Modeling Distal Pointing on Large Screens

The Influence of Target Depth

by

Izabelle Janzen

B.A.Sc. Honours Computer Science, McMaster University, 2014

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Science

in

THE FACULTY OF GRADUATE AND POSTDOCTORAL
STUDIES
(Computer Science)

The University of British Columbia
(Vancouver)

August 2016

© Izabelle Janzen, 2016

Abstract

Pointing is a fundamental task within many interactions in current computer applications. It is incorporated into everything from selecting buttons to dragging files or positioning objects in a virtual environment. Thus, understanding, modeling and predicting pointing performance is crucial to the design and evaluation of many computer interfaces. Fitts's Law (1954) is the basis for modeling human pointing performance in the international standard on pointing device evaluation (ISO 9241-400:2007). However, while it is extremely robust for many standard desktop applications, previous work by Shoemaker et al. (2012) has suggested that Fitts's Law may not be robust enough to accurately model pointing at more extreme levels of gain and has proposed alternatives to Fitts's Law based on earlier work by Welford (1968). This thesis extends preliminary research by Rajendran (2012) that further examined these alternatives to Fitts's Law for distal pointing. Distal pointing is common in virtual and augmented reality interfaces. We first reänalyze results reported by Rajendran using a variety of Welford-style models to explore the relationship between target depth and a parameter k that was first suggested by Kopper et al. (2010) but is inherent in Welford's model. We then present a new experiment that removes the confound of system latency from Rajendran's approach. Our analyses provide evidence that k varies monotonically (possibly linearly) with target depth, which further supports the claim by Shoemaker et al. that Welford-style two-part models are preferable to Fitts-style one-part models in some situations. Our analyses also challenge Kopper et al.'s suggestion that angular measures of task difficulty are superior to linear measures for pointing models. We close with a discussion of how our findings about the variation of k with target depth might be used in calibration procedures for virtual environments.

Preface

Much of this thesis builds upon previous research. This project was pitched to me by my adviser, Kellogg Booth, as a continuation of these projects and a chance to investigate some of the questions brought up in their "future work" recommendations. The questions included whether angular measures account for the deficiencies in Fitts's Law, whether latency was causing the variance observed in the k parameter, and whether the variance in k was linear with target depth or something more complex. I was given leeway to decide which of these questions to tackle and how to do that, but many of the project goals came from Kelly.

Most of the code developed as part of my research was for administering pointing experiments or analyzing the data recorded from them. This thesis reports on two main experiments. They are presented in Chapter 3 and Chapter 5. Chapter 3 provides re-analysis and interpretation of an experiment conducted by Vasanth Rajendran in 2012 for his master's thesis. Garth Shoemaker's experimental software was used as the basis for Vasanth Rajendran's original experiment but was modified by Vasanth to better support VR. However, much of Vasanth's code and analysis was lost when a hard drive was corrupted. Therefore, starting from the code written by Garth Shoemaker, I independently modified the code base to create an application that would duplicate Vasanth's experimental software as closely as possible. After replicating Vasanth's experimental results in a small pilot study, I continued to modify the code base to conduct a second experiment. Chapter 5 reports on my study to investigate whether the trends observed in Vasanth's original experiment are simply an artifact of system latency. I did most of the writing for this chapter. Aside from modifications to the experimental software, I developed computer vision software to process video captured during the experiment in order to

obtain pointing performance data without relying on computer logs that are potentially latency-plagued because they rely on real-time sensing. Documentation and a primer about how the computer vision software works is provided in Chapter 4, all of which I wrote.

Data analysis for both experiments was performed through R-Scripts written independently by me. My adviser, Kellogg Booth, helped me interpret the graphs and results from the experiments. After performing the small pilot study that replicated the general trends in Vasanth's data, I was able to reänalyze Vasanth's original data files using my R-Scripts (he had used SPSS). Through trial and error, I was able to reverse engineer the analysis that was reported in Vasanth's thesis. During this process, I discovered a number of problems with how the original analysis was done, particularly with the method for filtering outliers that affected some of the results described in the text of Vasanth's thesis. Because of this, I independently reänalyzed the data using methods that are more consistent with previous research.

My reänalysis of Vasanth's data was reported in the paper "Modeling the Impact of Depth on Pointing Performance" that was presented at the CHI 2016 conference [16]. Much of the text from the paper was re-used in Chapter 3, although it has been edited and reörganized. Kelly and I had roughly similar levels of contribution to the paper writing process, with me focusing more on the results and analysis sections while Kelly focused more on the framing and related work. Vasanth and Kelly designed the initial experiment and Vasanth did the data collection. His initial analysis was used as the basic structure for the paper. The introduction and literature review from that paper were expanded and fleshed out to become the basis for Chapter 1 and Chapter 2 in this thesis.

Ethics approval for the experiments reported in this thesis was provided by the UBC Behavioral Research Ethics Board under certificate number H11-01756 "Interacting With Large Displays V."

Funding for the research was generously provided by NSERC, the Natural Sciences and Engineering Research Council of Canada, under the Discovery Grant Program, and by GRAND, the Graphics, Animation and New Media Network of Centres of Excellence. Facilities and research infrastructure administered by ICICS, the Institute for Computing, Information and Cognitive Systems purchased with funds from the Canada Foundation for Innovation were used for the research.

Table of Contents

Abstract . . . . . **ii**

Preface . . . . . **iii**

Table of Contents . . . . . **v**

List of Tables . . . . . **viii**

List of Figures . . . . . **x**

Glossary . . . . . **xiv**

Acknowledgments . . . . . **xvi**

1 Introduction **1**

2 Related Work **5**

2.1 Fitts and Welford Models of Pointing Performance 5

2.2 Effective Width W_e and Effective Amplitude A_e 6

2.3 Throughput and Cross Study Comparisons 7

2.4 The Relationship Between Gain and Depth 9

2.5 Distal Pointing Interactions 10

2.6 The Kopper Model for Mid-air Pointing 11

2.7 Statistically Comparing Nested Pointing Models 11

2.8 Impact of Latency on Pointing Performance 12

2.9 Virtual Reality (VR) 13

2.10 Calibrating Virtual Environments 14

3 Target Depth and *k* in Computer-Mediated Pointing 16
3.1 Method 16
3.1.1 Participants 17
3.1.2 Apparatus & Materials 17
3.1.3 Procedure 21
3.2 Results 24
3.2.1 ANOVA for Movement Time 24
3.2.2 Regressions for One-Part and Two-Part Models 26
3.2.3 The Ratio k as a Function of D_V 28
3.3 Discussion 29
3.3.1 A Closer Look at the Data 30
3.3.2 One-Part and Two-Part Models for Pointing 31
3.3.3 The k Parameter and Target Depth 32
3.3.4 Angular Amplitude and Angular Width 33
3.4 Conclusions and Future Work 34
3.4.1 Welford Models vs. Fitts Models 34
3.4.2 Lack of Improvement Using Angular Models 34
3.4.3 Calibrating VR Systems Using k-values 35
3.4.4 The Effect of Binocular Depth on Pointing Performance . 36

4 Incorporating Computer Vision for Latency Free Analysis of Pointing Data 40
4.1 Introduction 40
4.2 Room Setup and Camera Placement 41
4.2.1 Synchronizing Video with Experiment Conditions 42
4.3 Designing a Vision System to Record Pointing Data 44
4.3.1 Finding Moving Objects - Cursor Position 45
4.3.2 Finding Stationary Objects - Target Position 48
4.3.3 Moving From Pixels to Real World - Offset 51
4.4 Conclusions and Recommendations 53

5 Real-World Pointing, Target Depth, and *k* 55
5.1 Method 57
5.1.1 Participants 58
5.1.2 Apparatus & Materials 58
5.1.3 Procedure 62
5.2 Results 65
5.2.1 Removal of Outliers 66
5.2.2 ANOVA for Movement Time *MT* 66
5.2.3 Comparing Pointing Models 67
5.2.4 Testing Angular Measures of Target Difficulty 69
5.2.5 Modeling *k* as a Function of Target Depth 70
5.3 Discussion 74
5.3.1 One-Part Versus Two-Part Models of Pointing Performance 77
5.3.2 Angular Versus Classic Measures of Target Difficulty . . . 78
5.3.3 The Relationship Between *k* and Target Depth 78
5.3.4 Differences in Task and *k* 81
5.4 Future Work 82
5.4.1 Isolating Factors That Affect *k* 82
5.4.2 Addressing VR Calibration 84
5.5 Conclusion 85

6 Where Are We? The Path Ahead 86
6.1 One-Part and Two-Part Models of Pointing 86
6.2 Measuring Target Difficulty 89
6.3 Variation of Shoemaker's *k* Parameter with Target Depth 90
6.4 Impact on Virtual Reality Calibration 91
6.5 Final Words 92

Bibliography 94

A Supporting Materials 101
A.1 Study Materials 101
A.2 Raw Data 101

List of Tables

Table 3.1 Significant ANOVA main effects and interactions for movement time (MT). No other interactions were found. All tests were adjusted with Greenhouse-Geisser. 25

Table 3.2 Modeling movement time (ms) using Fitts and Welford formulations (top) and Shannon-Fitts and Shannon-Welford formulations (middle) for actual width W and effective width W_e for each $D_V = D_S$ condition (no added virtual binocular disparity) and for all $D_V = D_S$ conditions combined. Fitts and Welford models computed using Angular A and W are also reported (bottom). The coefficients and adjusted R^2 for each model with the F-ratios, p-values and significance from an $F(1,9)$-test (Eq. 2.9 with $p_1 = 2$, $p_2 = 3$, and $n = 9$) in the last three columns compare nested Fitts and Welford (or Shannon-Fitts and Shannon-Welford) models. 27

Table 3.3 Linear regression indicating how k varies as binocular depth D_V (cm) changes for each of the four models. 28

Table 5.1 ANOVA results for the impact of depth (D), amplitude (A) and width (W) on movement time (MT). Statistically significant factors are bolded. 67

Table 5.2 Modeling movement time (ms) using Fitts and Welford formulations (top) and Shannon-Fitts and Shannon-Welford formulations (middle) for actual width W and effective width W_e. For each D condition and for all D conditions combined. The coefficients and adjusted R^2 for each model with the F-ratios, p-values and significance from an $F(1,9)$-test (Eq. 2.9 with $p_1 = 2$, $p_2 = 3$, and $n = 9$) in the last three columns compare nested Fitts and Welford (or Shannon-Fitts and Shannon-Welford) models. 68

Table 5.3 Quality of fit for regression analyses as determined by the R-squared values is shown for angular and classic linear measures of movement amplitude A and target width W, and for movement amplitude A and effective target width W_e. Bold cells show which model in a pair of columns has a better R^2. The left columns are for classic and right for angular. 69

Table 5.4 Regression modeling of how k varies as target depth D (cm) changes for each of the four pointing models. 71

Table 5.5 Regression modeling of how k varies as distance from the screen D (cm) changes for each of the four pointing models. The 165 condition has been removed as a potential outlier. 74

Table A.1 Processed raw data from experiment one. Outliers have already been filtered out and we present effective widths, amplitudes, average movement times and 95 percent confidence intervals for all conditions. 104

Table A.2 Processed movement time data from experiment two. Outliers have already been filtered out and we present effective widths, amplitudes, average movement times with low and high 95 percent confidence intervals for all conditions. 106

List of Figures

Figure 2.1 Computing effective width W_e using the standard deviation *sigma* of the observed pointing or tapping accuracy (after MacKenzie [26]). 7

Figure 2.2 A simple diagram of the effect of depth on distal pointing. Note that the same physical movement intersects further horizontally away from screen center when depth is far. This creates an effect similar to gain with target depth. 9

Figure 2.3 Basic distal pointing interaction implemented with Nintendo Wii Remote to select buttons in a user interface. 10

Figure 2.4 Shadow reaching is a proposed distal pointing interaction using the shadow as a metaphor for changing pointing gain by moving back and forth through the room. 10

Figure 2.5 Oculus rift and other VR systems have commonly been applied to gaming and other methods of exploring VR systems. In this example a treadmill is placed under the user to allow them to traverse by running in place. 13

Figure 2.6 A secondary example of a VR system in practice allows flying and exploration like a bird. A specially designed apparatus helps aide this effect. 13

Figure 2.7 Image displaying an application using dynamic adjustment of 3D parameters. On the left, a farther object has its stereo parameters made more extreme to highlight the depth effect and improve depth estimation. 15

Figure 3.1 Apparatus: (a) Large screen stereoscopic display, (b) Hand-held pointer, (c) Head-tracking gear, and (d) Wii Remote for click events. 18
Figure 3.2 Placement of the targets and the participant in an experimental trial. The virtual plane of the targets is green; the physical plane of the screen is black. 19
Figure 3.3 Computing the cursor position from the head and hand-held pointer positions. 20
Figure 3.4 An illustration of the participant performing the pointing task for $D_V = D_S$. Targets are in the plane of the screen. 21
Figure 3.5 An illustration of the participant performing the pointing task for $D_S > D_V$. Targets are perceived in front of the screen. . . . 22
Figure 3.6 R^2 values for each target depth using W (top) and W_e (bottom). 26
Figure 3.7 k values calculated using the Welford formulation (a) for A and W (b) for A and W_e. 29
Figure 3.8 Scatterplot of MT vs. effective ID. Points are identical in the right and left plots. Points are connected in two different ways to illustrate the separability of A and W: Lines on the left connect points representing tasks with the same movement amplitude A; lines on the right connect points representing tasks with the same target width W. 38
Figure 3.9 The three k-lines for $D_S = 110$ (red / squares), 220 (green / triangles) and 330 (purple / circles), along with the k-line for the non-VR $D_V = D_S$ conditions (blue / diamonds). To calibrate binocular depth using k values, the desired binocular depth (A) determines a k-value (B) on the blue $D_V = D_S$ line. That same k-value on the red $D_S = 110$ line (C) determines D'_V to be the corresponding binocular depth (D) that the software should use to insure the desired pointing performance if the screen is 110cm from the viewer. 39

Figure 4.1 Example view of what is seen by the camera placed behind the screen. The green bars are the targets used in the pointing experiment. The blue box is used for camera calibration. . . . 43
Figure 4.2 Starting image for finding the pointer and targets. 46
Figure 4.3 Example Result of the output from thresholding for the pointer. White spots are within the threshold and thus likely within the pointer. 47
Figure 4.4 Example of a median filter operation. Image taken from http://tinyurl.com/j26e7cf 47
Figure 4.5 View of the average thresholded result, edges are noisy, but the rough shape is there. 49
Figure 4.6 Example result for the target position after thresholding and processing the averaged passing pixels. 50
Figure 4.7 Example contours of the targets overlayed over top of the original example image. Circles are positioned at the corners. . . . 50
Figure 4.8 Example of the calibration procedure through grid wave. A regular grid of known size is pictured from various angles, machine learning finds the intrinsic distortions of the camera. . . 52

Figure 5.1 Apparatus: Pointer used for interaction and camera used for data recording. 59
Figure 5.2 Image of the projection room behind the screen. Camera is placed on tripod at fixed location to record experimental data. Two projectors were used to display the experimental stimulus. 59
Figure 5.3 Experimental stimuli. Participant stands at a fixed target depth along the orange line. Specific conditions are marked with crosses. The experimental software displays two green targets on the screen. The user smoothly moves the laser pointer between the two targets. 61
Figure 5.4 Visual comparison of different models for the relationship between k and D. Data presented for regular width models of pointing performance. 72

Figure 5.5 Visual comparison of different models for the relationship between k and D. Data presented for effective width models of pointing performance. 73
Figure 5.6 Visual comparison of different models for the relationship between k and D. Data presented for regular width models of pointing performance. The 165 condition has been removed as a potential outlier. 75
Figure 5.7 Visual comparison of different models for the relationship between k and D. Data presented for effective width models of pointing performance. The 165 condition has been removed as a potential outlier. 76

Figure 6.1 The three k-lines for $D_S = 110$ (red / squares), 220 (green / triangles) and 330 (purple / circles), along with the k-line for the non-VR $D_V = D_S$ conditions (blue / diamonds). To calibrate binocular depth using k values, the desired binocular depth (A) determines a k-value (B) on the blue $D_V = D_S$ line. That same k-value on the red $D_S = 110$ line (C) determines D'_V to be the corresponding binocular depth (D) that the software should use to ensure the desired pointing performance if the screen is 110cm from the viewer. 92

Figure A.1 Consent forms for experiment one which investigated the impact of depth on pointing performance. Data gathered and performed by Vasanth Rajendran. 107
Figure A.2 Consent forms for experiment two which investigated the impact whether the impact of depth on pointing performance is an artifact of system latency. 108
Figure A.3 Demographics questionaire used to gather qualitative data in both studies presented in this thesis. 109

Glossary

A	Movement Amplitude, Horizontal distance between consecutive targets in a pointing experiment.
ANOVA	Analysis of Variance, a set of statistical techniques to identify sources of variability between groups.
D	Target Depth, Distance from the user to the target in physical space.
D_S	Screen Depth, Distance from the user to the image of the target on the screen in physical space.
D_V	Virtual Depth, Distance from the user to the target as perceived (or intended to be perceived) in stereoscopic or binocular viewing, independent of the physical screen position.
HCI	Human Computer Interaction
k	Kopper's k, Referring to the parameter k noted by Shoemaker et al. in 2012 that captures the relative impact of movement amplitude and target width on pointing performance in a Welford-like two-part model.
MT	Movement Time, Time to complete a pointing action.
R^2	R-squared, a measure of how much variance in a model is accounted for by the predictors. Refers to the percentage of variance that is attributable to between groups differences.
TP	Throughput, , a measure of balanced overall pointing performance in bits per second. Takes both speed and accuracy into account.

VR Virtual Reality, Referring to virtual environments enhanced with stereoscopic 3D viewing to enhance depth.

W Target Width, Width of a target in a pointing experiment.

W_e Effective Width, a measure of target size adjusted to account for how accurate participants were.

Acknowledgments

I'd like to thank my academic adviser, Dr. Kellogg Booth, in particular for all his support and encouragement during my Master's degree. Thanks go to Vasanth Rajendran and Garth Shoemaker for providing me with access to their raw data and their experimental software to start my investigations.

I'd also like to thank my parents, Lorraine and Fred Janzen, for their support and understanding.

Special thanks go to Ron Rensink who was the second reader for the thesis. He provided extremely helpful and timely advice and suggestions that have improved the thesis in a number of ways.

Chapter 1

Introduction

Pointing is the act of touching or selecting an object by positioning one's hand, or some intermediary tool, over top of a desired object or in the general direction of the object if it is not close by. In human computer interaction pointing is incorporated as a sub-component of many basic tasks. These can vary from selecting buttons, highlighting text and moving files within a desktop to positioning boxes within a virtual world or game. Due to its ubiquity, it is important that we understand and can predict the behavior of the sensorimotor systems involved in pointing when we evaluate new pointing devices and techniques. Even small improvements and refinements to a pointing device or technique could add up to save a significant amount of time when interacting with the system as a whole.

Pointing devices such as the computer mouse are robust, ubiquitous and highly effective for desktop applications. However, these often break down when used in virtual environments or with screens that get very large. This often occurs in virtual reality systems or classroom interaction. This style of interaction is often called distal pointing, where users stand a good distance away from the screen and position the cursor with their hand as if using a laser pointer. While numerous pointing tools have been and are currently being developed for distal pointing, some researchers have noted problems properly evaluating and modeling the tools' performance using standard procedures [43]. This has led to attempts to expand and refine our models of human pointing performance to better evaluate these techniques [23]. This is a fundamental concern that we explore in this thesis. If we do

not have an accurate and robust measure of how a device performs, how can we say if it was effective or not?

We focus our work on investigating the impact of target depth on *mid-air pointing* techniques (also known as distal pointing) in which a person's hand moves freely, unconstrained by contact with any surface or object. This is a very common task in many training, simulation and entertainment activities where interaction with real or virtual objects is required. It is also a task performed every day in the real world, often with 2-D information displays such as in classrooms or lecture halls. Some virtual environments employ "virtual hand" techniques to grasp and manipulate targets at a distance, but we restricted our attention to pointing because it mimics the physical world. We sought to learn more about the fundamental act of pointing, both when targets are at a depth in the real world, and when targets appear to be at a depth in a virtual world. Target depth is either the physical depth or the virtual depth, depending on the situation.

Fitts's Law [10] is often used to measure pointing performance by modeling movement time as a function of target size and distance moved. Kopper at al. [23] recommended a variant of Fitts's Law that uses angular measures of target size and movement distance instead of the classic linear measures normally used with Fitts's Law. Shoemaker et al. [40] showed that a classic measures version of Kopper et al.'s model is just a special case of Welford's two-part model of pointing [53]. The constant k that Kopper et al. introduced is actually the ratio of two coefficients in the much earlier Welford model. Welford's model is similar to Fitts's Law, with the change that it separates amplitude and width into their own terms with separate impacts. Shoemaker et al. also showed that k varies monotonically and possibly linearly with gain.

In two experiments, we looked at how k varies with target depth (distance to the target) and we reëxamined Kopper et al.'s claim that angular measures are better than linear measures for assessing mid-air pointing performance. Furthermore, we tested a variety of different possible models for pointing and suggested ways to make the most common models more robust for different pointing techniques and gain.

The first experiment had previously been conducted to investigate whether virtual depth and physical depth to the screen have similar impacts. Reänalyzing the

data from that experiment, we showed that for normal viewing with real-world physical depth cues, target depth affects pointing performance. In binocular stereo, where perceived or virtual target depth can differ from physical target depth, we found that both virtual and physical target depths may affect pointing performance. We did not find evidence that angular measures are better than linear measures, but we did find that two-part models are better than one-part models in some but not all conditions. The Kopper-Shoemaker k factor appeared to vary monotonically with gain, but we couldn't conclude much about the exact type of relationship (e.g. linear versus logarithmic) from the limited dataset we had.

Our second experiment investigated whether or not the effects observed for target depth were merely an artifact of latency from the computer mediation in the experimental apparatus. Latency is the delay between physical movement and output being displayed on the screen and has been shown to reduce pointing performance [31]. We were concerned that much of the effect of target depth might be simply latency becoming more problematic as a user moves further from the screen. This experiment provided further evidence that two-part models outperform one-part models in conditions where k deviates from one (which are common in distal pointing). We reinforced our conclusions that angular measures do not really improve our models of pointing performance. We also showed that even after removing computer mediation from the task, k still grew larger as target depth increased. Overall, k was smaller and had slower increases than in our first experiment, which we argue is likely caused by task differences or sampling biases. Our models of target depth's impact on k were refined from previous studies to reinforce that a linear approximation is robust and explanatory. We also noted there is some possible merit to a logarithmic model, though many conditions produced less favorable statistics than linear. This has implications on ideas we've presented to calibrate VR systems through the use of a ground truth k variation.

In the chapters that follow we present a review of relevant literature in the field (Chapter 2) and then present the results of the first experiment (Chapter 3). After that we describe a method for performing latency-free pointing evaluation using computer vision tchniques (Chapter 4), which can be read as a succinct primer on a number of the basic tools and techniques for incorporating computer vision into experimental designs, a practice that seems unfortunately to be underused. Lastly,

we present the methods, results and conclusions of a new experiment that replicates the first experiment without latency artifacts (Chapter 5). Further discussion of our results, potential impact on the field, and key avenues for further research are then discussed (Chapter 6).

Chapter 2

Related Work

Modeling pointing performance is an extremely common and classic research topic in HCI that has been studied since the dawn of the field in the 1960's [29]. Experiments to quantify and compare the pointing performance of different input methods were critical to the introduction of the first graphical user interfaces and computer mice [5, 6]. Given this massive breadth of research it would be entirely impractical to discuss all the papers and work that have been done on the subject. Therefore, this chapter provides a selective discussion of key work that is particularly relevant to distal pointing and the questions we investigate. We focus mainly on the last decade of research, which follows the pivotal work by Scott Mackenzie's lab in the 1990's and the early 2000's, which led to the current standard methods for evaluating pointing performance [3, 26–28, 41].

2.1 Fitts and Welford Models of Pointing Performance

The best known model of pointing performance is Fitts's Law (Eq. 2.1), formulated by Paul Fitts [10]. It was originally used to model "reciprocal tapping" where the time to move between successive taps was related to the distance moved (*amplitude*) and the size of the area being tapped (*target width*).

$$MT = a + b \log\left(\frac{A}{W}\right) \tag{2.1}$$

$$MT = a + b\log\left(\frac{A}{W} + 1\right) \tag{2.2}$$

Movement time (MT) depends only on the ratio of movement amplitude (A) and target width (W), but not their individual values. The logarithmic term is often called the *index of difficulty* (ID). Soukoreff and MacKenzie [41] argue for an information-theoretic interpretation, the Shannon-Fitts formulation (Eq. 2.2), with an additive constant in the ID term. This often produces better R^2 values than does the simpler version of Fitts's Law.

Welford later introduced a two-part model of pointing performance (Eq. 2.3) in which the one-part ID term is replaced by a linear combination of two logarithmic terms [53]. Shoemaker et al. [40] introduced a variation on Welford's model analogous to the Shannon-Fitts formulation that they called the Shannon-Welford model (Eq. 2.4).

$$MT = a + b_1\log(A) - b_2\log(W) \tag{2.3}$$

$$MT = a + b_1\log(A + W) - b_2\log(W) \tag{2.4}$$

The Fitts formulations are special cases of the Welford formulations. A parameter k can be defined (Eq. 2.5) that measures how closely the Welford models match the Fitts models.

$$k = b_2/b_1 \tag{2.5}$$

Eqs. 2.1 and 2.3 are *nested models* because setting $b = b_1 = b_2$ ($k = 1$) turns the 3-DOF (degree-of-freedom) Welford model into the 2-DOF Fitts model; similarly, the two Shannon models Eqs. 2.2 and 2.4 are also nested.

Throughout our discussion, we follow the authoritative and venerable advice of Strunk and White [42] and use Fitts's as the possessive case of the singular proper noun Fitts.

2.2 Effective Width W_e and Effective Amplitude A_e

Target width W is the physical size of the target being pointed at (or tapped on, in the original studies by Fitts). Crossman and Welford [53] compute the effective

width (W_e) of a target using the observed distribution of tap positions and then adjust the width of a target in their calculations to reflect what participants actually do, rather than what participants are expected to do. This post-hoc adjustment to target width maintains the information-theoretic analogy for Fitts and other models, which are for rapid, aimed movements (such as reciprocal tapping or pointing in 2-D or 3-D). It assumes a nominal and consistent error rate (traditionally 4%). When this condition is not met, further adjustment to the target width is introduced such that the error rate becomes 4%.

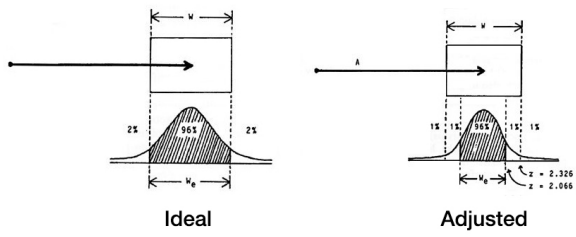

Figure 2.1: Computing effective width W_e using the standard deviation *sigma* of the observed pointing or tapping accuracy (after MacKenzie [26]).

Assuming a normal distribution with standard deviation σ of the taps, $P[Z \geq 2.066] = 0.02$, a width of 2×2.066 in z-units (4.133σ) around the mean has a probability of 96%. The probability of error (tapping outside this area) is 4%, so W_e can be calculated using Eq. 2.6.

$$W_e = 4.133\sigma \tag{2.6}$$

Whenever W_e is used, we also use effective amplitude, A_e, which is simply the mean of the observed amplitudes. Further background on the four pointing models, effective width, and the F-test used to compare nested models, is provided by Shoemaker et al. [40], an essential reference for the work discussed in this thesis.

2.3 Throughput and Cross Study Comparisons

One of the great benefits of Fitts's Law is the introduction of the Index of Performance (IP), more commonly called throughput (TP), which is the term we will use.

Throughput is operationally defined by "The average rate of information generated by a series of movements is the average information per movement divided by the time per movement" [9]. Mathematically, throughput is defined as the average index of difficulty divided by the average movement time.

$$TP = (ID/MT) \tag{2.7}$$

After correcting for participant strategy by using effective width, throughput is often used to compare different pointing devices across studies. It has become the de facto standard metric of pointing performance and has been adopted by many research groups [41]. We note that it is often used to make broad general statements such as "device A performs 30 percent better than device B" because it incorporates estimates of both accuracy and speed into a single value.

Unfortunately, no standard alternative to throughput exists for two-part models of pointing performance. Perhaps because using a two-part model itself is not standard practice, no concerted effort has been made to define what should be used to compare studies done with them. However, we argue that the research community could quickly adapt if a strong effort was made to switch over to two-part models. One could easily define two separate throughputs, one for ballistic movements and one for precision adjustments with the two ID coefficients in a two part model. This would allow us to make more nuaced comparisons such as "device A performs 20 percent better for precision tasks than device B, so it should be used for interaction with small screens". Perhaps the sum of these two indexes of difficulty over movement time could be used in place of the general throughput parameter so we could still make general effectiveness conclusions about overall device effectiveness. The results may even be mathematically similar to one-part throughput if amplitude and width had similar impacts to start with.

We discuss these ideas and provide our ideas of what the standard should be in Chapter 6. While providing this standard is not a key focus of this thesis, we do wish to further motivate the pointing research community to address this issue. This will hopefully lead to more nuaced interpretations of costs and benefits of input devices and allow the deployment of more effective models of pointing performance.

2.4 The Relationship Between Gain and Depth

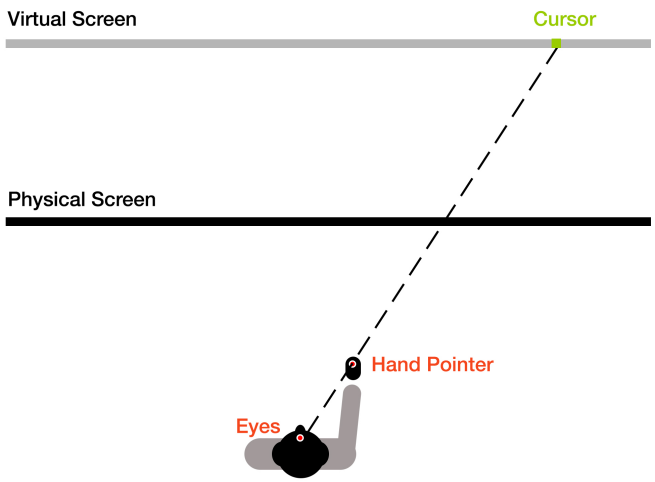

Figure 2.2: A simple diagram of the effect of depth on distal pointing. Note that the same physical movement intersects further horizontally away from screen center when depth is far. This creates an effect similar to gain with target depth.

Gain is the reciprocal of the control-to-display ratio, which is the relative movement of the pointing device to the movement of the cursor that is being used as visual feedback for the pointing. *Depth* is the distance from an observer to an object that is being viewed. There is an obvious relationship between the distance at which pointing is done and induced gain: the farther one is from objects being pointed at, the less hand movement is required to point to different objects. In a real-world environment, target depth is the physical distance to a target. In a virtual stereoscopic 3-D environment, target depth is the perceived or virtual distance to a target. Depth perception uses many cues. In a virtual environment an important cue is the binocular disparity between the images seen by the left and right eyes. One of our primary interests was to better understand how target depth affects pointing performance in the real world, but we also looked at virtual environments where binocular depth may not be the same as the physical depth to the left and right images.

2.5 Distal Pointing Interactions

Figure 2.3: Basic distal pointing interaction implemented with Nintendo Wii Remote to select buttons in a user interface.

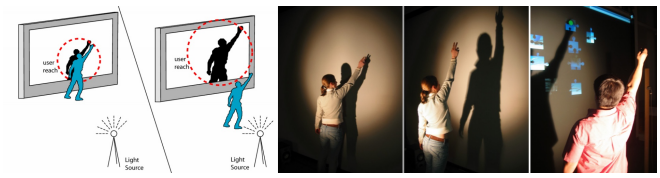

Figure 2.4: Shadow reaching is a proposed distal pointing interaction using the shadow as a metaphor for changing pointing gain by moving back and forth through the room.

Distal pointing interaction has become increasingly common in the last decade. In particular, numerous distal pointing interfaces have been employed in recent gaming technology. Various industrial input products including Microsoft Kinect, Playstation Move, Nintendo Wii Remote, and Leap motion have provided support for distal pointing. Many of these have been picked up by the research community in order to evaluate which are the most effective for specific applications [34, 37, 56]. Many of these are provided by and for gaming applications and traversing 3D virtual environments. However, this is far from the only application for distal pointing. Further improvements and refinements such as shadow reaching have

been suggested to allow variable precision [39]. Classroom interaction has also been highlighted as a key opportunity for applying distal pointing [11]. Given the glut of new distal pointing devices hitting the market, ensuring that we can consistently and effectively evaluate their strengths and weaknesses is an arguably important and timely research area.

2.6 The Kopper Model for Mid-air Pointing

Kopper et al. [23] investigated ways to revise Fitts's Law specifically for mid-air pointing by incorporating target depth into their model (Eq. 2.8) through the use of an angular index of difficulty based on angular amplitude α and angular width ω.

$$MT = a + b\left[\log\left(\frac{\alpha}{\omega^k} + 1\right)\right]^2 \tag{2.8}$$

While Kopper et al.'s techniques show some promise, they unnecessarily narrow the scope of their model by fixing the exponent of ω as $k = 3$, which might prevent its applicability in contexts other than the particular physical set up they used. Moreover, Shoemaker et al. noted that Eq. 2.8 is mathematically similar to Welford's two-part formulation if we ignore the squaring of the logarithmic term and replace $k = 3$ with b_2/b_1, and they showed experimentally that Welford's formulation seems to capture the impact of different gains on pointing performance [40]. In a way, the target depth for mid-air pointing is very similar to gain. Moving farther away from targets increases the effect of small hand movements and vice versa. Thus we suspect that any improvement in modeling seen by Kopper et al. was likely due to their model being similar to Welford's, not to the incorporation of angular *ID* measurement.

2.7 Statistically Comparing Nested Pointing Models

Soukeroff and MacKenzie [41] and others have observed that two-part models inevitably perform better than their one-part counterparts because of additional degrees of freedom (*DOF*). However, there are statistical tests to determine whether nested models such as Fitts's Law and Welford's formulation are actually better at describing the data beyond just the benefit of the increase in degrees of free-

dom. Following Shoemaker et al. [40], we use the F-test in Eq. 2.9 to determine whether a two-part Welford-type model is statistically better than a one-part Fitts-type model: $p_2 = 3$ is the number of parameters in the greater-DOF Welford model, $p_1 = 2$ is the number of parameters in the smaller-DOF Fitts model, and $n = 9$ is the number of sample points in our data. A p-value of .05 or less is considered significant.

$$F(p_2 - p_1, n - p_2) = \frac{\frac{RSS_1 - RSS_2}{p_2 - p_1}}{\frac{RSS_2}{n - p_2}} \tag{2.9}$$

2.8 Impact of Latency on Pointing Performance

By necessity, no computer can react or display feedback to user input instantenously. There is always some amount of time between a button being clicked and the electrical signal travelling down a wire to be interpreted by the CPU. Thus, any computer mediated pointing system must be robust enough to work in the presence of this latency without becoming cumbersome to use. There has been a large amount of work in literature characterizing the relative impact of this effect on many input techniques, but pointing is particularly well studied.

Generally speaking most pointing research has found that the effect of latency gets more severe as latency is increased and that 100ms of latency degrades performance by around 10-15 percent [18, 31, 32, 49]. Cursor jitter imposed by tradeoffs in hardware had an impact as well, but was less than latency [31]. However, others have noted that a similar amount of latency impacts pointing performance much more when targets are moving, as in a virtual environment [16].

It is entirely possible that other factors could cause latency to become more problematic than expected by previous studies. By convention studies use consistent, reasonable input gains and turn off mouse acceleration. This enables greater internal consistency and easily reproducible experimental conditions. The effect of latency is also usually evaluated by adding latency into the system, rather than comparing to a "natural" baseline. Shoemaker further raised the concern that these gains were unintentionally cherry picked to maintain the effectiveness of one part models of pointing [40]. We felt the objections raised by Shoemaker were reasonable and were concerned that latency might become more problematic in distal

pointing as you moved further from the screen. This naturally lead to our second experiment to evaluate natural human baseline performance without the presence of system latency.

2.9 Virtual Reality (VR)

Figure 2.5: Oculus rift and other VR systems have commonly been applied to gaming and other methods of exploring VR systems. In this example a treadmill is placed under the user to allow them to traverse by running in place.

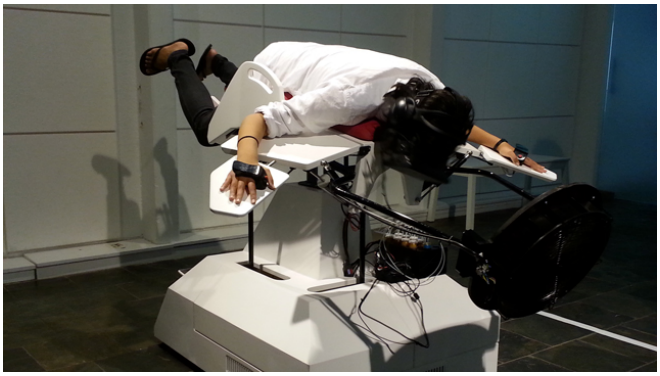

Figure 2.6: A secondary example of a VR system in practice allows flying and exploration like a bird. A specially designed apparatus helps aide this effect.

Virtual Reality (VR) Ellis [4] defines virtualization as "The process by which a human viewer interprets a patterned sensory impression to be an extended object in

an environment other than that in which it physically exists." Virtual Reality (VR) often consists of headcoupled stereoscopic displays, with means of interaction and input to generate a coordinated sensory experience. Some systems include auditory and tactile information. VR environments are used in entertainment (gaming, movies, virtual experiences, etc.), training and simulation (military training, driving and flight simulators, etc.), and medical and scientific visualizations.

Recently, popular interest in virtual reality HMDs (head mounted displays) has been reinvigorated by the announcement and success of the oculus rift [30]. Oculus is a consumer grade VR headset that's been touted as a cheap and easy way to bring VR to the masses [36]. Unsurprisingly, this has helped reinvigorate the HCI community's interest in virtual reality and applications of head mounted displays. Of particular note is incorporating haptic sensations to VR experiences, as well as augmenting and improving VR hardware to improve peripheral vision and reduce motion sickness [1, 21, 22, 54].

Depending on the application, the requirements for a VR system might differ. In training applications, it is often desirable that the VR system evoke responses and task performance nearly identical to that in the real world. Stereoscopic 3-D systems may be unable to completely live up to the ideal of perfect simulation. There are depth cues aside from binocular disparity. The physical screen distance in a VR system is often fixed, and thus rarely the same as the binocular depth of objects being viewed. Our next section discusses attempts to calibrate VR systems and reduce the impact of these disparities.

2.10 Calibrating Virtual Environments

There has been much previous work on VR calibration. Jones, Lee, Holliman and Ezra describe calibrating the camera space against a user space based on user-reported depth values [17]. Yuan, Pan and Daly perform depth tuning of stereoscopic 3-D content based on models of human visual comfort [55]. Iyer, Chari and Kannan describe a method for determining parallax values for stereoscopic viewing that minimize visual artifacts in 3-D television [15]. Since the introduction of the Oculus Rift, this trend has continued. Kulshreshth et al. provide a method to improve depth estimation by dynamic stereo parameter adjustment [24]. Fernandes

Figure 2.7: Image displaying an application using dynamic adjustment of 3D parameters. On the left, a farther object has its stereo parameters made more extreme to highlight the depth effect and improve depth estimation.

et al. suggest subtly modifying field of view to reduce motion sickness [7]. Lastly Finnegan et al. tackle the problem of calibrating audio feedback timing in such a virtual environment [8].

These approaches are all based on stereoscopic perception models either derived from the literature, or from user-reported values. They attempt to reduce distortions associated with binocular viewing. If we want to produce task performance identical to that in real-life tasks, not simply to reduce perceived distortion, we may have to take task performance explicitly into account during calibration. To our knowledge however, no other work currently has attempted to validate previous calibration methods to a performance task, or hit on the idea of calibrating a VR system based on performance metrics.

Chapter 3

Target Depth and k in Computer-Mediated Pointing

The analysis I present in this chapter was done on data collected by Vasanth Rajendran. Previous work in our lab by Shoemaker et al. had investigated the variance of k with respect to gain. They had found that k varies quite linearly and monotonically with different gain settings and thus could be used as a predictor of relative gain settings [40]. However, we'd had one important observation; in distal pointing environments, target depth may play a role similar to gain. With laser pointer style interaction, if you are standing far away it may be easier to cross large gaps and harder to make fine adjustments. This effect is similar to increasing the gain settings of the device, and thus might be predicted by k. If we could use pointing performance to tell how far away the screen is, can we use it as a more reliable indicator of VR calibration than self report data? Therefore in this chapter we present an experiment to test and verify if the k factor also captures target depth, as well as whether that variation is the same for virtual and physical depths.

3.1 Method

We examined the effect of physical target depth D_S (the depth to the display screen) and binocular depth D_V (the depth at which objects are intended to be perceived) on pointing performance (movement time) in a VR environment. Our primary analysis

is for the conditions in which $D_V = D_S$, a real-world physical environment with no virtual binocular disparity (the images for the left and right eyes are identical on the screen), but we also examined the cases where $D_V \neq D_S$, virtual environments that had virtual binocular disparity (the images for the left and right eyes were not identical on the screen). The design was based on an experiment by Shoemaker et al. [40] that evaluated physical 2-D pointing on a large display for varying gain values. In our experiment, we evaluated 3-D pointing for different binocular depths and for different target depths with system gain held at one.

3.1.1 Participants

We recruited 21 participants at our university through on-campus advertising. As a requirement, all were right-handed with normal or corrected-to-normal vision. One participant's data had to be discarded due to equipment malfunction. Of the remaining 20, 8 were female and 12 male. Age ranged from 20 to 28 years (mean 23.3). All participants were regular computer users (9+ hours per week). The study was approved by the behavioral Research Ethics Board at our university. Participants signed a consent form prior to the start of the study. Each was compensated with $10.

3.1.2 Apparatus & Materials

The hardware and software used for this experiment were based on previous experiments investigating the impact of gain on distal pointing by Shoemaker et al. One of the main additions was the supporting of VR and different virtual depths.

Hardware

The wall display was a 5.16m×2.85m (width × height) glass screen rear-projected by a 4×3 array of 800px×600px stereo projectors. The images of adjacent projectors overlapped by 160px with a blending function to minimize the appearance of discontinuities. The overall resolution of the display was 2720px×1480px. Viewing was binocular frame-sequential stereo at 60Hz. Users viewed the display through shutter glasses synchronized with the projectors. We calibrated the binocular disparity based on an initial estimate followed by visual inspection and

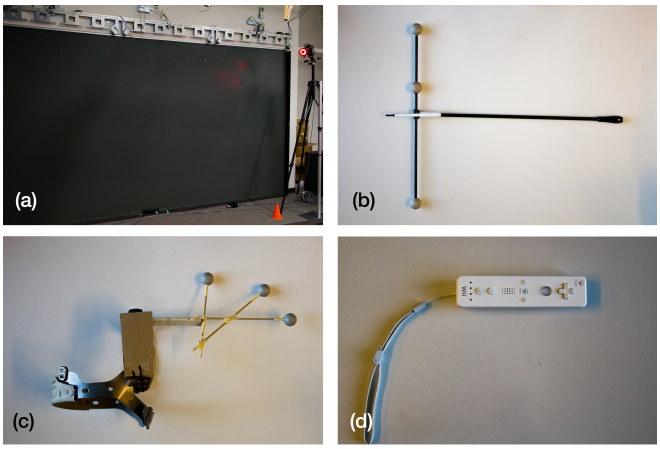

Figure 3.1: Apparatus: (a) Large screen stereoscopic display, (b) Hand-held pointer, (c) Head-tracking gear, and (d) Wii Remote for click events.

adjustment to reduce visual distortions. The display was driven by an 8-core Intel Processor with 6GB of RAM and dual NVIDIA GeForce GTX 260 graphics processors, running Windows 7.

Five Vicon cameras (a high speed infrared-based motion capture system) tracked participants using head gear and a hand-held pointer each fitted with reflective balls (Figure 3.1) after using Vicon's standard wand wave calibration procedure. We tracked participants' heads and the pointer positions and orientations to compute and project a virtual cursor on the screen. Participants stood at one of three depths D_S from the screen with targets displayed at one of three depths D_V (Figure 3.2). The display was not head-coupled; it was "movie theatre stereo" with a fixed viewer position assumed because we wanted no virtual binocular disparity when the target was in the plane of the physical screen (i.e. when $D_V = D_S$ binocular depth would be the same as physical depth and the images for the left and the right eyes would thus be identical, and this would not change with head movement).

'Tapping' on targets was performed using the thumb (A) button on a hand-held Nintendo Wii Remote. Participants held the remote with the left hand (the non-pointing hand) to minimize any disturbance caused by clicking.

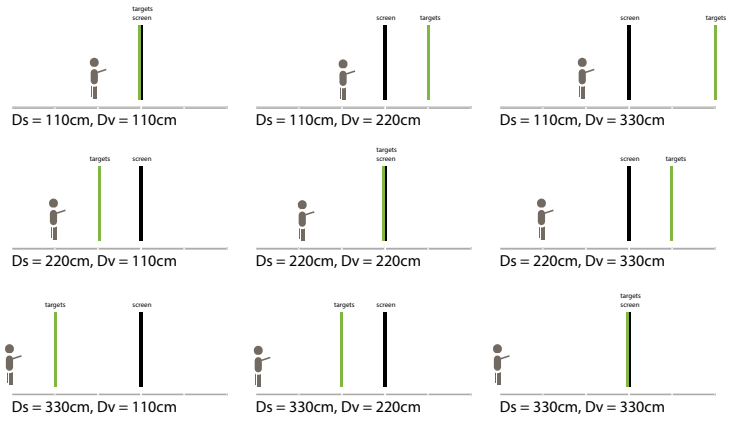

Figure 3.2: Placement of the targets and the participant in an experimental trial. The virtual plane of the targets is green; the physical plane of the screen is black.

Software

The experimental software was written in C# using the Microsoft XNA Game Studio 4.1 library. The WiimoteLib library [33] was used to communicate with the remote and to detect click events. The Vicon motion tracking system managed by Vicon IQ software provided the positions of the user's head and the pointer. Our software recorded all click events and the timing for each trial condition. Raw tracking data was captured 60 times a second.

Interaction-plane pointing was used to compute the cursor position on the screen. An imaginary vertical plane containing the user's eye and hand was extrapolated to intersect with the virtual screen, corresponding to the binocular depth D_V for a trial condition. The cursor was rendered on the physical screen with appropriate binocular disparity for this depth (Figure 3.3). The imaginary plane was used only by the software: participants did not sight along the hand but instead watched the cursor on the screen much as they would use a mouse by watching only the cursor not their hand. The plane of the cursor was always the plane of the target, in keeping with our desire to mimic physical pointing such as might be done with a laser pointer in a classroom.

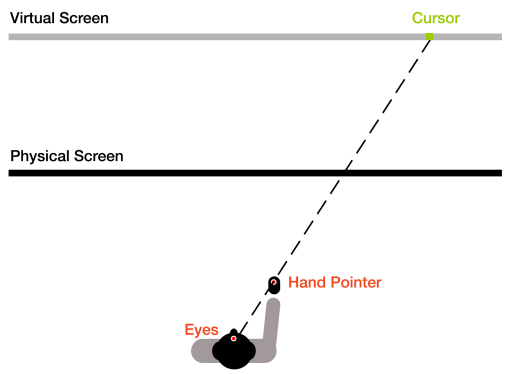

Figure 3.3: Computing the cursor position from the head and hand-held pointer positions.

Layout, Task, and stimuli

The experimental task was a serial 1-D tapping task between two target pairs, modeled closely after Shoemaker et al. [40] and the original Fitts experiments [10]. ISO 9241-9 [3] defines a 2-D task for pointing performance; we used a 1-D task because we were concerned with the fundamental applicability of Fitts's Law. Although the task was 1-D, it is still an example of 3-D pointing because targets were perceived at different depths in a 3-D stereoscopic environment.

Targets were two identical vertical bars of variable width W and fixed height, spaced apart by variable amplitude A. The cursor was a thin (0.5 cm) vertical line of the same height as the targets. Lighting in the room was fairly dim to reduce any depth cues other than binocular disparity.

Each trial condition was defined by four independent variables: movement amplitude A, target width W, binocular depth of the targets D_V, and physical depth D_S of the participant from the screen. If $D_V = D_S$, there was no virtual binocular disparity from VR (targets were intended to be perceived as in the plane of the physical screen so the images on the screen for the left and right eyes were identical, which we refer to as having no virtual binocular disparity). If $D_V > D_S$, targets were perceived *behind* the physical screen (positive parallax) but if $D_V < D_S$, targets were perceived *in front of* the physical screen (negative parallax). The $D_V = D_S$ and $D_v < D_S$ cases are shown in Figures 3.4 and 3.5, respectively.

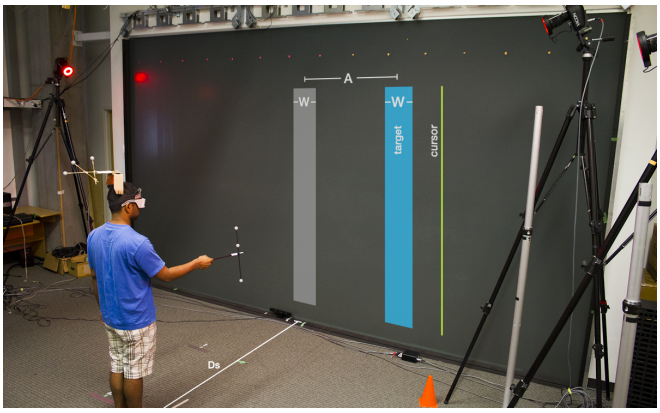

Figure 3.4: An illustration of the participant performing the pointing task for $D_V = D_S$. Targets are in the plane of the screen.

For each target pair, a participant first tapped the start target and then performed a sequence of eight back-and-forth taps between the two targets. The destination target for a tap was highlighted blue and the starting target was grey. The participant was required to correctly tap the first starting target to initiate a trial. After each tap the destination target turned grey and the former starting target turned blue and became the new destination target.

Times and locations of all nine taps were recorded. When tapped, the active target briefly flashed green to indicate success, or red to indicate an incorrect tap. Participants were not required to correct errors. One target was always directly in front of the participant, the other was to the right, at a distance defined by the current amplitude (A) condition. This was done to avoid a cross-lateral inhibition effect (CIE) [19, 38], the observation that hand movements that cross the midline of the body are more complex than those that do not.

3.1.3 Procedure

Each participant performed the experiment in a single session of approximately 45 minutes. After filling out a consent form and a pre-questionnaire that gathered demographic information, participants were introduced to the apparatus and the pointing task. They were instructed to complete the task as quickly as possible with a goal of $\sim$95% accuracy. Each completed a practice session of at least five

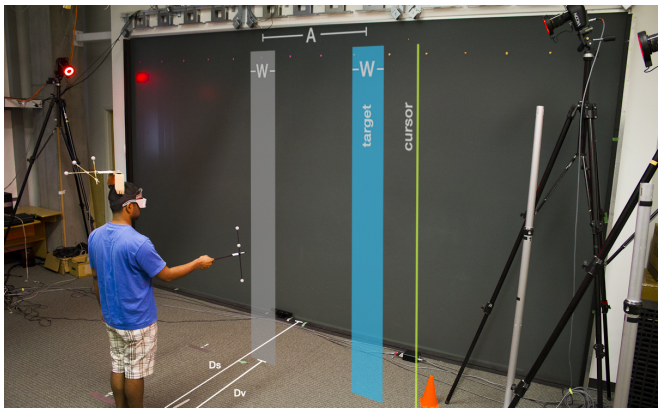

Figure 3.5: An illustration of the participant performing the pointing task for $D_S > D_V$. Targets are perceived in front of the screen.

randomly chosen (D_V, D_S, A, W) combinations (without duplicates) that included all D_V and D_S values. They were invited to practice until comfortable with the system.

At the beginning of the block of trials for each D_V and D_S pair, a practice trial with one A and W pair was presented to familiarize participants to the new depth condition. These trials were presented in the regular flow of the experiment; a participant was not informed these were only practice trials. Between each block of the experiment representing a D_S value, participants were required to take a break for at least three minutes. They were encouraged to take more time if desired, but very few did.

After all blocks were completed, participants filled out a post-questionnaire of feedback and comments on the experiment. We do not report on this here.

Measures

Pointing performance was measured as the time taken to execute each individual tap action. A trial began when the participant successfully tapped on the start target, and ended with the eighth tap on a destination target. For each tap, the software recorded movement time MT, and position of the cursor when the click was made. The position Δ_{tap} was an offset from the center of the destination target. There were a total of 12,960 observations. As is common practice in pointing evaluation,

mistrials and outliers were removed. Any click landing more than half the trial amplitude away from the target (likely to be caused by an accidental click) was ignored and any tap whose movement time was more than three standard deviations from the mean was removed. In total 312 taps were removed from the dataset.

Experimental Design

The within-participants design had independent variables binocular depth of targets D_V (110cm, 220cm, 330cm), screen depth from the participant D_S (110cm, 220cm, 330cm), target width W (5cm, 10cm, 20cm), and movement amplitude A (25cm, 50cm, 75cm). All the variables were fully crossed.

$$\boxed{\text{20 participants } (N)} \times$$

$$\boxed{\text{3 binocular depths } (D_V)} \times \boxed{\text{3 screen depths } (D_S)} \times$$

$$\boxed{\text{3 movement amplitudes } (A)} \times \boxed{\text{3 target widths } (W)} \times$$

$$\boxed{\text{8 taps}} = \boxed{\text{20 participants}} \times \boxed{\text{648 taps}} = \boxed{\text{12,960 total taps}}$$

For a given D_S, all three D_V conditions were presented consecutively; for a given D_V-D_S pair, a participant performed trials for all combinations of A and W before switching to another D_V-D_S condition. We decided against completely mixing up D_S and D_V to avoid participants having to move back and forth between different locations in the room. Condition order was partially counterbalanced across participants.

Hypotheses

We had five hypotheses, the fourth of which has both a weak and a strong variant. Four of the hypotheses are similar to those of Shoemaker et al. [40], who conducted studies of mid-air pointing where gain was varied. They found that Welford models were better than Fitts models and that experimentally measured k ratios varied linearly with gain. Our hypotheses replace "gain" with "target depth" but are otherwise the same as those of Shoemaker et al.

H1 One-part formulations (Fitts's Law and the "Shannon" version of Fitts's Law) will not accurately model pointing performance at all target depths.

H2 Two-part formulations (Welford's formulation and the "Shannon" version of Welford's formulation) will accurately model pointing performance at all target depths.

H3 Physical target depth has no significant effect on pointing performance, only binocular depth matters.

H4 (weak) The ratio $k = b_2/b_1$ of coefficients in a Welford formulation varies *monotonically* with target depth.

H4 (strong) The ratio $k = b_2/b_1$ of coefficients in a Welford formulation varies *linearly* with target depth.

H5 Using an angular index of difficulty will not improve the strength of our models of pointing performance.

3.2 Results

Analyzing data from individual trials or from individual participants is rare in the literature on pointing performance. We performed both ANOVA and a linear regression analysis using data averaged over all participants [10, 41]. For each A and W pair in each D_V-D_S pair condition, we first averaged within a trial and then the trial averages were averaged over all trials and all participants, resulting in nine values for each of the nine A-W pairs in each of the nine D_V-D_S conditions (81 values in total). A Shapiro-Wilks test for normality was conducted on the participant average movement times, which showed a violation of the normalcy assumption ($P < 0.01$) which could be due to outliers that we attempted to remove through filtering. Overall average movement times and 95 percent confidence intervals for each condition are included in Appendix A.

3.2.1 ANOVA for Movement Time

Significant main effects of D_V, D_S, A and W were found. Significant interactions of $D_V \times D_S$, $D_V \times W$,$D_V \times A$, $D_S \times A$, $D_S \times W$, and $A \times W$, and three-way interactions of $D_V \times D_S \times W$ and $D_V \times D_S \times A$ were also found. No other interactions were found. This is summarized in Table 3.1. The fact that movement amplitude A

Factor	ε	df_effect	df_{error}	F	p	Partial η^2
D_V		1.292	24.557	14.830	.000	0.438
D_S		1.606	30.516	10.695	.001	0.360
A		1.276	24.248	406.567	.000	0.955
W		1.099	20.887	162.062	.000	0.895
$D_V \times D_S$		2.727	51.814	39.463	.000	0.675
$D_V \times A$		3.164	60.108	10.786	.000	0.362
$D_V \times W$		2.131	40.482	16.494	.000	0.465
$D_S \times A$		3.357	63.790	4.183	.007	0.180
$D_S \times W$		2.311	43.914	8.321	.001	0.305
$A \times W$		3.138	59.617	11.152	.000	0.370
$D_V \times D_S \times A$		4.240	80.563	5.113	.001	0.212
$D_V \times D_S \times W$		3.840	72.959	6.464	.000	0.254

Table 3.1: Significant ANOVA main effects and interactions for movement time (MT). No other interactions were found. All tests were adjusted with Greenhouse-Geisser.

and target width W affect pointing performance is fundamental to any discussion of pointing performance so effects of A, W and the $A \times W$ interaction were not surprising.

We anticipated an effect of D_V because varying D_V is a factor that is similar to changing gain for 2-D pointing, and other researchers [2, 20, 28, 40] report gain affects pointing. The interactions of $D_V \times W$ and $D_V \times A$ were not anticipated, but perhaps not surprising in retrospect. The role of D_V in influencing MT is not entirely understood. If it affects pointing performance in a manner similar to gain it is easy to imagine that increasing gain might cause target amplitude to have less of an effect than width because it would be easier to cross large distances but harder to make small corrections. Thus the impact of A and W might depend on D_V through a change in perceived gain, which could lead to the interactions we observed.

The effect of D_S on MT and its interactions with other factors were not anticipated. In an ideal VR setting, the virtual scene perceived by a user should not be affected by the position of the physical screen. Viewed binocularly, the destination

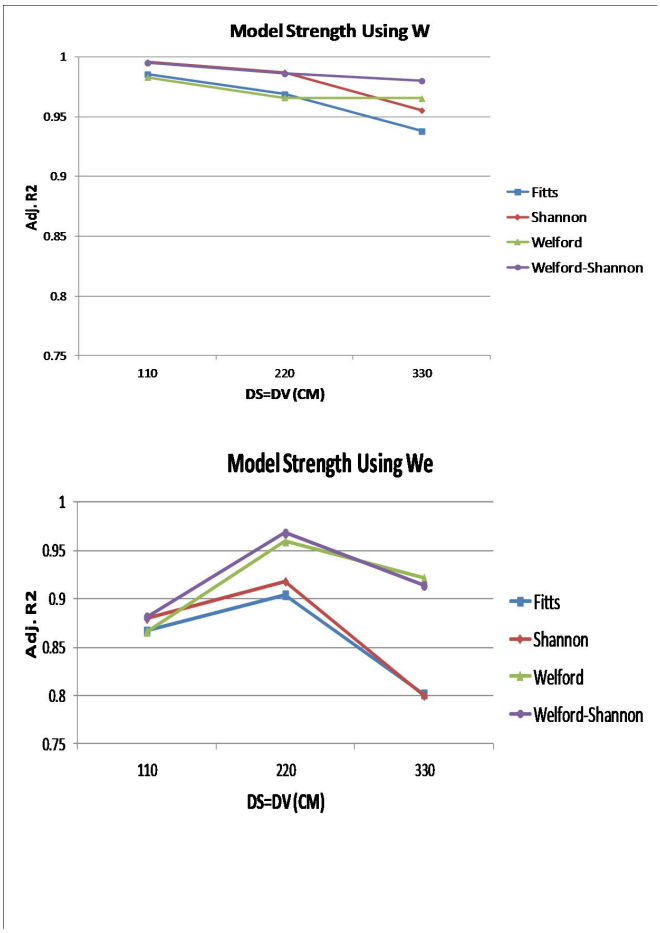

Figure 3.6: R^2 values for each target depth using W (top) and W_e (bottom).

target should be perceived to be on the virtual plane at D_V, which is independent of D_S. However, there are other depth queues at work that may have had enough influence to effect pointing performance.

3.2.2 Regressions for One-Part and Two-Part Models

Regressions for the Fitts and Welford models and for the two Shannon variants were performed only for the nine real-world $D_V = D_S$ conditions. The regression parameters are shown in Table 3.2. To adjust for bias due to accuracy, we performed the regressions using both target width W and effective target width W_e

[57]. We computed W_e by the recommended approach using standard deviations of tap positions as in Eq. 2.6.

Width	$D_V = D_S$	Fitts				Welford						F-test		
		a	b	R^2	RSS	a	b_1	b_2	k	R^2	RSS	F-ratio	p	sig?
W	110	542.10	301.72	0.98	11436	535.76	302.54	301.18	0.996	0.98	11432	0.002	.966	no
	220	565.91	324.56	0.97	29057	698.49	307.20	335.70	1.093	0.97	27150	0.421	.540	no
	330	526.50	392.04	0.94	86862	1173.19	307.39	446.42	1.452	0.96	41501	6.558	.043	yes
	all	544.84	339.44	0.91	321031	802.48	305.71	361.10	1.181	0.91	299432	1.731	.201	no
W_e	110	332.27	456.23	0.87	105935	−7.83	491.62	416.33	0.847	0.87	90976	0.987	.359	no
	220	424.70	448.92	0.90	89424	1242.19	384.77	575.75	1.496	0.96	32101	10.714	.017	yes
	330	423.09	555.58	0.80	277761	1902.41	459.28	804.89	1.753	0.92	94268	11.679	.014	yes
	all	427.43	467.06	0.76	844546	758.11	439.32	515.15	1.173	0.76	810745	1.001	.327	no
Width	$D_V = D_S$	Shannon-Fitts				Shannon-Welford						F-test		
		a	b	R^2	RSS	a	b_1	b_2	k	R^2	RSS	F-ratio	p	sig?
W	110	244.00	378.57	0.99	3417	204.88	384.26	376.80	0.981	0.99	3288	0.235	.645	no
	220	241.40	408.75	0.98	11931	341.73	394.17	413.28	1.048	0.99	11083	0.459	.523	no
	330	134.74	493.65	0.95	86862	811.84	395.25	524.20	1.326	0.98	23785	15.912	.007	yes
	all	206.72	426.99	0.92	272932	452.82	391.23	438.10	1.120	0.93	257626	1.426	.244	no
W_e	110	-149.50	594.66	0.88	105935	−525.55	640.22	564.96	0.882	0.88	80741	1.872	.220	no
	220	-41.84	580.31	0.92	76474	800.28	500.56	681.98	1.362	0.98	25437	12.038	.013	yes
	330	-157	720.76	0.80	280066	1404.69	595.20	935.75	1.572	0.91	103397	10.252	.019	yes
	all	-61.31	606.07	0.77	822633	277.49	572.04	643.38	1.125	0.77	792872	0.901	.352	no
Width	$D_V = D_S$	Shannon-Fitts Angular A and W				Shannon-Welford Angular A and W						F-test		
		a	b	R^2	RSS	a	b_1	b_2	k	R^2	RSS	F-ratio	p	sig?
W	110	403.49	438.58	0.99	11070	426.04	434.30	438.86	1.010	0.99	3292	14.176	.009	yes
	220	412.88	478.99	0.99	26666	509.97	456.79	613.00	1.705	0.99	7175	16.298	.007	yes
	330	341.72	580.28	0.96	40996	793.11	460.31	788.33	1.632	0.98	18396	7.371	.035	yes
	all	389.25	497.20	0.92	289693	795.79	407.10	506.67	1.245	0.96	127298	30.617	.000	yes

Table 3.2: Modeling movement time (ms) using Fitts and Welford formulations (top) and Shannon-Fitts and Shannon-Welford formulations (middle) for actual width W and effective width W_e for each $D_V = D_S$ condition (no added virtual binocular disparity) and for all $D_V = D_S$ conditions combined. Fitts and Welford models computed using Angular A and W are also reported (bottom). The coefficients and adjusted R^2 for each model with the F-ratios, p-values and significance from an $F(1,9)$-test (Eq. 2.9 with $p_1 = 2$, $p_2 = 3$, and $n = 9$) in the last three columns compare nested Fitts and Welford (or Shannon-Fitts and Shannon-Welford) models.

We performed F-tests as in Eq. 2.9 to compare pairs of nested models to determine if the two-part models actually explained the data better. Two-part models were found to characterize the data significantly better than one-part models for target depth 330cm in regular width, and both 220cm and 330cm for effective width. There was no signficant difference between one-part and two-part models for the other condtions. The R^2 values for effective width were generally lower than for regular width, but greater than 0.8. This is somewhat abnormal, but not especially worrisome as the models are still reasonable, as will be commented on in the Discussion. The variation of the R^2 values for $D_V = D_S$ is shown in Figure 3.6.

We also performed regressions for the Fitts and Welford models using A and W measured in angular space in a manner similar to Kopper et al. [23]. Targets were directly in front of and offset to the right of a participant, so angular width was slightly different for the two targets positions. To account for this we averaged the two values. Regression coefficients changed in many cases, but trends of reduced R^2 values at higher target depths were replicated and the R^2 values obtained from the angular models were similar to those for the corresponding classic models. Two-part models were again sometimes significantly better than one-part models. Table 3.2 (bottom) shows regressions for just the Shannon-Fitts and Shannon-Welford models using angular W where two-part models were always better than one-part models.

3.2.3 The Ratio k as a Function of D_V

Model	Width Used	Intercept	Slope	Adjusted R^2
Welford	W	0.7728	1.728×10^{-1}	0.901
Shannon-Welford	W	0.7234	2.284×10^{-1}	0.890
Welford	W_e	0.4596	4.528×10^{-1}	0.941
Shannon-Welford	W_e	0.5826	3.449×10^{-1}	0.951

Table 3.3: Linear regression indicating how k varies as binocular depth D_V (cm) changes for each of the four models.

To test the hypothesis that k values varied at least monotonically and perhaps linearly with target depth, we performed a linear regression analysis on k computed by the Welford and Shannon-Welford models using both actual target width W and

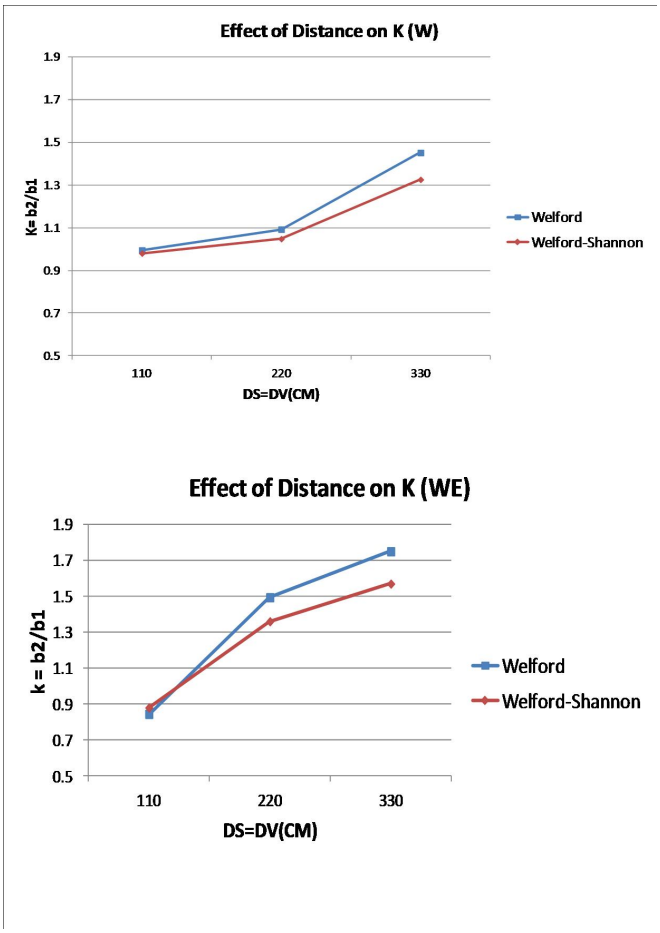

Figure 3.7: k values calculated using the Welford formulation (a) for A and W (b) for A and W_e.

effective target width W_e. The results are presented in Table 3.3 and plotted in Figure 3.7. The k values vary linearly ($R^2 > 0.85$) with target depth D_V using width W. Using effective width W_e, the linear model for k is somewhat more pronounced ($R^2 > 0.90$).

3.3 Discussion

We summarize the results according to our hypotheses, and then discuss each in more depth.

H1 One-part formulations (Fitts's Law and the "Shannon" version of Fitts's Law) will not accurately model pointing performance at all target depths. *Consistent with data.*

H2 Two-part formulations (Welford's formulation and the "Shannon" version of Welford's formulation) will accurately model pointing performance at all target depths. *Somewhat consistent with data.*

H3 Physical target depth has no effect on performance, only binocular depth matters. *Not consistent with data.*

H4a The ratio $k = b_2/b_1$ between the two linear coefficients in a Welford formulation varies *monotonically* with target depth. *Consistent with data.*

H4b The ratio $k = b_2/b_1$ between the two linear coefficients in a Welford formulation varies *linearly* with target depth. *Somewhat consistent with data.*

H5 Using angular index of difficulty will not improve the strength of our models of pointing performance. *Consistent with data.*

3.3.1 A Closer Look at the Data

Movement time data is shown in Figure 3.8. The graphs in the first column present a scatterplot for different binocular depths D_V, with lines connecting points of the same movement amplitude A. The graphs in the second column present exactly the same data, but with lines connecting points of the same target width W.

Figure 3.8 reveals a pattern similar to one shown by Welford (Figure 5.8, page 158 [53]). Movement time increases roughly linearly with ID within either a fixed A value or a W value, but not across changes in both. This separable effect of A and W grows with increases in D_V. A similar effect has been found by Shoemaker et al. [40] in a reänalysis of other researchers' data [2, 12] and of their own data. Both Graham and Shoemaker et al. concluded that Welford's two-part formulation is necessary to account for this pattern and to accurately model movement time when gain varies. The applicability of a two-part formulation is further supported by Graham's analysis of hand velocity and acceleration profiles during pointing,

which revealed separable effects of A and W during different temporal segments of target acquisition.

Looking at our data, we see that one-part models perform poorly at modeling the movement time data for $D_V = 330$cm and $D_V = 220$cm, but two-part models do better. The k values from Eq. 2.5 serve to quantify the separability of contributions of A and W to movement time. From Figure 3.8, we expect k to be close to unity for a binocular depth D_V of 110cm and to increase as D_V increases, as borne out by Figure 3.7.

3.3.2 One-Part and Two-Part Models for Pointing

There is no universally accepted threshold for a "good" R^2 value to determine if a formulation accurately models pointing performance. MacKenzie's suggestion [27] of $R^2 \geq 0.90$ as a guideline when evaluating Fitts's Law results has been used in the literature, and we employ this as our threshold.

Hypothesis **H1** was *consistent with out data*. One-part models (Fitts and Shannon-Fitts) were successful in characterizing movement time using actual target width W. The Fitts formulation produced fits ranging from $R^2 = 0.94$ to $R^2 = 0.98$ at different target depths. The Shannon-Fitts formulation was slightly better at modeling the data, with fits ranging from $R^2 = 0.95$ to $R^2 = 0.99$. These R^2 values beat MacKenzie's 0.90 threshold. However, for both formulations, it is clear that R^2 values decreased as D_V increased (Figure 3.6).

Using effective width W_e, the one-part models were less successful. For the Fitts formulation, the fit ranged from $R^2 = 0.80$ to $R^2 = 0.90$. Using MacKenzie's 0.90 threshold, the Fitts formulation only produced one barely acceptable model for one target depth and was worst for the largest D_V. The Shannon-Fitts formulation fared slightly better, with the fit ranging from $R^2 = 0.80$ to $R^2 = 0.92$. Again, Shannon-Fitts only produced an acceptable fit for one target depth, and the $D_V = 330$ condition was the worst.

Effective width is sometimes more relevant because W_e can be a more accurate representation of the task [41]. One reason for the poorer fit for one-part models using effective width W_e is apparent: examining the k values from the two-part models (Figure 3.7) shows that k deviated more from a value of one when using W_e

than it did using W. The more k deviates from unity, the worse a one-part model will be at describing the data. R^2 values were lower in general using effective width. We argue that this is likely due to noise and outliers in the data that are particularly common with mid-air pointing. Hand tremors and input noise happen more frequently than with other pointing techniques (such as a mouse, where the hand rests securely on a desk). We saw a few cases where movement time was normal (so the point was not removed by outlier filtering), but the selection was made very inaccurately. This would have no impact on regular width models which do not care about accuracy, but would degrade effective width models.

For both W and W_e the quality of fit decreases with increasing D_V. Using W_e, the one-part models had unacceptable R^2 values for all but one D_V. Both two-part models (Welford and Shannon-Welford) produced a consistently good fit at every target depth D_V using actual target width W. Regression fits for Welford ranged from $R^2 = 0.96$ to $R^2 = 0.98$. For Shannon-Welford they ranged from $R^2 = 0.98$ to $R^2 = 0.99$. Using effective width W_e, the regression fits ranged from $R^2 = 0.87$ to $R^2 = 0.96$ for Welford and from $R^2 = 0.88$ to $R^2 = 0.98$ for Shannon-Welford.

Hypothesis **H2** was *consistent with our data*. An F-test determined that two-part models described the data significantly better than the corresponding one-part models for $D_V = 330$cm using both W and W_e. They also outperformed one-part models for $D_V = 220$cm using W_e (Table 3.2). For two-part models, only the $D_V = 110$ using W_e condition did not meet the 0.9 threshold, and that was by but a small margin.

3.3.3 The k Parameter and Target Depth

Hypothesis **H3** was *not consistent with our data*. Most of our analysis was done only on the real-world $D_V = D_S$ conditions. When we analyzed all D_V-D_S conditions, including those with $D_V \neq D_S$, as shown in Figure 3.9, the variation of k depended both on physical depth and on binocular or perceived depth. It may be that this happens when the VR environment is not well calibrated or it may be a fundamental flaw in stereoscopic 3-D. Although we did a rather careful calibration to reduce the visual distortion of our system, perhaps it was not good enough; in a perfectly calibrated system target depth might not matter. We explain later how to

use the k-lines in Figure 3.9 to possibly improve calibration.

Hypotheses **H4a** and **H4b** were *somewhat consistent with our data*. Using actual target width W, the k values fit a linear model with $R^2 = 0.901$ for the Welford formulation and $R^2 = 0.890$ for the Shannon-Welford formulation. Using effective target width W_e, the k values were more accurately modeled, $R^2 = 0.941$ for the Welford formulation and $R^2 = 0.951$ for the Shannon-Welford formulation. However, given that we only have three data points it is hard to make strong conclusions about the linearity of the data. Looking at Figure 3.7 we notice the data seems somewhat curved using width, but using effective width it is much closer to linear. Future studies should further investigate this issue.

3.3.4 Angular Amplitude and Angular Width

Hypothesis **H5** was *consistent with our data*. Using angular amplitude α and angular width ω did not show consistent real improvement over the classic one- and two-part formulations. Simply substituting α and ω for A and W into the four models made little difference (Table 3.2). All trends, including the improvement of Welford over Fitts and k increasing as depth increases remained the same. Using the angular model proposed by Kopper et al. (Eq. 2.8), R^2 was less than 0.70 for both ω and ω_e in every condition. Using $k = 1.47$ (an average obtained from our analyses) instead of $k = 3$ in Eq. 2.8, R^2 values were closer to those for our angular Welford model, but were uniformly not as good. Angular measures incorporate DV in the model, so Table 3.2 presents models for the "all" condition (exactly the $D_V = D_S$ conditions). This best case test shows a minor 0.04 R^2 improvement for two-part models, but none for one-part models. This suggests that any improvement Kopper et al. saw for their model may be due mostly to its mathematical similarity to Welford's formulation. At least for small angles, measuring amplitude and width as angles acts like a simple scaling operation; it keeps the fundamental model the same, but changes some of the constants, which, as Shoemaker et al. [40] discuss at length, does not help explain the behaviors shown in Figure 3.8. Kopper et al.'s choice of $k = 3$ seems arbitrary. Shoemaker et al. [40] report values as low as $k = 0.3$.

3.4 Conclusions and Future Work

In this chapter we examined pointing performance using stereoscopic 3-D when perceived depth to the target (binocular depth) and actual target depth varied. Four research contributions were reported: (1) Welford-like two-part models outperformed Fitts-like one-part models by more robustly characterizing pointing performance across varying target depths; (2) angular measures for movement amplitude and target width did not improve model strength for our data; (3) pointing performance was affected by binocular depth and the k-ratio of b_2/b_1 that characterizes this appeared to vary linearly with binocular depth but there was a confound with target depth that is unexplained; and (4) task-specific calibration for VR environments could be based on Welford-like pointing models.

3.4.1 Welford Models vs. Fitts Models

Traditionally, Fitts's Law has been the model of choice for pointing performance. Similar to findings by Welford [53], Graham and MacKenzie [13], and Shoemaker et al. [40], our data show that contributions to movement time of movement amplitude A and target width W are separable and two-part models such as Eq. 2.3 and Eq. 2.4 are better than one-part models such as Eq. 2.1 and Eq. 2.2 for some target depths and are always at least as good at any target depth. Using F-tests, we found in some cases there were significant differences between the two types of models. Two-part models cannot be simplified to one-part models by introducing scale factors for A and W (Shoemaker et al. [40] provide a detailed discussion of this). Our data suggest that simply measuring movement amplitude and target width as angles does not improve modeling strength. These findings add to an understanding of how to best model pointing in VR environments and they provide further support for the use of Welford-like models when Fitts-like models do not provide adequate power.

3.4.2 Lack of Improvement Using Angular Models

Our opinion is that classic two-part models are superior to Kopper et al.'s one-part angular model, but if such a model is used, k should be variable, which makes it a two-part model. We see no justification for squaring the logarithmic term, espe-

cially because it goes against the information-theoretic interpretation that is fundamental to all variants of Fitts's Law. Our experiment used a fairly large screen, such as might be in a classroom, and the depths we chose were typical of where a lecturer might stand relative to a screen, so the conditions were fair representations of at least some types of real-world usage. Nevertheless, we cannot rule out that for large movement amplitudes angular models might be better.

3.4.3 Calibrating VR Systems Using *k*-values

Ellis [4] defines virtualization as "The process by which a human viewer interprets a patterned sensory impression to be an extended object in an environment other than that in which it physically exists." Depending on the application, the requirements for a VR system might differ. In training applications, it is often desirable that the VR system evoke responses and task performance nearly identical to that in the real world.

There are depth cues other from binocular disparity. Physical target depth in a VR system is often fixed, and rarely the same as the binocular depth of objects being viewed. Research by Teather and various co-authors investigated how stereoscopic 3-D impacts pointing performance for a variety of interaction techniques [44–48, 50]. They noted problems modeling pointing performance with accepted Fitts's Law methods specifically for "ray casting" techniques similar to the mid-air pointing techniques we investigated [43]. Hypothesis **H3** explored whether this might affect pointing performance in VR. Our data, though preliminary, suggests it does. Our findings might lead to calibration techniques that could ameliorate this.

Jones, Lee, Holliman and Ezra describe calibrating the *camera space* against a *user space* based on user-reported depth values [17]. Yuan, Pan and Daly perform depth tuning of stereoscopic 3-D content based on models of human visual comfort [55]. Iyer, Chari and Kannan describe a method for determining parallax values for stereoscopic viewing that minimize visual artifacts in 3-D television [15]. These approaches are all based on stereoscopic perception models – either derived from the literature, or from user-reported values. They attempt to reduce distortions associated with binocular viewing. If we want to produce task performance

identical to that in real-world tasks, not simply reduce perceived distortion, we may have to take task performance explicitly into account during calibration.

Our finding that pointing performance is not independent of physical target depth D_S suggests that for a given target depth we might need to adjust binocular depth if we want to induce pointing performance that matches real-world pointing. Figure 3.9 shows measured k-values for different target depths. It also shows the k-line for the conditions where $D_V = D_S$, which has a distinctive slope different from two of the others. We can treat the viewing parameters for VR as a "black box" and use measured k-values as a compensation table. If D_V is the desired binocular depth, the $D_V = D_S$ line gives the k-value expected for real-world pointing performance. Using the k-line for the target depth of the VR environment we can find that k-value and determine the D'_V corresponding to it and then use that binocular depth (instead of the intended binocular depth) to invoke the performance we want. An optimization algorithm could thus "tune" its parameters so that measured pointing performance is close to measured performance in a physical environment. This no doubt requires additional pointing performance data. Our first experiment had only nine points, but with sufficient data, intermediate target depths could be interpolated to achieve the desired corrections at a suitable granularity. This is definitely an area for further investigation.

3.4.4 The Effect of Binocular Depth on Pointing Performance

Our data revealed the relative magnitudes of contributions of A and W can be captured by a single parameter k that seems to vary with target depth. This extends earlier findings by Shoemaker et al. [40] who found that k varied linearly with gain for similar mid-air pointing in non-VR environments. We offer a possible explanation for why target depth matters by appealing to the information-theoretic interpretation of Fitts-like pointing models in which the coeffcient of the index of difficulty is the rate at which the sensorimotor system processes information. Two-part models such as Eq. 2.3 and Eq. 2.4 have coefficients for the A-term and for the W-term. These might be interpreted as the rate at which different components of the sensorimotor system process information, the first for the ballistic stage of movement and the second for the homing stage. However, they might also sim-

ply reflect an information-theoretic measure of uncertainty in the initial and final positions. Our data, and previous findings by Welford [53], Graham and MacKenzie [13], and Shoemaker et al. [40], all show that the two rates differ in some circumstances. For target depth, we might imagine that the perceived depth to the target has a different influence on the homing stage than on the ballistic movement stage. This is definitely a question to be examined in future research.

There is also an important point to be investigated in future work. We've shown that k varies linearly with target depth in a computer mediated application (where the computer processes and provides feedback to the user) but can we use this to conclude that this k variation is a true aspect of human senorimotor performance? One possible explanation for this effect (or at least its extremity) is system latency. In any computer system there is going to be a small lag between when users perform a physical action and when the computer can react and respond to it. This has been shown to reduce pointing performance and become more problematic the larger this gap [31]. It seems possible that such latency would make fine adjustments in the homing stage much more difficult, but provide little impact on rough ballistic motion. Perhaps the variation of k is made more extreme by the presence of latency than we would normally see in natural pointing. We investigate this Chapter 5.

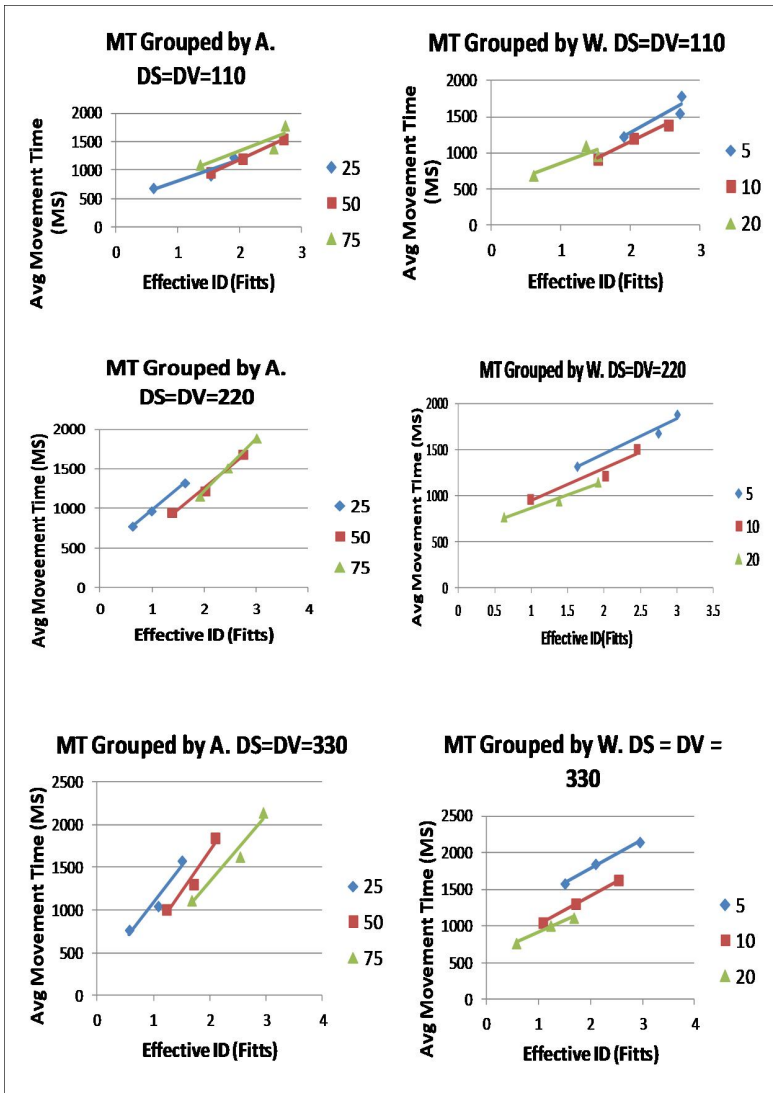

Figure 3.8: Scatterplot of MT vs. effective ID. Points are identical in the right and left plots. Points are connected in two different ways to illustrate the separability of A and W: Lines on the left connect points representing tasks with the same movement amplitude A; lines on the right connect points representing tasks with the same target width W.

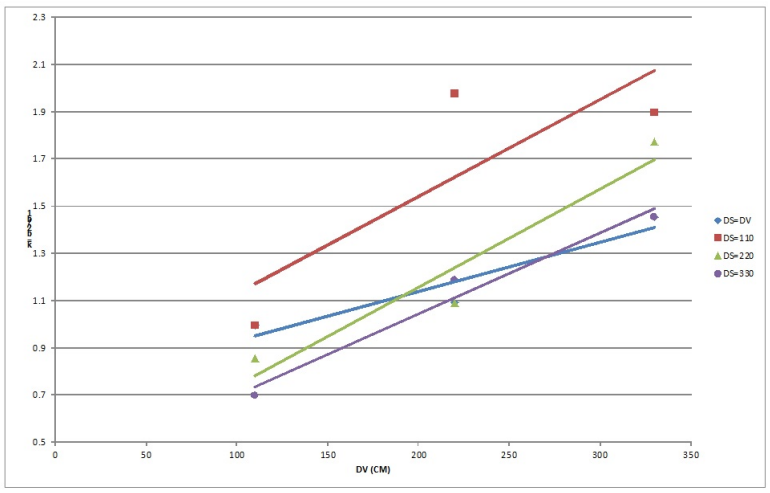

Figure 3.9: The three k-lines for $D_S = 110$ (red / squares), 220 (green / triangles) and 330 (purple / circles), along with the k-line for the non-VR $D_V = D_S$ conditions (blue / diamonds). To calibrate binocular depth using k values, the desired binocular depth (A) determines a k-value (B) on the blue $D_V = D_S$ line. That same k-value on the red $D_S = 110$ line (C) determines D'_V to be the corresponding binocular depth (D) that the software should use to insure the desired pointing performance if the screen is 110cm from the viewer.

Chapter 4

Incorporating Computer Vision for Latency Free Analysis of Pointing Data

4.1 Introduction

One of the potential concerns with much of our work has been the effect of latency on human pointing performance. By necessity, any computer based pointing system has a delay between when a physical movement is completed and when the system finishes processing and is able to render a response to the display. A number of papers have investigated this relationship between latency and pointing performance and shown it has a performance cost that increases as latency gets larger [31, 49].

One possible explanation for the variance of k as gain or depth is changed is the impact of latency becoming increasingly problematic as you move further back from the screen or make gain higher. We do not agree with this conclusion and hypothesize that k is reflective of the underlying physical sensorimotor processes. We therefore decided to investigate if this k variation remains clear when there is no significant latency between physical movement and response. Latency free evaluations of human motor performance are rare in HCI (where a computer is

expected to be in the loop), but more common in psychology and kinesielogy. In particular, Fitts' original study employed metal plates and conductive pens for pointing on a desktop, but to our knowledge no such test has been extended to distal pointing [9].

We think this is likely because performing such a test in a cost effective manner has only recently become feasible through computer vision techniques. The metal plates in Fitts' experiment could be easily connected to a circuit that records when it was tapped (and a circuit closed through the conductive pen). However, distal pointing requires some form of laser pointer and visual analysis. Video taken from participant interactions can be analyzed by the experimenter in post to obtain rough movement time results. However, this would require demanding frame by frame analysis for potentially dozens of hours of video. Furthermore, accurate physical measurements in the same units as the target sizes would be impossible, since all you can tell is the final pixel distances. These are always subject to noise and distortion. This naturally leads to the results from such analysis being biased and less trustworthy than intended.

We decided to tackle this problem and developed a computer vision system to process participant video in post and obtain cost effective accurate real world measurements of pointing performance. The next chapter describes the actual procedure and results of this experiment. However, we believe that the computer vision tools and techniques we employed can be broadly useful to other HCI practitioners for cost-effective supplementary analysis. This chapter is intended both as documentation of our technique and a primer to employing simple computer vision techniques that we believe can and should be employed in future HCI lab studies. The actual sub-steps and operations employed are all implemented by the standard computer vision library, OpenCV. OpenCV is licenced for free academic and commercial use, and we highly recommend it for research use. http://opencv.org/

4.2 Room Setup and Camera Placement

One of the key concerns in computer vision is of course, *what can the camera see?* The camera aperture can only capture so much of what is in front of them, so it is important that the entirety of the interactable area is within view. Furthermore, par-

ticipants can move and it is possible for them to occlude an interaction. No matter how complex a computer vision algorithm, it can not accurately recover data that was occluded and therefore not recorded. Triangulation from multiple viewpoints can be used to overcome occlusion, but that at least triples the data recorded, and requires a complex system to combine the results from multiple cameras. These are fundamental limitations to computer vision, and are still being researched in the field.

It is therefore not likely that any solitary HCI researcher will have the expertise and skills to solve these problems. They can also become major stumbling blocks to computer vision techniques being employed in field work or in-context user studies. However, lab studies are uniquely poised to take advantage of computer vision techniques because they are by design a highly controlled environment. It is common practice for lighting conditions to be tightly controlled, participant positions to be somewhat constrained and the range of interaction space limited. Furthermore you also have control over the display technology used and the room setup used to perform the study.

In our study, we specifically took advantage of the fact that the experiment was conducted on a rear-projected large screen display. An example of the camera placement is shown in Figure 4.1. This allows us to entirely eliminate the problem of participant occlusion. Simply put, participants stood on one side of the display and pointed the laser pointer at the screen. This scatters when it hits the translucent glass and appears as a bright red dot to cameras situated in the projection room behind the screen. Regardless of where participants move in the experiment space, the camera can always see the position of the targets and laser pointer. Targets would also be presented at fixed positions with fixed sizes set out before hand. A stable, high quality tripod was positioned behind the screen and used to adjust the camera placement until it could see a good area around the largest target positioned as far away from each other as possible.

4.2.1 Synchronizing Video with Experiment Conditions

Assuming you've solved the problem with camera placement and created a computer vision system that records pointing data, you still need to relate that to your

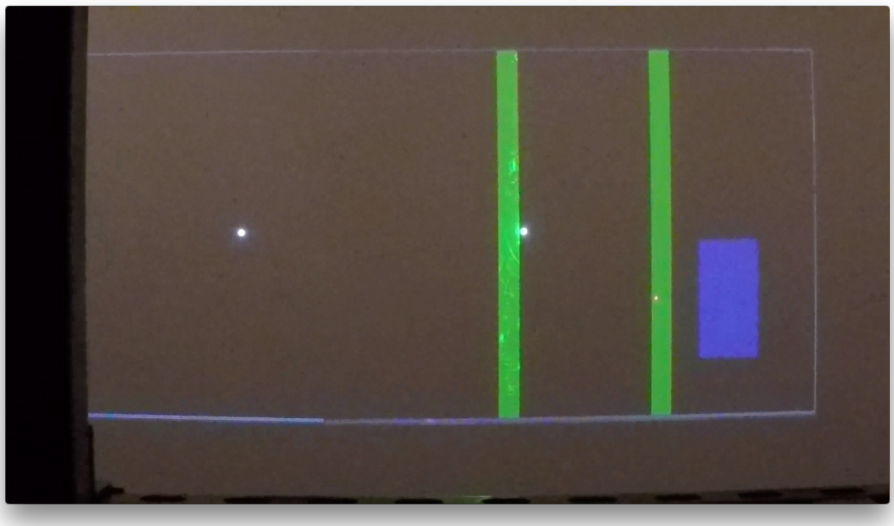

Figure 4.1: Example view of what is seen by the camera placed behind the screen. The green bars are the targets used in the pointing experiment. The blue box is used for camera calibration.

experiment. Video timesteps are usually only recorded with reference to the start of the video. It is therefore actually a non-trivial problem to decide which experiment condition your actually recording data for.

Theoretically the most reliable way would be to position some shape (e.g. a condition identifier) in some out of the way place on the screen. As you read in each frame of data, you have the vision system find and identify that shape and use it to decide which condition that data is for. However, while this is an easy task for a human, it is computationally complex. Likely it will involve implementing some fairly complex shape analysis such as scale invariant feature analysis. (for an introduction to such a technique, see http://docs.opencv.org/3.1.0/da/df5/tutorial_py_sift_intro.html#gsc.tab=0) This will greatly slow down the analysis and may also be subject to noise and mistakes. Such algorithms at best produce machine learning estimates which is undesirable for a tightly controlled lab study. Furthermore, a visible condition identifier may be something participants can decode to understand what conditions they are performing. This may be undesirable

in experiments where participant knowledge could skew results.

We found that the most effective and simple method was to have the experiment application and vision systems synchronize. The experiment application would display a text box with a welcome message before the experiment started. When a button was clicked to start the experiment, internal logging registered the timestamp of that click as the start of the experiment and removes the welcome message. The system then recorded a simple text file of the timestamps each individual condition was started relative to that text disappearing. When processing the video we would manually review the start of the video to find when the welcome text disappeared. That frame was used as timestep 0 for the analysis and then the relative timestamps from the text file were used to decide what condition each additional frame of data was for. This method may be subject to some amount of drift, but this was not noticeably problematic for a short 30 minute to one hour lab study.

4.3 Designing a Vision System to Record Pointing Data

Now we come to actually implementing our computer vision system. At a high level, the pointing experiment we want to analyze requires the time each movement takes and how far off center the selection was. Thus, the data our vision system needs to provide us with is as follows:

- The position of the cursor in any frame
- The shape and central position of the targets
- The time between the cursor "selecting" each target
- The offset (distance) between the cursor and the center of the target to be selected in centimeters (the scale used to model effective width)

The rest of the sections in this chapter will discuss some of the relevant theoretical reasons for using the approach we used, and provide key samples of code for implementing similar systems yourself.

4.3.1 Finding Moving Objects - Cursor Position

Finding a small moving target like the laser pointer is at once both the simplest operation of computer vision and one we found required the most fine tuning. Most edge detection or object recognition algorithms are designed to minimize the effect of camera noise. Noise can come from many sources but essentially makes the image look grainy with lots of small bright or dark spots. Even in a high resolution image the laser dot is by definition a small bright spot a few pixels across. These efforts to remove noise often remove the object you want to find entirely! Furthermore, feature detection algorithms often rely on smooth shading/texturing that comes from natural light interacting with a real object. Somewhat counterintuitively, we found a very simple "thresholding" operation was more applicable for our needs.

The high level idea behind thresholding is very straightforward, you decide what it is you are looking for (e.g. "find the bright red dot") and then decide what minimum and maximum colors would fit that criteria (through inspection with a color meter). OpenCV will go through each pixel and return whether or not that pixel meets the criterion. Then some relatively simple measures of central tendency tell you the X,Y position of the centre of mass for pixels that pass your threshold. There are however a number of complications to be aware of if implementing something similar.

Images read in from a video file by OpenCV are treated as two dimensional matrices of BGR (amount of blue, green and red light) pixel values. This is extremely difficult to threshold, since end color is a complex combination of the three channels. You might set a threshold looking for something with high red, but a dull red might not go through. Furthermore, the laser pointer color will also blend with the target color as it passes over the targets, which causes the color to shift and BGR thresholds to fail. Thus it is standard practice to perform thresholding operations after converting the image to HSV (hue - the color of the pixel, saturation - how much of the color there is, value - how bright that color is) format. HSV is much more robust to changes in lighting conditions and shading, and easier to formulate thresholds for. The following line of openCV code takes the image in the variable frame, and puts an HSV version of it in to the variable hsvFrame.

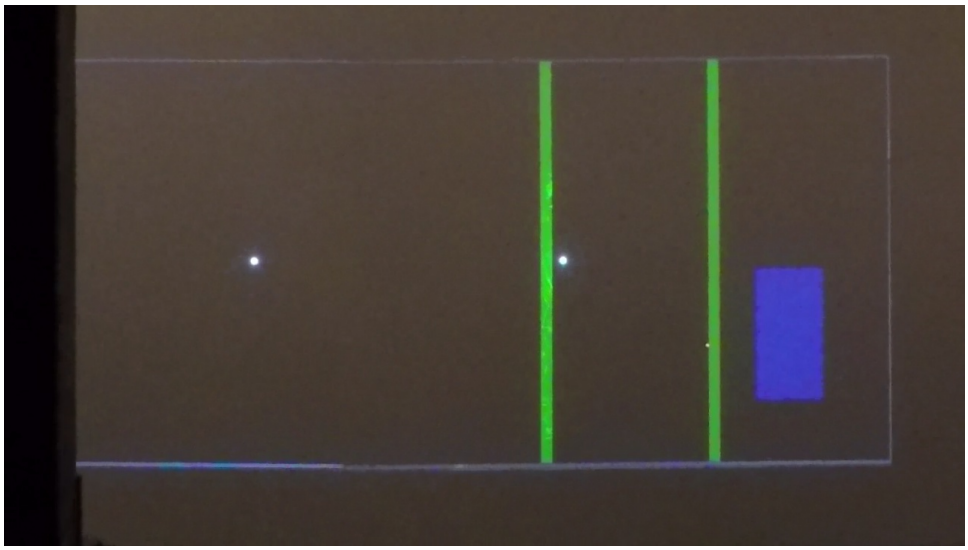

Figure 4.2: Starting image for finding the pointer and targets.

```
cvtColor(frame, hsvFrame, COLOR_BGR2HSV);
```

Next we perform the actual thresholding operation. We have predefined the array hsvPointer with the target threshold for this operation. The first and second elements are the minimum and maximum hues (color) the second and third are the min/max of saturation and so on. Thresholded frame will contain a two dimensional array of booleans where each cell represents whether the corresponding pixel passes the threshold.

```
inRange(hsvFrame, Scalar(hsvPointer[0],hsvPointer[2],hsvPointer[4]),
Scalar(hsvPointer[1],hsvPointer[3],hsvPointer[5]), thresholdedFrame);
```

We also found that occasionally noise became a problem, and we'd have random pixels pass the threshold and throw off the estimate of position. A median filter of size 3 was used to remove noise. In laymans terms, the following code replaces each pixel with the median of its immediate neighbors. If a noise spike causes one or two pixels to pass they will be completely removed from the result. Median filters are fantastic ways to remove noise, but you have to be careful to use a small filter size (three is the smallest possible size). Bigger filters will cause the image to blur. This will remove larger passed areas and cause the laser dot to disappear very quickly.

Figure 4.3: Example Result of the output from thresholding for the pointer. White spots are within the threshold and thus likely within the pointer.

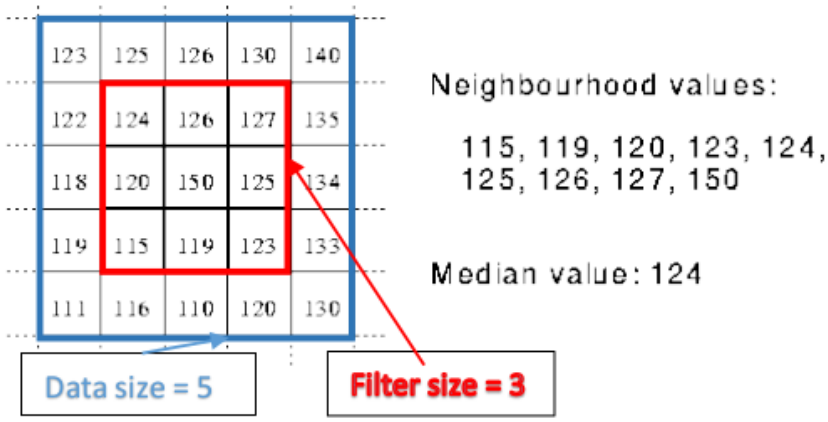

Figure 4.4: Example of a median filter operation. Image taken from http://tinyurl.com/j26e7cf

```
medianBlur(thresholdedFrame, pointerFrame, 3);
```

The last step in finding the laser dot is to find the central tendency. This is implemented by the moments library of openCV and the code snippet is shown below.

```
Moments oMoments = moments(pointerFrame);

double dM01 = oMoments.m01;
double dM10 = oMoments.m10;
double dArea = oMoments.m00;

int posX =dM10/dArea;
int posY =dM01/dArea;
```

PosX and posY should now contain the position of the cursor.

4.3.2 Finding Stationary Objects - Target Position

The next task in our vision system is finding the position and extent of the targets to select. In order to know whether the laser position is inside the target we need to know where the edges and center of the targets are. At its heart this is accomplished by a similar thresholding operation to the laser pointer selection. However there are a number of complications and further processing we need to accomplish.

The first interesting thing to note is that unlike the laser pointer, the targets are produced by computer projectors refreshing at 60 Hz. There tends to be some flicker and blank space between projector refreshes. The exact edges of the targets can be a little blurry and noisy. Also, we note that the camera and targets are stationary within a condition. This leads to the conclusion that we do not necessarily have to find the targets every frame. If we can get a robust estimate of their position in the first few frames, we can save a huge amount of computation for the rest of the condition.

We employ an averaging method for solving both of these problems. Essentially, at the start of every condition we threshold the image through the same process as the laser pointer. We maintain a count of how many times each pixel has passed the threshold successfully.

After a sufficient number of frames, we perform an additional threshold to get rid of pixels that only passed the threshold a few times. This lets pixels around the edge that are slightly noisy, but often still passing the threshold to be considered part of the target itself. The edge of these targets are still very jagged, so some image processing is used to smooth them out. A series of erosions (which cut out the jaggies on the edges) and dilations (which extend the edges back out to keep

Figure 4.5: View of the average thresholded result, edges are noisy, but the rough shape is there.

object size consistent) are used for that purpose.

```
erode(targetFrame, targetFrame, getStructuringElement(MORPH_ELLIPSE, Size(5,5)));
dilate(targetFrame, targetFrame, getStructuringElement(MORPH_ELLIPSE, Size(5,5)));
dilate(targetFrame, targetFrame, getStructuringElement(MORPH_ELLIPSE, Size(5,5)));
erode(targetFrame, targetFrame, getStructuringElement(MORPH_ELLIPSE, Size(5,5)));
```

Now that we have a smooth greyscale image of what pixels contain the targets, we need to decide where the edges and center of the targets exactly are. This is implemented by the OpenCV function, findContours. The following snippet puts a list of every contour into the variable contours. Most of the parameters can be handwaved, but note the parameter CV CHAIN APPROX NONE which makes the system store every point in the contour rather than compressing them.

```
findContours( *imageFrame, contours, hierarchy, CV_RETR_LIST,
              CV_CHAIN_APPROX_NONE, Point(0, 0) );
```

The last step is to iterate through the points within each contour and do some processing to find the maximum/minimum X and Y values to get the corners of the targets. The center is simply the average of their position. Also note that it is important to sort the contours by size in order to get rid of any noisy small contours.

```
for( int i = 0; i< contourInd.size(); i++ )
```

Figure 4.6: Example result for the target position after thresholding and processing the averaged passing pixels.

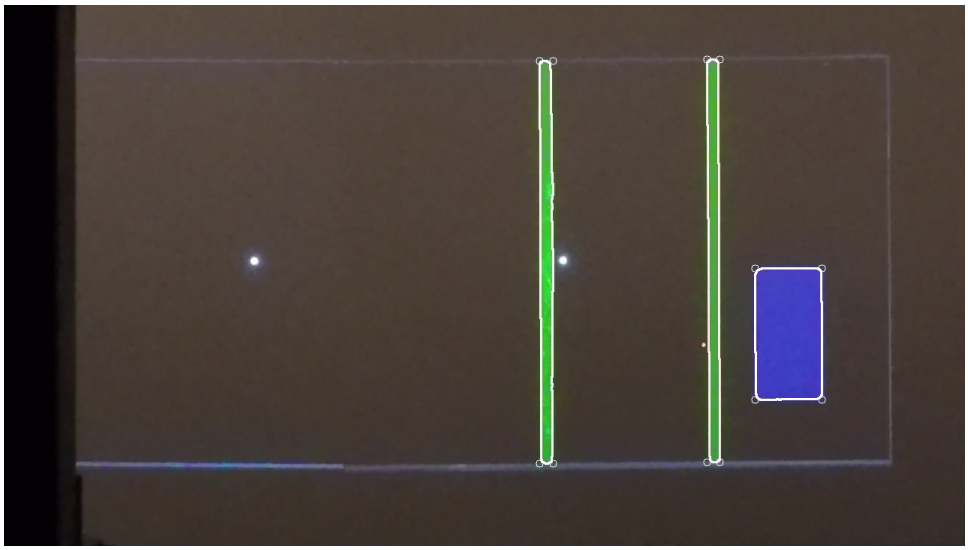

Figure 4.7: Example contours of the targets overlayed over top of the original example image. Circles are positioned at the corners.

```
{
    for (int j = 0; j +1 < contours[ci].size(); j++)
    {
        Point temp = contours[ci][j];
        minX = min(temp.x, minX);
        maxX = max(temp.x, maxX);
        minY = min(temp.y, minY);
        maxY = max(temp.y, maxY);
    }
}
```

4.3.3 Moving From Pixels to Real World - Offset

The last major piece of information the vision system needs to provide us with is the offset of the cursor from the center of the target. At this point it would be trivial to simply subtract the two pixel positions, but that would only give us results in camera pixel coordinates. The target sizes are all recorded in centimeters, so comparing a pixel offset to physical target sizes will be an unfair comparison at best. Pixel coordinates are also subject to perspective and intrinsic camera distortion, and thus cannot be trivially converted to the physical coordiantes required for accurate modeling. This is a well-understood problem in computer vision, and this section provides a brief description of our method to solve it.

The first step in this process is commonly called "camera calibration". Cameras, especially cheap ones are subject to mechanical inconsistencies, lens distortions and perspective projection. We can mathematically categorize these into two matrices which can be inverted to correct these errors [58]. The "intrinsic" parameters of the camera refer to the internal properties of the camera that are consistent for the same camera regardless of where it is positioned or what it is looking at. For example, this would include radial distortion from a fish-eye lens. There are also "extrinsic" parameters that refer to the cameras positional properties relative to objects in the scene (e.g. rotation, translation away ect). Put together, these two matrices form a camera "homography" which allows you to convert from a pixel location into physical positions relative to an object of known size and orientation [58].

Finding the intrinsics and extrinsics of a camera is a common problem usually completed by waving a grid of known size in front of the camera. We will

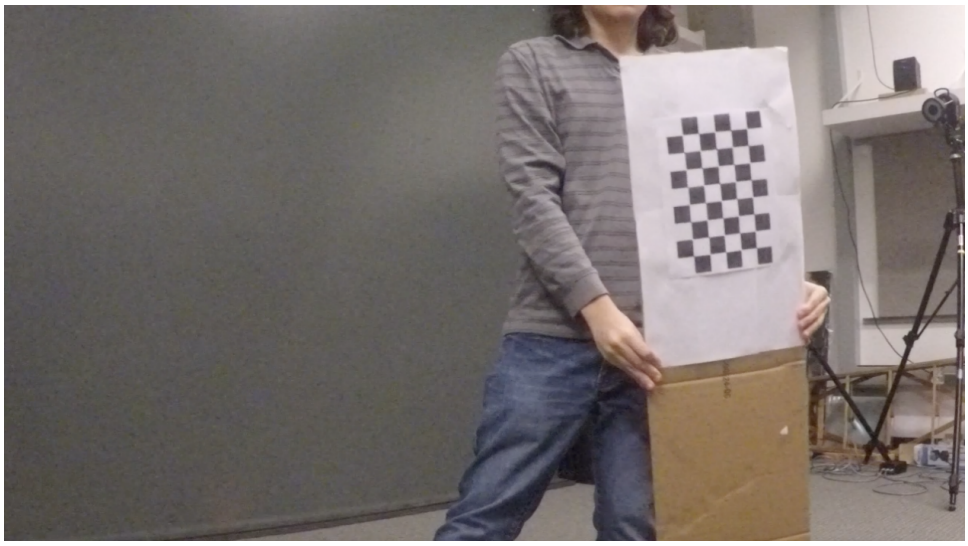

Figure 4.8: Example of the calibration procedure through grid wave. A regular grid of known size is pictured from various angles, machine learning finds the intrinsic distortions of the camera.

refer readers to the standard openCV tutorial for this procedure. Find this explanation and full code listing of calibrating a camera at http://docs.opencv.org/2.4/doc/tutorials/calib3d/camera_calibration/camera_calibration.html.

We used this as the basis for our camera calibration procedure, but made some additions and changes. For one, this process is extremely computationally expensive, (solving a linear system minimization over dozens of images) and would be infeasible to run on every frame of our video. We decided to do a robust camera calibration (Figure 4.8 showcases a picture from the calibration procedure) ahead of time to find good estimates of the intrinsics of the camera. These will be consistent for the camera regardless of position, so can be reused over and over without re-computing the homography. The following line of code corrects the intrinsic errors in the image. CameraMat and distortionCoeff come directly from the output file of the openCV camera calibration.

```
undistort(rawFrame, frame, cameraMat, distortionCoeff);
```

Once we have corrected for intrinisic distortion we need to correct for perspective/extrinsic properites and compute the physical distances in the plane of the

screen. To do this, we need a reference object of known physical size in the plane that we want to measure. In Figure 4.7 of the back of the screen you may have noticed the blue box beside the green targets. This is the reference object we use, and is of a clearly different and easy to find color from the rest of the scene. Before running the experiment we measured its final physical size in CM which was 29.3cm high and 15cm wide. We used the same averaging and thresholding process for finding this calibration object as the green targets. The following line of code takes the pixel X,Y values and compares them to the real world distances in centimeters to get the perspective matrix that converts between them.

```
img2World = getPerspectiveTransform(cornerPixelXY, cornerInCM);
```

Then it is a relatively trival operation to take the pixel location of any given object in the plane of the calibrator and multiply it through a matrix that inverts the projection to get X,Y relative to the bottom left corner of the calibrator. The following line of code tells openCV to populate worldPts with the physical position of the input imgPts.

```
perspectiveTransform(imgPts, worldPts, img2World);
```

4.4 Conclusions and Recommendations

Most experiments record data using internal software logging in their experimental software. While this is quick to implement and useful, Software logging can only capture the specific moments of interaction with the technology and entirely loses data about the preparatory phase. Video recordings of the interaction context are often employed for post-hoc data analysis and identification of key usability issues. However, manual video analysis is extremely time consuming and often reliant on inaccurate relative positions to landmarks (e.g. the person chose to move *away* from the *group* to check an incoming text message). We would argue that the vision techniques presented here could reduce analysis workload through automation while also providing more accurate real world measurements of distance. These can enrich our understanding of video data and can provide additional insight to standard machine logging. Future work should investigate creating a more general package for vision analysis of experiment context syncronized to software logs.

We are still a good distance away from having a generalizable tool that could be immediately applied to other research areas. By having a few main objects of clearly distinct primary colors we were able to easily identify them from the background and process their shape with just simple thresholding operations. While this works well after some parameter adjustment (while being extremely efficient), more complex visual scenes and experiments are going to require more than just simple thresholding operations, particularly when you are looking to find an object within a cluttered desktop interface.

For broader implementation that could be applied to more complex interactions we direct the user to the SIFT algorithm http://docs.opencv.org/3.1.0/da/df5/tutorial_py_sift_intro.html#gsc.tab=0. SIFT techniques (scale invariant feature transformations) are often used in combination with machine learning techniques in order to detect and identify known groups of objects of varying orientations and sizes. These techniques have been shown to be robust and helpful for identifying key features within a image that can be used to uniquely identify objects. Ideally, a standard set of these objects could be identified (cursors, buttons, mice, keyboards, subjects) and then combined into a software package that can identify relative real world positions over time. Some work has already been done in similar problems, but we know of no current tools that easily package this up for application in research yet. We consider this a strong project area for future research and application of vision technology.

Chapter 5

Real-World Pointing, Target Depth, and *k*

In this chapter we describe an experiment, similar to the one described earlier in Chapter 3, in which we further explore the variation of k in Welford models. The new experiment examines "real-world" pointing rather than computer-mediated pointing. The primary difference is that the new experiment involves no computer intervention: there is no feedback loop involving computer control. A secondary difference is that experiment 2 examines a larger set of target depths and it does not use binocular stereo to display the targets at virtual target depths: all target depths are physical (in the terminology of Chapter 3, $D_V = D_S$), which is why we consider the pointing in this experiment to be "real world" compared to the more synthetic pointing in the earlier experiment.

A computer was used in the both experiments to display targets, but the computer in the new experiment does this entirely independently of a human participant's movements and thus it is not "in the loop" with the human. A computer also recorded a participant's movements in both experiments, but in the new experiment the software described in Chapter 4 was used to capture movement data with a camera for subsequent analysis off-line. The computer's role in the new experiment is thus entirely passive: it presents stimuli independent of the human participant who is performing pointing actions, and it collects data about the human's actions but only for later analysis without providing any accuracy feedback

during the experiment. This constrasts with the earlier experiment where the computer actively tracked the human participant in order to determine where the human was pointing. This allowed it to flash the targets different colors to provide feedback about selection accuracy. Limiting the computer to a passive role removes any latencies or other artifacts that might have been introduced into the earlier experiment through a feedback loop between the human participant doing the pointing and the computer system mediating the experiment.

Experiment 1 investigated the impact of target depth on pointing performance and showed that Kopper's k-factor (the ratio of the impact of target width and movement amplitude on pointing performance) varies with target depth (the distance from a user to the targets). Evidence was found that increasing target depth increases k, which means that the width of targets has a larger effect on movement time relative to the distance between the targets (movement amplitude) as target depth increases. This is similar to the effect reported by Shoemaker et al. [40] that increasing gain also changes the relative importance of target width compared to movement amplitude for predicting movement time. For high gain, a pointer is more responsive to hand motion and thus can cross large gaps between targets easily, but it may have difficulty making sensitive fine adjustments once it gets reaches a target. As the depth of targets increases, the apparent speed of the pointer in the plane of the target increases relative to target planes that are closer. So even though there is no actual change in gain in terms of C:D ratio, there could be an apparent change in gain.

We focus our attention on three likely models for the impact of depth on k. A linear model has been suggested by our previous work and is consistent with the information theoretic interpretation proposed by Mackenzie et al. A logarithmic model may also be appropriate to capture a slow tail off, as increasing depth may have diminishing returns on k. A polynomial model adds another degree of freedom to the linear model and may also capture a trail off over depth. We feel that any more complex models will likely not be worth investigating as more degrees of freedom will come closer and closer to purely interpolating our six data points rather than predictively modeling it.

It is possible that the observed increased impact of width on pointing performance as target depth increases is an artifact of system latency rather than being

purely due to target depth. Latency is the delay between physical movement and the resulting feedback being displayed on the screen. Latency has been shown to have a performance cost of around 10-15 percent per 100ms of latency [31, 49]. It is possible reasonable that a delay between input and visual feedback in the form of a cursor displayed on the screen (which is how the experiment in Chapter 3 was conducted) might be more problematic when trying to do fine adjustments for small targets rather than a rough ballistic trajectory.

Testing whether the k variation is an artifact of the computer mediation or is instead a true sensorimotor pattern was one of the goals experiment 2. Examining a larger range of target depths is another goal. The experiment in Chapter 3 used a limited set of target depths and saw a trend of increasing k with target depth. We would like to determine if it is a steady linear increase, or if it has some other pattern. The design of the experiment presented in this chapter uses a larger set of target depths in order better tease apart the relationship between target depth and k for a more nuanced understanding that might lead to practical calibration procedures for virtual environments based on measuring k at various depths (this will be explored further in Chapter 6.

5.1 Method

We investigate the effect of physical target depth D on pointing performance (movement time MT) in a distal pointing task. We examine the same physical target depths as did the experiment previously described in Chapter 3 plus additional depths intermediate between those previously used and also depths further away to extend the range of our earlier results. A more important difference is that this experiment did not have active computer processing within the feedback loop. Instead, stimuli were presented on a fixed timeline, independent of pointing activity by a participant. The only visual or other feedback is that produced by a hand-held laser pointer that is not connected in any way to the computer that displays the stimuli. This approach was adopted to remove the possibility of feedback latency or other temporal artifacts being introduced by the hardware and software used to control the experiment. Taking these two differences into account (the larger number of target depths and the elimination of computer-mediation that might in-

troduce latency artifacts), the experiment is otherwise very similar to experiment 1. In particular, the experimental task was again a Fitts-style one degree-of-freedom reciprocal tapping task between two target pairs, modeled closely after Shoemaker et al. [40] and the original experiments by Fitts [10]. ISO 9241-9 [3] defines a two degrees-of-freedom task for pointing performance; we used a one degree-of-freedom task because we were concerned with the fundamental applicability of Fitts's Law. Although the task was one degree-of-freedom, it is still an example of three-dimensional pointing because the interaction setup is identical to distal pointing which is common on stereoscopic 3D.

5.1.1 Participants

We recruited 20 participants at our university through on-campus advertising. As a requirement of participation, all were right-handed with normal or corrected-to-normal vision. Two participants' data had to be discarded due to equipment malfunction (the batteries on a laser pointer ran out). Of the remaining 18 participants, 14 were female, 3 male and one indicated "other" for gender when completing a pre-experiment questionnaire. Age ranged from 19 to 29 years (mean 23.9 years). All participants were regular computer users (9+ hours per week). The experiment was approved by the Behavioural Research Ethics Board at our university (certificate H11-01756). Participants signed a consent form prior to the start of the experiment. Each was compensated with $10.

5.1.2 Apparatus & Materials

The hardware and software used for the experiment were similar to that used in experiment 1 except that a laser pointer was used instead of a software-based pointer and a camera subsystem captured continuous video of the screen that was subsequently analyzed to provide movement data instead of computing the movement data in real-time during the experiment using Vicon motion capture sensors.

Hardware

A 5.16m$\times$2.85m (width $\times$ height) wall display with a glass screen was rear-projected by two side-by-side 1024$\times$768px projectors, giving a total resolution of 2048$\times$768px.

Figure 5.1: Apparatus: Pointer used for interaction and camera used for data recording.

The display was driven by an 8-core Intel Processor with 6GB of RAM and dual NVIDIA GeForce GTX 260 graphics processors, running Windows 7. The projectors were carefully aligned to reduce visual discontinuities between projectors. Unlike the experiment in Chapter 3, no stereo projection was used and thus the two projectors were driven directly from the computer rather than through a video processor. A standard frame rate of 60Hz was employed.

Figure 5.2: Image of the projection room behind the screen. Camera is placed on tripod at fixed location to record experimental data. Two projectors were used to display the experimental stimulus.

A GoPro Hero 4 Black Edition camera was mounted on a stable tripod behind the screen of the wall display and positioned so the entire interaction area on the

screen was within view. The camera was chosen because it could serve as a relatively low-cost, high-frame-rate data recorder. Video was recorded during trials at 240 frames per second with a resolution of 1280×720px. The narrow lens setting on the camera was used to reduce possible distortion due to the normal fish-eye effect of the GoPro lens (distortion was further corrected in post-processing software through camera calibration).

Software

There were two programs used for the experiment. One program, the experimental software, was a modification of the experimental software used for experiment 1. It was run during the experiment to display stimuli for the trials and to capture time-stamped data for subsequent analysis. Data comprised log files describing the stimuli and video files of the screen on the wall display that captured both the stimuli and the image of the laser pointer that was being used for pointing. The second program was post-processing software that used low-level computer vision techniques to produce time-stamped coordinate data on the position of the center of the image of the laser pointer relative to the targets in the stimuli being displayed by the software.

The experimental software displayed to a participant the stimuli for the various conditions in the experiment. It was written in C# using the Microsoft XNA Game Studio 4.1 library. The stimuli were two green bars to represent a pair of equal-width targets in a one degree-of-freedom Fitts-style reciprocal tapping task. Target width depended on the trial. The left target was always directly in front of a participant. The distance to the right target was determined by the movement amplitude for the trial. A 29.3cm by 15cm blue rectangle was displayed at a constant position just to the left of the left target to give the vision system a calibration object. Experiment 1 had rendered similar stimuli using non-headtracked stereo VR. The virtual camera would render the targets at different sizes (according to projection) depending on how far the participant was away from the screen. Since we were concerned with the fundamental sensorimotor processes, the targets presented by the software in this task were always rendered from a consistent viewpoint and thus at consistent size and position regardless of where the participant stood. Any

apparent change in size of targets was thus due solely to the participant standing further away from the display. The software recorded timestamps for the start of each trial (the end of a trial was the start of the next trial).

Pointing data was produced by post-processing the video images acquired during the experiment by the experimental software. The timestamps for each trial were fed into the post-processing computer vision software to synchronize the visual images captured of the laser pointer and the target stimuli with the movement amplitude, target width, and target depth parameters for the trial. The post-processing software calculated the movement time and amplitude (horizontal location) of all tapping events (defined as the peak distance before a direction change in the movement) and recorded them into a log file. The computer vision techniques used in the post-processing software are described in detail in Chapter 4.

Physical Layout, Task, and Stimuli

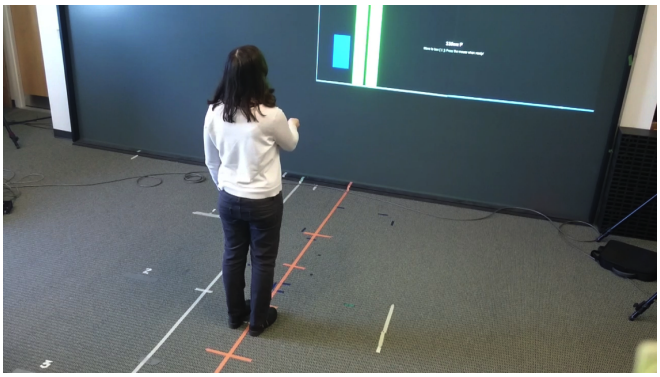

Figure 5.3: Experimental stimuli. Participant stands at a fixed target depth along the orange line. Specific conditions are marked with crosses. The experimental software displays two green targets on the screen. The user smoothly moves the laser pointer between the two targets.

Participants stood at one of six target depths D (110cm, 165cm, 220cm, 275cm, 330cm, and 365cm) from the wall display screen for a block of trials. 365cm was chosen instead of 385cm (which would have followed the 55cm increase pattern) because we were running out of space in the room. All participant locations were along a straight line perpendicular to the plane of the wall display and were marked

on the floor with tape beforehand. Every trial within a block had a pair of identical vertical bars (targets) displayed on the screeen separated by one of three movement amplitudes A (25cm, 50cm, 75cm), both targets having the same widths chosen from one of three target widths W (5cm, 10cm, and 20cm). Targets were 90cm high, roughly centered at a typical participant's elbow level. The left target was always directly ahead of the participant and the right target was to the participant's right. This was done to avoid a participant having to move an arm across the body midline while pointing because the cross-lateral inhibition effect (CIE) [19, 38] predicts that hand movements that cross the midline of the body are more complex than those that do not. Participants held down the button on a standard laser pointer to point at the targets. The distance between the centers of the two targets was the movement amplitude for that trial. Lighting in the room was kept consistent across trials by making sure that all sources of natural light (windows, doors) were blocked off, and that the same artificial lights were always turned on.

5.1.3 Procedure

Each participant performed the experiment in a single session of approximately 30 minutes. After filling out a consent form and a pre-questionnaire that gathered demographic information, a participant was introduced to the apparatus and the pointing task. Participants were instructed to complete the task as quickly as possible while being as accurate as possible. At the start of each block of trials, a participant was instructed to stand at the location marked on the floor that corresponded to the target depth D that was used for all trials in the block. Participants were then invited to complete a practice trial by tapping between the first two targets before data recording began. They were allowed to practice until comfortable with the system. Most participants quickly indicated they were ready to begin the experiment trials. Data collection was begun by the experimenter once the participant had completed the practice trial.

Each experimental trial was defined by three independent variables: movement amplitude A, target width W, and target depth D, the distance at which the participant had been placed relative to the screen for a block of trials. For the first target pair in a block, a participant tapped the left target that was directly ahead

and then performed a sequence of reciprocal back-and-forth taps between the right and left targets. Participants were instructed to keep the laser pointer smoothly moving between targets and continually tap between them. Subsequent trials were run continuously, without any break between trials. Every ten seconds a new trial began and the next target condition (A and W) was displayed, regardless of how many taps had been performed in the previous trial. The first tap in a trial was discarded because it was assumed that participants would have to adjust to the new condition while in the middle of a movement and thus that tap might not be representative of the participant's sensorimotor performance as it is assumed for a Fitts-style reciprocal tapping task.

Between each block of the experiment participants were required to take a break of at least three minutes. They were encouraged to take more time if desired, but very few did. The participant was then repositioned to the target depth D for the upcoming block and a practice trial followed by the experimental trials for the block commenced.

Timestamps and locations on the screen for all taps were recorded for subsequent post-processing. No accuracy feedback was provided during the experiment other than the image that the laser pointer made on the screen. Participants were not required to correct errors. They were told to keep smoothly moving between targets.

Measures

Pointing performance was measured by the time taken to execute each individual tap action. Participants tapped as many times as possible within the 10 seconds given for each trial. For each tap, the post-processing software recorded movement time MT, and the offset of the cursor from the center of the target when the tap (reversal of direction) was made.

Experimental Design

The within-participants design had independent variables of target depth (distance of the screen from the participant) D (110cm, 165cm, 220cm,275cm, 330cm, 365cm), target width W (5cm, 10cm, 20cm), and movement amplitude A (25cm, 50cm,

75cm). All three variables were fully crossed.

$\boxed{\text{18 participants } (N)} \times \boxed{\text{6 target depths } (D)} \times$

$\boxed{\text{3 movement amplitudes } (A)} \times \boxed{\text{3 target widths } (W)}$

For a given D a participant performed all trials for every combination of A and W before switching to another D condition. We decided against intermixing the D conditions to avoid participants having to repeatedly move back and forth between different locations in the room. The presentation of target depths D was counter-balanced across participants according to a balanced latin square. The presentation of movement amplitude and target width pairs A-W conditions was randomized within every block corresponding to a fixed target depth D.

Hypotheses

The hypotheses for this experiment are based on previous studies. Most directly, they are drawn from the previous experiment that was described in Chapter 3.

H1a (*weak*) the parameter k will increase monotonically with greater target depth

H1b (*strong*) the parameter k will increase linearly with greater target depth

H2 One-part models of pointing performance will not accurately model all target depths

H3 Two-part models of pointing performance will perform better than one-part models in conditions where k diverges from 1.

H4 Angular measures of target difficulty will not improve our models of pointing performance.

The first hypothesis is again split into weak and strong variants, as it was for experiment 1. We expected to find further support that k varies monotonically with target depth, but also felt that there might be evidence that it varies linearly because of the additional target depths that could provide a better picture of how k behaves

as target depth varies. By removing latency as a confound we hoped to support both variants of the hypothesis better than in the previous experiment.

The second and third hypotheses are based on our conclusions in the previous experiment about one-part and two-part models of pointing performance, which were that a Welford-style two-part formulation is more robust and will, at least for some conditions, better predict movement time than will a Fitts-style one-part models.

The fourth hypothesis arose from the post-hoc analysis in the previous experiment that indicated that using angular measures instead of classic linear measures for A and W does not improve modeling strength.

5.2 Results

Our analysis proceeded in four steps after removing outliers from the data. We conducted an ANOVA on movement time for the three independent variables. We then compared two Fitts-style and two Welford-style models using a regression analysis for each of the four model types for each target depth D using both target width W and effective target width W_e. An F-test was conducted on four sets of pairs of these models (each a Fitts-style model and the corresponding Welford-style model) to determine whether the additional degree of freedom in the Welford-style models was justified by our data. The model comparison was repeated using the angular measures corresponding to A and W to determine whether the classic linear measures or the angular measures advocated by Kopper et al. [23] are most appropriate for analyzing pointing performance. We then examined the behavior of the parameter k in the Welford-style models as a function of target depth D. A Shapiro-Wilk test for normality was conducted on the participant average movement times and showed that data was unlikely to be normally distributed ($p < 0.001$) which may have been caused by outliers and may degrade the effectiveness of our models. We thus attempted to remove outliers before begining our analysis. Overall average movement times and 95 percent confidence intervals for each condition are included in the Appendix A.

5.2.1 Removal of Outliers

There were a total of 38,307 observations across the 18 participants whose data was analyzed. As is common practice in pointing evaluation, mistrials and outliers were removed from the data before analysis. Any tap more than half the trial amplitude away from the target (likely to be caused by a premature reversal of movement direction) was ignored and any tap whose movement time was more than three standard deviations from the mean for the participant's performance in that A and W condition was also ignored.

It was discovered that for two conditions (movement amplitude A being 75cm and target width W being either 10cm or 20cm) the recording of taps for the left target was less accurate than for taps in other conditions. Consequently, all of these trials were removed to avoid skewing the results. Taps on the right target in these two conditions were recorded accurately, so were included in the analysis.

In total, 3881 taps (10.1%) were removed from the data. The remaining taps were used in all of the analyses, except when calculating the models discussed in Section 5.2.5 where additional points were excluded for reasons that are discussed in that section.

5.2.2 ANOVA for Movement Time *MT*

The results of an ANOVA for movement time MT with independent variables movement amplitude A, target width W, and target depth D are presented in Table 5.1. We used the R Library's ez package [25], which utilizes Type-two sum of squares error. This sometimes differs slightly from SPSS and other packages that use Type-three sum of squares error for ANOVA.

As expected, movement amplitude ($F_{2,34} = 114.729$, $p < 0.001$) and target width ($F_{2,34} = 116.62$, $p < 0.001$) had strong effects. That target size and position effects pointing performance is fundamental to any study of pointing and thus is not surprising. There was also an interaction between amplitude and width ($F_{4,68} = 7.443$, $p < 0.001$). The non-linear relationship between these variables and movement time in both Fitts-style and Welford-style models predicts this.

What is surprising is that target depth D did not have a effect ($F_{5,85} = 1.553$, $p > 0.05$). This conflicts with the findings of experiment 1 showing that target

Factor	ε	df_{effect}	df_{error}	F	p	Partial η^2
A		2	34	114.729	**.000**	0.871
W		2	34	116.62	**.000**	0.873
D		5	85	1.553	.182	0.084
$A \times W$		4	68	7.443	**.000**	0.305
$A \times D$		10	170	0.688	.734	0.039
$W \times D$		10	170	2.009	**.035**	0.105
$A \times W \times D$		20	340	1.011	.448	0.056

Table 5.1: ANOVA results for the impact of depth (D), amplitude (A) and width (W) on movement time (MT). Statistically significant factors are bolded.

depth had an effect, which was the basis for identifying a trend of k increasing with target depth. There was, however, an interaction between target width and target depth ($F_{10,170} = 2.009$, $p < 0.05$). The impact of target width thus depends in part on target depth.

5.2.3 Comparing Pointing Models

To assess the relative ability of one-part and two-part models to predict pointing performance, we performed many of the same model comparisons described in Chapter 3. We conducted regression tests for a set of pointing models at each of the six target depths (110cm, 165cm, 220cm, 275cm, 330cm, 365cm) and one global model that aggregated data for all target depths. The results of the regression tests are summarized in Table 5.2.

Overall, our models of pointing performance seem quite strong. All but one of the pointing models reached or exceeded Mackenzie's 0.9 R-squared threshold for a good model. The only case that failed was at 110cm from the screen with regular width. Most of the time, one-part and two-part models performed comparably, but for regular width at target depth 165cm, a Welford two-part formulation was statistically better than Fitts one-part model and at both target depth 110cm and target depth 165cm a Shannon-Welford two-part model was statistically better

Width	D	Fitts				Welford						F-test		
		a	b	R^2	RSS	a	b_1	b_2	k	R^2	RSS	F-ratio	p	sig?
W	110	58.39	147.33	0.94	12282.57	-132.81	172.35	131.25	0.76	0.95	8317.43	2.86	0.12	no
	165	126.96	128.97	0.92	11903.79	-143.19	164.33	106.25	0.65	0.97	3987.97	11.91	0.007	yes
	220	131.89	137.09	0.94	10884.38	-16.06	156.46	124.65	0.8	0.94	8509.91	1.67	0.227	no
	275	115.65	130.75	0.94	8836.39	1.85	145.65	121.18	0.83	0.94	7431.56	1.13	0.315	no
	330	79.9	138.22	0.97	5508.86	-40.12	153.93	128.12	0.83	0.97	3946.38	2.37	0.157	no
	365	73.63	148.51	0.97	5767.75	-53.18	165.11	137.85	0.83	0.98	4023.48	2.6	0.141	no
	all	97.74	138.48	0.94	74522.95	-63.92	159.64	124.88	0.78	0.95	57515.69	15.08	0.0002	yes
W_e	110	-3.87	184.22	0.87	24966.37	-168.02	201.67	164.32	0.81	0.87	22097.87	0.78	0.400	no
	165	59.97	168.89	0.9	15223.05	-177.38	193.71	139.6	0.72	0.93	9068.54	4.07	0.07	no
	220	56	179.12	0.9	17110.21	22.29	182.38	-174.51	0.96	0.88	16994.53	0.041	0.844	no
	275	54.33	169.41	0.91	13466.63	-39.47	179.33	158.1	0.88	0.91	12486.13	0.47	0.509	no
	330	7.94	186.03	0.96	7364.14	30.58	183.93	189.19	1.03	0.95	7311.93	0.042	0.841	no
	365	20.77	182.7	0.94	10762.11	52.79	179.45	186.79	1.04	0.94	10662.13	0.056	0.818	no
	all	20.77	182.7	0.94	10762.11	52.79	179.45	186.79	1.04	0.94	10662.13	2.22	0.141	no

Width	D	Shannon-Fitts				Shannon-Welford						F-test		
		a	b	R^2	RSS	a	b_1	b_2	k	R^2	RSS	F-ratio	p	sig?
W	110	-88.55	185.4	0.95	9090.23	-328.18	220.23	174.59	0.79	0.97	4253.28	6.82	0.028	yes
	165	4.6	159.82	0.91	14501.23	-295.39	203.41	146.28	0.72	0.95	6920.78	6.57	0.031	yes
	220	-2.52	171.61	0.94	10117.21	-181.32	197.59	163.54	0.83	0.95	7424.31	2.17	0.174	no
	275	-12.88	163.8	0.95	7859.08	-153.73	184.27	157.45	0.85	0.95	6187.94	1.62	0.234	no
	330	-55.53	172.98	0.97	4798.7	-202.2	194.29	166.36	0.86	0.98	2986.68	3.64	0.088	no
	365	-69.76	185.02	0.96	6916.3	-215.75	206.24	178.44	0.87	0.97	5121.14	2.10	0.181	no
	all	-37.44	173.1	0.94	72886.91	-229.43	201.01	164.44	0.82	0.95	54258.17	17.5	0.0001	yes
W_e	110	-207.44	242.07	0.91	18094.92	-401.61	265.95	226.15	0.85	0.91	14812.89	1.32	0.278	no
	165	-113.14	216.6	0.9	15365.83	-367.71	247.32	194.83	0.79	0.93	9535.44	3.66	0.088	no
	220	-135.69	232.99	0.93	12708.29	-169.39	236.76	229.58	0.97	0.91	12610.51	0.046	0.834	no
	275	-117.94	216.73	0.93	11385.81	-229.69	230.33	207.46	-0.9	0.92	10249.86	0.66	0.435	no
	330	-188.53	241.26	0.97	4329.47	-169.09	239.15	243.28	1.02	0.97	4297.3	0.044	0.837	no
	365	-157.28	230.16	0.94	10912.82	-121.72	226.03	233.45	1.03	0.94	10810.64	0.057	0.817	no
	all	-157.28	230.16	0.94	10912.82	-121.72	226.03	233.45	1.03	0.94	10810.64	2.81	0.100	no

Table 5.2: Modeling movement time (ms) using Fitts and Welford formulations (top) and Shannon-Fitts and Shannon-Welford formulations (middle) for actual width W and effective width W_e. For each D condition and for all D conditions combined. The coefficients and adjusted R^2 for each model with the F-ratios, p-values and significance from an $F(1,9)$-test (Eq. 2.9 with $p_1 = 2$, $p_2 = 3$, and $n = 9$) in the last three columns compare nested Fitts and Welford (or Shannon-Fitts and Shannon-Welford) models.

than a Shannon-Fitts one-part model. No differences were found using effective width, although there was a potential trend ($p < 0.1$) at target depth 165cm. The global model that aggregated all target depths showed significant improvements over a Fitts-style model for both Welford and Shannon-Welford using width, but no differences were found for effective width.

5.2.4 Testing Angular Measures of Target Difficulty

D_S	Width Type	Classic Fitts	Ang Fitts	Classic Welford	Ang Welford	Classic Shannon	Ang Shannon	Classic Shannon-Welford	Ang Shannon-Welford
110	W	0.938	**0.943**	0.951	0.951	0.954	**0.962**	0.975	**0.9788**
165	W	0.923	**0.928**	0.969	0.969	**0.905**	0.895	**0.947**	0.923
220	W	0.936	**0.939**	0.942	0.942	**0.941**	0.939	**0.949**	0.943
275	W	0.943	**0.944**	0.944	0.944	**0.949**	0.948	**0.953**	0.950
330	W	0.967	**0.968**	0.972	0.972	**0.971**	0.967	**0.979**	0.972
365	W	0.970	**0.971**	0.975	0.975	**0.964**	0.954	**0.969**	0.955
all	W	0.937	**0.938**	**0.951**	0.941	**0.938**	0.935	**0.953**	*0.93*
110	W_E	0.874	**0.879**	**0.870**	0.869	0.908	**0.929**	0.913	**0.930**
165	W_E	0.901	**0.908**	0.931	0.931	0.900	**0.901**	**0.928**	0.917
220	W_E	0.900	**0.901**	0.884	**0.885**	0.926	**0.937**	0.914	**0.926**
275	W_E	0.913	**0.915**	0.906	**0.907**	0.927	**0.931**	0.923	**0.926**
330	W_E	0.956	0.956	0.949	**0.950**	0.974	**0.980**	0.970	**0.977**
365	W_E	0.945	0.945	0.936	**0.937**	**0.944**	0.939	**0.935**	0.929
all	W_E	0.913	0.913	**0.915**	0.912	0.926	**0.929**	**0.929**	0.928

Table 5.3: Quality of fit for regression analyses as determined by the R-squared values is shown for angular and classic linear measures of movement amplitude A and target width W, and for movement amplitude A and effective target width W_e. Bold cells show which model in a pair of columns has a better R^2. The left columns are for classic and right for angular.

We examined whether one-part pointing models are improved if A and W are determined by angular measurements. Kopper et al. [23] suggest that this is a more accurate way to model pointing. We recalculated the regression coefficients for all of the models after replacing A and W with there angular equivalents, α and ω. Table 5.3 shows this comparison for each target depth and for the "global" model that does not treat target depth as a parameter but instead aggregates the data for

all target depths. W

We see mixed results in terms of improvements using angular measures for individual target depth models using width W. One-part Fitts models were slightly improved by angular measures, whereas the Shannon-Welford and Shannon one-part performed better with classic measures. The Welford models had almost idencical R-squared values. The patterns are less clear in effective width W_e where some models were improved by angular and others were made worse even for the same type of model. If we look more carefully at the magnitude of these differences for various target depths, we see they are fairly minor. The biggest improvement was for the one-part Shannon at target depth 110cm with effective width. Classic linear measures produced an R-squared of 0.908 but angular measures produced an R-squared of 0.929. The angular model accounted for two percent more of the variation in the data, which is arguably a relatively minor difference given that both models were above Mackenzie's 0.9 threshold for a "good" model.

The best case scenario for angular measures is for the global model shown in the final "all" row in Table 5.3 for both the width-based models (upper rows in the table) and the effective width models (lower rows in the table). In this case we see very little difference for one-part models, but for two-part models angular measures are worse in this condition. Regular width Shannon-Welford had an R-squared of 0.953 with classic linear measures compared to an R-squared of 0.938 for angular measures. This loss was similar in magnitude to the best case angular improvement achieved in the individual target-depth models (0.024 for target depth 165 versus 0.020 for the aggregated depth).

5.2.5 Modeling k as a Function of Target Depth

A fundamental goal of this experiment was developing a robust interpretation of how k varies with target depth. We examined three potential models of k as a function of target depth. The first is a simple linear function of target depth, which we expect to be a reasonable estimation that may not fully capture the trend. We think it likely there might be some point at which increasing target depth no longer increases k, or at least not as much as a linear model would predict. Thus we also looked at logarithmic and quadratic polynomial models as alternatives that

might better capture the relationship between k and target depth. We felt these were the most reasonable models to consider because our experiment only has six data points from which to verify a model. Higher order polynomials will quickly approach pure interpolation of the data rather than predictively modeling it. We looked at models of k for both regular and effective width. These results are shown in Table 5.4. For a visual comparison of these models refer to Figure 5.4 for regular width and Figure 5.5 for effective width.

Pointing Model	Width Type	k Model	k Equation	R^2
Welford	W	Linear	$k = 0.0005D + 0.6543$	0.503
Shannon-Welford	W	Linear	$k = 0.0004D + 0.7115$	0.600
Welford	W_e	Linear	$k = 0.0011D + 0.6428$	0.710
Shannon-Welford	W_e	Linear	$y = 0.0008D + 0.7220$	0.711
Welford	W	Logarithmic	$k = 0.1080ln(D) + 0.1988$	0.448
Shannon-Welford	W	Logarithmic	$k = 0.0906ln(D) + 0.3285$	0.544
Welford	W_e	Logarithmic	$k = 0.2223ln(D) - 0.2973$	0.643
Shannon-Welford	W_e	Logarithmic	$y = 0.1725ln(D) - 0.0081$	0.650
Welford	W	Polynomial	$k = 0.0000006D^2 + 0.0002381D + 0.6844$	0.506
Shannon-Welford	W	Polynomial	$k = 0.0000004D^2 + 0.0002501D + 0.7311$	0.602
Welford	W_e	Polynomial	$k = 0.0000022D^2 + 0.0000002D + 0.7540$	0.725
Shannon-Welford	W_e	Polynomial	$k = 0.0000016D^2 + 0.0000553D + 0.8023$	0.725

Table 5.4: Regression modeling of how k varies as target depth D (cm) changes for each of the four pointing models.

Overall the results of our modeling are mixed. The values obtained for k are lower than in our previous studies. There is also more variance in the data. Using regular width, a simple linear model is more effective than a logarithmic model, with an R^2 around 0.503 compared to 0.448. The polynomial model is slightly better than linear at 0.506. Looking at the equation ($y = 0.0000006x^2 + 0.0002x + 0.6844$), the polynomial model assigns much less weight to the quadratic term and thus is actually approaching linearity. All of these are worse fits than in the previous experiment where a linear model using width had an R^2 upwards of 0.9. Using effective width is often considered a truer reflection of user performance, and our models of k also seem to improve using effective width. A polynomial trend is the best model, but still only reaches an R^2 of 0.725. Overall, these models seem much less effective than our previous experiment, which showed k varying linearly with target depth with an R^2 between 0.89 and 0.95.

Taking a closer look at the specific points being modeled, we see that the 165

Figure 5.4: Visual comparison of different models for the relationship between k and D. Data presented for regular width models of pointing performance.

target depth condition has a lower k than the 110 target depth condition (0.64 compared to 0.76), whereas the rest of the points roughly produce similar or higher k values as target depth increases. This runs counter to the trend in the rest of our data, and is likely pulling the model away from the trend of the other points. For the sake of argument, we highlighted this as a potential outlier and attempted the same models on the rest of the points. Table 5.5 shows the modeling results on the set of points without the 165cm outlier, while Figure 5.6 and Figure 5.7 show this trend visually.

After dropping this one point, the modeling results are uniformly improved in all conditions. This is not surprising. The overall variance should decrease when an outlier is removed. However, especially with regular width we see dramatic

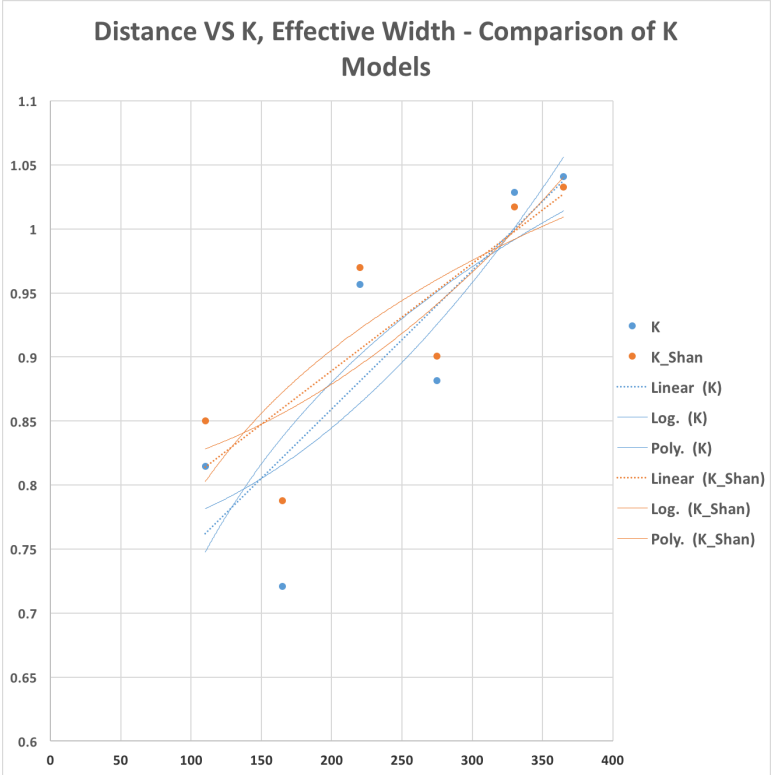

Figure 5.5: Visual comparison of different models for the relationship between k and D. Data presented for effective width models of pointing performance.

improvements in the modeling. Welford's regular width was very effectively modeled by a linear estimation of k (R^2 = 0.91) but was further improved using both logrithmic (R^2 = 0.94) and polynomial models (0.97). These models account for over ninety percent of the variation in k, compared to only fifty to sixty percent before the one outlier is removed, and they align nicely with behavior of k in our previous experiment. The existence of such dramatic outliers does cause concern. How confident are we of the monotonic (or even linear) relationship? We do not know how often will outliers become a problem because we do not know what caused this one. We discuss the implication of this and other aspects of our results in the next section.

Pointing Model	Width Type	k Model	k Equation	R^2
Welford	W	Linear	$y = 0.0003x + 0.7319$	0.915
Shannon-Welford	W	Linear	$y = 0.0003x + 0.7645$	0.955
Welford	W_e	Linear	$y = 0.0008x + 0.7248$	0.771
Shannon-Welford	W_e	Linear	$y = 0.0007x + 0.7809$	0.744
Welford	W	Logarithmic	$y = 0.0653ln(x) + 0.4536$	0.943
Shannon-Welford	W	Logarithmic	$y = 0.061ln(x) + 0.5049$	0.973
Welford	W_e	Logarithmic	$y = 0.1741ln(x) - 0.0096$	0.741
Shannon-Welford	W_e	Logarithmic	$y = 0.1376ln(x) + 0.2001$	0.717
Welford	W	Polynomial	$y = -0.0000008x^2 + 0.0007x + 0.694$	0.952
Shannon-Welford	W	Polynomial	$y = -0.0000006x^2 + 0.0006x + 0.7376$	0.977
Welford	W_e	Polynomial	$y = 0.0000008x^2 + 0.0005x + 0.8093$	0.777
Shannon-Welford	W_e	Polynomial	$y = 0.0000006x^2 + 0.0004x + 0.8093$	0.746

Table 5.5: Regression modeling of how k varies as distance from the screen D (cm) changes for each of the four pointing models. The 165 condition has been removed as a potential outlier.

5.3 Discussion

We begin by summarizing the results according to our hypotheses, after which we discuss some of the more nuanced aspects of our results. For each hypothesis we indicate how consistent it was with the data we found.

H1a (weak) The parameter k will increase monotonically with greater target depth **Somewhat consistent with data.**

H1b (strong) The parameter k will increase linearly with greater target depth. **Somewhat consistent with data.**

H2 One-part models of pointing performance will not accurately model all target depths. **Not consistent with data.**

H3 Two-part models of pointing performance will perform better than one-part models in conditions where k diverges from 1. **Consistent with data.**

H4 Angular measures of target difficulty will not improve our models of pointing performance. **Consistent with data.**

All of the pointing models produced fairly good results. Both one- and two-part models successfully modeled the data in all but one condition. One-part models did not pass Mackenzie's threshold in one condition, but were still fairly close even

Figure 5.6: Visual comparison of different models for the relationship between k and D. Data presented for regular width models of pointing performance. The 165 condition has been removed as a potential outlier.

then. Therefore, we argue that hypothesis **H2** (that one-part models will not accurately model all depths) was **not consistent with our data**. Two-part models were statistically more accurate in two conditions using regular width, and had a potential trend of improvement using effective width. These conditions had the smallest k values, which were the k values that diverged the most from unity, therefore hypothesis **H3** (two-part models will outperform one-part models when k diverges from 1.0) was **consistent with our data**.

Angular measures improved some models but degraded others to a similar extent. Absolute differences between models based on angular and classic linear measures of A and W (or W_e) were relatively minor and could be caused by variation of outliers pulling or pushing the model slightly away from the optimal value.

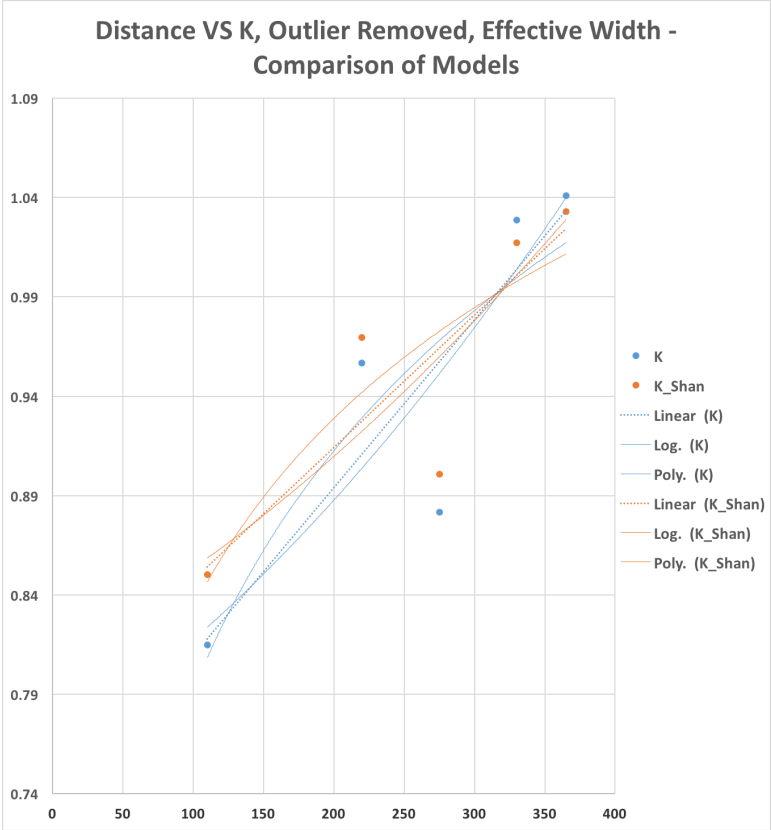

Figure 5.7: Visual comparison of different models for the relationship between k and D. Data presented for effective width models of pointing performance. The 165 condition has been removed as a potential outlier.

Furthermore, two-part models in the global condition got worse when using angular measures of target difficulty. The global condition, which tries to model pointing performance independent of target depth, is precisely the case that advocates of an angular model propose as being its strength. We therefore argue that hypothesis **H4** (that angular measures of difficulty will not improve our models) was **consistent with our data**. We found that subject to outliers, as target depth got larger k got bigger, and at least roughly corrleated along a linear model in all conditions. Logarithmic models were somewhat better at modeling the data when outliers were removed. Due to the problems with outliers we consider hypotheses

H1a and **H1B** to be **somewhat consistent with our data**.

5.3.1 One-Part Versus Two-Part Models of Pointing Performance

We found that one-part models were sufficient to model our data in almost all conditions, but they were statistically worse than two-part models in some conditions close to the screen. These results run somewhat counter to our previous experiment where we noticed that as you got further from the screen two-part models outperform one-part. Instead when you are very close to the screen we see a difference in modeling.

One needs to understand these trends in the presence of the k variable, which captures the relative impact of A and W on pointing performance. If k is exactly equal to one then one- and two-part models are mathematically identical [40]. In our previous experiment, k values started around 1.0 and got progressively larger as target depth increased, up to a maximum of about 1.7. It was therefore unsurprising that the models started out similar but two-part models became necessary as k diverged from one.

In this experiment, k values start around 0.7 and get larger, but rarely surpassed 1.0 by any large margin. Therefore, in this new experiment, where k was much closer to one all the time, one would expect fewer differences between the models and for one-part models to fail less often. Moreover, when two-part were statistically better, k diverges more from one than in other conditions, but this was still less divergence than in the previous experiment, which saw values of k that were 1.7 larger than unity whereas this experiment had a ratio of only 1.4, which was less, as was the absolute difference (0.7 in the first experiment but only 0.3 in the second). This is consistent with our claim that the improvement is only seen when k is not approximately one. We therefore argue that this reinforces and supports our previous conclusions about modeling distal pointing performance.

Two-part models are not inherently necessary to describe all pointing tasks. Many common tasks coincidentally give amplitude and width similar impact. However, specific experiemental conditions and different input techniques may cause this to change somewhat unpredictably and produce scenarios in which two-part models become necessary. Devices that could be useful but that produce non-

standard k values (ones that are not close to unity) quite possibly have been rejected by the research community because they failed to be accurately described by a one-part Fitts model. Two-part models seem more robust to changes in gain or target depth, or any other factor that might make k diverge from unity. We therefore advocate that they be the default models, at least until it has been verified that the range of k values is never far from 1.0 for the task.

5.3.2 Angular Versus Classic Measures of Target Difficulty

Angular measures were intended to help the model be robust to changes in depth, but hadn't been tested in isolation before these two experiments. Studies in literature had compared one-part models using classic measures of difficulty to their new proposed model. While their proposed model used angular measures of difficulty it was also very similar to a two-part Welford's model. Our previous experiment provided data that suggested the improvements of this model may be coming more from the Welford's similarity than swapping to angles.

As we noted in our results section, the differences between angular and classic measures in this experiment were fairly minor and inconsistent. Some models were slightly improved (usually within 5 percent R^2) while others actually got worse when swapping to angular measures. Even the best case scenario of the global multiple depth condition only saw improvements in some conditions, and got worse in others. It seems reasonable to argue that the differences between angular and classic are relatively minor and perhaps a form of random variation after changing units under the effect of perspective projection. While perspective will cause the units to be scaled differently at different depths, we were not able to show in either experiment that this provided any real improvement in modeling.

5.3.3 The Relationship Between k and Target Depth

One thing we were very surprised by in this experiment was the low value of k and its shallow growth with increasing target depth. For example, using regular width the lowest k was 0.67 at 165cm and then raised to 0.832 by 275cm. However, between 275cm and 365cm we see shallow growth in k between 0.832 and 0.835. This corresponds to less than a percent of absolute difference between half of the

target depths tested. Effective width saw k grow more sharply in this range, where it varied from 0.88 to 1.04. However, both variations are still smaller than the variation we saw in the original experiment which went from 1.4 to 1.7 in a similar area. It is plausible that there might be some sort of diminishing returns as you get further and further from the screen. Similar increases in depth may have less and less of an impact an impact on k as target depth becomes more extreme.

We noted in our results that target depth had an interaction with target width in this experiment, rather than an omnibus effect. This is key, as rather than target depth making the task inherently more difficult, it may simply exacerbate the problems with small widths, while not hampering more reasonable targets much. This may be compounded with the changes in absolute target size due to not using stereo projection in this experiment. Perhaps using a slightly different set of absolute widths gave us a different k?

This would make intuitive sense as visual depth discriminations are less precise at farther target depths where binocular disparity plays less of a role [51]. Similar changes in absolute target depth will also have smaller relative changes in total target depth. It could also be caused by hitting the limit of sensorimotor processing. As k gets larger, width has more and more of an impact on accuracy because the task needs to be more and more precise. This will make hand jitter start to become increasingly problematic for selecting smaller targets. It may well be that we are hitting some fundamental sensorimotor limit on fine motor precision and participants can only be so accurate on the targets we are asking them to select.

Given these and the modeling results we presented earlier, we would consider the two most likely models for our k trend as logarithmic and linear. Linear appears to be a somewhat naive but explanatory approximation of the variation in k. By definition it assumes variation is the same at all points. Theoretically a linear trend would imply that the same increase in target depth will always have the same impact on k. Using a linear approximation will throw out the pattern of smaller differences as you get further away, but does capture the positive correlation in our data. Furthermore once outliers were removed the regression software was able to find a reasonable linear compromise between the variation and produced a R^2 higher than 0.9.

We note that second order polynomial models gave us the highest R^2 values in

all conditions, but also introduced additional degrees of freedom that would naturally improve fit. Looking at the absolute values of the parameters, the quadratic term was given much smaller co-efficients than the linear (0.0000006 comapred to 0.0002). This implies that the solved solution was really much more close to linear but slightly improved by a small amount of curve. This is reinforced by a nested model anova that we carried out to compare it to a linear model. No statistically significant differences were found for regular width ($F_{1,6} = 0.355$, $p > 0.05$) or effective width ($F_{1,6} = 0.282$, $p > 0.05$). Furthermore, the actual curve of the regular width polynomial models had increasing slope as target depth increased. This would go completely counter to the visual trend in the data, and may be a product of outliers. For a model to be useful it must consistently describe the trend in the data. While we might be able to see slightly higher R^2 using a polynomial model, it does not actually improve our model visually or statistically and we reject it as an accurate model of k.

A logarithmic model may provide an explanation for the sloping off trend we are seeing in the regular width data. Large increases early on, but smaller as target depth got larger. However, it was statistically the worst model before removing the outlier points (R^2 of 0.44 compared to 0.503 and 0.506). The specific outlier was much lower than the the surrounding points. The logarithmic model has to pull up higher than linear above this point and diverges from it dramatically. We think the logarithmic regression was in a way rejecting this point as an outlier. It maintained the visual trend in the data but jettisoned consideration for the absolute best R^2. Once the outlier was removed, logarithmic models surpassed linear in pure R^2 while more closely reflecting the visual trend in the data than polynomial models. It is important to note however, that this sloping off trend was much less apparent when using effective width, which is a truer representation of performance. Furthermore, even after removing the outlier a logarithmic model of effective width k never surpassed linear in R^2. While it has its appeal, it seems hard to argue conclusively that a logarithmic model provides a consistently better explanation of our data than a simple linear estimation.

It is important to note that while having a high R^2 is important for mathematical consistency, the point of such models is to reflect the relationship of the variables and help us understand their effects. Having a high R^2 is nice, but is no substitute

for a robust explanation of the phenomena involved. Furthermore, there is no universal threshold for what constitues a "good" model of k. While the results were much lower before removing the outlier point, one might reasonably argue that accounting for 45 to 50 percent of all variation in the data is actually reasonably predictive. Especially for a variable like k that might be impacted by dozens of other potential nuisance factors (e.g. gender balance, coffee consumed, participant strategy variation ect.). It is not quite the direct massive correlation of target difficulty with movement time, but that doesn't mean it isn't explanatory or helpful for us understanding the relationship. Further investigation would be useful to nail down what might be causing these outliers, but the rough trend of sloping off increase appears to be there. Given all this, we would argue that k is probably best modeled with a simple linear estimation.

5.3.4 Differences in Task and *k*

We noted in our results that the absolute values of k were much lower in this experiment than the last. In fact, the largest k values in this experiment, were around 1.07, when the smallest in the last experiment were only slightly smaller than 1 (e.g. 0.87 for effective width Welfords).

We would consider the most likely explanation for this as differences in the experimental task and feedback. Targets were not displayed with dynamic stereo projection, so would appear at different real world sizes than experiment one, which could have effected the results. In this task participants also had no feedback of selection accuracy (laser pointer position is a form of feedback) and were instructed to keep the laser smoothly moving between the two targets at all times. For disambiguation we considered all selections to occur at the peak movement acheived before changing directions. Simply put, participants couldn't really do post selection adjustment. If participants overshot their target they would not take the time to carefully line it back up as wherever they stopped originally is where the selection was placed. Thus any precise adjustment due to the impact of width was done in the deceleration motion.

After cutting out the correction phase one would expect participants to spend less overall effort on being precisely accurate. This would cause the width co-

efficient (which captures precision adjusments) to have a smaller impact on movement time and lead to smaller k values. However, the relative impact of amplitude probably wouldn't be very different between the two experiments. The ballistic phase of crossing a large distance should theoretically be pretty similar even when not trying to be precise. These would combine to make k smaller overall in our experiment. We note directions for nailing down these complications in our future work section.

Another possible explanation is the task differences between the two experiments, especially those related to explicit visual feedback about accuracy. Barry Po notes in his PhD dissertation [35] that their are two human visual systems (or *streams*): the *ventral system* that has a more cognitive response when some types of feedback are present, and the *dorsal system* that has more of a pure sensorimotor response when little feedback is present. The two visual systems often have different performance and different error patterns. Participants may be engaging a different visual system in each experiment. It is not clear that the observed trend in k should be similar in both visual systems. More research needs to be done to discern to what extent each experiment engaged each system, perhaps by examining the exact co-efficient values as opposed to just relative k values.

5.4 Future Work

In this section we consider two main avenues for future work. Further investigating and reinforcing our understanding of the k parameter and attempting to utilize this new understanding to implement a VR calibration system.

5.4.1 Isolating Factors That Affect k

We've already noted that the different experimental task may have resulted in a lower k than previous studies. However, we had to remove one outlier point from the data in order to model k as well as in the last experiment. This new experiment was intended to provide a fundamental human baseline for expected pointing performance performance without computer mediation. Given that participant instructions were the same for all conditions and they were partially counterbalanced, what could cause such a dramatic outlier? Resolving this issue will be important

before going forward with any calibration systems relying on an accurate model of k.

One plausible avenue to investigate is our sample being biased. Our average participant age was similar to the previous experiment, but through random sampling we had quite a good deal more women than men (14 compared to 3). Our previous experiment was biased in favor of more men than women (12 compared to 8). Gender variation in motor skills is highly studied and produces significant differences in a variety of tasks. Women tend to perform slightly better at precision motor tasks, while men better at spatial reasoning [14, 52]. Perhaps gender variation might account for some of why k was lower in the second experiment, as women might be able to make precise adjustments more quickly. This could lower the impact of width on pointing performance and lead to a lower k, such as in our outlier point. It therefore seems plausible that perhaps the different biasing of the sample caused k to be lower in our experiment than the last. However, this would not explain why specific depths were impacted so much more than others. Theoretically this should have been accounted for by our counterbalanced within-participants design. We leave post-hoc testing for this to future work, as the sample sizes in this experiment (3 men vs 14 women) are not appropriate for such a between subjects comparison.

Perhaps more likely to cause individual variations in conditions is changes in participants strategy. Of course there is a tradeoff between speed and accuracy in motor tasks like pointing. Depending on the instructions given, participants might emphasize selecting quickly or precisely which creates a tradeoff between movement time and success rate. Thus it is common practice in pointing studies to use effective width to model pointing performance. This adjusts the predicted size of the targets according to what participants actually did in post. Effectively, this makes our analysis consider modeled targets larger if participants sacrificed accuracy. However, an open question remains; Does the speed accuracy tradeoff also effect k. Since k captures the relative impact of amplitude and width on pointing performance, could participants choosing the sacrifice accuracy for a condition cause the impact of width to lessen and k to fall? If so, does modeling k with effective width also account for this tradeoff? Do we need to develop an adjustment similar to effective width for the k parameter itself? Through random chance could

participants have chosen to change their strategy in one condition?

Future studies should look to isolate some of these factors potentially impacting k so we are more sure of what causes the outliers. The same experiment could be run with computer mediation to validate our assertions that task differences caused the lower k values in this experiment. More careful sampling could address gender balance issues, or test whether this effected our results through a between participants design. More careful analysis should be done to tease apart the relationship between strategy and k values. Perhaps a future experiment should try a between participants design where some participants are instructed to be accurate and some are instructed to be fast. We think it is reasonable that at least a few of these studies could be run to hopefully discover how we can make sure such outliers do not happen in the future.

5.4.2 Addressing VR Calibration

Lets assume for now that we've investigated the factors causing k to vary in the different target depths and can come up with a model that is both effective and robust to noise/outliers. Assuming that this model is still linear, what can we practically accomplish? We had suggested in our previous experiment an idea where the virtual target depth we render the targets at could be adjusted to reflect the human baseline k curve determined experimentally. This idea has promise but has a few problems. Does adjusting the actual target depth of the object (and thus its size) change people's perceived target depths and is that a valid way of adjusting performance in a simulation? Given the multitude of potential factors effecting the absolute values of k is it plausible to come up with a single human baseline and say what k "should" be at a given target depth? If individual strategy plays a role, how much can we trust the k values from our participants? Furthermore, our naive idea proposed in our last experiment had some edge cases whereby the corrected target depth wouldn't make sense around the corner cases (e.g. producing a negative target depth extremely close). There seems to be a fair amount of work to be done to integrate this knowledge into a working VR calibration system.

We propose that for any VR calibration system to work it is going to be dependent upon an model from a small set of participants that crossed real and virtual

depth. Furthermore, such a system must ignore the absolute values of k and simply strive to make the VR and real k curves more similar through relative differences. We think its impossible to make the curves exactly match up. However, one could run a quick pointing task crossing physical and virtual depth. If one then observed that k variation is steeper with virtual target depth than real, one could then change stereo parameters to play down the depth effect at far target depths and make the target seem closer while keeping size consistent. Such dynamic stereo parameter adjustment has already been applied to enhance depth estimation in literature before. We argue that it would be valuable to connect this technique to actual task performance and validate the results it gives from a simulation perspective. Perhaps an experiment could be designed to test multiple methods of calibrating a VR system and arrive at a unified system the provides more accurate and comfortable VR experiences.

5.5 Conclusion

This experiment has investigated the variation of Shoemaker's k-factor with depth while removing the confounding factors of virtual reality and computer mediated pointing. We provided further evidence that two-part models outperform one-part models in conditions where k deviates from one. We reinforced our conclusions that angular measures do not really improve our models of pointing performance. We also showed that even after removing computer mediation from the task, k still got bigger as target depth increased. Overall, k was smaller and had slower increases than our previous experiment, which we argue is likely caused by task differences or sampling biases. Our models of target depths impact on k were refined from previous studies to show that a linear approximation still provides the most reliable and effective modeling. Future work should investigate a number of factors, such as the speed-accuracy tradeoff that could explain the noise we noticed in the k-factor. We also argued that dynamic stereo parameter adjustment may be employed as a means of making the virtual and real k curves line up more closely. Future work will investigate these factors, attempt to create a unified model of k and apply it to VR calibration.

Chapter 6

Where Are We? The Path Ahead

This thesis has investigated the research communities methods for evaluating pointing performance in distal pointing tasks. In particular, we tested a variety of common models of pointing and specific improvements and modifications that have been suggested in the literature. These included Fitts's one-part models versus Welford's style formulations, as well as angular versus classic measures of target difficulty. We also investigated the natural human baseline of distal pointing performance without computer mediation.

6.1 One-Part and Two-Part Models of Pointing

Through repeated experiments we have shown that amplitude and width have separable impacts on pointing performance. In many common devices and interactions their impact happens to be similar by happenstance. In such situations standard one-part models of pointing, such as Fitts's Law and the Shannon formulation were shown to be as effective as more complex two-part models. However, we have demonstrated that the relative impact of amplitude and width, as characterized by Shoemaker's k parameter, varies with target depth in distal pointing. When k implies equal impact, Fitts's and Welford's style models are mathematically identical and produce similar results. However, when k implies divergence, Welford's outperforms Fitts's Law in a statistically significant way. This causes one-part models to be less robust to modeling distal pointing interactions, which are particularly

common in virtual reality interactions and a popular research area at the moment. Thus we recommend the two-part Welford's formulation, or its Shannon equivalent as a valid and effective alternative to Fitts's Law.

We note that one-part Fitts's Law allows use of the throughput parameter to compare aggregate pointing performance across studies. Mathematically defined as the average index of difficulty over average movement time, it provides us with a rough metric of how effective the device is. Two-part models of pointing performance separate index of difficulty out into two terms and thus it is non-obvious how they should be incorporated into the throughput metric. Lacking this standard metric of device quality has arguably been one of the main stumbling blocks to using two-part models in practice. Given our data showing that two-part models will become more necessary while modeling the distal interactions that are common in VR, we feel this is a common problem that needs to be addressed. However, the solution is perhaps much simpler than imagined. In fact, why does throughput even need to change at all?

Of course, Fitts's Law is formalized as the logarithm of the ratio of amplitude and width, with throughput dividing the logarithm by average movement time.

$$MT = a + b\log\left(\frac{A_e}{W_e}\right) \tag{2.1}$$

$$TP = (\log\left(\frac{A_e}{W_e}\right)/MT) \tag{2.7}$$

A key observation here is that the standard definition of throughput removes any consideration of the magnitude of the b coefficient. Instead, it just focuses on the magnitude of the ID logarithm (adjusted for consistent accuracy by effective width) compared to ultimate movement time. Welford's formulation splits index of difficulty into amplitude and width terms that are subtracted. This allows a linear regression to apply two separate coefficients to each ID term that reflect their separable impacts.

$$MT = a + b_1\log(A_e) - b_2\log(W_e) \tag{2.3}$$

One could, if they were inclined to do so, follow the same logic as Fitts through-

put and create separate throughput terms for both logarithms in the Welford's model. This would naturally lead to Equation 6.1 and Equation 6.2 which represent the throughput impacts of A and W. These could be added together to form the final throughput of a Welford's model. However, higher widths make the task easier, so a final throughput would arguably subtract the two as in Equation 6.3.

$$TP_a = \frac{log(A_e)}{MT} = \frac{ID_a}{MT} \tag{6.1}$$

$$TP_b = \frac{log(W_e)}{MT} = \frac{ID_b}{MT} \tag{6.2}$$

$$TP_2 = TP_a - TP_b \tag{6.3}$$

Alternatively, one could consider the whole of Welfords as its index of difficulty and formulate the new throughput quite simply as Equation 6.4. Like one-part models we drop consideration of the actual model coefficients and just compare effective target difficulty to movement time.

$$TP_w = \frac{log(A_e) - log(W_e)}{MT} = \frac{ID_w}{MT} \tag{6.4}$$

This would create a metric of throughput for the Welford's formulation that is philosophically analogous to the standard Fitts's Law throughput equation. However, mathematically the results of all of these formulations of throughput are actually identical. By the logarithm quotient rule (see Equation 6.5) we can immediately simplify one-part throughput to TP_w. Simply applying some basic fraction rules ($(A-B)/W = (A/W - B/W)$) farther simplifies TP_w to TP_2. This is similar to our observation that when the coefficents of a Welfords model are the same (or as here, not considered) Fitts's Law and Welfords produce the same model of pointing.

$$log(\frac{x}{y}) = log(x) - log(y) \tag{6.5}$$

Therefore, because our classical definition of throughput does not consider model coefficients, there is no need to change our mathematical definition of through-

put. It is simply a matter of notation as mathematically they produce the same value. However, since we have split throughput into two terms we could assign weights to each term in Welford's throughput (Equation 6.3). This would let us specify whether throughput cares more about accuracy or speed but it is not clear what benefit we gain by considering this. Throughput is intended as a rough metric of overall quality considering both accuracy and speed that can be consistently compared across studies. Deciding whether we care more about ballistic speed or precision accuracy is a more nuanced analysis than throughput was intended for. Rather than getting rid of or radically changing the useful metric of throughput, we argue that it should be kept in its original form or merely changed in notation to reflect the use of two-part models. In addition, future studies should also report on the value of the k parameter their models produced. This would allow us to have the general throughput classifier ("x is ultimately more efficient than y") as well as a more nuanced understanding of its strengths ("because k is positive this device is better at rough ballistic movements"). This may be helpful in tailoring our pointing devices to specific interactions or display configurations.

6.2 Measuring Target Difficulty

There has long been the question in pointing research of how should we measure and quantify how difficult a target is to select. More specifically, what sort of units should we use and where should they be measured? In fact, this was partly the motivation for the one-part formulation of Fitts's Law. By dividing amplitude by width, the units cancel out and we are left with unitless metrics of target difficulty [41]. Most studies simply measure the size in centimeters on the screen, but other options have been suggested. In particular, distal interactions have attempted to measure the difficulty in "hand space" by measuring it in terms of angles the arm has to move. In repeated experiments we have demonstrated that angular measures of target difficulty do not provide any consistent, regular modeling strength improvement over classic measures in centimeters. In some cases, we saw minor improvements but in other cases minor deterioration. We consider this an indication of some natural random variation due to scale change under perspective projection as opposed to any coherent improvement. On a more global level, we have

argued that the units with which one measures pointing performance largely acts as a scaling factor. While using consistent units is important to keep mathematical comparability for throughput, it does not seem to truly change the underlying model.

6.3 Variation of Shoemaker's *k* Parameter with Target Depth

Shoemaker's k factor is defined as the ratio of the amplitude and width coefficients in a Welford's model of pointing. It captures the relative impact of amplitude and width on pointing performance and previous work had already shown that this varies linearly with gain [40]. In this thesis, we have extended this line of enquiry and demonstrated through repeated experiments that target depth has a similar impact on distal pointing. This relationship is at least roughly monotonic. While there is some appeal to modeling it with a logarithmic model, we argue that it is approximated by a linear trend. This is a very simple and explanatory model that produces consistently high R^2 while more complex models do not add any statistical power. Furthermore, we have shown that while latency may exacerbate the magnitude of k variation, this relationship still occurs when latency is removed. Thus we argue that k variation with target depth is reflective of a true sensorimotor trend, and not unique to computer mediated interaction. This has impact on calibration of virtual environments which will be discussed in the next section.

While we have shown evidence to suggest that k varies logarithmically with target depth, there is still much work to be done in making our understanding of the trend more nuanced. There remains an issue with outliers where specific points seem to vary out of step with the rest. We feel tighter experimental controls should be developed to more closely understand and minimize this variation. We therefore suggest three follow-up investigations.

- Post-hoc analysis or between subjects comparison to determine if the skewed gender balance in our second experiment could have affected our trend in k.
- Computer mediated experiment performing the same task as our latency free experiment to confirm how much of the reduction in k is attributed to task

versus latency.

- Post-hoc analysis or between subjects comparison to determine how much of an impact user strategy has upon K.

We feel these are the most likely explanations for why the outliers may have occurred in our second experiment. Hopefully, by investigating these impacts we could devise some metric for the deviation of k that either accounts for these nuisance factors or corrects them in post. Creating a robust human baseline for how k varies with target depth would be important and interesting for calibrating VR pointing tasks for lifelike performance.

6.4 Impact on Virtual Reality Calibration

Given the recent spate of new ideas and work being done in virtual reality, an important and pertinent research topic is ensuring these devices are simulating reality effectively. In this work we have provided some preliminary ideation towards implementing a performance based method for VR calibration. This would insure that participants are not just self-reporting that the objects look correct, but that they are actually interacting with them in a similar manner to real life. In our initial experiment, we suggested simply adjusting the target depth at which virtual objects are rendered to match the k curve of the ground truth condition. Theoretically this would line up each VR condition with the ground truth condition of equivalent performance. A diagram for this is presented in Figure 3.9

While the simplicity of this idea is useful and intuitive, after seeing the potential for outliers in our next experiment we have decided to move away from it. There is simply too great a chance for an outlier to cause the curves to diverge from truth and give us nonsense depths. This is particularly problematic towards the ends of the interaction zone where the corrected depth may not even make sense. Instead, we now believe that dynamic stereo parameter adjustment may provide a more promising avenue. It has already been shown in literature that changing the stereo parameters can be used to make depth estimation more accurate [24]. We hypothesize that this method can be augmented with an understanding of the real and virtual curves of k. One could note where k in virtual space exceeds the

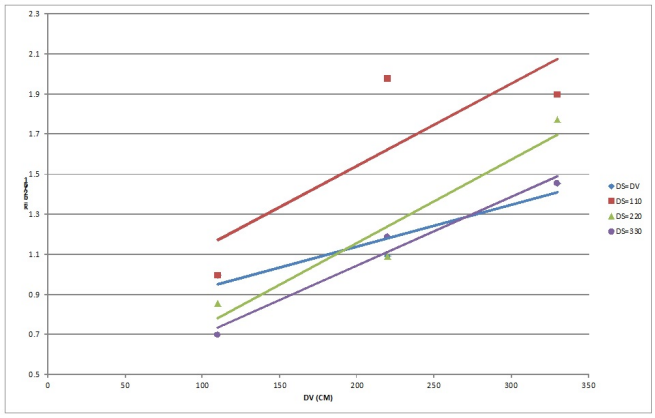

Figure 6.1: The three k-lines for $D_S = 110$ (red / squares), 220 (green / triangles) and 330 (purple / circles), along with the k-line for the non-VR $D_V = D_S$ conditions (blue / diamonds). To calibrate binocular depth using k values, the desired binocular depth (A) determines a k-value (B) on the blue $D_V = D_S$ line. That same k-value on the red $D_S = 110$ line (C) determines D'_V to be the corresponding binocular depth (D) that the software should use to ensure the desired pointing performance if the screen is 110cm from the viewer.

intended physical result and subtly adjust the stereo parameters to play down the depth effect. The exact amount of shift would need to be decided and the results of this calibration would need to be validated for simulation effectiveness. Does it appear reasonable, and how does it effect simulation sickness? While still very much preliminary, it seems a promising extension to accepted and relevant work in the field.

6.5 Final Words

Distal interaction is a fundamental sub-component to exploring and working with many virtual environments. This work has gone back to the fundamentals of the human sensorimotor systems and re-evaluated how we choose our models of pointing performance. We have demonstrated situations where current experimental practice fails unpredictably. This may cause devices and tasks with non-standard gain and or depth parameters to not be accepted by the pointing community. Welford's

two-part model of pointing performance has been shown to be a more robust and effective model of pointing. We have futher explained how we can extend two-part models to include an equivalent metric of throughput to one-part models. In our opinion, this removes the last major objection to using two-part models over one-part. Enhancements such as angular measures of target difficulty did not consistently improve our models. We've also argued there is a logarithmic relationship between k (the relative impact of amplitude and width on pointing performance) and target depth. However, this relationship can be reasonably approximated by a more rough linear trend. This work has impact upon and provides new avenues for work in VR calibration while pushing forward our methods for pointing evaluation. Hopefully, it provides a strong theoretical basis for how we evaluate new pointing techniques in VR interaction.

Bibliography

[1] M. Azmandian, M. Hancock, H. Benko, E. Ofek, and A. D. Wilson. Haptic retargeting: Dynamic repurposing of passive haptics for enhanced virtual reality experiences. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 1968–1979. ACM, 2016. → pages 14

[2] G. Casiez, D. Vogel, R. Balakrishnan, and A. Cockburn. The impact of control-display gain on user performance in pointing tasks. *HumanComputer Interaction*, 23(3):215–250, 2008. doi:10.1080/07370020802278163. URL http://www.tandfonline.com/doi/abs/10.1080/07370020802278163. → pages 25, 30

[3] S. A. Douglas, A. E. Kirkpatrick, and I. S. MacKenzie. Testing pointing device performance and user assessment with the iso 9241, part 9 standard. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '99, pages 215–222, New York, NY, USA, 1999. ACM. ISBN 0-201-48559-1. doi:10.1145/302979.303042. URL http://doi.acm.org/10.1145/302979.303042. → pages 5, 20, 58

[4] S. R. Ellis. Nature and origins of virtual environments: A bibliographical essay. *Computing Systems in Engineering*, 2(4):321–347, May 1991. → pages 13, 35

[5] D. C. Engelbart and W. K. English. A research center for augmenting human intellect. In *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*, pages 395–410. ACM, 1968. → pages 5

[6] W. K. English, D. C. Engelbart, and M. L. Berman. Display-selection techniques for text manipulation. *IEEE Transactions on Human Factors in Electronics*, HFE-8(1):5–15, March 1967. ISSN 0096-249X. doi:10.1109/THFE.1967.232994. → pages 5

[7] A. S. Fernandes and S. K. Feiner. Combating vr sickness through subtle dynamic field-of-view modification. In *2016 IEEE Symposium on 3D User Interfaces (3DUI)*, pages 201–210. IEEE, 2016. → pages 15

[8] D. Finnegan, E. O'Neill, and M. Proulx. Compensating for distance compression in audiovisual virtual environments using incongruence. In *SIGCHI Conference in Human Factors in Computing Systems 2016*. University of Bath, 2016. → pages 15

[9] P. M. Fitts. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of experimental psychology*, 47(6):381, 1954. → pages 8, 41

[10] P. M. Fitts. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology*, 47(6):381–391, 1954. → pages 2, 5, 20, 24, 58

[11] F. E. Gonzalez. *A camera-based approach to remote pointing interactions in the classroom*. PhD thesis, University of British Columbia, 2015. → pages 11

[12] E. D. Graham. *Pointing on a computer display*. PhD thesis, Simon Fraser University, 1996. → pages 30

[13] E. D. Graham and C. L. MacKenzie. Physical versus virtual pointing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '96, pages 292–299, New York, NY, USA, 1996. ACM. ISBN 0-89791-777-4. doi:10.1145/238386.238532. URL http://doi.acm.org/10.1145/238386.238532. → pages 34, 37

[14] E. Hampson. Estrogen-related variations in human spatial and articulatory-motor skills. *Psychoneuroendocrinology*, 15(2):97–111, 1990. → pages 83

[15] K. Iyer, M. Chari, and H. Kannan. A novel approach to depth image based rendering based on non-uniform scaling of depth values. In *Future Generation Communication and Networking Symposia, 2008. FGCNS '08. Second International Conference on*, volume 3, pages 31–34, Dec 2008. doi:10.1109/FGCNS.2008.46. → pages 14, 35

[16] B. F. Janzen and R. J. Teather. Is 60 fps better than 30?: the impact of frame rate and latency on moving target selection. In *CHI'14 Extended Abstracts on Human Factors in Computing Systems*, pages 1477–1482. ACM, 2014. → pages iv, 12

[17] G. R. Jones, D. Lee, N. S. Holliman, and D. Ezra. Controlling perceived depth in stereoscopic images. *Proc. SPIE Stereoscopic Displays and Virtual Reality Systems VIII*, 4297:42–53, June 2001. → pages 14, 35

[18] R. Jota, A. Ng, P. Dietz, and D. Wigdor. How fast is fast enough?: a study of the effects of latency in direct-touch pointing tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2291–2300. ACM, 2013. → pages 12

[19] N. C. Kerphart. *The slow learner in the classroom*. Columbus, OH: Merrill, 2 edition, 1971. → pages 21, 62

[20] H. H. Koester, E. LoPresti, and R. C. Simpson. Toward Goldilocks' pointing device: Determining a "just right" gain setting for users with physical impairments. In *Proceedings of the 7th international ACM SIGACCESS conference on Computers and accessibility*, Assets '05, pages 84–89, New York, NY, USA, 2005. ACM. ISBN 1-59593-159-7. doi:10.1145/1090785.1090802. → pages 25

[21] J. Kohn and S. Rank. Evaluating physical movement as trigger for transitioning between environments in virtual reality. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pages 1973–1979. ACM, 2016. → pages 14

[22] R. Konrad, E. A. Cooper, and G. Wetzstein. Novel optical configurations for virtual reality: evaluating user preference and performance with focus-tunable and monovision near-eye displays. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 1211–1220. ACM, 2016. → pages 14

[23] R. Kopper, D. A. Bowman, M. G. Silva, and R. P. McMahan. A human motor behavior model for distal pointing tasks. *Int. J. Hum.-Comput. Stud.*, 68(10):603–615, Oct. 2010. ISSN 1071-5819. doi:10.1016/j.ijhcs.2010.05.001. → pages 1, 2, 11, 28, 65, 69

[24] A. Kulshreshth and J. J. LaViola Jr. Dynamic stereoscopic 3D parameter adjustments for enhanced depth discrimination. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 177–187. ACM, 2016. → pages 14, 91

[25] M. A. Lawrence. ez: Easy analysis and visualization of factorial experiments (version 4.3). https://cran.r-project.org/web/packages/ez/, November 2015. Accessed: 2016-08-26. → pages 66

[26] I. S. MacKenzie. *Fitts' law as a performance model in human-computer interaction.* PhD thesis, University of Toronto, Toronto, ON, Canada, 1991. → pages x, 5, 7

[27] I. S. MacKenzie. Fitts' law as a research and design tool in human-computer interaction. *Hum.-Comput. Interact.*, 7(1):91–139, Mar. 1992. ISSN 0737-0024. doi:10.1207/s15327051hci0701_3. URL http://dx.doi.org/10.1207/s15327051hci0701_3. → pages 31

[28] I. S. MacKenzie and S. Riddersma. Effects of output display and control-display gain on human performance in interactive systems. *Behaviour and Information Technology*, 13:328–337, 1994. → pages 5, 25

[29] B. A. Myers. A brief history of human-computer interaction technology. *interactions*, 5(2):44–54, Mar. 1998. ISSN 1072-5520. doi:10.1145/274430.274436. URL http://doi.acm.org/10.1145/274430.274436. → pages 5

[30] V. Oculus. Oculus rift-virtual reality headset for 3d gaming. *URL: http://www. oculusvr. com*, 2012. → pages 14

[31] A. Pavlovych and W. Stuerzlinger. The tradeoff between spatial jitter and latency in pointing tasks. In *Proceedings of the 1st ACM SIGCHI symposium on Engineering interactive computing systems*, pages 187–196. ACM, 2009. → pages 3, 12, 37, 40, 57

[32] A. Pavlovych and W. Stuerzlinger. Target following performance in the presence of latency, jitter, and signal dropouts. In *Proceedings of Graphics Interface 2011*, pages 33–40. Canadian Human-Computer Communications Society, 2011. → pages 12

[33] B. Peek. Wiimotelib - .NET managed library for the Nintendo Wii remote, 2012. URL brianpeek.com/page/wiimotelib. Accessed: 2016-08-26. → pages 19

[34] A. Pino, E. Tzemis, N. Ioannou, and G. Kouroupetroglou. Using kinect for 2d and 3d pointing tasks: performance evaluation. In *Human-Computer Interaction. Interaction Modalities and Techniques*, pages 358–367. Springer, 2013. → pages 10

[35] B. A. Po. *Open Loop Pointing in Virtual Environments*. PhD thesis, University of British Columbia, 2002. → pages 82

[36] P. Rubin. The inside story of oculus rift and how virtual reality became reality. *Wired (May 20).(accessedJanuary 24-2015)*, 2014. → pages 14

[37] L. Sambrooks and B. Wilkinson. Comparison of gestural, touch, and mouse interaction with fitts' law. In *Proceedings of the 25th Australian Computer-Human Interaction Conference: Augmentation, Application, Innovation, Collaboration*, pages 119–122. ACM, 2013. → pages 10

[38] W. N. Schofield. Do children find movements which cross the body midline difficult? *Quarterly Journal of Experimental Psychology*, 28(4):571–582, 1976. doi:10.1080/14640747608400584. → pages 21, 62

[39] G. Shoemaker. *Body-centric and shadow-based interaction for large wall displays*. PhD thesis, University of British Columbia, 2010. → pages 11

[40] G. Shoemaker, T. Tsukitani, Y. Kitamura, and K. S. Booth. Two-part models capture the impact of gain on pointing performance. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 19(4):28, 2012. → pages 2, 6, 7, 11, 12, 16, 17, 20, 23, 25, 30, 33, 34, 36, 37, 56, 58, 77, 90

[41] R. W. Soukoreff and I. S. MacKenzie. Towards a standard for pointing device evaluation, perspectives on 27 years of Fitts' law research in HCI. *Int. J. Hum.-Comput. Stud.*, 61(6):751–789, Dec. 2004. ISSN 1071-5819. doi:10.1016/j.ijhcs.2004.09.001. → pages 5, 6, 8, 11, 24, 31, 89

[42] W. Strunk, Jr. and E. B. White. *Elements of style*. New York, NY: Macmillan, 1959. → pages 6

[43] R. J. Teather and W. Stuerzlinger. Pointing at 3D targets in a stereo head-tracked virtual environment. In *3D User Interfaces (3DUI), 2011 IEEE Symposium on*, pages 87–94. IEEE, 2011. → pages 1, 35

[44] R. J. Teather and W. Stuerzlinger. A system for evaluating 3D pointing techniques. In *Proceedings of the 18th ACM Symposium on Virtual Reality Software and Technology*, VRST '12, pages 209–210, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1469-5. doi:10.1145/2407336.2407383. URL http://doi.acm.org/10.1145/2407336.2407383. → pages 35

[45] R. J. Teather and W. Stuerzlinger. Pointing at 3D target projections with one-eyed and stereo cursors. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 159–168, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-1899-0. doi:10.1145/2470654.2470677. URL http://doi.acm.org/10.1145/2470654.2470677. → pages

[46] R. J. Teather and W. Stuerzlinger. Visual aids in 3D point selection experiments. In *Proceedings of the 2Nd ACM Symposium on Spatial User Interaction*, SUI '14, pages 127–136, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2820-3. doi:10.1145/2659766.2659770. URL http://doi.acm.org/10.1145/2659766.2659770. → pages

[47] R. J. Teather and W. Stuerzlinger. Depth cues and mouse-based 3D target selection. In *Proceedings of the 2Nd ACM Symposium on Spatial User Interaction*, SUI '14, pages 156–156, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2820-3. doi:10.1145/2659766.2661221. URL http://doi.acm.org/10.1145/2659766.2661221. → pages

[48] R. J. Teather and W. Stuerzlinger. Factors affecting mouse-based 3D selection in desktop vr systems. In *Proceedings of the 3rd ACM Symposium on Spatial User Interaction*, SUI '15, pages 10–19, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3703-8. doi:10.1145/2788940.2788946. URL http://doi.acm.org/10.1145/2788940.2788946. → pages 35

[49] R. J. Teather, A. Pavlovych, W. Stuerzlinger, and S. I. MacKenzie. Effects of tracking technology, latency, and spatial jitter on object movement. In *3D User Interfaces, 2009. 3DUI 2009. IEEE Symposium on*, pages 43–50. IEEE, 2009. → pages 12, 40, 57

[50] R. J. Teather, W. Stuerzlinger, and A. Pavlovych. Fishtank fitts: A desktop vr testbed for evaluating 3D pointing techniques. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '14, pages 519–522, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2474-8. doi:10.1145/2559206.2574810. URL http://doi.acm.org/10.1145/2559206.2574810. → pages 35

[51] W. H. Teichner, J. L. Kobrick, and R. F. Wehrkamp. The effects of terrain and observation distance on relative depth discrimination. *The American journal of psychology*, 68(2):193–208, 1955. → pages 79

[52] D. Voyer, S. Voyer, and M. P. Bryden. Magnitude of sex differences in spatial abilities: a meta-analysis and consideration of critical variables. *Psychological bulletin*, 117(2):250, 1995. → pages 83

[53] A. T. Welford. *Fundamentals of skill*. Methuen's Manuals of Modern Psychology. London: Methuen, 1968. → pages 2, 6, 30, 34, 37

[54] R. Xiao and H. Benko. Augmenting the field-of-view of head-mounted displays with sparse peripheral displays. In *Proceedings of the 2016 CHI*

Conference on Human Factors in Computing Systems, pages 1221–1232. ACM, 2016. → pages 14

[55] C. Yuan, H. Pan, and S. Daly. Stereoscopic 3D content depth tuning guided by human visual models. *SID Symposium Digest of Technical Papers*, 42(1): 916–919, 2011. ISSN 2168-0159. → pages 14, 35

[56] A. Zaranek, B. Ramoul, H. F. Yu, Y. Yao, and R. J. Teather. Performance of modern gaming input devices in first-person shooter target acquisition. In *CHI'14 Extended Abstracts on Human Factors in Computing Systems*, pages 1495–1500. ACM, 2014. → pages 10

[57] S. Zhai, J. Kong, and X. Ren. Speed-accuracy tradeoff in Fitts' law tasks: On the equivalency of actual and nominal pointing precision. *Int. J. Hum.-Comput. Stud.*, 61(6):823–856, Dec. 2004. ISSN 1071-5819. doi:10.1016/j.ijhcs.2004.09.007. → pages 27

[58] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence*, 22(11): 1330–1334, 2000. → pages 51

Appendix A

Supporting Materials

This appendix primarily contains the annonymized raw data from our different experiments. As the total amount of data would be unmanageable to report on, we provide aggreate statistics including average movement time and effective width. Study instruments including our demographics questionaire are also reported on.

A.1 Study Materials

A.2 Raw Data

D_S	D_V	A	W	W_E	A_E	MT	LowCI	HighCI
110	110	25	5	6.70	25.45	1229.85	1138.89	1320.82
110	110	25	10	8.65	25.46	911.54	820.57	1002.50
110	110	25	20	16.43	25.92	688.41	597.44	779.37
110	110	50	5	7.67	50.14	1551.79	1460.82	1642.75
110	110	50	10	12.10	50.35	1205.60	1114.64	1296.57
110	110	50	20	17.31	51.00	965.46	874.49	1056.42
110	110	75	5	11.31	75.62	1798.58	1707.62	1889.55
110	110	75	10	12.86	75.42	1385.95	1294.99	1476.92
110	110	75	20	29.25	75.07	1099.31	1008.35	1190.28
110	220	25	5	6.63	25.29	1887.37	1796.41	1978.34
110	220	25	10	6.67	25.40	1192.47	1101.50	1283.43

D_S	D_V	A	W	W_E	A_E	MT	LowCI	HighCI
110	220	25	20	10.46	25.35	852.15	761.19	943.12
110	220	50	5	7.43	50.28	1972.32	1881.36	2063.29
110	220	50	10	9.06	50.24	1385.73	1294.77	1476.69
110	220	50	20	11.47	49.73	1109.63	1018.66	1200.59
110	220	75	5	6.33	75.44	2317.52	2226.56	2408.49
110	220	75	10	14.78	75.33	1656.73	1565.77	1747.69
110	220	75	20	17.61	74.55	1221.71	1130.75	1312.68
110	330	25	5	9.44	25.02	2174.89	2083.93	2265.86
110	330	25	10	8.70	25.39	1713.43	1622.46	1804.39
110	330	25	20	9.95	25.18	963.01	872.04	1053.97
110	330	50	5	10.71	50.37	2492.41	2401.44	2583.37
110	330	50	10	9.48	49.78	1948.03	1857.07	2039.00
110	330	50	20	12.06	49.84	1268.31	1177.35	1359.27
110	330	75	5	9.22	75.51	2949.67	2858.70	3040.63
110	330	75	10	8.31	75.11	2069.95	1978.99	2160.92
110	330	75	20	13.79	75.11	1495.17	1404.21	1586.13
220	110	25	5	11.18	25.54	1351.85	1260.89	1442.82
220	110	25	10	23.86	25.93	1003.88	912.92	1094.85
220	110	25	20	25.19	26.92	828.98	738.01	919.94
220	110	50	5	11.90	50.10	1623.27	1532.31	1714.23
220	110	50	10	16.81	50.49	1321.52	1230.56	1412.49
220	110	50	20	30.67	50.15	1038.77	947.80	1129.73
220	110	75	5	15.85	76.29	1991.38	1900.41	2082.34
220	110	75	10	26.39	76.53	1569.39	1478.43	1660.36
220	110	75	20	31.66	77.61	1320.33	1229.36	1411.29
220	220	25	5	8.08	25.79	1332.75	1241.79	1423.72
220	220	25	10	12.61	25.61	973.90	882.94	1064.87
220	220	25	20	16.22	26.20	776.36	685.40	867.33
220	220	50	5	7.46	50.57	1698.42	1607.46	1789.39
220	220	50	10	12.36	50.91	1228.03	1137.07	1319.00
220	220	50	20	19.22	51.29	949.56	858.60	1040.53
220	220	75	5	9.36	75.51	1906.07	1815.10	1997.03

D_S	D_V	A	W	W_E	A_E	MT	LowCI	HighCI
220	220	75	10	13.72	75.62	1515.02	1424.06	1605.99
220	220	75	20	19.89	76.24	1159.64	1068.67	1250.60
220	330	25	5	6.79	25.38	1663.40	1572.44	1754.37
220	330	25	10	10.16	25.26	1234.16	1143.19	1325.12
220	330	25	20	12.37	25.88	817.61	726.65	908.58
220	330	50	5	9.39	50.51	1998.51	1907.55	2089.48
220	330	50	10	10.57	50.65	1447.49	1356.53	1538.45
220	330	50	20	14.70	50.57	1087.13	996.17	1178.10
220	330	75	5	19.13	74.84	2142.74	2051.77	2233.70
220	330	75	10	20.99	74.92	1608.96	1518.00	1699.93
220	330	75	20	23.59	74.81	1192.52	1101.56	1283.49
330	110	25	5	29.92	24.68	1392.45	1301.49	1483.42
330	110	25	10	38.96	22.56	1075.40	984.44	1166.37
330	110	25	20	43.36	22.71	819.65	728.69	910.62
330	110	50	5	16.94	50.58	1971.15	1880.19	2062.12
330	110	50	10	36.52	48.10	1537.05	1446.09	1628.02
330	110	50	20	56.41	45.86	1137.21	1046.25	1228.18
330	110	75	5	78.31	74.66	2392.42	2301.45	2483.38
330	110	75	10	29.19	76.91	1936.15	1845.18	2027.11
330	110	75	20	85.21	74.40	1546.01	1455.04	1636.97
330	220	25	5	10.93	25.88	1348.55	1257.58	1439.51
330	220	25	10	19.95	25.80	1011.31	920.35	1102.28
330	220	25	20	21.92	26.42	767.65	676.69	858.62
330	220	50	5	11.80	50.46	1586.63	1495.67	1677.60
330	220	50	10	15.39	50.51	1296.33	1205.36	1387.29
330	220	50	20	35.53	50.91	947.00	856.03	1037.96
330	220	75	5	11.06	75.46	1908.90	1817.94	1999.87
330	220	75	10	37.28	75.07	1421.08	1330.11	1512.04
330	220	75	20	26.73	75.22	1139.59	1048.63	1230.56
330	330	25	5	8.81	25.73	1585.54	1494.57	1676.50
330	330	25	10	11.80	25.35	1053.63	962.66	1144.59
330	330	25	20	16.84	25.12	772.66	681.70	863.63

D_S	D_V	A	W	W_E	A_E	MT	LowCI	HighCI
330	330	50	5	11.71	50.52	1851.61	1760.65	1942.58
330	330	50	10	15.25	50.85	1306.28	1215.32	1397.25
330	330	50	20	21.30	50.27	1012.67	921.71	1103.64
330	330	75	5	9.73	75.71	2161.65	2070.68	2252.61
330	330	75	10	12.93	75.48	1629.48	1538.52	1720.45
330	330	75	20	23.46	75.05	1118.22	1027.26	1209.19

Table A.1: Processed raw data from experiment one. Outliers have already been filtered out and we present effective widths, amplitudes, average movement times and 95 percent confidence intervals for all conditions.

D_S	A	W	W_E	A_E	MT	LowCI	HighCI
110	25	5	6.12	25.13	386.57	345.63	427.51
110	25	10	8.66	25.18	319.44	278.50	360.38
110	25	20	18.98	25.27	264.70	223.76	305.64
110	50	5	6.09	50.15	580.21	539.27	621.15
110	50	10	9.60	50.19	427.55	386.61	468.49
110	50	20	17.89	50.40	353.82	312.88	394.76
110	75	5	8.46	75.35	685.88	644.94	726.82
110	75	10	11.71	74.08	595.95	555.01	636.89
110	75	20	17.96	73.28	394.91	353.97	435.85
165	25	5	6.36	25.10	419.56	378.62	460.50
165	25	10	8.20	25.07	337.50	296.56	378.44
165	25	20	18.33	25.19	256.94	216.00	297.88
165	50	5	7.15	50.15	579.63	538.69	620.57
165	50	10	9.38	50.18	445.14	404.20	486.08
165	50	20	17.38	50.28	362.62	321.68	403.56
165	75	5	7.49	75.26	695.37	654.43	736.31
165	75	10	12.62	74.23	610.42	569.48	651.36
165	75	20	19.83	72.85	409.95	369.01	450.89
220	25	5	6.16	25.21	451.97	411.03	492.91
220	25	10	8.54	25.10	332.64	291.70	373.58

D_S	A	W	W_E	A_E	MT	LowCI	HighCI
220	25	20	18.02	25.16	265.62	224.69	306.56
220	50	5	7.86	50.23	606.37	565.43	647.31
220	50	10	9.40	50.17	454.98	414.04	495.92
220	50	20	16.92	50.40	378.36	337.42	419.30
220	75	5	8.31	75.18	735.65	694.71	776.59
220	75	10	9.46	74.61	647.69	606.75	688.62
220	75	20	19.00	73.95	423.50	382.56	464.44
275	25	5	6.47	25.27	463.19	422.25	504.13
275	25	10	8.87	25.12	325.69	284.75	366.63
275	25	20	17.59	25.04	241.32	200.38	282.26
275	50	5	7.22	50.30	569.21	528.27	610.15
275	50	10	9.51	50.15	471.30	430.36	512.24
275	50	20	16.35	50.18	351.50	310.57	392.44
275	75	5	7.16	75.26	707.29	666.35	748.23
275	75	10	12.88	74.57	548.03	507.09	588.97
275	75	20	22.79	74.38	393.52	352.58	434.46
330	25	5	7.11	25.14	468.17	427.23	509.11
330	25	10	9.37	25.17	328.94	288.00	369.87
330	25	20	17.66	25.08	239.47	198.53	280.41
330	50	5	7.15	50.19	650.81	609.87	691.75
330	50	10	9.84	50.02	472.34	431.40	513.28
330	50	20	17.85	50.21	346.88	305.94	387.81
330	75	5	8.55	75.18	746.30	705.36	787.24
330	75	10	11.82	75.27	579.51	538.57	620.45
330	75	20	20.45	74.98	422.92	381.98	463.86
365	25	5	7.57	25.21	515.28	474.34	556.22
365	25	10	9.01	25.20	340.05	299.11	380.99
365	25	20	17.31	25.11	267.13	226.19	308.07
365	50	5	7.32	50.25	656.25	615.31	697.19
365	50	10	9.51	50.16	524.42	483.48	565.36
365	50	20	17.34	50.22	361.11	320.17	402.05
365	75	5	7.09	75.15	736.34	695.40	777.28

D_S	A	W	W_E	A_E	MT	LowCI	HighCI
365	75	10	9.76	75.38	616.20	575.26	657.14
365	75	20	22.36	75.47	432.06	391.12	473.00

Table A.2: Processed movement time data from experiment two. Outliers have already been filtered out and we present effective widths, amplitudes, average movement times with low and high 95 percent confidence intervals for all conditions.

Interacting With Large Displays II

UBC Department of Computer Science
ICICS/CS Building
201-2366 Main Mall
Vancouver, B.C., V6T 1Z4

Consent Form

Principal Investigator
Kellogg S. Booth, Professor, Department of Computer Science, (604) 822-8193

Co-Investigators
Vasanth Rajendran, M.Sc. Student, Department of Computer Science, (778) 991-2616
Garth Shoemaker, Postdoctoral Fellow, Department of Computer Science, (604) 827-3993
Tao Su, M.Sc. Student, Department of Computer Science, (778) 318-8499
Julia Rose Freedman, B.Sc. Student, Department of Computer Science, (604) 603-9200

Project Purpose and Procedures
The purpose of this study is to evaluate different methods for interacting with different sized electronic displays. Because of the properties of some displays, and how they are used, standard devices such as mice and keyboards are ill suited. You will be asked to point at targets on the display, and press a button to select the targets.

Confidentiality
Your identity will remain anonymous and will be kept confidential. A computer will record performance and motion data as you perform the tasks, but no identifying information (such as your name) will be stored with this data, nor will it be associated with the data after it has been analyzed.

The results will be made public through publications; however, no identifying information will be included in any published disclosure of the research.

No audio recordings or photographs will be made of your participation.

Risks/Remuneration/Compensation
There are no anticipated risks to you participating in this research. Use of stereoscopic glasses may cause slight discomfort or fatigue in some subjects. You are free to take a break or withdraw from the study.

You will receive an honorarium of $10 for your participation. You will be eligible for the honorarium even if you withdraw from the study.

Version 1.2 2011-August-28 page 1/2

Figure A.1: Consent forms for experiment one which investigated the impact of depth on pointing performance. Data gathered and performed by Vasanth Rajendran.

Interacting With Large Displays V

UBC Department of Computer Science
ICICS/CS Building
201-2366 Main Mall
Vancouver, B.C., V6T 1Z4

Consent Form

Principal Investigator
Kellogg S. Booth, Professor, Department of Computer Science, 604-822-8193

Co-Investigator
Izabelle F. Janzen, M.Sc. Student, Department of Computer Science, 604-345-4263

Project Purpose and Procedures
The purpose of this study is to evaluate different methods for interacting with different sized electronic displays. Because of the properties of some displays, and how they are used, standard devices such as mice and keyboards are ill suited. You will be asked to point at targets on the display, and select the targets.

Confidentiality
Your identity will remain anonymous and will be kept confidential. A computer will record performance and motion data as you perform the tasks, but no identifying information (such as your name) will be stored with this data, nor will it be associated with the data after it has been analyzed.

The results will be made public through publications; however, no identifying information will be included in any published disclosure of the research.

No audio recordings or photographs will be made of your participation.

Risks/Remuneration/Compensation
There are no anticipated risks to you participating in this research. You are free to take a break or withdraw from the study.

You will receive an honorarium of $10 for your participation. You will be eligible for the honorarium even if you withdraw from the study.

Consent Form – Version 1.8 – 2015-Dec-1 page 1/2

Figure A.2: Consent forms for experiment two which investigated the impact whether the impact of depth on pointing performance is an artifact of system latency.

Interaction With Large Displays III Phase 1 Participant # ☐☐☐

Pre-Experiment Questionnaire

1. How old are you?
   ☐☐ years

2. What is your gender? *(tick one)*

   ○ Male
   ○ Female
   ○ Other

3. How much time do you spend **per week** using a computer? *(tick one)*

   ○ Less than 1 hour
   ○ 1 to 3 hours
   ○ 4 to 8 hours
   ○ More than 8 hours

4. Do you normally wear glasses or contact lenses? *(tick one)*

   ○ Yes

   If yes, what is your prescription?
   ○ _____

   ○ I don't know

   ○ No

[Some questions not relevant to the current experiment have been deleted from earlier versions of the questionnaire]

Version 1.3 2012-December 27 Pre-Questionnaire page 1/1

Figure A.3: Demographics questionaire used to gather qualitative data in both studies presented in this thesis.