

Virtual Tomographic Reflection for Psychophysical Analysis of the Sonic Flashlight

Damion Shelton

CMU-RI-TR-??-??

*Submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Robotics*

Robotics Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

March 2007

© 2007 Damion Shelton. All rights reserved.

For Bob Parker

Abstract

Augmented reality systems show great promise for improving the ability of medical personnel to incorporate image guidance into a wider variety of procedures than they are currently able. In its present form, medical image guidance often requires that patients be moved to a specialized interventional radiology suite and that specially trained personnel conduct the image guided therapy. Much of the difficulty in medical image guidance centers on mentally fusing the medical image with the patient's body. Augmented reality systems can alleviate this difficulty by visually fusing the medical image with the patient's body, allowing much more natural guidance. Among the questions that can be asked about augmented reality systems in medicine are A) "Do these systems actually improve the ability of medical personnel to treat patient?" and B) "How do these systems affect perception of the medical image information?"

This dissertation focuses on the Sonic Flashlight, an augmented reality device that provides real-time *in situ* visualization of ultrasound images by reflecting calibrated images displayed on a flat-panel monitor from a partially transparent half silvered mirror. Quantitative evaluation of the efficacy of the Sonic Flashlight versus conventional ultrasound guidance is difficult because of the lack of a real-world "gold-standard" reference frame in which to report results. In addition, psychophysics experimentation with the Sonic Flashlight is difficult due to the complexity of manipulating the image characteristics of conventional ultrasound phantoms.

In this dissertation I explore the use of a virtual environment for conducting accuracy and psychophysical experiments on both the Sonic Flashlight and conventional ultrasound. In contrast to conventional phantom-based analysis of augmented reality systems, the virtual environment provides quantitative performance measures within a high-precision global frame of reference. In addition, the ease with which the virtual environment may be manipulated allows for a much more rapid exploration of psychophysical phenomena than would be possible with physical phantoms. The long term goals for this project are to support an ongoing effort to examine the impact of augmented reality technologies on medical procedures, and to provide a test environment for future augmented reality technology.

Acknowledgements

Thanks to my advisor George Stetten for his many years of friendship, guidance, and support; I look forward to many additional years of collaboration. Thanks also to my committee members Roberta Klatzky, Yoky Matsuoka, and Mel Siegel for their invaluable guidance in designing and conducting my thesis research. Collaboration with my labmates over the years has been extremely helpful - in particular, thanks to Wilson Chang for the design of the clinical Sonic Flashlight version on which my research is based, and Bing Wu for his many hours of help with statistics and experiment design.

I am grateful for the generosity of the many philanthropic organizations that have funded my education and research over the past ten years: The Arnold and Mabel Beckman Foundation, Barry M. Goldwater Scholarship Fund, and the Whitaker Foundation.

A special thanks to Ken Fischer, Jamie Pfaeffle, and Ted Manson, who got me started on this whole research thing in the first place and took a chance on an unproven undergrad who liked to play with computers.

Grad school is a serious enough time commitment that you're well advised to adopt it as a lifestyle. Thanks to all of my friends who have helped to make my time here enjoyable, and proved that grad school really *can* be more fun than undergrad.

Most importantly, thanks to my parents for their support and encouragement to follow my own path. They spent many long evenings helping me with projects while I was growing up and several of these gadgets were entered in a CMU sponsored robotics competition in 1988 and 1989. Graduating with a Robotics Ph.D from the same institution 19 years later is not something I would have predicted at the time, but it just goes to show you...

Table of Contents

1	Introduction	1
1.1	Thesis Statement	2
1.2	Overview of contributions	2
1.3	Thesis organization and themes	2
1.4	Mathematical notation conventions	3
2	Medical augmented reality	5
2.1	Image guidance of medical procedures	5
2.2	Medical augmented reality	6
2.3	Varieties of medical AR implementations	6
2.4	Quantitative and qualitative evaluation of medical AR systems	9
2.4.1	Quantitative evaluation of Fuchs’s system	9
2.4.2	Quantitative and qualitative and testing of Sauer’s system	9
2.4.3	Quantitative evaluation of the Sonic Flashlight	10
2.5	The psychophysics of depth perception	10
2.6	Motivation for an alternative testing environment	10
3	Virtual Tomographic Reflection	11
3.1	Introduction	11
3.2	Prototype VTR System	13
3.2.1	Hardware and software design	13
3.2.2	Prototype testing results	15
3.3	Second-generation VTR: System specification	16
3.4	Tool design	17
3.4.1	Mock Sonic Flashlight	17
3.4.2	Tracked needle tool	21
3.4.3	Tracked image container	21

3.5	Optotrak system usage	23
3.5.1	Physical system arrangement	23
3.5.2	Tracking rigid bodies	23
3.5.3	Using remote frames of reference	23
3.5.4	Marker visibility concerns	26
3.6	Frames of reference	26
3.7	Calibration of the VTR display	27
3.7.1	Known coordinate system transforms & data	28
3.7.2	Display calibration procedure	28
3.7.3	Results of display calibration	30
3.8	Calibrating the image frame of reference	31
3.9	Calibrating the needle tool	31
3.10	Computing the cutting plane	32
3.11	Computing the needle-plane intersection	35
3.12	VTR software	36
3.12.1	Calibration	36
3.12.2	Data collection and simulation with VUE	37
3.12.3	Data analysis	38
3.13	Photographs of the VTR system in operation	41
4	Overview of subject testing experiments and procedures	45
4.1	Introduction	45
4.2	Needle-slice feedback methods	45
4.3	Coordinate frames for analysis	46
4.4	Analysis procedures	52
4.5	Summary of experiments	52
4.5.1	Experimental subjects	53
5	Experiment 1: The effect of image complexity and target depth on SF accuracy	55
5.1	Experimental protocol	55
5.1.1	Experiment design	55
5.1.2	Task	55
5.1.3	Training	56
5.2	Key to reference frames	56
5.3	Statistics	57
5.4	Needle pointing results	58
5.4.1	Error vector magnitude	58

5.4.2	Error along V_x axis	59
5.4.3	Error along I_z axis	60
5.4.4	Error along V_y axis	61
5.4.5	Error along V_z axis	62
5.4.6	Error magnitude in V_{xy} plane	63
5.5	Needle insertion results	64
5.5.1	Error vector magnitude	64
5.5.2	Error along V_x axis	65
5.5.3	Error along I_z axis	66
5.5.4	Error along V_y axis	67
5.5.5	Error along V_z axis	68
5.5.6	Error magnitude in V_{xy} plane	69
5.6	Ranking subjects by performance	70
5.6.1	Introduction	70
5.6.2	Ranking subject performance by V_{xy} error vector magnitude	70
5.6.3	Correlation between mean subject performance and high - low noise performance	74
5.7	Summary of Experiment 1 results	77
6	Experiment 2: The effect of needle tip feedback on SF accuracy	79
6.1	Experimental protocol	79
6.1.1	Experiment design	79
6.1.2	Task	79
6.1.3	Training	80
6.2	Key to reference frames	80
6.3	Statistics	81
6.4	Needle pointing results	82
6.4.1	Error vector magnitude	82
6.4.2	Error along V_x axis	83
6.4.3	Error along V_y axis	84
6.4.4	Error along V_z axis	85
6.5	Needle insertion results	86
6.5.1	Error vector magnitude	86
6.5.2	Error along V_x axis	87
6.5.3	Error along V_y axis	88
6.5.4	Error along V_z axis	89
6.6	Tip insertion results	90

6.6.1	Tip insertion magnitude	90
6.6.2	Error along V_x axis	91
6.6.3	Error along V_y axis	92
6.6.4	Error along V_z axis	93
6.7	Summary of Experiment 2 results	94
7	Experiment 3: A comparison of SF and CUS guidance	95
7.1	Experimental protocol	95
7.1.1	Experiment design	95
7.1.2	Task	96
7.1.3	Training	96
7.2	Key to reference frames	96
7.3	Statistics	97
7.4	Needle pointing results	98
7.4.1	Error vector magnitude	98
7.4.2	Error along V_x axis	101
7.4.3	Error along V_y axis	104
7.5	Needle insertion results	105
7.5.1	Error vector magnitude	105
7.5.2	Error along V_x axis	108
7.5.3	Error along V_y axis	111
7.5.4	A comparison of performance with SF and CUS in V_x during insertion	112
7.6	Summary of Experiment 3 results	115
8	Experiment 4: Meta analysis of Experiments 1 and 3	117
8.1	Introduction	117
8.2	Needle pointing results	118
8.2.1	Error along V_x axis	118
8.2.2	Error along V_y axis	119
8.3	Needle insertion results	120
8.3.1	Error along V_x axis	120
8.3.2	Error along V_y axis	121
8.4	Summary of Experiment 4 Results	122
9	Conclusion and future work	123
9.1	Simulation design	123
9.2	Psychophysics	123

9.3	Contributions of this research	124
9.4	Future work	125
9.4.1	Additional analysis techniques	125
9.4.2	New phantom designs	125
9.4.3	Additional visualization techniques	125
A	Common Terms, Abbreviations, and Notations	127
B	Example data frame from VTR system	129
C	Experiments 1 & 2 image parameters	131
D	Experiment 3 image parameters	133
E	List of third party software, toolkits, and programming languages	137
E.1	Languages	137
E.2	Toolkits	137

List of Figures

2.1	Drascic and Milgram's reality - virtuality continuum.	6
2.2	Sauer's HMD and tracking system [4].	8
2.3	Fuch's HMD [6].	8
2.4	Schematic of the Sonic Flashlight.	9
3.1	Original concept of virtual tomographic reflection [16].	12
3.2	Overview of the virtual Sonic Flashlight [16].	14
3.3	Registration coordinates in the virtual tomographic reflection prototype [16].	14
3.4	VTR slice through an MRI of the brain, displayed on laptop computer (left) and Sonic Flashlight (right) [16].	15
3.5	Plot of needle tip to target distance (position error) and needle tip speed while aiming for a virtual sphere. The error is 0 if the operator achieves perfect placement [16]. . . .	16
3.6	Schematic of VTR system.	18
3.7	A single cycle of VTR execution.	19
3.8	CAD schematic of mock Sonic Flashlight casing.	20
3.9	Removable display shield.	20
3.10	Mock SF tracking cluster wiring and strain relief.	21
3.11	Mock SF showing operation of the OLED.	22
3.12	Components of the tracked needle; IRED cluster (top), removable needle holder (middle), additional needle (bottom).	22
3.13	The tracked image container, with the mock SF and needle tool for scale reference. . .	23
3.14	Configuration of the room in which the VTR system is installed.	24
3.15	Defining and using a remote reference frame.	25
3.16	Coordinate systems used in VTR display calibration.	27
3.17	Aligning the pen tool in S coordinates with two calibration targets in V coordinates. The white dot is reflected in the half-silvered mirror from the OLED display hidden beneath the glare-shield.	29

3.18	Image frame of reference calibration. Because the image container rests flat on the surface of the table, I_z is approximately parallel to gravity.	32
3.19	Needle calibration.	33
3.20	Virtual image coordinate system.	34
3.21	Virtual image coordinates V relative to image dataset coordinates I , illustrating a slice through the image data.	35
3.22	Needle intersection computation.	37
3.23	Screen capture of VUE, main window.	39
3.24	Screen capture of VUE, image generation window.	40
3.25	Screen capture of VUE, data collection window.	40
3.26	A subject using the VTR system in SF mode to view a target.	41
3.27	A subject using the VTR system in CUS mode to view a target.	42
3.28	A comparison of a real phantom viewed using the clinical SF (left) and similar VTR phantom (right).	43
4.1	Example of the red bar slice feedback mode. In this image, the subject can tell that they have hit the slice plane but cannot tell where on the slice plane the needle has hit. The white dot visible on the virtual image is the target, not the needle intersection.	47
4.2	Example of a high-noise point overlay. The subject has partially inserted the needle but has not yet touched the image plane.	48
4.3	Schematic of perceived target location when using the SF in a reduced cue condition.	49
4.4	Schematic of perceived target location when using the SF in a reduced cue condition, looking at the virtual image plane en face.	50
4.5	Schematic of perceived target location when using the SF in a reduced cue condition, looking at the virtual image plane on edge.	51
5.1	Pointing, error vector magnitude, mean \pm standard error.	58
5.2	Pointing, V_x error, mean \pm standard error.	59
5.3	Pointing, I_z error, mean \pm standard error.	60
5.4	Pointing, V_y error, mean \pm standard error.	61
5.5	Pointing, V_z error, mean \pm standard error.	62
5.6	Pointing, per-subject mean errors in V_{xy} plane for each target, mean \pm standard error.	63
5.7	Pointing, centroids of all subjects in V_{xy} plane, mean \pm standard error.	63
5.8	Insertion, error vector magnitude, mean \pm standard error.	64
5.9	Insertion, V_x error, mean \pm standard error.	65
5.10	Insertion, I_z error, mean \pm standard error.	66
5.11	Insertion, V_y error, mean \pm standard error.	67

5.12	Insertion, V_z error, mean \pm standard error.	68
5.13	Insertion, per-subject mean errors in V_{xy} plane for each target.	69
5.14	Insertion, centroids of all subjects in V_{xy} plane, mean \pm standard error.	69
5.15	Correlation of mean errors with standard deviation in V_{xy} plane, per subject, during pointing.	72
5.16	Correlation of mean errors with standard deviation in V_{xy} plane, per subject, during insertion.	73
5.17	Correlation between V_x error and high vs. low noise difference during pointing.	75
5.18	Correlation between V_x error and high vs. low noise difference during insertion.	76
6.1	Pointing, error vector magnitude, mean \pm standard error.	82
6.2	Pointing, V_x error, mean \pm standard error.	83
6.3	Pointing, V_y error, mean \pm standard error.	84
6.4	Pointing, V_z error, mean \pm standard error.	85
6.5	Insertion, error vector magnitude, mean \pm standard error.	86
6.6	Insertion, V_x error, mean \pm standard error.	87
6.7	Insertion, V_y error, mean \pm standard error.	88
6.8	Insertion, V_z error, mean \pm standard error.	89
6.9	Tip insertion, magnitude error, mean \pm standard error.	90
6.10	Tip insertion, V_x error, mean \pm standard error.	91
6.11	Tip insertion, V_y error, mean \pm standard error.	92
6.12	Tip insertion, V_z error, mean \pm standard error.	93
7.1	Pointing, error vector magnitude, mean \pm standard error.	98
7.2	Pointing, error vector magnitude, no noise condition, mean \pm standard error.	99
7.3	Pointing, error vector magnitude, high noise condition, mean \pm standard error.	100
7.4	Pointing, V_x error, mean \pm standard error.	101
7.5	Pointing, V_x error, no noise condition, mean \pm standard error.	102
7.6	Pointing, V_x error, high noise condition, mean \pm standard error.	103
7.7	Pointing, V_y error, mean \pm standard error.	104
7.8	Insertion, error vector magnitude, mean \pm standard error.	105
7.9	Insertion, error vector magnitude, no noise condition, mean \pm standard error.	106
7.10	Insertion, error vector magnitude, high noise condition, mean \pm standard error.	107
7.11	Insertion, V_x error.	108
7.12	Insertion, V_x error, no noise condition, mean \pm standard error.	109
7.13	Insertion, V_x error, high noise condition, mean \pm standard error.	110
7.14	Insertion, V_y error, mean \pm standard error.	111

7.15	Correlation between the improvement of SF over CUS and the mean error of SF and CUS.	113
7.16	Correlation between the improvement of SF over CUS and the mean error of SF and CUS, excluding the 2 subjects with the worst improvement.	114
8.1	Pointing, V_x error, mean \pm standard error.	118
8.2	Pointing, V_y error, mean \pm standard error.	119
8.3	Insertion, V_x error, mean \pm standard error.	120
8.4	Insertion, V_y error, mean \pm standard error.	121

List of Tables

5.1	Pointing, error vector magnitude, mean (stdev), values in mm.	58
5.2	Pointing, V_x error, mean (stdev), values in mm.	59
5.3	Pointing, I_z error, mean (stdev), values in mm.	60
5.4	Pointing, V_y error, mean (stdev), values in mm.	61
5.5	Pointing, V_z error, mean (stdev), values in mm.	62
5.6	Insertion, error vector magnitude, mean (stdev), values in mm.	64
5.7	Insertion, V_x error, mean (stdev), values in mm.	65
5.8	Insertion, I_z error, mean (stdev), values in mm.	66
5.9	Insertion, V_y error, mean (stdev), values in mm.	67
5.10	Insertion, V_z error, mean (stdev), values in mm.	68
5.11	Subject ranking by V_{xy} error during pointing, mean (stdev), values in mm.	71
5.12	Subject ranking by V_{xy} error during insertion, mean (stdev), values in mm.	71
5.13	Subject ranking by High - No Noise difference in V_x during pointing.	74
5.14	Subject ranking by High - No Noise difference in V_x during insertion.	75
6.1	Pointing, error vector magnitude, mean (stdev), values in mm.	82
6.2	Pointing, V_x error, mean (stdev), values in mm.	83
6.3	Pointing, V_y error, mean (stdev), values in mm.	84
6.4	Pointing, V_z error, mean (stdev), values in mm.	85
6.5	Insertion, error vector magnitude, mean (stdev), values in mm.	86
6.6	Insertion, V_x error, mean (stdev), values in mm.	87
6.7	Insertion, V_y error, mean (stdev), values in mm.	88
6.8	Insertion, V_z error, mean (stdev), values in mm.	89
6.9	Tip insertion, magnitude error, mean (stdev), values in mm.	90
6.10	Tip insertion, V_x error, mean (stdev), values in mm.	91
6.11	Tip insertion, V_y error, mean (stdev), values in mm.	92
6.12	Tip insertion, V_z error, mean (stdev), values in mm.	93

7.1	Pointing, error vector magnitude, mean (stdev), values in mm.	98
7.2	Pointing, V_x error, mean (stdev), values in mm.	101
7.3	Pointing, V_y error, mean (stdev), values in mm.	104
7.4	Insertion, error vector magnitude, mean (stdev), values in mm.	105
7.5	Insertion, V_x error, mean (stdev), values in mm, mean \pm standard error.	108
7.6	Insertion, V_y error, mean (stdev), values in mm.	111
7.7	Difference between CUS and SF during insertion, V_x error, values in mm. Positive CUS-SF values indicate an improvement of the SF over CUS.	112
8.1	Pointing, V_x error, mean (stdev), values in mm.	118
8.2	Pointing, V_y error, mean (stdev), values in mm.	119
8.3	Insertion, V_x error, mean (stdev), values in mm.	120
8.4	Insertion, V_y error, mean (stdev), values in mm.	121
C.1	Experiments 1 & 2 practice set.	131
C.2	Experiments 1 & 2 experiment set.	132
D.1	Experiment 3 practice set.	133
D.2	Experiment 3, CUS experiment set.	134
D.3	Experiment 3, SF experiment set.	135

Chapter 1

Introduction

Image-based guidance of medical procedures is a common way of improving the outcome of what would otherwise be exceptionally difficult tasks. Biopsy, catheter insertion, and highly-localized injections are all commonly performed under image guidance. In addition, many more common tasks such as routine IV insertion would be good candidates for medical image guidance if not for the training requirements and/or the expense of the equipment involved.

Among the real-time medical imaging modalities ultrasound is unique in being relatively inexpensive and portable, making it a good candidate for deployment throughout a medical facility. Ultrasound is safe for extended use, posing no risk to either the operator or patient. Modern machines provide extremely high-quality images over a variety of focus depths, and many offer additional features such as doppler flow display. Interpretation of ultrasound images, however, and in particular the process of using them to provide real-time feedback of an invasive procedure such as needle biopsy, can be a very challenging task for novice users.

Much of the difficulty in medical image guidance centers on mentally fusing the medical image with the patient's body. Augmented reality (AR) systems can alleviate this difficulty by visually fusing the medical image with the patient's body, allowing much more natural guidance. Questions that can be asked about augmented reality systems in medicine include A) "Do these systems actually improve the ability of medical personnel to treat the patient?" and B) "How do these systems affect perception of the medical image information?"

In this dissertation I explore the use of a virtual environment for conducting accuracy and psychophysical experiments on medical augmented reality systems by implementing a virtual version of the Sonic Flashlight (SF). In contrast to conventional phantom-based analysis of augmented reality systems, the virtual environment provides quantitative performance measures within a high-precision global frame of reference. In addition, the ease with which the virtual environment may be manipulated allows for a much more rapid exploration of psychophysical phenomena than would be possible with physical phantoms. The long term goals for this project are to support an ongoing effort to examine the impact of augmented reality technologies on medical procedures, and to provide a test environment for future augmented reality displays.

1.1 Thesis Statement

This thesis addresses the design and implementation of a Virtual Tomographic Reflection (VTR) system that mimics the behavior of the Sonic Flashlight (SF) augmented reality device. Using the VTR system, I explore the psychophysics of virtual image displays and how various image parameters affect a user's perception of target location. I also examine the accuracy of the Sonic Flashlight versus conventional ultrasound (CUS) using criteria that would be difficult to measure in a non-virtual system.

1.2 Overview of contributions

This thesis makes contributions in two areas, simulation design and psychophysics:

- ★ Simulation design (Chapter 3)
 - Defining the functional components of Virtual Tomographic Reflection (Section 3.3)
 - Design of a tracked Sonic Flashlight (Section 3.4)
 - A VTR calibration algorithm (Section 3.7)
 - Computational methods for high-performance 3D image slicing (Section 3.10)
- ★ Psychophysics
 - Analysis of the effect of image background complexity and target depth on user performance (Chapters 5 and 8).
 - Analysis of the effect of needle/image intersection feedback on user performance (Chapter 6).
 - A comparison of SF to CUS guidance with varying background complexity and target depths (Chapter 7).

1.3 Thesis organization and themes

Two primary themes appear throughout this thesis and relate to the general categories of simulation and psychophysics contributions listed above.

1. Simulation - A simulated environment can provide many advantages over its real world counterpart when looking at complicated relationships between the environment and human performance.
2. Psychophysics - Consider a target that a user is attempting to hit under AR guidance. In addition to characteristics of the target such as size and depth, certain parameters of the guidance image also affect the user's performance.

The first of these themes was the primary motivation for this project. In Chapter 2, I discuss existing AR implementations and how researchers have characterized the performance of AR systems relative

to conventional practice. The majority of these devices have been evaluated using physical phantom-based tests, and I identify specific limitations inherent to this approach. I also summarize work that has been done in our lab to evaluate the performance of the SF, as well as the psychophysics of viewing in-situ images. These observations lead directly to the design of a virtual environment for testing the SF, Chapter 3, where I first discuss an early prototype of VTR and then present the design and calibration of the final system.

An observation I made during calibration of the VTR system – the importance of background structure in the virtual image for correct perception of target depth – leads to the second primary theme of this thesis. In Chapter 4, I explain the general design of psychophysics experiments performed using the VTR device, and in Chapters 5 through 8, I present the results of these experiments. Finally, in Chapter 9, I discuss ways that VTR may be extended in the future to areas such as general medical image visualization and training, and suggest future directions for psychophysics research using the VTR system.

1.4 Mathematical notation conventions

I have endeavored to abide by the following notation scheme throughout this document:

1. Matrices are identified with bold capital letters; e.g. **M**
2. Matrix dimensions are specified in row-major format, i.e. rows x columns
3. N-d vectors are identified with bold lowercase letters; e.g. **v**
4. N-d points are also identified with bold lowercase letters and carry the implication that they are a vector from the global origin to a particular location in N-space. Points may be distinguished from vectors by context.
5. A vector from point **a** to **b** is computed by taking **b** – **a**. Assignment to a single letter is permissible, e.g. **x** = **b** – **a**
6. I define two vector norm operators. The first operator returns the scalar magnitude *m* of the vector **v** and is notated $m = ||\mathbf{v}||$. The second operator returns a normalized vector rather than a scalar: $\mathbf{v} = |\mathbf{b} - \mathbf{a}|$, which can be written more verbosely as $\mathbf{v} = \frac{\mathbf{b} - \mathbf{a}}{||\mathbf{b} - \mathbf{a}||}$
7. Points and vectors should be construed to be in either homogeneous or non-homogeneous form, whichever is appropriate given the context.
8. Reference frames/coordinate systems are identified with italicized capital letters; e.g. *R*. Mathematically they are represented by 4x4 homogeneous matrices and are named by the matrix (for frame *R* the underlying matrix is **R**) that describes their position and orientation relative to an implied global reference frame *G*.
9. The transformation from reference frame *A* to frame *B* is notated as $T_{A \rightarrow B}$.
10. Given a point **p** and two reference frames *A* and *B*, the location of **p** in each frame is *A*(**p**) and *B*(**p**). Mathematically, given **p** in global coordinates and expressed in homogeneous form, $A(\mathbf{p}) = \mathbf{A} \cdot \mathbf{p}$

11. Given a reference frame F in 3-space, the x-axis component of the rotation submatrix is notated by F_x , the y-axis by F_y , and the z-axis by F_z . The vector components of a rotation submatrix are always normalized.
12. Scalars are represented by non-bold lowercase letters; e.g. s
13. Given a 3-dimensional vector \mathbf{v} , its three scalar components are \mathbf{v}_x , \mathbf{v}_y , and \mathbf{v}_z .

Chapter 2

Medical augmented reality

2.1 Image guidance of medical procedures

Direct visual examination of a patient by a physician is both the oldest and simplest form of medical imaging. And, despite a wealth of alternative techniques developed in the past 100 years, direct vision is still the most widely used form of medical imaging. Conducting an external physical exam remains a fundamental skill in medicine, and only recently has conventional surgery (with large incisions and direct examination of the affected area) given way to minimally invasive surgery, where indirect methods are used to observe the surgical field. Direct vision of a patient has many advantages – it requires no special equipment, provides a wealth of spatial cues about size and shape, and relies on perceptual skills that humans have evolved to intuitively understand.

Direct visual examination fails, of course, on structures that are hidden beneath the skin. Since the development of x-ray technology in 1896, it is been possible to conduct noninvasive examination of internal structures that are not accessible to direct vision. Today, a wide variety of technologies exist for noninvasive medical imaging. In addition to x-rays (radiographs), physicians use computed tomography (CT), magnetic resonance imaging (MRI), positron emission tomography (PET), and ultrasound (US). It should also be noted that laparoscopy, and other techniques that are minimally invasive rather than noninvasive, are commonly used in clinical medicine.

Modern medical imaging serves two distinct purposes – diagnosis and image-guided therapy. In the former, a physician uses his or her knowledge of anatomy and understanding of the imaging modality's interaction with human tissue to identify abnormalities present in the image. Although a mental representation of the image inside the patient's body is helpful, it is not, in the strictest sense, required to make the diagnosis. In fact, traditional radiology rarely involves the physician "reading" the medical image in real-time while it is being acquired, or even while the patient is present.

In contrast to diagnosis, image-guided therapy requires a physician to respond to a medical image in real-time (or near real-time) in order to perform a procedure. Ultrasound is a particularly common imaging modality for image-guided therapy because it presents no risk to the operator or patient, is relatively portable, and (at least compared to MRI and CT) is low cost. Unlike direct vision of a patient, ultrasound guidance – and indeed image guidance in general – requires an additional step where the physician mentally places the image inside the patient's body. It is this step that proves challenging to novice users, and serves to restrict image-guided therapy to a relatively small number of highly skilled

clinical personnel.

Restricting image guidance to a small set of skilled users is unfortunate because a number of common procedures would otherwise be ideal candidates for image guidance. Instead, these procedures – among them, insertion of catheters into veins (cannulation) – are usually performed using direct vision or palpation of surface landmarks. Landmark guidance is unfortunately error-prone (even for experienced physicians), with failure rates of initial insertion of 18.7 percent, 15.9 percent, and 35.0 percent for cannulation of the internal jugular vein, subclavian vein, and femoral vein respectively [1].

2.2 Medical augmented reality

To address the difficulties that image guidance presents to novice users, a class of computer visualization techniques known as augmented reality (AR) have been applied to image-guided therapy. AR is part of what Drascic and Milgram define as the Reality-Virtuality Continuum [2]. At one end of the continuum are purely virtual environments, presented to the user via stereo graphics (SG), while at the other end are completely real environments, seen by the user with either direct vision (DV) or mediated by real-time stereo video (SV)

Between the extremes lie two classes of mixed reality environments. The first of these is augmented virtuality (AV), where a predominantly virtual environment is enhanced by the addition of images of objects from the real world. The second mixed reality technique, AR, enhances what is predominantly a real scene with virtual objects.

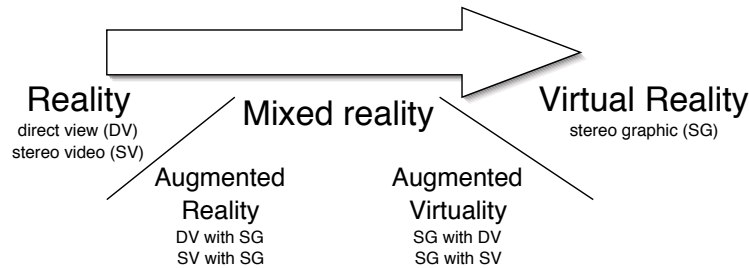


Figure 2.1: Drascic and Milgram's reality - virtuality continuum.

In the task domain of medicine, all four portions of the continuum are useful (pure virtual environments are useful for training, for example). However, for the particular case of needle biopsy AR techniques are clearly the most appropriate. After all, both the patient and needle are real, physical objects. The “virtual” addition to the scene need only consist of the image data being used to guide the biopsy procedure.

2.3 Varieties of medical AR implementations

The most common method of achieving image overlay is to either fully (via stereo video, SV) or partially replace the operator's direct vision (DV) by means of a head-mounted display (HMD). Several varieties of HMD exist, with the two most common being optical see-through and video see-through [3]. In

the former, a partially reflective (half-silvered) mirror combines direct vision of an object with a stereographic (SG) overlay. In the latter, the user's direct vision is completely occluded by video displays that show a combination of images from head-mounted video cameras and the virtual environment generated by a computer. A third HMD design known as retinal scanning draws the augmented image directly on the user's eye using low-powered lasers, but this technique is much less common than the techniques discussed previously.

Since the optical see-through approach consists of a predominately direct view of the environment, it is inherently higher resolution than the video see-through technique, which replaces the user's direct view with a lower resolution video image. Unfortunately, all optical see-through designs suffer from some degree of misregistration between the real and virtual environments. Since the virtual environment requires some amount of time to be generated, whereas direct vision occurs instantly, virtual objects generated by the AR system typically lag behind the real world whenever the user's perspective on the scene changes. Video see-through approaches can avoid this problem by delaying the display of the video image until the virtual environment rendering process can be completed. Although this does result in overall lag of the system, virtual and real objects always appear to line up correctly. Overall system lag, with correct registration between the real and virtual worlds, tends to be less noticeable than lag of the virtual world relative to the real world [4].

Two of the most advanced medical AR projects both use the video see-through approach to AR [4, 5]. These systems track the relative position and orientation of the operator's head, surgical tools, and patient's body in order to present a correct augmented image to the user. Fuch's system is specialized for augmenting ultrasound-guided breast biopsy, while Sauer's system has been demonstrated with both ultrasound and intraoperative MRI (iMRI).

Both Sauer and Fuchs utilize variants of optical marker tracking, although the former uses a custom passive¹ system, while the latter uses a commercial active system². Sauer's system uses a tracker which is integrated into the HMD (see Figure 2.2). As compared to the off-board tracking used by Fuchs, the integral tracking increases the portability of the system while somewhat limiting the workspace because of the smaller field of view of the head-mounted camera. Note that because the tracking in Fuchs's system is offboard, their HMD (see Figure 2.3) is quite a bit smaller than Sauer's.

Unfortunately, present HMD systems suffer from several problems, including tracking lag, low resolution of the displays, limited field of view, weight, and cost. Additionally, in a multi-user environment, each user must wear his or her own HMD in order to perceive the augmented environment. To overcome many of the limitations of HMDs, Stetten and Masamune separately proposed a simpler technology which we refer to as real-time tomographic reflection (RTTR) [7, 8]. Although similar in design, Stetten's system was originally developed for real-time visualization of ultrasound, while Masamune's was focused on static (i.e not real time) display of CT images.

RTTR avoids some of the drawbacks of HMD systems but achieves a similar effect by fixing the relative geometry of the transducer, the display, and a half-silvered mirror to produce a virtual image of the sonographic image within the body (see Figure 2.4). Each pixel in the sonographic image seems to emanate from its correct location. Thus the patient, the sonographic image, the instrument, and the

¹Passive optical systems function by tracking reflective markers with a video camera. Sauer tracks retroreflective spheres illuminated with an infrared light source mounted in a ring around the camera lens.

²Fuchs uses the FlashPointTM5000 tracking system, which tracks infrared LED markers using several cameras mounted around the workspace.



Figure 2.2: Sauer's HMD and tracking system [4].



Figure 2.3: Fuch's HMD [6].

operator's hands are merged into one environment for all observers looking through the half-silvered mirror. Because of the ray-like *in situ* appearance of a curvilinear ultrasound image, this implementation of RTTR has been named the Sonic Flashlight. Since no positional tracking or head-mounted apparatus is required, the cost of equipping a sonographic machine with RTTR is relatively small. In addition, the Sonic Flashlight requires only a minimal amount of computation to properly scale and locate the image on the screen. Given that the registration between virtual image and ultrasound beam is inherent in the geometry of the device, and image post-processing is minimal, there is very little lag between data acquisition and virtual image generation (as compared to devices that must also track both the user and probe and compute the geometric relationship dynamically).

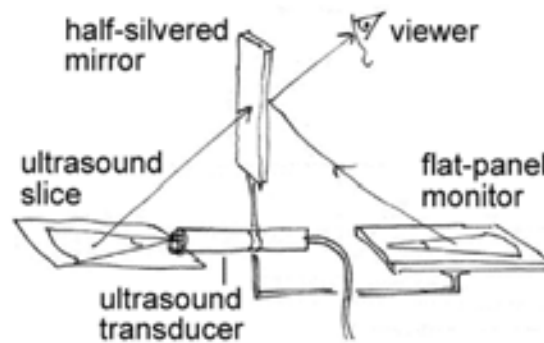


Figure 2.4: Schematic of the Sonic Flashlight.

2.4 Quantitative and qualitative evaluation of medical AR systems

2.4.1 Quantitative evaluation of Fuch's system

A study by Rosenthal and colleagues at UNC of Fuch's system [6] - the first that I am aware of to provide a quantitative analysis of AR guided needle biopsy - found that a "video see-through" device produced a statistically significant improvement in the accuracy of needle placement during ultrasound guided biopsy. In their study, a board-certified radiologist performed 5 biopsies (at the center, and 4 equally spaced points around the perimeter) on each of 10 mock tumor targets within a breast phantom, with the choice of augmented reality or standard ultrasound guidance made randomly prior to selecting each target.

A second radiologist, who was blind to the type of guidance used during each biopsy, evaluated needle placement in each trial by examining video footage of the ultrasound image recorded during the procedure. Needle placement error was computed by physically measuring the video image on a video monitor. Results of mean error over all 5 locations were 2.48 ± 0.44 mm for standard guidance and 1.62 ± 0.48 mm for augmented reality guidance. Results for the error at the center were 4.20 ± 1.92 mm for standard guidance and 1.50 ± 1.41 mm for augmented reality guidance.

2.4.2 Quantitative and qualitative and testing of Sauer's system

Sauer et al. have extensively validated their system using a variety of approaches, ranging from a completely synthetic approach based on an abstract pointing task [9], a phantom-based approach [10, 11], and most recently animal models [12]. Their initial phantom-based evaluation was based on a straightforward task pressing buttons by guiding a needle through a foam barrier, although the system has progressed in complexity and now incorporates simulation components that require the user to avoid other critical structures. This most recent development is of particular interest for both PICC insertion, where the doctor is required to hit a vein while avoiding surrounding vessels, and breast biopsy, where the lesion must be hit while avoiding inadvertent puncture of the lung.

2.4.3 Quantitative evaluation of the Sonic Flashlight

SECTION NOT YET COMPLETE 

2.5 The psychophysics of depth perception

SECTION NOT YET COMPLETE 

2.6 Motivation for an alternative testing environment

SECTION NOT YET COMPLETE 

Chapter 3

Virtual Tomographic Reflection

3.1 Introduction

In this chapter I describe my implementation of a Virtual Tomographic Reflection (VTR) system to explore the psychophysics and operator performance issues described in Chapter 2. As the name implies, VTR uses the tomographic reflection display technique from the Sonic Flashlight to allow the operator to interact with a virtual reality environment. I refer to this environment as “virtual” because the images involved are either generated synthetically or derived from stored image data such as CT or MRI. From a user’s perspective - in the ideal case - it would be difficult or impossible to distinguish between an actual Sonic Flashlight and a VTR system displaying what appeared to be real time in situ ultrasound images.

A note about the name *Virtual Tomographic Reflection*: I use the word *virtual* here in the same manner as in *virtual reality*, where it implies that the system is simulated or synthetic. It is important to distinguish this from the term *virtual image*, which I use in this dissertation to refer to a specific optical effect. Unless otherwise noted, any use of the phrase *virtual image* refers to the optics definition. Any other use of *virtual* means simulated or synthetic (See Appendix A).

It is tempting to think of VTR as “virtual ultrasound” – in fact, the simulation software used in my implementation of VTR is name *Virtual Ultrasound Environment* (VUE) – however, VTR in general, and VUE in particular, does not simulate ultrasound per se. Rather, VTR simulates tomographic slicing of any 3D image volume with an infinitely thin slice. While VTR is visually similar to ultrasound in many respects, it’s worth noting that a proper simulation of ultrasound would model the acoustic behaviors or tissue and generate the image based on a model of the ultrasound image reconstruction process. There are two primary advantages to the way I have chosen to implement VTR over the ultrasound modeling approach. First, VTR is extremely fast and runs at video frame rates on a standard desktop computer even with very large image volumes (Section 3.10 describes my hardware accelerated implementation of tomographic slicing). Secondly, VTR allows extremely fine-grained control of the generated generated, a feature that is useful for both psychophysics and operator performance experimentation.

The VTR system includes the following general components (see Figure 3.1):

1. A mock Sonic Flashlight device for displaying a virtual image. “Mock” means that there is no

functioning ultrasound system, although in Future Work I describe an extension of VTR that makes use of a functional ultrasound transducer to improve the realism of the simulation.

2. A method of tracking the position and orientation of the Sonic Flashlight and any tools (such as a needle) that need to interact with the medical image.
3. Software that computes the cutting plane through the medical image volume, simulates any interaction of tools with the medical image, and allows proper calibration of the simulated workspace.

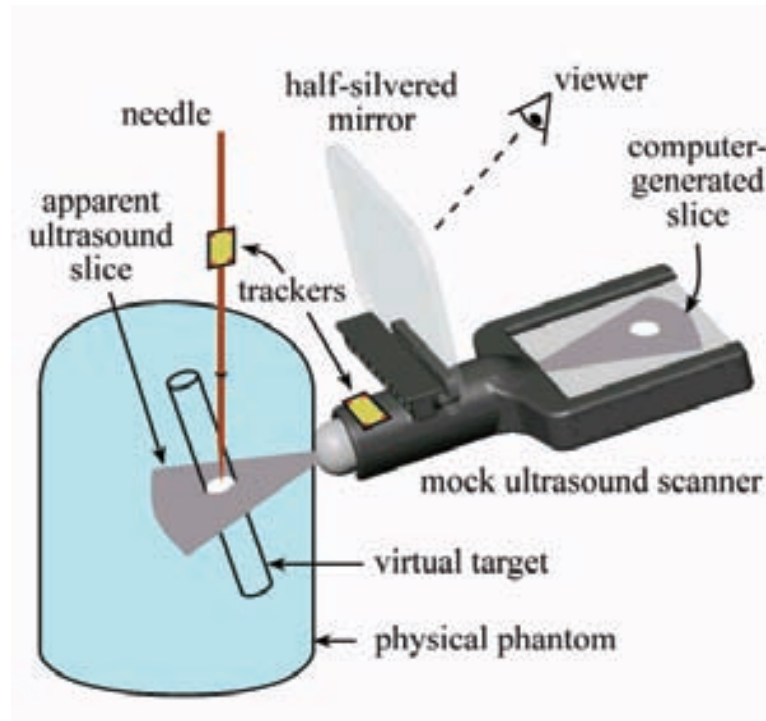


Figure 3.1: Original concept of virtual tomographic reflection [16].

For psychophysical studies that compare the Sonic Flashlight to conventional ultrasound, the VTR system can be operated in *virtual CUS mode*, where the Sonic Flashlight display is disabled and the simulated ultrasound image is instead sent to an external flat panel display that mimics a conventional ultrasound display. When operating in CUS mode, the subject still uses the virtual Sonic Flashlight physical apparatus, eliminating any performance differences between the two display modalities that would result from variations in weight or shape of the probe.

As compared to ultrasound simulations based on physical phantoms, VTR provides the following advantages:

1. Since the needle and the mock ultrasound transducer are both tracked simultaneously with the generation of the simulated target, VTR is capable of precisely calculating the distance between the tip of the needle and the target point within the virtual object.

2. It is extremely easy to switch between different virtual phantoms, since this only involves a change in the image data. This in contrast with the need to physically switch between phantoms with different targets, when using the conventional ultrasound phantoms containing gel.
3. VTR provides fine-grained control of the image presented to the user. Features such as background noise density and target size can be controlled very precisely, in ways that would be difficult or impossible using physical phantoms.

3.2 Prototype VTR System

My first implementation of VTR [16] evaluated a low-cost magnetic tracking technology and explored calibration and simulation requirements. As a prototype, this system was not intended to be the final implementation of VTR and lacked several features - particularly in regards to calibration - that are quite important in the final design. This section discusses the prototype device and how the lessons learned motivated my design of the second “feature-complete” system.

3.2.1 Hardware and software design

The prototype system was based around the miniBirdTM magnetic tracking system from Ascension Technologies. At the time, the line-of-sight constraints imposed by an optical tracking system appeared to be an overriding concern. Magnetic tracking avoids this problem but is sensitive to the presence of electromagnetic fields or metal objects in the workspace. To help avoid magnetic interference, I used a wooden stylus in place of a needle and mounted the Sonic Flashlight tracker on a plastic crossbar that distanced it from the transducer body and flat panel display, both of which contain metal. The Sonic Flashlight used in this prototype was a modified form of what designated the model 4, without an active ultrasound scanner but otherwise identical in terms of display capabilities [17, 18, 19].

Registration of the transducer in tracker coordinates was accomplished by using the needle as a digitization stylus and sampling three points on the surface of the transducer (see Figure 3.3). The virtual ultrasound image is anchored at point b , and points a and c are used to establish an orthogonal coordinate system by taking successive cross products, as shown in Equations 3.1-3.3¹.

$$\mathbf{x} = |\mathbf{c} - \mathbf{a}| \tag{3.1}$$

$$\mathbf{y} = |\mathbf{c} - \mathbf{a}| \times |\mathbf{b} - \mathbf{a}| \tag{3.2}$$

$$\mathbf{z} = \mathbf{x} \times \mathbf{y} \tag{3.3}$$

The software for the magnetically tracked prototype used components of the Visualization Toolkit [20] to dynamically generate a planar cross-section through a 3D volume of image data, based on the computed position/orientation of the transducer tip. An additional component of the simulation was the

¹As described in Section 1.4, please note the distinction between the norm operator that returns a scalar, $\|\mathbf{v}\|$, and that which returns a vector, $|\mathbf{v}|$.

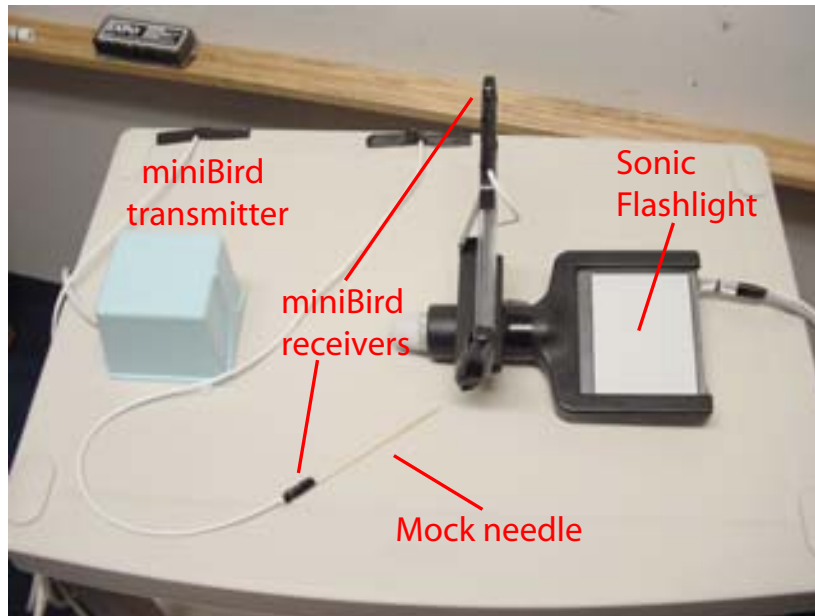


Figure 3.2: Overview of the virtual Sonic Flashlight [16].

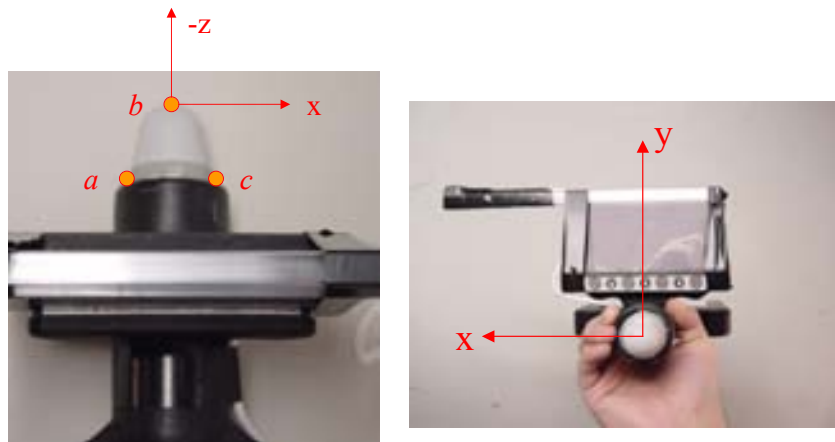


Figure 3.3: Registration coordinates in the virtual tomographic reflection prototype [16].

computation of the intersection, if any, of the needle vector with the virtual image plane. This intersection point was displayed as a small circle of bright intensity in the virtual image, mimicking the appearance of a needle viewed in cross section with ultrasound. The method for computing this intersection is described in Section 3.11.

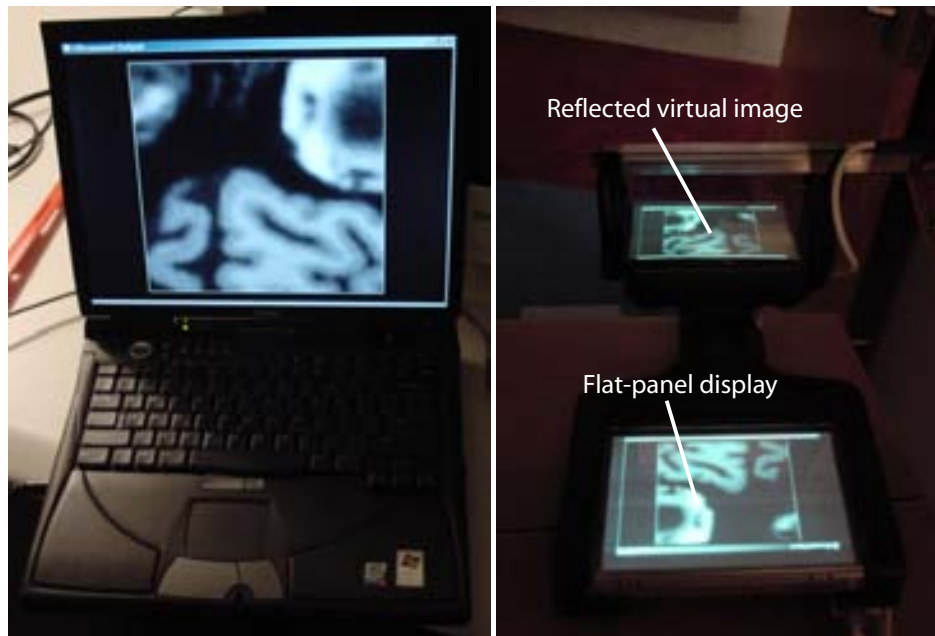


Figure 3.4: VTR slice through an MRI of the brain, displayed on laptop computer (left) and Sonic Flashlight (right) [16].

The prototype VTR system operated in a loop consisting of data acquisition, tool position/orientation computation, and slice rendering at approximately 20 Hz. One important aspect of biopsy simulation - haptic feedback - was extremely limited in the prototype system. A cardboard box registered in the miniBird coordinate system provided physical resistance to the needle. During needle insertion the walls of the box provided a lateral movement constraint on the needle entry point. By appropriately scaling and positioning the virtual image volume, the prototype achieved the illusion that the virtual image was “contained” within the box.

3.2.2 Prototype testing results

I performed a brief study of a single individual in order to qualitatively assess the functionality of our experimental setup. The user was presented with a virtual sphere containing an interior sphere of darker intensity to simulate a tumor and was instructed to insert the mock needle into the cardboard bounding box in a single smooth motion, attempting to terminate the movement as close to the center of the interior sphere as possible. The subject performed 5 sequential insertions and withdrawals of the mock needle. A plot of position error and needle tip speed for the second insertion is shown in Figure 3.5.

The subject consistently overshoot the center of the interior sphere by approximately 5 mm. This

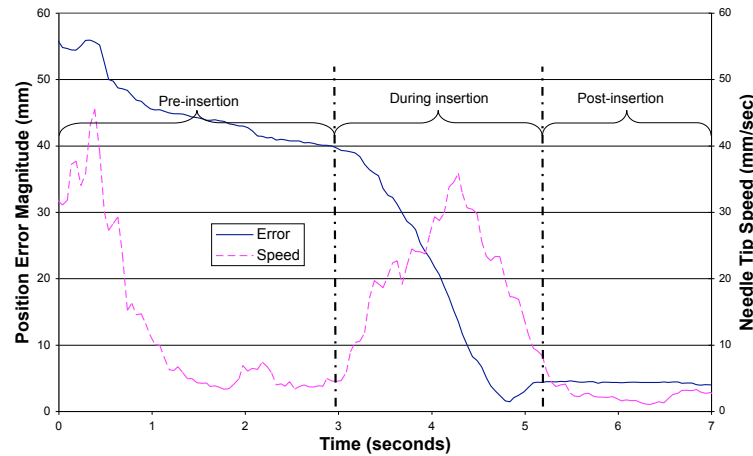


Figure 3.5: Plot of needle tip to target distance (position error) and needle tip speed while aiming for a virtual sphere. The error is 0 if the operator achieves perfect placement [16].

is visible in Fig. 3.5 as a rise in error magnitude at $t=5$ seconds, prior to the error stabilizing as the needle was held still. The consistent overshoot error most likely results from limitations in the magnetic tracking. It is well known that large pieces of metal within the Flock-of-birds workspace adversely affect is performance. In this case the effect appeared to be a systematic distortion of the tracked position, rather than random error.

Conclusions that I drew from the prototype are as follows:

1. Magnetic tracking is not a suitable tool for use with VTR, because of the unavoidable inclusion of metal objects within the workspace and subsequent distortion of reported tracking coordinates.
2. More precise calibration techniques were required in order to obtain meaningful data.
3. The way that the medical imaging slicing technique was implemented did not scale well to particularly large medical image volumes and was probably not suitable for large, high-resolution test images.

Despite our decision to switch tracking technologies, this initial implementation of virtual ultrasound phantoms was sufficient to demonstrate that such a system could acquire data for human factors analysis of various modes of image guided needle procedures.

3.3 Second-generation VTR: System specification

The design of the second-generation VTR system was directly motivated by the results from the prototype device (from this point forward any reference to the VTR device can be assumed to refer to the second-generation device). In particular, I wished to address the three areas of concern identified after conducting the brief pilot study with the prototype:

1. Magnetic tracking concerns: The second generation system is based on optical tracking, which is not subject to magnetic interference.
2. Calibration inaccuracies: The second generation system is calibrated using an extremely accurate method described in Section 3.7.
3. A higher-performance image reslicing technique: I now use a hardware accelerated 3D texture memory approach that gives very high performance, as described in Section 3.10.

VTR has three primary components²: a tracking system, tracked tools that the user manipulates, and a computer to perform the simulation of interaction between the tools and a stored 3D image. I use an Optotrak 3020 (Northern Digital, Inc.) optical tracking system, which supports high-frequency position and orientation capture of infrared LEDs (IREDs) grouped into coherent units known as rigid bodies. The Optotrak system is composed of a system control unit (SCU) that attaches to a host computer via a SCSI cable, one or more arrays of cameras that provide input to the SCU, and one or more IRED marker drivers that control the IREDs defining each rigid body.

Users operating the VTR system interact with three tools. Two of these, a mock SF and needle/pen tool are handheld, while the third - a tracked image “container” - rests on the surface of a table. The tracked image container may be freely positioned on the table but is typically not manipulated while the user is viewing images using the SF. Users typically hold the mock SF in their non-dominant hand, and the needle in their dominant hand.

Figure 3.6 shows a schematic representation of the VTR system. The three tools are connected to the SCU (A-C) and flash their IREDs in response to commands from the strober controllers. The Optotrak camera array observes the IREDs and reports positions for each marker to the SCU (D). The SCU collects the IRED position information and forwards this to the simulation computer (E), where driver software computes the 3D transformation of each rigid body collection based on IRED position information. Finally, the simulation computer performs the simulation of the tool-image interaction and displays the results on the mock SF OLED (F). Alternatively, the display may instead be routed to an external monitor when simulating CUS.

The VTR software is described in greater detail in Section 3.12, but can be summarized by the flowchart shown in Figure 3.7 (section numbers noted where applicable). In order to provide a convincing illusion of slicing through an actual object, each operational cycle needs to be completed at approximately video frame rate (25 Hz). Measurement of tool position and orientation via the Optotrak can occur substantially faster than this; the limiting factor is the ability of the simulation computer to display slices through the stored 3D image data. As mentioned earlier, Section 3.10 describes the hardware accelerated implementation of 3D slicing that addresses this problem.

3.4 Tool design

3.4.1 Mock Sonic Flashlight

The mock SF is physically quite similar to the clinical version (model 6 [21, 22]) with the exception of the tracking cluster add-on and the lack of a functional ultrasound machine. The organic LED (OLED)

²In Section 3.1 I list the general components of VTR. The list presented here specifically refers to the 2nd generation implementation.

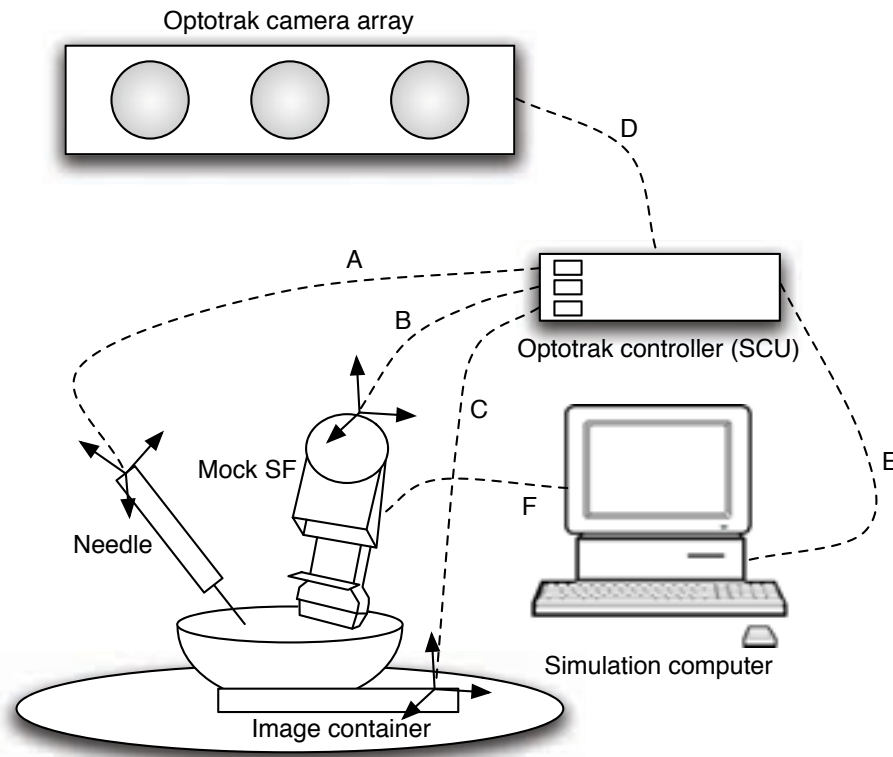


Figure 3.6: Schematic of VTR system.

display, mirror, ultrasound transducer body, and geometry of the mirror-OLED alignment are all identical to the clinical version of the device. The tracking cluster adds a slight amount of weight as compared to the clinical device. In practice, one of the main concerns is minimizing the bulk of the cables exiting from the rear of the SF, and although the tracking cluster does add an additional cable I was able to remove the transducer cable (not needed for the virtual device) and maintain a similar overall cable weight between the real and virtual devices.

The mock SF is constructed from the following components:

1. Plastic housing
 - (a) Left transducer housing (Figure 3.8)
 - (b) Right transducer housing (Figure 3.8)
 - (c) Display shield and tracking cluster base (Figure 3.9)
 - (d) Tracker cluster top shell (Figure 3.10)
2. Dummy ultrasound probe (Figure 3.11)
3. OLED display (Figure 3.11)

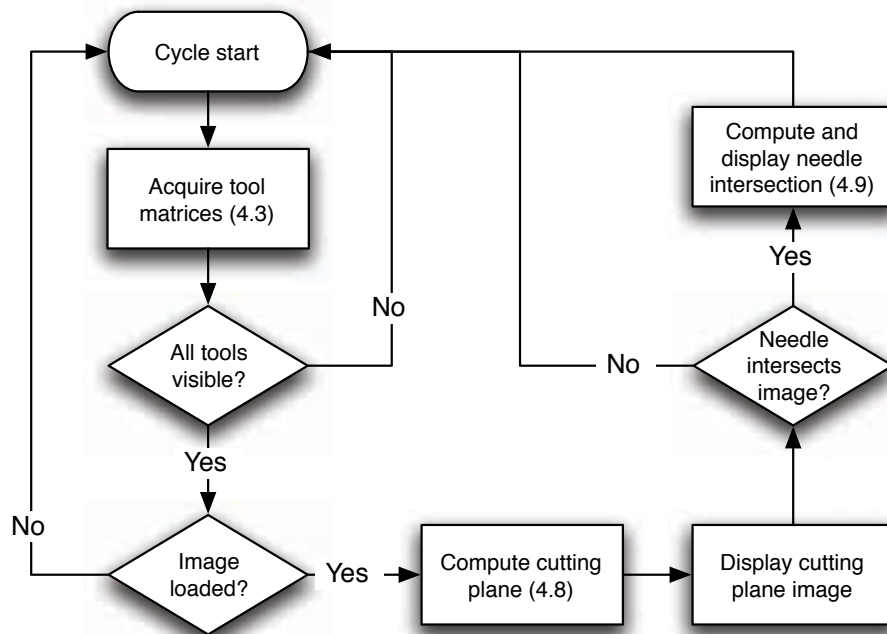


Figure 3.7: A single cycle of VTR execution.

4. OLED display interface board

I designed the plastic housing (Figure 3.8) using Solidworks, a CAD program, and had it manufactured by the Swanson Center (University of Pittsburgh) on a stereolithography machine. To minimize ambient light around the OLED display, the entire housing was cleaned with isopropanol after manufacturing and then painted with flat black paint. Threaded brass inserts were inserted into the left and right transducer housings using epoxy, and the housings were assembled around the dummy probe and OLED display using small screws. Although it should not be necessary to disassemble the mock SF during normal operation, it is easy to do so should the need arise.

After the left and right transducer housings were assembled around the dummy probe, the display shield (Figure 3.9) was attached using small machine screws. Display shield removal, though still uncommon, occurs more frequently than disassembly of the transducer housings, due to the periodic need to clean the OLED display.

The tracking cluster is a hemispherical shell (Figure 3.10) that serves as a mounting location for 24 infrared LED (IRED) Optotrak markers. The hemispherical design helps to maximize marker visibility when the mock SF is rotated relative to the tracking cameras. A thorough analysis of tracking cluster design can be found in the work of Hamza-Lup et al [23, 24]. Unlike the transducer housing, the tracking cluster is permanently assembled. During assembly, the IRED markers were glued to the exterior of the shell with their anode and cathode leads extending through small holes to the underside of the shell. The cathode leads were soldered together, and individual wires attached to each anode lead. The IRED wiring harness utilizes a rubber grommet for strain relief and exits the bottom rear of the tracking cluster

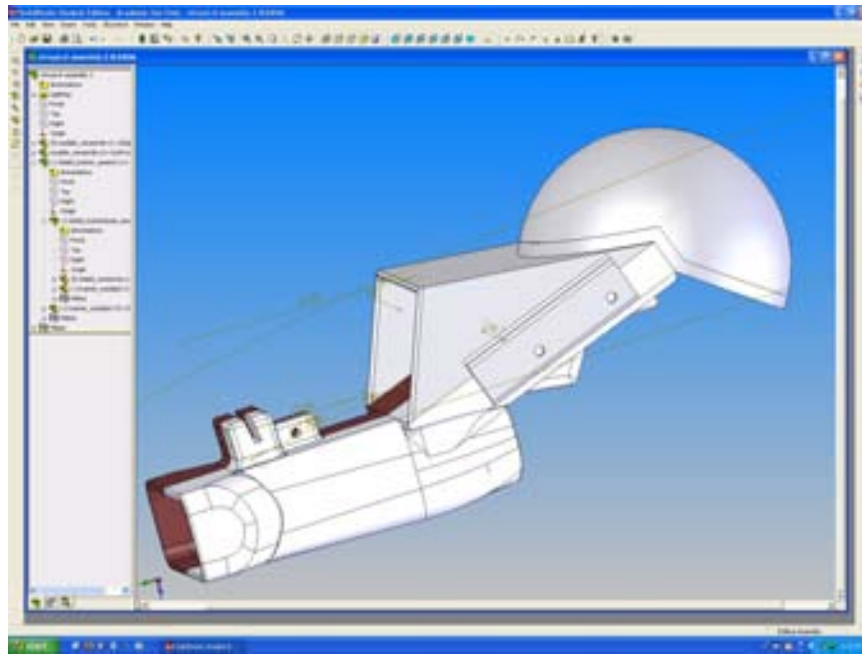


Figure 3.8: CAD schematic of mock Sonic Flashlight casing.

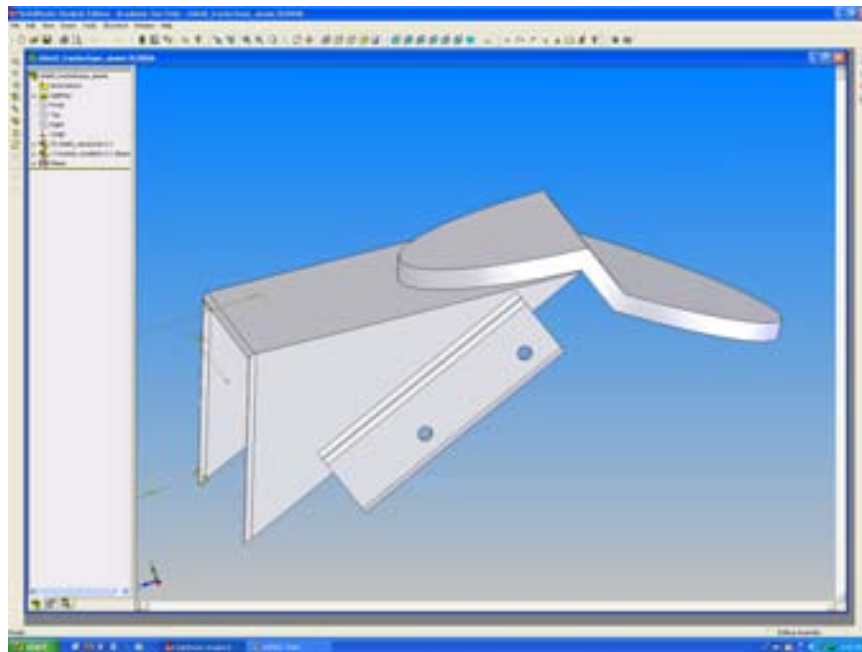


Figure 3.9: Removable display shield.

as a single cable (Figure 3.10).



Figure 3.10: Mock SF tracking cluster wiring and strain relief.

3.4.2 Tracked needle tool

The needle tool was purchased from Northern Digital and includes a removable blunt tip that is useful for surface digitization. In order to use the same tool for modeling medical procedures that employ a needle, I designed a removable fixture that holds 14 gauge needles of varying lengths (Figure 3.12). Due to the machining tolerances of the various tips it is possible to switch between them (and load an appropriate configuration file) without having to re-calibrate the tool.

3.4.3 Tracked image container

The purpose of the tracked image container (Figure 3.13) is to provide a physical surface that appears to “contain” the simulated targets. I use a relatively simple design consisting of plastic bowl rigidly attached to a wooden board. Six IRED markers define the board as a rigid body and allow tracking with the Optotrak. A fixed transform from the rigid body frame of the board to the surface of the container is applied to the raw tracking matrix so that the net result is to track the surface of the image container.



Figure 3.11: Mock SF showing operation of the OLED.



Figure 3.12: Components of the tracked needle; IRED cluster (top), removable needle holder (middle), additional needle (bottom).

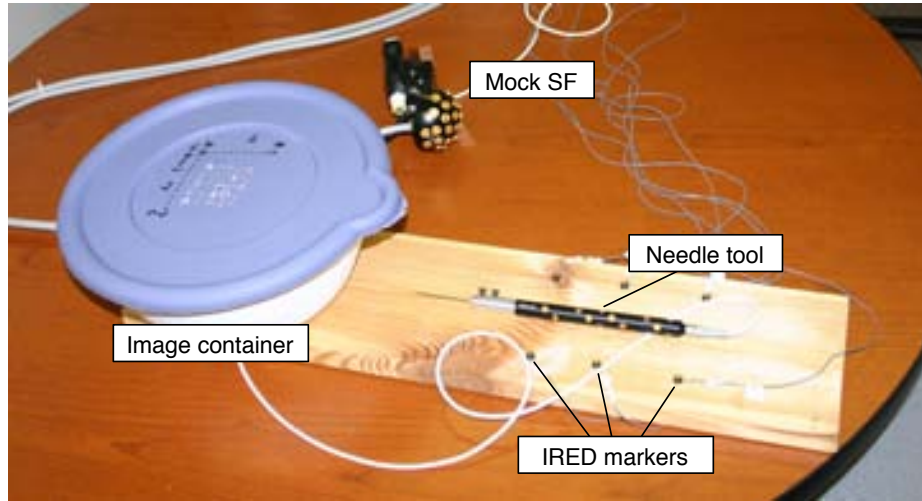


Figure 3.13: The tracked image container, with the mock SF and needle tool for scale reference.

3.5 Optotrak system usage

3.5.1 Physical system arrangement

I was fortunate to obtain a dedicated space in which to install the Optotrak equipment (Figure 3.14). Although the Optotrak system can utilize multiple cameras to either enlarge the workspace or improve marker visibility within a smaller workspace, I was constrained by the near focus distance of the camera lenses and was only able to use a single camera bar.

3.5.2 Tracking rigid bodies

The tracking system reports a 4x4 homogeneous transformation matrix for each rigid body defined by the Optotrak system configuration. A 4x4 homogeneous transformation matrix \mathbf{T} can be decomposed into a 3x3 rotation component \mathbf{R} and a 3x1 column vector \mathbf{t} representing translation.

$$\mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix} \quad (3.4)$$

It is useful to think of this matrix as representing a transform from a global reference frame G , rigidly fixed to one the Optotrak camera bars, to a reference frame B affixed to the rigid body in question. This transformation is defined as $\mathbf{T}_{G \rightarrow B}$

3.5.3 Using remote frames of reference

The global frame G is the *Optotrak* reference frame, defined internally by the tracking system. The Optotrak cameras are placed in locations that maximize visibility of the workspace and the native co-

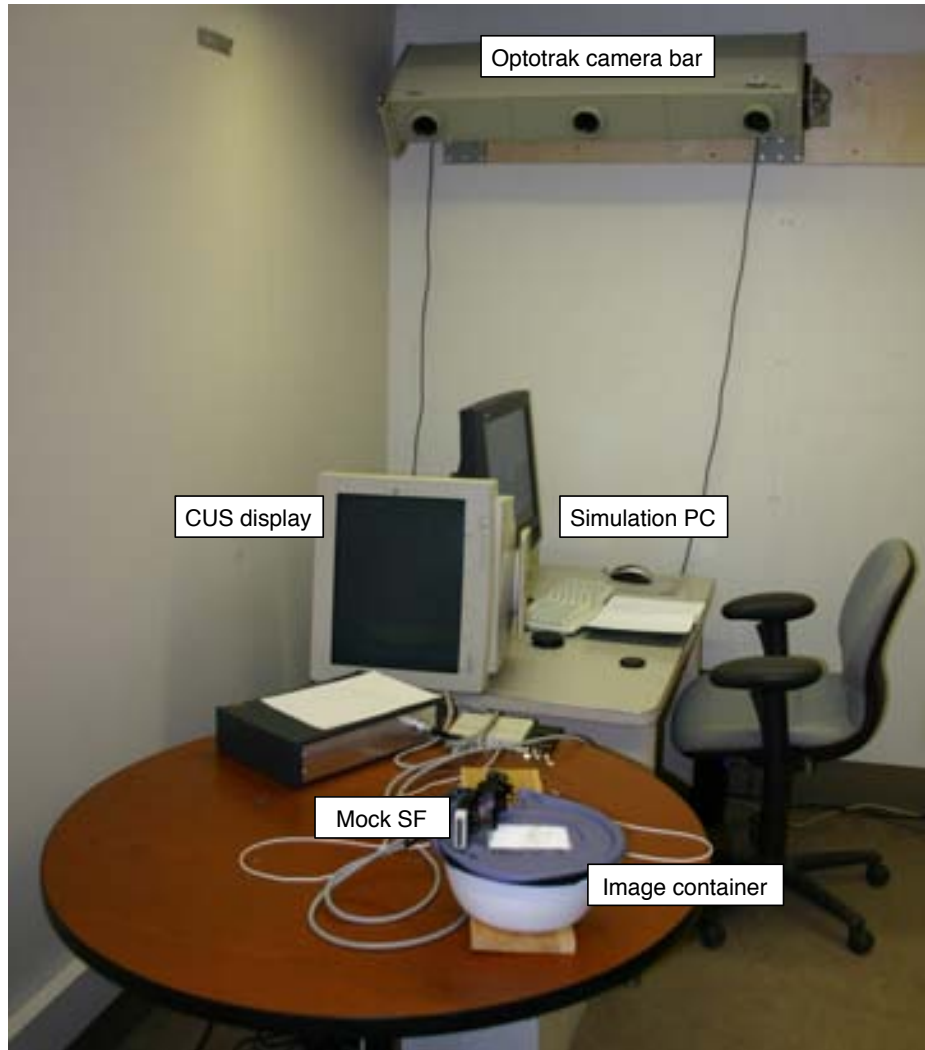


Figure 3.14: Configuration of the room in which the VTR system is installed.

ordinate system has little intuitive meaning to the user. The origin of the Optotrak coordinate system is inside the body of one of the camera mounts, and the entire coordinate system is tilted relative to gravity.

It is simple to define any arbitrary coordinate system R relative to G coordinates by simply picking three points in G using the needle tool. If the 3 points in G are denoted \mathbf{a} , \mathbf{b} , and \mathbf{c} , three unit vectors describing the orientation of the plane containing these points can be found by:

$$\mathbf{x} = |\mathbf{c} - \mathbf{a}| \quad (3.5)$$

$$\mathbf{y} = |\mathbf{c} - \mathbf{a}| \times |\mathbf{b} - \mathbf{a}| \quad (3.6)$$

$$\mathbf{z} = \mathbf{x} \times \mathbf{y} \quad (3.7)$$

The full homogeneous matrix $\mathbf{T}_{G \rightarrow R}$ describing the transform from G to the newly defined planar coordinate system R can be composed as:

$$\mathbf{T}_{G \rightarrow R} = \begin{bmatrix} \mathbf{x}_x & \mathbf{y}_x & \mathbf{z}_x & \mathbf{a}_x \\ \mathbf{x}_y & \mathbf{y}_y & \mathbf{z}_y & \mathbf{a}_y \\ \mathbf{x}_z & \mathbf{y}_z & \mathbf{z}_z & \mathbf{a}_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.8)$$

As mentioned above, the G coordinate system is non-intuitive and it is convenient to move G to a different (arbitrary but fixed) location in space. Assuming that a rigid table top is present in the workspace, a point on this surface is an ideal “global” frame of reference. The VTR system supports this feature by defining a remote frame R as described previously, and then premultiplying any Optotrak tracking matrices by $\mathbf{T}_{G \rightarrow R}^{-1}$ before using them in further calculations. Figure 3.15 and Equation 3.9 demonstrate how this is implemented.

$$\mathbf{T}_{R \rightarrow F} = (\mathbf{T}_{G \rightarrow R})^{-1} \cdot \mathbf{T}_{G \rightarrow F} \quad (3.9)$$

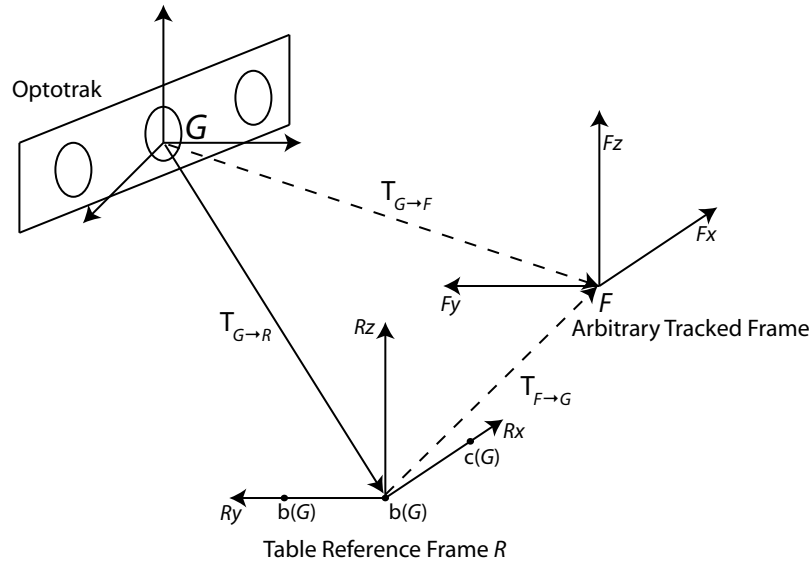


Figure 3.15: Defining and using a remote reference frame.

Both the VUE software and this dissertation use G to refer to the global reference frame, even if the user chooses to move the global frame to a location other than the Optotrak camera bar. This is simple because $\mathbf{T}_{G \rightarrow R}^{-1}$ is a constant and the only effect of this operation is to move G to a more intuitive location.

3.5.4 Marker visibility concerns

In order to compute the location of a rigid body the Optotrak cameras must have line of sight to at least 4 markers. As a user manipulates a rigid body, the change in orientation of its markers relative to the cameras will cause markers to move in and out of view. If too few markers from any of the rigid bodies are visible, causing the Optotrak to be unable to solve for the rigid body transformation, the simulation software will pause the simulation and sound a warning tone to alert the user that tracking has been interrupted. Most often, this problem is due to occlusion of the markers on the needle tool by the user's hand, or by inadvertent rotation of the needle so that it is parallel to the axis of the camera bar (making the IRED markers invisible).

3.6 Frames of reference

1. F - The rigid body frame of the board that the image container is mounted to. As will be described in Section 3.8, this frame is never explicitly used in the simulation, and is instead used to derive the location of the image container frame I .
2. G - The global coordinate system internal to the Optotrak cameras. This coordinate system has an arbitrary but fixed location, typically chosen to provide m. Any rigid body tracked within the virtual environment initially reports its coordinates in this frame of reference.
3. I - The reference frame of the image container, used to place a 3D image inside the container. The orientation of the image container ensures that axis I_z is approximately collinear with gravity.
4. P - The reference frame of the needle tool.
5. S - The coordinate system defined by the marker cluster rigidly attached to the rear of the mock Sonic Flashlight; defined during calibration of the tracking cluster using tools provided with the Optotrak. The placement of this coordinate system has no physical meaning with respect to the design of the mock Sonic Flashlight.
6. V - The coordinate system defined by the reflection of the flat panel monitor in the mirror mounted on the mock Sonic Flashlight (the virtual image frame). Alternatively, this may be thought of as the coordinate system that defines "where to slice", since it would - in the actual Sonic Flashlight - correspond to the particular slice being acquired by the ultrasound transducer. The x-y plane of V is in the plane of the reflection of the display, aligned with the x-y coordinate system defined by the placement of pixels on the display. The z axis of V is normal to the reflection of the display.
7. V^* = The coordinate system defined by the flat panel monitor. There is a fixed transformation between V^* and V , determined by the geometry of the device and dependent on the design requirements of the actual Sonic Flashlight Model 6 on which the virtual ultrasound system is based. However, since the entire mock Sonic Flashlight is rigid there is no need to solve for this relationship explicitly and calibration can proceed purely by considering V .

3.7 Calibration of the VTR display

In RTTR a combination of geometric constraints and a planar calibration technique [25] insure that the reflected virtual image and ultrasound slice are coincident. Proper calibration of an RTTR system guarantees that each pixel of rendered data exactly corresponds with the point in physical space from which the ultrasound data was acquired. Owing to the absence of an actual ultrasound machine, VTR can be thought of as the process of slicing a simulated dataset with the virtual image (i.e. reflection of the planar display) itself. Much as in RTTR, the reflected pixels in VTR can be thought of as occupying real physical space within the virtual dataset. In other words, the pixels are both the *representation* and *source* of the data.

The goal of the calibration is to take 2D screen coordinates used to display pixels on the flat-panel monitor and translate them into 3D world coordinates shared by the computer-generated target and tools simulating invasive procedures. Alternatively, one may think of the calibration procedure as determining a transformation matrix which, when multiplied by that of a tracking device rigidly fixed to the mock Sonic Flashlight, results in the coordinate system of the virtual image in global space.

Figure 3.16 shows the relationships between the various coordinate systems and known data involved in the VTR display calibration process.

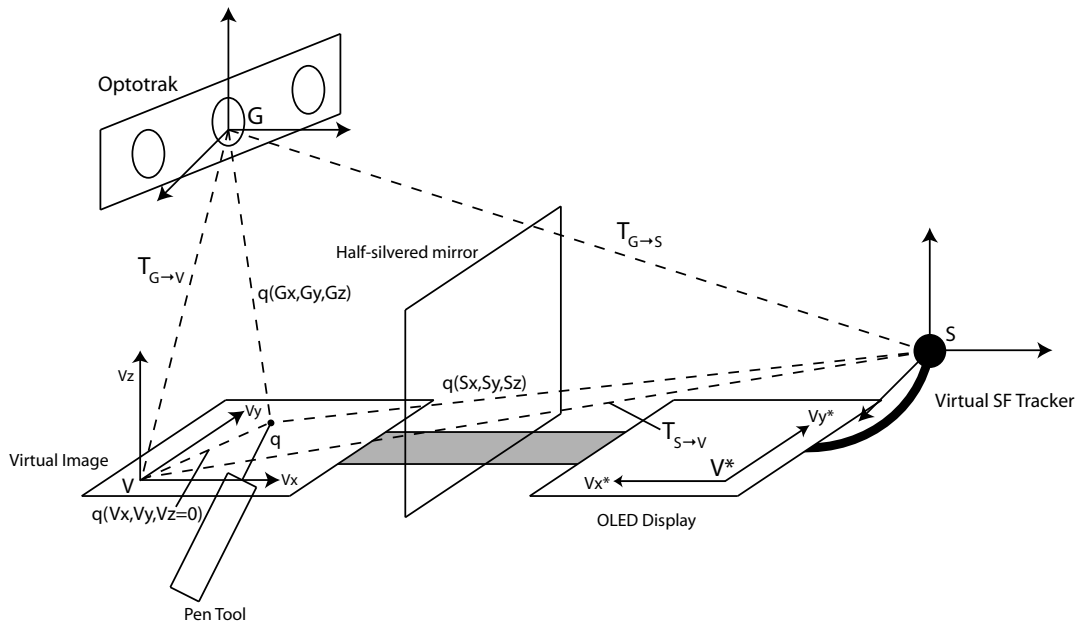


Figure 3.16: Coordinate systems used in VTR display calibration.

3.7.1 Known coordinate system transforms & data

1. The transform of the mock Sonic Flashlight tracker in global coordinates, $\mathbf{T}_{G \rightarrow S}$.
2. The position of a point \mathbf{q} in global coordinates, $\mathbf{q}(G_x, G_y, G_z)$. \mathbf{q} is the tip of a needle/stylus that has previously been calibrated for use with the Optotrak.
3. Since $\mathbf{T}_{G \rightarrow S}$ is known, it is straightforward to compute the position of point \mathbf{q} in S coordinates, $\mathbf{q}(S_x, S_y, S_z)$.
4. The position of \mathbf{q} in V , $\mathbf{q}(V_x, V_y, V_z = 0)$, as described in the following section.

3.7.2 Display calibration procedure

The goal of the display calibration is to determine $\mathbf{T}_{S \rightarrow V}$, the position and orientation of the virtual image coordinate system V relative to the sonic flashlight coordinate system S . During operation of the virtual ultrasound system, this transform will be applied to $\mathbf{T}_{G \rightarrow S}$ in order to yield the transformation of the virtual image in global coordinates, $\mathbf{T}_{G \rightarrow V}$ (not shown in Figure 3.16 for purposes of clarity).

Calibration occurs by establishing correspondences between two sets of points $S(q_{i=1...n})$ and $V(q_{i=1...n})$ and recovering the optimal similarity transform which aligns corresponding points. Corresponding point pairs are determined by displaying a marker in the virtual image and visually aligning the tip of the pen tool with the virtual marker (see Figure 3.17). The initial selection of coordinate frame V is essentially arbitrary, although the most logical choice is to work in the pixel-based coordinate system native to the display. Starting with V in pixel coordinates, I recover an isotropic scale factor c which is the number of pixels per world coordinate unit (mm), the inverse of which is the pitch of the pixels on the display. Since pixel pitch is a known parameter of the OLED display, this is a useful check that the calibration has been successful.

Although the calibration requires only 4 point pairs, better results are obtained by sampling a wide variety of points over the range of possible positions in the virtual image. Several techniques have been proposed for solving point correspondence problems of this type [26, 27, 28]. I chose to implement a refined version of the algorithm developed by Umeyama [29]. *Note that for ease of comparison to Umeyama's algorithm, I have preserved the original notation from that paper. Mathematical notation used in equations 3.10 through 3.19 should be assumed to apply locally to this section unless otherwise noted in the text.*

Let $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ and $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n\}$ be $n \times m$ matrices, where m is the dimensionality of the coordinate systems involved (in this case 3), and n is the number of point pairs acquired, with $\mathbf{x}_i = S(\mathbf{q}_i)$ and $\mathbf{y}_i = V(\mathbf{q}_i)$.

$$\mu_x = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i \quad (3.10)$$

$$\mu_y = \frac{1}{n} \sum_{i=1}^n \mathbf{y}_i \quad (3.11)$$

$$\sigma_x^2 = \frac{1}{n} \sum_{i=1}^n \|\mathbf{x}_i - \mu_x\|^2 \quad (3.12)$$

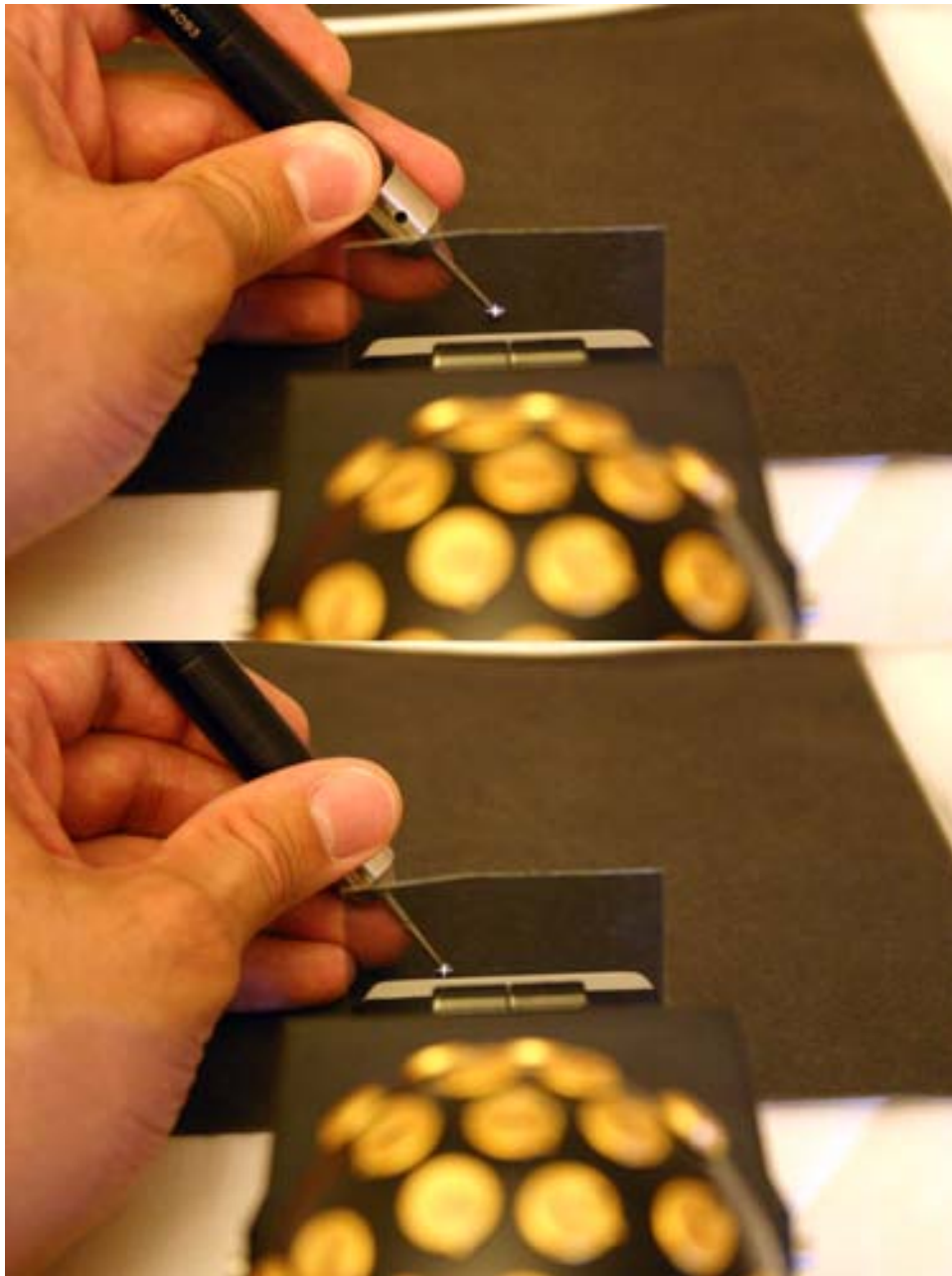


Figure 3.17: Aligning the pen tool in S coordinates with two calibration targets in V coordinates. The white dot is reflected in the half-silvered mirror from the OLED display hidden beneath the glare-shield.

$$\sigma_y^2 = \frac{1}{n} \sum_{i=1}^n \|\mathbf{y}_i - \mu_y\|^2 \quad (3.13)$$

$$\Sigma_{xy} = \frac{1}{n} \sum_{i=1}^n (\mathbf{y}_i - \mu_y)(\mathbf{x}_i - \mu_x)^T \quad (3.14)$$

Let \mathbf{UDV}^T be a singular value decomposition of Σ_{xy} . The optimal similarity transform which aligns the two point clouds can be decomposed into a rotation matrix \mathbf{R} , scale factor c , and translation vector \mathbf{t} . Note that in Equations 3.15 through 3.18, \mathbf{S} refers to a matrix used in the standard naming convention for SVD decompositions, not to the Sonic Flashlight tracking frame of reference S . If $\text{rank}(\Sigma_{xy}) \geq m-1$ (true in our case), then:

$$\mathbf{R} = \mathbf{USV}^T \quad (3.15)$$

$$\mathbf{t} = \mu_y - c\mathbf{R}\mu_x \quad (3.16)$$

$$c = \frac{1}{\sigma_x^2} \text{tr}(\mathbf{DS}) \quad (3.17)$$

where S is chosen as:

$$\mathbf{S} = \begin{cases} \mathbf{I} & \text{if } \det(\mathbf{U}) \det(\mathbf{V}) = 1 \\ \text{diag}(1, 1, \dots, 1, -1) & \text{if } \det(\mathbf{U}) \det(\mathbf{V}) = -1 \end{cases} \quad (3.18)$$

Although I could apply the entire similarity transform at once, it is more intuitive to perform an initial calibration trial to recover c and then apply the scale factor in software when creating the display coordinate frame V . Applying c separately from the rest of the transform allows enables the system to draw graphical objects in more natural units of physical rather than pixel distance. The desired transformation $T_{S \rightarrow V}$ results from composing a 4x4 homogeneous transformation matrix:

$$\mathbf{T}_{S \rightarrow V} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix} \quad (3.19)$$

3.7.3 Results of display calibration

I performed two successive calibrations of the VTR device, the first to recover the scale c and the second to verify repeatability of the calibration after accounting for c when defining the V coordinate frame.

Initial calibration, assuming V in pixel coordinates and using 16 point pairs, yielded a scale c of 18.3256 pixels/mm. Inverting this to yield pixel pitch I obtain a pitch of 0.05457 mm/pixel. The actual display pixel pitch, obtained from the Kodak OLED display specs, is 0.05478 mm/pixel, for a pixel pitch error of approximately 0.39%. If the entire display is considered and I multiply the isotropic pixel pitch error by the length of each display dimension in pixels, we find an overall error magnitude of 0.11 mm for the x display length and 0.18 mm for the y display length.

A second calibration was performed after applying c to the V coordinate system as described previously, resulting in $c = 1.00647$ (this indicates that the second calibration differed from the first by less than 1%). Overall percentage of error for the display size in the second calibration is nearly identical to the first calibration, showing good repeatability of results. In addition, calibration results demonstrate good repeatability when multiple users perform the same calibration procedure.

Quantitative verification of the rotation and translation components of the calibration is somewhat more difficult than deriving known display parameters, since we lack a gold standard such as known pixel pitch. However, a straightforward visual verification method is to compute the intersection of the spherical pen tip in G coordinates with the computed image plane location and then display this intersection point in the virtual image in V coordinates. Correct calibration of R and t can be confirmed if the displayed intersection point and the physical pen tip coincide when viewed through the flat panel monitor.

For purposes of comparison, the Optotrak static positioning accuracy, as reported by Northern Digital, is 0.1 mm. Although I might expect to exceed this accuracy by virtue of a large number of point pairs (and the resulting increase in signal-to-noise) during calibration, the overall interaction of the various tracked coordinate systems is non-trivial. This means that the overall tracking error present in the VTR system, considered as a whole rather than on a per rigid body basis, will vary depending on the orientation of each component relative to the trackers.

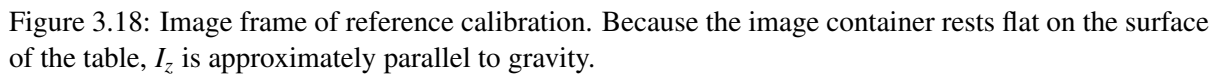
3.8 Calibrating the image frame of reference

The image frame of reference is calibrated in a manner virtually identical to that used to track any arbitrary reference frame (see Section 3.5.3). Three points on the top surface of the image container – **a**, **b**, and **c** – are sampled in G and used to define frame I relative to F (since $T_{G \rightarrow F}$ is known). This yields the transform $T_{F \rightarrow I}$, which is premultiplied by the image frame rigid body tracking vector $T_{G \rightarrow F}$ to obtain the position of the image frame in global coordinates, $T_{G \rightarrow I}$.

3.9 Calibrating the needle tool

Calibrating the needle tool is relatively straightforward. As purchased from NDI, the needle tool arrives with the origin of its rigid body frame placed at the tip of the blunt stylus. After replacing the blunt tip with the 14 gauge needle holder described earlier, a calibration routine included with the NDI software can be used to generate a similar calibration file with the rigid body frame P centered at a point denoted \mathbf{p}_1 . Using the NDI algorithm P_z is approximately, though not exactly, the long axis of the needle. More precisely, P_z extends along the mean axis of the IREDs located on the exterior of the needle tool. It should be noted that the NDI routine is designed with the assumption that the goal of calibration is to establish the location of the blunt stylus tip relative to the IREDs for the intended purpose of using the tip as a 3D digitizer. Under this assumption, the coordinate system axes may be oriented arbitrarily as long as the origin \mathbf{p}_1 is located accurately. When used with the VTR system, knowing the needle tip location is important, but it is equally important to know the orientation of the needle shaft.

Unfortunately, the needle holder introduces a slight displacement of the needle relative to the centerline of the body of the tool. Though small, this displacement - which results in an angular error of P_z relative to the true axis of the needle - is magnified when the intersection of the needle and virtual



This error can be eliminated by repeating the NDI calibration routine with needles of two different lengths. The origins of the two resulting coordinate systems are points \mathbf{p}_1 and \mathbf{p}_2 . It is now possible to use the original calibration frame P with the additional knowledge that the point $\mathbf{p}_1 + (\mathbf{p}_2 - \mathbf{p}_1)$ lies exactly on the centerline of the needle. The other two axes of P , P_x and P_y are perpendicular to the rotational axis of symmetry P_z and do not affect any future computations.

3.10 Computing the cutting plane

³Wildcat Realizm 200, 3Dlabs

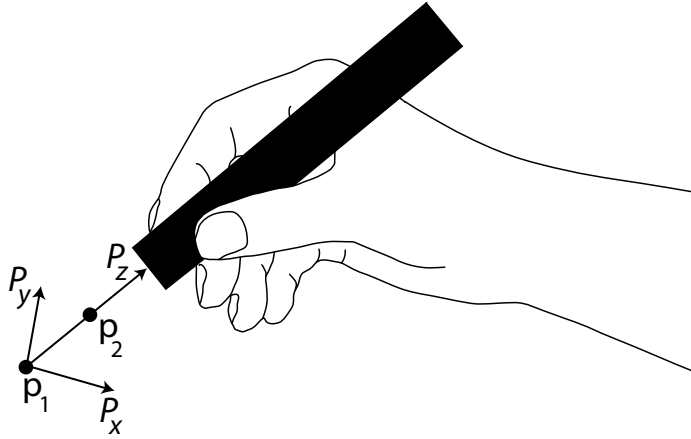


Figure 3.19: Needle calibration.

the correct cutting plane in software, the medical image dataset is converted to a 3D OpenGL texture and stored in the video card memory, allowing for very fast update rates. The video card I use provides multiple video outputs, so I use a multiple display setup where the test control interface is presented on a flat panel monitor and the simulated slice image is sent to the OLED interface board for display on the mock Sonic Flashlight.

The VTR system models the tomographic slice as a simple quadrilateral, with a thickness of 0, and the other two dimensions the same size as the reflection of the OLED display. The previous section describes how I recover $T_{S \rightarrow V}$, the transform from the mock Sonic Flashlight to the virtual image coordinate frame. The results of this calibration, specifically the scale factor c , are used to determine the position of 4 points in V coordinates that can be used to establish the corners of the quadrilateral slice.

The maximum extent of the display in physical coordinates (mm) can be computed as:

$$v_xmax = xpixelsize \times c \quad (3.20)$$

$$v_ymax = ypixelsize \times c \quad (3.21)$$

Similarly, the coordinates in P of the points defining the virtual image quadrilateral are (where **ll** denotes lower left, **ur** denotes upper right, etc.):

$$\mathbf{ll} = (v_xmin = 0, v_ymax = 0, 0) \quad (3.22)$$

$$\mathbf{ul} = (v_xmin = 0, v_ymax, 0) \quad (3.23)$$

$$\mathbf{ur} = (v_xmax, v_ymax, 0) \quad (3.24)$$

$$\mathbf{lr} = (v_xmax, v_ymin = 0, 0) \quad (3.25)$$

An orthographic projection of the virtual image quadrilateral, omitting the z -axis of V (not used, since the virtual image has no thickness), is shown in Figure 3.20.

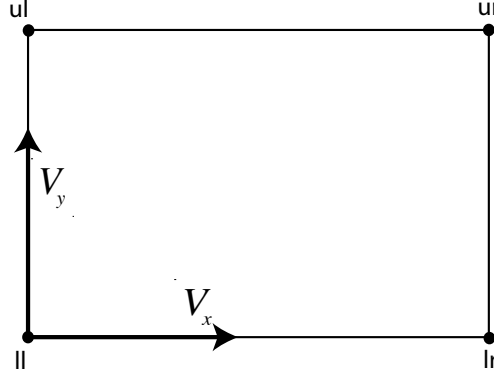


Figure 3.20: Virtual image coordinate system.

During operation, the VTR system allows the user to display arbitrary slices through a stored image data set. This 3D data set occupies a certain volume of physical space and has an associated frame of reference I . I is determined by tracking a physical bounding box structure (e.g. a blank gel phantom, or cardboard box), though the only assumption made regarding I is that it is tracked in global coordinates – through the intermediate frame F associated with the image container trackers – and therefore $\mathbf{T}_{G \rightarrow I}$ is known⁴. As shown in Figure 3.21, the points that define the virtual image quadrilateral have coordinates in both V and I .

The V coordinates are fixed, so it is straightforward to compute the quadrilateral coordinates in I , using the point \mathbf{ur} as an example, as follows:

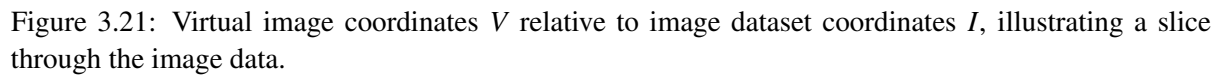
$$I(\mathbf{ur}) = V(\mathbf{ur}) \cdot (\mathbf{T}_{I \rightarrow V})^{-1} \quad (3.26)$$

More formally, since I typically do not know $T_{P \leftarrow I}$ explicitly and know V only by virtue of the calibration described in Section 3.7, the full expression for the example point $I(\mathbf{ur})$ can be written:

$$I(\mathbf{ur}) = V(\mathbf{ur}) \cdot (\mathbf{T}_{S \rightarrow V})^{-1} \cdot (\mathbf{T}_{G \rightarrow S})^{-1} \cdot \mathbf{T}_{G \rightarrow I} \quad (3.27)$$

Repeating this computation for the other 3 points in the quadrilateral yields a complete definition of the virtual image plane in the I reference frame. The final stage of the cutting plane extraction is to compute the OpenGL texture coordinate of each of the 4 quadrilateral points; knowing the texture coordinates allows efficient hardware accelerated computation of the cutting plane image. The I reference frame has an analogous reference frame I_{tex} that defines the underlying data space coordinates of

⁴Equivalently, per Section 3.5.3, it is possible to transform G to an arbitrary remote frame R .



The cutting slice is displayed on the OLED or external CRT monitor (when in CUS mode) by creating an orthographic projection of a quadrilateral polygon with the coordinates previously defined in V . The OpenGL viewport of the display is sized so that this polygon exactly covers the available pixels. Each corner of the quadrilateral has an associated texture coordinate I_{tex} .

3.11 Computing the needle-plane intersection

Computing the needle-plane intersection (Figure 3.22) can be simplified by modeling the needle

as a vector and the virtual image as an infinitely thin plane. Let \mathbf{p}_1 and \mathbf{p}_2 be the far and near needle calibration points from Section 3.9. Point \mathbf{p}_3 is the origin of the V reference frame, and \mathbf{n} is the vector normal to the virtual image plane (by definition, V_z). Since \mathbf{n} is simply the z -component of the V frame and \mathbf{p}_3 its origin, I find it easiest to compute the needle-plane intersection in V coordinates. This requires \mathbf{p}_1 and \mathbf{p}_2 to be expressed as $V(\mathbf{p}_1)$ and $V(\mathbf{p}_2)$, a straightforward operation since both V and P are known in global coordinates. All of the calculations below take place after the relevant points and vectors have been transformed to V coordinates.

Point \mathbf{p}_4 , the intersection between the needle and plane, can be computed as follows [31]:

$$u = \frac{\mathbf{n} \cdot (\mathbf{p}_3 - \mathbf{p}_1)}{\mathbf{n} \cdot (\mathbf{p}_2 - \mathbf{p}_1)} \quad (3.28)$$

$$\mathbf{p}_4 = \mathbf{p}_1 + u \cdot (\mathbf{p}_2 - \mathbf{p}_1) \quad (3.29)$$

Note that if the denominator in equation 3.28 is zero, the needle axis and V_{xy} are either parallel (no solution) or coincident (infinite solutions), and the VTR software aborts the calculation without rendering the intersection point. Additionally, I am only interested in solutions where the intersection point lies on the needle (as opposed to the forward or backward projection of the needle axis), so the solution for the value of \mathbf{p}_4 is discarded if $u > 1$ or $u < 0$.

As mentioned earlier, this computations yields \mathbf{p}_4 in V coordinates. The intersection point is easily displayed by drawing a point in OpenGL that overlays the slice rendering from Section 3.10. Since both the cutting plane and \mathbf{p}_4 are expressed in V coordinates, no additional coordinate transformations are needed when rendering the slice intersection point.

An alternative rendering technique I use for certain experiments described in later chapters is to check for the existence of a valid needle-slice intersection, which occurs when $0 \leq u \leq 1$, and then display a solid red bar along the near (transducer) edge of the virtual image. This provides an indication to the user that the needle has touched or penetrated the virtual image, but does not provide feedback on where in the image the intersection has occurred. The reason for this alternative feedback is discussed further in Section 4.2.

3.12 VTR software

The VTR system uses several discrete pieces of software throughout the calibration, data collection / simulation, and data analysis stages of operation. These 3 applications were developed using different tools, to address the specific needs of each task. The most complicated of these applications, the data collection / simulation program, is called the Virtual Ultrasound Environment (VUE).

3.12.1 Calibration

Calibration tasks are split between VUE (described in the next section) and a *Mathematica* helper program that performs the calibration of the virtual image display. Calibration of the needle, image container, and remote frames of reference occur entirely within VUE, while calibration of the virtual image requires data collection within VUE and subsequent post-processing with the *Mathematica* program. The virtual image calibration code was written separately from VUE for several reasons including ease

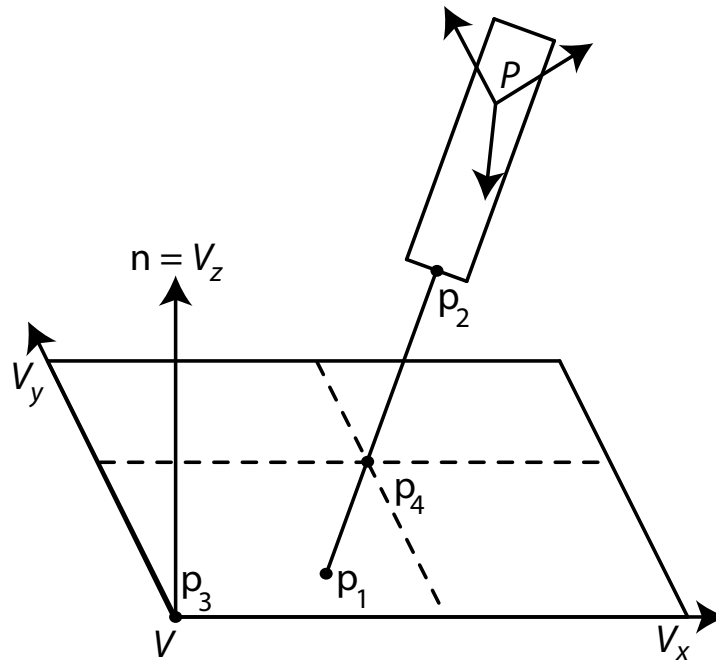


Figure 3.22: Needle intersection computation.

of debugging, integration with a calibration simulation program also developed in *Mathematica*, and the ability of the *Mathematica* language to facilitate a more-or-less direct translation of the calibration routine from Section 3.7 to software.

3.12.2 Data collection and simulation with VUE

VUE was developed in C++ using several toolkits for data acquisition and graphics rendering (see Appendix E), and is responsible for controlling the Optotrak unit, performing the slicing simulation, and providing a user interface to the experimenter that allows them to control the subject testing process. VUE has been developed as an extensible research platform and is highly object-oriented, making it easy to add new features with minimal development time.

Among the main capabilities of VUE are:

- ★ Rendering of 3D images from either stored image data or images computed on-the-fly from shape primitives.
- ★ Control of both the size and resolution of images generated on the fly.
- ★ Support for 3D images of arbitrary size, limited only by system and video card memory.
- ★ Support for both color and black and white images.

- ★ Rendering of two forms of image noise. The first is integrated into the 3D image itself, and the second functions as an overlay that draws an overlay of points onto the image slice. The former is used during the training portion of the experiments described later in my thesis, and the latter is a parameter controlled during experimental trials.
- ★ Multiple monitor support, allowing the image slice to be displayed on a separate monitor or on the mock SF.
- ★ Data collection in both streaming mode, where data is written to disk as fast as possible, and single point mode, where a data frame is stored in response to a mouse click.

The primary user interface components are:

- ★ The main window (Figure 3.23) – Presents an overview 3D rendering of the simulation environment and contains menus that control calibration and overall system function. A status bar along the bottom of the 3D rendering provides feedback on the current operation of the system (though in Figure 3.23 it incorrectly reports the absence of an image).
- ★ The image generation window (Figure 3.24) – Provides controls for managing the properties of internally generated images. Also supported are pre-defined *test sets* that define a sequence of image parameters used to generate images; this capability is used during subject testing to present the same sequence of images to each subject.
- ★ The data collection window (Figure 3.25) – Allows selected experimental results to be stored in a user defined file.

3.12.3 Data analysis

Data analysis is performed in several stages, the first of which is to post-process the data acquired by VUE using several custom Matlab programs. Matlab programs are easier to modify than C++ and can be easily tweaked depending on the goals of a particular experiment. A more detailed explanation of the data analysis process appears in the next chapter.

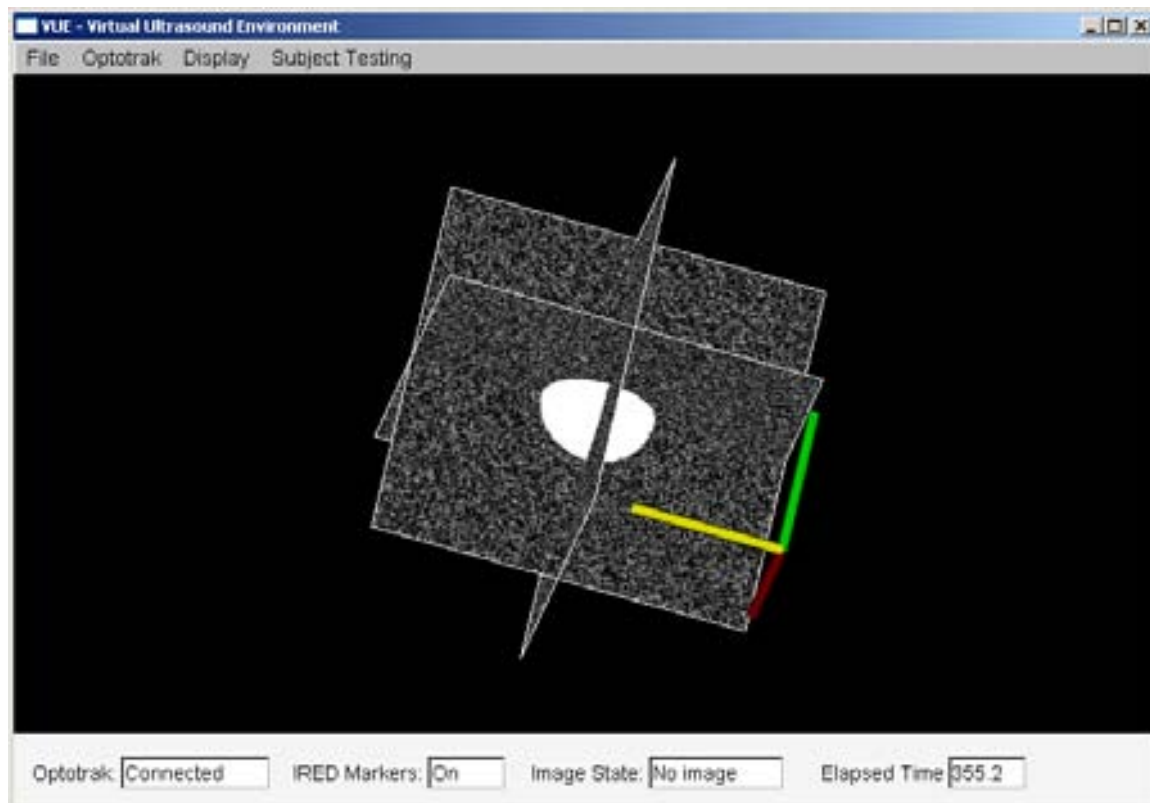


Figure 3.23: Screen capture of VUE, main window.

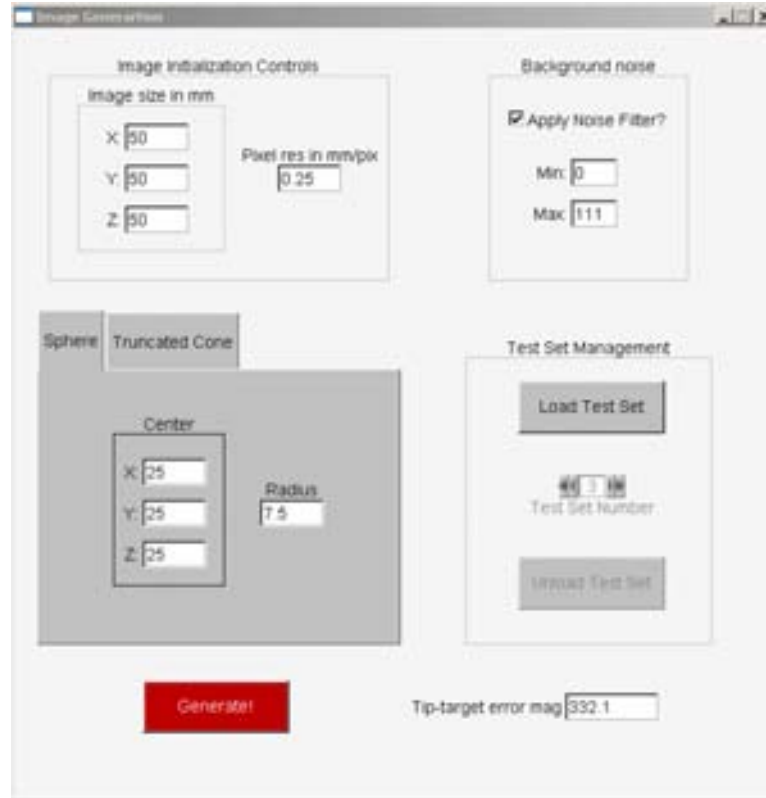


Figure 3.24: Screen capture of VUE, image generation window.

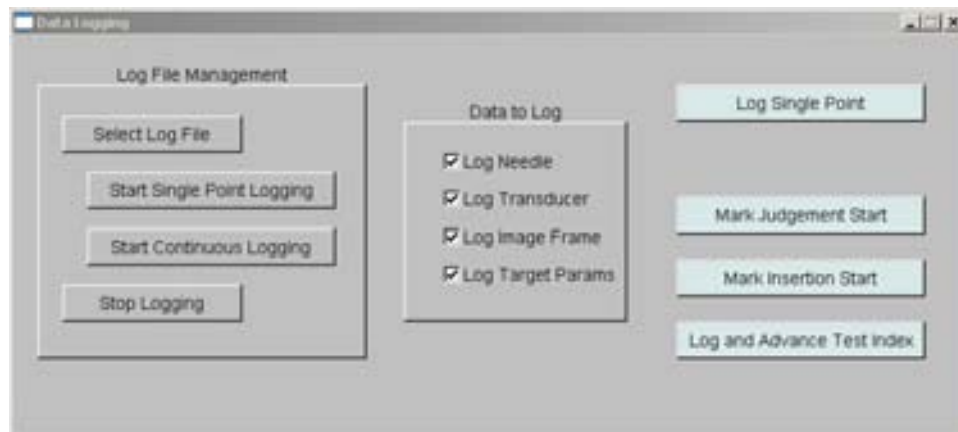


Figure 3.25: Screen capture of VUE, data collection window.

3.13 Photographs of the VTR system in operation



Figure 3.26: A subject using the VTR system in SF mode to view a target.

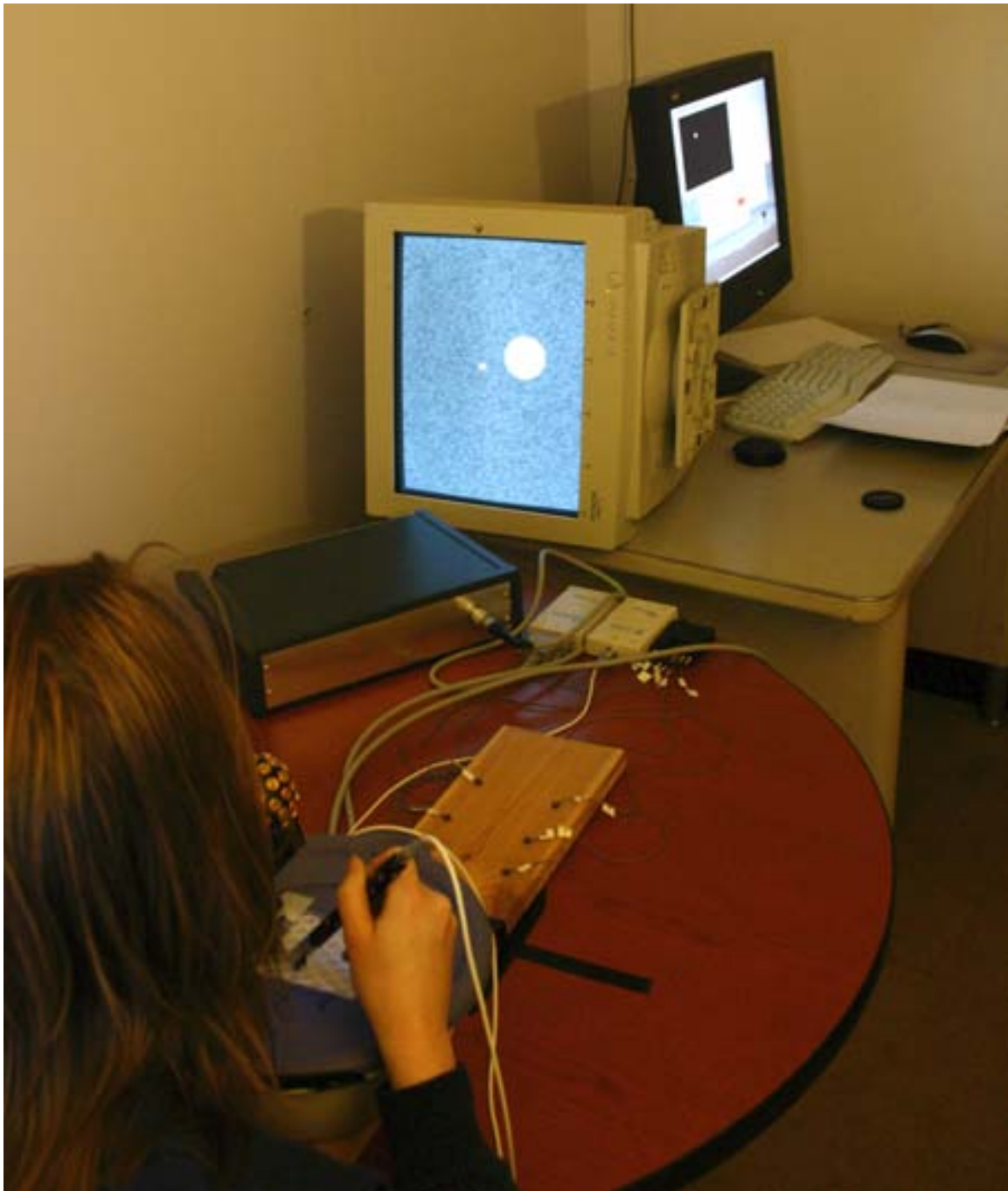


Figure 3.27: A subject using the VTR system in CUS mode to view a target.

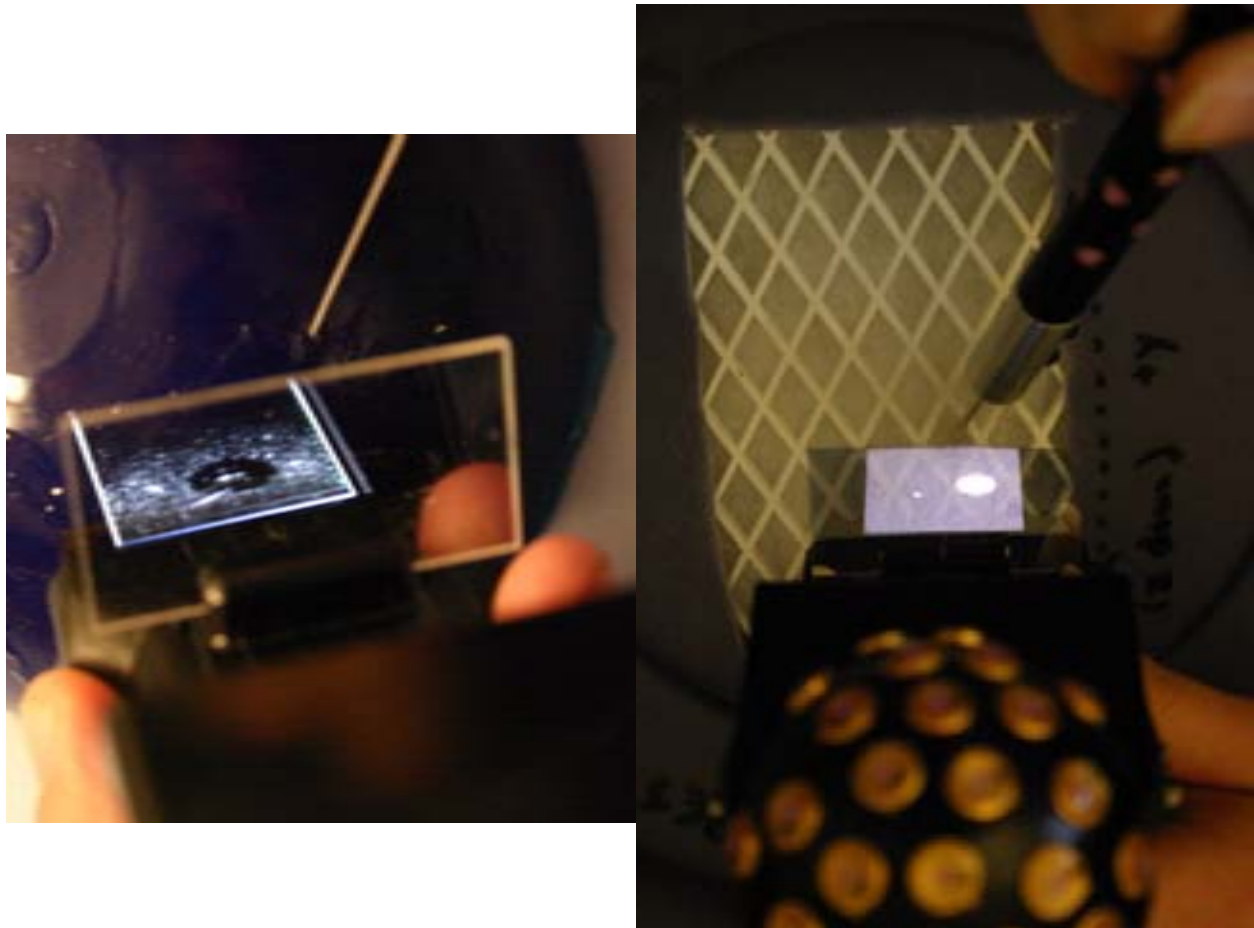


Figure 3.28: A comparison of a real phantom viewed using the clinical SF (left) and similar VTR phantom (right).

Chapter 4

Overview of subject testing experiments and procedures

4.1 Introduction

In this chapter I describe how VTR can be used to conduct psychophysics experiments, and how the data from such experiments may be analyzed in physically meaningful reference frames. In order to validate VTR as a useful psychophysics research tool, I performed 3 experiments that examine the effect of particular image parameters on the performance of the Sonic Flashlight (Chapters 5 and 6), as well as on the performance of both the SF and CUS (Chapter 7). In addition, I show how combining the data from Experiments 1 and 3 (Chapter 8) increases the statistical power of the results.

Prior research has shown that one of the primary reasons that there is a performance difference between the SF and CUS is that the former is less prone to errors in the perceived depth of the target. In the broad sense, experiments 1 and 2 are designed to test the effect of several image parameters on perceived target depth with the SF alone, and experiment 3 tests similar parameters using both the SF and CUS.

4.2 Needle-slice feedback methods

When using an ultrasound device one has two choices about how to hold the needle. One can either keep the needle within the plane of the ultrasound beam (in-plane insertion), and hence always have it visible on the ultrasound display, or one can approach the beam plane from an external point (through-plane insertion) and attempt to hit a target location without the needle being visible until it intersects the slice plane. Aiming during in-plane insertions is relatively easy since the needle is always in view, however through-plane insertions with CUS are quite difficult. The VTR system is primarily designed to model through-plane insertions due to their difficulty (they are a more interesting research problem). A through-plane insertion may be broken down into 3 basic steps.

1. Find the target with the SF or CUS.
2. Aim the needle at the target.

3. Advance the needle until it hits the slice plane and becomes visible.

Step 3 is problematic for psychophysics research because it provides subjects with feedback about their performance during the experiment; i.e. they are able to see where the needle ended up relative to the target. More importantly, the virtual image container, though it contains densely packed polyester fibers to provide needle resistance, does not prevent the user from moving the needle normal to the long axis after it is inserted. Such lateral motion would be impossible or extremely limited in a real needle insertion. The result is that if the needle is visible in the simulated ultrasound image, it is relatively straightforward for the user to visually servo the needle until it exactly lines up with the target.

It is easy to imagine ways to avoid this problem. Most obvious is an automated technique that would record the first point where the needle contacts the slice. This would tend to reward extremely careful initial aiming and would not allow subjects to adjust their aim once inside the container (or rather, it would prevent them from adjusting their aim if such an adjustment would cause the needle to contact the image plane prior to what they intend). In any case, this seems counter to the goal of measuring perceived depth. Ideally, the user would be allowed as much time and as many adjustments as needed before deciding that a particular angle and amount of inserted needle is correct.

The method I have implemented to solve this problem is to display a red bar along the transducer edge of the virtual image (or in CUS mode, at the top of the external display), when the needle has touched or passed through the virtual image plane. This gives the user information about when to stop inserting the needle but eliminates the in-plane component of the feedback. The red bar can be used to manipulate the needle such that the tip is always in contact with the virtual image, but does not provide any feedback about where in the virtual image the tip intersection occurs. Figure 4.1 shows what the user sees when they have touched the virtual image, and Figure 4.2 is a similar target where the user has not touched the virtual image.

4.3 Coordinate frames for analysis

As described in the previous chapter, the VTR system tracks a number of coordinate frames. In order to interpret the results of psychophysics experiments it is important to use a coordinate system that has physical meaning with respect to both the user and the simulated target. There are two coordinate frames that can fill this role, the virtual image frame V and the image container I . The Sonic Flashlight is held in such a way that the V_x axis extends away from the transducer tip and makes a shallow acute angle with the user's viewing vector. Thus, errors made along the V_x axis are approximately, though as will be explained below, not exactly equal to, errors in the perceived depth of simulated targets. The image container frame has one axis, I_z which is nearly identical to the earth's gravity vector, and two axes I_x and I_y which are normal to each other and to I_z but are otherwise free to rotate about I_z as the user repositions the image container on the table.

Users are instructed to hold the mock SF vertically (i.e. normal to image container surface). This orientation is easy to hold, helps to ensure adequate visibility of the IRED markers in both SF and CUS modes, and helps prevent inadvertent monocular viewing in SF mode (the users are specifically instructed to maintain binocular viewing of the target). When held normal to the image container, V_x and I_z are colinear but with opposite signs.

As shown in Figure 4.3, V_x and V_y define the xy plane of the virtual image (SF) coordinate frame V .

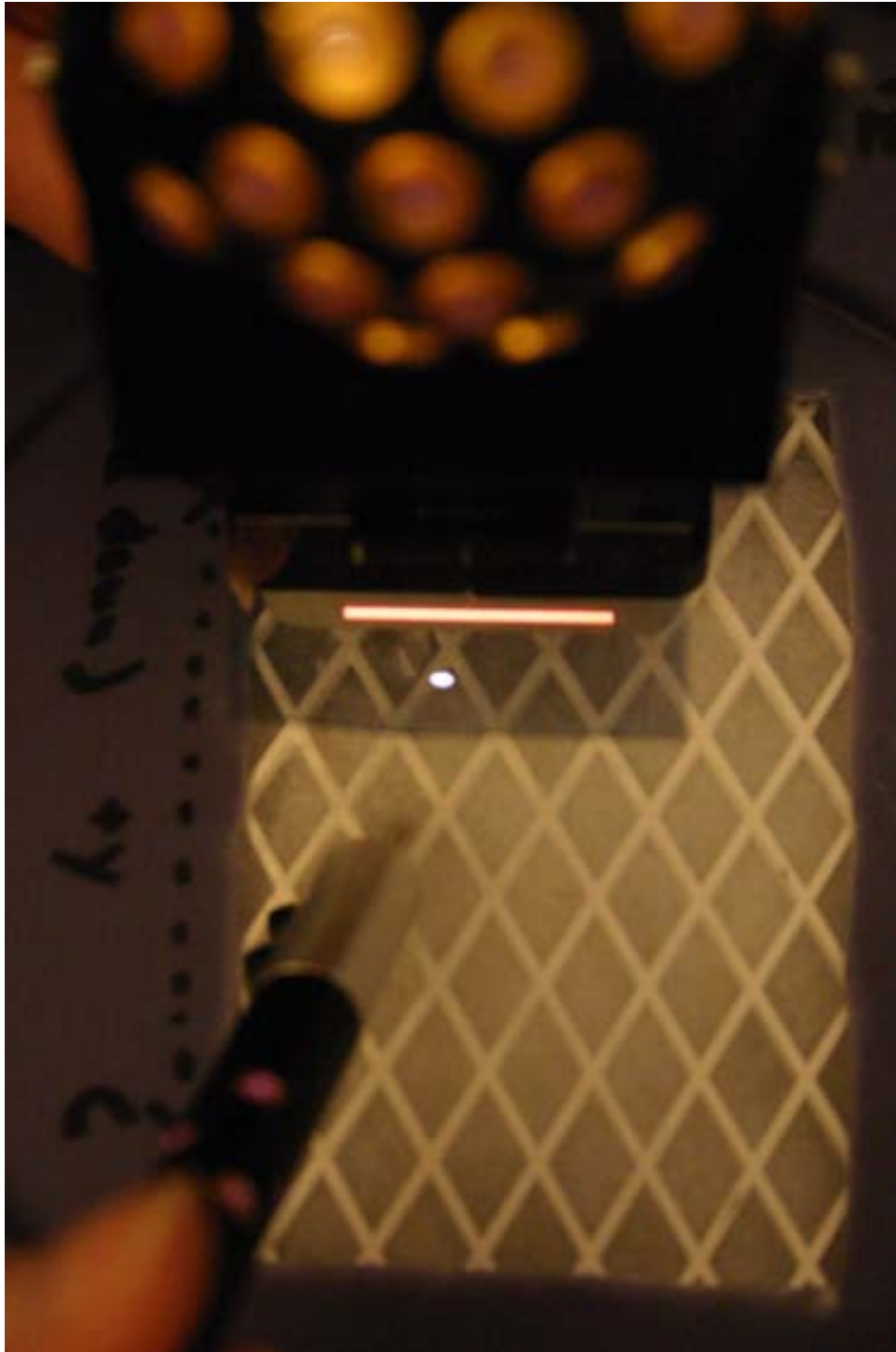


Figure 4.1: Example of the red bar slice feedback mode. In this image, the subject can tell that they have hit the slice plane but cannot tell where on the slice plane the needle has hit. The white dot visible on the virtual image is the target, not the needle intersection.

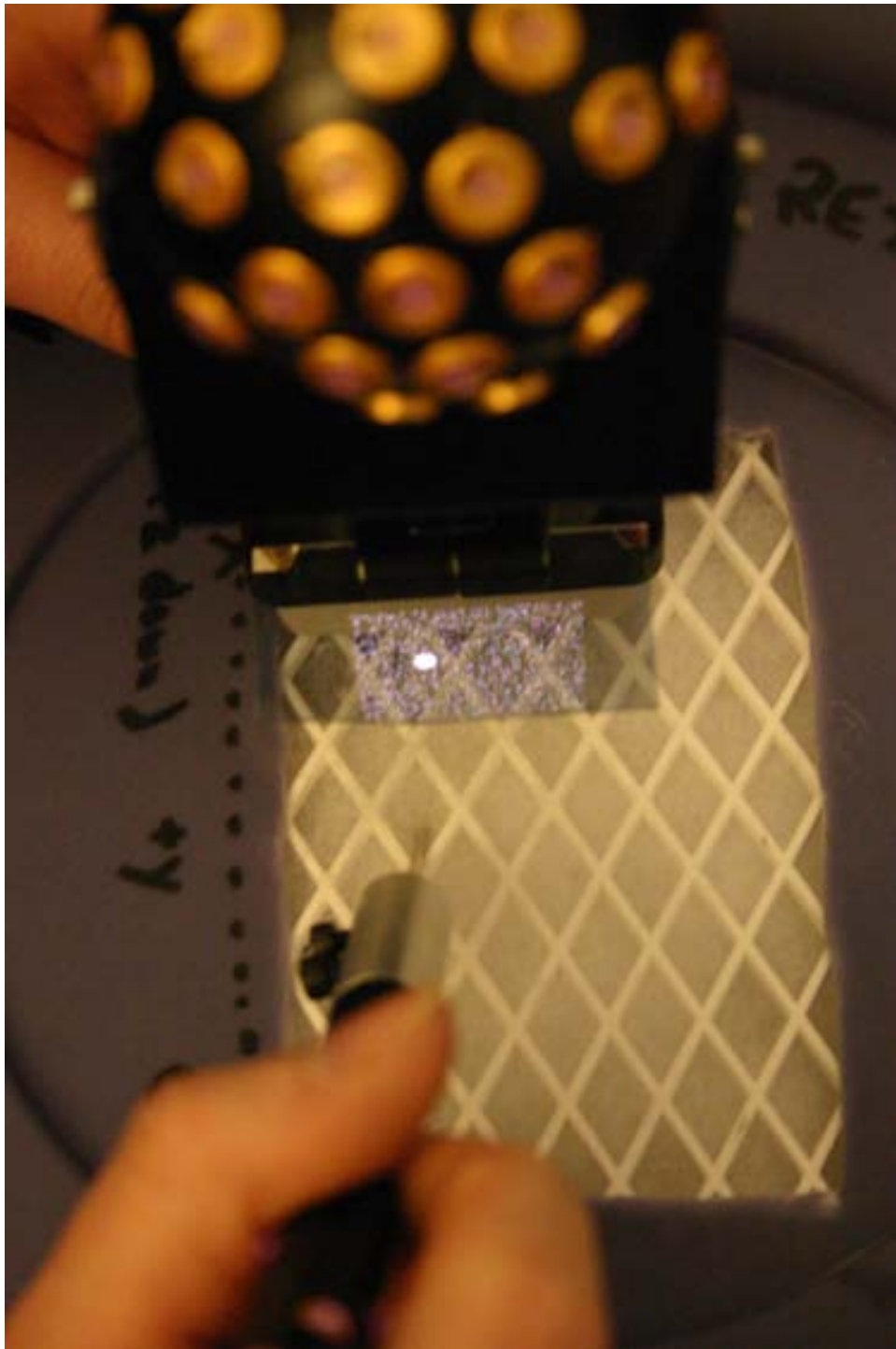


Figure 4.2: Example of a high-noise point overlay. The subject has partially inserted the needle but has not yet touched the image plane.

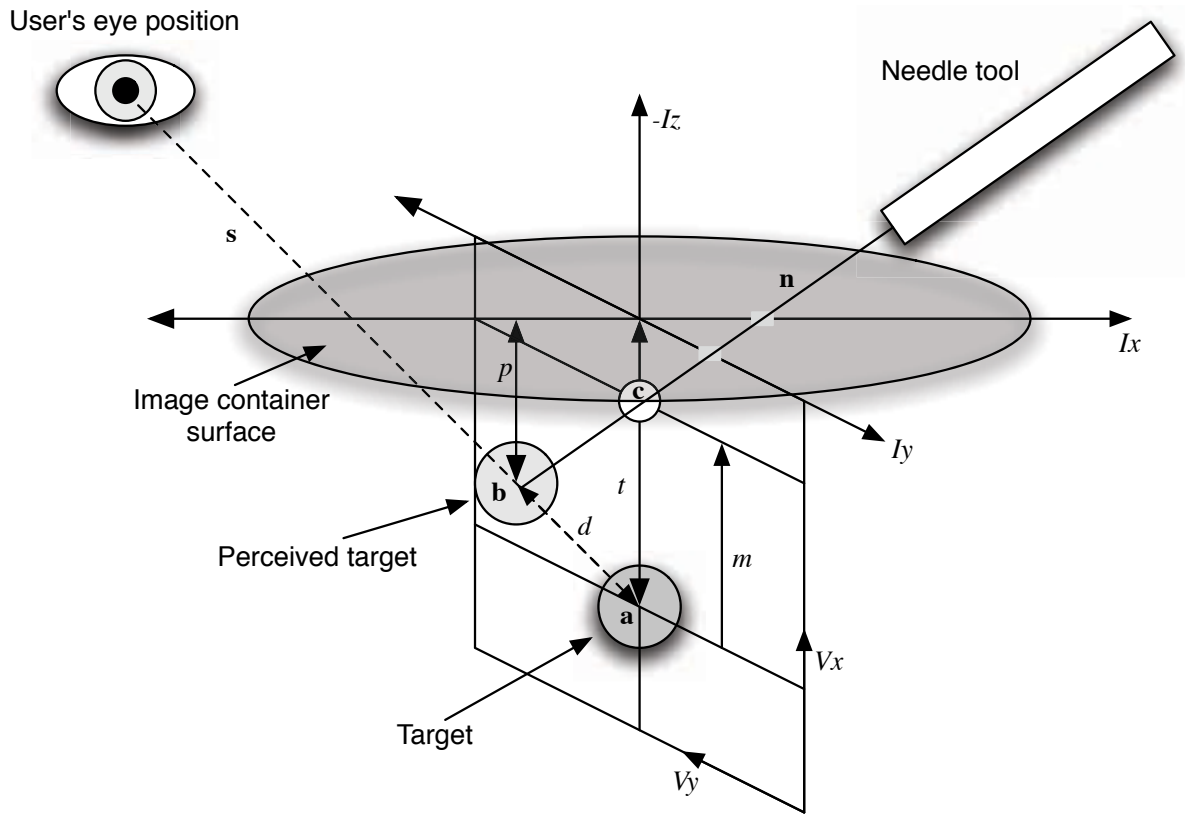


Figure 4.3: Schematic of perceived target location when using the SF in a reduced cue condition.

I_x , I_y , and I_z define the image container coordinate frame I . This illustration shows the SF being held perpendicular to the image container, in which case $-I_z = V_x$. This illustration also shows $I_y = -V_y$, although as mentioned above the I_{xy} plane has a rotational degree of freedom about I_z .

Foreshortening effects cause the target to displace, from its true location at a , along the user's sight vector s an amount d that depends on the magnitude of s . The user therefore aims the needle tool (with vector n) at a point b , the perceived location of the target. The needle intersects the virtual image plane at point c .

Regarding the ability of the VTR system to measure perceived depth relative to the surface of the image container, I observe the following:

1. The actual perceived depth is p
2. The actual target depth is t
3. The measured error in depth is m , therefore measured perceived depth is $t - m$.
4. The degree to which m fails to approximate p is a function of both the needle vector n (which is known) and the sight vector s (which is not known). In general, for shallow angles of s relative

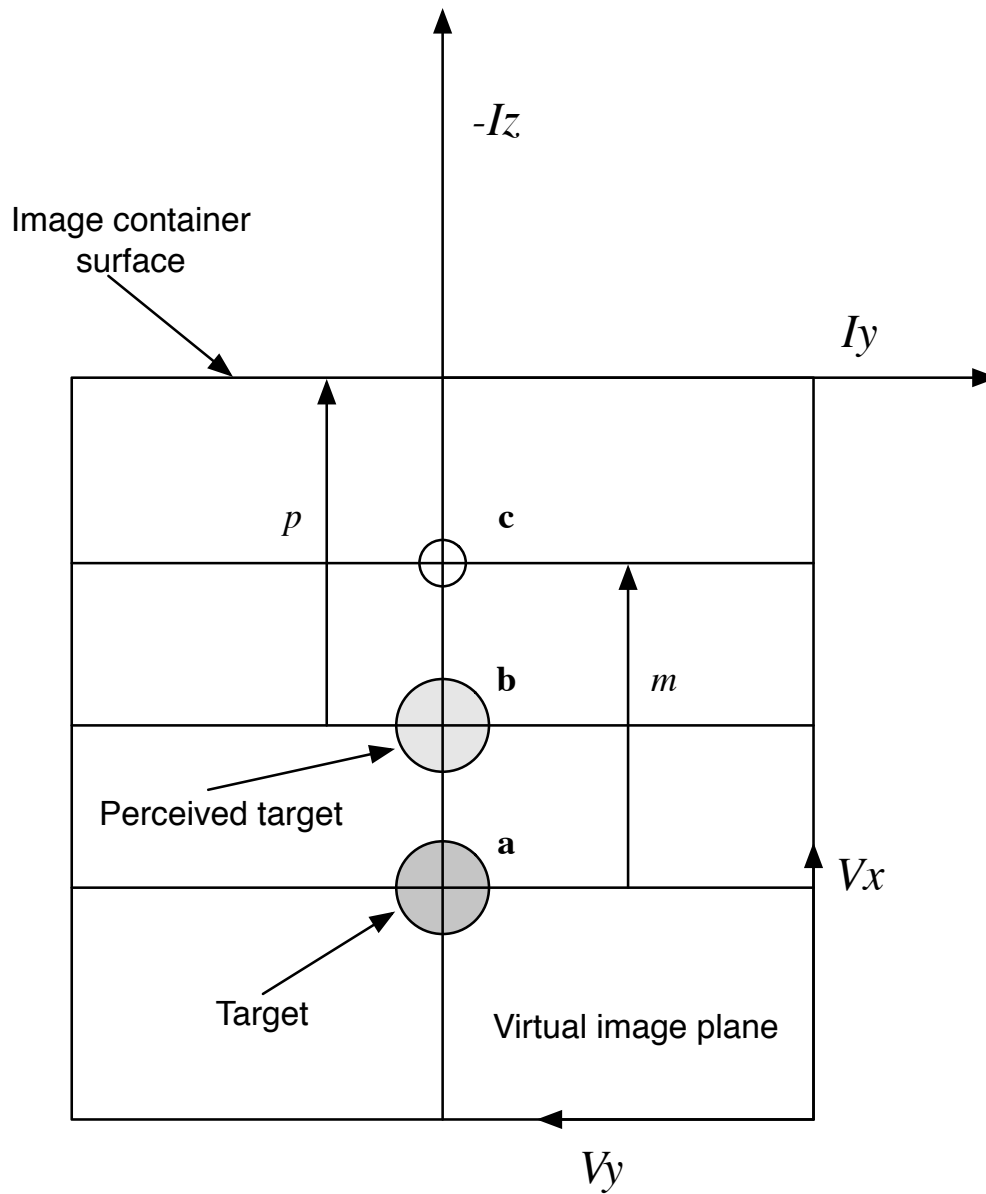


Figure 4.4: Schematic of perceived target location when using the SF in a reduced cue condition, looking at the virtual image plane en face.

errors are not shown in Figures 4.3 through 4.5. V_y error is left-right error in the virtual image plane. Correct aim in V_y is relatively easy to achieve due to strong cues from the body of the mock SF.

V_z error is somewhat less obvious. Point **c**, the intersection of the needle and virtual image plane, is by definition on the plane V_{xy} ($V_z = 0$). Assuming that the user is able to position the virtual image such that it passes exactly through the center of the target (point **a**), the error in V_z will be 0. A nonzero error in V_z is a measure of the normal distance from the plane V_{xy} to the center of the target (point **a**). In practice, users tend to be quite good at positioning the slice correctly, and the error in V_z is small.

4.4 Analysis procedures

To avoid slowing down the simulation, the VTR system does not perform any additional computations beyond what is needed to run the simulation. Instead, data are logged to an ASCII text file and later post-processed in Matlab to yield results for the axes described in the previous section. An example of a data frame can be found in Appendix B. The steps involved in post-processing the results of an experiment are:

1. Load the text file from a particular experimental subject.
2. Parse the text file to extract rigid body matrices and the location of the target.
3. Load the image container and mock SF calibration matrices.
4. Transform all rigid bodies to both I and V coordinates (from G).
5. Compute the intersection of the needle with the virtual image in both I and G .
6. Compute the error along each axis of interest and write the results to a file

The steps above are repeated for all subjects in a particular experiment and the results compiled into a single text file for each axis of interest. Finally, Microsoft Excel is used to collate the individual axes into a single spreadsheet for each experiment. Additional processing occurs within the spreadsheet to generate data that is formatted for statistical analysis with SPSS® statistical software.

4.5 Summary of experiments

While calibrating the virtual image plane, I observed that the presence or absence of a background – a grid, for instance – in the image strongly affected the perceived difficulty of the calibration task (aligning the needle tip with a calibration crosshairs). This led me to wonder whether performance during routine use of the SF might be subject a similar effect of background complexity. If binocular depth perception in near space depends on the number of points available to match between each eye, I would expect subjects to perform better with a strongly textured background than one that is weak or absent altogether.

Experiment 1 is designed to test the effect of background complexity and target depth on subject performance during a simulated biopsy task using the SF. Experiment 2 uses the same set of targets as Experiment 1, but removes the red bar feedback described earlier, requiring subjects to gauge whether

or not the needle is in contact with the image plane. Finally, Experiment 3 compares the SF and CUS under varying levels of background noise and with targets of different depths.

4.5.1 Experimental subjects

These experiments were performed with IRB approval on volunteers from the Carnegie Mellon undergraduate psychology research pool. Subjects were given a pretest to assess their binocular acuity. Since previous work performed in our lab [32] demonstrates the importance of binocular vision when using the SF, I discarded the data from subjects with a binocular acuity worse than 40" of arc. To comply with IRB requirements, subjects who did not meet the binocular acuity standard were given an abbreviated form of the study and appropriately debriefed without reference to the binocular acuity results. All subjects gave informed consent.

Chapter 5

Experiment 1: The effect of image complexity and target depth on SF accuracy

5.1 Experimental protocol

5.1.1 Experiment design

In this experiment, I examined the effect of background complexity and target depth on the ability of SF users to accurately hit a virtual target with a needle. A 3 (depth) \times 3 (background) 2-way design was implemented with 3 replications. All targets in this experiment were 1.5 mm radius white spheres, located at 15 mm, 25 mm, and 35 mm below the surface of the image container (I_z). Location in the horizontal plane (the I_{xy} plane) was randomized for each target. The background of target image was black, with overlays of 0, 250, or 5000 randomly generated white points (referred to as “no”, “low”, and “high” noise).

All 27 trials were blocked by repetition, with a 2 minute break between each block. After every 3rd trial the subject was presented with a dummy trial at a depth other than 15, 25, or 35 mm, for a total of 36 trials (12 in each block). Trial order was randomized within each block. Appendix C lists the parameters used for each trial.

15 naive subjects participated in this experiment, 10 men and 5 women.

5.1.2 Task

Subjects began each trial with the needle held outside of the image container and the SF resting in a marked location on the edge of the image container where the image would not be visible between trials. Subjects were asked to first indicate the location of a target by pointing, holding the needle tip on the surface of the image container at a location of their choosing and aiming the long axis of the needle at the target. Subjects were instructed to verbally report when their aim was correct and then hold the needle still while a single frame of data was stored. Next, the subject was instructed to hit the target with the tip of the needle, stopping on the exact center location of the target. Once the subject verbally

indicated success, another frame of data was stored, and the subject withdrew the needle and moved the SF back to the rest position.

5.1.3 Training

Prior to beginning the study, each subject was given a demonstration of how to use the SF. The subject then practiced on a series of 10 images. Unlike the images presented during the experiment itself, the training images had a randomly textured background, rather than a black background with a random point overlay. In addition, the training target spheres were of a variety of sizes.

The first 5 practice trials provided needle-slice intersection feedback in the form of a white dot. I provided the subject with verbal feedback about how to correct needle placement so that the needle tip to sphere center error was less than 0.5 mm. The second 5 practice trials used the red bar feedback system, and I verbally corrected the subject's aim until the error was less than 2.0 mm.

This training design had several purposes:

1. It is somewhat easier, in my experience, for novices to understand a visual slice through a textured background rather than a random dot overlay.
2. Verbal coaching of the subject helped to ensure each subject completely understood the task.
3. The training set was short, and different enough from the experimental set to avoid having the subjects learn to correct any perceptual errors.

5.2 Key to reference frames

As described in greater detail in Chapter 4, the following coordinate frames are used when analyzing the results from this experiment.

★ The Sonic Flashlight virtual image frame V

- V_x is “depth”, or distance along the extended centerline of the mock transducer, in the SF image plane.
- V_y is left or right in the SF image plane.
- V_z is normal to the SF image plane.

★ The target image frame I

- I_x and I_y may be arbitrarily oriented depending on the placement of the image container.
- I_z is defined as the earth's gravity vector. In other words, a displacement in the $+I_z$ direction is down.

If the user is holding the SF perpendicular to the lid of the image container, then $+V_x = -I_z$. When these values are not equal, the user is holding the SF at an angle relative to gravity. As described previously, both of these axes are approximate measures of perceived depth, thus one may interpret an error in either $+V_x$ or $-I_z$ as an indication of foreshortening of the target depth relative to the user's eyes.

5.3 Statistics

No significant effect of block (1, 2, or 3) was found on any of the experimental data. In other words, no learning took place over the course of this experiment. This is not unexpected, due to the limited feedback provided to subjects about actual needle placement during the experiment. Data were therefore averaged across the 3 blocks to yield a per-subject average for each of the background noise and depth conditions.

I present the results from the I_z (depth relative to gravity) axis for the sake of completeness. As this experiment shows, subjects are quite good at holding the SF perpendicular to the image container, and the results in I_z are equal and opposite to those in V_x . This may be taken as verification that the calibration of the VTR system is accurate, since the I_z and V_x results are arrived at through separate computations.

5.4 Needle pointing results

5.4.1 Error vector magnitude

Depth	No Noise	Low Noise	High Noise
15	5.2 (2.2)	4.7 (2.4)	5.0 (2.0)
25	8.0 (4.1)	8.0 (4.5)	8.0 (4.2)
35	11.4 (6.1)	10.8 (6.4)	11.2 (6.0)

Table 5.1: Pointing, error vector magnitude, mean (stdev), values in mm.

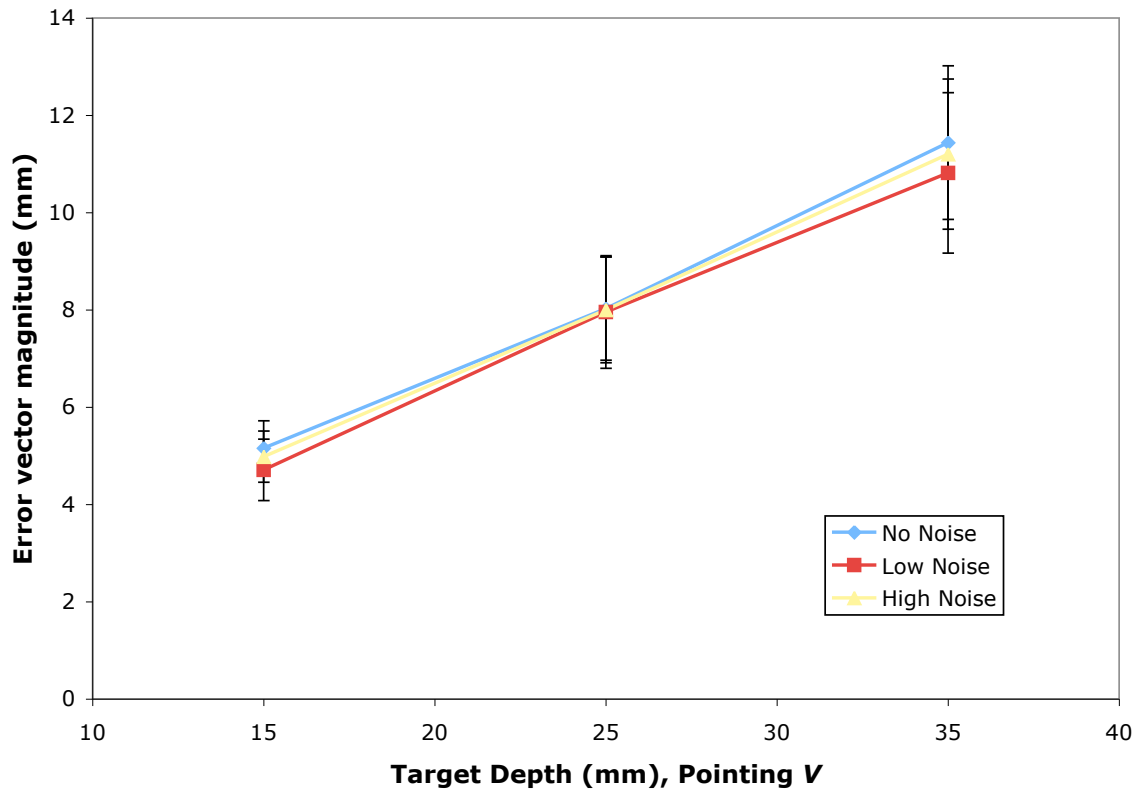


Figure 5.1: Pointing, error vector magnitude, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth was significant ($F(2, 28) = 26.443, p < 0.001$)
- ★ The main effect of background noise was not significant ($F(2, 28) = 0.660, p > 0.5$)
- ★ There was no interaction between target depth and background noise ($F(4, 56) = 0.138, p > 0.9$)

5.4.2 Error along V_x axis

Depth	No Noise	Low Noise	High Noise
15	3.0 (4.1)	2.4 (4.4)	1.4 (4.7)
25	7.1 (5.2)	6.4 (6.0)	5.4 (6.8)
35	9.8 (8.0)	9.8 (7.3)	9.0 (8.6)

Table 5.2: Pointing, V_x error, mean (stdev), values in mm.

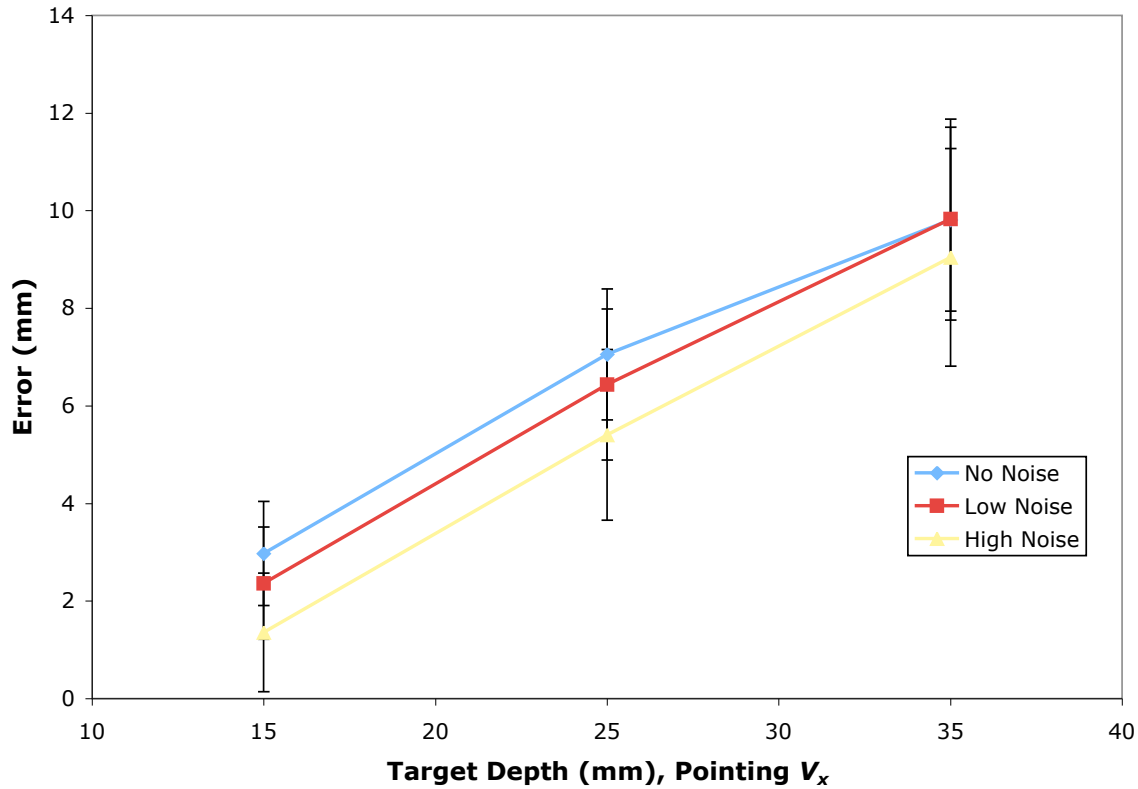


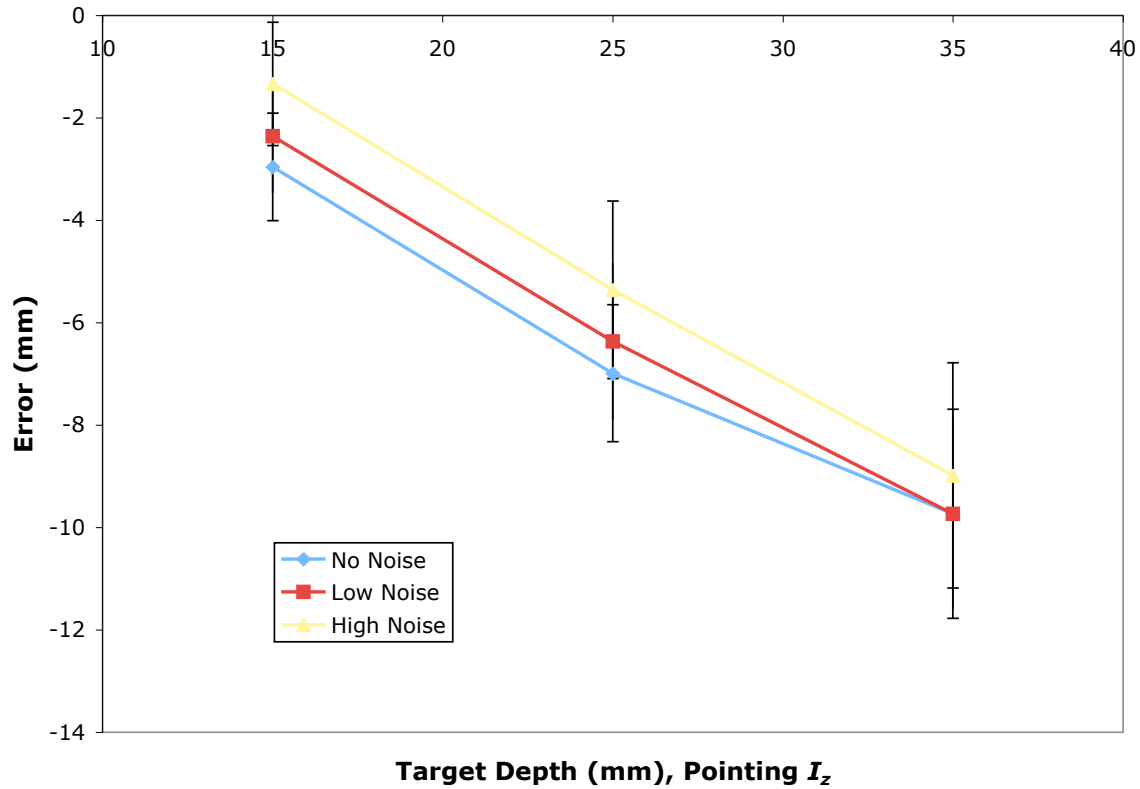
Figure 5.2: Pointing, V_x error, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth was significant ($F(2, 28) = 40.593, p < 0.001$)
- ★ The main effect of background noise was significant ($F(2, 28) = 5.905, p < 0.01$)
- ★ There was no interaction between target depth and background noise ($F(4, 56) = 0.283, p > 0.8$)

5.4.3 Error along I_z axis

Depth	No Noise	Low Noise	High Noise
15	-3.0 (4.1)	-2.4 (4.3)	-1.3 (4.7)
25	-7.0 (5.2)	-6.4 (5.9)	-5.4 (6.7)
35	-9.7 (7.9)	-9.7 (7.3)	-9.0 (8.5)

Table 5.3: Pointing, I_z error, mean (stdev), values in mm.Figure 5.3: Pointing, I_z error, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth was significant ($F(2, 28) = 40.031, p < 0.001$)
- ★ The main effect of background noise was significant ($F(2, 28) = 5.913, p < 0.01$)
- ★ There was no interaction between target depth and background noise ($F(4, 56) = 0.295, p > 0.8$)

5.4.4 Error along V_y axis

Depth	No Noise	Low Noise	High Noise
15	1.0 (1.1)	0.9 (1.1)	0.9 (1.3)
25	1.2 (1.3)	1.3 (1.6)	1.0 (1.6)
35	1.3 (1.8)	1.3 (2.0)	1.2 (1.6)

Table 5.4: Pointing, V_y error, mean (stdev), values in mm.

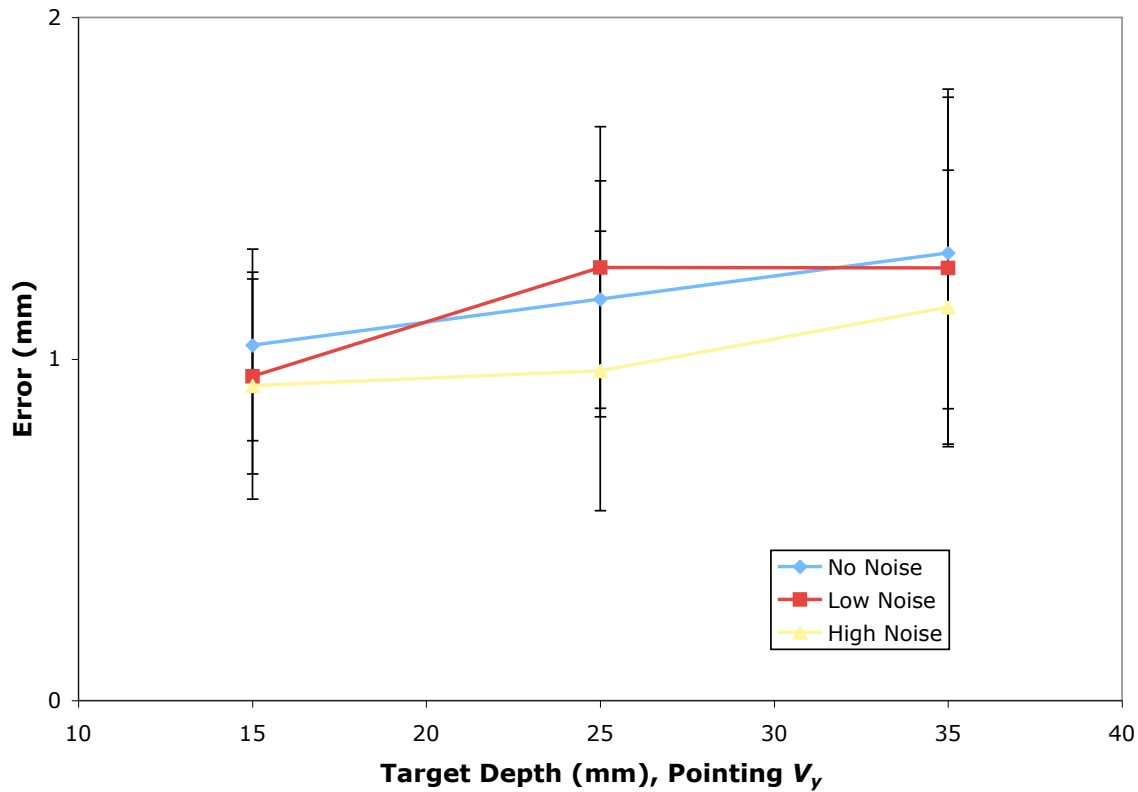


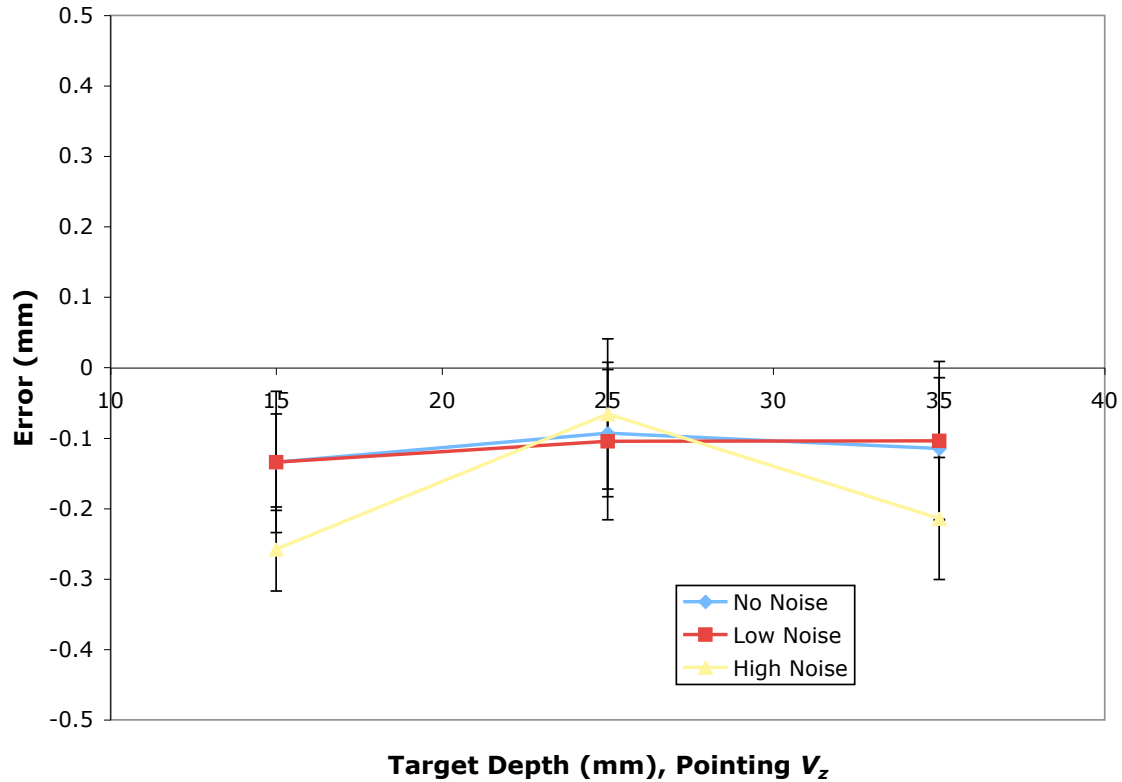
Figure 5.4: Pointing, V_y error, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth was not significant ($F(2, 28) = 0.776, p > 0.4$)
- ★ The main effect of background noise was not significant ($F(2, 28) = 0.727, p > 0.4$)
- ★ There was no interaction between target depth and background noise ($F(4, 56) = 0.186, p > 0.9$)

5.4.5 Error along V_z axis

Depth	No Noise	Low Noise	High Noise
15	-0.1 (0.3)	-0.1 (0.4)	-0.3 (0.2)
25	-0.1 (0.3)	-0.1 (0.4)	-0.1 (0.4)
35	-0.1 (0.4)	-0.1 (0.4)	-0.2 (0.3)

Table 5.5: Pointing, V_z error, mean (stdev), values in mm.Figure 5.5: Pointing, V_z error, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth was not significant ($F(2, 28) = 0.721, p > 0.4$)
- ★ The main effect of background noise was not significant ($F(2, 28) = 0.924, p > 0.4$)
- ★ There was no interaction between target depth and background noise ($F(4, 56) = 0.286, p > 0.8$)

5.4.6 Error magnitude in V_{xy} plane

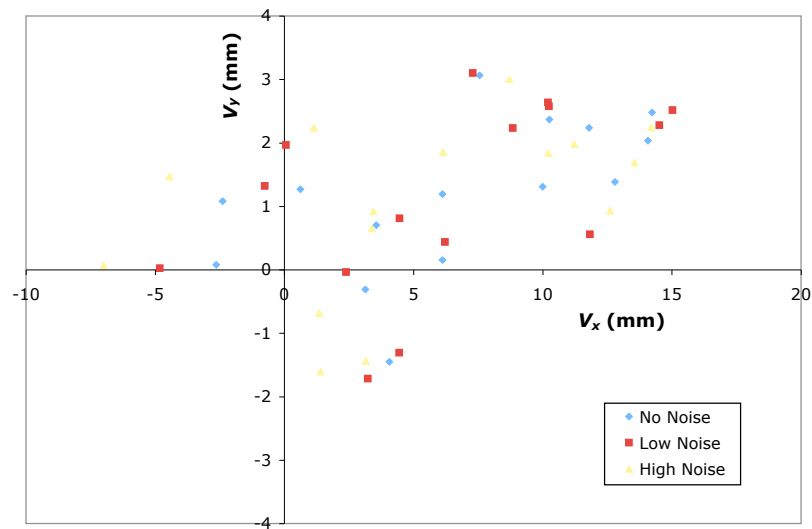


Figure 5.6: Pointing, per-subject mean errors in V_{xy} plane for each target, mean \pm standard error.

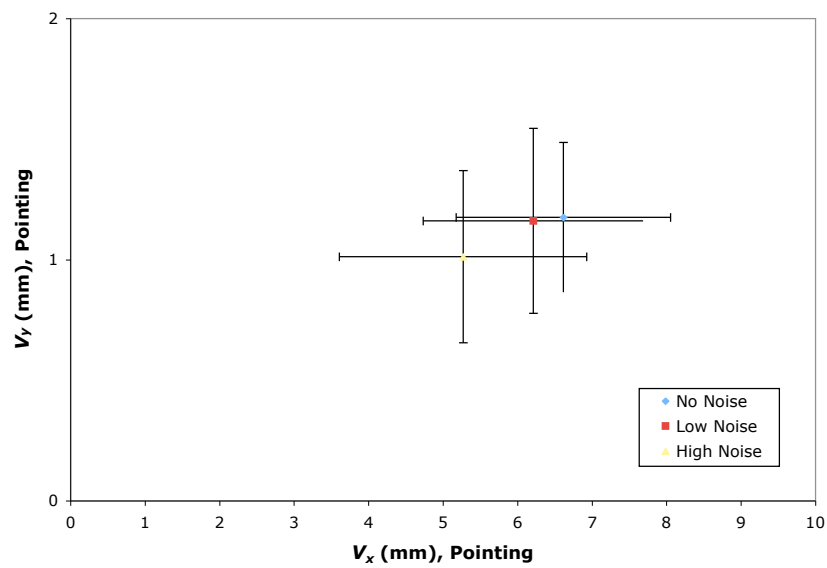


Figure 5.7: Pointing, centroids of all subjects in V_{xy} plane, mean \pm standard error.

5.5 Needle insertion results

5.5.1 Error vector magnitude

Depth	No Noise	Low Noise	High Noise
15	4.5 (1.7)	4.4 (2.6)	4.6 (2.1)
25	7.8 (3.6)	7.6 (4.4)	7.2 (3.9)
35	10.4 (5.7)	9.6 (5.6)	10.1 (6.0)

Table 5.6: Insertion, error vector magnitude, mean (stdev), values in mm.

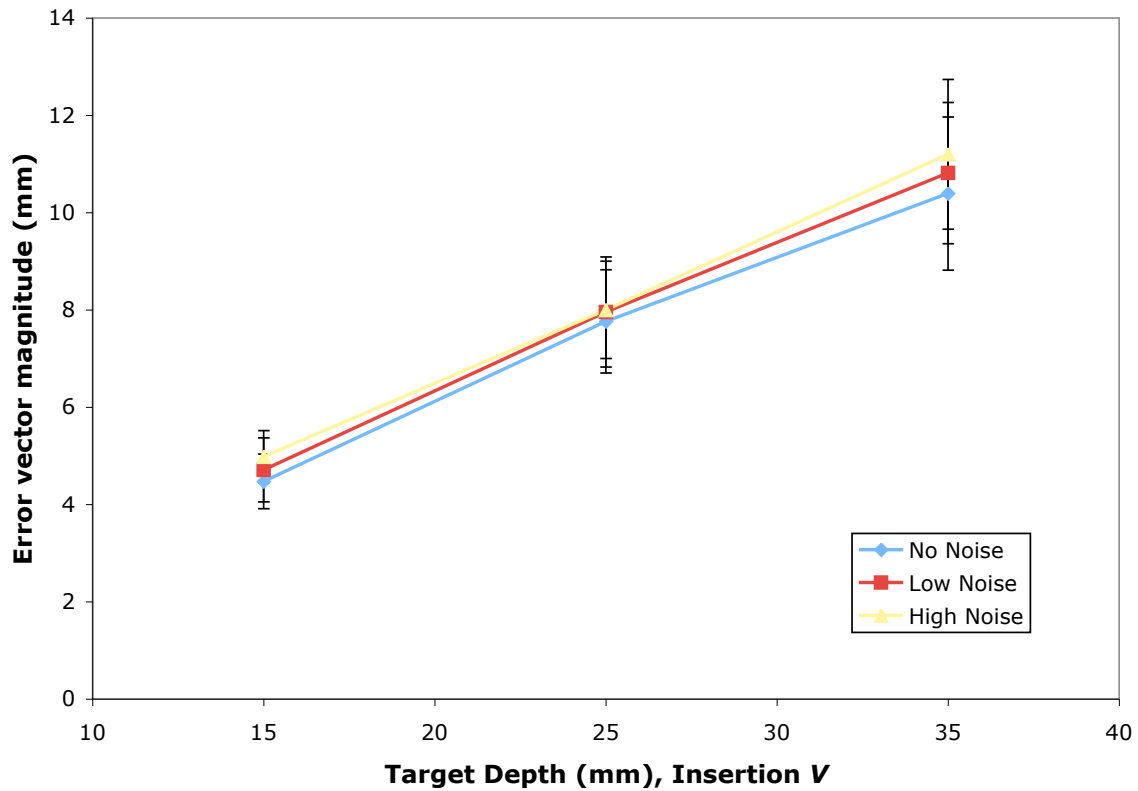


Figure 5.8: Insertion, error vector magnitude, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth was significant ($F(2, 28) = 23.482, p < 0.001$)
- ★ The main effect of background noise was not significant ($F(2, 28) = 0.394, p > 0.6$)
- ★ There was no interaction between target depth and background noise ($F(4, 56) = 0.538, p > 0.7$)

5.5.2 Error along V_x axis

Depth	No Noise	Low Noise	High Noise
15	3.0 (3.0)	2.2 (4.3)	2.0 (4.1)
25	7.0 (4.5)	6.1 (5.8)	4.8 (6.1)
35	9.2 (6.7)	8.4 (6.8)	9.2 (6.9)

Table 5.7: Insertion, V_x error, mean (stdev), values in mm.

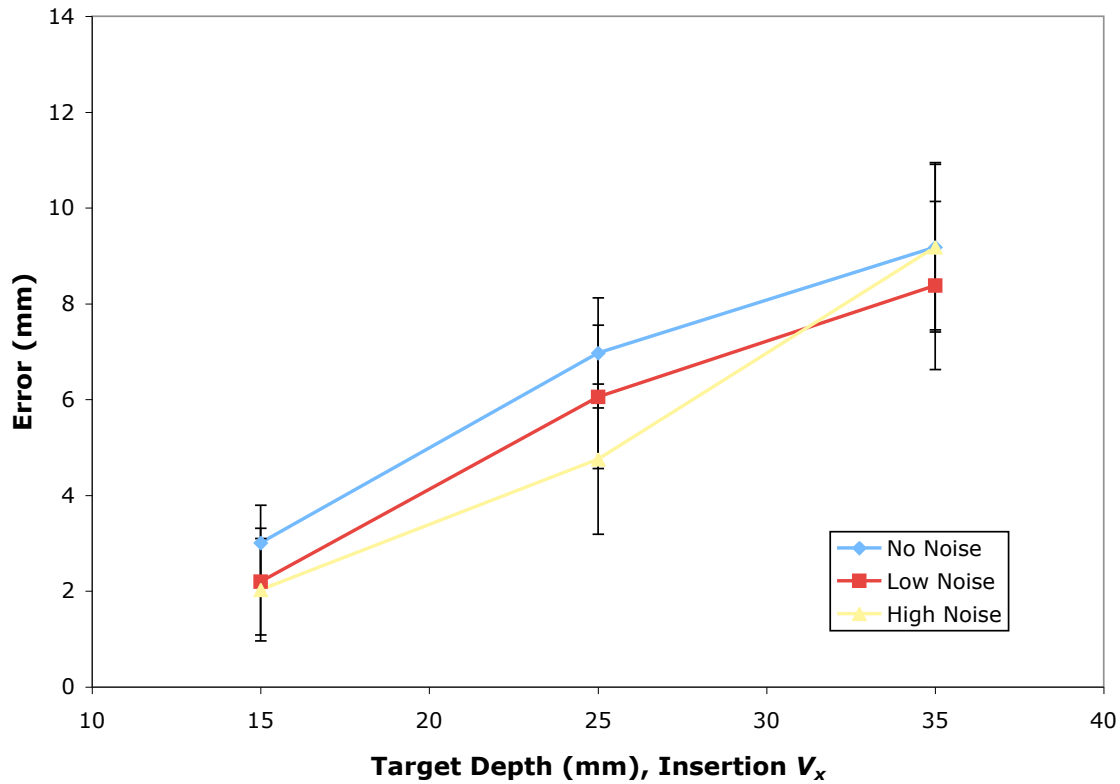


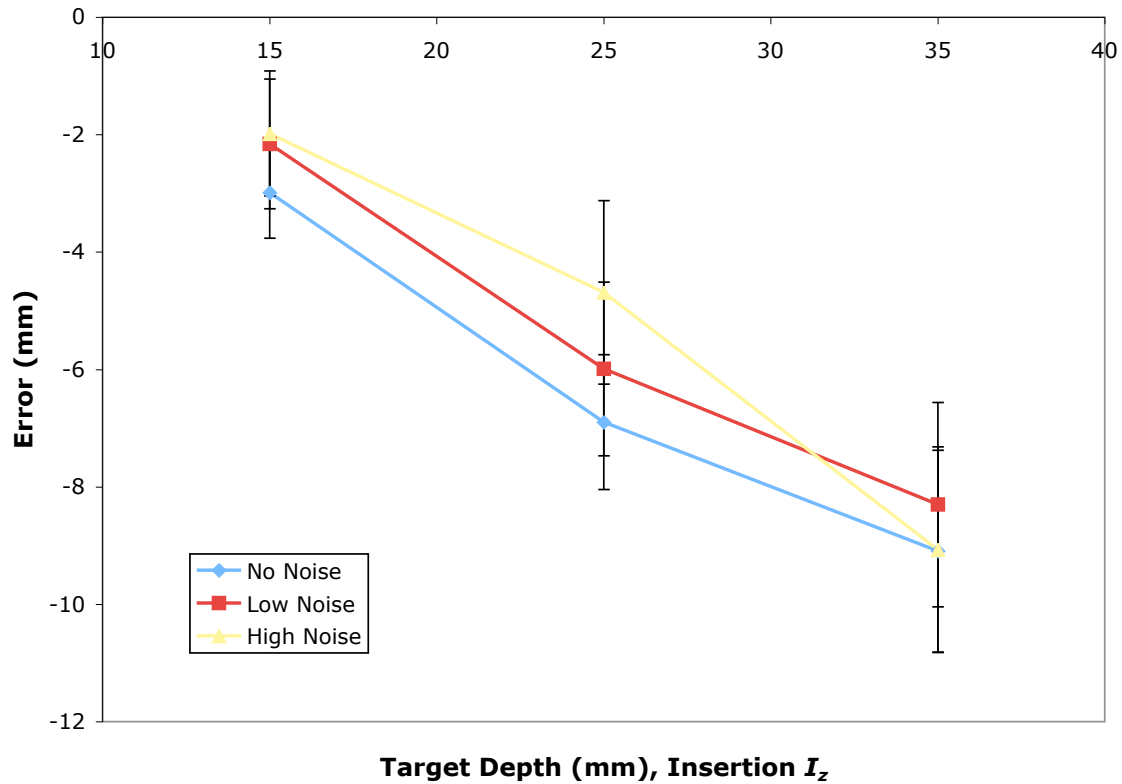
Figure 5.9: Insertion, V_x error, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth was significant ($F(2, 28) = 40.033, p < 0.001$)
- ★ The main effect of background noise was barely not significant ($F(2, 28) = 3.013, p = 0.065$)
- ★ There was no interaction between target depth and background noise ($F(4, 56) = 1.701, p > 0.1$)

5.5.3 Error along I_z axis

Depth	No Noise	Low Noise	High Noise
15	-3.0 (3.0)	-2.2 (4.3)	-2.0 (4.1)
25	-6.9 (4.4)	-6.0 (5.7)	-4.7 (6.1)
35	-9.1 (6.7)	-8.3 (6.7)	-9.1 (6.8)

Table 5.8: Insertion, I_z error, mean (stdev), values in mm.Figure 5.10: Insertion, I_z error, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth was significant ($F(2, 28) = 40.272, p < 0.001$)
- ★ The main effect of background noise was barely not significant ($F(2, 28) = 3.125, p = 0.060$)
- ★ There was no interaction between target depth and background noise ($F(4, 56) = 1.661, p > 0.1$)

5.5.4 Error along V_y axis

Depth	No Noise	Low Noise	High Noise
15	1.4 (0.9)	0.9 (1.1)	1.1 (1.3)
25	1.3 (1.4)	1.6 (1.3)	1.1 (1.3)
35	1.6 (2.1)	1.3 (1.8)	1.5 (1.7)

Table 5.9: Insertion, V_y error, mean (stdev), values in mm.

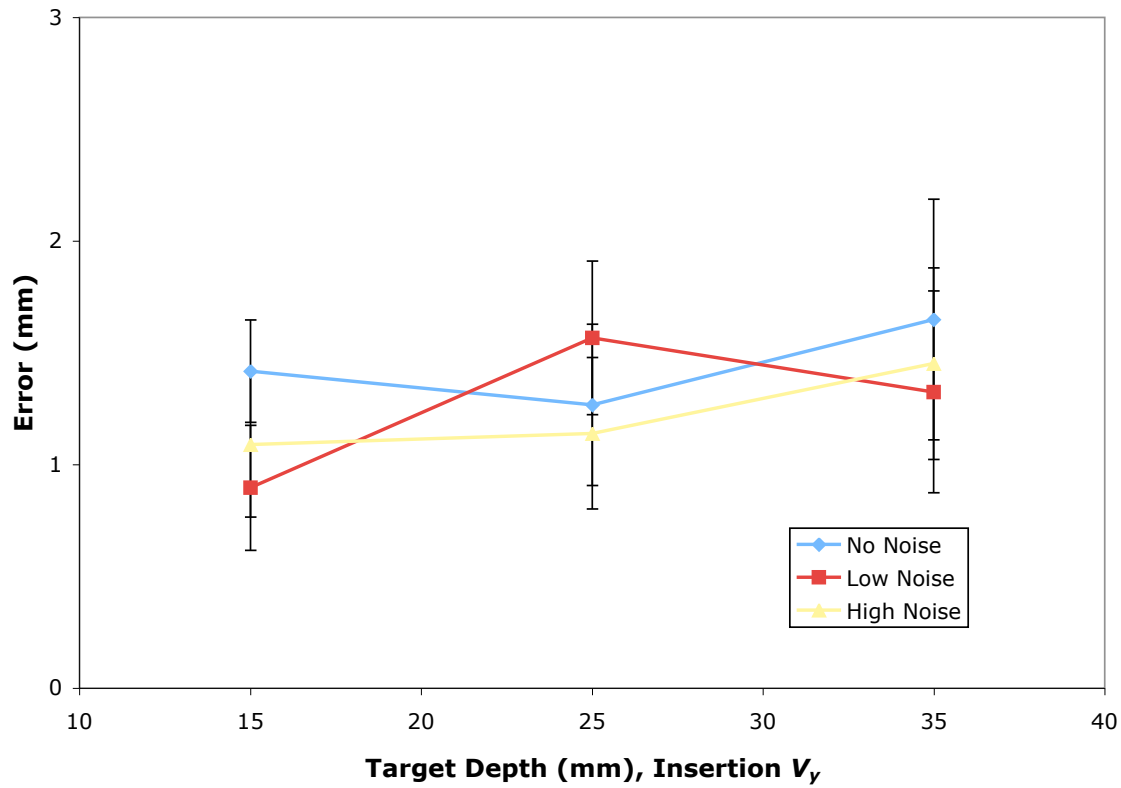


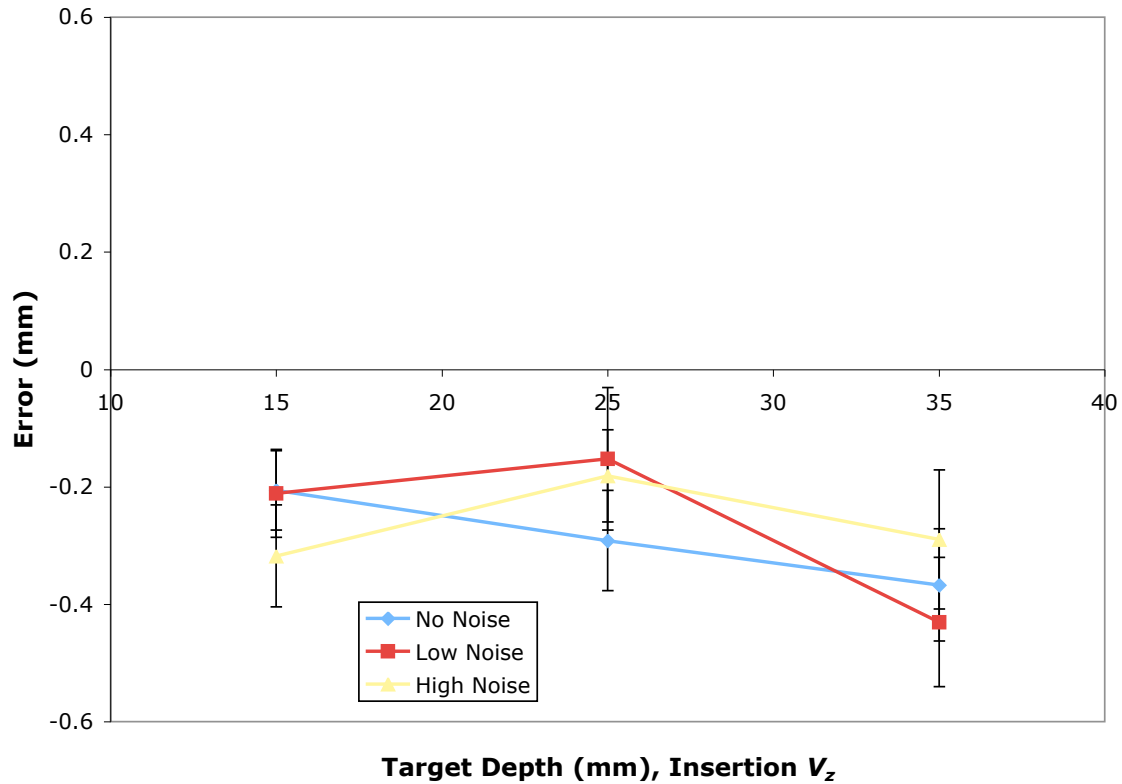
Figure 5.11: Insertion, V_y error, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth was not significant ($F(2, 28) = 1.457, p > 0.2$)
- ★ The main effect of background noise was not significant ($F(2, 28) = 1.717, p > 0.1$)
- ★ The interaction between target depth and background noise was barely not significant ($F(4, 56) = 2.321, p = 0.068$)

5.5.5 Error along V_z axis

Depth	No Noise	Low Noise	High Noise
15	-0.2 (0.3)	-0.2 (0.3)	-0.3 (0.3)
25	-0.3 (0.3)	-0.2 (0.5)	-0.2 (0.3)
35	-0.4 (0.4)	-0.4 (0.4)	-0.3 (0.5)

Table 5.10: Insertion, V_z error, mean (stdev), values in mm.Figure 5.12: Insertion, V_z error, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth was barely not significant ($F(2, 28) = 2.702, p = 0.085$)
- ★ The main effect of background noise was not significant ($F(2, 28) = 0.086, p > 0.9$)
- ★ There was no interaction between target depth and background noise ($F(4, 56) = 1.528, p > 0.2$)

5.5.6 Error magnitude in V_{xy} plane

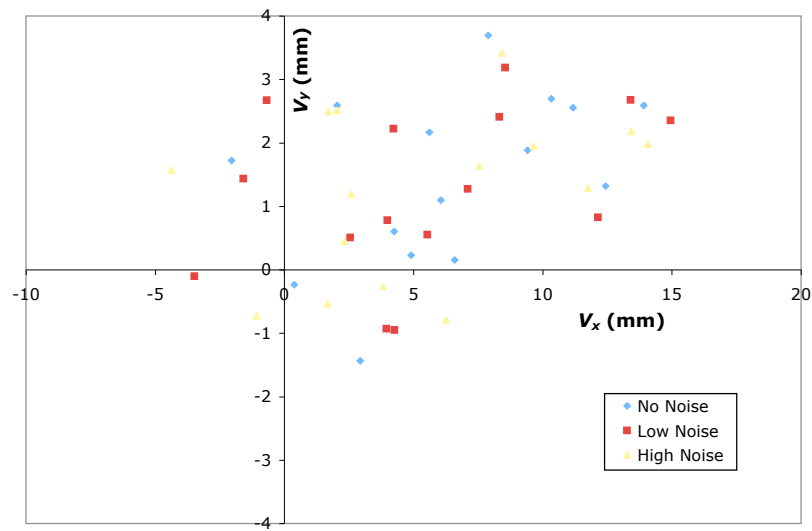


Figure 5.13: Insertion, per-subject mean errors in V_{xy} plane for each target.

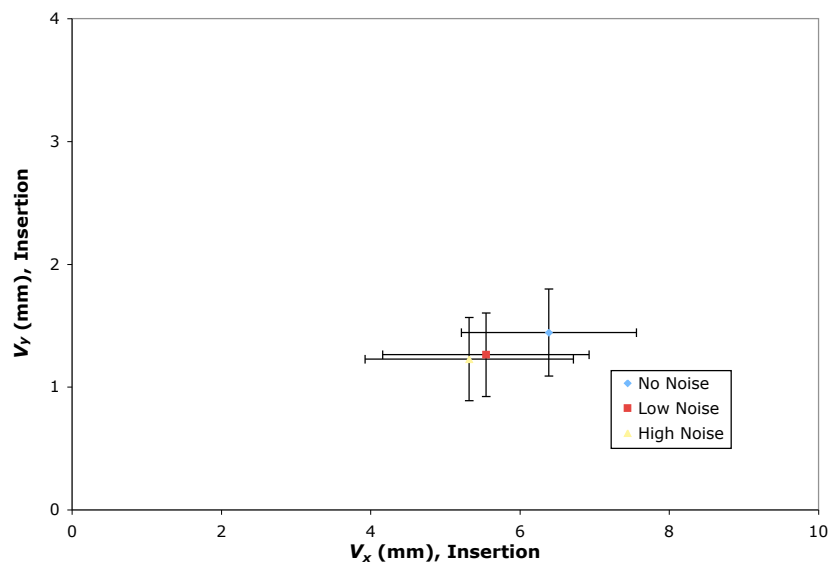


Figure 5.14: Insertion, centroids of all subjects in V_{xy} plane, mean \pm standard error.

5.6 Ranking subjects by performance

5.6.1 Introduction

For a variety of reasons, some subjects perform consistently better than other subjects. This experiment is not designed to address the mechanism behind the variability of performance between subjects, although I can speculate that the following factors may be involved:

- ★ The natural hand-eye coordination of the subject.
- ★ The degree of interest the subject has in the experiment, or their desire to “do a good job”.
- ★ Failure by the subject to adequately ensure stereo vision of the target, despite reminders to do so.

I propose two methods of ranking the participants in this experiment. The first, ranking by error in the V_{xy} plane, is based on the mean performance of each subject over the full set of background noise and target depths for the pointing and insertion tasks. The second ranking method looks at the correlation between the overall mean error of each subject and the improvement (or lack thereof) between the high and low noise background conditions.

5.6.2 Ranking subject performance by V_{xy} error vector magnitude

Ranking subjects by their overall V_{xy} error vector magnitude serves two purposes. First, if a consistent ranking is present in both the pointing and insertion tasks, this is evidence that the same perceptual mechanism is used in both tasks. I.e., pointing and insertion are different ways of measuring perceived depth of the target, and subjects who are good with one response method are probably good with the other. Second, such a ranking is a useful for gaining insights into the relationship between error magnitude and error variability. In other words, are subjects who make large errors more or less likely to have high variability in performance?

As the rankings presented in Tables 5.11 and 5.12 clearly show, the top 50% of subjects (top 8 of 15) are identical between both the pointing and insertion rankings, although the ranking within the top 50% varies. In addition, Figures 5.15 and 5.16 show the correlation between V_{xy} error magnitude and V_{xy} error magnitude standard error. In other words, subjects who perform well (have a low mean error) are also the most consistent (smallest standard error). These results are significant for both pointing ($R^2 = 0.8299$, $p < 0.001$) and insertion ($R^2 = 0.8707$, $p < 0.001$).

Subject	V_x Error	V_y Error	Error in V_{xy} plane
11	2.3 (3.6)	-0.3 (1.3)	3.4 (2.8)
3	-2.5 (3.3)	1.3 (0.7)	3.6 (2.5)
9	3.8 (3.3)	0.8 (0.7)	4.4 (2.7)
14	0.6 (5.2)	1.8 (1.9)	5.0 (3.0)
10	4.2 (3.7)	-1.0 (1.5)	5.0 (3.0)
7	3.3 (5.2)	-1.5 (1.2)	5.4 (3.4)
8	-4.8 (4.6)	0.1 (1.4)	5.6 (3.9)
12	5.2 (5.0)	0.8 (1.3)	5.9 (4.4)
2	7.9 (3.9)	3.1 (1.3)	8.6 (3.6)
4	8.4 (6.2)	2.2 (1.4)	9.6 (4.7)
6	10.1 (4.2)	1.9 (1.0)	10.4 (4.1)
5	11.1 (4.3)	2.3 (1.2)	11.4 (4.3)
13	12.4 (5.5)	1.0 (1.2)	12.6 (5.3)
1	14.0 (6.3)	2.0 (1.8)	14.3 (6.3)
15	14.5 (5.6)	2.4 (1.4)	14.7 (5.6)

Table 5.11: Subject ranking by V_{xy} error during pointing, mean (stdev), values in mm.

Subject	V_x Error	V_y Error	Error in V_{xy} plane
9	3.0 (2.4)	0.5 (0.8)	3.3 (2.2)
3	-2.7 (2.7)	1.6 (0.6)	3.7 (1.7)
7	3.0 (4.2)	-1.0 (1.1)	4.5 (2.8)
14	1.1 (4.4)	2.6 (0.8)	4.7 (2.3)
12	4.2 (4.1)	1.0 (0.9)	4.8 (3.5)
11	4.8 (3.3)	0.2 (1.0)	4.9 (3.3)
8	-1.4 (5.5)	-0.4 (2.0)	4.9 (3.3)
10	5.6 (3.6)	-0.5 (1.4)	5.8 (3.6)
4	3.8 (5.2)	2.3 (1.0)	6.0 (3.3)
5	8.3 (4.3)	1.9 (1.4)	8.7 (4.1)
2	8.3 (4.1)	3.4 (1.6)	9.1 (4.0)
6	9.1 (3.7)	2.1 (1.1)	9.4 (3.7)
13	12.1 (5.4)	1.1 (1.1)	12.3 (5.2)
1	12.7 (5.5)	2.5 (1.5)	13.0 (5.4)
15	14.3 (5.6)	2.3 (1.3)	14.5 (5.7)

Table 5.12: Subject ranking by V_{xy} error during insertion, mean (stdev), values in mm.

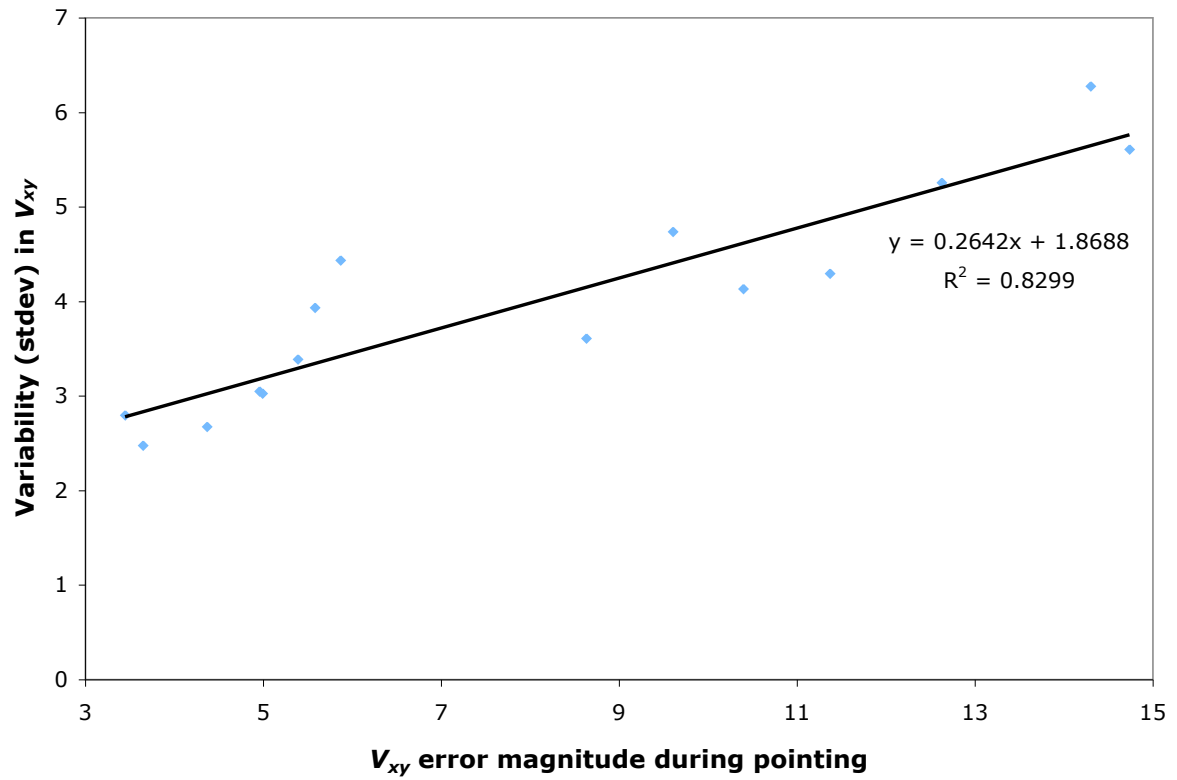


Figure 5.15: Correlation of mean errors with standard deviation in V_{xy} plane, per subject, during pointing.

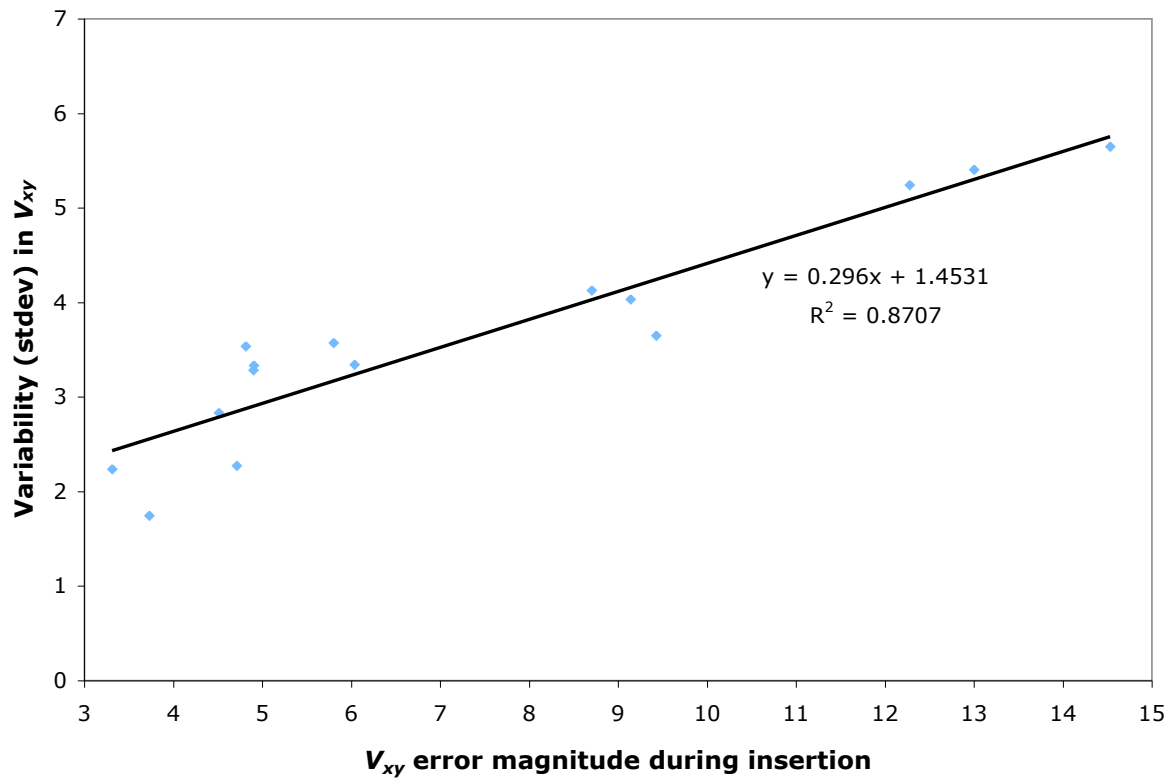


Figure 5.16: Correlation of mean errors with standard deviation in V_{xy} plane, per subject, during insertion.

5.6.3 Correlation between mean subject performance and high - low noise performance

A second way to examine the performance of individual subjects is to look at the difference between their mean performance with a high noise background and with a no noise background (ignoring the low noise case). This asks the question, “Does the subject do better or worse with the high noise background, as opposed to the no noise?” When computing error (high noise) - error (no noise), a negative value indicates an improvement (i.e. reduction of error) of high relative to no noise.

Table 5.13 shows the ranking of subjects by amount of improvement in the high noise condition. The correlation between V_x error and high vs. low noise difference during pointing (Figure 5.17) is barely not significant ($R^2 = 0.236$, $p = 0.067$).

Subject	Mean V_x Error	High - No noise
8	-4.8	-4.4
4	8.4	-4.1
10	4.2	-3.0
12	5.2	-2.7
7	3.3	-2.7
3	-2.5	-2.0
11	2.3	-1.8
5	11.1	-0.6
1	14.0	-0.5
13	12.4	-0.2
9	3.8	-0.1
15	14.5	0.0
6	10.1	0.2
14	0.6	0.5
2	7.9	1.2

Table 5.13: Subject ranking by High - No Noise difference in V_x during pointing.

Table 5.14 shows the ranking of subjects by amount of improvement in the high noise condition. The correlation between V_x error and high vs. low noise difference during insertion (Figure 5.18) is significant ($R^2 = 0.275$, $p < 0.05$).

This effect is present on the I_z axis as well – as would be expected since V_x and I_z are approximately equal and opposite – but not present on the V_y (left-right) or V_z (normal) axes during either pointing or insertion.

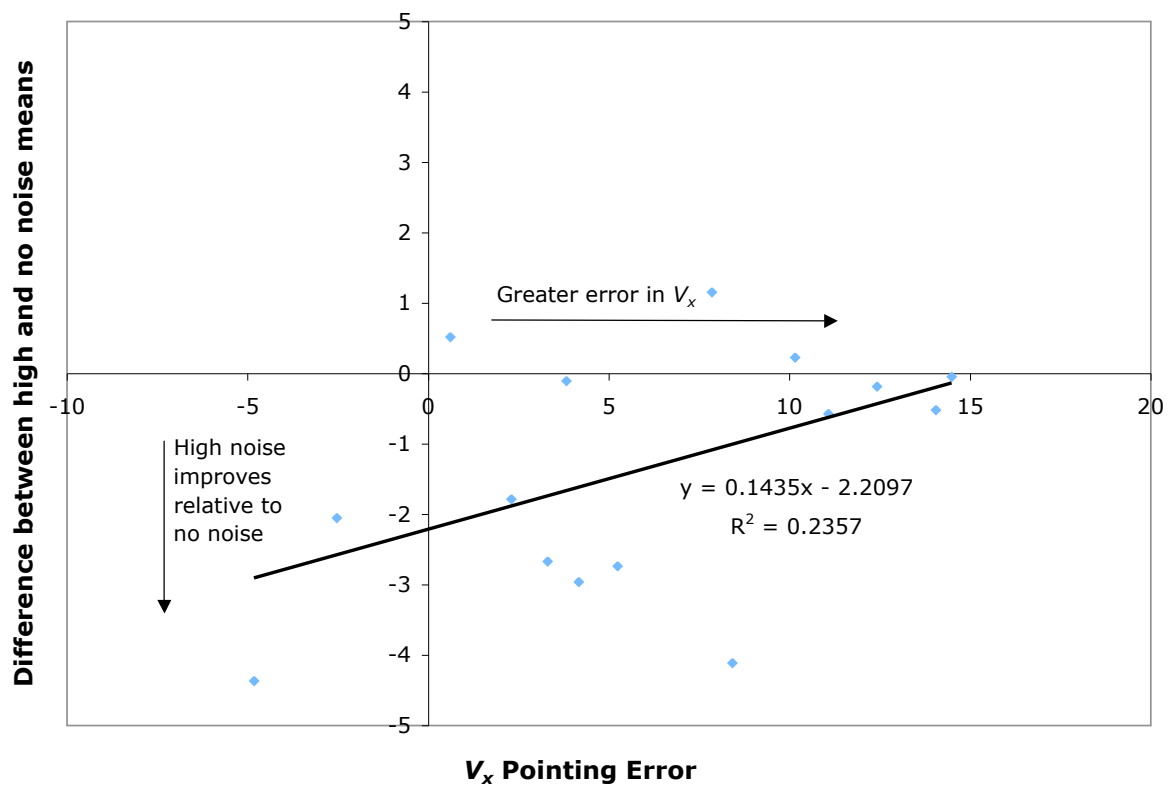


Figure 5.17: Correlation between V_x error and high vs. low noise difference during pointing.

Subject	Mean V_x Error	High - No noise
4	3.8	-3.9
12	4.2	-3.5
5	8.3	-2.8
3	-2.7	-2.3
9	3.0	-1.9
8	-1.4	-1.5
7	3.0	-1.3
11	4.8	-1.1
13	12.1	-0.7
10	5.6	-0.3
14	1.1	0.0
15	14.3	0.2
6	9.1	0.2
2	8.3	0.5
1	12.7	2.3

Table 5.14: Subject ranking by High - No Noise difference in V_x during insertion.

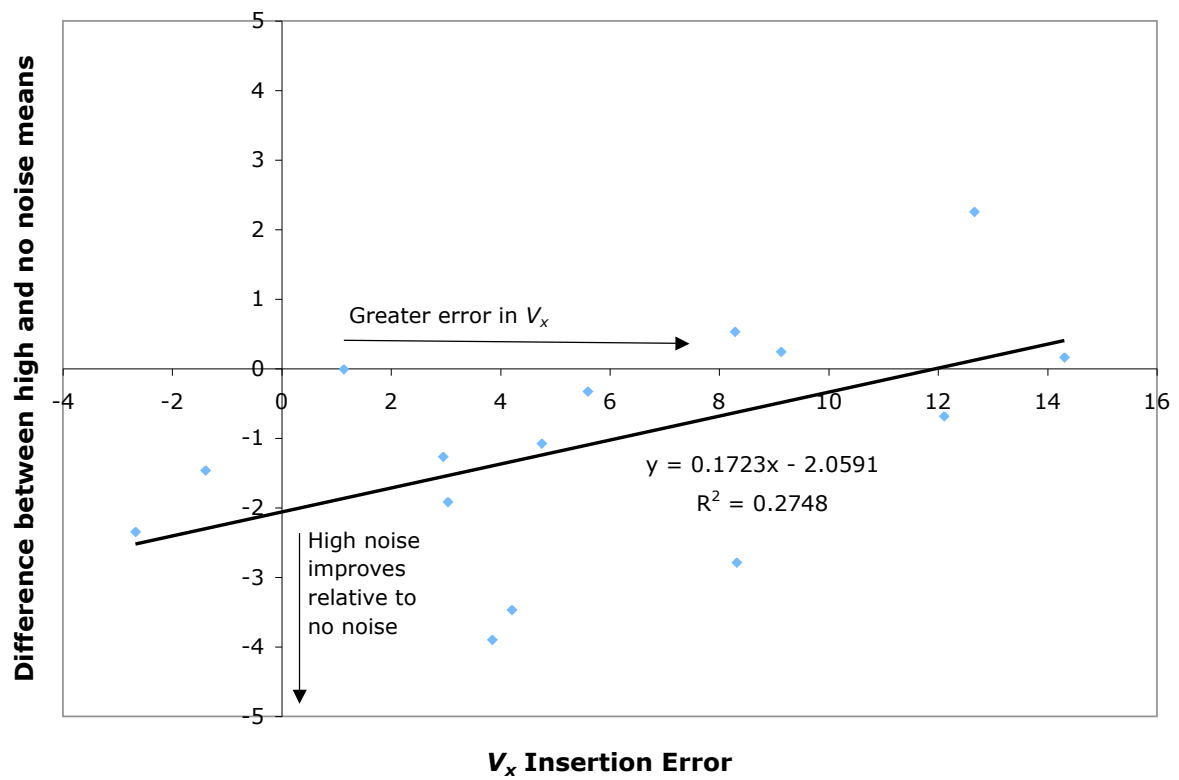


Figure 5.18: Correlation between V_x error and high vs. low noise difference during insertion.

5.7 Summary of Experiment 1 results

The take-home message from Experiment 1 is as follows:

- ★ Target depth has a significant effect on error in the V_x or I_z directions during both the pointing and insertion tasks. The deeper a target is, the greater the foreshortening error (movement of the perceived location towards the subject).
- ★ Background noise has a significant effect on perceived target depth during pointing, and is only barely not significant during insertion. Without the presence of a strongly textured background, subjects are prone to foreshortening errors in V_x and I_z .
- ★ No significant effects of background noise were found in V_y (left-right) during pointing or insertion.
- ★ No significant effects of background noise were found in V_z (normal to the image plane) during pointing or insertion. Subjects are quite good at slicing through the center of the target regardless of its depth or the amount of background noise.
- ★ The correlation between mean error magnitude in the V_{xy} plane and standard deviation is significant. Subjects who perform well are the most consistent; similarly, subjects who perform poorly have high variability.
- ★ The correlation between mean error magnitude in V_x and the improvement of high noise versus low noise performance ~~are~~ is significant during insertion (and barely not significant during pointing). Subjects who perform well, as measured by V_x (or foreshortening) error, rely more on the presence of background noise/texture than do subjects who perform poorly. This effect is not present on the V_y (left-right) or V_z (normal) axes.

Chapter 6

Experiment 2: The effect of needle tip feedback on SF accuracy

6.1 Experimental protocol

6.1.1 Experiment design

In this experiment, I examined the effect of background complexity and target depth on the ability of SF users to accurately hit a virtual target with a needle. Unlike Experiment 1, subjects received *no feedback* regarding the location of the needle tip relative to the image plane; in Experiment 1 a red bar illuminated to alert subjects that the needle tip was at or through the image plane. This study had three primary purposes:

1. Explore the effect that lack of tip-slice feedback has on the penetration of the needle through the slice. In other words, can subjects stop on the slice absent feedback to that effect?
2. Test the effect of background noise on penetration (in the V_z direction) of the needle through the slice.
3. Provide additional validation of the depth and background noise effects from Experiment 1. Experiment 2 shares a common pointing task with Experiment 1, since slice feedback is irrelevant to pointing.

Aside from the elimination of tip-slice intersection feedback, the design of this experiment is otherwise the same as Experiment 1 (see Chapter 5 and Appendix C).

9naive subjects participated in this experiment, 6 men and 3 women.

6.1.2 Task

Subjects began each trial with the needle held outside of the image container and the SF resting in a marked location on the edge of the image container where the image would not be visible between trials. Subjects were asked to first indicate the location of a target by pointing, holding the needle tip on the surface of the image container at a location of their choosing and aiming the long axis of the needle

at the target. Subjects were instructed to verbally report when their aim was correct and then hold the needle still while a single frame of data was stored. Next, the subject was instructed to hit the target with the tip of the needle, stopping on the exact center location of the target. Once the subject verbally indicated success, another frame of data was stored, and the subject withdrew the needle and moved the SF back to the rest position.

The primary difference between the task in this experiment and that in Experiment 1 is the addition of the judgement of when the needle is hitting the virtual image plane. This yields an additional piece of data for each trial, the *location of the needle tip after the needle is inserted* (needle tip location during pointing is not relevant to my analysis).

6.1.3 Training

Prior to beginning the study, each subject was given an explanation of how the SF works and given a brief demonstration of its usage. The subject then practiced on a series of 10 images. Unlike the images presented during the experiment itself, the training images had a randomly textured background, rather than a black background with a random point overlay. In addition, the training target spheres were of a variety of sizes. All 10 practice trials used the red bar feedback system, and I verbally corrected the subject's aim until the error was less than 2.0 mm. This differs from the Experiment 1 training by providing limited experience with white dot needle feedback to minimize learning effects due to feedback about actual needle location.

6.2 Key to reference frames

As described in greater detail in Chapter 4, the following coordinate frames are used when analyzing the results from this experiment.

★ The Sonic Flashlight virtual image frame V

- V_x is “depth”, or distance along the extended centerline of the mock transducer, in the SF image plane.
- V_y is left or right in the SF image plane.
- V_z is normal to the SF image plane.

★ The target image frame I

- I_x and I_y may be arbitrarily oriented depending on the placement of the image container.
- I_z is defined as the earth's gravity vector. In other words, a displacement in the $+I_z$ direction is down.

If the user is holding the SF perpendicular to the lid of the image container, then $+V_x = -I_z$. When these values are not equal, the user is holding the SF at an angle relative to gravity. As described previously, both of these axes are approximate measures of perceived depth, thus one may interpret an error in either $+V_x$ or $-I_z$ as an indication of foreshortening of the target depth relative to the user's eyes.

6.3 Statistics

No significant effect of block (1, 2, or 3) was found on any of the experimental data. In other words, no learning took place over the course of this experiment. This is not unexpected, due to the limited feedback provided to subjects about actual needle placement during the experiment. Data were therefore averaged across the 3 blocks to yield a per-subject average for each of the background noise and depth conditions.

I do not present the results from I_z axis in this experiment (although it was collected and analyzed), since as Experiment 1 demonstrates, it is virtually identical to the V_x axis.

6.4 Needle pointing results

6.4.1 Error vector magnitude

Depth	No Noise	Low Noise	High Noise
15	4.3 (1.8)	4.3 (1.8)	3.7 (1.7)
25	8.3 (3.4)	6.5 (3.1)	6.6 (3.9)
35	13.0 (4.4)	11.7 (4.1)	9.6 (4.5)

Table 6.1: Pointing, error vector magnitude, mean (stdev), values in mm.

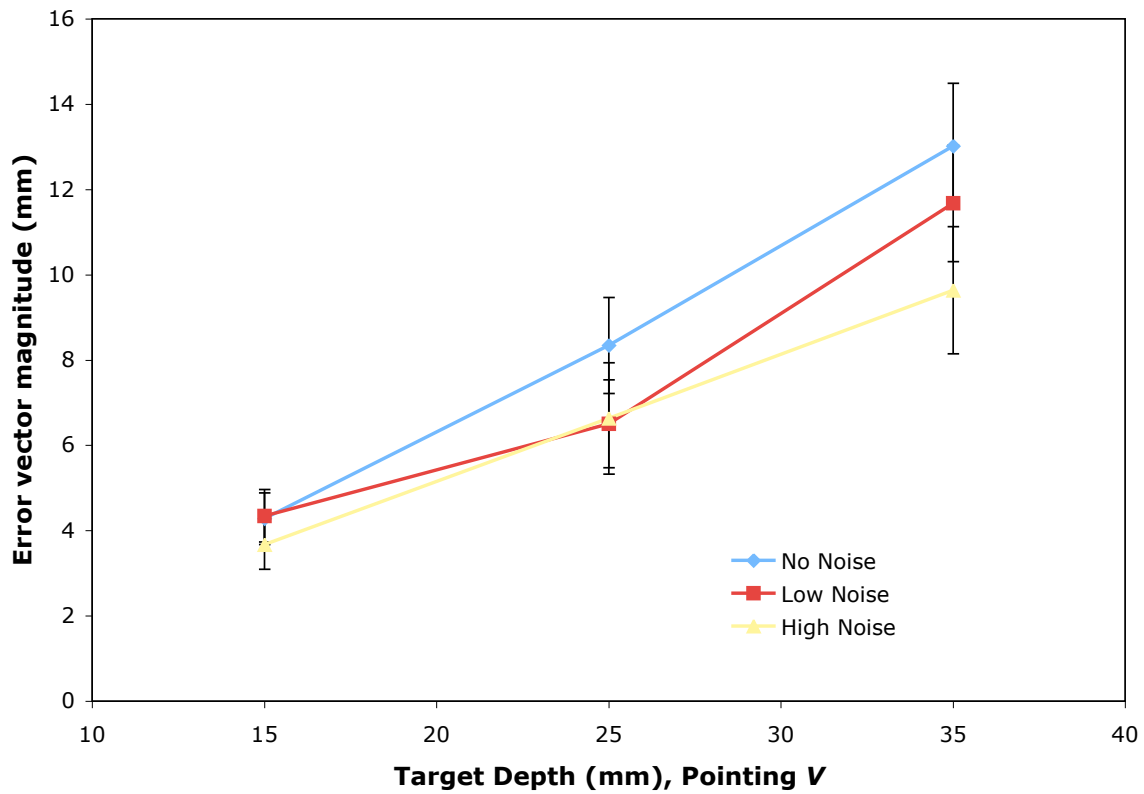


Figure 6.1: Pointing, error vector magnitude, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth was significant ($F(2, 16) = 48.002, p < 0.001$)
- ★ The main effect of background noise was significant ($F(2, 16) = 17.309, p < 0.001$)
- ★ There was significant interaction between target depth and background noise ($F(4, 32) = 3.556, p < 0.05$)

6.4.2 Error along V_x axis

Depth	No Noise	Low Noise	High Noise
15	3.4 (2.4)	3.1 (3.0)	2.3 (2.5)
25	8.0 (3.4)	5.5 (3.9)	5.7 (4.2)
35	12.8 (4.5)	11.4 (4.1)	8.8 (5.4)

Table 6.2: Pointing, V_x error, mean (stdev), values in mm.

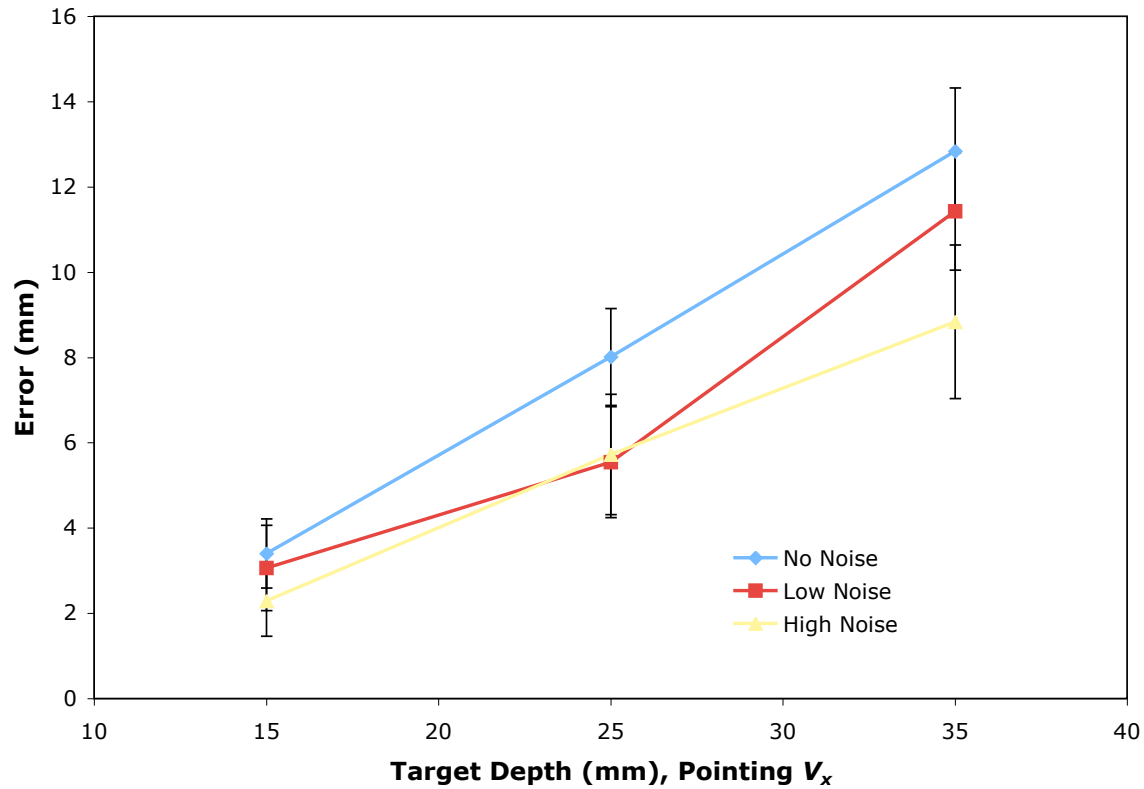


Figure 6.2: Pointing, V_x error, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth was significant ($F(2, 16) = 60.788, p < 0.001$)
- ★ The main effect of background noise was significant ($F(2, 16) = 15.835, p < 0.001$)
- ★ There was significant interaction between target depth and background noise ($F(4, 32) = 3.396, p < 0.05$)

6.4.3 Error along V_y axis

Depth	No Noise	Low Noise	High Noise
15	1.1 (0.3)	1.2 (0.7)	1.1 (0.5)
25	1.6 (1.1)	1.6 (1.3)	1.0 (1.6)
35	1.1 (1.1)	0.5 (1.5)	0.4 (0.9)

Table 6.3: Pointing, V_y error, mean (stdev), values in mm.

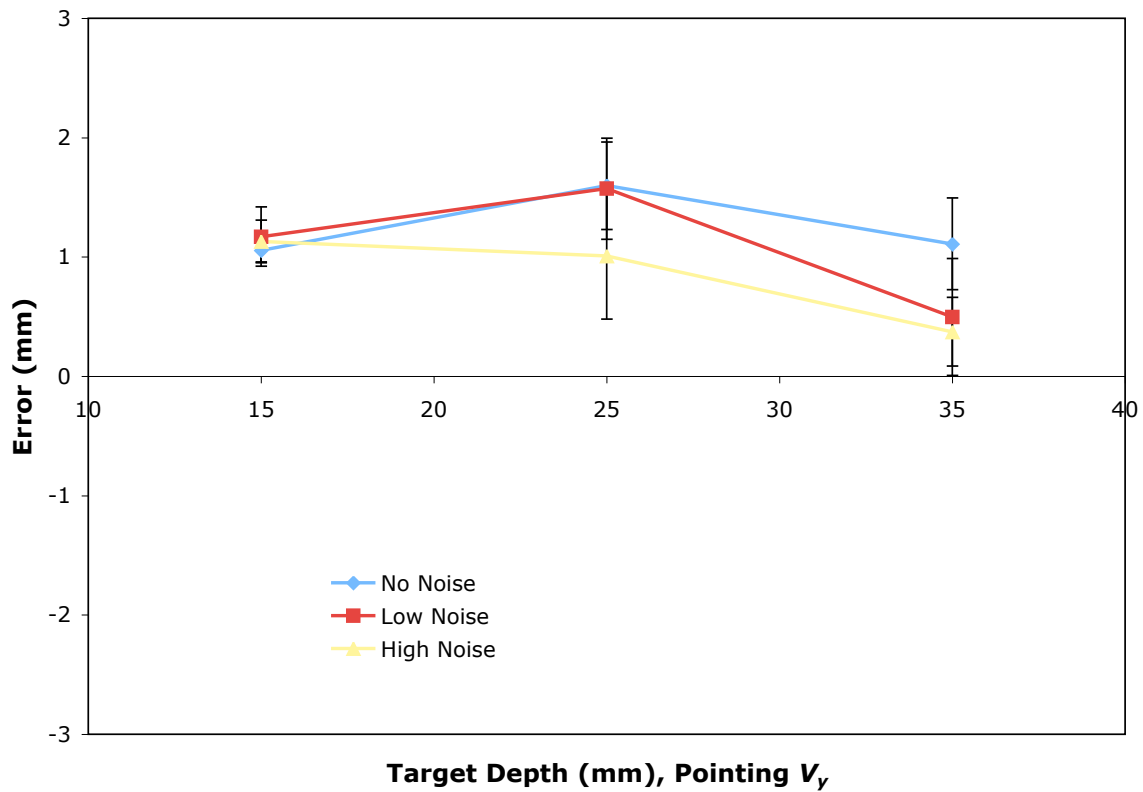


Figure 6.3: Pointing, V_y error, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth was significant ($F(2, 16) = 5.393, p < 0.05$)
- ★ The main effect of background noise was not significant ($F(2, 16) = 2.351, p > 0.1$)
- ★ There was no interaction between target depth and background noise ($F(4, 32) = 0.570, p > 0.6$)

6.4.4 Error along V_z axis

Depth	No Noise	Low Noise	High Noise
15	-0.1 (0.2)	-0.1 (0.4)	-0.1 (0.4)
25	-0.1 (0.2)	-0.3 (0.4)	-0.3 (0.1)
35	-0.2 (0.6)	-0.2 (0.3)	0.0 (0.5)

Table 6.4: Pointing, V_z error, mean (stdev), values in mm.

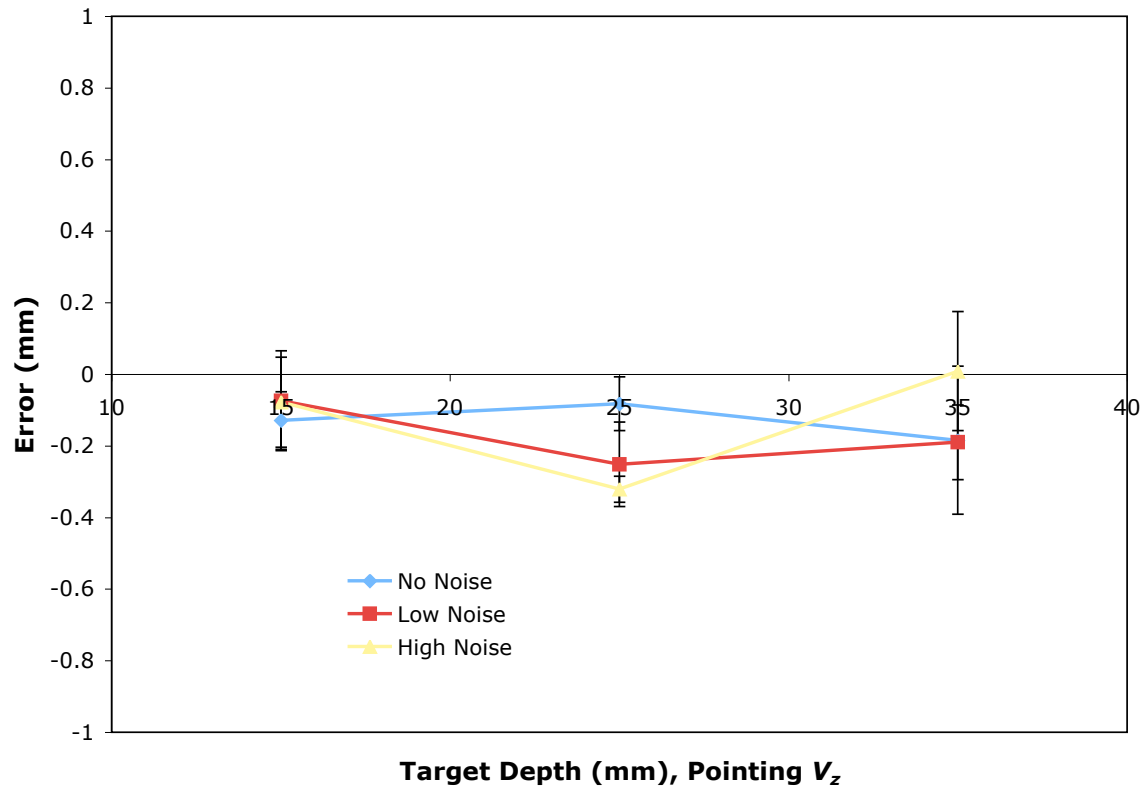


Figure 6.4: Pointing, V_z error, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth was not significant ($F(2, 16) = 0.831, p > 0.4$)
- ★ The main effect of background noise was not significant ($F(2, 16) = 0.183, p > 0.8$)
- ★ There was no interaction between target depth and background noise ($F(4, 32) = 0.733, p > 0.5$)

6.5 Needle insertion results

6.5.1 Error vector magnitude

Depth	No Noise	Low Noise	High Noise
15	4.3 (1.4)	4.3 (1.3)	3.9 (1.2)
25	8.2 (2.4)	6.7 (2.0)	6.7 (2.3)
35	11.3 (4.2)	11.1 (4.3)	8.8 (3.7)

Table 6.5: Insertion, error vector magnitude, mean (stdev), values in mm.

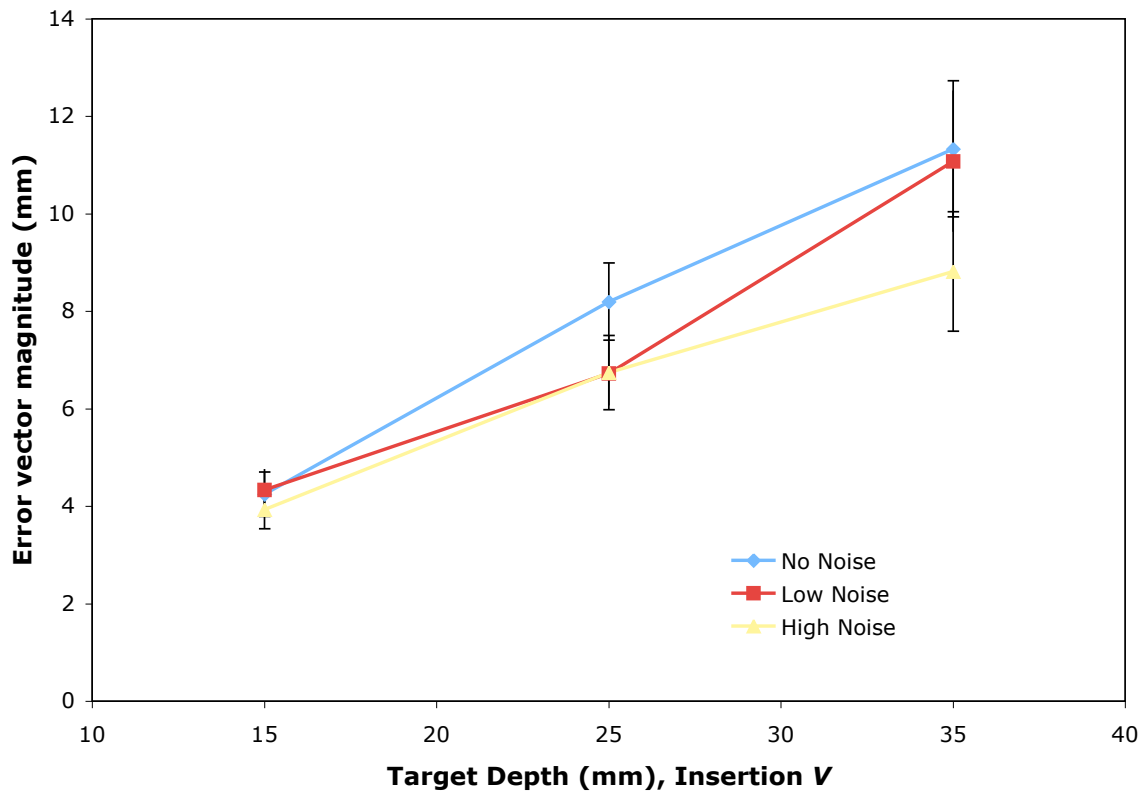


Figure 6.5: Insertion, error vector magnitude, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth was significant ($F(2, 16) = 28.151, p < 0.001$)
- ★ The main effect of background noise was significant ($F(2, 16) = 14.649, p < 0.001$)
- ★ There was significant interaction between target depth and background noise ($F(4, 32) = 3.929, p < 0.05$)

6.5.2 Error along V_x axis

Depth	No Noise	Low Noise	High Noise
15	3.3 (1.8)	3.2 (2.4)	2.8 (1.8)
25	7.8 (2.2)	6.2 (2.1)	6.1 (2.9)
35	11.1 (4.3)	10.6 (4.2)	8.3 (3.9)

Table 6.6: Insertion, V_x error, mean (stdev), values in mm.

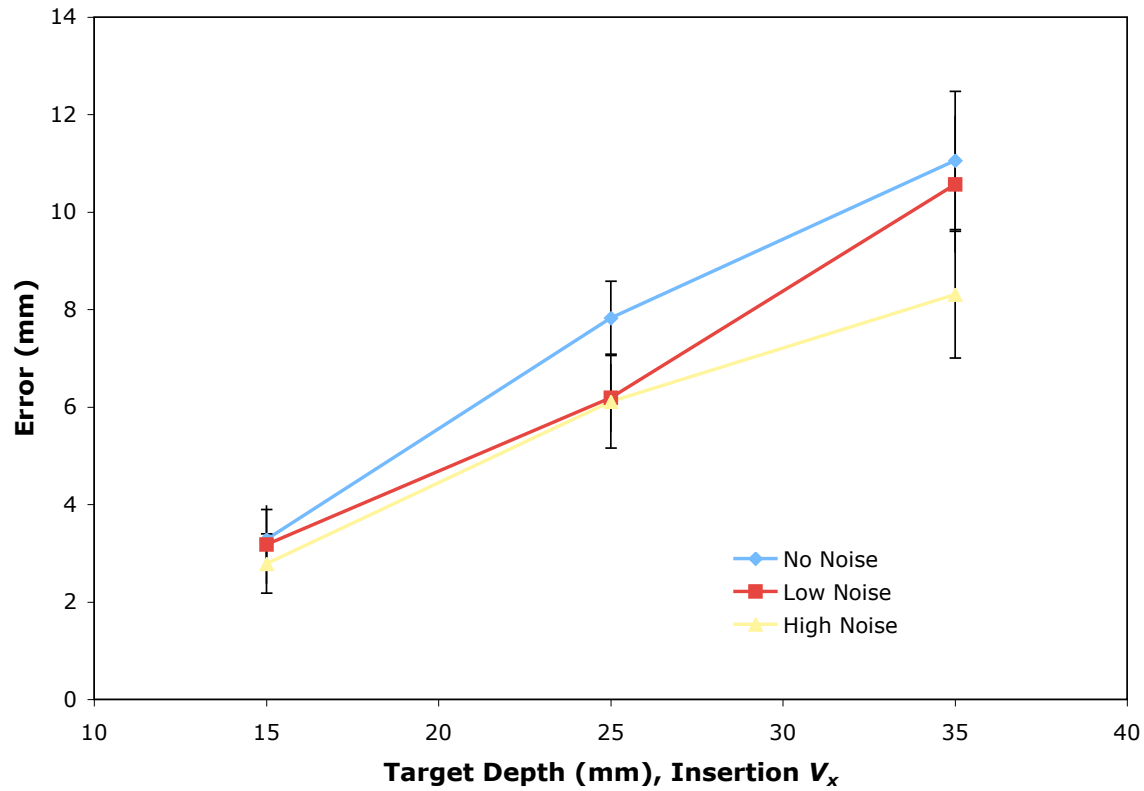


Figure 6.6: Insertion, V_x error, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth was significant ($F(2, 16) = 33.484, p < 0.001$)
- ★ The main effect of background noise was significant ($F(2, 16) = 14.154, p < 0.001$)
- ★ The interaction between target depth and background noise was barely not significant ($F(4, 32) = 2.437, p = 0.067$)

6.5.3 Error along V_y axis

Depth	No Noise	Low Noise	High Noise
15	1.5 (0.4)	1.3 (0.7)	1.4 (0.6)
25	1.9 (1.0)	1.7 (1.1)	1.5 (0.5)
35	1.2 (1.4)	1.4 (1.3)	1.3 (1.1)

Table 6.7: Insertion, V_y error, mean (stdev), values in mm.

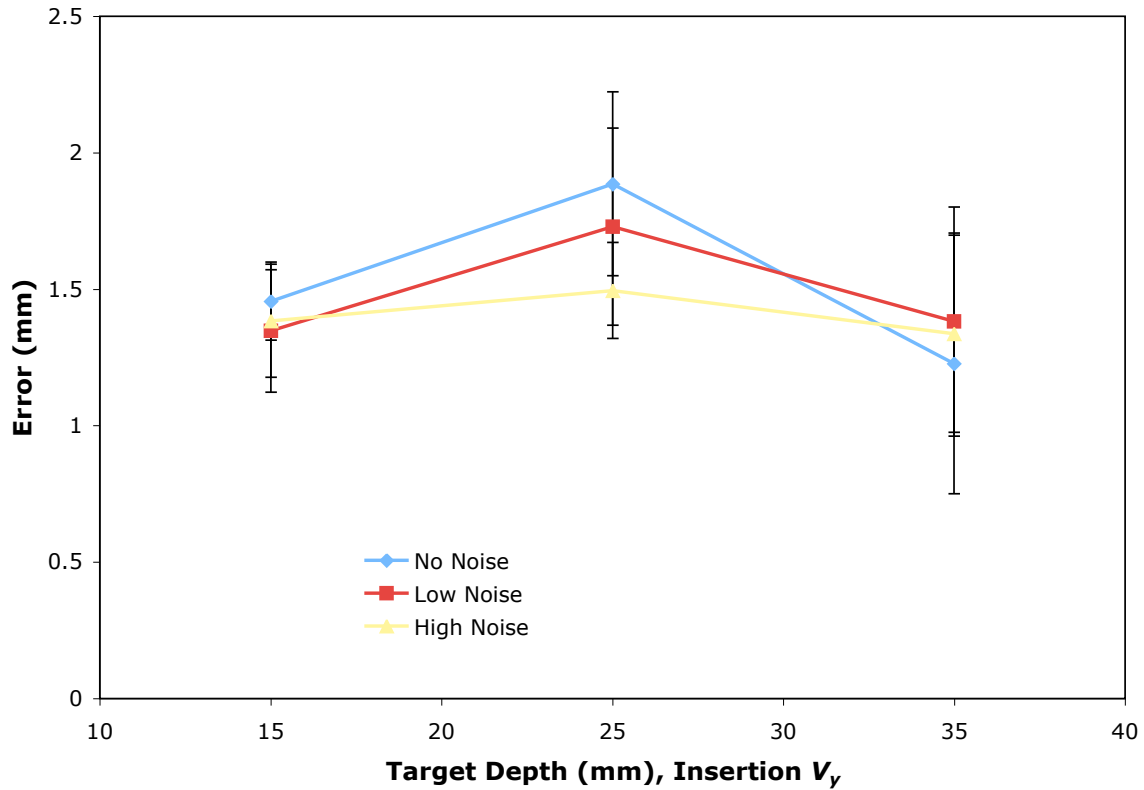


Figure 6.7: Insertion, V_y error, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth was not significant ($F(2, 16) = 1.617, p > 0.2$)
- ★ The main effect of background noise was not significant ($F(2, 16) = 0.280, p > 0.7$)
- ★ There was no interaction between target depth and background noise ($F(4, 32) = 0.235, p > 0.9$)

6.5.4 Error along V_z axis

Depth	No Noise	Low Noise	High Noise
15	-0.1 (0.3)	-0.1 (0.5)	-0.2 (0.4)
25	-0.4 (0.4)	-0.3 (0.4)	-0.3 (0.3)
35	-0.4 (0.6)	0.1 (1.6)	-0.4 (0.8)

Table 6.8: Insertion, V_z error, mean (stdev), values in mm.

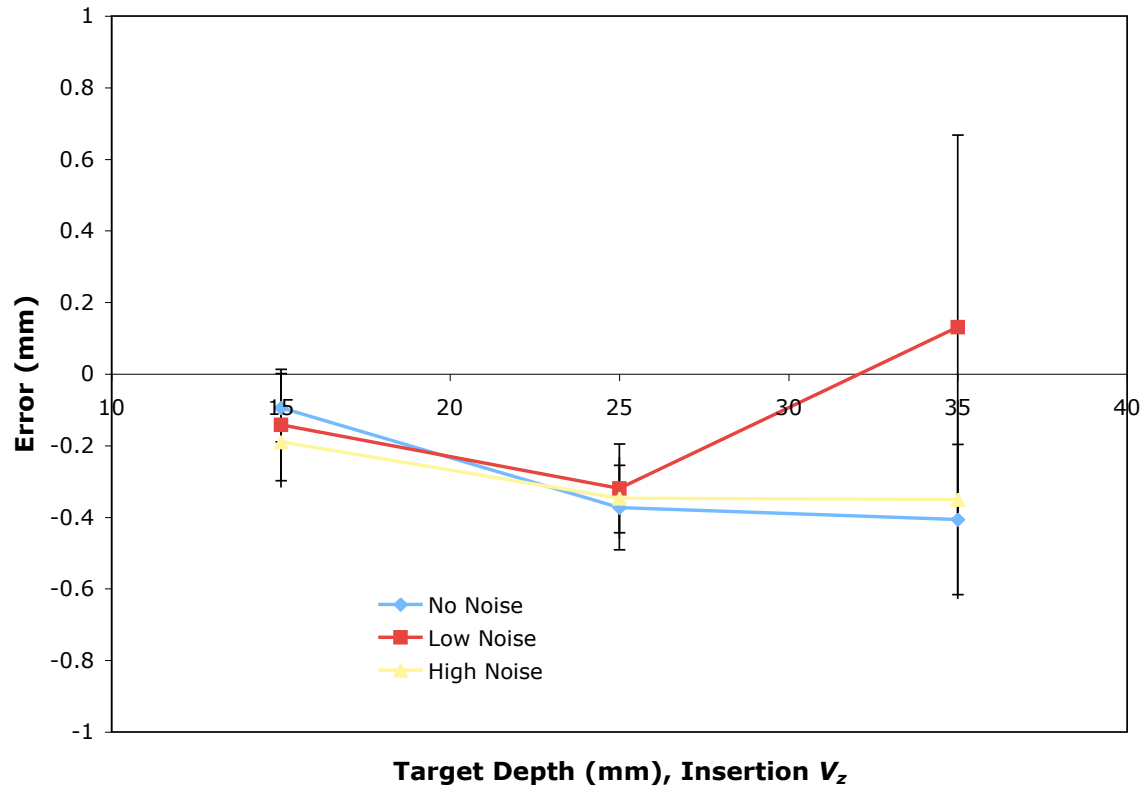


Figure 6.8: Insertion, V_z error, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth was not significant ($F(2, 16) = 0.370, p > 0.6$)
- ★ The main effect of background noise was not significant ($F(2, 16) = 0.707, p > 0.5$)
- ★ There was no interaction between target depth and background noise ($F(4, 32) = 0.642, p > 0.6$)

6.6 Tip insertion results

6.6.1 Tip insertion magnitude

Depth	No Noise	Low Noise	High Noise
15	9.4 (4.5)	8.9 (3.9)	11.1 (3.6)
25	10.4 (2.7)	8.4 (3.2)	8.7 (2.3)
35	11.0 (5.1)	11.9 (5.9)	8.7 (3.8)

Table 6.9: Tip insertion, magnitude error, mean (stdev), values in mm.

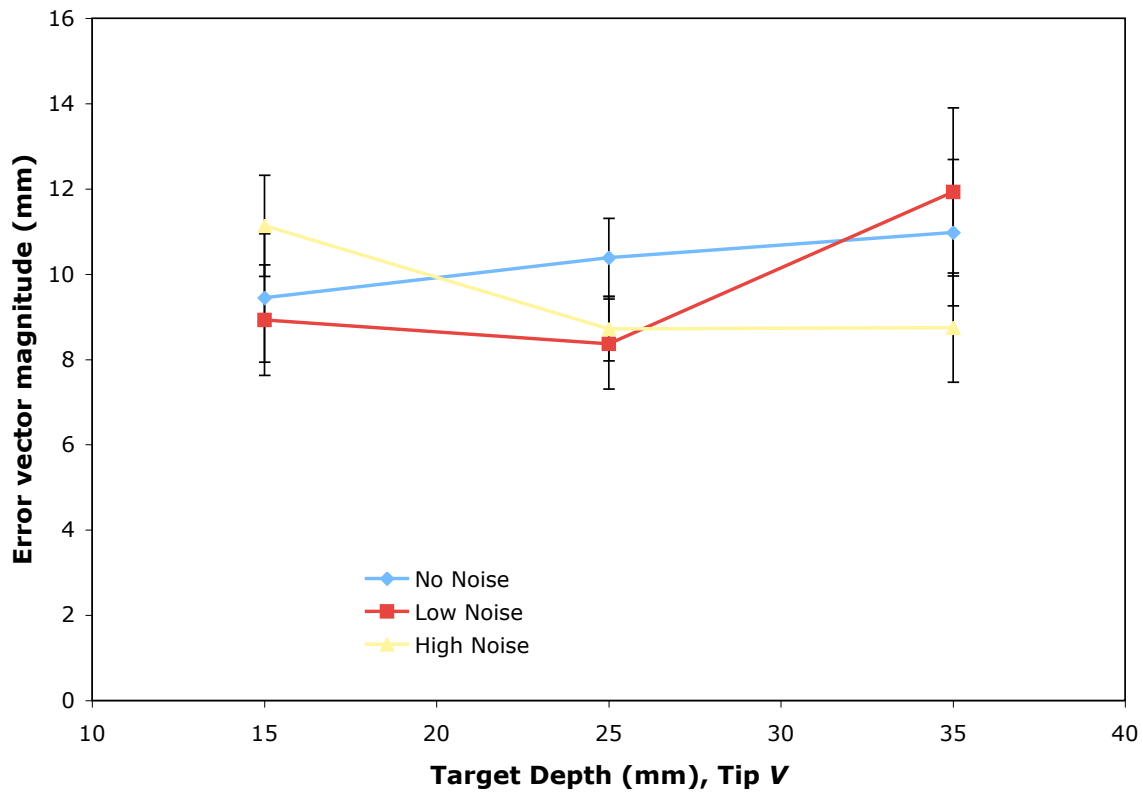


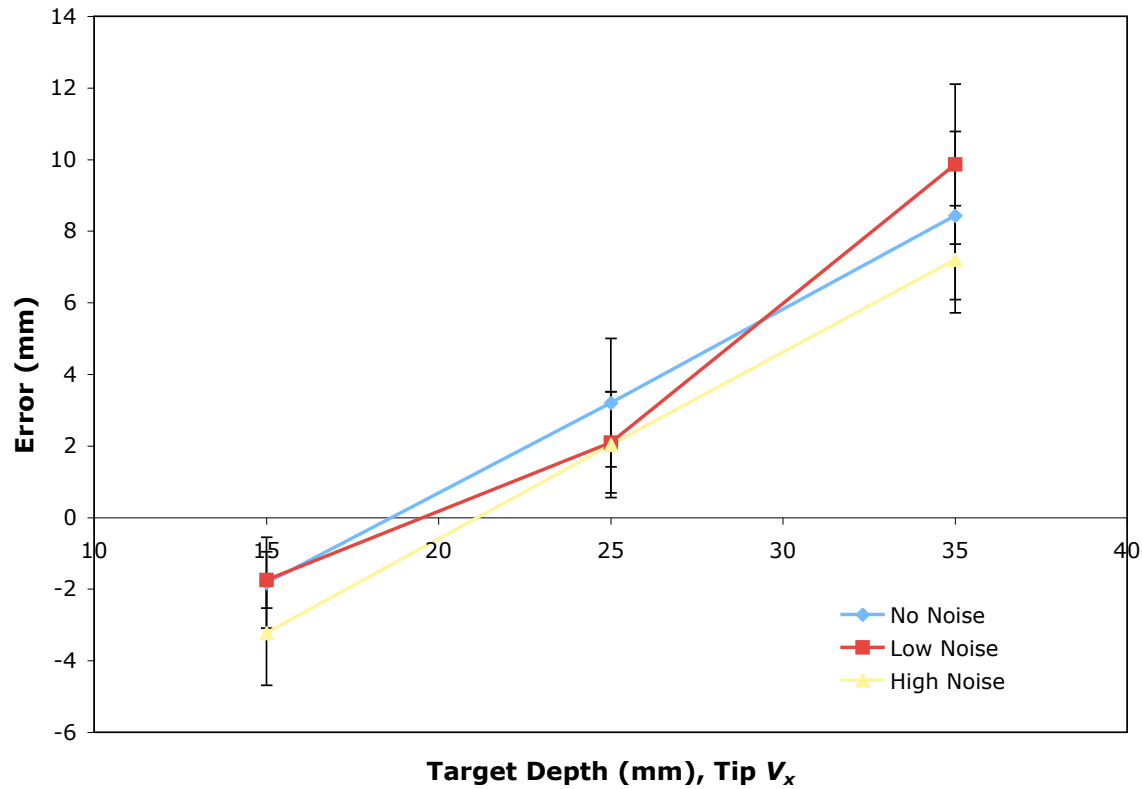
Figure 6.9: Tip insertion, magnitude error, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth was not significant ($F(2, 16) = 0.392, p > 0.6$)
- ★ The main effect of background noise was not significant ($F(2, 16) = 2.157, p > 0.1$)
- ★ There was significant interaction between target depth and background noise ($F(4, 32) = 6.204, p < 0.01$)

6.6.2 Error along V_x axis

Depth	No Noise	Low Noise	High Noise
15	-1.8 (3.8)	-1.7 (2.4)	-3.2 (4.4)
25	3.2 (5.4)	2.1 (4.2)	2.0 (4.4)
35	8.4 (7.0)	9.9 (6.7)	7.2 (4.5)

Table 6.10: Tip insertion, V_x error, mean (stdev), values in mm.Figure 6.10: Tip insertion, V_x error, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth was significant ($F(2, 16) = 27.089, p < 0.001$)
- ★ The main effect of background noise was significant ($F(2, 16) = 7.785, p < 0.01$)
- ★ There was no significant interaction between target depth and background noise ($F(4, 32) = 0.943, p > 0.4$)

6.6.3 Error along V_y axis

Depth	No Noise	Low Noise	High Noise
15	1.4 (0.9)	1.1 (0.6)	1.1 (0.8)
25	1.6 (1.7)	1.4 (1.6)	1.0 (1.4)
35	1.3 (1.4)	1.2 (1.5)	1.2 (1.3)

Table 6.11: Tip insertion, V_y error, mean (stdev), values in mm.

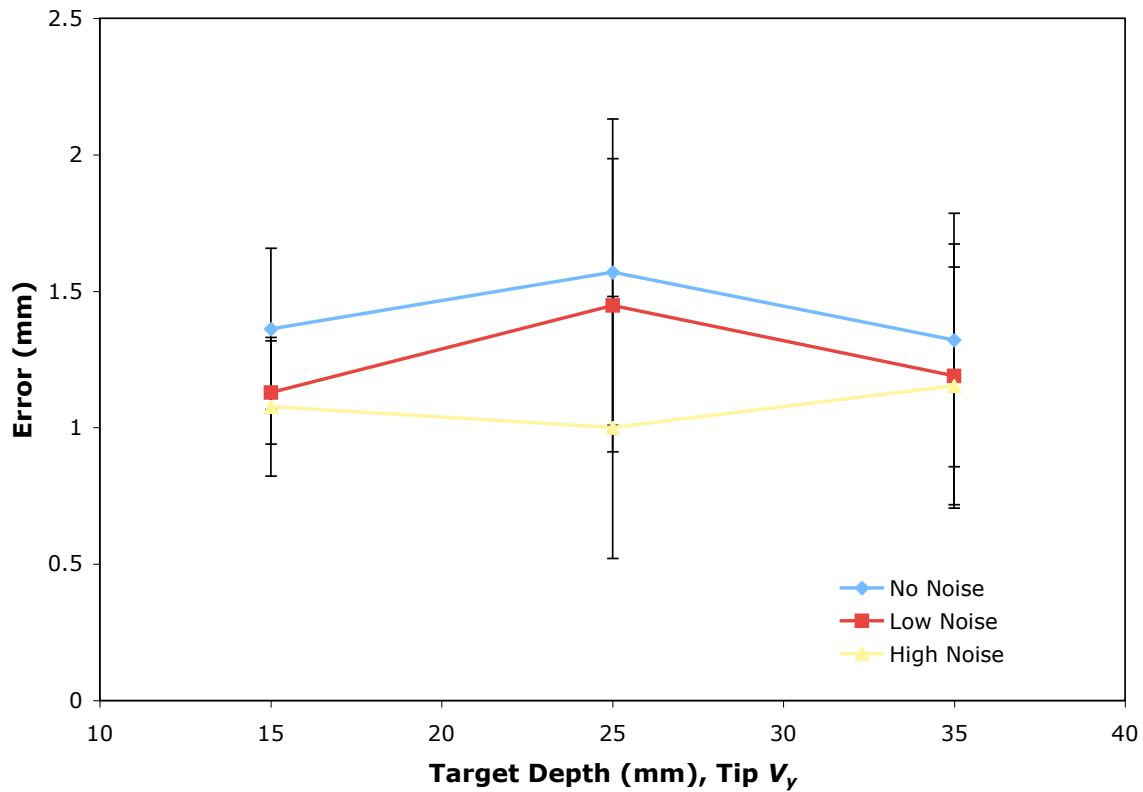


Figure 6.11: Tip insertion, V_y error, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth was not significant ($F(2, 16) = 0.203, p > 0.8$)
- ★ The main effect of background noise was not significant ($F(2, 16) = 1.330, p > 0.2$)
- ★ There was no interaction between target depth and background noise ($F(4, 32) = 0.124, p > 0.9$)

6.6.4 Error along V_z axis

Depth	No Noise	Low Noise	High Noise
15	7.5 (5.1)	6.9 (5.4)	8.8 (4.1)
25	4.5 (5.5)	4.3 (4.4)	4.2 (4.1)
35	1.7 (3.7)	0.2 (3.7)	0.6 (3.0)

Table 6.12: Tip insertion, V_z error, mean (stdev), values in mm.

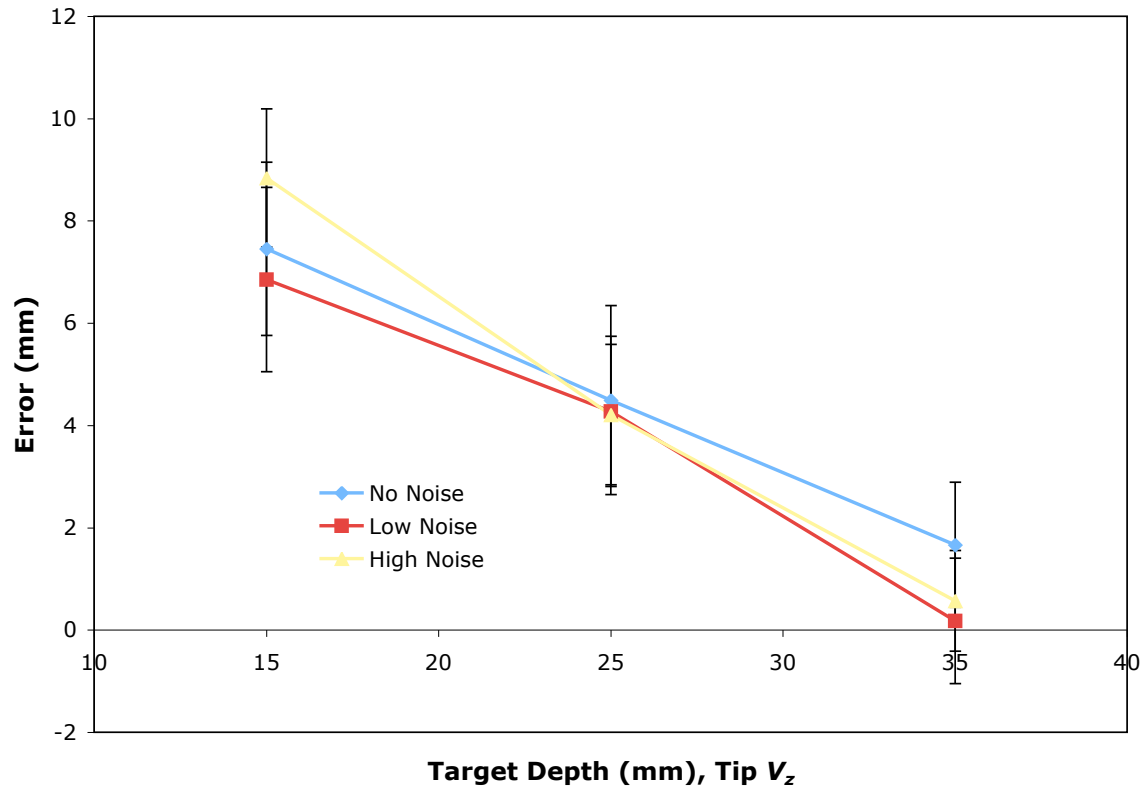


Figure 6.12: Tip insertion, V_z error, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth was significant ($F(2, 16) = 27.837, p < 0.001$)
- ★ The main effect of background noise was barely not significant ($F(2, 16) = 2.794, p = 0.091$)
- ★ There was no interaction between target depth and background noise ($F(4, 32) = 1.103, p > 0.3$)

6.7 Summary of Experiment 2 results

In this summary, references to “pointing” and “insertion” refer to using the intersection of the needle axis with the virtual image plane as the subject’s indication of perceived depth. References to “tip insertion” refer to the actual location of the needle tip during insertion.

The take-home message from Experiment 2 is as follows:

- ★ As I showed in Experiment 1, target depth has a significant effect on error in V_x in both the pointing and insertion, with subjects making greater foreshortening errors at deeper depths.
- ★ The effect of background noise on perceived depth in V_x was significant for both pointing and insertion. Subjects made smaller errors with the high background noise as compared to the no background noise. In addition, there was significant interaction between background noise and target depth during both pointing and insertion. The effect of background noise is greater with deep targets, and less pronounced with shallow targets.

I hypothesize that the interaction between depth and background noise appeared in this experiment, and not in Experiment 1, as a result of the no-feedback task being more difficult than the red bar task. Since subjects are free to move the needle around inside the container before deciding on a final insertion point, the red bar provides feedback about the location of the image plane that may be helpful in gauging depth. This does not, however, explain why this interaction was also present during pointing, which is an identical task to Experiment 1.

- ★ The effect of depth on V_y error during pointing was significant, but not during insertion. The size of this effect is quite small, with subjects performing slightly better on shallow targets during pointing. No significant effects of background noise were found in V_y during pointing or insertion.
- ★ No significant effects of depth or background noise were found in V_z during pointing or insertion. Subjects are quite good at slicing through the center of the target regardless of its depth or the amount of background noise.
- ★ The effect of target depth on penetration of the needle through the image plane during tip insertion was significant. Subjects penetrated the image by much smaller amounts with deep targets. This effect is most likely due to the inability of subjects to substantially penetrate through deep targets due to the physical constraints of the needle. In other words, subjects were able to overshoot shallow targets, but “bottomed out” with the needle tool when hitting deep targets. Given that there is a large effect of target depth on perceived depth in the other direction – i.e., subjects do worse with deep targets – I conclude that the lack of red bar feedback rendered subjects essentially incapable of judging overpenetration of the image plane.

Chapter 7

Experiment 3: A comparison of SF and CUS guidance

7.1 Experimental protocol

7.1.1 Experiment design

This experiment compared the ability of subjects to hit targets with the VTR system operating in both SF and CUS modes. Like Experiments 1 and 2, targets were presented at 3 depths and with varying levels of background noise. This study had two primary purposes:

1. Compare SF and CUS guidance.
2. Examine the effect of target depth and background noise on each guidance method.

A $3 \text{ (depth)} \times 2 \text{ (background)} \times 2 \text{ (guidance mode)}$ 3-way design was implemented with 2 replications. All targets in this experiment were 1.5 mm radius white spheres, located at 15 mm, 25 mm, and 35 mm below the surface of the image container (I_z). Location in the horizontal plane (the I_{xy} plane) was randomized for each target. The background of target image was black, with overlays of either 0 or 5000 randomly generated white points (referred to as “no” and “high” noise); the number of noise levels was reduced from the prior experiments because of the addition of guidance mode (CUS vs. SF) as an experiment parameter, and the desire to keep testing times under an hour.

Switching between SF and CUS guidance required a change in hardware connections, so this experiment was broken into 2 blocks by guidance method and subjects were assigned to use either SF or CUS in the first block to yield a counterbalanced design. Within each block, the subject was presented with 20 total targets, grouped into two repeating sub-blocks of 10 targets (3 depths x 2 noise levels + 4 dummy targets). Target order was randomized within each sub-block.

14 naive subjects participated in this experiment, 7 men and 7 women.

Appendix D lists the parameters used for each trial.

7.1.2 Task

Subjects began each trial with the needle held outside of the image container and the mock transducer tip resting in a marked location on the edge of the image container where the image would not be visible between trials. Subjects were asked to first indicate the location of a target by pointing, holding the needle tip on the surface of the image container at a location of their choosing and aiming the long axis of the needle at the target. Subjects were instructed to verbally report when their aim was correct and then hold the needle still while a single frame of data was stored. Next, the subject was instructed to hit the target with the tip of the needle, stopping on the exact center location of the target. Once the subject verbally indicated success, another frame of data was stored, and the subject withdrew the needle and moved the mock transducer tip back to the rest position.

7.1.3 Training

Because this experiment used both SF and CUS guidance, subjects required training with each guidance mode. A demonstration of each guidance method was provided prior to the appropriate block and the subject was allowed to practice on 1 image where they saw a white dot to indicate the needle-plane intersection and 5 images where they saw the red bar. I verbally corrected the subject's aim until their error was less than 2.0 mm. Identical training sets were used for both the SF and CUS.

7.2 Key to reference frames

As described in greater detail in Chapter 4, the following coordinate frames are used when analyzing the results from this experiment.

★ The Sonic Flashlight virtual image frame V

- V_x is “depth”, or distance along the extended centerline of the mock transducer, in the SF image plane.
- V_y is left or right in the SF image plane.
- V_z is normal to the SF image plane.

★ The target image frame I

- I_x and I_y may be arbitrarily oriented depending on the placement of the image container.
- I_z is defined as the earth's gravity vector. In other words, a displacement in the $+I_z$ direction is down.

If the user is holding the SF perpendicular to the lid of the image container, then $+V_x = -I_z$. When these values are not equal, the user is holding the SF at an angle relative to gravity. As described previously, both of these axes are approximate measures of perceived depth, thus one may interpret an error in either $+V_x$ or $-I_z$ as an indication of foreshortening of the target depth relative to the user's eyes.

7.3 Statistics

I do not present the results from I_z axis in this experiment (although it was collected and analyzed), since as Experiment 1 demonstrates, it is virtually identical to the V_x axis. For the sake of brevity, I also do not present the V_z results, since it has already been demonstrated that subjects are quite good at placing the image plane through the center of the target (this is true of CUS as well).

An additional note regarding the figures that follow. Because of the visual complexity of the plots (2 devices and 2 noise levels), I provide separate plots for the high and no noise conditions in addition to a combined plot for axes which show significant results. All three figures are formatted identically, and are derived from the same means table presented at the beginning of each section. This makes it easier to compare the SF to CUS under high and no noise conditions.

7.4 Needle pointing results

7.4.1 Error vector magnitude

Depth	No Noise, CUS	No Noise, SF	High Noise, CUS	High Noise, SF
15	5.7 (2.7)	7.8 (5.9)	8.6 (4.0)	7.2 (3.9)
25	12.2 (4.1)	6.8 (2.9)	7.5 (3.2)	7.4 (3.1)
35	12.3 (6.4)	10.3 (5.7)	12.0 (6.6)	8.1 (5.1)

Table 7.1: Pointing, error vector magnitude, mean (stdev), values in mm.

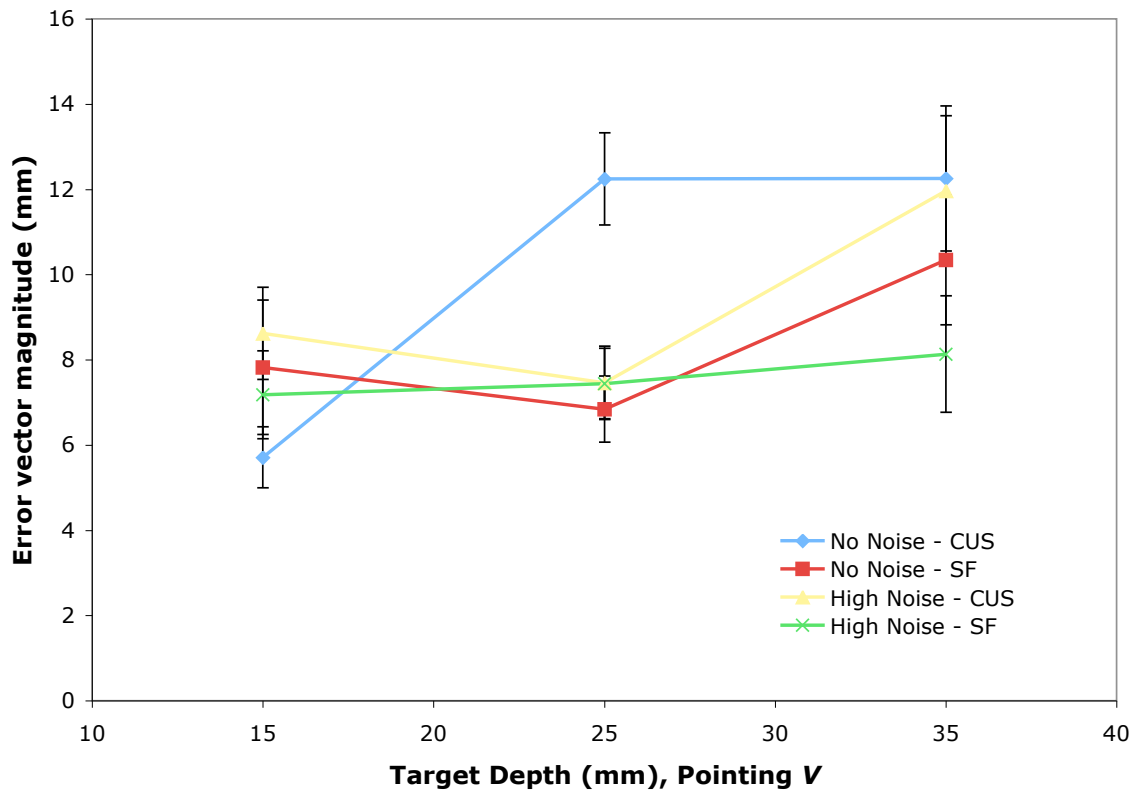


Figure 7.1: Pointing, error vector magnitude, mean \pm standard error.

Summary of statistics:

1. The main effect of noise is significant ($F(2, 26) = 5.525, p < 0.05$)
2. The main effect of device is significant ($F(2, 26) = 5.179, p < 0.05$)
3. The interaction between target depth and background noise is significant ($F(2, 26) = 6.110, p < 0.01$)
4. The interaction between target depth and device is significant ($F(2, 26) = 3.886, p < 0.05$)

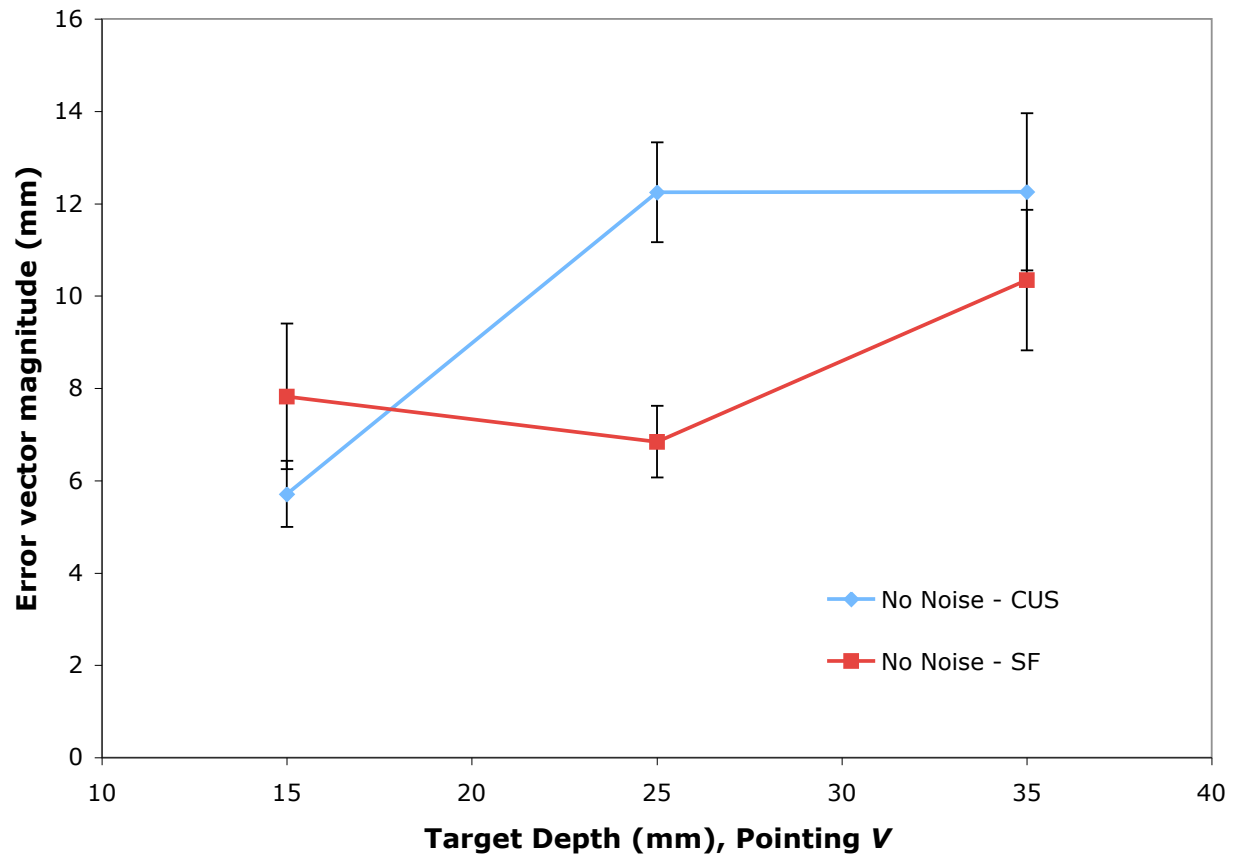


Figure 7.2: Pointing, error vector magnitude, no noise condition, mean \pm standard error.

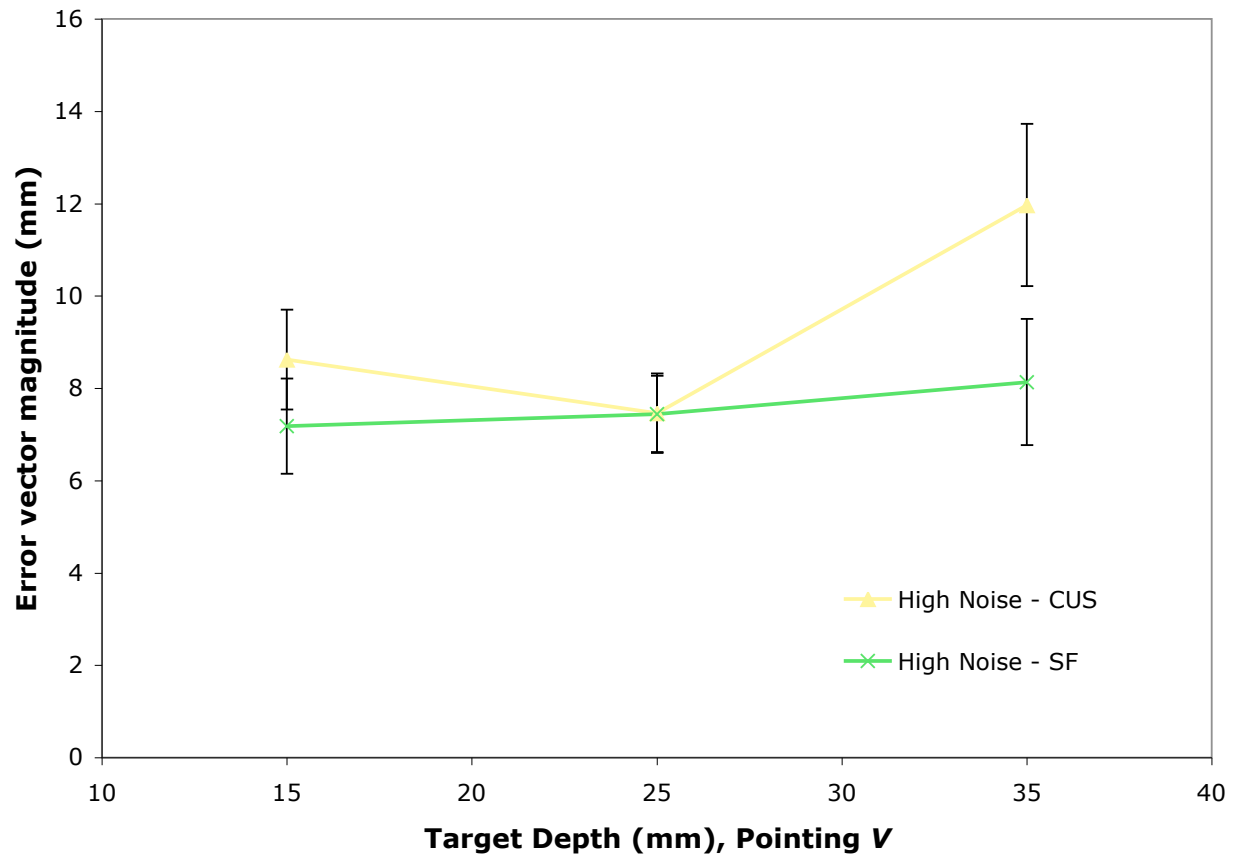
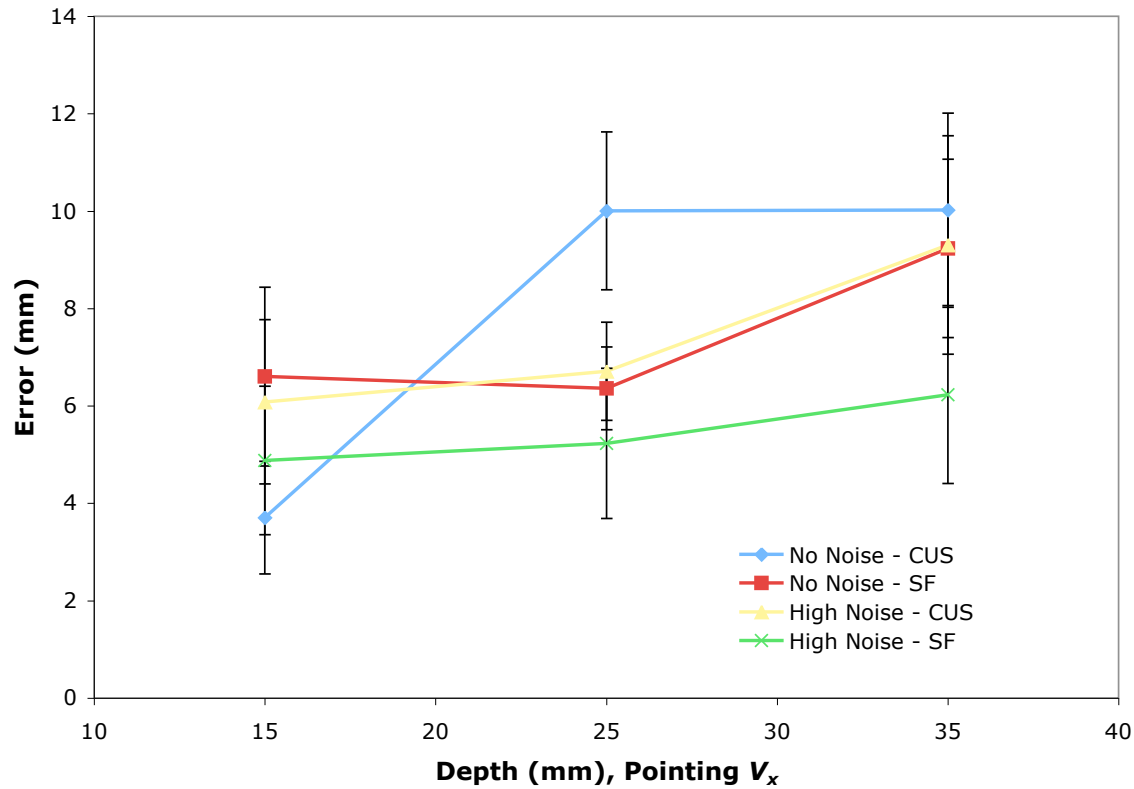


Figure 7.3: Pointing, error vector magnitude, high noise condition, mean \pm standard error.

7.4.2 Error along V_x axis

Depth	No Noise, CUS	No Noise, SF	High Noise, CUS	High Noise, SF
15	3.7 (4.3)	6.6 (6.9)	6.1 (6.3)	4.9 (5.7)
25	10.0 (6.1)	6.4 (3.2)	6.7 (3.8)	5.2 (5.8)
35	10.0 (7.5)	9.2 (6.8)	9.3 (8.4)	6.2 (6.8)

Table 7.2: Pointing, V_x error, mean (stdev), values in mm.Figure 7.4: Pointing, V_x error, mean \pm standard error.

Summary of statistics:

- ★ The main effect of background noise is significant ($F(2, 26) = 8.376, p < 0.05$)
- ★ The interaction between depth and device is significant ($F(2, 26) = 3.862, p < 0.05$)
- ★ The interaction between background noise and device is barely not significant ($F(2, 26) = 4.514, p = 0.053$)

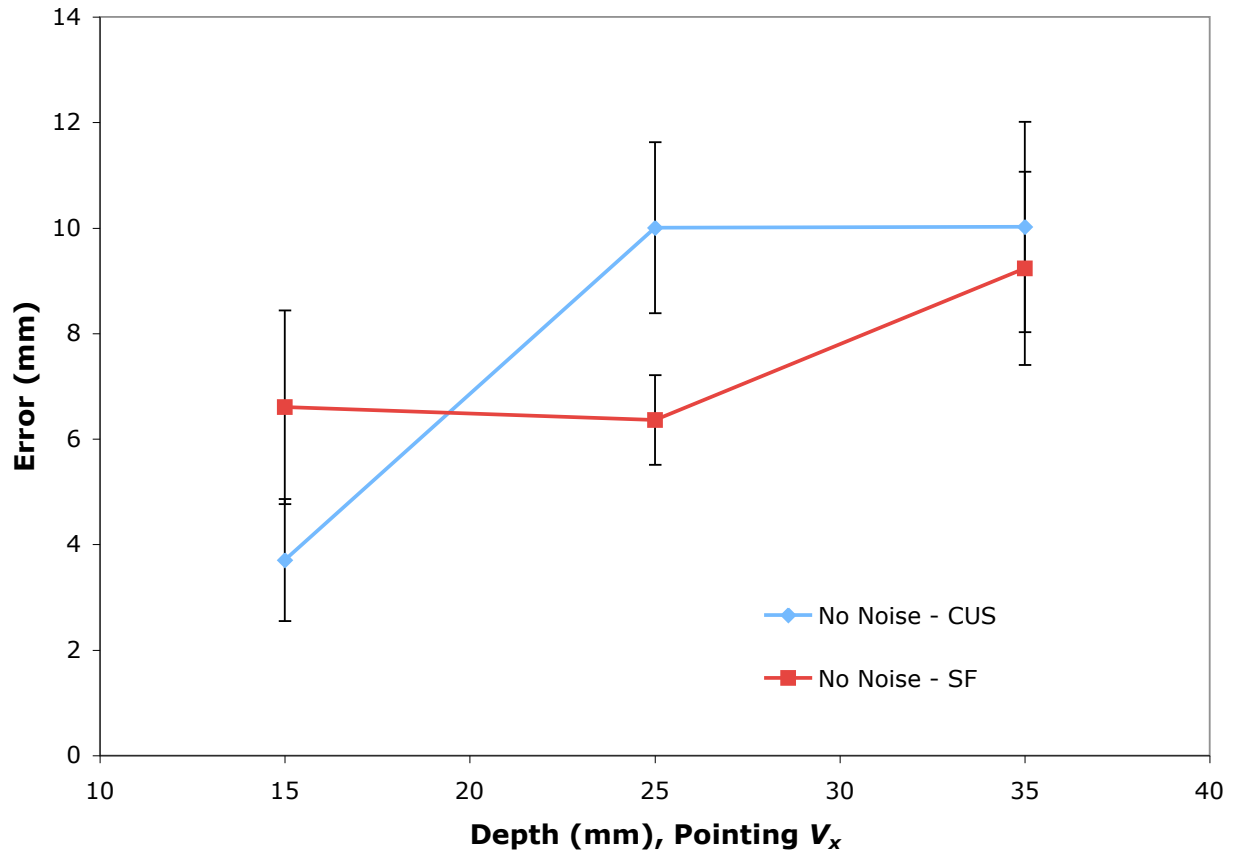


Figure 7.5: Pointing, V_x error, no noise condition, mean \pm standard error.

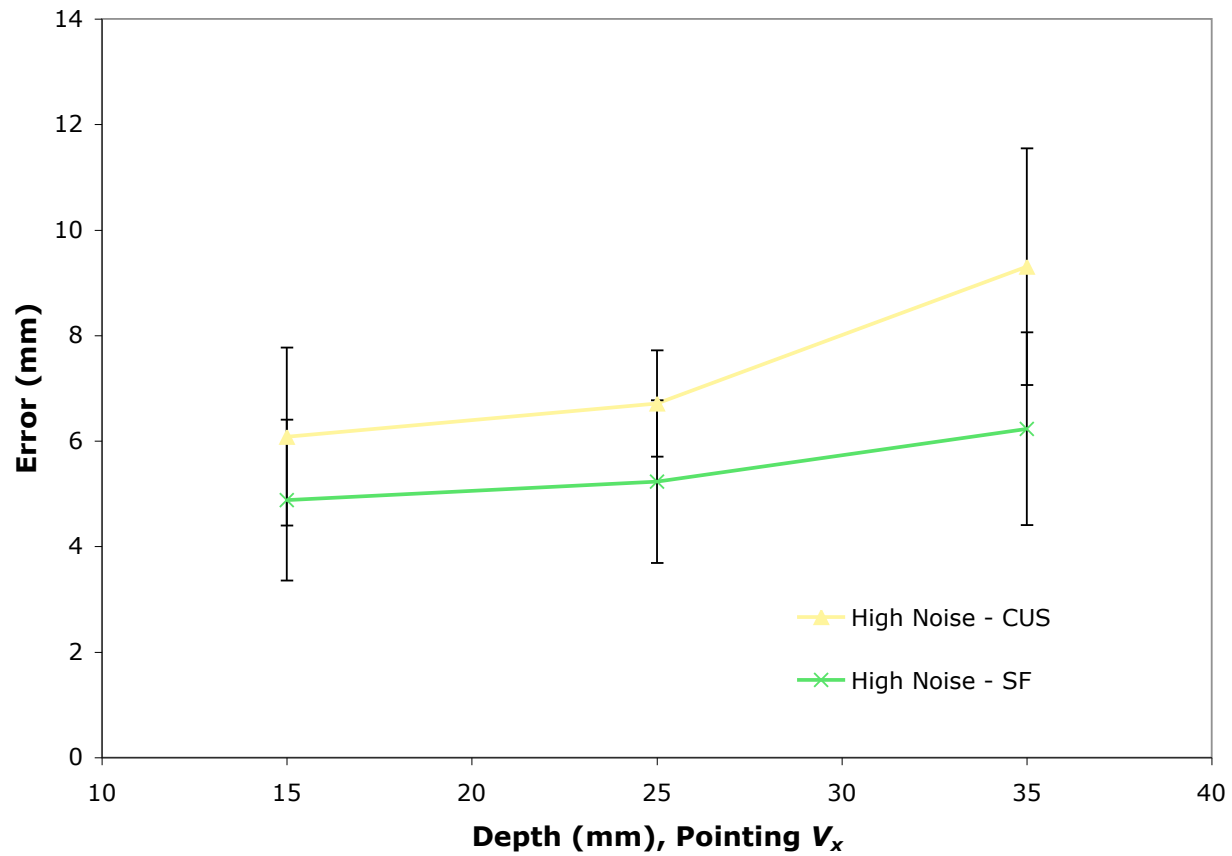


Figure 7.6: Pointing, V_x error, high noise condition, mean \pm standard error.

7.4.3 Error along V_y axis

Depth	No Noise, CUS	No Noise, SF	High Noise, CUS	High Noise, SF
15	-0.1 (1.9)	0.3 (1.9)	-0.3 (3.1)	0.6 (1.3)
25	-0.1 (5.1)	0.3 (1.5)	0.7 (2.3)	0.0 (2.0)
35	-0.4 (5.8)	0.4 (2.4)	0.0 (4.6)	0.2 (2.2)

Table 7.3: Pointing, V_y error, mean (stdev), values in mm.

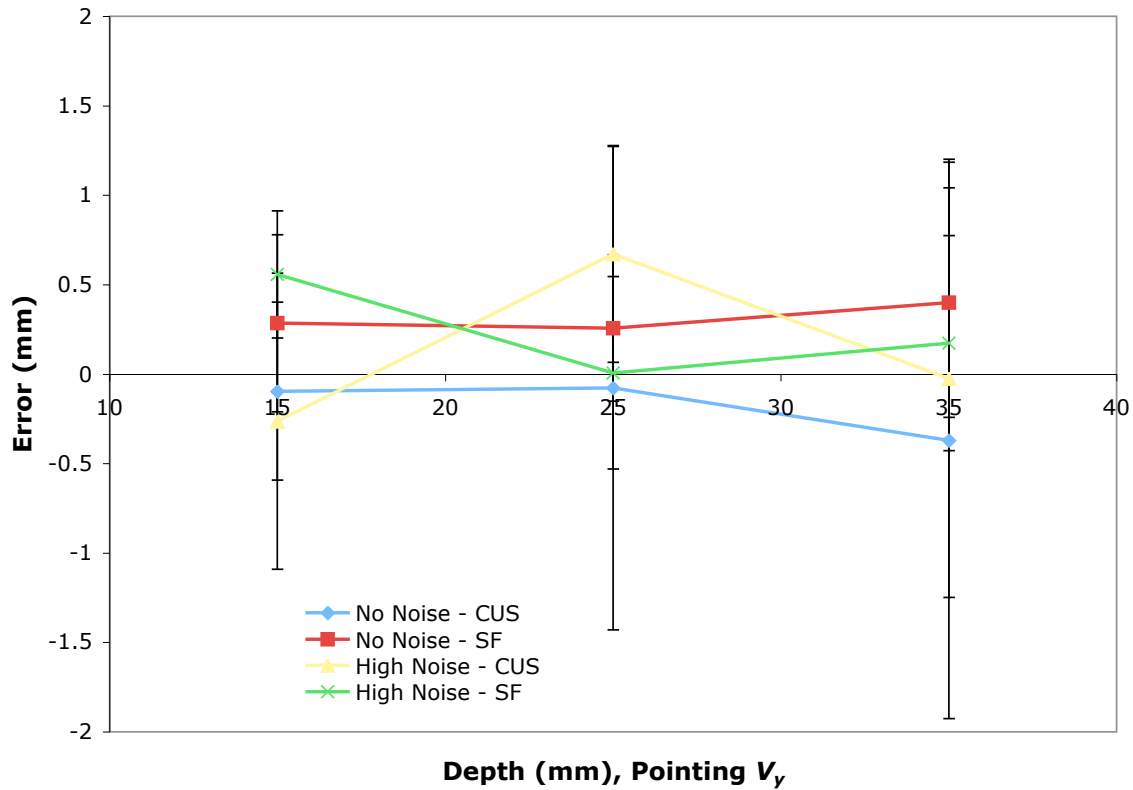


Figure 7.7: Pointing, V_y error, mean \pm standard error.

Summary of statistics:

★ No main effects or interactions are significant.

7.5 Needle insertion results

7.5.1 Error vector magnitude

Depth	No Noise, CUS	No Noise, SF	High Noise, CUS	High Noise, SF
15	6.2 (3.1)	7.3 (5.2)	7.1 (2.7)	6.4 (3.4)
25	11.3 (2.1)	6.4 (2.7)	7.3 (3.9)	6.2 (3.3)
35	10.5 (5.5)	9.9 (5.7)	11.6 (6.6)	7.5 (4.4)

Table 7.4: Insertion, error vector magnitude, mean (stdev), values in mm.

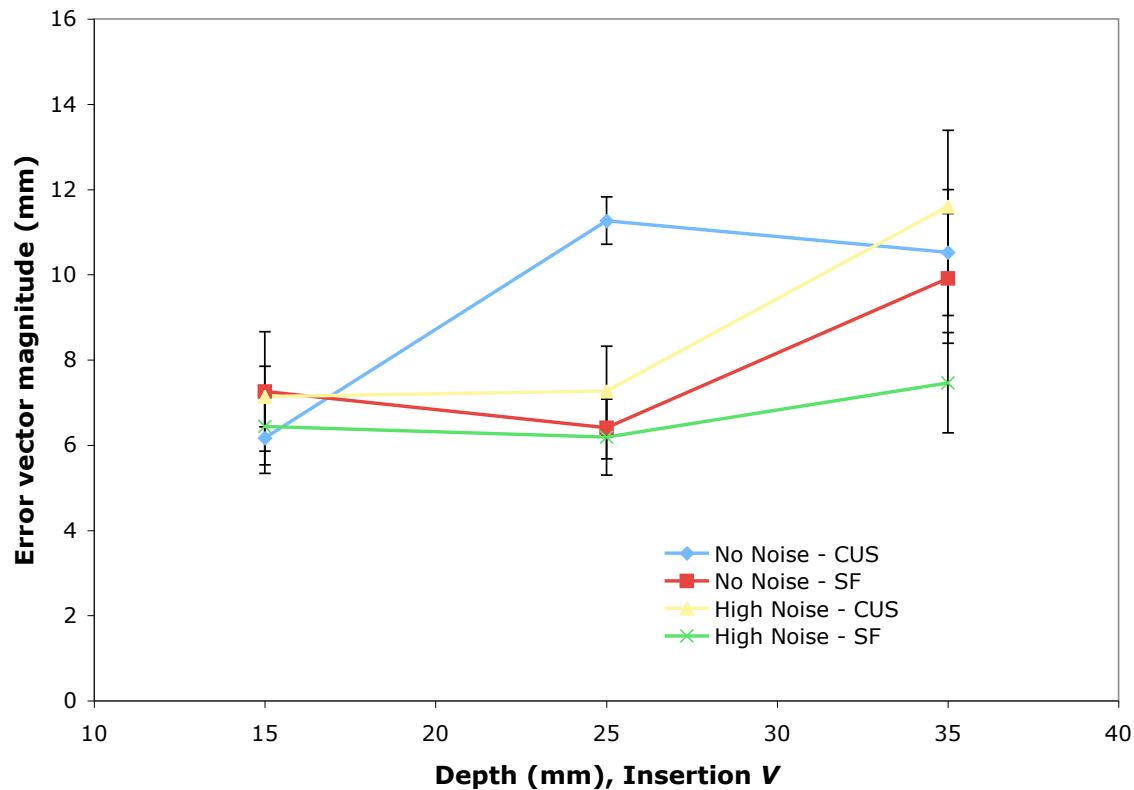


Figure 7.8: Insertion, error vector magnitude, mean \pm standard error.

Summary of statistics:

1. The main effect of depth is significant ($F(2, 26) = 3.526, p < 0.05$)
2. The main effect of noise is barely not significant ($F(2, 26) = 4.455, p = 0.055$)
3. The main effect of device is barely not significant ($F(2, 26) = 4.366, p = 0.057$)
4. The interaction between target depth and background noise is barely not significant ($F(2, 26) = 2.903, p = 0.073$)

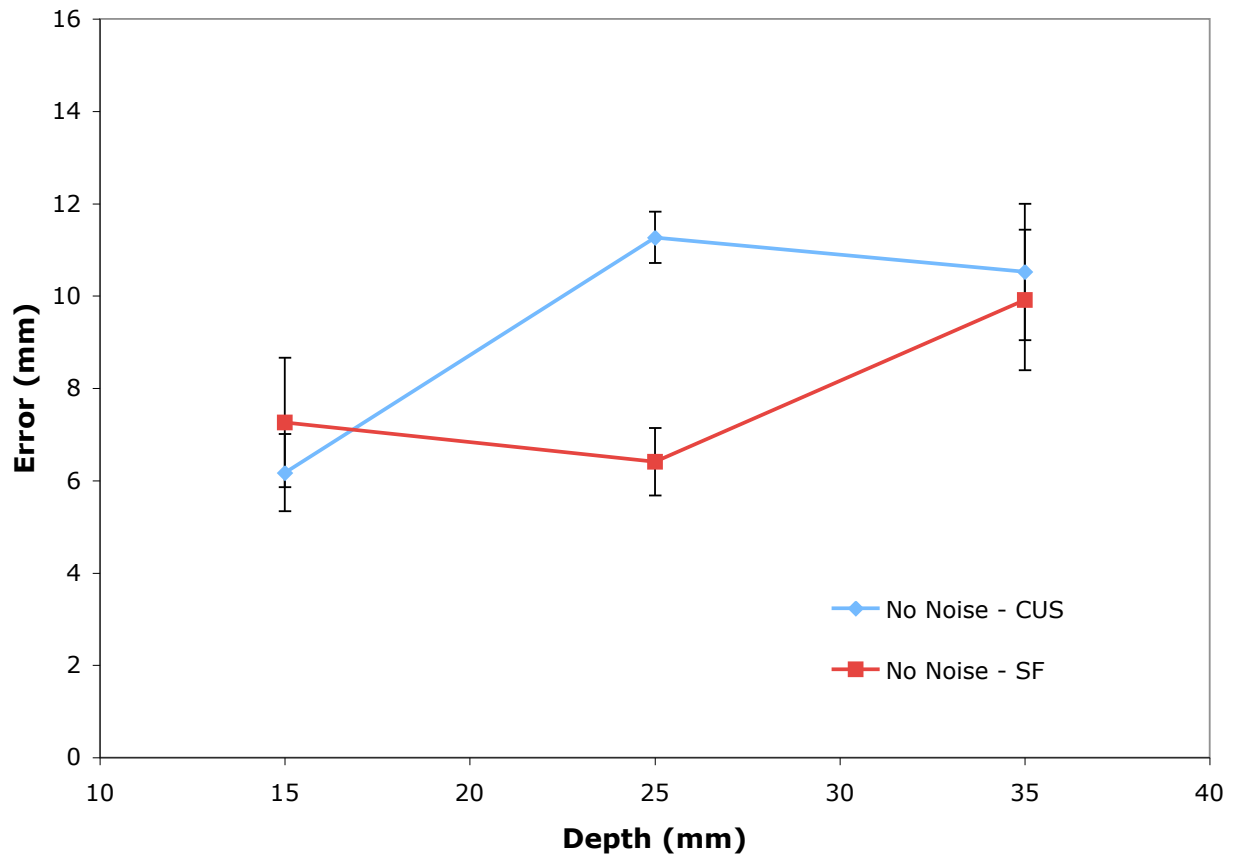


Figure 7.9: Insertion, error vector magnitude, no noise condition, mean \pm standard error.

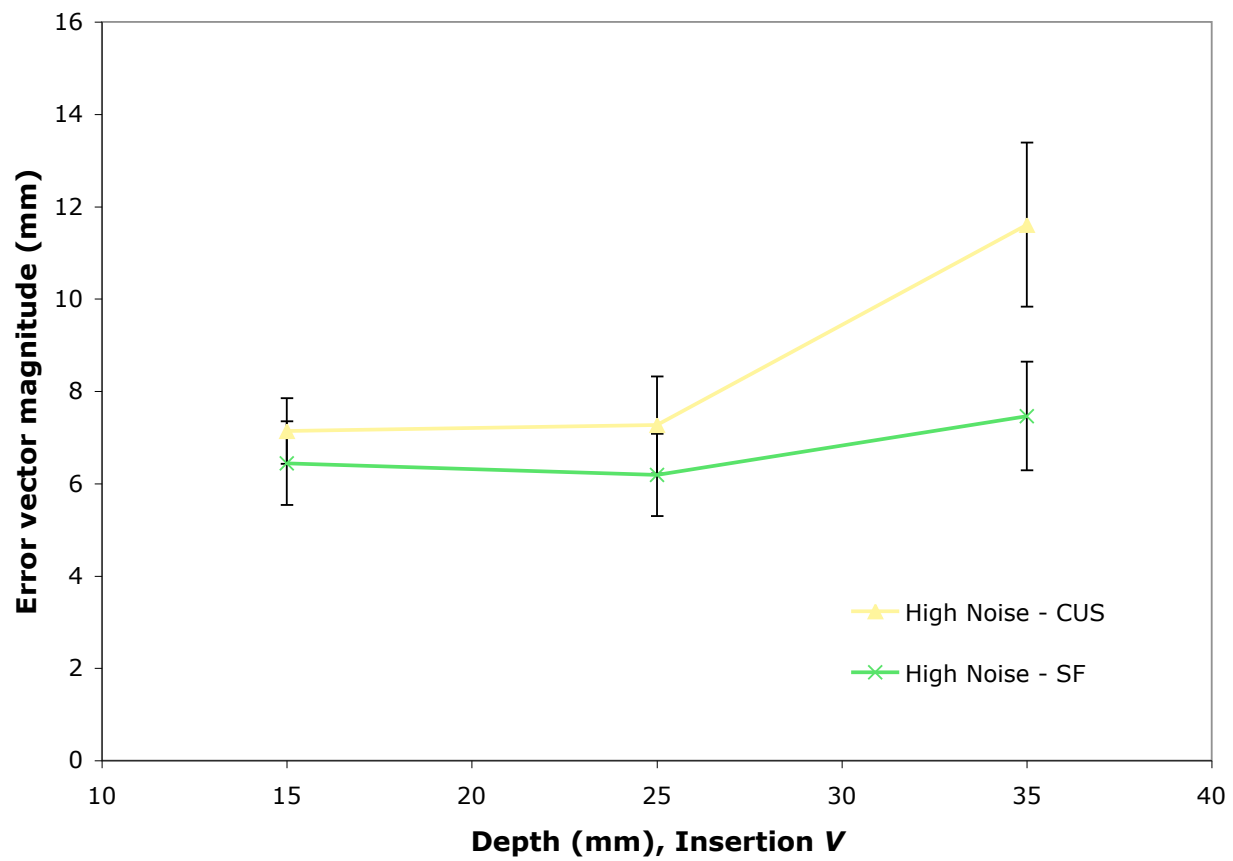


Figure 7.10: Insertion, error vector magnitude, high noise condition, mean \pm standard error.

7.5.2 Error along V_x axis

Depth	No Noise, CUS	No Noise, SF	High Noise, CUS	High Noise, SF
15	3.7 (5.2)	6.2 (6.0)	4.2 (5.4)	5.3 (4.4)
25	8.4 (6.7)	5.9 (3.1)	6.7 (4.3)	4.2 (5.0)
35	8.0 (7.8)	8.8 (6.6)	9.7 (8.1)	5.5 (6.2)

Table 7.5: Insertion, V_x error, mean (stdev), values in mm, mean \pm standard error.

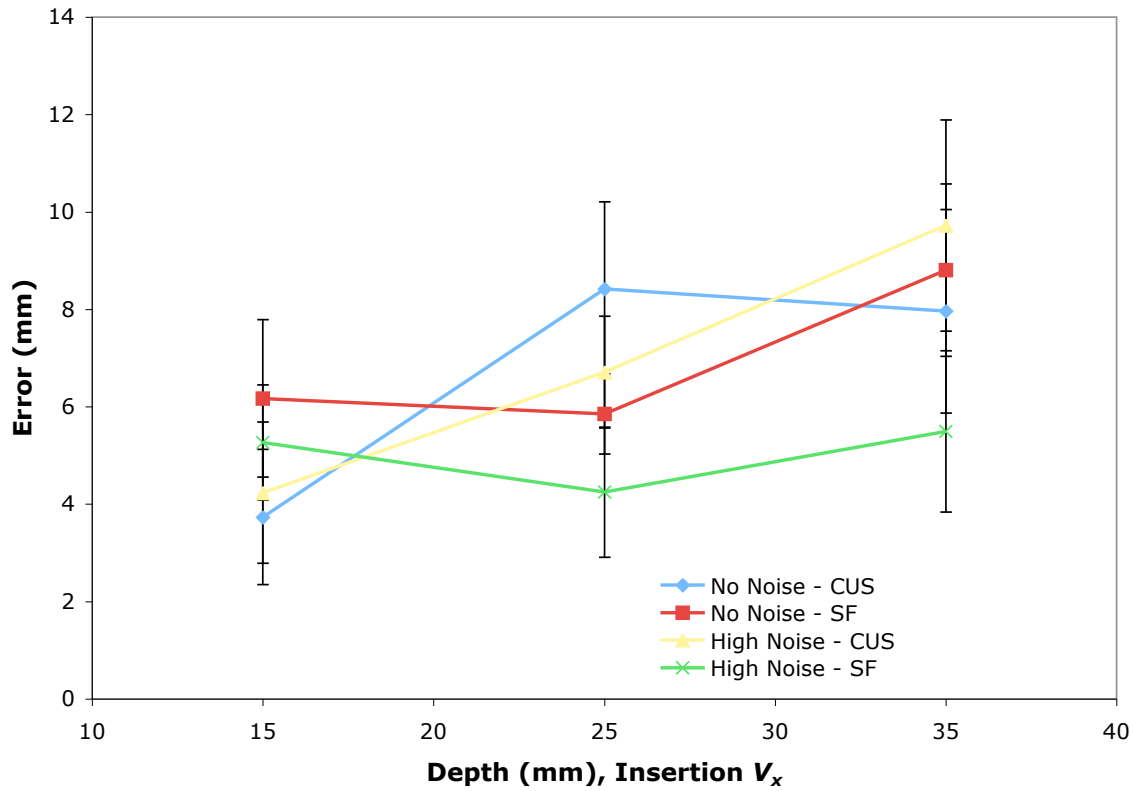


Figure 7.11: Insertion, V_x error.

Summary of statistics:

- ★ The main effect of background noise is barely not significant ($F(2, 26) = 3.683, p = 0.077$)
- ★ The interaction between depth and device is significant ($F(2, 26) = 4.354, p < 0.05$)
- ★ The interaction between background noise and device is significant ($F(2, 26) = 4.747, p < 0.05$)

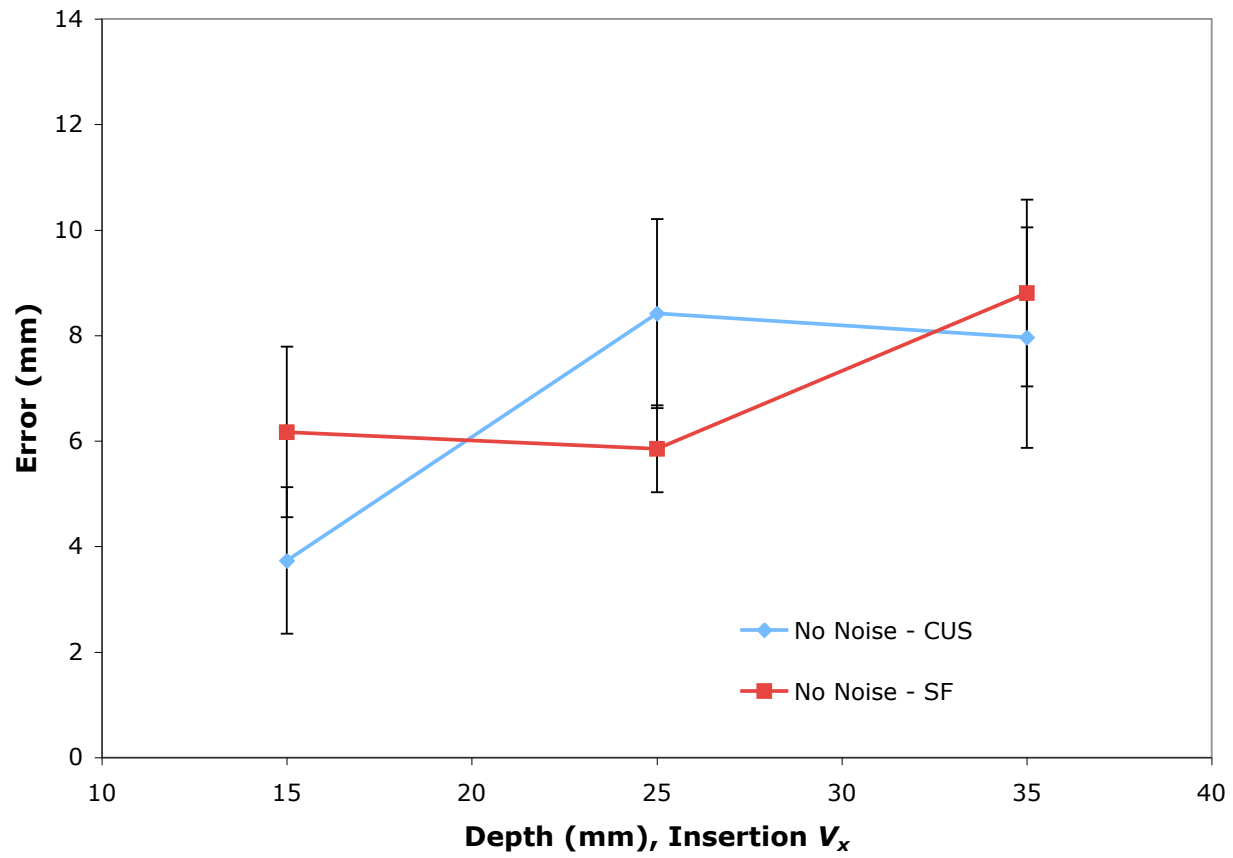


Figure 7.12: Insertion, V_x error, no noise condition, mean \pm standard error.

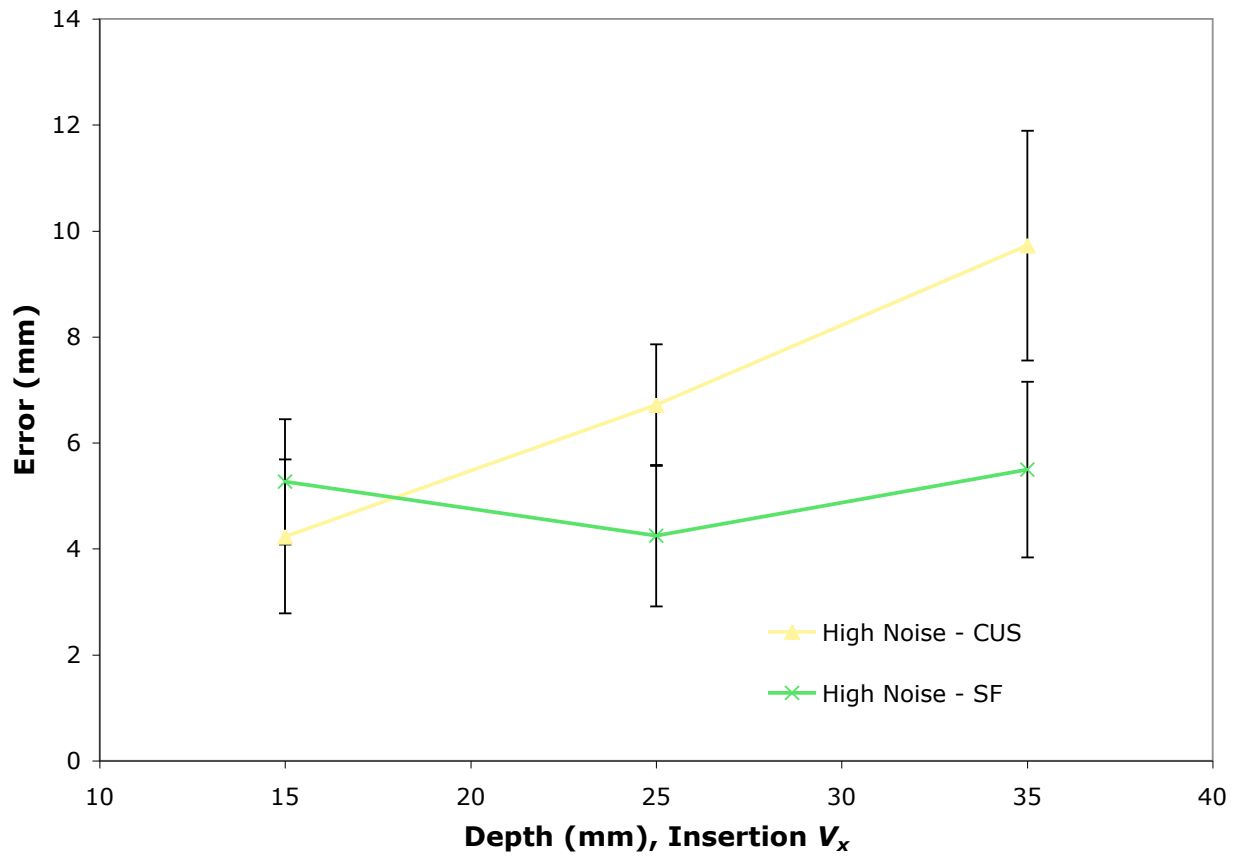


Figure 7.13: Insertion, V_x error, high noise condition, mean \pm standard error.

7.5.3 Error along V_y axis

Depth	No Noise, CUS	No Noise, SF	High Noise, CUS	High Noise, SF
15	0.1 (2.2)	0.6 (1.6)	-0.1 (2.9)	0.9 (1.5)
25	1.3 (3.2)	1.1 (1.5)	1.0 (1.5)	0.6 (2.0)
35	1.0 (3.6)	0.9 (2.1)	0.6 (3.7)	0.8 (2.2)

Table 7.6: Insertion, V_y error, mean (stdev), values in mm.

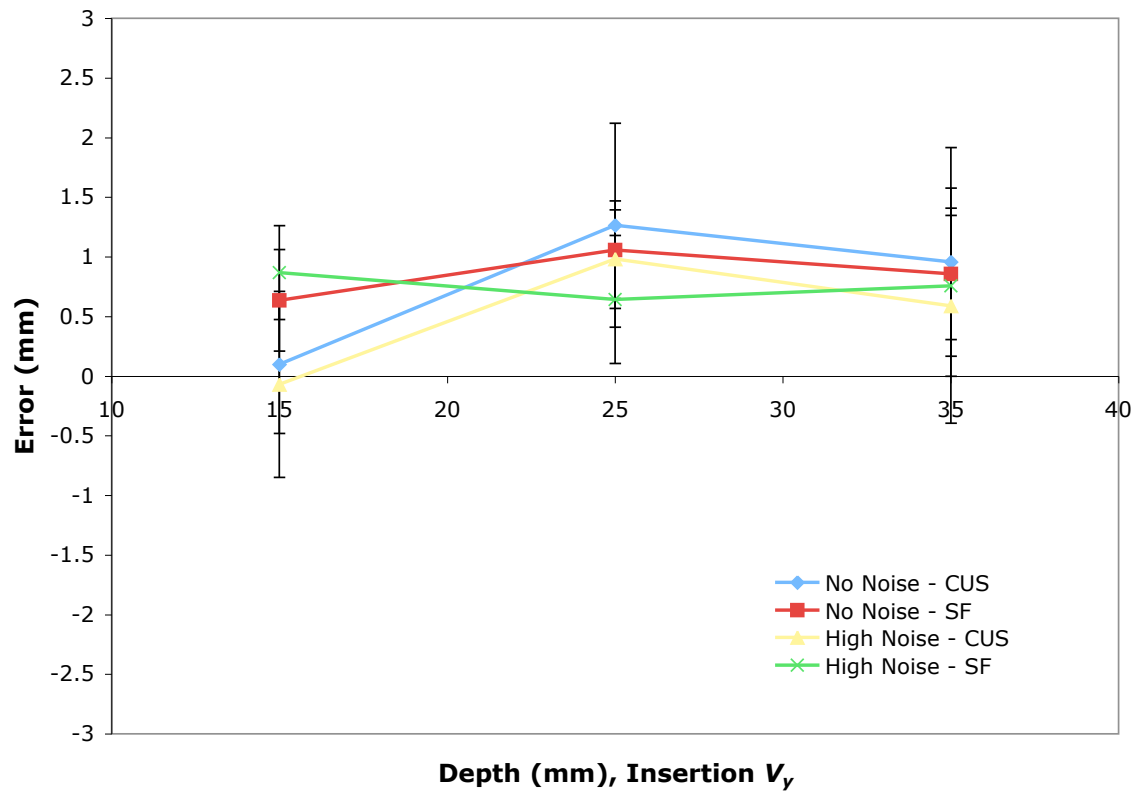


Figure 7.14: Insertion, V_y error, mean \pm standard error.

Summary of statistics:

★ No main effects or interactions are significant.

7.5.4 A comparison of performance with SF and CUS in V_x during insertion

As shown in Table 7.7 below, the majority of subjects (10 of 14) performed better with the SF than they did with CUS. By correlating the mean performance of each subject on a per modality basis with that subject's difference between SF and CUS (Figure 7.15), it is clear from the slope of each trendline that subjects tend to get worse with CUS much faster than they get better with the SF. This effect is even more pronounced if I exclude subjects 5 and 4 from the analysis because of their anomalous performance (Figure 7.16).

Subject	CUS - SF	Mean CUS	Mean SF
8	9.3	10.4	1.0
2	8.2	14.1	5.9
1	7.2	10.8	3.5
6	6.9	12.9	6.0
9	6.5	8.8	2.3
14	5.8	7.0	1.2
11	2.5	4.6	2.1
7	1.1	5.4	4.3
3	1.1	9.4	8.3
12	0.2	8.4	8.3
13	-1.1	3.0	4.1
10	-1.5	0.1	1.6
5	-5.8	5.5	11.3
4	-14.0	-3.9	10.1

Table 7.7: Difference between CUS and SF during insertion, V_x error, values in mm. Positive CUS-SF values indicate an improvement of the SF over CUS.

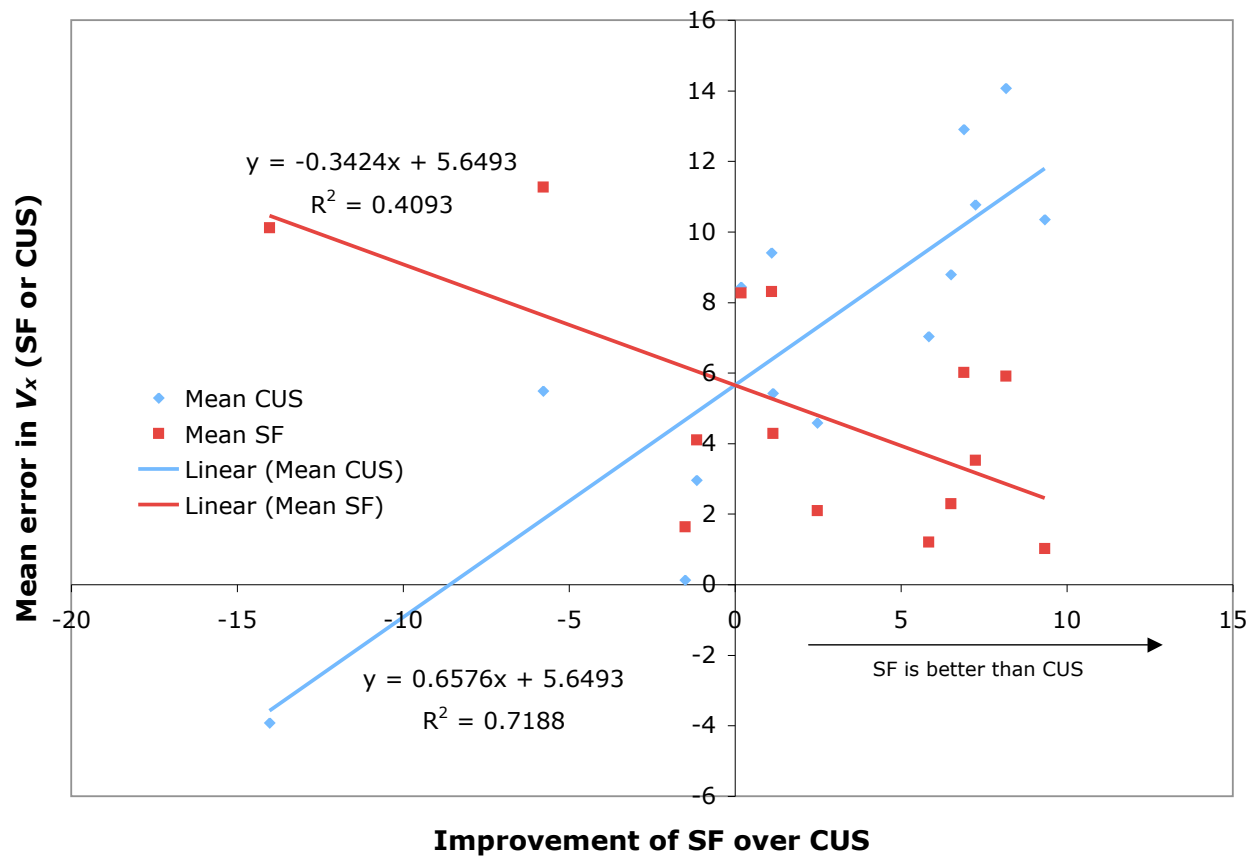


Figure 7.15: Correlation between the improvement of SF over CUS and the mean error of SF and CUS.

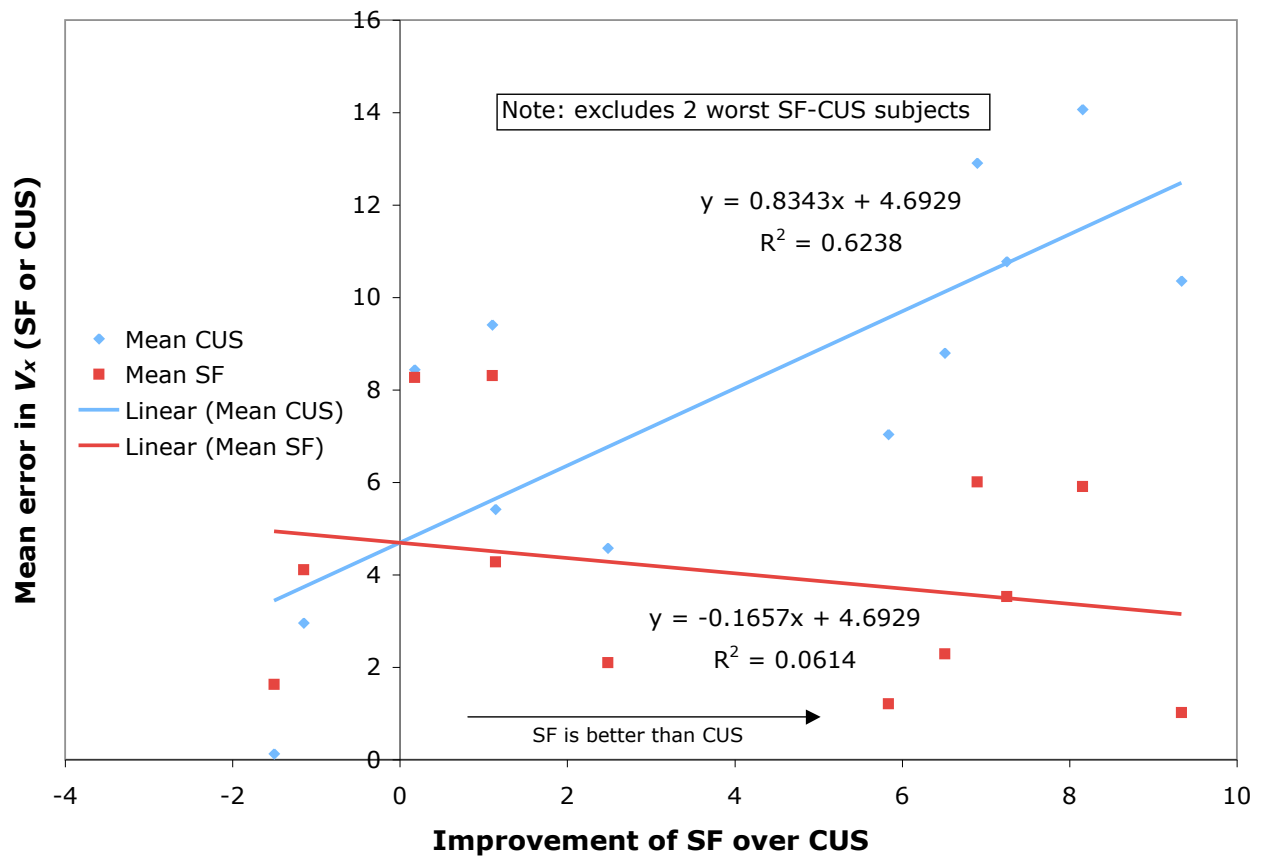


Figure 7.16: Correlation between the improvement of SF over CUS and the mean error of SF and CUS, excluding the 2 subjects with the worst improvement.

7.6 Summary of Experiment 3 results

The results from Experiment 3 are affected by a bug which was not found until the final subject. The software code that controls the point overlay mode will randomly malfunction in experiments that contain multiple target sets (such as SF and CUS). Since Experiment 3 is the only experiment that uses multiple target sets, the results from Experiments 1 and 2 were unaffected.

This bug will tend to reduce the significance of the background noise rather than generate false significance. Since experiments 1 and 2 have already clearly demonstrated that subjects perform best with the SF when the background is complex, an incorrect display of high noise during a trial which should be no noise, or vice versa, will only serve to reduce the difference between the means of each condition. Because the bug is random, it is not possible to tell how many subjects were affected. Given that there is a significant effect of noise present in the data, it seems likely that only a few subjects were affected.

The take-home message from Experiment 3 is as follows:

- ★ The SF performs significantly better than the CUS as measured in V_x in both pointing and insertion *provided that rich background cues are present* (Figures 7.6, 7.10, and 7.13). Absent adequate texture cues, there is no difference between the two (Figures 7.5 and 7.12).
- ★ The advantage of the SF over CUS increases at deeper depths. This is most apparent in Figure 7.13, but is also visible in Figure 7.6.
- ★ Subjects who perform better with the SF than the CUS do so largely due to their poor performance with CUS (Figures 7.15 and 7.16), rather than because of a substantial increase in their performance with the SF. This effect is most noticeable in the second figure, which excludes the two subjects who performed dramatically better with the CUS and SF (not the norm).

Chapter 8

Experiment 4: Meta analysis of Experiments 1 and 3

8.1 Introduction

Experiments 1 and 3 overlap in the sense that they both provide results where the SF is used on targets at 3 depths with no and high noise backgrounds. Red bar feedback is provided in both experiments, and both experiments collect both pointing and insertion data. This suggests that a meta analysis of the two could be used to increase the statistical power of the combined data.

Experiment number (either 1 or 3) was introduced as a factor in a between subject analysis of depth and background noise on the data from Experiments 1 and 3. No significant interaction of experiment number was found with either target depth or background noise – a main effect would be expected due to normal variation in the means of the two populations – so the analysis presented in this chapter pools the overlapping data together and treats them as a single experiment, Experiment 4.

This analysis includes 29 subjects, 17 men and 12 women.

8.2 Needle pointing results

8.2.1 Error along V_x axis

Depth	No Noise	High Noise
15	4.7 (5.8)	3.1 (5.4)
25	6.7 (4.3)	5.3 (6.2)
35	9.5 (7.3)	7.7 (7.8)

Table 8.1: Pointing, V_x error, mean (stdev), values in mm.

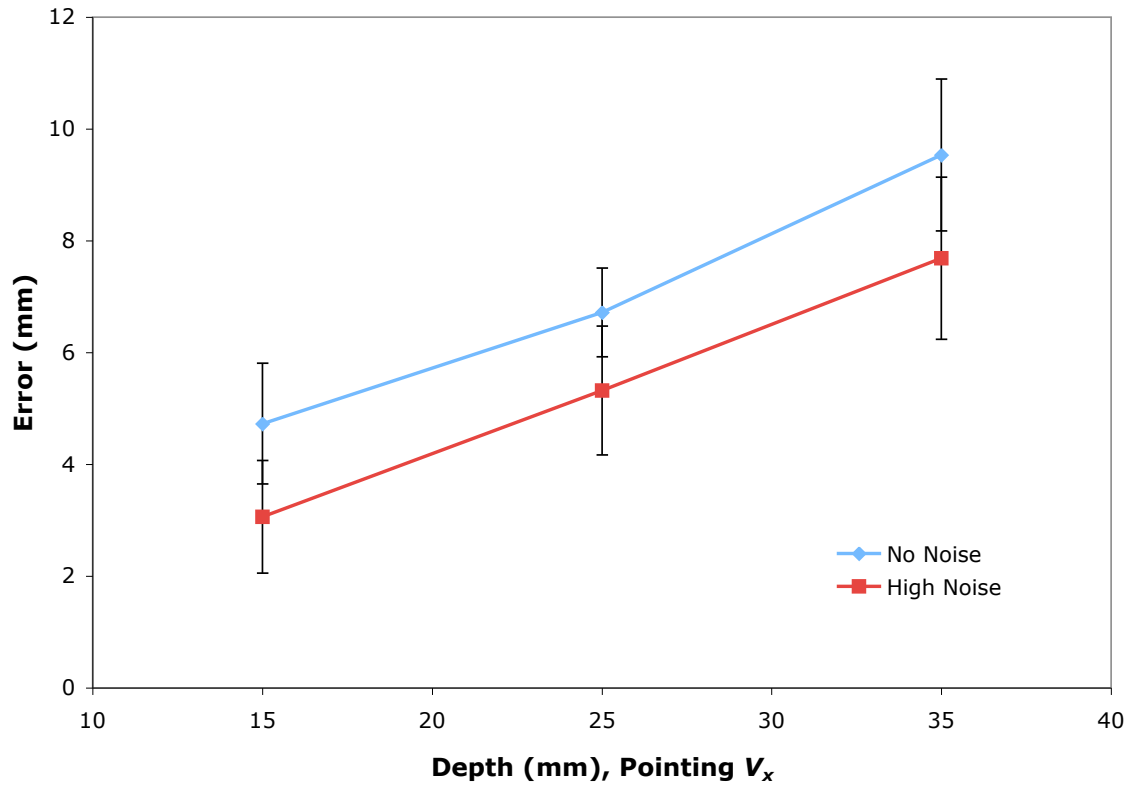


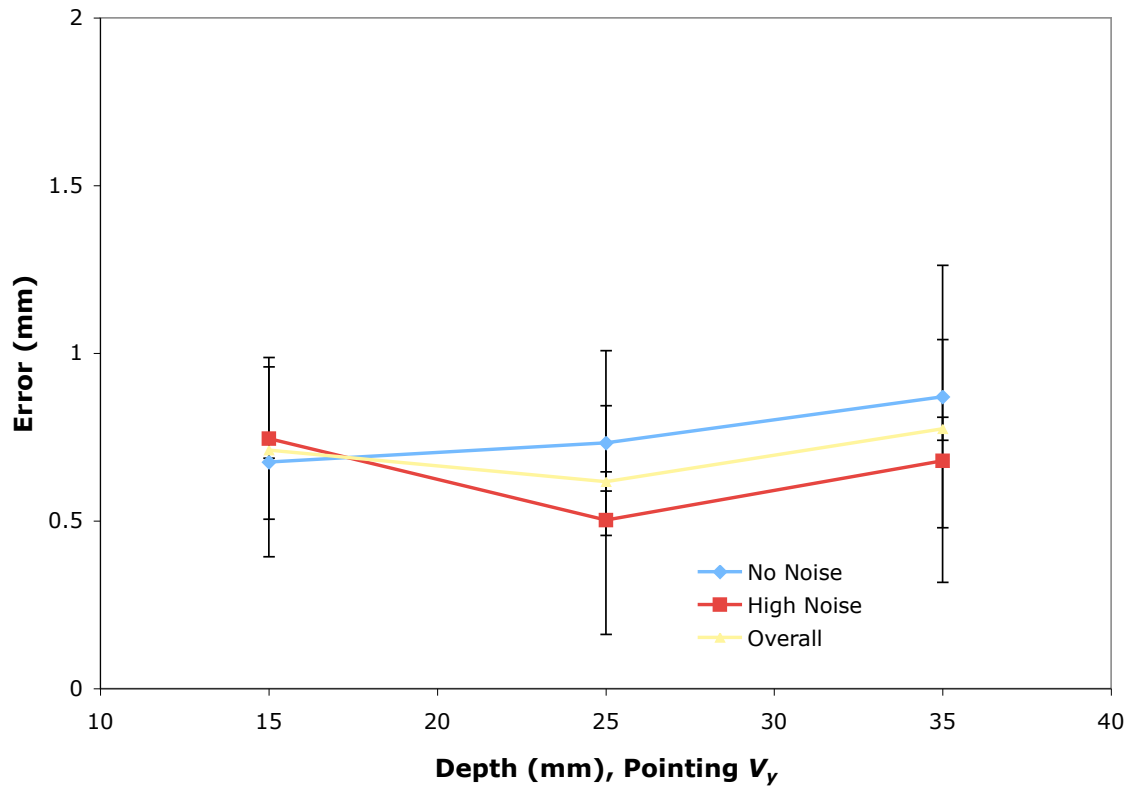
Figure 8.1: Pointing, V_x error, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth is significant ($F(2, 56) = 8.563, p < 0.01$)
- ★ The main effect of background noise is significant ($F(2, 56) = 17.727, p < 0.001$)
- ★ There is no interaction between target depth and background ($F(2, 56) = 0.116, p > 0.8$)

8.2.2 Error along V_y axis

Depth	No Noise	High Noise	Overall
15	0.7 (1.5)	0.7 (1.3)	0.7 (1.4)
25	0.7 (1.5)	0.5 (1.8)	0.6 (1.7)
35	0.9 (2.1)	0.7 (1.9)	0.8 (2.0)

Table 8.2: Pointing, V_y error, mean (stdev), values in mm.Figure 8.2: Pointing, V_y error, mean \pm standard error.

Summary of statistics:

★ No main effects or interactions are significant.

8.3 Needle insertion results

8.3.1 Error along V_x axis

Depth	No Noise	High Noise
15	4.5 (4.9)	3.6 (4.5)
25	6.4 (3.8)	4.5 (5.5)
35	9.0 (6.5)	7.4 (6.7)

Table 8.3: Insertion, V_x error, mean (stdev), values in mm.

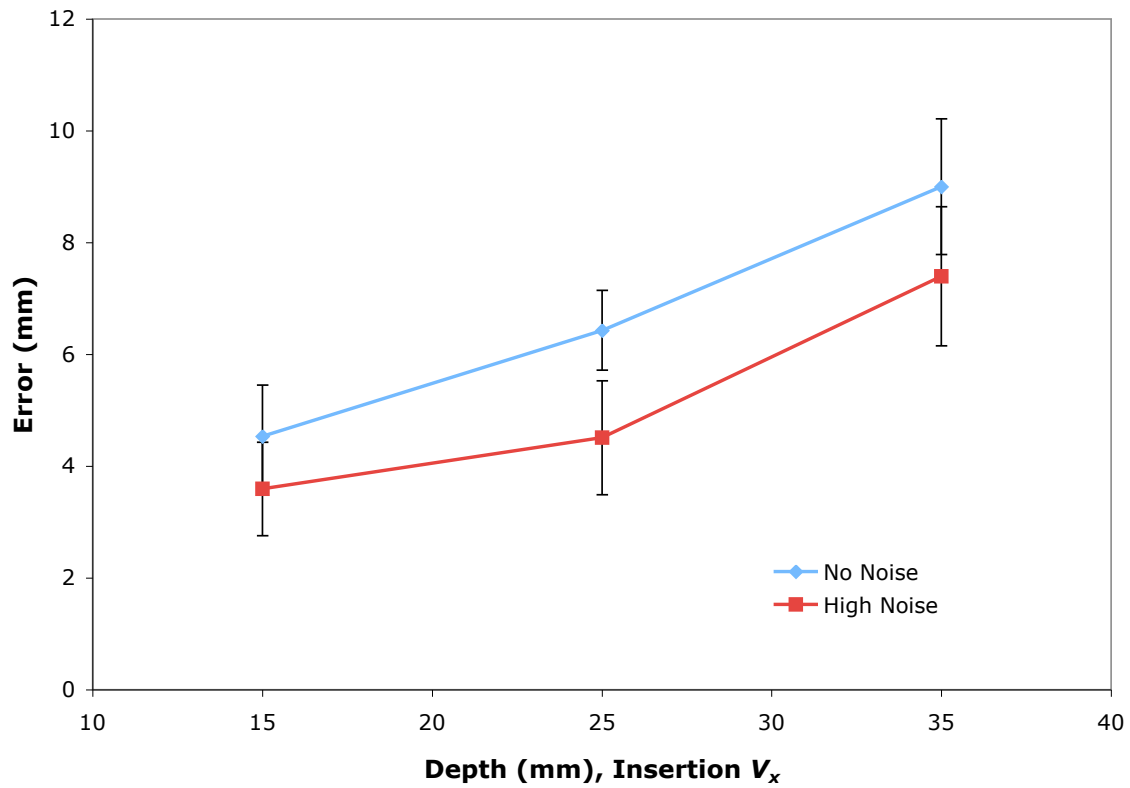


Figure 8.3: Insertion, V_x error, mean \pm standard error.

Summary of statistics:

- ★ The main effect of target depth is significant ($F(2, 56) = 8.417, p = 0.001$)
- ★ The main effect of background noise is significant ($F(2, 56) = 12.843, p = 0.001$)
- ★ There is no interaction between target depth and background ($F(2, 56) = 0.502, p > 0.6$)

8.3.2 Error along V_y axis

Depth	No Noise	High Noise	Overall
15	1.0 (1.3)	1.0 (1.3)	1.0 (1.3)
25	1.2 (1.4)	0.9 (1.7)	1.0 (1.6)
35	1.3 (2.1)	1.1 (1.9)	1.2 (2.0)

Table 8.4: Insertion, V_y error, mean (stdev), values in mm.

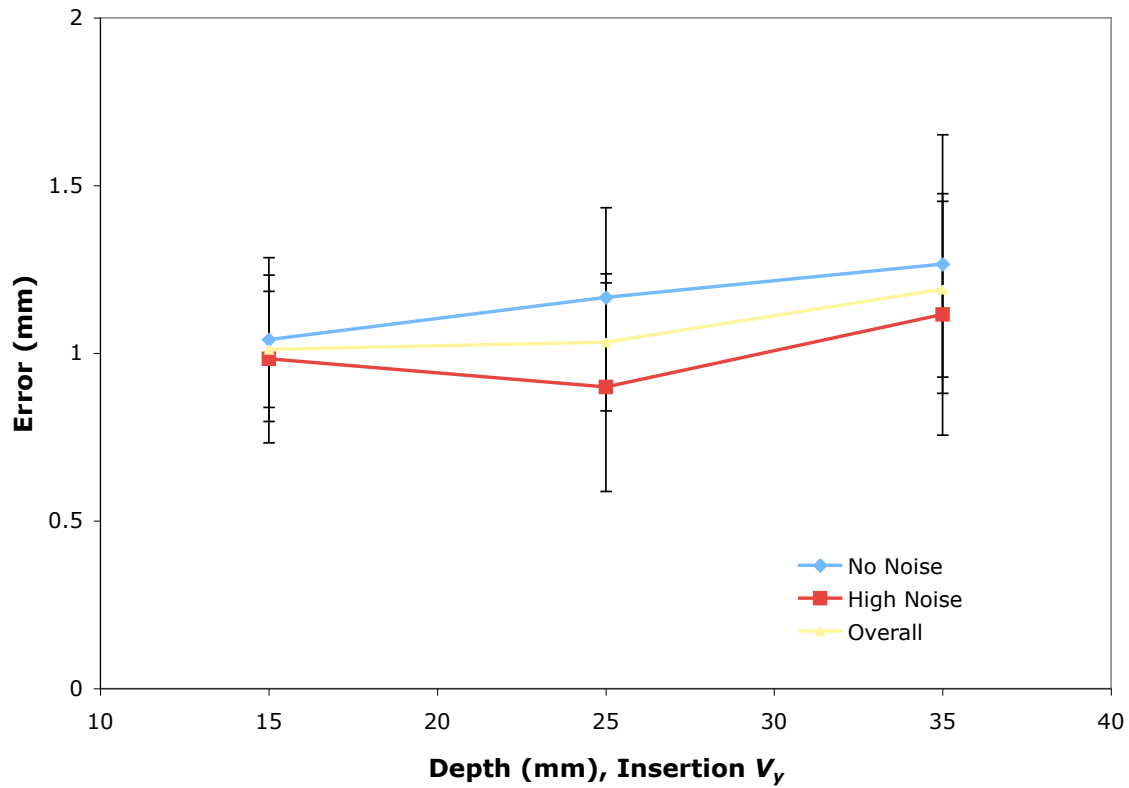


Figure 8.4: Insertion, V_y error, mean \pm standard error.

Summary of statistics:

★ No main effects or interactions are significant.

8.4 Summary of Experiment 4 Results

The take-home message from Experiment 4 is as follows:

- ★ Error in V_x increases as target depth increases.
- ★ SF performance suffers in the absence of a dense background.
- ★ Error in V_y does not depend on either target depth or background noise. In fact, the error in V_y is nearly constant – approximately 1 mm – across all depths, noise levels, and between pointing and insertion. I think that the most likely explanation is that the virtual image calibration is off by approximately that amount in the V_y direction. A second possibility is that subjects tend to make a small motor-induced error that is linked to their dominant hand and the relationship of their posture to the target. With the majority of subjects being right-handed, I would expect such a bias to be either consistently positive (as seen), or negative, depending on the orientation of the SF relative to the target.

Chapter 9

Conclusion and future work

In this dissertation I have described my implementation of Virtual Tomographic Reflection and presented a series of experiments where I used the VTR system to perform psychophysics research on the Sonic Flashlight. In this chapter I briefly summarize the main contributions of my work and describe the future potential of VTR, both as a research and visualization tool.

9.1 Simulation design

VTR is a novel method for simulating needle biopsy using either SF or CUS guidance. In the process of designing and implementing VTR I developed both the physical hardware of the system and the software that controls it. I also developed the routines used to calibrate the system.

- ★ System architecture – I explained the concept of VTR and why it is useful for psychophysics testing of augmented reality devices, in particular the Sonic Flashlight. I then identified the functional components of VTR and described my implementation of these components.
- ★ Tracked Sonic Flashlight – I presented my design for a tracked SF, a modification of the model 6 clinical SF, and showed how it was implemented in hardware.
- ★ Calibration routines – I developed calibration routines for the mock SF, needle tool, and image container. I also derived the matrix manipulations that relate the different reference frames within the VTR system.
- ★ 3D image slicing software – I described my implementation of a high-performance hardware accelerated method for intersecting a 2D cutting plane with a 3D image.

9.2 Psychophysics

I performed 3 experiments (and 1 meta analysis) to assess the data collecting abilities of the VTR system and examine psychophysical characteristics of SF and CUS guidance. These experiments examined the effect of target depth, image background, and guidance method on the ability of users to hit a simulated target with a needle.

- ★ When using both CUS and the SF, subjects underestimate target depth to a greater degree with deep targets, and to a lesser degree with shallow targets. This effect appears in both the pointing and insertion data.
- ★ Subjects make greater errors in depth when using the SF in the absence of rich depth cues (in the form of a textured background).
- ★ If subjects are using the SF with a rich depth cue (high noise) background, their performance is significantly better with the SF than with CUS.
- ★ If subjects are using the SF with a cue-reduced (no noise) background, their performance is not significantly better than CUS.
- ★ Subjects who perform “better” with the SF rely more heavily on a high noise background than subjects who perform “poorly”. “Better” subjects show a larger difference in mean performance between high and no noise conditions.
- ★ Several experiments suggest that the importance of a rich background increases with target depth.
- ★ Feedback regarding the intersection of the needle and slice plane is crucial to the ability of subjects to stop the needle on the surface of the slice plane. Absent this feedback, they are unable to judge over or under-penetration of the slice plane.

9.3 Contributions of this research

- ★ The VTR platform is expected to largely replace purely physical phantoms in future research on the psychophysics of the SF and CUS in our lab.
- ★ Calibration routines developed for this thesis can be applied outside of the specific domain of VTR – for example, with other augmented reality systems that require a virtual image to be registered to a tracked reference frame.
- ★ Individual software components of the VTR system, particularly the 3D image slicing routine, can easily be incorporated into other projects such as accumulation and enhanced display of real ultrasound data in 3D with a tracked SF.
- ★ I have demonstrated the importance of texture on perceived depth of near-space targets, which may motivate improvements in the clinical device.
- ★ I have confirmed prior work showing the superiority of SF over CUS guidance. In addition, my work suggests that background texture in the SF image is crucial to its performance gain relative to CUS.

9.4 Future work

9.4.1 Additional analysis techniques

The VTR system is capable of collecting data in a streaming mode in addition to collecting individual frames of data, as I did for this dissertation. In stream mode, the VTR system writes data to disk at the same frame rate as the simulation executes (30 frames per second as currently designed, although easily controlled in software). Stream data offers a great deal of potential for measuring subject behavior that is of both clinical and psychological interest. Examples of such behavior relating to catheter insertion or biopsy include repeated small insertions and withdrawals (“poking around” or “hunting”) and non-linear trajectories between aiming and final insertion. Although I did implement and test the streaming mode, analysis of the data it produces was beyond the scope of this thesis and will require additional techniques capable of handling the large amount of data acquired during streaming operation.

Streaming mode aside, there are numerous additional mathematical computations that may be performed even on data acquired in single frame mode. Examples include the extent of needle penetration through the surface of the image container, the extent of needle penetration through the virtual image plane¹, and the angles between the transducer, image container, and needle. It is unknown which, if any of these, would yield interesting results. It is possible that by assuming an approximate eye position for the user and measuring transducer and needle angles, I could remove some of the uncertainty regarding perceived depth discussed in Chapter 4. Additional tracking could also be implemented to actually establish the viewer’s position for research purposes.

9.4.2 New phantom designs

The image container I designed for this dissertation worked extremely well for my experiments, but more sophisticated designs offer additional benefits. One of these, a hybrid phantom, would be constructed of acoustic gel and is used with a *hybrid VTR system* where a real ultrasound transducer is used instead of a dummy transducer. A hybrid VTR system combines the actual ultrasound image with simulation results in order to produce a result that is both more believable than a pure simulation and more flexible than a physical phantom alone. Such a system is an immediate goal for future development of the VTR system.

9.4.3 Additional visualization techniques

Finally, the VTR system can be used as an interactively medical images by presenting them as occupying natural 3D space. If a 3D plastic or silicone model of the external anatomy were created, it would be simple to register the image of the interior using the Optotrak system and allow the user to virtually slice through the physical model. This would be useful both for preoperative planning and educational purposes. With regards to the latter, although I worked exclusively with black and white images in my experiments, the present VTR system is capable of displaying color images such as the cryosection data from the Visible Human project. By registering color cryosection data with CT and MRI images of the same anatomy and providing users with a mechanism to switch between the different image options,

¹In experiment 2 I measured this value with respect to the virtual image V_z axis. It may be beneficial to compute this distance along the needle axis as well.

VTR would offer a compelling tool for learning the relationship between actual anatomy and various imaging modalities.

Appendix A

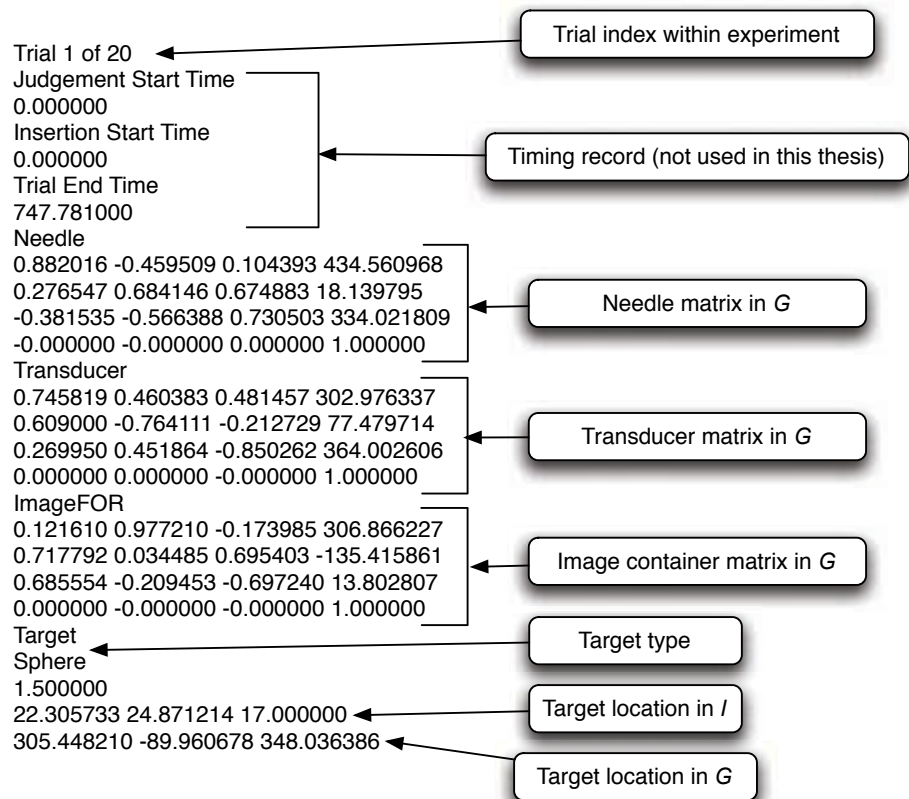
Common Terms, Abbreviations, and Notations

- ★ Augmented Reality (AR) - A fusion of a real and virtual scene, where the resulting scene is predominantly the real world. Contrast with **augmented virtuality**.
- ★ Augmented Virtuality (AV) - A fusion of a real and virtual scene, where the resulting scene is predominantly the virtual world. Contrast with **augmented reality**.
- ★ Conventional Ultrasound (CUS) - Use of an ultrasound scanner where the image is displayed on a computer monitor (i.e. not using an augmented reality display).
- ★ Frame of reference - The local coordinate system associated with any object in space.
- ★ Medical image - In the context of this dissertation, any 3D pixel-based image of the human body. In general usage, the image need not be 3D or discretized into pixels.
- ★ Orientation - The rotation of the frame-of-reference coordinate system attached to an object relative to an external frame of reference, if the origins of the two reference frames are assumed to be coincident.
- ★ Peripherally-inserted central catheter (PICC) - A catheter inserted into a limb intended for final placement in the torso, often the heart.
- ★ Position - The 3D cartesian coordinates of the origin of the **frame of reference** attached to an object, relative to an external **frame of reference**.
- ★ Real-time tomographic reflection (RTTR) - An **augmented reality** technique that uses a calibrated display, mirror, and real-time **tomographic** image source to combine a **virtual image** with **direct vision**.
- ★ Rigid body - In the context of an Optotrak tracking environment, any set of rigidly connected components which share a common set of tracking markers and a common **frame of reference**.

- ★ Sonic Flashlight (SF) - An implementation of **RTTR** where the real-time image source is ultrasound.
- ★ Tomographic - Any imaging modality that displays cross-sections through the object being imaged.
- ★ Virtual - When referring to anything other than a **virtual image**, should be taken to mean *simulated* or *synthetic*.
- ★ Virtual image - An image, generated by an optical system, which is formed when the principal rays in the system cross when projected backward from their path. A virtual image can be viewed by an observer, but will not generate an image on a piece of diffuse material placed at the observer location.
- ★ Virtual Ultrasound Environment (VUE) - The primary software application used in VTR. It controls the data collection process and performs the 3D slicing simulation.
- ★ Voxel - In a discretized image, a pixel with a dimensionality of greater than 2 (in the context of this dissertation, a 3D pixel).

Appendix B

Example data frame from VTR system



Appendix C

Experiments 1 & 2 image parameters

Trial	I_x	I_y	I_z	Radius	Feedback type
1	22.8	19.6	20	2	intersection
2	16.2	18.7	22	5	intersection
3	21.3	20.2	17	3	intersection
4	20.2	24.5	33	1.5	intersection
5	17.7	17.8	15	4	intersection
6	25.0	24.6	20	2	red bar
7	20.9	23.3	22	5	red bar
8	16.3	16.4	17	3	red bar
9	22.3	21.8	33	1.5	red bar
10	19.1	20.4	15	4	red bar

Table C.1: Experiments 1 & 2 practice set.

Trial	I_x	I_y	I_z	Noise level
1	22.8	20.9	25	5000
2	20.2	16.5	15	5000
3	23.0	19.2	15	0
4	22.3	19.7	18	5000
5	15.9	24.0	35	0
6	24.7	24.4	15	250
7	21.1	19.0	25	0
8	20.0	20.8	29	5000
9	23.0	20.5	35	250
10	19.5	18.9	35	5000
11	18.9	15.9	25	250
12	21.5	17.7	31	5000
13	20.0	16.9	25	5000
14	18.4	21.0	35	5000
15	21.6	17.8	35	0
16	16.9	18.8	17	5000
17	24.7	15.9	25	250
18	19.0	22.1	35	250
19	15.2	24.2	15	250
20	20.4	22.6	19	5000
21	20.5	17.3	15	5000
22	19.1	24.4	25	0
23	16.3	16.9	15	0
24	16.5	21.7	22	5000
25	19.0	20.6	35	250
26	17.2	15.7	25	250
27	21.8	22.9	15	0
28	23.6	19.3	24	5000
29	24.5	21.2	15	250
30	24.8	19.6	35	5000
31	20.1	20.5	25	0
32	24.6	24.1	31	5000
33	20.4	18.2	25	5000
34	21.0	23.6	35	0
35	22.9	16.3	15	5000
36	17.9	23.2	22	5000

Table C.2: Experiments 1 & 2 experiment set.

Appendix D

Experiment 3 image parameters

Trial	I_x	I_y	I_z	Radius	Feedback type
1	19.9	18.6	20	4.0	intersection
2	23.3	17.4	20	2.0	red bar
3	18.2	15.6	22	5.0	red bar
4	23.6	19.7	17	3.0	red bar
5	17.5	16.9	33	1.5	red bar
6	18.3	23.2	15	4.0	red bar

Table D.1: Experiment 3 practice set.

Trial	I_x	I_y	I_z	Noise level
1	25.1	41.7	23	5,000
2	30.4	39.4	25	5,000
3	32.4	39.7	19	5,000
4	28.2	43.3	15	5,000
5	25.8	43.7	21	0
6	31.8	41.4	25	0
7	33.8	42.4	17	0
8	25.9	40.5	35	0
9	34.1	44.9	35	5,000
10	31.6	39.4	15	0
11	26.1	38.8	25	0
12	29.6	43.8	35	5,000
13	27.7	37.9	35	0
14	33.1	41.6	33	0
15	27.6	35.5	25	5,000
16	28.5	42.1	31	5,000
17	27.5	36.2	15	0
18	32.2	37.1	29	0
19	28.7	39.4	27	5,000
20	34.1	35.4	15	5,000

Table D.2: Experiment 3 CUS experiment set.

Trial	I_x	I_y	I_z	Noise level
1	22.3	24.9	17	0
2	21.8	21.2	15	0
3	21.0	17.0	25	0
4	19.8	17.4	15	5,000
5	16.5	16.6	25	5,000
6	20.1	20.8	19	5,000
7	15.6	15.6	35	5,000
8	22.6	15.5	35	0
9	19.9	20.0	21	0
10	18.2	21.1	23	5,000
11	19.0	15.5	25	5,000
12	21.6	23.6	33	0
13	16.4	16.2	35	0
14	24.3	23.7	27	5,000
15	20.9	20.4	25	0
16	19.8	24.9	35	5,000
17	16.7	18.4	15	5,000
18	20.1	19.3	29	0
19	21.4	19.8	15	0
20	23.6	18.5	31	5,000

Table D.3: Experiment 3 SF experiment set.

Appendix E

List of third party software, toolkits, and programming languages

E.1 Languages

- ★ C++: Programming language used when implementing the VTR system.
- ★ OpenGL: 3D graphics language used to render the image cutting plane.
- ★ Mathematica: Mathematics language used for virtual image calibration.
- ★ Matlab: Mathematics language used for data post-processing.

E.2 Toolkits

- ★ The Fast Light Toolkit (FLTK): Cross-platform GUI development toolkit, used extensively in the VTR system.
- ★ The Insight Toolkit (ITK): Medical imaging toolkit available from <http://www.itk.org> used for the image generation, image loading, and image processing portions of the VTR system.
- ★ Northern Digital Optotrak 3020 API: Interface to the Optotrak tracking system.
- ★ The Visualization Toolkit (VTK): 3D visualization and graphics toolkit available from <http://www.vtk.org> used for the overview visualization portion of VTR operator's GUI.

Bibliography

- [1] S.P. Keenan, “Use of ultrasound to place central lines,” *J. Critical Care*, vol. 17, no. 2, pp. 126–137, 2002.
- [2] David Drascic and Paul Milgram, “Perceptual issues in augmented reality,” in *Stereoscopic Displays and Virtual Reality Systems III*, Scott S. Fisher, John O. Merrit, and Mark T. Bolas, Eds. 1996, vol. 2653 of *Proceedings of the SPIE*, pp. 123–124, SPIE.
- [3] Ronald Azuma, Yohan Baillot, Reinhold Behringer, Steven Feiner, Simon Julier, and Blair MacIntyre, “Recent advances in augmented reality,” *IEEE Computer Graphics and Applications*, vol. 21, no. 6, pp. 34–47, 2001.
- [4] Frank Sauer, Ali Khamene, B. Bascle, L. Schimmang, F. Wenzel, and S. Vogt, “Augmented reality visualization of ultrasound images: system description, calibration, and features,” in *International Symposium on Augmented Reality*. 2001, pp. 30–39, IEEE and ACM.
- [5] H. Fuchs, A. State, M. Livingston, W. Garrett, G. Hirota, M. Whitton, and E. Pisano, “Virtual environments technology to aid needle biopsies of the breast - an example of real-time data fusion,” *Stud Health Technol Inform*, vol. 29, pp. 60–61, 1996.
- [6] Michael Rosenthal, Andrei State, and Joohi Lee, “Augmented reality guidance for needle biopsies: a randomized, controlled trial in phantoms,” in *MICCAI 2001*. 2001, vol. 2208 of *Lecture Notes in Computer Science*, pp. 240–248, Springer-Verlag.
- [7] G. D. Stetten, V. S. Chib, and R. J. Tamburo, “Tomographic reflection to merge ultrasound images with direct vision,” in *Applied Imagery Pattern Recognition Workshop*, J. V. Aanstoots, Ed., 2000, pp. 200–205.
- [8] K. Masamune, Y. Masutani, S. Nakajima, I. Sakuma, T. Dohi, H. Iseki, and K. Takakura, “Three-dimensional slice image overlay system with accurate depth perception for surgery,” in *Medial Image Computing and Computer-Assisted Intervention (MICCAI)*. 2000, vol. 1935, pp. 395–402, Springer.
- [9] Frank Sauer, Ali Khamene, and S. Vogt, “An augmented reality navigation system with a single-camera tracker: system design and needle biopsy phantom trial,” in *MICCAI*, T. Dohi and R. Kikinis, Eds. 2002, vol. 2489 of *LNCS*, pp. 116–124, Springer-Verlag.

- [10] Frank Sauer, Uwe J. Schoepf, Ali Khamene, Sebastian Vogt, Marco Das, and Stuart G. Silverman, "Augmented reality system for ct-guided interventions: system description and initial phantom trials," in *Medical Imaging 2003: Visualization Image Guided Procedures, and Display*, Robert L. Galloway, Ed. International Society for Optical Engineering (SPIE), 2003, vol. 5029 of *Proceedings of the SPIE*, pp. 384–394.
- [11] Fred S. Azar, Nathalie Perrin, Ali Khamene, and Frank Sauer, "User performance analysis of different image-based navigation systems for needle placement procedures," in *Medical Imaging 2004: Visualization, Image-Guided Procedures, and Display*, Robert L. Galloway, Ed. International Society for Optical Engineering, 2004, vol. 5367 of *Proceedings of the SPIE*, pp. 110–121.
- [12] Sebastian Vogt, Frank Wacker, Ali Khamene, Daniel R. Elgort, Tobias Sielhorst, Heinrich Niemann, Jeff Duerk, Jonathan S. Lewin, and Frank Sauer, "Augmented reality system for mr-guided interventions: phantom studies and first animal test," in *Medical Imaging 2004: Visualization, Image-Guided Procedures, and Display*, Robert L. Galloway, Ed. The International Society for Optical Engineering (SPIE), 2004, vol. 5367 of *Proceedings of SPIE*, pp. 100–109.
- [13] R. L. Klatzky, Y. Lipka, J. M. Loomis, and R. G. Golledge, "Encoding, learning and spatial updating of multiple object locations specified by 3-d sound, spatial language, and vision," *Experimental Brain Research*, vol. 149, pp. 48–61, 2003.
- [14] R. L. Klatzky, Y. Lipka, J. M. Loomis, and R. G. Golledge, "Learning directions of objects specified by vision, spatial audition, or auditory spatial language," *Learning and Memory*, vol. 9, pp. 364–367, 2002.
- [15] J. M. Loomis, Y. Lipka, R. L. Klatzky, and R. G. Golledge, "Spatial updating of locations specified by 3-d sound and spatial language," *Journal of experimental psychology*, vol. 28, pp. 335–345, 2002.
- [16] D. Shelton, R. Klatzky, and G. Stetten, "Method for assessing augmented reality needle guidance using a virtual biopsy task," in *IEEE International Symposium on Biomedical Imaging*. 2004, vol. 1, pp. 273–276, IEEE.
- [17] G. Stetten, D. Shelton, W. Chang, V. Chib, R. Tamburo, D. Hildebrand, L. Lobes, and J. Sumkin, "Towards a clinically useful sonic flashlight," in *IEEE International Symposium on Biomedical Imaging*, 2002, pp. 417–420.
- [18] D. Shelton, G. Stetten, and W. Chang, "Ultrasound visualization with the sonic flashlight," in *Proceedings of SIGGRAPH 2002, Emerging Technologies*, 2002.
- [19] W. Chang, G. Stetten, L. Lobes, D. Shelton, and R. Tamburo, "Guidance of retrobulbar injection with real time tomographic reflection," *Journal of Ultrasound in Medicine*, vol. 21, pp. 1131–1135, 2002.
- [20] Will Schroeder, Ken Martin, and Bill Lorensen, *The Visualization Toolkit*, Kitware, Inc., 4 edition, 2003.

- [21] W. Chang, M. Horowitz, and G. Stetten, "Intuitive intra-operative ultrasound guidance using the sonic flashlight, a novel ultrasound display system," *Neurosurgery*, vol. 56, pp. 434–437, 2005.
- [22] W. Chang, N. Amesur, D. Wang, A. Zajko, and G. Stetten, "First human clinical trial of the sonic flashlight - guiding placement of peripherally inserted central catheters," in *2005 Meeting of the Radiological Society of North America*, 2005, number SSJ03-02.
- [23] F. Hamza-Lup, L. Davis, C. Hughes, and J. Rolland, "Marker mapping techniques for augmented reality visualization," in *XVIIth Intl. Symposium on Computer and Information Sciences (ISCIS)*. 2002, pp. 152–156, CRC Press.
- [24] L. Davis, F. G. Hamza-Lup, and J. Rolland, "A method for designing marker-based tracking probes," in *IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, 2004, pp. 120–129.
- [25] Wilson Chang, *Guiding vascular access with the Sonic Flashlight - preclinical development and validation*, Ph.d., University of Pittsburgh, 2004.
- [26] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-d point sets," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 9, no. 5, pp. 698–700, 1987.
- [27] Berthold K. P. Horn, "Closed-form solution of absolute orientation using unit quaternions," *Journal of the Optical Society of North America*, vol. 4, pp. 629–642, 1987.
- [28] Berthold K. P. Horn, Hugh M. Hilden, and Shahriar Negahdaripour, "Closed-form solution of absolute orientation using orthonormal matrices," *Journal of the Optical Society of North America*, vol. 5, pp. 1127–1135, 1988.
- [29] Shinji Umeyama, "Least-squares estimation of transformation parameters between two point patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 4, pp. 376–380, 1991.
- [30] L. Ibanez, W. Schroeder, L. Ng, and J. Cates, *The ITK Software Guide: The Insight Segmentation and Registration Toolkit (version 1.4)*, Kitware, Inc., Albany, NY, 2003.
- [31] P. Bourke, "Intersection of a plane and a line," <http://local.wasp.uwa.edu.au/pbourke/geometry/planeline/>, August 1991.
- [32] B. Wu, R. Klatzky, D. Shelton, and G. Stetten, "Psychophysical evaluation of in-situ ultrasound visualization," *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, vol. 11, no. 6, pp. 684–693, 2005.