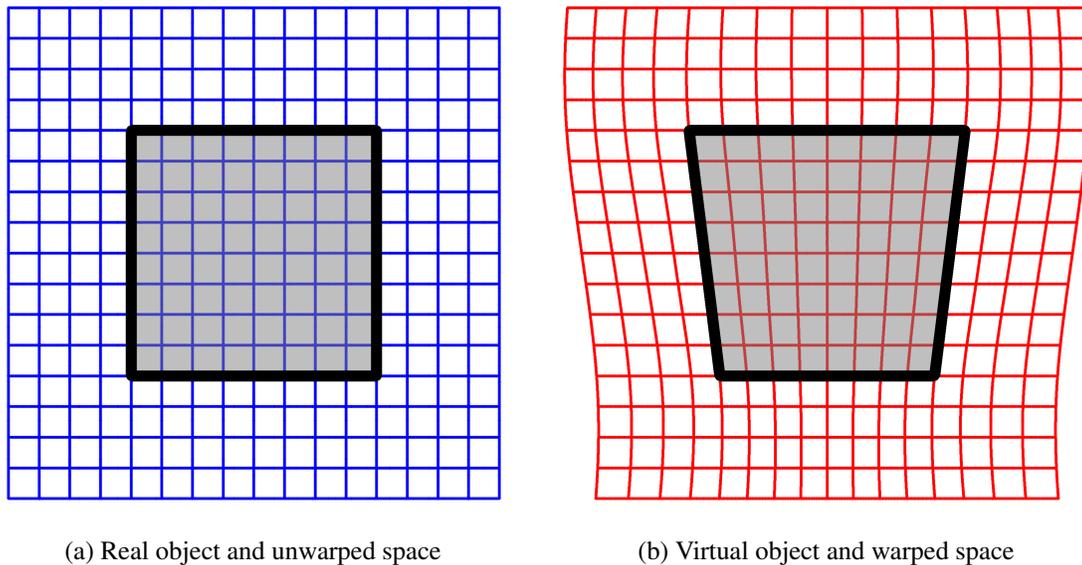


Figure 4.3: Horizontal slices of unwarped (a) and warped (b) spaces and objects (viewed from above)

To properly test Redirected Touching's effectiveness, I needed a method to warp space given a discrepant real-virtual object pair. There were several possible choices, such as free-form deformations [Sederberg and Parry, 1986] and thin-plate splines [Bookstein, 1989]. However, I first wanted to determine whether the idea was worth pursuing at all. I manually constructed warp fields for

several discrepant objects; I next present the results of an informal exploratory pilot study evaluating how those objects felt.

## 4.2 Preliminary exploration



Figure 4.4: A user touches a flat square table

I chose a 24"x24" square-top table as my real object (Figure 4.4). I manually constructed warp fields for several virtual objects:

1. Curved top, with a 4cm-amplitude sine wave,  $[0, \pi]$  (Figure 4.5a)
2. Curved top, with a 4cm-amplitude cosine wave,  $[-\pi, \pi]$  (Figure 4.5b)
3. Sloped top, with 3.5°, 7.5°, 14.7°, and 26.2° slopes (Figure 4.5c)
4. Tapered top, with the short edge 18" long (Figure 4.5d)

These shapes were chosen because they were easily defined by linear or trigonometric functions. The manually constructed warped spaces sometimes contained sharp direction changes as shown in Figure 4.6 (instead of being smooth as in Figure 4.3b). These direction changes were not a problem, because for this exploratory study I was most interested in how the warped space felt while users touched the surface and its edges, rather than while their hand was in the air. Only hand position was warped. Early implementations of the virtual cosine table (Figure 4.5b) oriented the virtual hand such that its vertical axis would coincide with the virtual table surface normal. This orientation warping

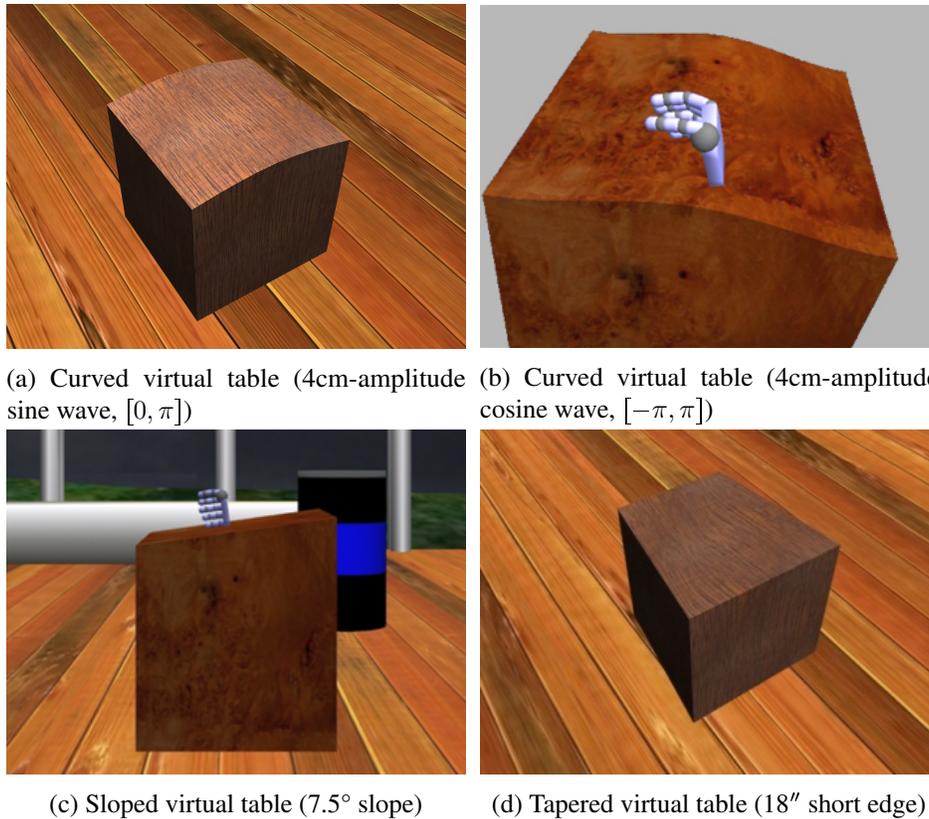


Figure 4.5: Differently shaped virtual objects for which a square-top real table (Figure 4.4) serves as haptic feedback

led to implausible virtual hand orientations for this discrepant virtual object, so it was disabled for this initial exploration. Figures 4.5b–c show the user’s virtual hand.

Six members of the research group experienced these discrepant virtual objects. They viewed the VE in a Virtual Research Systems V8 HMD. The head and hand were tracked by 3rdTech™ HiBall™-3000 wide-area trackers. Users informally explored the virtual objects by moving the bottom of their hand across the surface several times. All reported that the curved sine-wave tabletop (Figure 4.5a) felt curved. One user put both hands on the table simultaneously (one tracked, one not tracked and not shown in the VE) and moved them towards each other. He was surprised when they touched each other, because he expected his right hand to be above his left hand. In other words, seeing the curved table made him believe that his right hand was moving along a curved surface, whereas he felt his left hand moving along a flat surface. According to user reports, the curved cosine-wave tabletop (Figure 4.5b) succeeded in creating the intended illusion for three of the six users. The other three felt the table was flat. I suspected that because the virtual table had inflection

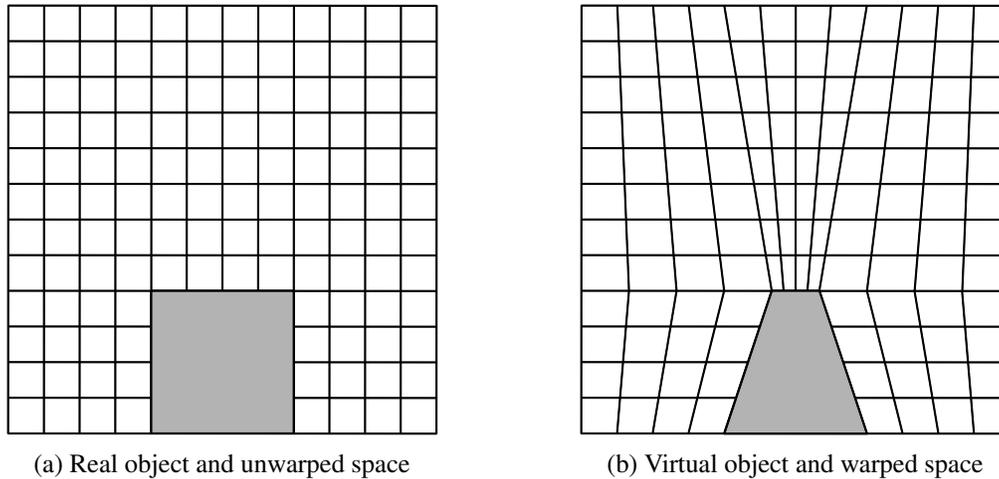


Figure 4.6: Horizontal slices of unwarped space and real square table (left) and warped space and tapered virtual table (Figure 4.5d), viewed from above (not drawn to scale). Note that the warped space is not smooth as in Figure 4.3b.

changes whereas the real table did not, the illusion was less plausible. All users found the 3.5°- and 7.5°-sloped virtual tables compelling, but two participants reported a strange feeling when presented with 14.7° and 26.2° slopes.

Two users were asked to move their hand in the space near and around the sharp direction change shown in Figure 4.6b. The direction changed by  $\sim 9.3^\circ$  just past the end of the table. Neither user reported having detected the direction change despite its abruptness. However, it is likely that larger abrupt direction changes would be noticeable, so a space-warping technique that generated smooth warps would likely be more generally useful.

### 4.3 Implementation changes

At this point, Redirected Touching seemed promising. However, I could not investigate the technique efficiently without some implementation changes.

#### 4.3.1 Tracking

Using the HiBall™ for hand-tracking was restrictive; it was unintuitive to explore surfaces with the bottom of the hand. I wanted to enable users to touch virtual objects with their fingers. I needed a lighter-weight tracking system that could track at least the tip of the index finger. I began using the PhaseSpace® IMPULSE™ tracking system [PhaseSpace, 2012]; it uses lightweight active LEDs that

can be mounted on the finger. Eight PhaseSpace® cameras were mounted on an aluminum frame (Figure 4.7). The index finger was tracked using three LEDs mounted using a self-adhesive athletic bandage (Figure 4.7 inset).



Figure 4.7: PhaseSpace® IMPULSE™ tracking cameras mounted on an aluminum frame. Inset shows tracking LEDs mounted onto an index finger using a self-adhesive athletic bandage.

### 4.3.2 Automating space warping: thin-plate splines

Manually constructing warped spaces took too much time; I needed a more automatic method. I also wanted to generate smooth warps as in Figure 4.3. The surface of the real geometry must be mapped to the surface of the virtual geometry, while smoothly and minimally warping the rest of the space.

I use the well-known thin-plate spline technique common in medical image analysis [Bookstein, 1989]. A thin-plate spline is a 2D interpolation method for passing a smooth and minimally bent surface through a set of points. The concept extends to higher dimensions; I use the 3D version

[Eberly, 2011; Saboo, 2011]. The method finds a function  $f$  that passes through a set of points while minimizing bending energy, or the integral over  $\mathbb{R}^3$  of the sum of squares of second-order partial derivatives of  $f$ . Displacements between corresponding real and virtual geometry points are computed. To generate the warp, the thin-plate spline method uses these displacements as points on which to operate.

Generating warped spaces requires correspondences between real and virtual geometries. Figure 4.8 shows correspondences between real and virtual object corners. Only four correspondences are shown for illustration purposes; typically, more correspondences are required. Additional correspondences can be generated by interpolating across the surface or by various other means. For each of my subsequent experiments, I describe as needed how I computed the correspondences for the particular geometries involved.

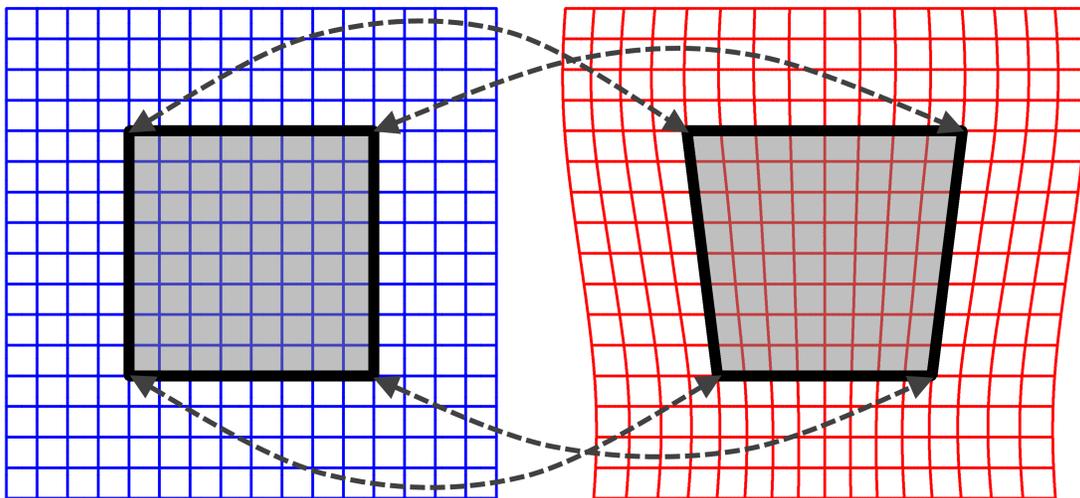


Figure 4.8: Correspondences between real (left) and virtual (right) geometries

## 4.4 Further exploration

With the new tracking system and warping technique in place, I created several more virtual objects to map onto the square real table, some of which are shown in Figure 4.9. To generate real-virtual correspondences, first I used 3D modeling software to create a coarse virtual model of the real table (a box with the real table's dimensions rather than a table with legs). I then deformed the model to create several different shapes, applying curves, twists, and other similar modifiers.

These modifiers only moved vertices; they did not introduce or remove vertices. Consequently, there were already correspondences between the starting geometry (real table) and the geometry of each deformed object. I explored these virtual objects on my own to evaluate what kinds of real-virtual discrepancies were plausible. For this implementation the hand model consisted of only a virtual fingertip (index finger distal phalanx) instead of a whole hand.



Figure 4.9: Eight different deformed virtual objects mapped onto the square real table (Figure 4.4)

This exploration yielded several observations:

1. When I dragged a finger across the surface, a real flat surface appearing as a concave surface was harder to believe than a real flat surface appearing as a convex surface. When I directly touched various points on the surface instead of dragging my finger, both concave and convex surfaces were equally believable.
2. A real flat surface appearing as a surface with a different slope was easy to believe.
3. For the warped spaces tested, I could not tell that my virtual finger was moving in directions different from my real finger when in the air.
4. Any particular warped object became more believable the longer I touched it.

5. Even if a particular space warp is detectable, a warped virtual object may be usable (e.g., if there's some user interface on the surface, the haptic feedback would likely enable precise interaction).
6. In some cases, particularly with the concave surfaces, it felt like my virtual finger was hovering above the surfaces when I touched them, rather than in contact. This feeling was likely caused by the lack of a hand shadow.

The last point about the shadow prompted me to implement hand shadows in later implementations. Because humans give more weight to more reliable sensory signals [Helbig and Ernst, 2007], I expected that visual quality would affect the believability of discrepant objects. In fact, in one implementation in which I could turn shadows on and off, a user reported that she did not believe the discrepant object at all until the virtual hand shadow was turned on.

## **4.5 Quickly set-up passive haptics**

Thus far, I have discussed mapping a variety of virtual objects onto a single real object. It is important to note that warped spaces can be used also to map a variety of real objects onto a single virtual object; all that is required is some real-virtual mapping. This idea may enable VE designers to quickly set up passive haptics in coarse locations to provide haptic feedback, instead of precisely measuring and placing passive haptics. I implemented a proof of concept of this idea.

I used the same system as before and placed a low-cost two-faced 20"x30" foam board on the square table (Figure 4.10). It was quickly taped to the table in an arbitrary position with an arbitrary angle between the two faces. The two board faces turned out to be close to but not quite the same size (widths ~14.25" and ~15.75").

### **4.5.1 Finger tracking and calibration**

Every user's index finger is different and the system must calibrate for these differences. The user's finger is placed flat at a known location on a surface (Figure 4.11). The difference between the tracker data and the known location is computed, and this difference is subsequently used to transform the tracker data so it accurately represents the user's fingertip location.

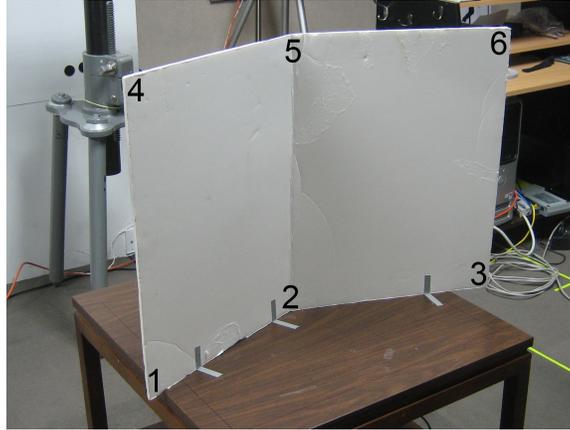


Figure 4.10: A low-cost two-faced passive-haptic foam board

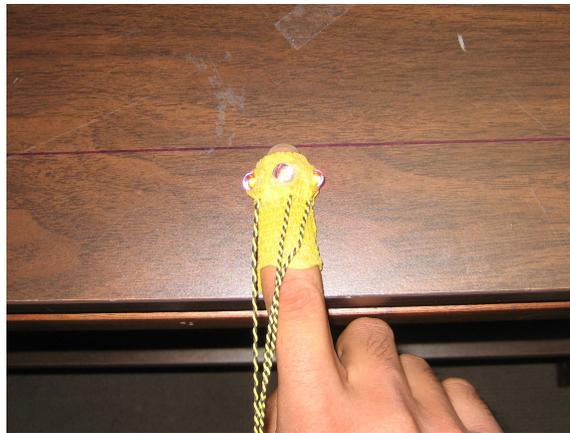


Figure 4.11: Finger calibration

#### 4.5.2 Determination of physical geometry

The system needs to know the (arbitrarily placed) foam board's geometry before a virtual object can be mapped onto it. In the current system, the user points, with a tracked finger, to each corner of the physical object (in the order specified in Figure 4.10). The system linearly interpolates these points to generate vertices for the physical geometry. This technique works for physical objects consisting of planar facets.

The generated vertices are already in the coordinate space of the finger's tracking system, so when the user touches a corner, no additional transformation (beyond the space warp) is required. If there are systematic errors in the tracking system, then sampling the physical surface with more data points should still enable the system to generate a space warp that is valid for the tracking system's coordinate space.

Once the physical geometry is captured, correspondences between points on its surface and predetermined points on the virtual surface are passed to the space warping system.

### 4.5.3 Discrepant objects

I mapped a variety of virtual objects onto the foam board. Three are shown in Figure 4.12. The initial implementation did not warp finger orientation. However, not having orientation warping introduced an unintended tactile discrepancy: the forces felt on the real fingernail did not match what was seen in the VE. For example, when a real board face was angled and a virtual board face was not, the real fingernail would experience a force near the side, rather than near the center as the VE would indicate. I therefore subsequently warped hand orientation as well. The entire process of determining geometry and warping space for a newly placed board took less than one minute.

As in earlier explorations, interacting with these discrepant objects was plausible. The curved virtual board in the upper right of Figure 4.12 felt odd near the real board's crease, but was otherwise effective.

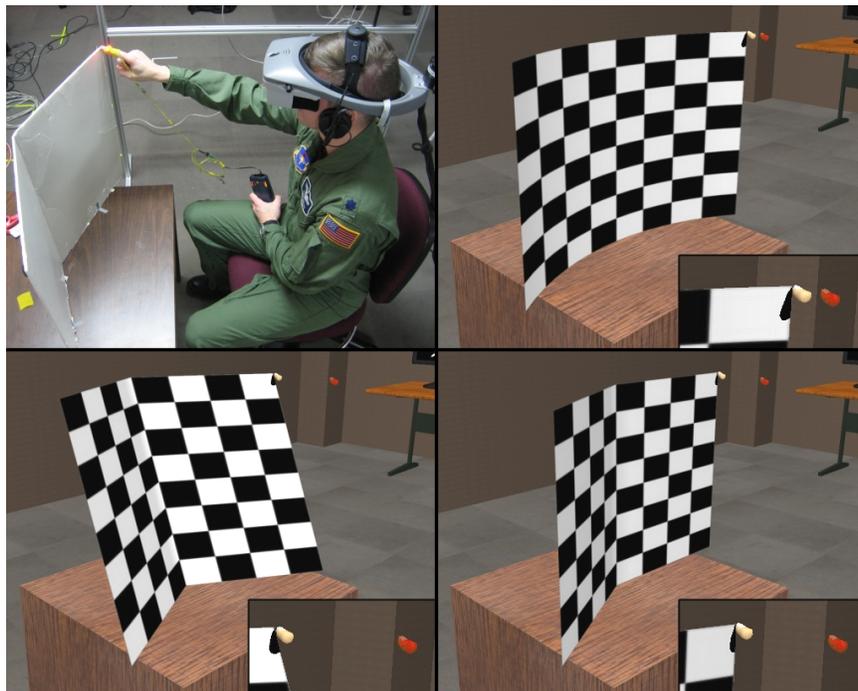


Figure 4.12: A user (wearing an NVIS nVisor SX60 HMD) touches a corner of the physical foam board. The virtual finger position is warped to touch the corners of three different virtual objects. The insets show the distances between the virtual finger and the superimposed real finger (red).

## CHAPTER 5

# Redirected Touching: Task Performance<sup>1</sup>

In Chapter 4, I introduced Redirected Touching, a perception-based technique for mapping many differently shaped virtual objects onto a single real object and vice versa. Redirected Touching warps virtual space and introduces real-virtual hand-motion discrepancies to accommodate real-virtual object discrepancies. While informally exploring many discrepant objects, I quickly discovered that the set of research questions is large and multidimensional. There are many kinds of discrepancies and many ways to warp spaces. Systematic investigation required a guiding application.

Military aircraft pilots and maintenance crews must learn to perform cockpit procedures, such as the sequences of buttons and switches required for aircraft start-up, shut-down, and emergencies. Real aircraft and full flight simulators can be used to train these skills, but can cost hundreds of dollars or more per hour [Vincenzi et al., 2008]. Full simulators are unavailable in deployed settings [Andre et al., 2004], and for many procedures they are not required; low-fidelity trainers with mock cockpits can effectively train cockpit procedures [Prophet and Boyd, 1970].



Figure 5.1: Lockheed Martin® C-130 Hercules® cockpit

---

<sup>1</sup>Portions of this chapter were previously published elsewhere [Kohli et al., 2012].

Cockpits are complex (Figure 5.1), and one of the challenges trainers face is training “muscle memory” and spatial knowledge [Degani and Wiener, 1993; Baseops.Net, 2012]. Pilots must use checklists for many procedures, but they must know the relative locations of cockpit controls so that they can be accessed efficiently when needed. Mock cockpits enable trainees to reach for cockpit controls. However, they are designed for a specific aircraft. For a new aircraft model, another mockup must be built, or the existing one must be changed [Wurpts, 2009]. Reconfigurable trainers do exist, but they typically support reconfiguration by repositioning or swapping out components [Schiefele et al., 1998; Williams et al., 2004; Lockheed Martin, 2012]. Instead, Redirected Touching may enable a single quickly set-up physical mockup to represent many virtual cockpits, eliminating the need to change the mockup for each new aircraft.

## 5.1 Research questions

The informal evaluations of introduced discrepancy via space-warping in Chapter 4 led me to two broad research questions:

1. **Is discrepancy detectable?**

What kinds and amounts of introduced discrepancy would go unnoticed by users?

2. **Does discrepancy hurt task performance?**

Can users perform tasks with discrepant objects as well as they can with one-to-one objects?

How does performance change as discrepancy increases?

Guided by the cockpit procedures application, I briefly investigated detectability, but I focused my work on task performance. In this chapter and the next, I present the results of two studies investigating the effects of warping space on task performance and training. I specifically investigated the following question:

**Can users use a virtual button interface presented on a discrepant object as well as they can an interface on a one-to-one object?**

There can be many discrepancies between real and virtual objects. To make the scope of this research manageable, I investigated angular discrepancy: the virtual object is rotated about the vertical axis relative to the real object. I used a real 10" x 10" foam board and a virtual 10" x 10"

interface panel. Because cockpit panels tend to be made of planar facets, angular discrepancy was a good place to start.

Answering the research question requires (1) a system to present real and virtual objects, (2) a method of warping virtual space, (3) a task-performance metric, and (4) a way to determine whether task performance with a discrepant object is no worse than with a one-to-one object. I describe these in order.

## 5.2 System



Figure 5.2: System constructed for space-warping studies. Inset shows the finger tracking ring.

A system similar to my earlier Redirected Touching implementations was constructed (Figure 5.2). It tracks a user's fingertip, presents real and virtual objects at arbitrary angles, maps the virtual object onto the real object, and detects when the user has touched the real object.

The user's right index finger is tracked using a PhaseSpace® IMPULSE™ optical motion capture system. The self-adhesive athletic bandage used to mount LEDs on the index finger from previous implementations was difficult to use efficiently. I therefore made a flexible ring out of sugru® [sugru, 2012] and mounted three PhaseSpace® LEDs on it (Figure 5.2, inset).

The system must calibrate for each user's index finger. The finger is placed flat at a known location on a table surface. The difference between the tracker data and the known location is used to determine the user's fingertip location.

The 10"x10" foam board also is tracked using three PhaseSpace® LEDs, and is mounted on a motorized Directed Perception® pan-tilt unit. The pan-tilt unit is rotated to present users with the real board at different angles. The VE is displayed on an NVIS nVisor SX60 HMD using Gamebase Co. Ltd's Gamebryo™ game engine running on a dual quad-core 2.3GHz Intel® Xeon® machine with 8GB of RAM and an NVIDIA® GeForce® GTX 280 GPU. The rendered output is corrected for the HMD's pincushion distortion using a technique based on the HMD calibration work by Kuhl et al. [2008]. Their pincushion correction shader leaves dark borders in the corners of the display; to use the full set of display pixels, I send a larger-than-display-resolution texture to the pincushion correction shader so that pixel data are available for the corners. A 3rdTech™ HiBall™-3000 tracking system is used for head tracking, and VRPN is used for tracker communication [Taylor et al., 2001]. End-to-end system latency is ~50-60ms.

A contact microphone is mounted on the back of the foam board. A volume spike indicates when the board is touched, and tracker readings indicate where the board is touched.

## **5.3 Warping space**

Space is warped using the 3D thin-plate spline technique described in section 4.3.2.

### **5.3.1 Choosing correspondences for space-warping**

From the LEDs mounted on the real board, the board's corners are determined and interpolated to find points on the board surface. These interpolated points are generated in the same order as the vertices for the virtual board, yielding correspondences between the real and virtual geometries. This

technique works well for the simple planar shapes in this study, but more complex shapes would require something more sophisticated.

### 5.3.2 Hand orientation

As in section 4.5.3, I warped virtual hand orientation because otherwise forces felt on the fingernail did not match what users expected from the visual orientation of the virtual hand. At the board's surface, the relative orientation between the virtual board and hand are made the same as the relative orientation between the real board and hand. The orientation discrepancy changes smoothly and non-uniformly as the user moves in the warped field.

## 5.4 Measuring task performance

In 1954, Fitts introduced a predictive model that relates movement time to distance and accuracy in rapid aimed movements, such as pointing [Fitts, 1954]. It is now known as *Fitts' law*, and is defined as:

$$MT = a + b \cdot ID, \text{ where } ID = \log_2\left(\frac{A}{W} + 1\right) \quad (5.1)$$

$MT$  is *movement time* in seconds,  $a$  and  $b$  are empirically determined constants, and  $ID$  is known as the Shannon formulation of the *index of difficulty*, measured in bits [MacKenzie, 1992].  $A$  is *amplitude* or *target distance*, the distance between successive targets.  $W$  is *target width*. Increasing  $A$  or decreasing  $W$  makes an aiming task more difficult.

Fitts also introduced a composite measure of speed and accuracy called *index of performance*, now known as *throughput* ( $TP$ ). Researchers commonly use throughput to compare and evaluate different pointing devices. Throughput combines speed and accuracy and, usefully, it is unaffected by whether users decide to focus on one or the other [MacKenzie and Isokoski, 2008].

The ISO 9241-9 document, *Ergonomic requirements for office work with visual display terminals (VDTs)* [International Organization for Standardization, 2007], describes a set of standardized tasks and measures for computing throughput.<sup>2</sup> To improve consistency and to enable comparisons across

---

<sup>2</sup>There is an ongoing debate in the HCI community about how best to compute throughput, and alternative methods have been proposed [Soukoreff and MacKenzie, 2004; Zhai, 2004; Drewes, 2010]. Of the many possible methods, I chose to follow the recommendations of Soukoreff and MacKenzie [2004].

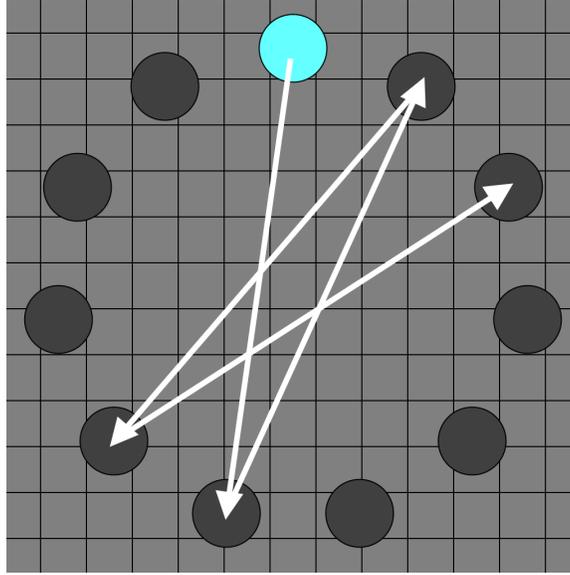


Figure 5.3: The multi-directional tapping task. Overlaid arrows show the target sequence.

studies, researchers have recently used ISO 9241-9 tasks for evaluating VE user interfaces [Teather et al., 2010; Teather and Stuerzlinger, 2011]. I also chose to use the ISO 9241-9 multi-directional tapping task, with eleven targets arranged in a ring (Figure 5.3). Users tap each target in a specific sequence (starting with the top target), and speed and accuracy are measured. Throughput is measured in bits per second (bps) and is defined as:

$$TP = \frac{ID_e}{MT}, \text{ where } ID_e = \log_2\left(\frac{A_e}{W_e} + 1\right) \quad (5.2)$$

$ID_e$  is the *effective index of difficulty*. Users are unlikely to perform the given task exactly; sometimes users will tap a large target very precisely as though it were small, or vice versa.  $ID_e$  models the task that users actually perform. It has the same form as  $ID$  in equation (5.1), but uses *effective amplitude* ( $A_e$ ) and *effective target width* ( $W_e$ ).  $A_e$  is the user's mean movement distance between targets for trials of a single target ring.  $W_e$  is:

$$W_e = 4.133 \cdot SD_x \quad (5.3)$$

A single trial consists of a user leaving one target and touching another (specific) target at a point I call an *endpoint*. To compute  $W_e$ , I first transform all trials to horizontal ( $0^\circ$ ), which also

rotates all the endpoints so that they appear on or near a single target. I then project each of the endpoints onto the horizontal axis, giving each endpoint a value along that axis.  $SD_x$  is the standard deviation of those values. This calculation is known as the *adjustment for accuracy*, and accounts for the spread of touch endpoints for the task the user actually performed [MacKenzie, 1992; Soukoreff and MacKenzie, 2004].

## 5.5 Non-inferiority testing

In Fitts'-law-based studies, it is common to look for statistically significant differences between different user interface techniques. However, even if there is a statistically significant difference between conditions, the magnitude of that difference could be very small in practical terms. What we really want to know is whether task performance while using a discrepant interface is *good enough* for a given application. We ideally want to show that task performance while using a discrepant interface is *no worse than* while using a one-to-one interface.

It is possible to test whether one condition is no worse than another by statistically testing for non-inferiority, a form of equivalence testing [Wellek, 2010]. To do so, one must define an indifference zone: the maximum difference between a discrepant condition and a one-to-one condition to be considered non-inferior in the context of the application. Once there is an indifference zone, the mean difference between conditions and the one-tailed 95% confidence interval of that mean difference are calculated. If the mean difference and the entire confidence interval lie within the indifference zone, the discrepant condition is considered to be no worse than the one-to-one condition.

In this study, the task-performance dependent variables were throughput, error rate, and movement time.

### 5.5.1 Indifference zone: throughput

I chose 1bps as the maximum allowable throughput difference between discrepant and one-to-one conditions. This value was chosen because a 2004 survey of ISO 9241-9 studies found that the range of throughputs for computer-mouse pointing in five studies was 3.7-4.9bps, a range of 1.2bps

[Soukoreff and MacKenzie, 2004]. Moreover, in Fitts' 1954 paper, he labeled 10-12bps, a range of 2bps, as consistent [Fitts, 1954]. I conservatively chose a tighter indifference zone of 1bps.<sup>3</sup>

### 5.5.2 Indifference zone: error rate

To be conservative, I chose 9% as the maximum allowable error-rate difference between discrepant and one-to-one conditions. 9% corresponds to my smallest unit of measurement for error rate: one out of eleven targets missed.

### 5.5.3 Movement time

It was not clear what indifference zone to choose for movement time. I therefore analyzed movement time to look for significant differences between pairs of conditions.

## 5.6 Study

### 5.6.1 Participants

Twenty-two paid participants (eleven male, eleven female, aged 18-28, mean 21) took part. There was a mix of majors of both undergraduate and graduate students, and a mix of video game experiences. All participants were right-handed and had normal or corrected-to-normal vision.

### 5.6.2 Notation

To determine values for which using discrepant objects is no worse than using one-to-one objects, I tested a range of discrepancies. A particular condition is represented as a tuple of the form (real angle, virtual angle), in degrees. Counter-clockwise rotations about the vertical axis are positive.

For each discrepant condition, there was an analogous one-to-one condition. Discrepant conditions always had a real board angle of  $0^\circ$ . For example, a one-to-one condition with both the real and virtual angles at  $-18^\circ$  is  $(-18, -18)$ , and its discrepant counterpart is  $(0, -18)$ . As in other recent research [Teather and Stuerzlinger, 2011], a trial is a single target touch. A target ring is the set

---

<sup>3</sup>The throughput formula used in Fitts' paper is different from that specified in the ISO 9241-9 standard. However, a reanalysis of Fitts' data [MacKenzie, 1992] using the Shannon formulation of ID and adjustment for accuracy suggests that the throughput range would not differ much from Fitts' range, even though the absolute values change.

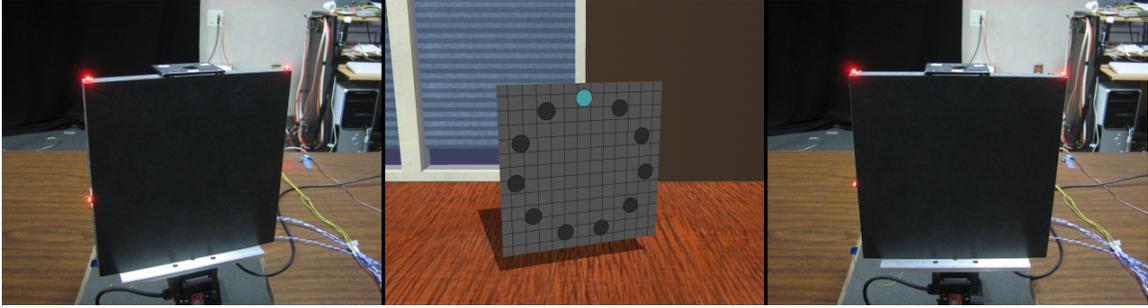


Figure 5.4: Left: front view of real board at  $18^\circ$ ; Middle: virtual board at  $18^\circ$ ; Right: real board at  $0^\circ$ . Left and middle panels make up a one-to-one condition, and right and middle panels a discrepant condition

of all eleven targets in a particular instance of the multi-directional tapping task. Figure 5.4 shows examples of one-to-one and discrepant conditions.

### 5.6.3 Angle range

In a pilot study, participants performed the multi-directional tapping task at various angles in the range  $-24^\circ$  to  $24^\circ$ ;  $-24^\circ$  and  $24^\circ$  were too discrepant to be used effectively (in terms of throughput) by pilot participants. For the full experiment, I discretized the range into six angles:  $-18^\circ$ ,  $-12^\circ$ ,  $0^\circ$ ,  $12^\circ$ ,  $18^\circ$ ,  $24^\circ$  ( $24^\circ$  was kept to verify pilot study results). These angles yielded eleven conditions:  $(0, -18)$ ,  $(-18, -18)$ ,  $(0, -12)$ ,  $(-12, -12)$ ,  $(0, 0)$ ,  $(0, 12)$ ,  $(12, 12)$ ,  $(0, 18)$ ,  $(18, 18)$ ,  $(0, 24)$ , and  $(24, 24)$ .

### 5.6.4 Adaptation

During the pilot study, I found that after each change of conditions, throughput values went down for one or two target rings, but then went higher and became more consistent. In the full experiment, I excluded from analysis two adaptation target rings at the beginning of each condition block. Adaptation is discussed in more detail in Section 5.9.

### 5.6.5 Target distances and widths

In Fitts'-law tasks, it is typical to include many different target distances and widths to explore a wide index-of-difficulty range. Data are aggregated across those indices of difficulty to yield overall task performance for a particular experimental condition (e.g., a particular user interface being tested)

[Soukoreff and MacKenzie, 2004]. Guiard [2009] and Wobbrock et al. [2011] suggest that the same results can be obtained by using a single target distance and varying only target width. In the angle discrepancy scenario, targets near the center of the board do not present much discrepancy. I therefore used a single large target distance, 21cm, to minimize the number of conditions and to maximize the discrepancy's effect.<sup>4</sup>

Target widths were chosen based on the US Department of Defense design criteria standard for human engineering MIL-STD-1472F [Department of Defense, 1999], which specifies criteria for design and development of military systems. The standard specifies that push buttons (e.g., in cockpits) have a diameter between 10mm and 25mm for bare fingertips, and at least 19mm for gloved fingertips. I chose six target widths: 10mm, 15mm, 20mm, 25mm, 30mm, and 35mm. The last two widths, despite being larger than described in the standard, were included to make the set of conditions more varied.

The chosen target distance and widths resulted in six distinct indices of difficulty ranging from ~2.8 to ~4.5 bits. Details about the target rings are in Appendix B.

### **5.6.6 Trials and counterbalancing**

All participants did all conditions. The study used a within-subjects design, and conditions were counterbalanced using a Williams design [Williams, 1949; Wang et al., 2009]. Each target ring had eleven targets. Because participants did not start with their hands in a standard position, timing information is not available for the first target. Thus, there were ten trials per target ring. Conditions were presented in eleven blocks of eight target rings. The first two target rings in each block were used for adaptation. The target widths on the last six target rings were randomized from the set of six target widths, without replacement. Including all participants, a total of 14,520 trials were recorded.

### **5.6.7 Procedure**

Participants sat on a chair in front of the real board (Figure 5.5). To minimize arm fatigue during the experimental task, the chair was adjusted such that each participant's shoulders were higher than the top of the board. I explained the multi-directional tapping task by showing a printout of a

---

<sup>4</sup>The actual target distance was ~20.8cm. See Appendix B for details.

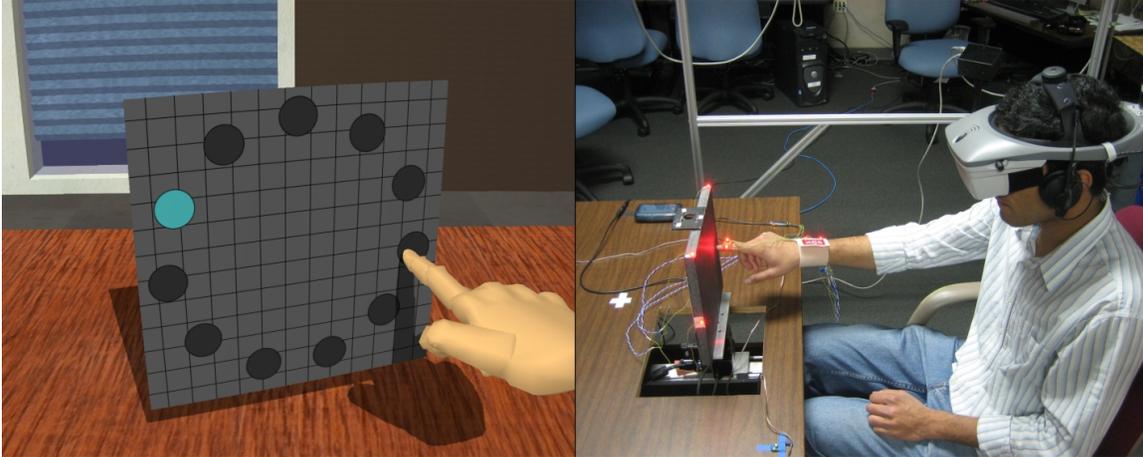


Figure 5.5: A user touches a virtual board that is oriented differently than the real board

target ring. Participants were instructed to touch each highlighted target as quickly and as accurately as possible, and to move on to the next target even if targets were missed (instructions typical of Fitts'-law studies). A small white dot was shown where the user touched the board, and a red X was shown on missed targets. Participants were told that things might occasionally feel strange (referring indirectly to the angular discrepancy), and that they should do the best they could in those situations.

Participants were fitted with the finger tracking equipment and HMD and completed seven training target rings to get used to the equipment and task. The training target rings consisted of some one-to-one conditions and some discrepant conditions, and were the same for all participants. After asking any questions they had, participants then proceeded with the eleven condition blocks of target rings (no connection to the eleven targets in a ring). They were given a break of at least one minute after the fourth and eighth blocks. After each target ring, the real and virtual boards rotated to the proper angles for the next target ring. In the VE, the virtual board immediately switched to the new angle while the real board rotated. The real board rotated regardless of whether the real angle of the next target ring was different. Because participants might determine how far the board rotated based on the pan-tilt unit's sound, all rotations were made to sound alike. Each was performed in 14 steps, each either  $+3^\circ$  or  $-3^\circ$ . For example, rotating from  $12^\circ$  to  $18^\circ$  involved two  $+3^\circ$  rotations, followed by twelve alternating  $+3^\circ$  and  $-3^\circ$  rotations.

After the VE session, participants completed a short questionnaire (Appendix A.2), and I conducted a semi-structured open-ended interview to elicit comments. Participants were then informed more fully about the study, and paid.

## 5.7 Results

### 5.7.1 Spatial outliers

In Fitts'-law studies, it is customary to remove spatial outliers from the target touch data. In the literature, spatial outliers have been defined as errors in which movement was less than half the nominal target distance  $A$ , or in which the touch endpoint was more than twice the target width  $W$  from the target center [Wobbrock et al., 2011]. There were no outliers of the first kind in the data.

I chose to not remove spatial outliers of the second kind. In cases where participants had not yet adapted to a condition, or where the angular discrepancy was large, touches would often land more than twice the target width from the target center. However, from direct observation, these misses were an indication that participants were unable to reach a target when desired. Removing these data points would artificially inflate throughput values in cases where a miss was due to a true difficulty with using the interface.

The microphone occasionally did not properly detect that the real board had been touched, leading to participants tapping the board twice or more times near a target until their tap had been detected. These multi-taps were visible in the finger tracker data; tracker data were trimmed such that motion paths between target pairs went until only the first tap, and the touch times and locations were adjusted accordingly. In a few cases, participants forgot to touch the final target of the board and started resting before finishing a particular target ring. These paths were removed entirely from analysis.

### 5.7.2 Learning effect and analysis

Upon visual inspection of the aggregate throughput data, I discovered that there was a substantial learning effect that lasted considerably longer than anticipated based on pilot data (Figure 5.6). When viewed in isolation, this learning effect is more pronounced in discrepant conditions than in one-to-one conditions, but the effect is generally present in both. Consequently, for analysis purposes I grouped the target ring blocks into the first three blocks and the last eight blocks. An effect for this grouping was then included in the statistical model. I compared discrepant vs. one-to-one conditions, adjusting standard errors and hypothesis tests for multiple observations within subjects.

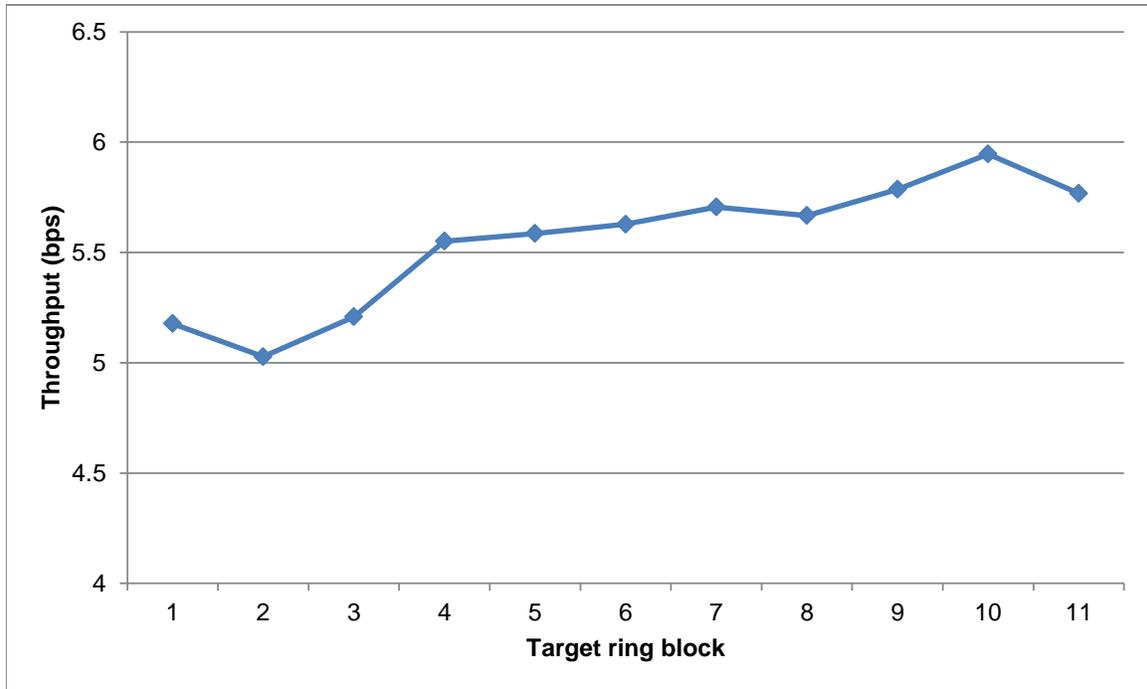


Figure 5.6: A learning effect is seen in the throughput data, most evident between the first three and last eight blocks of target rings

### 5.7.3 Throughput

Mean throughputs are shown in Figure 5.7. The differences between the mean throughputs of discrepant/one-to-one condition pairs and associated one-tailed 95% confidence intervals are shown in Table 5.1. Given an indifference zone of 1bps, the mean difference and the lower bound of the confidence intervals should be no less than  $-1$ bps for a discrepant condition to be considered no worse than a one-to-one condition.

All discrepant conditions except for  $(0, -18)$  were found to be no worse than their associated one-to-one conditions. The  $(0, -18)$  condition's confidence interval was just outside the indifference zone.

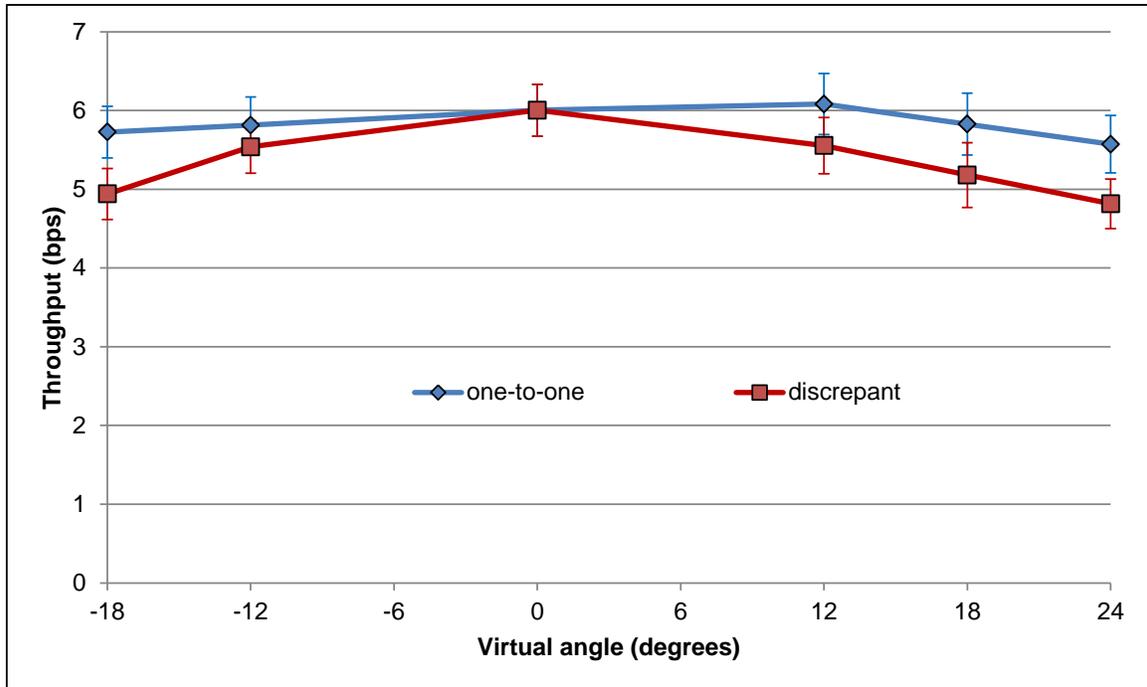


Figure 5.7: Throughput by virtual angle. Error bars are 95% confidence intervals.

Table 5.1: Differences between discrepant and one-to-one throughputs (bps)

Condition pair	Mean difference	One-tailed 95% conf. interval	Non-inferiority comparison
<b>(0, -18) - (-18, -18)</b>	<b>-0.79</b>	<b>&gt; -1.01</b>	<b>-1.01 &lt; -1.0</b>
(0, -12) - (-12, -12)	-0.28	> -0.50	-0.50 > -1.0
(0, 12) - (12, 12)	-0.53	> -0.77	-0.77 > -1.0
(0, 18) - (18, 18)	-0.65	> -0.95	-0.95 > -1.0
(0, 24) - (24, 24)	-0.76	> -0.99	-0.99 > -1.0

#### 5.7.4 Error rate

Mean error rates are shown in Figure 5.8. The differences between the mean error rates of discrepant/one-to-one condition pairs and associated one-tailed 95% confidence intervals are shown in Table 5.2. Given an indifference zone of 9%, discrepant conditions must have an error rate no more than 9% (1/11 targets) higher than associated one-to-one conditions to be considered no worse.

All discrepant conditions except for (0, 24) were found to be no worse than their associated one-to-one conditions.

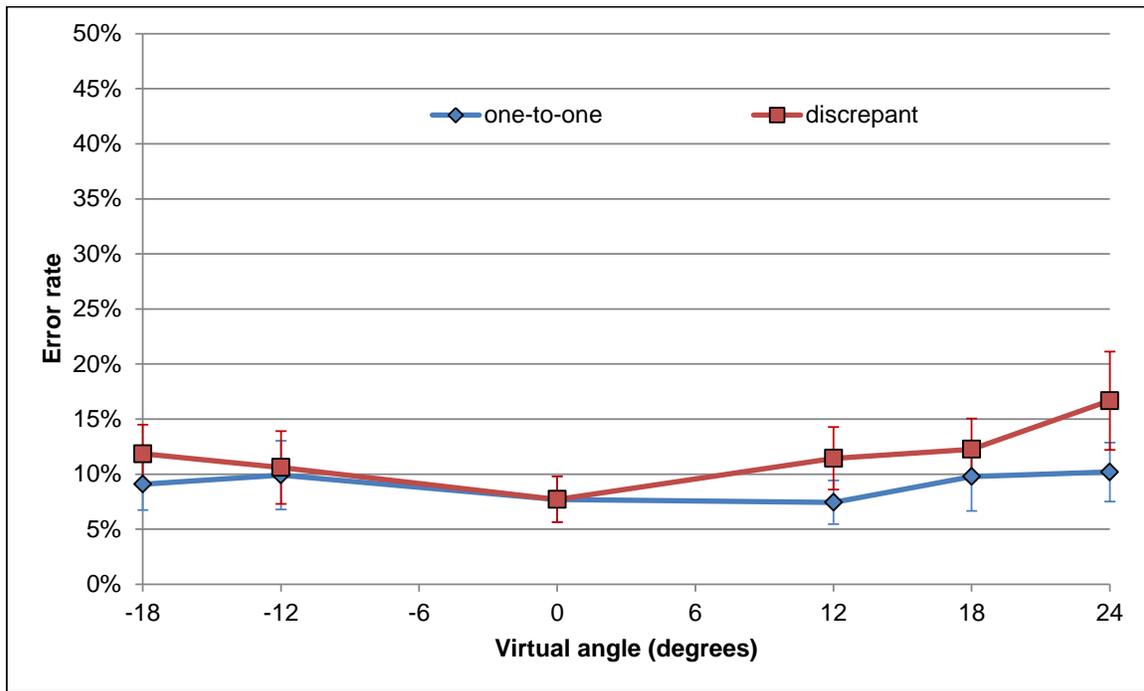


Figure 5.8: Error rate by virtual angle. Error bars are 95% confidence intervals.

Table 5.2: Differences between discrepant and one-to-one error rates

Condition pair	Mean difference	One-tailed 95% conf. interval	Non-inferiority comparison
(0, -18) - (-18, -18)	2.8%	< 5.1%	5.1% < 9%
(0, -12) - (-12, -12)	-0.7%	< 3.2%	3.2% < 9%
(0, 12) - (12, 12)	4.0%	< 6.3%	6.3% < 9%
(0, 18) - (18, 18)	2.5%	< 4.4%	4.4% < 9%
<b>(0, 24) - (24, 24)</b>	<b>6.5%</b>	<b>&lt; 9.2%</b>	<b>9.2% &gt; 9%</b>

### 5.7.5 Movement time

Mean movement times between targets are shown in Figure 5.9. Because I did not know of a reasonable value to use, I did not choose an indifference zone for movement time. I instead analyzed the data for significant differences using the Wald test. Within the context of the full model,

several planned contrasts were done. A significant difference between movement times was found for all pairs of discrepant/one-to-one conditions (Table 5.3). Movement times are represented in milliseconds in the figure and table, but in seconds for throughput computation.

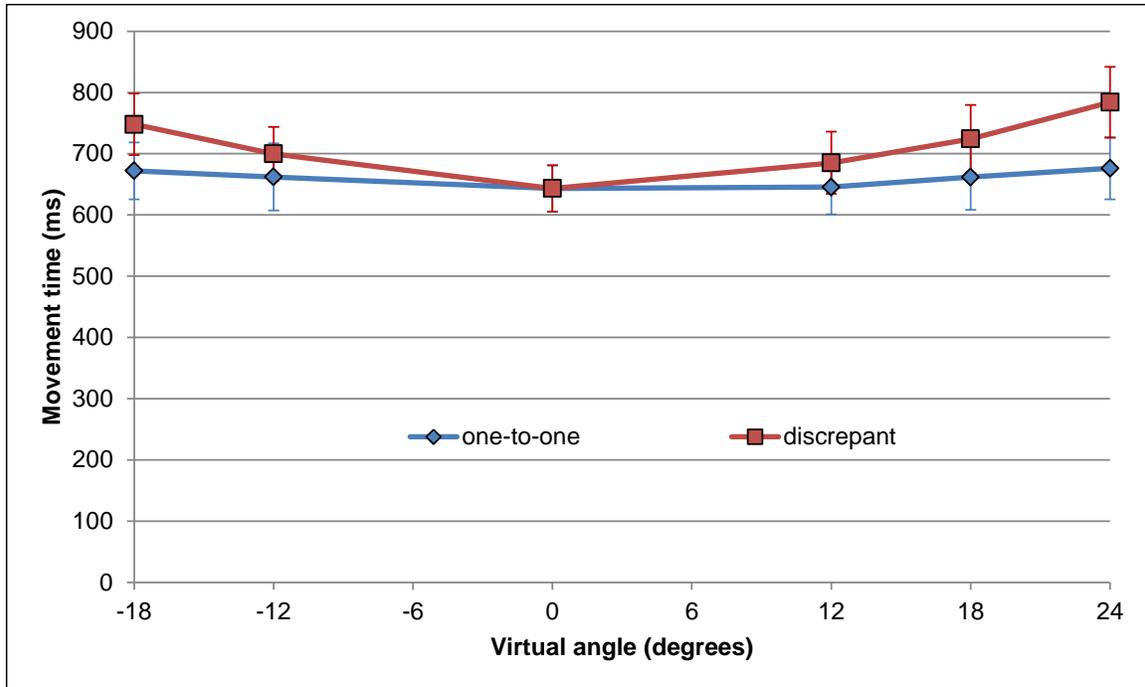


Figure 5.9: Movement time by virtual angle. Error bars are 95% confidence intervals.

Table 5.3: Differences between discrepant and one-to-one movement times (ms)

Condition pair	Mean difference	95% conf. interval	Significance
(0, -18) - (-18, -18)	76.0	(45.0, 106.9)	$\chi_1^2 = 23.15$ , $p < 0.0001$
(0, -12) - (-12, -12)	37.7	(7.4, 68.0)	$\chi_1^2 = 5.94$ , $p = 0.0148$
(0, 12) - (12, 12)	39.6	(8.6, 70.5)	$\chi_1^2 = 6.28$ , $p = 0.0122$
(0, 18) - (18, 18)	62.5	(27.6, 97.3)	$\chi_1^2 = 12.35$ , $p = 0.0004$
(0, 24) - (24, 24)	107.9	(69.4, 146.3)	$\chi_1^2 = 30.23$ , $p < 0.0001$

## 5.8 Discussion

The results above indicate that with non-inferiority indifference zones of 1bps and 9% for throughput and error rate respectively, the discrepant conditions (0, -12), (0, 12), and (0, 18) are no worse than their one-to-one counterparts, but significant differences in movement time were found between discrepant/one-to-one pairs.

### 5.8.1 Caveats

From directly observing participants, I would hesitate to say (0, 24) is no worse than (24, 24). While in the (0, 24) condition many participants said they had difficulty using the interface. Can I be sure that 1bps is a valid indifference zone?

Truthfully, I cannot be sure. Even though ranges of about 1–2bps have been labeled consistent in the past, many studies in the literature show significant differences in throughput between conditions with mean throughput differences less than 1bps. Despite showing non-inferiority under the chosen indifference zone, the graphs of mean throughput show a clear trend of performance degradation as discrepancy increases. Throughput, while being useful as a metric that combines speed and accuracy, is not nearly as concretely understandable as either metric alone. It is important to evaluate speed and accuracy separately along with the combined throughput in the context of the application.

Assume for the moment that 1bps is too generous a value for the throughput indifference zone. Now consider error rate and movement time. In all condition pairs other than (0, 24)–(24, 24), the highest confidence interval bound for error rate difference is ~6%. The highest confidence interval bound for movement time difference is ~100ms (again excluding (0, 24)–(24, 24)).

Certainly, only the clients of a VR system that employs space-warping can decide what differences are acceptable.<sup>5</sup> However, these small differences in error rate and movement time suggest that participants were able to perform the given task *almost* as well in discrepant conditions as in one-to-one conditions. In other words, users may not be able to touch virtual buttons as precisely in discrepant conditions as in one-to-one conditions, but I believe they can touch them precisely *enough* to activate them when desired, and that may be sufficient for many tasks.

---

<sup>5</sup>If other indifference zones are deemed more valid than the ones chosen for this study, they can be compared with the confidence intervals from this study to evaluate non-inferiority.

## 5.8.2 Computing throughput with 2D endpoint deviation

The effective target width computation defined in Section 5.4 projects touch endpoints onto a line. The adjustment for accuracy thus takes into account only one dimension of trial endpoint deviation. Because the ISO 9241-9 multi-directional tapping task is a 2D task, Wobbrock et al. suggest that it may be more appropriate to compute trial endpoint deviation in two dimensions to account for deviations orthogonal to the task axis [Wobbrock et al., 2011]. They present a formula for this 2D deviation:

$$SD_{x,y} = \sqrt{\frac{\sum_{i=1}^N (\sqrt{(x_i - \bar{x})^2 + (y_i - \bar{y})^2})^2}{N - 1}} \quad (5.4)$$

$(x_i, y_i)$  are the endpoints, and  $(\bar{x}, \bar{y})$  is the centroid of the endpoints. Wobbrock et al. are careful to note that  $SD_{x,y}$  does not represent a spread of distances from the centroid, but a spread of endpoints around a centroid. As described in Section 5.4, I first transformed all trials to horizontal, and then reanalyzed the data using  $SD_{x,y}$  instead of  $SD_x$ . Consistent with Wobbrock et al., throughput values were systematically lower than with the 1D computation (Figure 5.10). This is expected, because  $SD_{x,y}$  incorporates additional deviation that is not present in the 1D computation.

Mean throughput differences and confidence intervals are shown in Table 5.4. Using the same throughput indifference zone as before, 1bps, all discrepant conditions except for (0, 24) were no worse than their associated one-to-one conditions. These results are more consistent with my observations that (0, 24) is likely to be worse than (24, 24). Throughput computed using  $SD_{x,y}$  may be more appropriate for data from the 2D task.

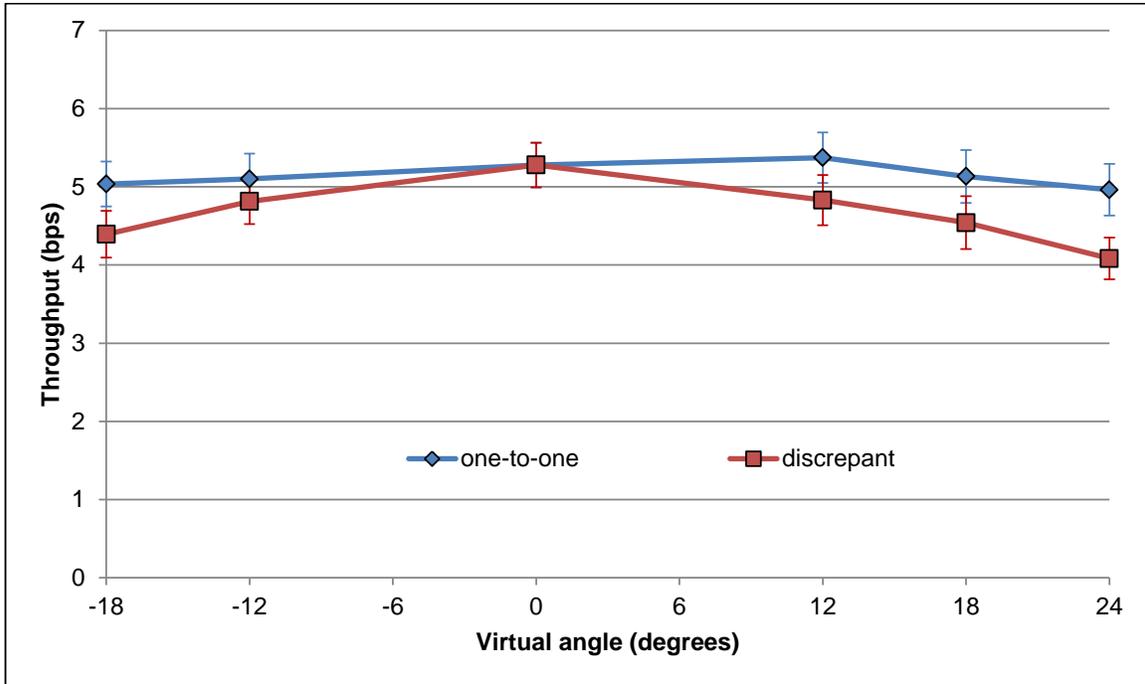


Figure 5.10: Throughput computed using 2D endpoint deviation. Error bars are 95% confidence intervals.

Table 5.4: Differences between discrepant and one-to-one throughputs (bps) computed using 2D endpoint deviation

Condition pair	Mean difference	One-tailed 95% conf. interval	Non-inferiority comparison
(0, -18) - (-18, -18)	-0.64	> -0.84	-0.84 > -1.0
(0, -12) - (-12, -12)	-0.29	> -0.48	-0.48 > -1.0
(0, 12) - (12, 12)	-0.54	> -0.75	-0.75 > -1.0
(0, 18) - (18, 18)	-0.59	> -0.86	-0.86 > -1.0
<b>(0, 24) - (24, 24)</b>	<b>-0.88</b>	<b>&gt; -1.1</b>	<b>-1.1 &lt; -1.0</b>

### 5.8.3 Participant comments and other observations

After the training part of each experiment session, no participants asked any questions. One participant did show signs that he discovered the nature of the discrepancy during the first experiment break. He asked whether the real board always rotated or if it sometimes did not. Indeed, after the

experiment, that participant stated that the real and virtual boards did not always seem to be at the same orientation.

During the post-experiment interview, most participants had similar comments. All 22 participants said that they experienced a strange feeling at times (the discrepant conditions). Five participants identified that the real and virtual boards were not always at the same orientation. The most commonly mentioned strangeness cue (17 participants) was that participants had to reach farther than (or not as far as) they expected from the virtual board they saw—a visual-proprioceptive discrepancy. Another cue (10 participants) was that when participants would move their finger from one side of the board to the other, the finger would often hit the board prematurely due to the discrepancy. Participants reacted by pulling their hand back a little farther before touching the next target. Most participants remained directly in front of the board during the task. One participant often reoriented his head to face the virtual board head-on, and another occasionally rotated his chair. I did not notice any difference in performance due to these strategies.

Participants rated arm fatigue from 1 (not tired at all) to 5 (very tired). Three participants rated arm fatigue as 3, and the rest rated it 1 or 2. The most common complaint, from about half of the participants, was that the HMD was too heavy or uncomfortable. Some participants said their neck was tired from looking down at the board. One participant felt disoriented after the first block of target rings. The HMD was removed for a short time during the first break, and there were no difficulties after that. No participants mentioned feeling nauseated or dizzy.

## **5.9 Revisiting adaptation**

I noted in 5.6.4 that participants generally had lower task performance immediately after being presented with a new condition and beginning the trials in that condition. With increasing exposure to the new condition, participants' task performance improved. I attributed this improvement to adaptation to the newly presented orientation discrepancy (or lack thereof). However, aside from the task performance changes, evidence of adaptation was primarily based on participant feedback during the pilot study. I suspected that if adaptation did indeed occur, it would be evident in participants' finger motion. I investigated finger motion more closely to corroborate other evidence of adaptation, because adaptation may have implications for training applications. In particular, if a user is trained

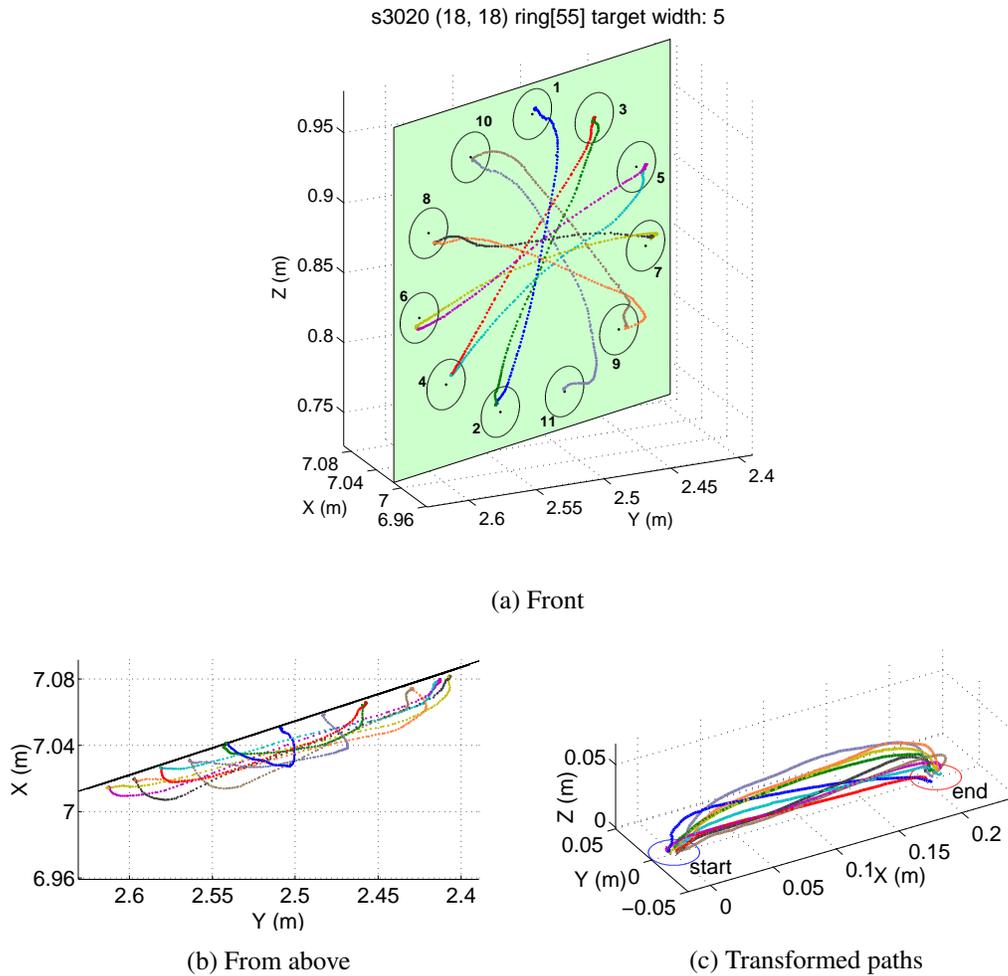


Figure 5.11: One participant’s motion paths for a target ring in the (18, 18) condition

in and adapts to a discrepant environment, the user would ideally readapt quickly to a one-to-one environment so that task performance in a real-world scenario does not suffer.

Figures 5.11–5.13 show examples of the finger motion tracker data for one participant’s (18, 18) and (0, 18) target rings. I refer to the motion between a pair of targets as a *path*. In the one-to-one target ring (Figure 5.11), the paths look fairly similar to each other in form and trajectory. In a discrepant ring (Figure 5.12), the paths look quite different from each other: there is a notable skew in the paths that move mostly horizontally. In the discrepant condition (0, 18), if a participant sees and believes the virtual board is oriented at  $18^\circ$  and has not adapted to the warped space, the tendency is to move the finger along a path consistent with the  $18^\circ$  orientation. For example, note that in the right-to-left paths (such as from target 3 to target 4), the finger moves away from the virtual board, and then the path changes direction sharply as it finally heads towards the target. The real board was

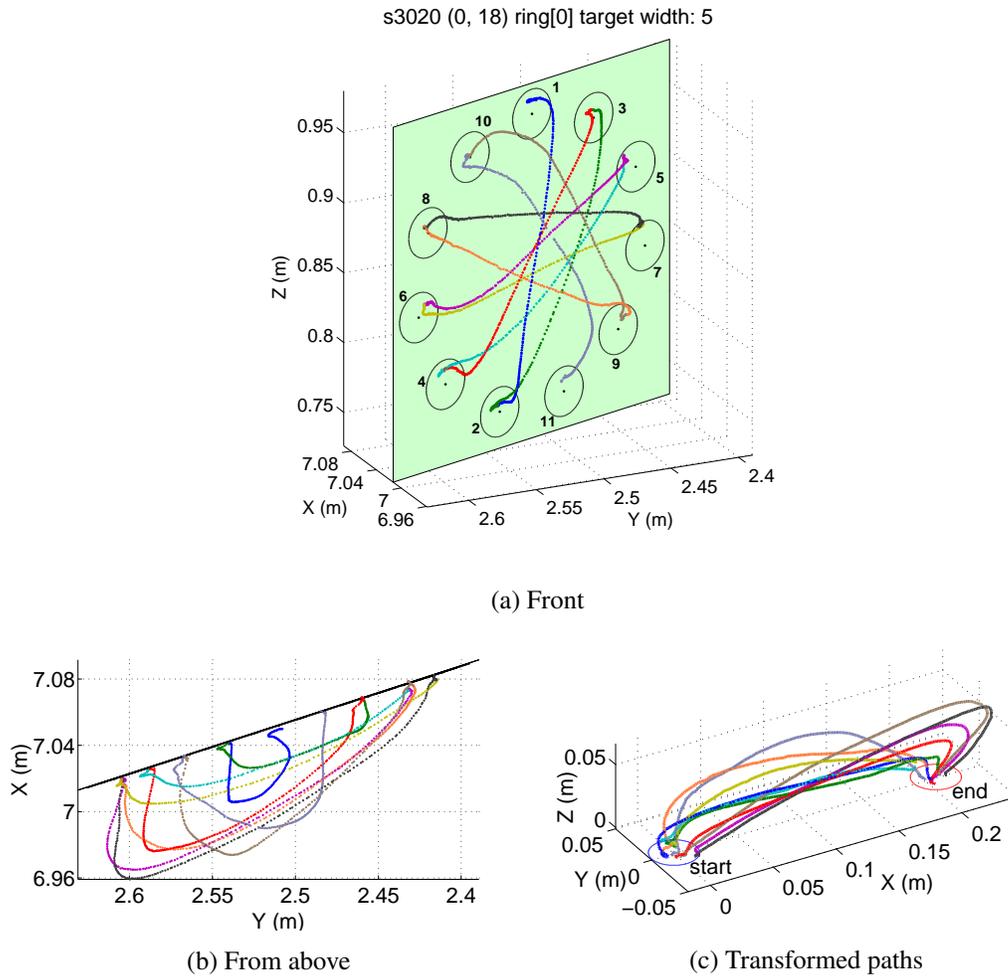


Figure 5.12: One participant’s motion paths for a target ring in the (0, 18) condition (first exposure to largest target width)

oriented at  $0^\circ$ , so achieving the correct motion in the virtual world required the real finger motion to be approximately parallel to the  $0^\circ$  board. Left-to-right paths (such as from target 4 to target 5) often hit the board early, again because of an expectation that the finger should move parallel to the  $18^\circ$  virtual board, rather than to the  $0^\circ$  real board. In the (18, 18) condition, any such path variability is not evident.

I suspect this difference in path variability is due to participants being not at all used to warped space. I hypothesized that with repeated exposure to a discrepant condition, path variability would decrease. To test this hypothesis, I defined a path variability metric similar to that used by Georgopoulos et al. [1981].

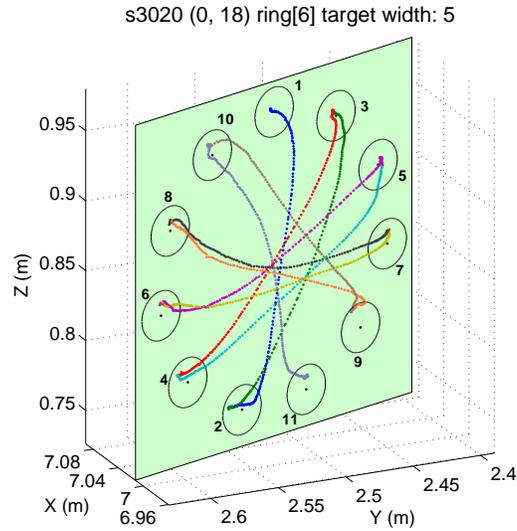
First, note that paths move in different directions and have different lengths and movement times. To compare them, I normalized them:

1. Each path was transformed onto a common task axis (Figures 5.11c, 5.12c, 5.13c).
2. Each path was then resampled at 50 equally spaced time intervals between its start and end touch points. Position data were interpolated for these sample points using spline interpolation.
3. For each of the 50 sample points, I collected the samples from all the transformed paths and computed their 3D standard deviation as a measure of path variability at that time sample (Equation (5.5), the 3D extension of Equation (5.4)).
4. I summed these path variability samples to get a particular ring's overall path variability over its ten paths (or in a few cases, nine paths if a path outlier had been removed).

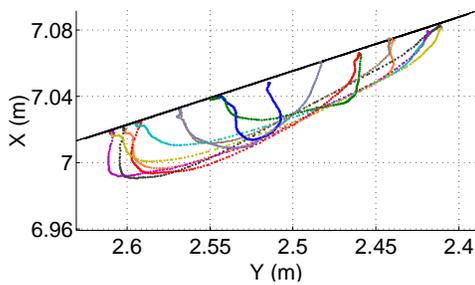
$$SD_{x,y,z} = \sqrt{\frac{\sum_{i=1}^N (\sqrt{(x_i - \bar{x})^2 + (y_i - \bar{y})^2 + (z_i - \bar{z})^2})^2}{N - 1}} \quad (5.5)$$

I hypothesized that this path variability would be highest when a participant was first exposed to a new condition and then lower after more exposure. Each of the eleven target ring blocks that each participant completed consisted of eight target rings: two adaptation rings which always had the largest target width (35mm) followed by the smallest target width (10mm), and six experimental rings with all six target widths presented in random order. On average, participants tended to move faster with the large targets than with the small targets. In rapid aiming tasks, hand motions generally have a ballistic movement to bring the hand near the target, followed by a slower deceleration phase to home in on the target [Woodworth, 1899]. Evidence indicates that visual feedback is used most during the deceleration phase, and less during the ballistic movement [Jones and Lederman, 2006]. I therefore expected that most of the path variability would be exhibited with the largest targets, because participants were more likely to move quickly during the ballistic movement.

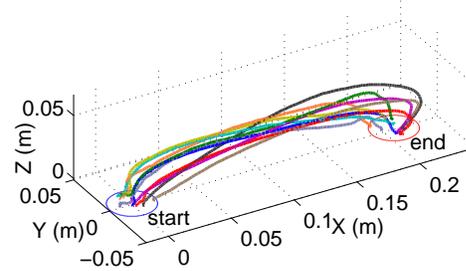
Because of the two adaptation target rings, the rings with the largest target widths were done twice in each block of eight target rings. This repetition enabled me to compare path variability at the beginning of each target block to path variability at some point during the experimental portion of each block. Because target ring widths were presented in random order, the two exposures to the largest target width were separated by variable numbers of target rings. However, from the pilot



(a) Front



(b) From above



(c) Transformed paths

Figure 5.13: One participant's motion paths for a target ring in the (0, 18) condition (second exposure to largest target width)

study, I expected most adaptation to occur during the two target rings immediately after beginning a new condition, and this variable separation likely did not have a large effect on the differences in path variability between exposures. One participant's first and second exposures to target rings with the largest target width in the (0, 18) condition are shown in Figures 5.12–5.13. Note that the paths vary much more in the first exposure than in the second exposure.

Figure 5.14 shows plots of path variability for all participants in the (18, 18) condition, and Figure 5.15 for the (0, 18) condition. The solid blue plot represents path variability by path sample for the first exposure, and the dashed red plot for the second exposure. With a few exceptions, total path variability appears to decrease or stay about the same between exposures. Note that overall the plots appear somewhat closer to each other in the (18, 18) condition than in the (0, 18) condition. I

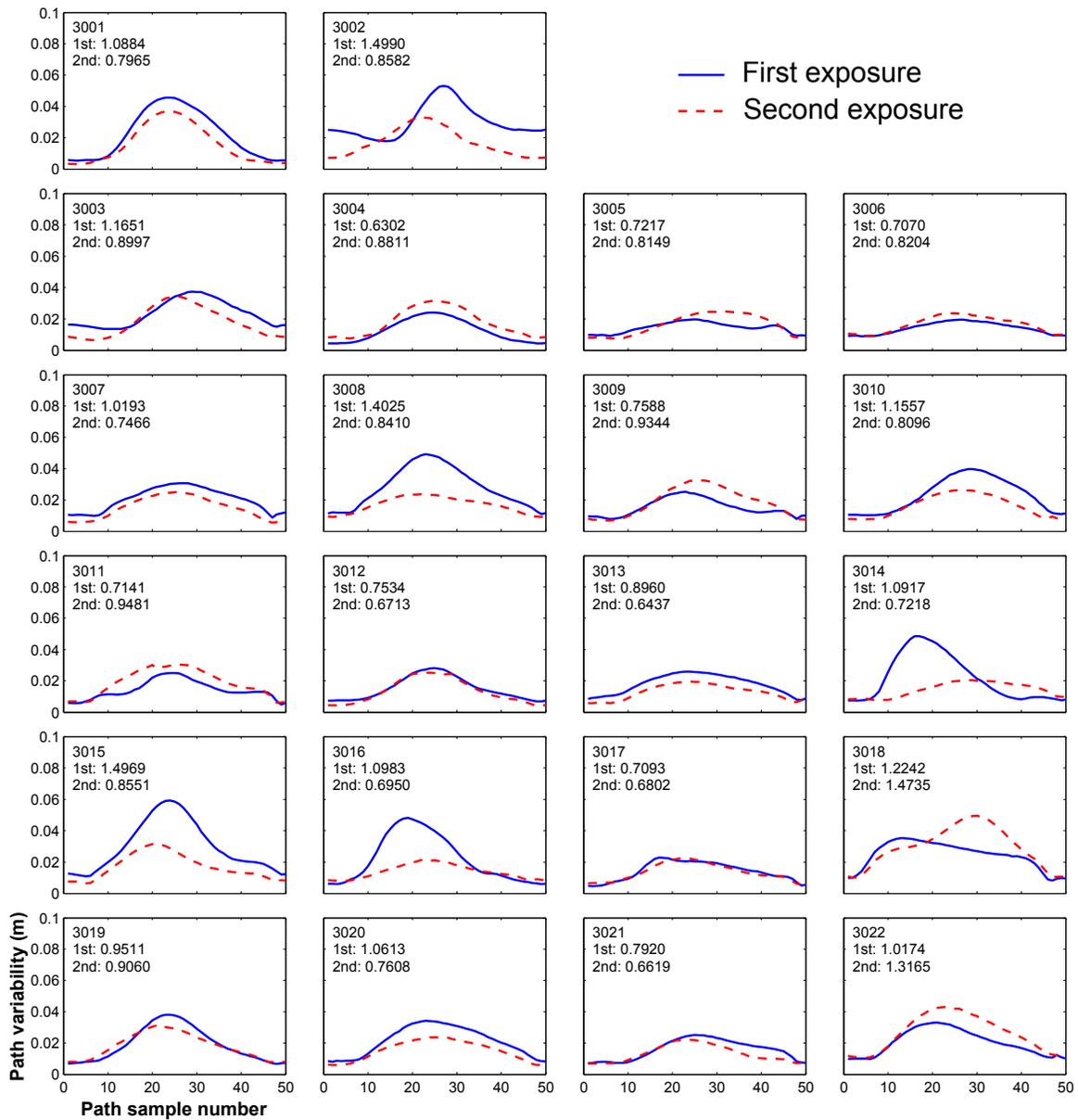


Figure 5.14: Path variability by path sample number (page 69) for each participant’s first and second exposures to the largest-target-width rings in condition (18, 18). The numbers in each plot are the participant number and the total path variability for the first and second exposures.

suspect that the differences between conditions are due to it taking longer for participants to adapt to discrepant conditions than to readapt to the one-to-one conditions from everyday life.

To test whether path variability decreased with more exposure to each condition, I modeled all first-to-second-exposure path variability differences across all participants and conditions simultaneously. I tested contrasts of interest with *t*-tests to determine if mean path variability differences were significantly different from zero. In all conditions, mean path variability differences were negative

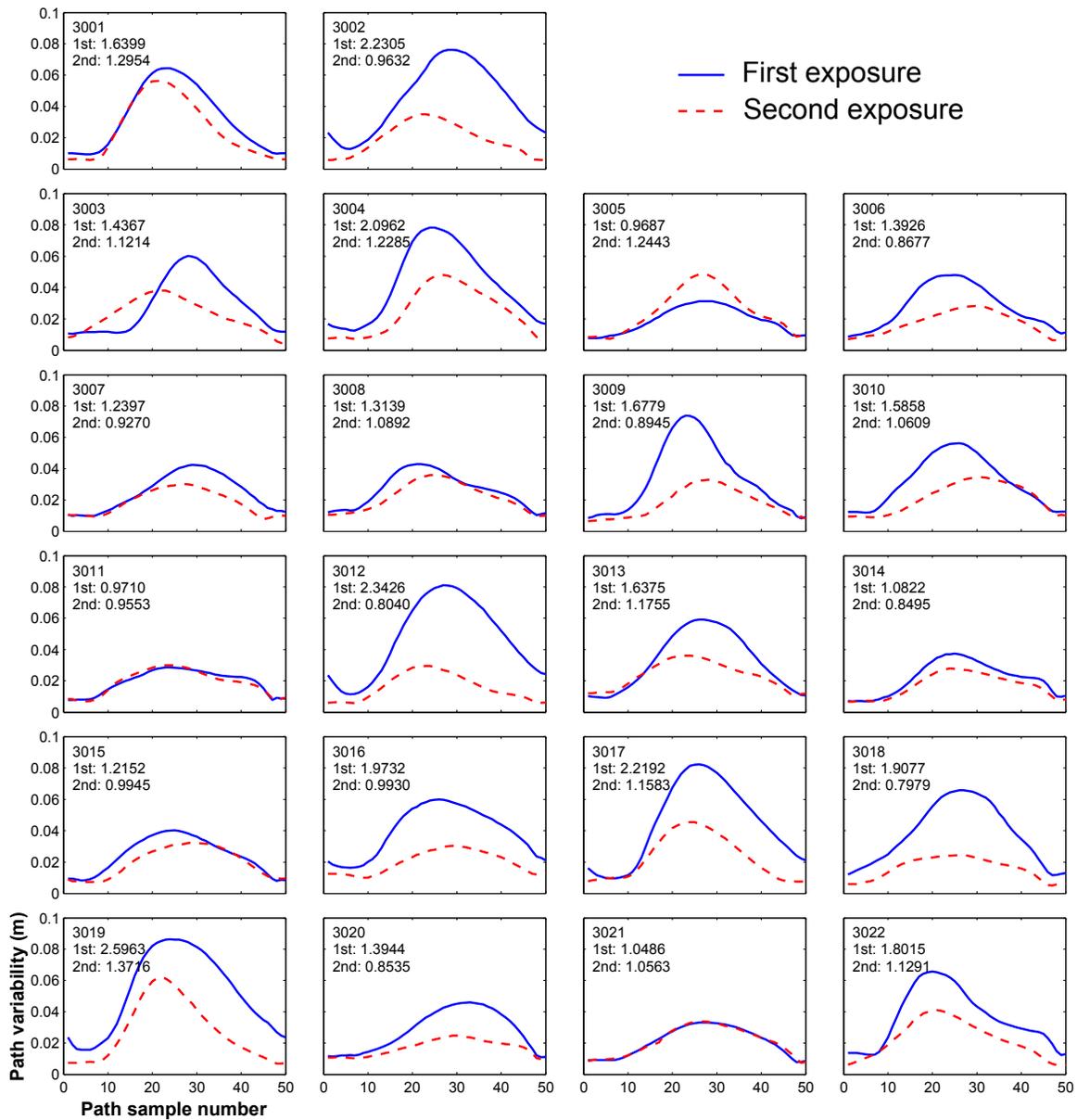


Figure 5.15: Path variability by path sample number (page 69) for each participant’s first and second exposures to the largest-target-width rings in condition (0, 18). The numbers in each plot are the participant number and the total path variability for the first and second exposures.

(lower path variability with more exposure). In all discrepant conditions and the (12, 12) and (18, 18) conditions, mean differences were significantly different from zero (Table 5.5).

Ideally, I would like to show that after some exposure to discrepant conditions, participants’ paths look very similar to the paths in analogous one-to-one conditions. Figure 5.16 shows path variability vs. virtual angle for the first and second exposures of all one-to-one and discrepant conditions across all participants. Having no basis for choosing a non-inferiority indifference zone, I cannot claim

Table 5.5: *t*-test results for path variability differences between first and second exposures to the largest-target-width ring in each condition

Condition	Mean path variability difference between exposures (m)	<i>t</i> -test result (path variability compared to 0)
(-18, -18)	-0.12	$t(210) = 1.86, p = 0.064$ , not significant
(0, -18)	-0.51	$t(210) = 8.16, p < 0.0001$ , significant
(-12, -12)	-0.071	$t(210) = 1.12, p = 0.27$ , not significant
(0, -12)	-0.36	$t(210) = 5.65, p < 0.0001$ , significant
(0, 0)	-0.053	$t(210) = 0.83, p = 0.4$ , not significant
(12, 12)	-0.13	$t(210) = 2.08, p = 0.039$ , significant
(0, 12)	-0.24	$t(210) = 3.8, p = 0.0002$ , significant
(18, 18)	-0.15	$t(210) = 2.31, p = 0.022$ , significant
(0, 18)	-0.59	$t(210) = 9.3, p < 0.0001$ , significant
(24, 24)	-0.082	$t(210) = 1.30, p = 0.19$ , not significant
(0, 24)	-0.71	$t(210) = 11.19, p < 0.0001$ , significant

that the discrepant second exposures have path variability no higher than the one-to-one exposures. However, it is evident that the discrepant first exposure generally has much higher path variability than any of the other discrepant or one-to-one exposures shown on the graph. Though currently lacking statistical support, these data suggest future hypotheses:

- Path variability decreases with more exposure to a discrepant condition
- Participants move their hands more consistently in one-to-one conditions

A decrease in path variability does not alone indicate adaptation; participants could lower path variability by moving more slowly, thereby minimizing the need to correct for over- or under-shooting targets. I posit that a general decrease in path variability coupled with a general increase in throughput indicates that participants are adapting to each condition. Figure 5.17 shows throughput vs. virtual

angle for the first and second exposures of all one-to-one and discrepant conditions across all participants. Again, the data for the discrepant first exposure look quite different from the discrepant second exposure and the two one-to-one exposures. Using the 1bps throughput indifference zone, however, I was unable to find that discrepant second exposures were no worse than one-to-one second exposures. Nevertheless, there appear to be simultaneous trends towards lower path variability and

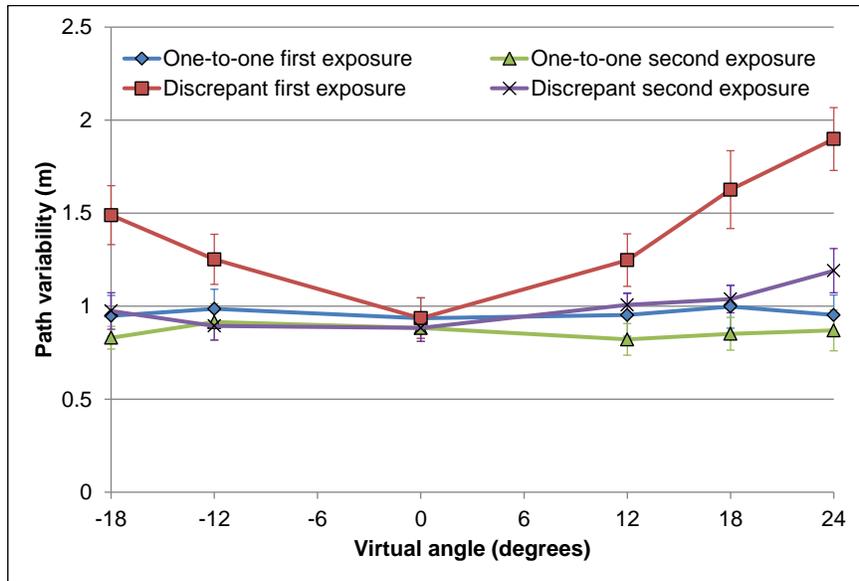


Figure 5.16: Path variability across all participants for one-to-one and discrepant conditions, first and second exposures to largest-target-width rings. Error bars are 95% confidence intervals.

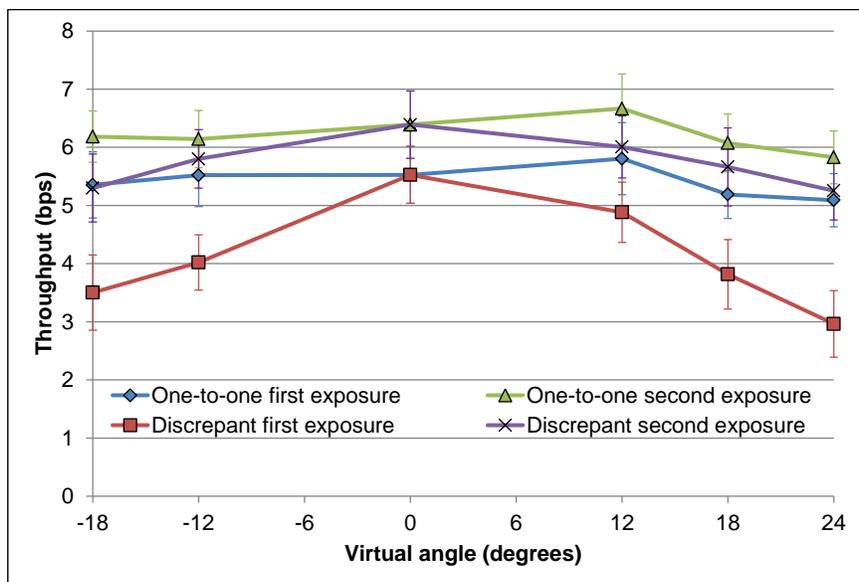


Figure 5.17: Throughput across all participants for one-to-one and discrepant conditions, first and second exposures to largest-target-width rings. Error bars are 95% confidence intervals.

higher throughput. With more time to interact with a discrepant condition, discrepant throughput may approach one-to-one throughput.

## 5.10 Exploratory detectability data

I also wanted to investigate at what discrepancy levels task-engaged users detect discrepancies, as well as task performance and detectability for discrepancies other than orientation. As a preliminary exploration of what levels of discrepancy users can detect, six members of the research group experienced several conditions in blocks similar to the main study. Participants were presented with nine blocks of target rings (four discrepant and five one-to-one generated from the angles  $-24^\circ$ ,  $-12^\circ$ ,  $0^\circ$ ,  $12^\circ$ , and  $24^\circ$ ). After every target ring, they judged whether what they saw and what they felt were the same or different, and gave a confidence value for their judgment (3 for most confident, 1 for least). The same/different responses were weighted by the confidence values and normalized giving a percentage—100%, all participants responded different with confidence 3 for all target rings in a given condition; 0%, all same with confidence 3; and 50%, same and different were equally frequent (Figure 5.18).

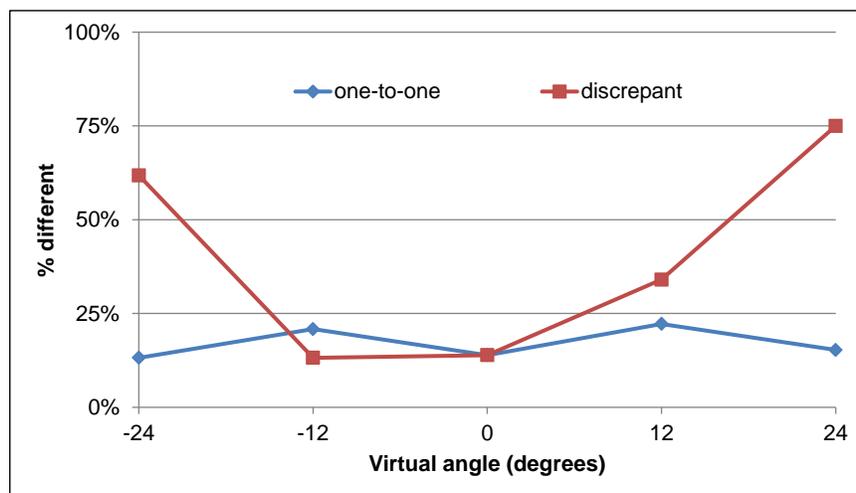


Figure 5.18: Exploratory discrepancy detectability data

There are many potential sources of bias in these data, including aggregating these data across participants who have different sensitivities and different definitions of confidence. However, the results inform future studies. Formal study is needed, but these data suggest that there is indeed a

non-zero amount of discrepancy that is undetectable. The  $-24^\circ$  and  $24^\circ$  discrepancies were generally easily detected, whereas the  $-12^\circ$  and  $12^\circ$  discrepancies were not.

## 5.11 Summary

I studied whether task performance changes when users are presented with discrepant real and virtual objects. The discrepant conditions (0,  $-12$ ), (0, 12), and (0, 18) yielded throughputs and error rates that are no worse than the analogous one-to-one conditions using the 1bps throughput and 9% error rate non-inferiority indifference zones. Significant differences were found in movement time. The choice of 1bps for throughput indifference might be too large, but the small mean differences in error rate and movement time suggest that discrepant objects can be used almost as well as one-to-one objects.

I also found that participants' motion paths tend to look more similar to each other as participants spend more time interacting with discrepant conditions. I take this as evidence that participants began adapting to discrepant conditions.

The results of this study suggest that Redirected Touching users adapt to discrepancies and can perform well once adapted. Therefore, I next investigated more deeply task performance and adaptation in the context of training.

## CHAPTER 6

# Redirected Touching: Training and Adaptation<sup>1</sup>

In Chapter 5, I presented a study investigating angular discrepancy in the context of aircraft cockpit procedures training. For a vertically oriented 10"x10" board, an angular discrepancy between seen and felt boards of 18° around the vertical axis yielded acceptable task performance relative to a one-to-one object mapping. The data also indicate that users adapt to the discrepancy. Study participants experienced several different angular discrepancies for short periods of time, requiring frequent adaptation. However, in an actual training scenario, users would typically interact with only one discrepancy. For example, though a system using Redirected Touching might support many aircraft, a pilot learning a cockpit procedure would train on a specific cockpit. Confident use of Redirected Touching for training requires knowing how well users perform with a longer exposure to a single warped space. It is also important to know how quickly users adapt to that warped space, and whether they easily readapt to the real world.

In the previous study, participants experienced only the virtual world. Aircraft pilots often train in simulators and in the real world, but always *for* the real world. I want to determine if Redirected Touching can be used effectively in a simulator to train for a real-world task, and whether that training can be as good as real-world training.

There is still much that is not known about how Redirected Touching affects users. If users must adapt to a warped space, it is possible their attention will be divided between the strange feeling they experience due to real-virtual discrepancies, and learning the high-level skill. This divided attention could hurt learning. Also, real-world task performance may suffer until users readapt to the real world. I therefore conducted a study to investigate more deeply the effects of interacting with a warped space:

---

<sup>1</sup>Portions of this chapter were previously published elsewhere [Kohli et al., 2013].

1. During a long exposure to a discrepant VE, how does task performance compare to an equivalent exposure to a one-to-one VE and a real environment?
2. Does virtual training transfer to the real world?
3. Do users adapt to the discrepant environment? How quickly?
4. How quickly do users readapt to the real world?
5. Do users notice the discrepancy?

To investigate these questions, I did a study similar to that of Chapter 5, but focused on comparing virtual training with real training.

## 6.1 Study

I wanted to investigate how task performance changes over time when users train with a discrepant virtual object, and how that task performance differs from when users train with a one-to-one virtual object or a real object.

To measure task performance, I once again used the ISO 9241-9 multi-directional tapping task (Section 5.4). This task reasonably approximates pushing a button sequence and is arguably applicable to cockpit procedures training. The study used a between-subjects design. Participants were assigned to one of three conditions:

1. Real ( $R$ ): task done on a real touchscreen monitor oriented at  $18^\circ$ , with no HMD (Figure 6.1, lower left).
2. Virtual one-to-one ( $V_{1-1}$ ): task done in HMD, with virtual and real monitors both oriented at  $18^\circ$  (Figure 6.1, upper and lower left).
3. Virtual discrepant ( $V_d$ ): task done in HMD, virtual monitor at  $18^\circ$  and real monitor at  $0^\circ$  (Figure 6.1, upper left and lower right). Virtual space was warped using thin-plate splines (Section 4.3.2).

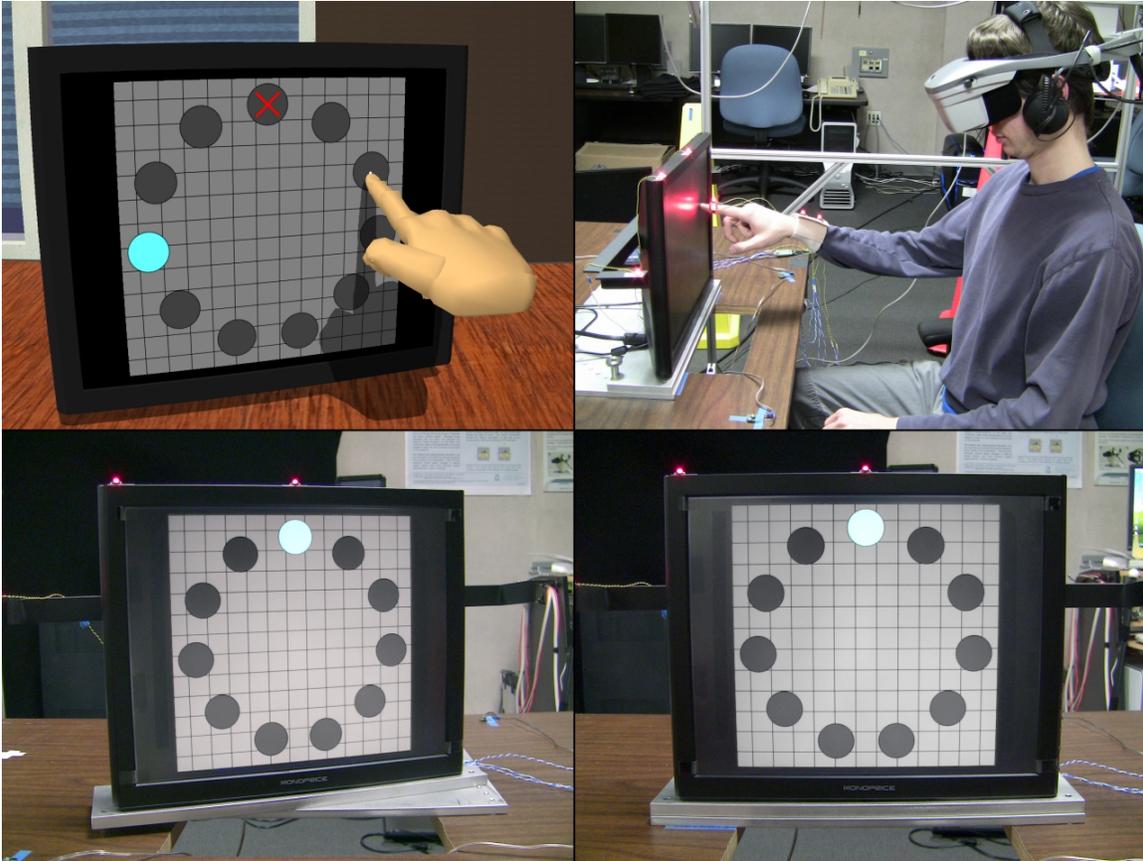


Figure 6.1: A user sees an  $18^\circ$  monitor in an HMD (upper left) while touching a  $0^\circ$  touchscreen monitor (upper right); Lower left: touchscreen oriented at  $18^\circ$ ; Lower right: touchscreen at  $0^\circ$ .

As in the previous study, the experimental stimuli consisted of a target distance of 21cm and six different target widths: 10mm, 15mm, 20mm, 25mm, 30mm, and 35mm.<sup>2</sup> Each condition proceeded in three phases:

1. Pretest: 2 target rings, the largest and the smallest target widths
2. Training: 72 target rings, presented in twelve blocks of six rings
3. Posttest: 18 target rings, in three blocks of six

These blocks of six rings consisted of one ring of each of the six target widths. In all conditions, the Pretest and Posttest phases were done in the real world, i.e., on the touchscreen monitor at  $18^\circ$  without the HMD. The Training phase was done in each participant's randomly assigned study condition.

---

<sup>2</sup>The actual target distance was  $\sim 20.8$ cm. See Appendix B for details.

The Pretest phase was very short to avoid too much training prior to the Training phase. In the Training and Posttest phases, the presentation order of different target widths was randomized per block, without replacement. Target rings were presented in the same order to all participants.

I did not expect the  $V_d$  condition to produce better results than  $V_{1-1}$  or  $R$ . I did want to determine if task performance in  $V_d$  is *no worse* than in  $V_{1-1}$  and  $R$ , which I again test for by using a non-inferiority test (Section 5.5).

### 6.1.1 Non-inferiority indifference zones

The dependent variables measuring task performance were throughput, error rate, and movement time. Non-inferiority testing requires defining indifference zones for the metrics of interest in the context of the application. I chose the same indifference zones for throughput and error rate as in the previous study, and defined for the first time an indifference zone for movement time. The maximum allowable difference between conditions for each metric was:

- Throughput: 1bps. A 2004 survey of ISO 9241-9 studies found that the range of throughputs for computer-mouse pointing in five studies was 3.7-4.9bps, a range of 1.2bps [Soukoreff and MacKenzie, 2004]. As before, I used the more conservative 1bps.
- Error rate: 1/11 missed targets (9%), the smallest unit of error I could measure.
- Movement time: 100ms. This indifference zone was not chosen in the previous study, but the study's data suggest that it is reasonable. Redirected Touching relies on visual feedback; a user will adjust her real hand motion based on what she sees her virtual hand doing. The difference in movement time between conditions should be no more than the time it takes to process visual feedback. Studies of discrete aiming movements have shown that it takes 100–200ms for visual feedback to influence the accuracy of an ongoing movement [Keele and Posner, 1968; Zelaznik et al., 1983; Jones and Lederman, 2006]. I chose the low end of that range, 100ms.

### 6.1.2 Apparatus

The experimental task is presented to the user on either a resistive touchscreen monitor when in the real world or in an NVIS nVisor SX60 HMD when in the VE. A 3rdTech™ HiBall™-3000 tracks

the user's head and a PhaseSpace® IMPULSE™ tracking system tracks the right index finger and the touchscreen monitor. VRPN communicates with trackers [Taylor et al., 2001]. The VE is rendered using Gamebase Co. Ltd's Gamebryo™ game engine running on a hex-core 3.3GHz Intel i7-3960X machine with 16GB of RAM and an NVIDIA® GeForce® GTX 680 GPU. End-to-end system latency is ~50-60ms. The rendered output is corrected for the HMD's pincushion distortion using the HMD calibration work by Kuhl et al. [2008]. As before, I send a larger-than-display-resolution texture to the pincushion correction shader so that pixel data are available for the corners.

The touchscreen monitor is mounted on a rotating base (Figure 6.1, lower left). The study task is presented in a 10"x10" area of the monitor to be consistent with the previous study.

### 6.1.3 Participants

Forty-two paid participants (26 male, 16 female, aged 18-33, mean 21) took part; 14 were randomly assigned to each condition. All were right-handed and had normal or corrected-to-normal vision.

### 6.1.4 Study procedure

Participants read and signed an informed consent form, entered the lab, and sat in front of the touchscreen. The touchscreen was oriented at 18° about the vertical axis (Figure 6.1, lower left). I explained the multi-directional tapping task. Participants were instructed to perform the task as quickly and as accurately as possible and to leave missed targets uncorrected. A white dot showed where users touched the touchscreen, and a red X was shown on missed targets.

Participants were fitted with the finger trackers, calibrated for finger length. In all three conditions, participants were presented with two Pretest target rings: the largest- and smallest-target-width rings.  $R$  condition participants moved directly to the Training phase of the study.  $V_{1-1}$  and  $V_d$  participants were instead fitted with the HMD, which initially showed a white "focus" screen with a black rectangle. I faded participants from the focus screen into the VE, where they saw a virtual touchscreen monitor (at 18°) and their virtual hand (Figure 6.1, upper left).

After participants tried touching the virtual screen once or twice, they were faded out to the focus screen again so I could reiterate instructions. These instructions were used in both virtual conditions, but were designed particularly for the  $V_d$  condition. The  $V_d$  condition proceeded similarly

except that I quietly rotated the real touchscreen monitor so that it was oriented at  $0^\circ$  (Figure 6.1, lower right). When  $V_{1-1}$  and  $V_d$  participants were faded back into the VE, they were shown a virtual touchscreen monitor at  $18^\circ$ . In the  $V_d$  case, space was warped to map the  $18^\circ$  virtual monitor onto the  $0^\circ$  real monitor.<sup>3</sup> Participants then did the multi-directional tapping task 72 times ( $= 72 \times 11$  targets). Participants were required to take a one-minute break after 24 and 48 target rings in the Training phase. They could also take breaks if desired before beginning any target ring.

After the Training phase,  $R$  participants went on to the 18-ring Posttest (three 6-ring blocks). For the  $V_{1-1}$  and  $V_d$  participants, the VE again faded out to the focus screen. More instructions were given, and in the  $V_d$  condition, the touchscreen was quietly rotated back to  $18^\circ$ . The HMD was then removed, and participants did the Posttest.

Participants then filled out a short post-questionnaire (Appendix A.3).  $R$  condition participants were allowed to try out the HMD. Where  $R$  participants tried on the HMD, they did a shortened version of the  $V_d$  condition to yield more feedback on discrepancy. The real monitor was quietly rotated when they entered the VE, but the HMD was removed without my having rotated the monitor back, to see each participant's reaction to the rotated monitor.

All participants then had a semi-structured open-ended interview, were told the study's purpose, and were paid. The study lasted 45–60 minutes total per participant, the experimental conditions 15–25 minutes.

## 6.2 Results: task performance

The results represent data after removing outliers (undetected or double touchscreen touches, forgotten targets, and motion paths with length more than  $\pm 3\sigma$  from the mean). In the graphs below, the horizontal axis is divided by study phase. Each data point is the mean for a particular block of target rings, across all participants in a particular condition. The error bars show 95% confidence

---

<sup>3</sup>During target rings, the warped space was computed using only the real and virtual monitor geometries and a distant boundary beyond which space was not warped; the table was not used. Consequently, when touching the real table straight on, the virtual finger would be shown at an angle. So that participants would not notice this discrepancy during breaks, I changed the warping behavior between target rings. The hand pose was interpolated between the warped and one-to-one spaces: the closer the finger to the monitor, the more warped the space, and the closer to the table edge, the more one-to-one the space.

intervals. The vertical dashed lines are the beginning of the Training phase, the target rings after the first break, the target rings after the second break, and the Posttest phase.

Figures 6.2–6.4 show mean throughput, error rate, and movement time respectively across all participants.

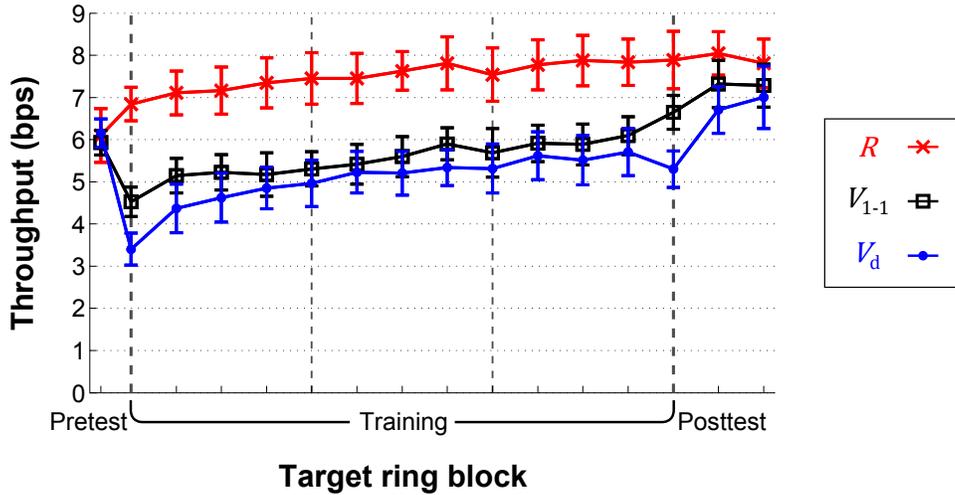


Figure 6.2: Mean throughput across all participants for  $R$ ,  $V_{1-1}$ , and  $V_d$  conditions. Error bars are 95% confidence intervals.

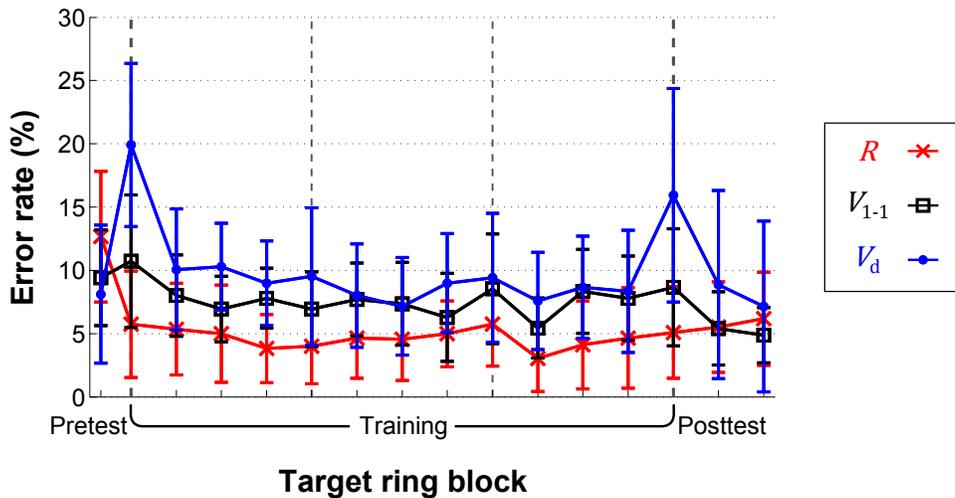


Figure 6.3: Mean error rate across all participants for  $R$ ,  $V_{1-1}$ , and  $V_d$  conditions. Error bars are 95% confidence intervals.

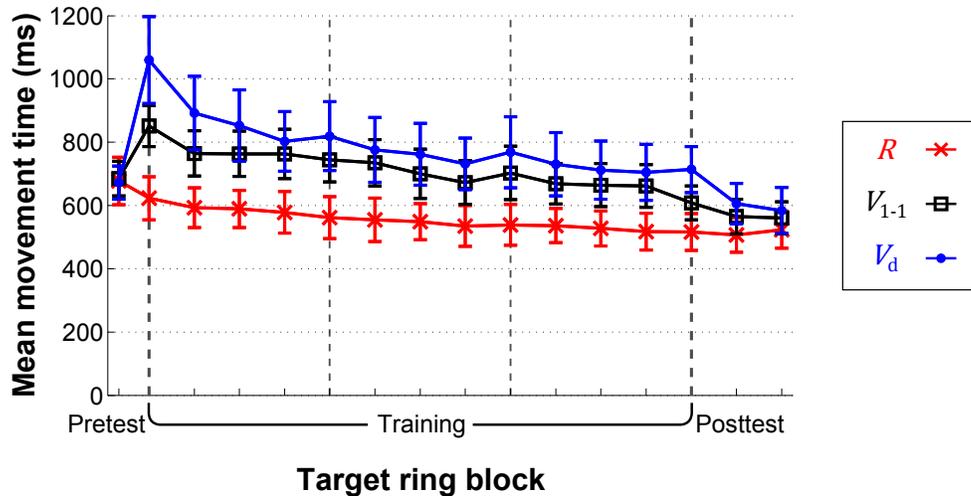


Figure 6.4: Mean movement time across all participants for  $R$ ,  $V_{1-1}$ , and  $V_d$  conditions. Error bars are 95% confidence intervals.

### 6.2.1 Posttest vs. Pretest

Effective training for the real-world task should show as a Posttest-Pretest difference. Figure 6.2 shows a general upward trend in all three conditions, although there is a drop at the beginning of the Training phase for both virtual conditions, probably related to users adjusting to the newly donned HMD. I suspected readaptation to the real world affected performance during the first Posttest target ring block. My focus here is on performance after (re)adaptation, so that block was removed from analysis.

I tested for significant differences between the Posttest and Pretest blocks using an ANOVA (condition: between-subjects; target ring block: within-subjects). There were significant differences in throughput (Table 6.1) and movement time (Table 6.3) for all conditions. For error rate, there was a significant difference for only the  $R$  condition (Table 6.2). In all cases where there was a significant difference, the bound of the associated one-tailed 95% confidence interval lay outside the predetermined indifference zone, so non-inferiority was not shown.

Table 6.1: ANOVA results for mean throughput differences

Condition	Mean throughput difference (Post-Pre)	One-tailed 95% conf. interval	Result
$R$	1.82 bps	< 2.16 bps	$F(1,711)=76.91$ $p < 0.0001$ significant
$V_{1-1}$	1.15 bps	< 1.50 bps	$F(1,711)=31.02$ $p < 0.0001$ significant
$V_d$	0.70 bps	< 1.05 bps	$F(1,711)=10.96$ $p = 0.001$ significant

Table 6.2: ANOVA results for mean error rate differences

Condition	Mean error rate difference (Post-Pre)	One-tailed 95% conf. interval	Result
$R$	-7%	> -10.8%	$F(1,711)=9.80$ $p = 0.0018$ significant
$V_{1-1}$	-3%	> -6.8%	$F(1,711)=1.88$ $p = 0.17$ not significant
$V_d$	-0.1%	> -3.9%	$F(1,711)=0.0025$ $p = 0.96$ not significant

Table 6.3: ANOVA results for mean movement time differences

Condition	Mean movement time difference (Post-Pre)	One-tailed 95% conf. interval	Result
$R$	-162ms	> -195ms	$F(1,711)=63.52$ $p < 0.0001$ significant
$V_{1-1}$	-107ms	> -141ms	$F(1,711)=27.98$ $p < 0.0001$ significant
$V_d$	-78ms	> -112ms	$F(1,711)=13.91$ $p = 0.0002$ significant

In Figure 6.3, the  $R$  Pretest error rate appears different from  $V_{1-1}$  and  $V_d$  (though within 2/11 missed targets). Compared to Training and Posttest, Pretest was more susceptible to between-subject variability: there were fewer Pretest rings. Pretest included only the largest and smallest target widths, and error rates were consistently higher on the smallest. Because a target is either hit or missed, small variations in movement speed can lead to very different error rates. Pretest movement speeds across all three conditions were similar to one another, but  $R$  participants varied more. I am not confident that the  $R$  Posttest-Pretest error rate represents a true difference. However, Pretest throughput and movement time values across all participants were similar (Figures 6.2, 6.4), so I am more confident in the throughput and movement time Posttest-Pretest results.

## 6.2.2 Non-inferior task performance

The data indicate that there was some training between the Pretest and Posttest phases of each condition. Ideally,  $V_d$  would also be shown to be no worse than  $R$  and  $V_{1-1}$ . I computed the mean differences between each pair of conditions and plotted their one-tailed 95% confidence intervals, as well as the indifference zones. With that data, I can test for non-inferiority, and also see when during the sequence of target rings non-inferiority is achieved, if at all.

Figure 6.5 shows mean throughput differences and confidence intervals. The shaded region is the 1bps indifference zone. For the Pretest, the confidence intervals all lie within the indifference zone, meaning that  $V_d$  is no worse than  $R$  and  $V_{1-1}$ , and  $V_{1-1}$  is no worse than  $R$ . However, when the Training phase begins, this is no longer true.  $V_d$  and  $V_{1-1}$  get better over time, but never get to be no worse than  $R$ . After about 18 target rings,  $V_d$  becomes no worse than  $V_{1-1}$  and remains that way for most of the Training phase. Upon starting the Posttest,  $V_d$  participants' performance degrades and then improves as they readapt to the real world, until the end when  $V_d$  is again no worse than  $V_{1-1}$ .

Figure 6.6 shows mean error rate differences and confidence intervals, with the 9% indifference zone shaded. There is generally evidence of non-inferiority between  $V_d$  and  $V_{1-1}$ , and between  $V_{1-1}$  and  $R$ . However, the  $(V_d - R)$  confidence intervals lie outside the indifference zone for the period

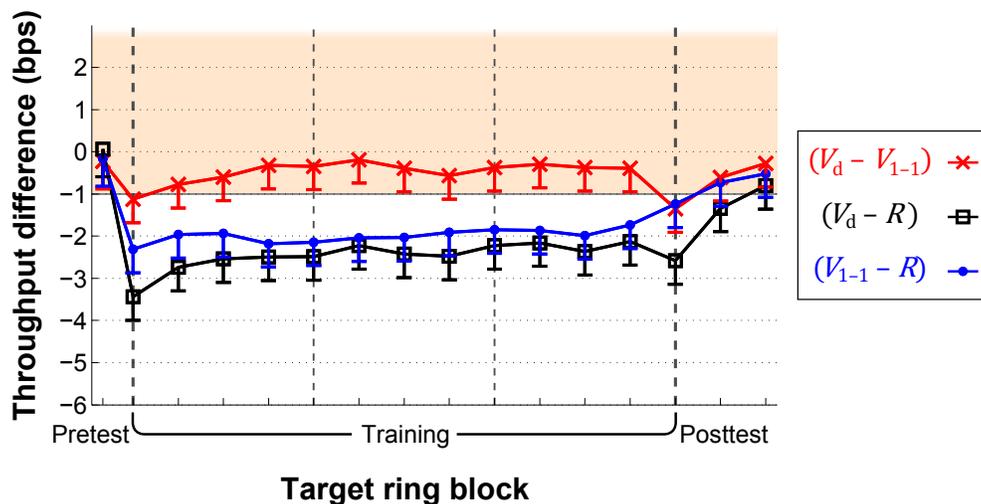


Figure 6.5: Mean throughput differences between pairs of conditions. Error bars are one-tailed 95% confidence intervals. Shaded area represents non-inferiority indifference zone.

before the first Training break, and then hover near the boundary of the indifference zone. By the end of the Posttest,  $V_d$  is no worse than  $V_{1-1}$  and  $R$ , and  $V_{1-1}$  is no worse than  $R$ .

Figure 6.7 shows mean movement time differences and confidence intervals, with the 100ms indifference zone shaded. In the Training and Posttest phases,  $V_d$  and  $V_{1-1}$  are never no worse than  $R$ , and  $V_d$  is never no worse than  $V_{1-1}$ . These results are consistent with my observations that participants moved faster in the  $R$  condition, and I suspect that adaptation to discrepancy led to slower movement times in the  $V_d$  condition.

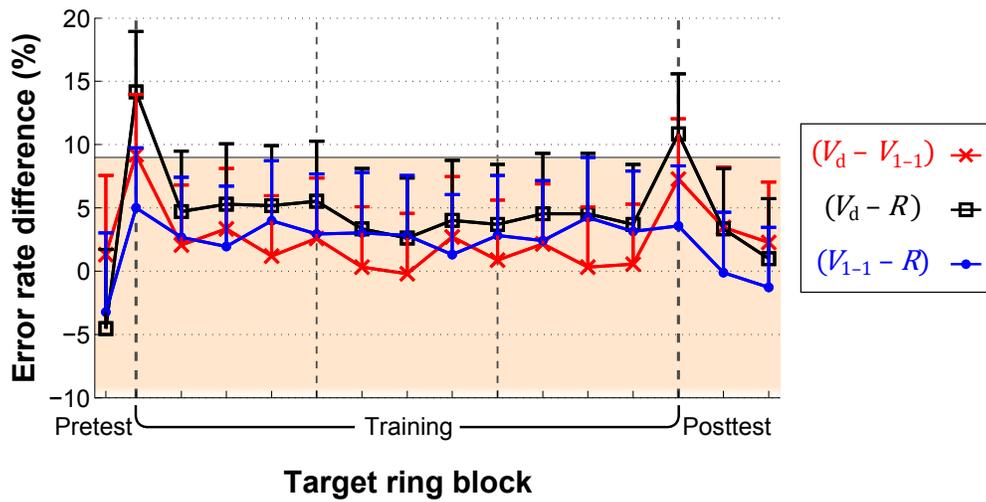


Figure 6.6: Mean error rate differences between pairs of conditions. Error bars are one-tailed 95% confidence intervals. Shaded area represents non-inferiority indifference zone.

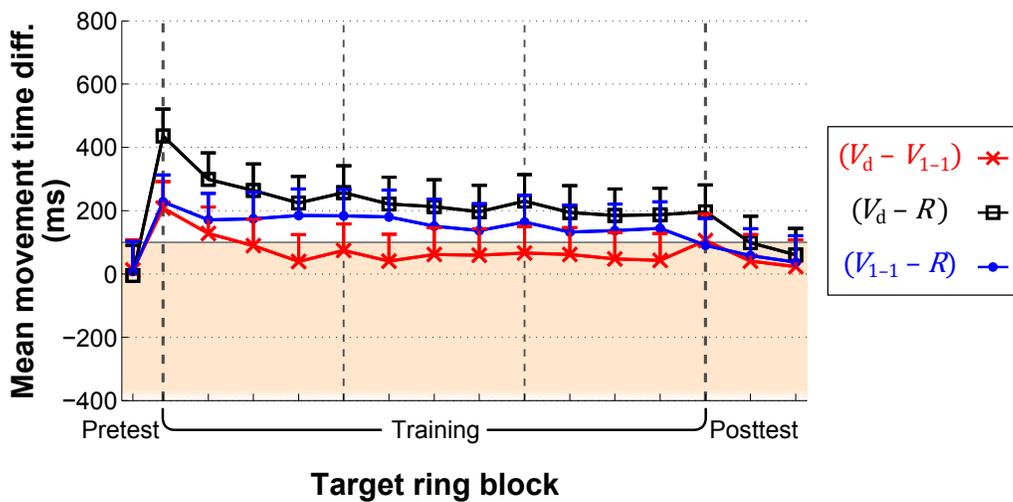


Figure 6.7: Mean movement time differences between pairs of conditions. Error bars are one-tailed 95% confidence intervals. Shaded area represents non-inferiority indifference zone.

### 6.3 Measuring adaptation by path variability

Most participants in the  $V_d$  condition reported a strange sensation when touching the monitor: something felt odd about their depth perception when first exposed to the discrepant environment, and again when returning to the real world, but the feeling mostly went away after doing the task repeatedly. These reports are consistent with J. J. Gibson's findings [1933]: participants wearing vision-distorting lenses adapted to a distorted visual field, and after removing the lenses, participants experienced a negative after-effect (distortion in the opposite direction) until they readapted to the undistorted real world. I suspect that our participants experienced something similar.

Adaptation has potential implications for training: if users adapt to a discrepant environment, it is important that they readapt quickly to the real environment so that they are not mistrained. I used path variability to measure whether participants adapted to the discrepant VE and how long it took them to adapt and readapt.

Figures 6.8–6.9 show one  $V_d$  participant's motion paths for the first Training target ring and a target ring just after the first break. Note that in the first target ring, the paths are not particularly smooth. Figures 6.10a–b show the paths viewed from above, and Figures 6.10c–d show the paths transformed to a common angle and normalized as in Section 5.9. The first target ring's paths exhibit a notable skew (Figures 6.10a, c). Because the real monitor was oriented at  $0^\circ$  and the virtual monitor at  $18^\circ$ , participants expected to need to pull their real finger back toward them when moving right-to-left, when in fact the finger needed to move directly left to stay near the monitor. As a result, there are sharp turns back towards the monitor on the left side when participants realize their finger is not moving where they expect. When moving left-to-right, participants often hit the right side of the monitor early. In the target ring just after the first break (Figure 6.9), the paths are much smoother, and there is no apparent skew despite the continued  $18^\circ$  discrepancy (Figures 6.10b, d).

p5008 (VirtualDiscrepant) Training[0] target width: 5

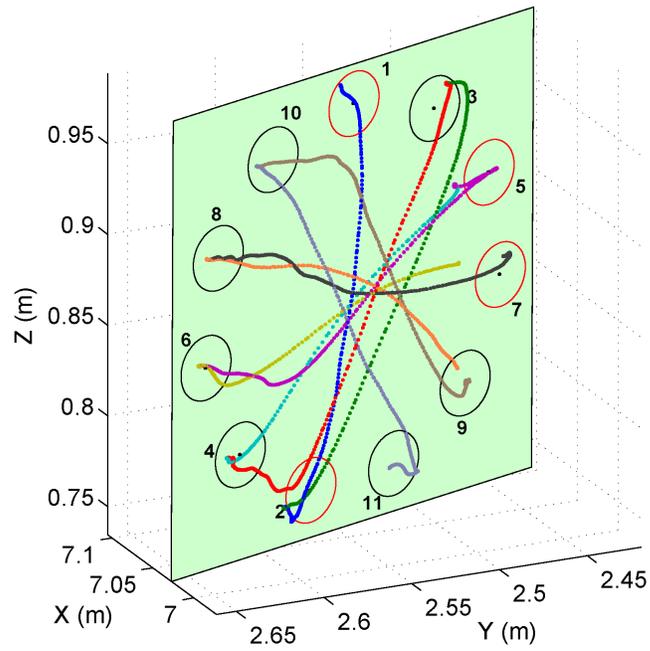


Figure 6.8: A  $V_d$  participant's motion paths, first Training target ring.

p5008 (VirtualDiscrepant) Training[27] target width: 5

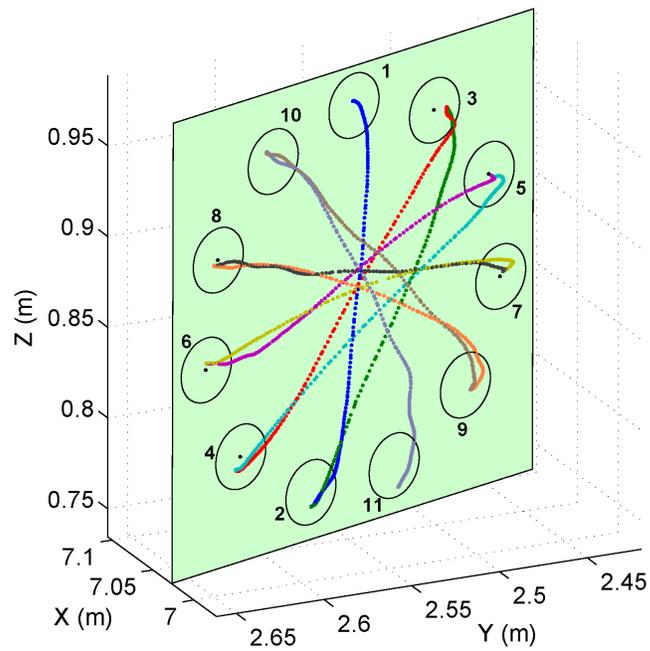


Figure 6.9: A  $V_d$  Training target ring just after the first break.

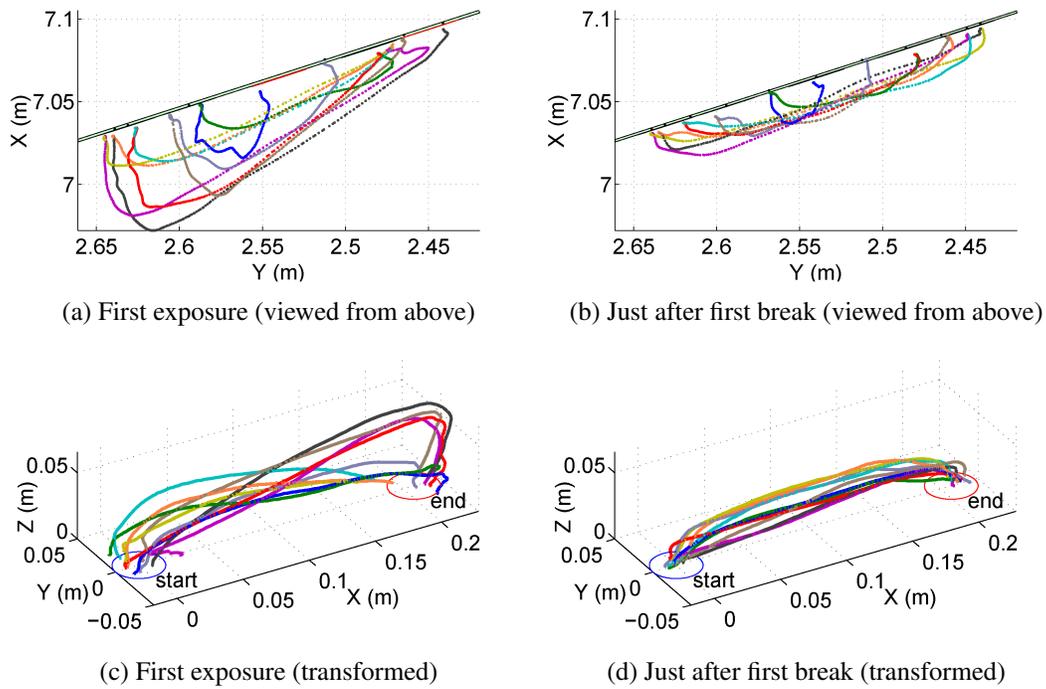


Figure 6.10: A  $V_d$  participant's motion paths

The paths in Figure 6.8 are quite different from one another, whereas the paths in Figure 6.9 are alike. These differences between motion paths were seen for all  $V_d$  participants, indicating that participants were (consciously or otherwise) adapting to the warped space. As with the previous study, I hypothesized that as participants adapt to a discrepant environment, their motion paths become more similar. I again tested this hypothesis using the path variability metric in Section 5.9.

Specifically, I hypothesized that path variability would decrease with more exposure to the discrepant condition, and that after adaptation,  $V_d$  path variability would be no higher than  $V_{1-1}$  and  $R$  path variability. I had no prior basis for choosing path-variability indifference zones. Therefore, under the assumption that  $R$ -condition motion paths would best exhibit low path variability, I used  $R$ -condition data to define indifference zones.

I computed path-variability standard deviation for each  $R$ -condition target ring block, and used one standard deviation as that block's indifference zone. Consequently, the indifference zones change slightly from block to block.

## 6.4 Results: adaptation

Figure 6.11 shows mean path variability across all participants, by block. Note that the path variability starts at about the same level in the Pretest, but the path variability is much higher in  $V_d$  than in  $V_{1-1}$  and  $R$  when the Training phase starts. With repeated exposure to the discrepancy,  $V_d$  path variability decreases and appears to converge with  $V_{1-1}$  and  $R$ . When the Posttest starts, there is another spike in  $V_d$  path variability (and a smaller spike in  $V_{1-1}$  as well), but the three conditions appear to converge again after the first Posttest target ring block.

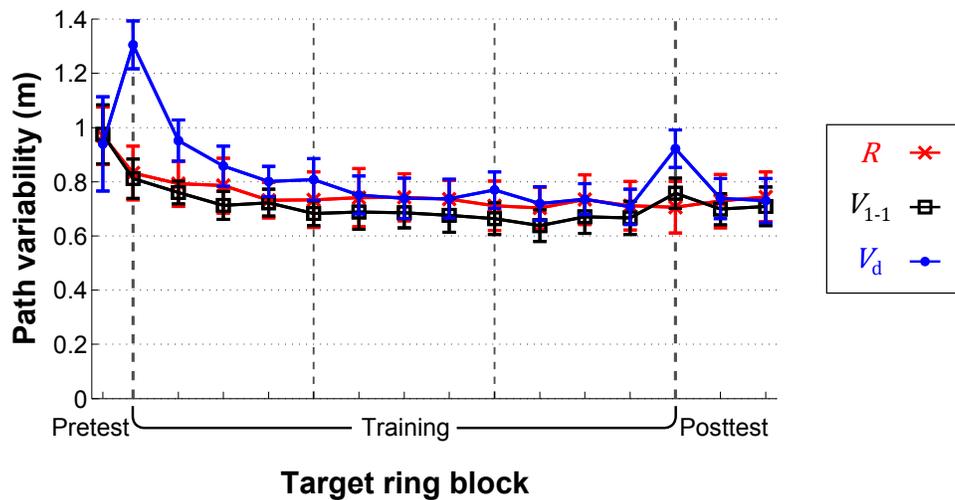


Figure 6.11: Mean path variability across all participants for  $R$ ,  $V_{1-1}$ , and  $V_d$  conditions. Error bars are 95% confidence intervals.

Figure 6.12 shows mean path variability differences and confidence intervals, with the indifference zones shaded. In the Pretest,  $V_d$  and  $V_{1-1}$  are no worse than  $R$ , and  $V_d$  is no worse than  $V_{1-1}$ . When the Training phase begins,  $V_{1-1}$  is no worse than  $R$ , but  $(V_d - V_{1-1})$  and  $(V_d - R)$  are substantially outside the indifference zone. However, with continued exposure to the task, path variability across all three conditions begins to look more alike. After the first break time,  $V_d$  is mostly no worse than  $V_{1-1}$ , and  $V_d$  is no worse than  $R$ . There are small spikes in path variability at the first and second break times. I suspect these spikes are due to having to readapt slightly to the discrepant environment after having rested. Another spike is evident at the beginning of the Posttest, which indicates that participants needed to readapt to the real world. That is consistent with my

observations and participant comments. After the first Posttest target ring block,  $V_d$  is once again no worse than  $V_{1-1}$  and  $R$ , and  $V_{1-1}$  is no worse than  $R$ .

It is interesting to note that in general,  $V_{1-1}$  path variability was lower than  $R$  path variability. I suspect this is because  $R$  participants tended to move faster than  $V_{1-1}$  participants.

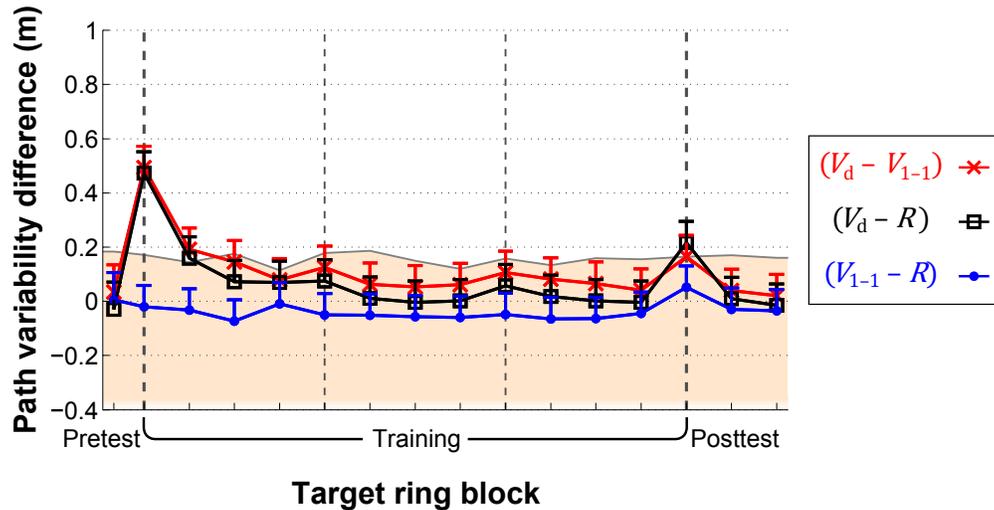


Figure 6.12: Mean path variability differences between pairs of conditions. Error bars are one-tailed 95% confidence intervals. Shaded area represents non-inferiority indifference zone.

## 6.5 Discussion

For all conditions, Posttest throughput and movement time were better than in the Pretest; I cannot definitively say that error rates in Posttest were better than in Pretest for any condition. These results suggest that training for the real task occurred in all conditions and that training in the virtual world transferred to the real world. However, training in the real condition seemed more effective than in either virtual condition.

For the virtual conditions, one could posit that Posttest performance was due to training during Posttest rather than during Training. However, the magnitude of the subsequent performance increase throughout Posttest is unlikely to be explained by continued real-world training. There was a performance drop when Posttest started, particularly in the  $V_d$  condition. I argue that the initial decrease in task performance in the  $V_d$  Posttest was due to readaptation to the real world; path variability metrics and participant comments support that argument. I further argue that the overall

Posttest task-performance increase relative to Pretest was due to repeated VE task exposure during the Training phase.

I generally observed that participants performed much better once adapted to the warped space: after adaptation,  $V_d$  throughput was generally no worse than in  $V_{1-1}$ ;  $V_d$  error rates were generally no worse than in  $V_{1-1}$  and  $R$ . However,  $V_d$  and  $V_{1-1}$  participants were not able to perform as quickly as in  $R$ . Aside from the discrepancy, factors such as HMD weight and limited field of view may have caused participants to move more slowly.

A decrease in path variability coupled with an increase in task performance clearly indicates that participants adapted to the discrepancy. Though it usually took until the first break time (~24 target rings, ~5–6 min.) for participants to fully adapt, most of the adaptation occurred during the first six Training target rings (~6–12 target rings, ~1–2 min.). Training with Redirected Touching may be effective if a user is first given time to adapt to a discrepant environment.

### **6.5.1 Participant comments: discrepancy detected?**

I asked participants to describe their study experience. I wanted to find out whether people noticed the  $18^\circ$  discrepancy between the real and virtual monitors, particularly for the  $V_d$  participants and those  $R$  participants who experienced an abbreviated version of the  $V_d$  condition.

All 14  $V_d$  participants felt a strange sensation when first starting the virtual task and when first starting the Posttest. A typical comment was that depth perception felt a little off, but that by the first break time or sooner, it felt more normal. Several participants mentioned that until they grew accustomed to the VE, it felt like the virtual hand lagged behind their real hand. This perceived lag was likely due to the warped space causing discrepant real and virtual hand motion velocities. Ten of the 14  $V_d$  participants could not identify that an orientation discrepancy caused the strange sensation; the other 4 did.

Seven  $R$  participants tried the abbreviated  $V_d$  condition. Five did not notice that the real monitor was oriented differently than the virtual monitor, and when the HMD was removed, were surprised to see the monitor's orientation. Two correctly identified that the monitors were at different orientations.

All but 2 of the 21 participants said that their perception in the VE began to feel more normal after doing the task a few times. One  $V_d$  participant noted that about half-way into the Training phase,

he no longer had to think about his hand motion—it became more instinctive. All  $V_d$  participants found the task to become easier in the Posttest after an initial strange feeling, indicating readaptation.

### 6.5.2 Other participant comments

Two  $V_{1-1}$  participants felt they moved more slowly in the virtual world. The data and my observations confirmed that.

About half of all participants responded that they did not experience any fatigue. Others experienced only moderate fatigue; only one indicated that it may have slightly affected her performance. Otherwise, participants found rest opportunities sufficient.

The most common complaints in the virtual conditions were that participants had to look down to see the entire virtual monitor (likely due to the HMD's limited field of view), and the HMD was too heavy, leading to some neck strain. Some would have preferred the chair lowered.

Two of the twenty-eight virtual condition participants felt dizzy near the end of the experience, but felt fine after about five minutes. One  $V_{1-1}$  participant felt dizzy around the first Training break time and asked to have the HMD removed. He then drank some water, re-donned the HMD, and did not subsequently mention feeling dizzy.

## 6.6 Summary and conclusions

Evidence indicates that after users adapt to real-virtual discrepancy,  $V_d$  is no worse than  $V_{1-1}$  for throughput, error rate, and path variability. However,  $V_{1-1}$  and  $V_d$  participants were not as fast as  $R$  participants and I cannot claim that  $V_{1-1}$  and  $V_d$  are generally no worse than  $R$ . Whether training in a discrepant environment is useful depends largely on whether training in a VE at all is useful. Also, it is unknown what effect discrepant environments have on spatial knowledge acquisition. I suspect that Redirected Touching would work better for a higher-level task that does not require as much rapid motor control. For example, a button sequence may be easier to learn in a discrepant environment than tasks that require rapid motion from point to point on a virtual object.

If participants were given time to adapt to the discrepant environment before Training and to readapt to the real environment before Posttest,  $V_d$  would likely end up even closer to  $V_{1-1}$  in all performance metrics.

I believe that Redirected Touching continues to be worth exploring. Despite a few indications of degraded performance in discrepant environments, evidence suggests that after adaptation, users can perform tasks in a discrepant VE generally no worse than in a one-to-one VE.

## CHAPTER 7

# Conclusions and Future Work

### 7.1 Revisiting the thesis statement

**Part I:** *In a virtual environment, computer-controlled variable mappings between real and virtual objects can enable a single passive real object to provide haptic feedback for different instances and kinds of virtual objects.*

**Part II:** *By warping virtual space, one can incorporate a non-zero amount of real-virtual shape discrepancy without impacting task performance.*

I developed three techniques that demonstrated the truth of my thesis:

1. Redirected Passive Haptics (Chapter 2) moves the virtual world to map different instances of a virtual object onto a single real object. A proof of concept used Redirected Walking and an engaging task to map five virtual cylindrical pedestals onto a single real cylinder, enabling users to feel all five virtual pedestals as if distinct.
2. The Haptic Hand (Chapter 3) demonstrated how to move a virtual object to map different parts of a larger-than-haptic-surface virtual object onto the same haptic surface. I developed a technique that attaches a larger-than-hand virtual user interface to the non-dominant hand. Feasibility-study participants were able to interact tangibly and naturally with the whole interface through simple hand gestures that map the currently relevant part of the interface onto the non-dominant hand.
3. Redirected Touching (Chapters 4–6) warps virtual space to map many differently shaped virtual objects onto a single real object, and vice versa. I conducted two studies that suggest that users adapt rather quickly to warped spaces, and that after adaptation, users perform no worse with

discrepant virtual objects than with one-to-one virtual objects. Though training with either kind of virtual object was not as effective as real-world training, Redirected Touching shows promise for training and entertainment applications.

## **7.2 Future work**

There are many potential avenues for future work.

### **7.2.1 Training**

The studies I presented suggest that Redirected Touching may be useful for aircraft cockpit procedures training. However, that kind of training involves a higher-level task than was used in my studies. Pilots need to learn sequences of buttons and switches for various cockpit procedures, as well as the relative locations of those buttons and switches. It is not known whether the process of adapting to a warped space will interfere with learning, or whether there is any effect on spatial knowledge acquisition. Future work should investigate whether participants can learn a cockpit procedure as well with Redirected Touching as with traditional training methods.

In my studies, participants adapted quickly to warped spaces, after which task performance was generally no worse than in one-to-one space. For a training scenario, it might be best to give users time to adapt prior to beginning training, so that they are not focusing on any strange feelings they experience instead of the training task. Users should also be given time to readapt to the real world after training. Depending on the real-virtual discrepancy magnitude, these adaptation phases could be as simple as tapping a warped virtual object or real object for about a minute. Also, foreknowledge of the discrepancy may reduce adaptation time.

Redirected Touching may also apply to training in other domains. Recent work extending Redirected Touching for surgical training has proposed mapping different virtual patients onto a single physical anatomical model [Spillmann et al., 2013].

### **7.2.2 Discrepancy detectability**

I informally investigated how well users can detect real-virtual object discrepancies in warped virtual spaces. Formal psychophysical studies to determine detection thresholds may better bound

the classes of shapes that can be warped and used effectively. Knowing the maximum amount of real-virtual hand position and orientation discrepancy users cannot detect may govern how much real-virtual object discrepancy can be introduced.

### **7.2.3 Space-warping**

#### **7.2.3.1 Automatic geometry determination and correspondence generation**

In my Redirected Touching studies, I predetermined real-virtual correspondences to pass to the space-warping algorithm. A useful extension would be to scan a real object, automatically determine its geometry, and automatically map into onto a virtual object. Object scanning techniques using depth cameras and other technologies exist. By appropriately remeshing the resulting geometry, a future system could automatically map real and virtual objects onto each other.

#### **7.2.3.2 Multiple objects**

Imagine that in a cockpit training application, a foam board is used for haptic feedback and the virtual cockpit is shaped differently enough that Redirected Touching is used to map it onto the real board. Now suppose that this training application also requires a thrust lever. The VE designer decides to place a mock-up of the thrust lever next to the user. The mock-up and the virtual thrust lever are mapped one-to-one. The environment therefore now has a discrepant virtual object and a one-to-one virtual object; the user must be able to touch both.

A warp field is computed for the discrepant object. If the user now touches the real thrust lever, the virtual hand will not touch the virtual thrust lever; the virtual hand's pose is governed by the warp field generated for the cockpit presented on the foam board. To address this problem, the virtual hand pose can be weighted: when the real hand is near the discrepant object, the virtual hand pose is determined by the warp field; when the real hand is near the one-to-one object, the virtual hand pose is the same as the real hand pose. In between, the virtual hand pose is determined by interpolating between the warped and one-to-one poses. Any suitable interpolation method can be used (e.g., linear interpolation for position, and spherical linear interpolation [Shoemake, 1985] for orientation).

I used this technique in the training study (Chapter 6). During target rings, the warped space was computed using only the real and virtual monitor geometries and a distant boundary beyond which

space was not warped; the table was not used. Consequently, when touching the real table straight on, the virtual finger would be shown at an angle. So that participants would not notice this discrepancy during breaks, I changed the warping behavior between target rings. The hand pose was interpolated between the warped and one-to-one spaces: the closer the finger to the monitor, the more warped the space, and the closer to the table edge, the more one-to-one the space.

Large VEs can have many passive-haptic props: walls, floors, bookshelves, balls, etc. There may be more than one discrepant object. More than one warp field can be computed, and the virtual hand pose can again be determined via a weighted interpolation based on the distance from each object.

### **7.2.3.3 Moving rigid objects**

For rigid objects, warp fields need not be generated whenever an object moves. The warp field can be precomputed, and a rigid transformation can be applied to the warp field to follow the object as it moves. For example, suppose there were a physical cube with a flat front face appearing as a curved face in the virtual environment. When the user's real finger moves along the flat surface, the virtual finger moves along a curved virtual surface. If the user rotates the cube, then the warp field rotates with it, now instead applying to a different region of space. This idea can be used with multiple objects as described in Section 7.2.3.2.

### **7.2.3.4 Whole-hand interaction**

My Redirected Touching implementations tracked only the index finger. Whole-hand interaction with discrepant virtual objects may lead to more compelling results. Also, adding a virtual arm model may make Redirected Touching more effective.

Recent extensions of Redirected Touching use a video see-through system to warp the imagery of a user's own real hand while she interacts with warped virtual objects [Ban et al., 2012]. Seeing one's own hand instead of a virtual hand will likely make Redirected Touching even more compelling.

### **7.2.3.5 Adding active haptics**

For some real-virtual object pairs, Redirected Touching may be insufficient. For example, a smooth real object would not believably represent a virtual object with a crease in it. Augmenting Redirected Touching with active haptics may enhance the experience. If a user moves her hand over

the virtual crease, she could be presented a vibration on her fingertip. This vibration coupled with the visual crease could induce the illusion that the smooth object she feels has a crease.

### 7.3 Implementing Redirected Touching with off-the-shelf hardware

Recent technology innovations such as the Nintendo® Wii™, Microsoft® Kinect®, and Sony® PlayStation® Move have reduced costs for virtual reality interfaces. Low-cost hardware is enabling VE interaction and techniques to become more widely available. Inexpensive head-mounted displays like the Oculus VR™ Oculus Rift [Oculus VR, 2013] and hand-tracking devices like the Leap Motion™ Controller [Leap Motion, 2012] will enable Redirected Touching to be used in consumer entertainment applications. Using a Leap Motion™ Controller development kit, I successfully reimplemented Redirected Touching [Kohli, 2013]. Figure 7.1 shows a user touching a flat real board that provides haptic feedback for differently shaped virtual objects. This proof of concept shows that Redirected Touching could be integrated into games so that users not only see a VE, but feel it too.

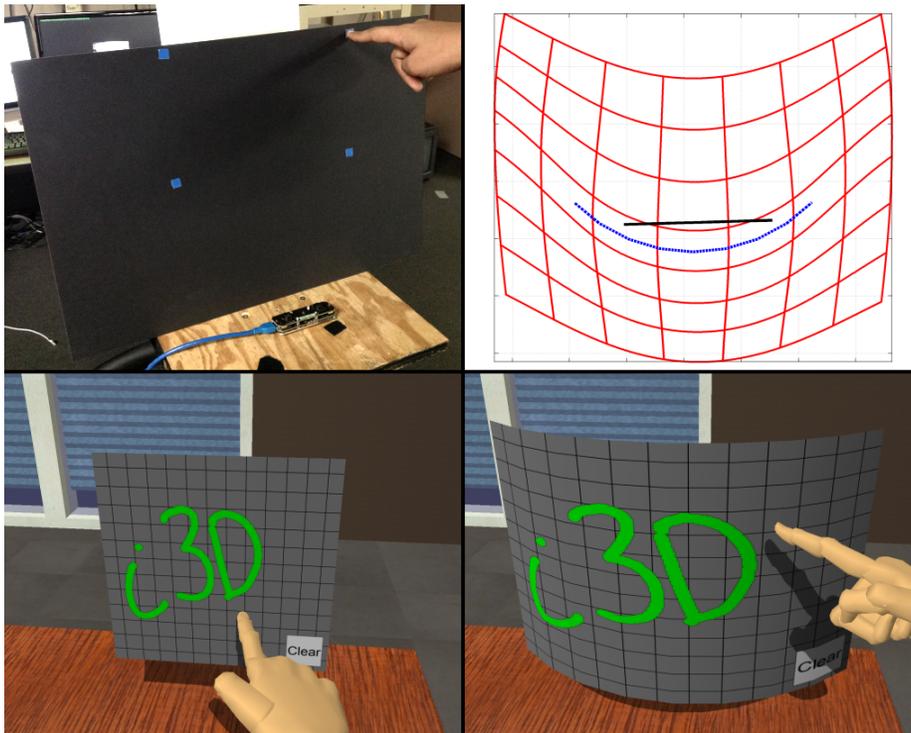


Figure 7.1: Upper left: a user's hand is tracked by the Leap Motion™ Controller while he touches the corner of a physical object onto which virtual objects will map; Upper right: horizontal slice of warped space, viewed from above (real object: black; virtual: blue); Bottom half: two differently shaped virtual objects mapped onto the same real object

## APPENDIX A

# Questionnaires

### A.1 The Haptic Hand

#### Debrief Session

*(to be completed after the experiment)*

#### The Haptic Hand: Providing User Interface Feedback with the Non-Dominant Hand in Virtual Environments

Debrief done by: \_\_\_\_\_

#### Tape Information:

- Audio filename \_\_\_\_\_
- Start Counter \_\_\_\_\_ Stop Counter \_\_\_\_\_

Questions	Comments
Do your arms or hands feel fatigued?	
What did you think about touching your non-dominant hand while manipulating widgets on the interface panel?	

<p>Do you feel that you were able to manipulate widgets on the user interface with as much precision as you would have liked?</p>	
<p>Did you have any difficulty getting rid of or bringing back the user interface panel?</p>	
<p>Did you have any difficulty moving and rotating the user interface panel or other objects in the environment with your non-dominant hand?</p>	
<p>Any comments on the difficulty of the tasks?</p> <p>Did the UI contribute to or alleviate that difficulty?</p> <p>Suggestions to make the tasks easier?</p>	

<p>Any comments on the display, or other used equipment? Anything uncomfortable, etc?</p>	
<p>Comments on procedure? -too long? -tiring?</p>	

**Exit-Questionnaire (& text for web form)**  
(to be completed after the experiment)

**The Haptic Hand: Providing User Interface Feedback with the Non-Dominant Hand in Virtual Environments**

*Please check or fill in your answers as necessary*

The following questions (1-12) relate to your Training and VR experience:

1. How easy or difficult was it for you to activate widgets on the user interface panel with your non-dominant hand? Rate this on a scale from one (1) to seven (7) where 1 represents you having no problem at all and 7 represents you having a really difficult time.

<i>I found activating widgets to be...</i>	<b>Please check your answer</b>
1. extremely easy	1.
2. easy	2.
3. moderately easy	3.
4. neutral	4.
5. moderately difficult	5.
6. difficult	6.
7. extremely difficult	7.

2. How easy or difficult was it for you to manipulate buttons with your dominant hand? Rate this on a scale from one (1) to seven (7) where 1 represents you having no problem at all and 7 represents you having a really difficult time.

<i>I found manipulating buttons to be...</i>	<b>Please check your answer</b>
1. extremely easy	1.
2. easy	2.
3. moderately easy	3.
4. neutral	4.
5. moderately difficult	5.
6. difficult	6.
7. extremely difficult	7.

3. How easy or difficult was it for you to manipulate sliders with your dominant hand? Rate this on a scale from one (1) to seven (7) where 1 represents you having no problem at all and 7 represents you having a really difficult time.

<i>Overall I found manipulating sliders to be...</i>	<b>Please check your answer</b>
1. extremely easy	1.
2. easy	2.
3. moderately easy	3.
4. neutral	4.
5. moderately difficult	5.
6. difficult	6.
7. extremely hard	7.

4. How easy or difficult was it for you to get rid of or bring back the user interface panel? Rate this on a scale from one (1) to seven (7) where 1 represents you having no problem at all and 7 represents you having a really difficult time.

<i>Overall I found getting rid of/bringing back the user interface panel...</i>	<b>Please check your answer</b>
1. extremely easy	1.
2. easy	2.
3. moderately easy	3.
4. neutral	4.
5. moderately difficult	5.
6. difficult	6.
7. extremely difficult	7.

5. How easy or difficult was it for you to move and rotate the user interface panel or other objects with your non-dominant hand? Rate this on a scale from one (1) to seven (7) where 1 represents you having no problem at all and 7 represents you having a really difficult time.

<i>Overall I found moving and rotating the user interface panel and other objects...</i>	<b>Please check your answer</b>
1. extremely easy	1.
2. easy	2.
3. moderately easy	3.
4. neutral	4.
5. moderately difficult	5.
6. difficult	6.
7. extremely hard	7.

The following questions relate (6-12) to you and your experience with computers:

6. Age and Gender

Age\_\_\_\_\_

I am ...	<b>Please check your answer</b>
1. Male	1.
2. Female	2.

7. What is your University status?

<i>I am a...</i>	<b>Please check your answer</b>
1. Undergraduate student	1.
2. Graduate student	2.
3. Research Associate	3.
4. Staff members - systems/technical staff	4.
5. Faculty	5.
6. Administrative staff	6.
7. Other (please write below)	7.

8. How coordinated do you think you are (arm/leg motion)?

<i>I am coordinated...</i>	<b>Please check your answer</b>
1. not at all	1.
2.	2.
3.	3.
4.	4.
5.	5.
6.	6.
7. all the time (including when I dance)	7.

9. Have you used virtual reality before?

<i>Before today I have used virtual reality...</i>	<b>Please check your answer</b>
1. never	1.
2.	2.
3.	3.
4.	4.
5.	5.
6.	6.
7. every day at school, work, or in gaming	7.

10. To what extent do you use a computer in your daily activities?

<i>I use computers...</i>	<b>Please check your answer</b>
1. never	1.
2.	2.
3.	3.
4.	4.
5.	5.
6.	6.
7. every day	7.

11. Over the past two years, what is the *most* you have played video games in a single week?

<i>I have played computer games ...</i>	<b>Please check your answer</b>
1. Less than 1 hour	1.
2. 1 – 2 hours	2.
3. 2 – 5 hours	3.
4. 5 – 10 hours	4.
5. 10 – 20 hours	5.
6. 20 – 40 hours	6.
7. 40 or more hours	7.

12. To what extent do you play video or arcade games (Playstation, Nintendo, XBox, Gameboy or other special devices)?

<i>I play video or arcade games ...</i>	<b>Please check your answer</b>
1. never	1.
2.	2.
3.	3.
4.	4.
5.	5.
6.	6.
7. every day	7.

## A.2 Redirected Touching: Task Performance

### Post-Questionnaire

Please enter your Participant ID code.

What is your age?

Please enter your gender.

Male

Female

Prefer not to answer

What is your ethnic background? Check one.

Hispanic or Latino

Not Hispanic or Latino

Prefer not to answer

What is your racial background? Check one.

- American Indian or Alaska native
- Asian
- Native Hawaiian
- Black or African American
- White
- More than one race
- Other
- Prefer not to answer

**Please select the options that most closely represent your video game and virtual reality experience:**

	Never / No experience before today	Once a year / At least one experience before today	Once a month	Once a week	Daily
How often do you play video games?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
How often do you use the Nintendo Wii, Kinect for Xbox 360, or PlayStation Move?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
How often do you use virtual reality?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

**How tired is your right arm?**

Not tired at all		Somewhat tired		Very tired
1	2	3	4	5
<input type="radio"/>				

**What cues did you use to make judgments about the virtual objects?**

**Do you have any comments about the study, equipment, procedure, or anything else (uncomfortable display, uncomfortable finger ring, task length, etc.)? Is there anything you would change?**

### A.3 Redirected Touching: Training and Adaptation

#### Post-Questionnaire

**Condition (filled in by experimenter):**

- R
- V<sub>00</sub>
- V<sub>D</sub>

**Please enter your Participant ID (filled in by experimenter):**

**Please enter your age:**

**Please select your gender:**

- Male
- Female
- Prefer not to answer

**What is your ethnic background? Select one.**

- Hispanic or Latino
- Not Hispanic or Latino
- Prefer not to answer

**What is your racial background? Select one.**

- American Indian or Alaska native
- Asian
- Native Hawaiian
- Black or African American
- White
- More than one race
- Other
- Prefer not to answer

**Please select the options that most closely represent your video game and virtual reality experience:**

	Never / No experience before today	Once a year / At least one experience before today	Once a month	Once a week	Daily
How often do you play video games?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
How often do you use the Nintendo Wii, Kinect for Xbox 360, or PlayStation Move?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
How often do you use virtual reality?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

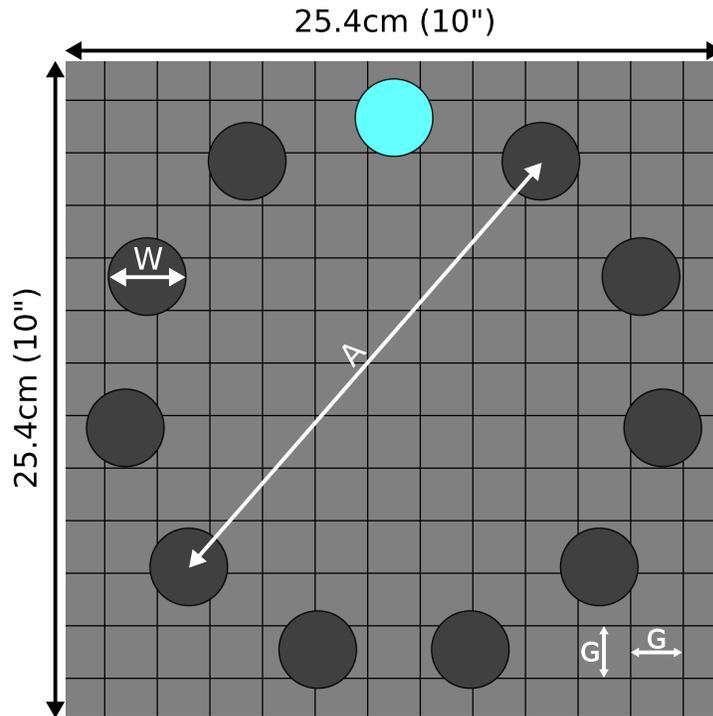
**During the experiment, did you experience any fatigue?**

	Not tired at all 1	2	Somewhat tired 3	4	Very tired 5
Right hand	<input type="radio"/>				
Right arm	<input type="radio"/>				
Right shoulder	<input type="radio"/>				
Neck	<input type="radio"/>				
Head	<input type="radio"/>				
Other (please specify in comments below)	<input type="radio"/>				

**Do you have any comments about the study, equipment, procedure, or anything else (uncomfortable equipment, task length, etc.)? Is there anything you would change?**

## APPENDIX B

# Target Ring Details



### Dimensions:

11 targets, equally spaced around a 21cm-diameter circle  
W (*target width*): 10mm, 15mm, 20mm, 25mm, 30mm, 35mm  
A (*amplitude or target distance*): ~20.8cm  
G (*distance between gridlines*): 2cm  
Gridline width: 0.5mm

### Colors (R, G, B):

Background: (128, 128, 128)  
Active target: (100, 255, 255)  
Inactive target: (64, 64, 64)  
Gridlines: (0, 0, 0)

## BIBLIOGRAPHY

- Andre, T. S., Bennett, W. R., Castillo, A. R., McClain, D. P., and Purtee, M. D. (2004). Generalized operations simulation environment for aircraft maintenance training. In *RTO-MP-HFM-101: Advanced Technologies for Military Training*.
- Balakrishnan, R. and Kurtenbach, G. (1999). Exploring bimanual camera control and object manipulation in 3d graphics interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 56–62.
- Ban, Y., Kajinami, T., Narumi, T., Tanikawa, T., and Hirose, M. (2012). Modifying an identified angle of edged shapes using pseudo-haptic effects. In *Proceedings of the International Conference on Haptics: Perception, Devices, Mobility, and Communication*, pages 25–36.
- Baseops.Net (2012). JSUPT phase i - academics and preflight training. [http://www.baseops.net/militarypilot/phase1\\_upt.html](http://www.baseops.net/militarypilot/phase1_upt.html).
- Bookstein, F. (1989). Principal warps: thin-plate splines and the decomposition of deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(6):567–585.
- Burns, E. (2007). *MACBETH: Management of Avatar Conflict By Employment of a Technique Hybrid*. PhD thesis, Department of Computer Science, The University of North Carolina at Chapel Hill.
- Burns, E., Razzaque, S., Whitton, M. C., and Jr., F. P. B. (2007). MACBETH: Management of avatar conflict by employment of a technique hybrid. *IJVR*, 6(2):11–20.
- Buxton, W., Hill, R., and Rowley, P. (1985). Issues and techniques in touch-sensitive tablet input. In *Proceedings of ACM SIGGRAPH*, pages 215–224.
- Buxton, W. and Myers, B. A. (1986). A study in two-handed input. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 321–326.
- Coquillart, S. and Wesche, G. (1999). The virtual palette and the virtual remote control panel: A device and an interaction paradigm for the Responsive Workbench™. In *Proceedings of IEEE Virtual Reality*, pages 213–216.
- Cutler, L. D., Fröhlich, B., and Hanrahan, P. (1997). Two-handed direct manipulation on the Responsive Workbench. In *Proceedings of the ACM Symposium on Interactive 3D Graphics*, pages 107–114.
- Degani, A. and Wiener, E. L. (1993). Cockpit checklists: Concepts, design, and use. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 35(2):345–359.
- Department of Defense (1999). Design criteria standard, human engineering. Technical Report MIL-STD-1472F.
- Dezfuli, N., Khalilbeigi, M., Huber, J., Müller, F., and Mühlhäuser, M. (2012). PalmRC: imaginary palm-based remote control for eyes-free television interaction. In *Proceedings of the 10th European Conference on Interactive TV and Video*, pages 27–34.
- Drewes, H. (2010). Only one Fitts' law formula please! In *CHI Extended Abstracts on Human Factors in Computing Systems*, pages 2813–2822.

- Duh, H. B.-L., Parker, D. E., Philips, J. O., and Furness, T. A. (Spring 2004). “Conflicting” motion cues to the visual and vestibular self- motion systems around 0.06 Hz evoke simulator sickness. *Human Factors*, 46(1):142–153.
- Eberly, D. (2011). Thin-plate splines. <http://www.geometrictools.com/Documentation/ThinPlateSplines.pdf>.
- Fitts, P. M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology*, 47(6):381–391.
- Georgopoulos, A. P., Kalaska, J. F., and Massey, J. T. (1981). Spatial trajectories and reaction times of aimed movements: effects of practice, uncertainty, and change in target location. *Journal of neurophysiology*, 46(4):725–743.
- Gibson, J. J. (1933). Adaptation, after-effect and contrast in the perception of curved lines. *Journal of Experimental Psychology*, 16:1–31.
- Guiard, Y. (1987). Asymmetric division of labor in human skilled bimanual action: the kinematic chain as a model. *J Mot Behav*, 19(4):486–517.
- Guiard, Y. (2009). The problem of consistency in the design of Fitts’ law experiments: consider either target distance and width or movement form and scale. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1809–1818.
- Gustafson, S., Holz, C., and Baudisch, P. (2011). Imaginary phone: learning imaginary interfaces by transferring spatial memory from a familiar device. In *ACM Symposium on User Interface Software and Technology*, pages 283–292.
- Harrison, C., Tan, D., and Morris, D. (2010). Skinput: appropriating the body as an input surface. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 453–462.
- Helbig, H. B. and Ernst, M. O. (2007). Optimal integration of shape information from vision and touch. *Experimental Brain Research*, 179:595–606.
- Hinckley, K., Pausch, R., and Proffitt, D. (1997a). Attention and visual feedback: the bimanual frame of reference. In *Proceedings of the ACM Symposium on Interactive 3D Graphics*, pages 121–126.
- Hinckley, K., Pausch, R., Proffitt, D., and Kassell, N. F. (1998). Two-handed virtual manipulation. *ACM Trans. Comput.-Hum. Interact.*, 5(3):260–302.
- Hinckley, K., Pausch, R., Proffitt, D., Patten, J., and Kassell, N. (1997b). Cooperative bimanual action. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 27–34.
- Insko, B. E. (2001). *Passive Haptics Significantly Enhances Virtual Environments*. PhD thesis, Department of Computer Science, The University of North Carolina at Chapel Hill.
- International Organization for Standardization (2007). *ISO 9241-9:2000, Ergonomic requirements for office work with visual display terminals (VDTs) - Part 9: Requirements for non-keyboard input devices*. Multiple. Distributed through American National Standards Institute (ANSI).

- Jones, L. A. and Lederman, S. J. (2006). *Human Hand Function*. Oxford University Press, USA.
- Keele, S. W. and Posner, M. I. (1968). Processing of visual feedback in rapid movements. *J Exp Psychol*, 77(1):155–158.
- Kitahara, I., Nakahara, M., and Ohta, Y. (2010). Sensory properties in fusion of visual/haptic stimuli using mixed reality. In Zadeh, M. H., editor, *Advances in Haptics*. InTech.
- Kohli, L. (2009). Exploiting perceptual illusions to enhance passive haptics. In *Proceedings of the IEEE VR Workshop on Perceptual Illusions in Virtual Environments*, pages 22–24.
- Kohli, L. (2010). Redirected touching: Warping space to remap passive haptics. In *Proceedings of the IEEE Symposium on 3D User Interfaces*, pages 129–130.
- Kohli, L. (2013). Warping virtual space for low-cost haptic feedback. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, page 195.
- Kohli, L., Burns, E., Miller, D., and Fuchs, H. (2005). Combining passive haptics with redirected walking. In *Proceedings of the International Conference on Artificial Reality and Telexistence*, pages 253–254.
- Kohli, L. and Whitton, M. (2005). The haptic hand: Providing user interface feedback with the non-dominant hand in virtual environments. In *Proceedings of Graphics Interface*, pages 1–8.
- Kohli, L., Whitton, M. C., and Brooks, Jr., F. P. (2012). Redirected touching: The effect of warping space on task performance. In *Proceedings of the IEEE Symposium on 3D User Interfaces*, pages 105–112.
- Kohli, L., Whitton, M. C., and Brooks, Jr., F. P. (2013). Redirected touching: Training and adaptation in warped virtual spaces. In *Proceedings of the IEEE Symposium on 3D User Interfaces*, pages 79–86.
- Kuhl, S. A., Thompson, W. B., and Creem-Regehr, S. H. (2008). HMD calibration and its effects on distance judgments. In *Proceedings of the ACM SIGGRAPH Symposium on Applied Perception in Graphics and Visualization*, pages 15–22.
- Leap Motion (2012). <http://leapmotion.com>.
- Lécuyer, A., Coquillart, S., Kheddar, A., Richard, P., and Coiffet, P. (2000). Pseudo-haptic feedback: can isometric input devices simulate force feedback? In *Proceedings of IEEE Virtual Reality*, pages 83–90.
- Lindeman, R. W. (1999). *Bimanual Interaction, Passive-Haptic Feedback, 3D Widget Representation, and Simulated Surface Constraints for Interaction in Immersive Virtual Environments*. PhD thesis, Department of Electrical Engineering and Computer Science, The George Washington University.
- Lindeman, R. W., Sibert, J. L., and Hahn, J. K. (1999). Towards usable VR: An empirical study of user interfaces for immersive virtual environments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 64–71.
- Lindeman, R. W., Sibert, J. L., and Templeman, J. N. (2001). The effect of 3D widget representation and simulated surface constraints on interaction in virtual environments. In *Proceedings of IEEE Virtual Reality*, pages 141–148.

- Lockheed Martin (2012). Multi-function training aid. <http://www.lockheedmartin.com/us/products/MFTA.html>.
- MacKenzie, I. S. (1992). Fitts' law as a research and design tool in human-computer interaction. *Hum.-Comput. Interact.*, 7(1):91–139.
- MacKenzie, I. S. and Isokoski, P. (2008). Fitts' throughput and the speed-accuracy tradeoff. In *Proceeding of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1633–1636.
- McNeely, W. A. (1993). Robotic graphics: a new approach to force feedback for virtual reality. In *IEEE Virtual Reality Annual International Symposium*, pages 336–341.
- Mine, M. R., Brooks, Jr., F. P., and Sequin, C. H. (1997). Moving objects in space: exploiting proprioception in virtual-environment interaction. In *Proceedings of ACM SIGGRAPH*, pages 19–26.
- Mistry, P., Maes, P., and Chang, L. (2009). WUW - wear Ur world: a wearable gestural interface. In *CHI '09 Extended Abstracts on Human Factors in Computing Systems*, pages 4111–4116.
- Oculus VR (2013). <http://www.oculusvr.com/>.
- Pausch, R., Crea, T., and Conway, M. (1992). A literature survey for virtual environments: military flight simulator visual systems and simulator sickness. *Presence: Teleoper. Virtual Environ.*, 1(3):344–363.
- Peck, T. (2010). *Redirected Free Exploration with Distractors: A Large-Scale Real-Walking Locomotion Interface*. PhD thesis, Department of Computer Science, The University of North Carolina at Chapel Hill.
- Peck, T. C., Whitton, M. C., and Fuchs, H. (2008). Evaluation of reorientation techniques for walking in large virtual environments. In *Proceedings of IEEE Virtual Reality*, pages 121–127.
- PhaseSpace (2012). <http://www.phasespace.com/>.
- Poupyrev, I., Tomokazu, N., and Weghorst, S. (1998). Virtual notepad: Handwriting in immersive VR. In *Proceedings of the Virtual Reality Annual International Symposium*, pages 126–132.
- Prophet, W. W. and Boyd, H. A. (1970). Device-task fidelity and transfer of training: Aircraft cockpit procedures training. Technical Report Technical Report 70-10, HumRRO.
- Razzaque, S. (2005). *Redirected Walking*. PhD thesis, Department of Computer Science, The University of North Carolina at Chapel Hill.
- Rock, I. and Victor, J. (1964). Vision and touch: An experimentally created conflict between the two senses. *Science*, 143(3606):594–596.
- Saboo, R. (2011). *Atlas Diffeomorphisms via Object Models*. PhD thesis, Department of Computer Science, The University of North Carolina at Chapel Hill.
- Schiefele, J., Albert, O., van Lier, V., and Huschka, C. (1998). Simple force feedback for small virtual environments. In *SPIE*, volume 3367, pages 100–110.

- Schmalstieg, D., Encarnação, L. M., and Szalavári, Z. (1999). Using transparent props for interaction with the virtual table. In *Proceedings of the ACM Symposium on Interactive 3D Graphics*, pages 147–153.
- Sederberg, T. W. and Parry, S. R. (1986). Free-form deformation of solid geometric models. In *Proceedings of ACM SIGGRAPH*, pages 151–160.
- Shoemake, K. (1985). Animating rotation with quaternion curves. In *Proceedings of ACM SIGGRAPH*, pages 245–254.
- Soukoreff, R. W. and MacKenzie, I. S. (2004). Towards a standard for pointing device evaluation, perspectives on 27 years of fitts law research in HCI. *International Journal of Human-Computer Studies*, 61:751–789.
- Spillmann, J., Tuchschnid, S., and Harders, M. (2013). Adaptive space warping to enhance passive haptics in an arthroscopy surgical simulator. *IEEE Transactions on Visualization and Computer Graphics*, 19(4):626–633.
- Steinicke, F., Bruder, G., Jerald, J., Frenz, H., and Lappe, M. (2008a). Analyses of human sensitivity to redirected walking. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, pages 149–156.
- Steinicke, F., Bruder, G., Kohli, L., Jerald, J., and Hinrichs, K. (2008b). Taxonomy and implementation of redirection techniques for ubiquitous passive haptic feedback. In *Proceedings of the International Conference on Cyberworlds*, pages 217–223. IEEE Computer Society.
- sugru (2012). <http://sugru.com/>.
- Tachi, S., Maeda, T., Hirata, R., and Hoshino, H. (1994). A construction method of virtual haptic space. In *Proceedings of the International Conference on Artificial Reality and Telexistence*, pages 131–138.
- Taylor, II, R. M., Hudson, T. C., Seeger, A., Weber, H., Juliano, J., and Helser, A. T. (2001). VRPN: a device-independent, network-transparent VR peripheral system. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, pages 55–61.
- Teather, R. J., Natapov, D., and Jenkin, M. (2010). Evaluating haptic feedback in virtual environments using ISO 9241-9. In *Proceedings of IEEE Virtual Reality*, pages 307–308.
- Teather, R. J. and Stuerzlinger, W. (2011). Pointing at 3D targets in a stereo head-tracked virtual environment. In *Proceedings of the IEEE Symposium on 3D User Interfaces*, pages 87–94.
- Vincenzi, D. A., Wise, J. A., Mouloua, M., and Hancock, P. A. (2008). *Human Factors in Simulation and Training*. CRC Press.
- Wang, B.-S., Wang, X.-J., and Gong, L.-K. (2009). The construction of a williams design and randomization in cross-over clinical trials using SAS. *Journal of Statistical Software, Code Snippets*, 29(1):1–10.
- Wellek, S. (2010). *Testing Statistical Hypotheses of Equivalence and Noninferiority*. Chapman and Hall/CRC, 2nd edition.
- Williams, E. J. (1949). Experimental designs balanced for the estimation of residual effects of treatments. *Australian Journal of Scientific Research A Physical Sciences*, 2:149.

- Williams, G. W., Lawrence, K., and Weeks, R. (2004). Modeling and simulation technologies: Reconfigurable flight simulators in modeling and simulation. In *AIAA Modeling and Simulation Technologies*.
- Wobbrock, J. O., Shinohara, K., and Jansen, A. (2011). The effects of task dimensionality, endpoint deviation, throughput calculation, and experiment design on pointing measures and models. In *Proceeding of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1639–1648.
- Woodworth, R. S. (1899). The accuracy of voluntary movement. *The Psychological Review: Monograph Supplements*, 3(3):1–119.
- Wurpts, M. (2009). Poster: Updating an obsolete trainer using passive haptics and pressure sensors. In *Proceedings of the IEEE Symposium on 3D User Interfaces*, pages 155–156.
- Zelaznik, H. N., Hawkins, B., and Kisselburgh, L. (1983). Rapid visual feedback processing in single-aiming movements. *Journal of Motor Behavior*, 15(3):217–236.
- Zhai, S. (2004). Characterizing computer input with Fitts' law parameters—the information and non-information aspects of pointing. *International Journal of Human-Computer Studies*, 61(6):791–809.