GUIDING VASCULAR ACCESS WITH THE SONIC FLASHLIGHT – PRECLINICAL DEVELOPMENT AND VALIDATION

by

Wilson Ming-Wei Chang

BS, University of Wisconsin – Madison, 1998

MS, University of Wisconsin – Madison, 1999

Submitted to the Graduate Faculty of

School of Engineering in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2004

UNIVERSITY OF PITTSBURGH

SCHOOL OF ENGINEERING

This dissertation was presented

by

Wilson Ming-Wei Chang

It was defended on

August 5, 2004

and approved by

Andrew V. Stenger, Ph.D., Assistant Professor, Departments of Radiology and Bioengineering

Douglas D. Robertson, M.D., Ph.D., Associate Professor, Departments of Radiology and Bioengineering

Roberta L. Klatzky, Ph.D., Professor, Carnegie Mellon University Department of Psychology

George D. Stetten, M.D., Ph.D., Associate Professor, Department of Bioengineering
Dissertation Director

GUIDING VASCULAR ACCESS WITH THE SONIC FLASHLIGHT – PRECLINICAL DEVELOPMENT AND VALIDATION

Wilson Ming-Wei Chang, Ph.D.

University of Pittsburgh, 2004

This dissertation concerns the development of a device called the Sonic Flashlight, which employs a novel method for viewing real-time ultrasound images inside the body exactly at the location where it is being scanned. While other augmented reality methods have previously been developed to view ultrasound and other medical imaging modalities within the body, they are generally much more complicated, slower and less robust than the Sonic Flashlight.

In this dissertation, we aim to develop the Sonic Flashlight towards one particular clinical application, central vascular access, and lay the groundwork leading to the first clinical trials. The goal of central vascular access is to insert a catheter into a major vein to deliver medications in large quantities. These veins are usually not visible to the naked eye, so real-time ultrasound is employed to guide the needle into them. While real-time ultrasound guidance significantly enhances the safety of central venous access, learning this skill can be a challenge for the novice user, one major obstacle being the displaced sense of hand-eye coordination that occurs when the operator must look away from the operating field to view the conventional ultrasound monitor.

We developed the 5th generation Sonic Flashlight, as well as a novel calibration method, called thin-gel calibration, as part of this dissertation. The thin-gel system allows us to accurately calibrate the Sonic Flashlight and measure the calibration accuracy. Finally, experiments were conducted with a variety of subject populations using vascular ultrasound phantoms and cadavers to validate Sonic Flashlight guidance, demonstrating that the device is ready for clinical trials.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CG	Computer Generated
CI	Confidence Interval
CT	Computer Aided Tomography
CUS	Conventional Ultrasound
HMD	Head-Mounted Display
IJV	Internal Jugular Vein
IV	Intravenous (catheter)
MRI	Magnetic Resonance Imaging
SF	Sonic Flashlight
SV	Subclavian Vein
SVC	Superior Vena Cava
US	Ultrasound

PREFACE

This dissertation marks the transition from my research training back to my medical training. Although I have often felt left behind by many of my friends from medical school, I would not trade the experiences during the past three years for anything. I would like to thank George Stetten, my advisor, for his support, guidance, teaching, and general attitude towards life. I could not have asked for a better advisor or learning environment. Who else would have understood my personal need to explore life, in addition to my research career, by going backpacking in the Indian Himalayas and complete my first Ironman. I would also like to thank my thesis committee, especially Bobby Klatzky for her expert guidance during this work.

I also owe many thanks everybody else in the lab, past and present, including Damion Shelton, Robert Tamburo, Aaron Cois, John Galeotti, Samuel Clanton, Bing Wu, Gaurav Shukla, and Apryle Craig. Throughout my journey, I have had my friends and family there to support me, for which I am truly grateful.

1.0 INTRODUCTION

This dissertation encompasses the development of the Sonic Flashlight (SF), an augmented reality ultrasound (US) display system. While the SF has many potential medical applications, this work specifically focuses on its development for guiding vascular access. Gaining central venous access is one of the common and important procedures in managing critically ill patients.

We begin this chapter by describing why central catheters are placed in the body, as well as common placement locations. Then, we will show the role that US plays in placing these catheters. Finally we introduce the SF and the importance of a calibration system, followed by an outline to the rest of the dissertation.

1.1 GOALS AND MOTIVATIONS

Central venous catheters are small, flexible tubes placed into the body, ending in the superior vena cava (SVC), for people who require frequent access to their bloodstream. These devices are most often used for the administration of medications such as antibiotics, chemotherapeutics, fluids and nutritional compounds, transfusion of blood products, and for multiple blood draws for diagnostic testing. Compared to regular intravenous (IV) lines, which are usually inserted in the small veins of the hand or arm, central venous catheters avoid many

problems associated with the administration of strong medications through small veins, including inflammation, scarring, and blood clots.

Central catheters are commonly placed in the internal jugular vein (IJV) of the neck, subclavian vein (SV) of the upper chest, or less commonly, in the femoral vein of the groin. An alternative to this approach are peripherally inserted central catheter (PICC) lines, unlike traditional central catheters that are placed in the neck, chest, or groin, which are not inserted directly into the central vein, but rather into the veins of the arm, advanced into the SV, ending in the SVC. In the arm, these lines can be placed in the superficial veins in the antecubital fossa at the elbow where the nurse or physician can directly see the vein, or in deeper veins further up in the arm. The basilic vein and the brachial vein in the upper arm, for example, are not directly visible, and therefore a PICC line placed in the upper arm requires US guidance. Placing PICC lines in the upper arm requires more training than in the antecubital fossa, but has several advantages. Placement in the upper arm is more acceptable to patients, interferes less with activities of daily living, and has a lower catheter failure rate (27).

The recent trend in miniaturization of inexpensive US machines has prompted an increase in US-guided PICC line placements at the bedside by trained nurses, but the success rate remains low compared to PICC lines placed by interventional radiologists. Improving this success rate would help alleviate the bottleneck posed by PICC line placements in the IR suite, reduce the risk of transporting patients, and decrease health care costs (26).

Many factors limit performance in CUS needle guidance. For example, freehand US needle guidance involves a displaced sense of hand-eye coordination, where the operator looks away from the operating field to see the US monitor. The operator holds the US probe in one hand to scan the body, while the other hand holds the needle. After locating the target of interest

on the US image, the operator inserts the needle tip just below the surface of the skin. Then, the needle is advanced toward the target while simultaneously looking back and forth between the US monitor and operating field, and sweeping the US probe back and forth to continually visualize the needle tip location.

This dissertation investigates the Sonic Flashlight (SF) as an alternative to conventional US guidance (CUS) for vascular access. The SF is an augmented reality display system that uses a small monitor and half-silvered mirror mounted on the US transducer. Looking through the half-silvered mirror, a virtual image of the US displayed on the small monitor is seen *in situ*, beneath the surface of the skin, exactly where the data are being scanned. The operator can then aim the needle directly at the target of interest, circumventing the displaced sense of hand-eye coordination. By making these procedures more intuitive, procedure time, and thus patient comfort may be improved. A factor in the decreased procedure time may be a reduction in the number of "sticks" before success.

Calibration of the SF is a very critical procedure for the SF to ensure that the US virtual image is seen at the actual location of the scanned anatomy. Without a properly calibrated system, a SF user might think she is aiming for the IJV, but upon inserting the needle, find the needle entering the carotid artery, resulting in thromboemboli to the brain leading to severe injury or death. Calibration can be performed by applying an affine transform to rotate, translate, and scale the US image on the SF display. This dissertation describes a new method for calibration, called thin-gel calibration, which aligns the virtual image of the US display with the actual object being scanned in a thin slab of US-conducting gel. Using vascular access US phantoms, we show that vascular access with the SF is easier to learn than CUS, and that even users already proficient in CUS guidance perform vascular access faster with the SF, despite

having no prior experience with the SF. Finally, we demonstrate SF guided vascular access in a cadaver.

1.2 CLAIMS

Claim 1. The 5th generation handheld Sonic Flashlight has sufficient scanning resolution, display resolution, and sufficiently low system lag to be used for vascular access.

Claim 2. Accurate calibration of the Sonic Flashlight can be obtained using a method called thin-gel calibration.

Claim 3. Vascular access is easier to learn using the Sonic Flashlight than conventional US guidance.

Claim 4. Without any prior experience using the Sonic Flashlight, users proficient in conventional US guided vascular access perform vascular access faster using the Sonic Flashlight than conventional US.

Claim 5. The Sonic Flashlight can be used to gain vascular access in a cadaver.

1.3 OUTLINE

The next chapter describes in more detail the vascular access and the US guidance problem, followed by a survey of existing US technologies. It also elaborates on augmented reality methods that researchers have applied to medical imaging, including previous versions of the SF. Chapter 3 introduces the current prototype of the SF, followed by a new method to calibrate the SF in Chapter 4. Chapter 5 encompasses the validation work of this dissertation, followed by a summary of this work, concluding remarks, and future directions in Chapter 6.

2.0 BACKGROUND

This chapter begins with a description of blind and CUS guided venous access. Next, we describe commercially available devices to aid in vascular access. We then survey the state of the art augmented reality medical display systems relevant to this dissertation, including Real-Time Tomographic Reflection (RTTR), the concept behind the SF. Finally, we study the developmental history of the SF.

2.1 BLIND CENTRAL VENOUS ACCESS

Central venous access has traditionally been obtained blindly, without US guidance, using the surface anatomy and palpation to locate the target vein. For example, the blind Seldinger technique, using an infraclavicular approach to the SV, begins by identifying the juncture of the medial and central thirds of the clavicle. An introducer needle is then "walked" from this juncture under and around the clavicle towards the sternal notch while aspirating back with the syringe, which is held parallel to the chest wall. Free blood flow into the syringe indicates entry into a vessel. Bright red, pulsating blood return indicates entry into the subclavian artery, signaling the need to withdraw and redirect the needle until the SV is accessed.

After the introducer needle is located in the SV, a flexible guide wire is inserted through the needle and into the vein, so that the tip of the guide wire sits inside the SV while the end of

the guide wire is external to the body. Next, the introducer needle is removed, while ensuring that the guide wire tip remains in the SV. An incision is made next to the guide wire to allow passage of the catheter. Then a dilator is advanced over the guide wire, followed by the catheter. Finally, the guide wire is removed, and the catheter is sutured into place. A similar technique is used in the neck for gaining access to the IJV.

Most of the complications encountered during these procedures occur during insertion of the introducer needle into the vein. In gaining access to the SV as describe above, the needle may penetrate critical structures surrounding the SV. Puncturing the subclavian artery may result in hemorrhaging and/or hemothorax, characterized by blood in the pleural cavity around the lung. Puncturing the parietal pleura of the lung, possible if the needle penetrates through both walls of the SV, may lead to a pneumothorax. Needle damage to the nerves of the brachial plexus or phrenic nerve can lead to motor or sensory deficits of the hand, arm or shoulder, or paralysis of the ipsilateral hemidiaphragm. In gaining access to the IJV in the neck, the needle may puncture the carotid artery leading to cerebral thromboembolism.

2.2 CONVENTIONAL ULTRASOUND GUIDED VASCULAR ACCESS

US guidance has been shown to increase patient safety, comfort, and success rate of central line placements in a convincing number of studies (10, 13, 16, 18, 24, 34, 40-42, 45). With the advent of lower-cost, portable US machines, there has been a push to increase vascular access guided by US (15). US guided vascular access begins with scanning the area of interest to locate the target vein: the IJV in the neck, the SV in the upper chest, and the brachial or basilic vein of the upper arm. In most cases, veins are differentiated from arteries by their ability to be

compressed. Typical blood pressure in a vein is ~8 mmHg, compared to 80 to 140 mmHg in an artery. Pressure from the US probe on a vein easily overcomes the internal venous pressure, causing the veins to collapse, while the same pressure on an artery does not overcome the internal blood pressure, which therefore does not easily compress. It is important to mention, however, that pathologies exist that can render veins stiff and incompressible, such venous stenosis.

The Seldinger technique described above for placing central lines can be combined with US guidance. Holding the probe perpendicular to the patient, oriented to scan a cross section of the vessel, the operator identifies the vessel. The vessel is then centered horizontally on the US screen and the depth, d, of the vessel is measured. Starting distance d away from the scanning plane, the needle tip is inserted at a 45-degree angle relative to the probe. The probe is moved towards the insertion site until the needle tip is visualized on the US. Next, the needle is advanced forward while moving the US probe accordingly to keep the needle tip in view, correcting the needle trajectory to intersect the vein. When the needle tip is visualized centered over the vein, the needle is advanced into the vein. Free blood flow confirms entry into the vein. From this point, the procedure continues as described in the blind procedure.

The operator has to focus his/her attention upon two areas that are not necessarily adjacent - the operating field and the US screen - requiring the operator to shift his/her attention back and forth (Figure 2.1). With normal hand-eye coordination, a person can see their hand and/or hand-held instrument while performing a touching task. Asking a person to guide their hand or hand-held instrument to a particular target location is an example of a touching task. In CUS guided procedures, the operator cannot see the instrument and target simultaneously, nor can he see the instrument tip directly after insertion below the skin. After insertion below the

surface of the skin, the instrument tip is only visible through the US scan. We believe that this displaced sense of hand-eye coordination in addition to the requisite attention shift are the chief reasons why CUS guided procedures are difficult to learn.

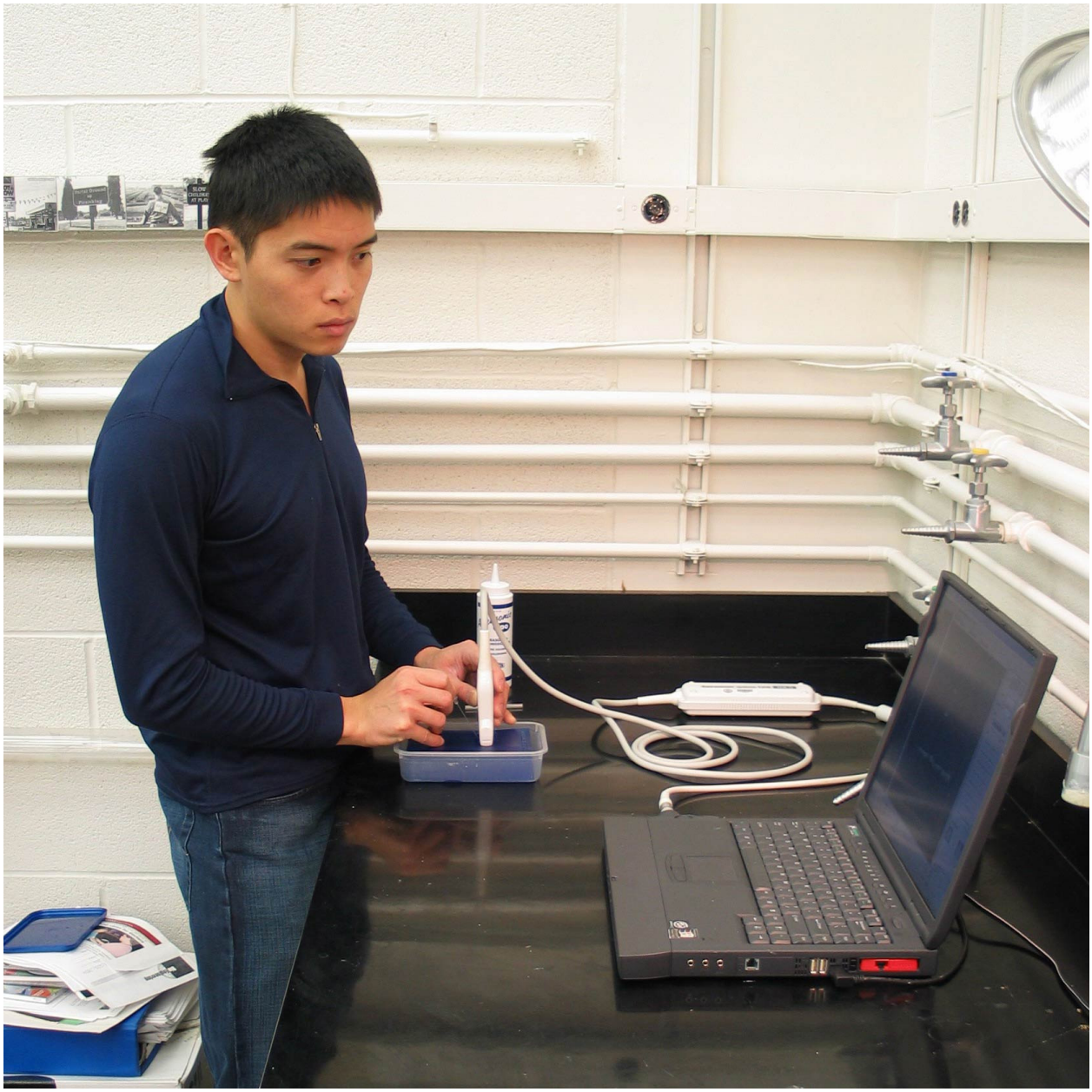


Figure 2.1 CUS vascular access on a phantom

Notice how the operator's attention must shift between the US and operating field

2.3 CURRENT ULTRASOUND SYSTEMS USED FOR VASCULAR ACCESS

In this section, we will survey current commercially available US systems used for vascular access. We start by looking at systems that do not produce the traditional B-mode images, but rather use Doppler shift to convert flow into an audible signal, indicating the presence of a vessel directly in front of the needle. Next is the group of traditional US machines, which perform B-mode scanning to produce an image displayed on a monitor. This section is followed by a short description of in-plane needle guides that are fixed to the US probe. Finally we describe variations on traditional US machines that have been designed with vascular access in mind.

2.3.1 Doppler Flow Methods

The Doppler US guidance methods include audio-guided Doppler (11), fingertip pulsed Doppler (5) and probe-in-the-needle technology (4, 11, 12, 43). These are all fairly similar, using an US Doppler flow meter to indicate the location of a vessel to the user, so the description of one system is sufficient. The Smart Needle (Peripheral Systems Group, ACS, Mountain View, CA) is a Doppler sensor that fits into the lumen of a standard 18-gauge needle. It uses a 14 MHz US beam to continuously detect flowing blood directly in front of the needle tip, with flow quantified by a series of LEDs on the monitor portion of the device as well as an audible signal. To gain access to the vessel, the needle is advanced towards the direction of maximum signal intensity, and when the needle penetrates a vessel, a marked increase in the signal occurs and blood flows into the needle hub. Arterial flow is identifiable by a pulsatile, higher frequency sound, while venous flow is identified as a lower frequency with pulsatility varying in intensity with respiration. It is debated in the literature whether there is an increased success rate with a

decreased number of complications using these Doppler flow devices to perform central venous cannulation, especially in populations other than children (4, 6, 21, 28).

2.3.2 Traditional Ultrasound Machines

This class of devices is by far the most common US machine today, producing a two-dimensional B-mode image of the anatomy. In the late 1970s, based on sonar technology, real-time US imaging to produce grey-scale acoustic images was introduced. Numerous brands exist currently, including GE, Siemens, Acuson, Philips, and Sonosite, and for the purpose of this dissertation, all of these systems are considered more or less comparable. These systems typically operate in the 2-15MHz range, produce a planar scan in front of the probe, with the US image displayed on a CRT or LCD monitor on the US machine. The CUS guided vascular access described in the previous section assumes the use of one of these conventional systems.

2.3.3 In-Plane Needle Guides

Typical needle guides, called *in-plane needle guides*, are attached to the US probe and restrict the needle to travel along a specific path in the US plane. They have been routinely used to perform needle biopsies of organs, including the liver, kidney, prostate, and breast (19, 20, 29, 44). The needle pathway is indicated on the monitor by means of guide lines superimposed on the US image. While steerable in-plane needle guides are currently being developed (14), the needle is still restricted to travel in the scanning plane (as opposed to the Site-Rite system described in section 2.3.5).

2.3.4 PunctSURE

The PunctSURE vascular access imaging system (Inceptio Medical Technologies, L.C., Kaysville, Utah) is a variation on traditional US systems, producing real-time cross-sectional and longitudinal B-mode scans simultaneously. The system also features a hands-free magnetic and adhesive mount to secure the probe to the patient, allowing the operator to use both hands to perform the procedure. With the vein centered on the cross-sectional scan, the longitudinal US array is properly oriented parallel to the vein, with the cross-sectional array orthogonal to the vein. Once the probe is properly positioned, the needle is inserted in the plane of the longitudinal scan, allowing the operator to visualize the needle along its entire trajectory. Traditional US machines can also visualize the needle along its entire trajectory by rotating the probe by 90 degrees on the patient, but the PunctSURE system provides it without rotating the probe.

2.3.5 BARD/Dymax Site-Rite Needle Guide System

The Site-Rite needle guide system shares several similarities with the in-plane needle guides described previously. The needle guide attaches to the US probe, restricting the needle to a specific pathway. The Site-Rite needle guide, however, is an out-of-plane needle guide (Figure 2.2). Instead of visualizing the needle along its entire pathway, a Site-Rite needle guide specifies a particular location where the needle will intersect the US scanning plane, ranging from 5 mm to 35 mm in depth in 1 cm increments depending on which of 4 disposable needle guides is attached to the probe. After guidance of the needle into the vein, the needle guide can be separated from the needle, facilitating insertion of a catheter.

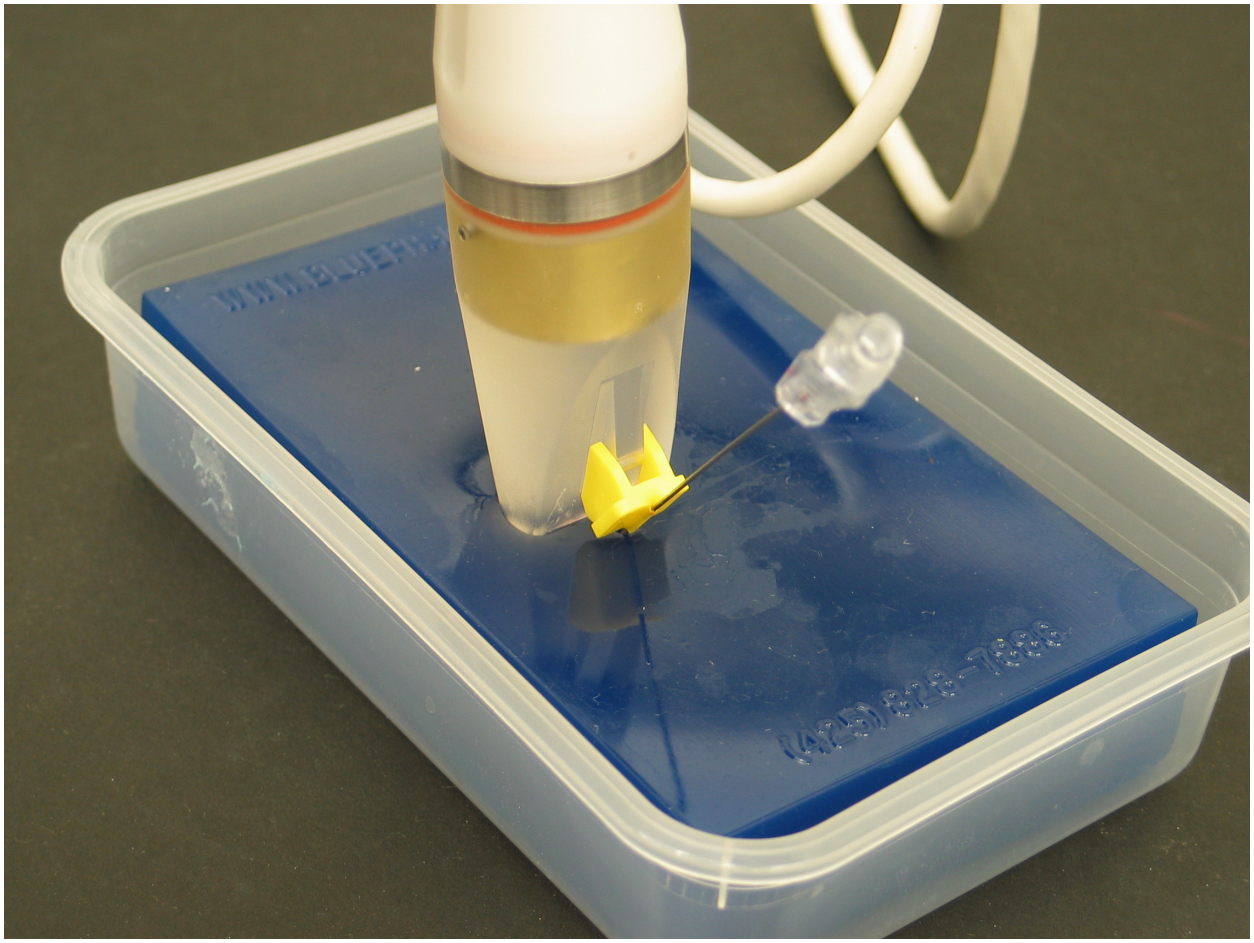


Figure 2.2 Site-Rite 3 out-of-plane needle guide

One problem noted by our institution's interventional radiologists and IV nurses with this needle guide is bending of the introducer needle during insertion into human tissues. The standard 21-gauge introducer needle is flexible enough to bend when interacting with the inhomogeneities of human tissues, resulting in an unpredictable intersection point between the needle and the US plane. Another problem encountered with the needle guidance system is that removing the needle guide from the needle occasionally causes the needle tip dislodge from the

lumen of the vessel. The needle guide contains a tube through which the needle passes. A lengthwise slit runs down the entire length of the tube that the needle is pulled through when removing the needle guide. During this process the needle sometimes dislodges. Due to these two problems with the design, most operators use the Site-Rite US system without the needle guides, reducing the system into a traditional US machine with the previously described CUS guidance technique.

2.4 AUGMENTED REALITY

Augmented reality display systems make use of either head-mounted displays (HMDs) or optical instruments to overlay computer-generated (CG) virtual images onto the user's view of the real world. In this section we will review a number of such systems that have been used to merge US, magnetic resonance imaging (MRI), and computer aided tomography (CT) images with normal human vision, leading up to RTTR and early versions of the SF.

2.4.1 Head-Mounted Displays

HMDs are a class of devices that the user wears on his head and looks through in order to see augmented reality. The HMD tracks head movement relative to the environment and/or instrument. Typically, these systems require precise calibration, rapid head tracking, and high resolution displays to maintain the illusion. HMD systems are generally divided into two major categories: optical see-through systems and video see-through systems.

Optical see-through HMD systems operate under the principle that the user sees the real world through a half-silvered mirror through which the CG images are superimposed. To

provide the illusion of a medical image floating within the patient, the system must track the relative position of the HMD user's head, and for US, the location of the probe. For non-real-time images, the system must also track the position of the patient to register the image correctly. Based on their relative locations, the system must calculate the proper image to present to each eye of the HMD user. The main difficulty in these systems is registration. To merge with the real-world environment, the CG image must be continually registered with respect to real-world coordinates. To present a convincing illusion, registration of the CG image must be very accurate since the human eye is sensitive to registration errors of even a few pixels. Much of the research in optical see-through HMDs focuses on correcting image-registration errors (1, 2, 17, 25). Due to these registration issues, most of the current research to develop HMDs to display US use video see-through.

Video see-through HMD systems use a video camera mounted on the HMD to capture what the user would be seeing. The video capture is merged with the CG images and sent to the HMD for the user to see. Instead of looking at the real world directly, as in optical-see through systems, video see-through systems present live video images of the real world environment on small monitors in front of the eyes. These systems track the location of the US transducer relative to the HMD so that the US image can be computed and overlaid on the video image in the HMD with the correct stereoscopic perspective and location. Video see-through systems have many similarities to optical see-through systems, including tracking of the user and US probe, as well as accurate registration to achieve a convincing illusion.

Several research systems have been developed that use video see-through HMDs to present the user with US images (9, 31, 32, 35). Highlighting some of the issues typical of HMD systems, the HMD breast biopsy system at University of North Carolina has several

drawbacks as noted by the authors: the tracked probe is sometimes cumbersome, the user must attempt to keep the head-tracking landmarks in view, the pixel resolution of the US slice as it appears within the HMD is poor, the weight of the video see-through HMD is excessive, the system lag is unacceptable (9), and the system too expensive. Furthermore, if multiple observers are cooperating in a procedure, each observer requires a separate HMD to observe the same *in situ* US image. These drawbacks apply to both optical and video see-through systems. Some of the problems will undoubtedly be addressed in time, but the HMD system remains an inherently complex and challenging approach.

2.4.2 Image Overlay

Image Overlay is an optical see-through augmented visualization method that does not require a HMD (3) to display pre-operative CT images to the user in total hip replacement surgery (8). It is comprised of a semi-transparent mirror, a monitor, three positional trackers, and an optional set of LCD shutter-glasses. The user looks through the half-silvered mirror to see a reflection of the monitor (Figure 2.3). The shutter glasses, which alternately occlude each eye, are synchronized with the monitor to present different images to the user's right and left eye (Figure 2.3). The images that are presented are based on the position and orientation of the user's head relative to the location of the patient, both of which require a tracking device. This system is currently being adapted for use in computer-assisted total knee replacement surgery. Image overlay represents a middle ground between HMDs and the SF.

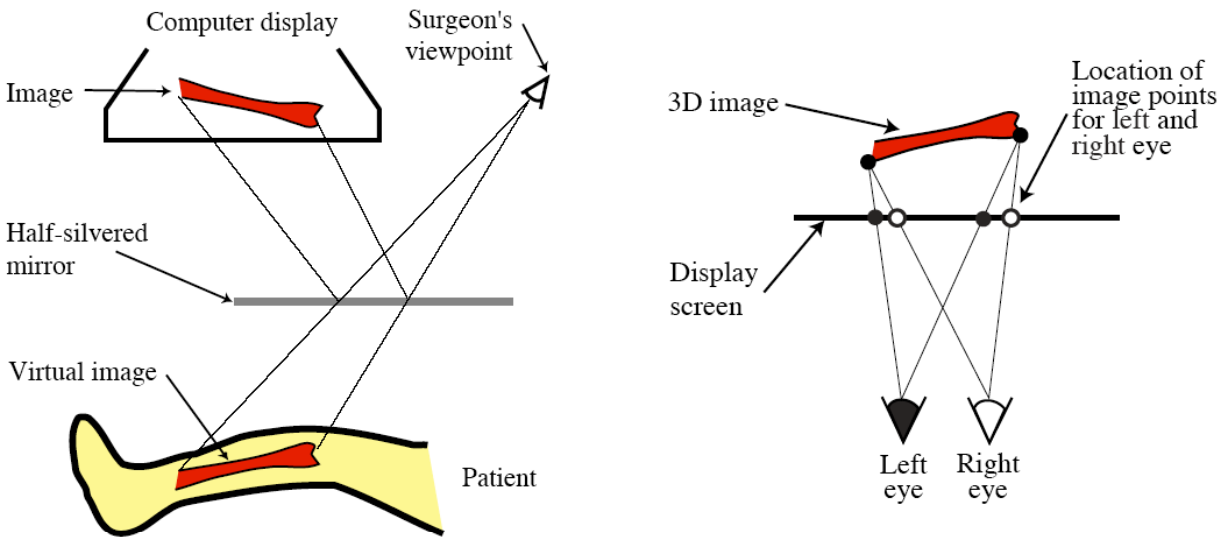


Figure 2.3 DiGioia's Image Overlay System

(Left) Using a half-silvered mirror to produce a "virtual" display.

(Right) Image points on the display to produce stereoscopic effect.

Artist: Branislav Jaramaz

2.4.3 Real-Time Tomographic Reflection (RTTR)

RTTR, the concept behind the SF, is a simplification of DiGioia's image overlay system that merges real-time US images into the perceptual real world without positional tracking or a HMD system (7, 37). RTTR fixes the relative geometry of the imaging device, display, and a half-silvered mirror to produce a virtual image of the tomographic slice (i.e. US) inside the patient (Figure 2.4). Each pixel of the virtual image appears to emanate from its correct anatomical location within the patient. The virtual image is stable and independent of viewer location, meaning that all users looking through the mirror from any vantage point will see the virtual image properly registered with the internal anatomy. The SF is the US application of RTTR.

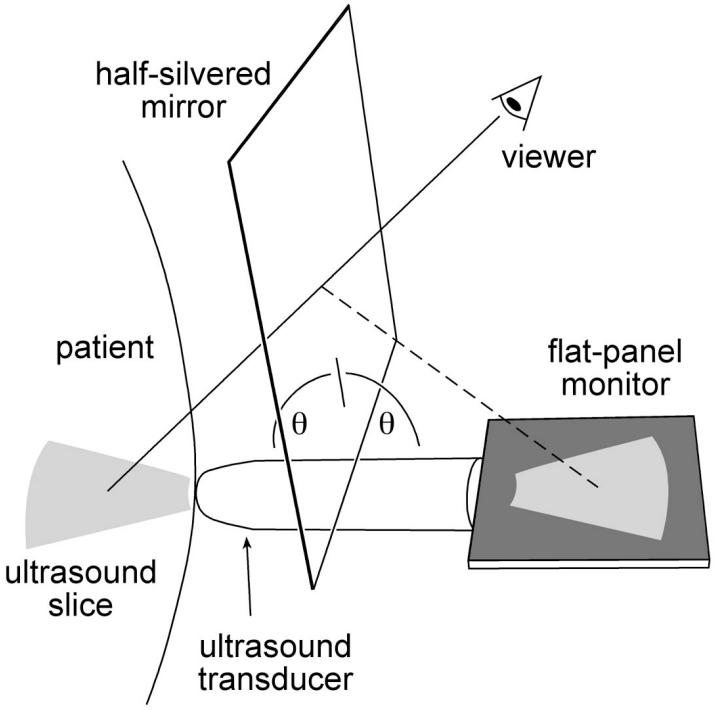


Figure 2.4 Sonic Flashlight optics overview

The SF requires no positional tracking or special eyewear, setting it apart from all previously described augmented reality systems. Therefore, it has significantly less complexity and computational requirements. No head mounted apparatus and less complexity results in a relatively lightweight, low cost system with low lag time and high resolution. Calibration of the SF, assuming a properly located planar US scan and monitor, consists of displaying the US image with the correct orientation, scale, and placement on the monitor. Theoretically, the SF only needs to be calibrated once, although in practice such calibration would probably need to be confirmed periodically. Rough methods of calibration during early development of the SF included eye-balling the US virtual image with the view of a water balloon, and a water-filled calibration tank that contained a set of 3 light-emitting diodes (LEDs) inside of hollow tubes

(38). These systems will be described in detail in section 2.4.3.2. A more exact calibration method has been developed as part of this dissertation.

Masamune, et al., have developed a system using RTTR, which they call "Slice Image Overlay," for displaying CT data (22, 23). This system was previously described by Stetten (36). The monitor and half-silvered mirror are mounted directly to the CT scanner gantry. Looking through the half-silvered mirror at the patient, the user sees the CT image appearing to float within the patient at the correct size and location. Since CT is generally not a real-time imaging modality, the patient must remain motionless after imaging for the registration to stay valid. Also there is no real-time imaging feedback of the operating instruments during invasive procedures.

2.4.3.1 Previous Versions of the Sonic Flashlight The first prototype of the SF was a floor-mounted model with a 2'x3' half-silvered glass mirror, a 15" LCD monitor, and a 3.5 MHz US probe (Acoustic Imaging 5200) mounted to an 8' triangular metal frame (Figure 2.5). While this first prototype proved the concept of RTTR, it also highlighted the need for a handheld system that the operator could freely manipulate. This motivated our first and second handheld systems, which used a 30 cm x 30 cm mirror, and a 5" LCD monitor. While it could be operated by one hand, one of our clinical consultants pointed out that the "pistol grip" that we built on the system was not well suited for the fine motor skills required for handling an US probe (Figure 2.6).

Figure 2.5 Floor-mounted Sonic Flashlight

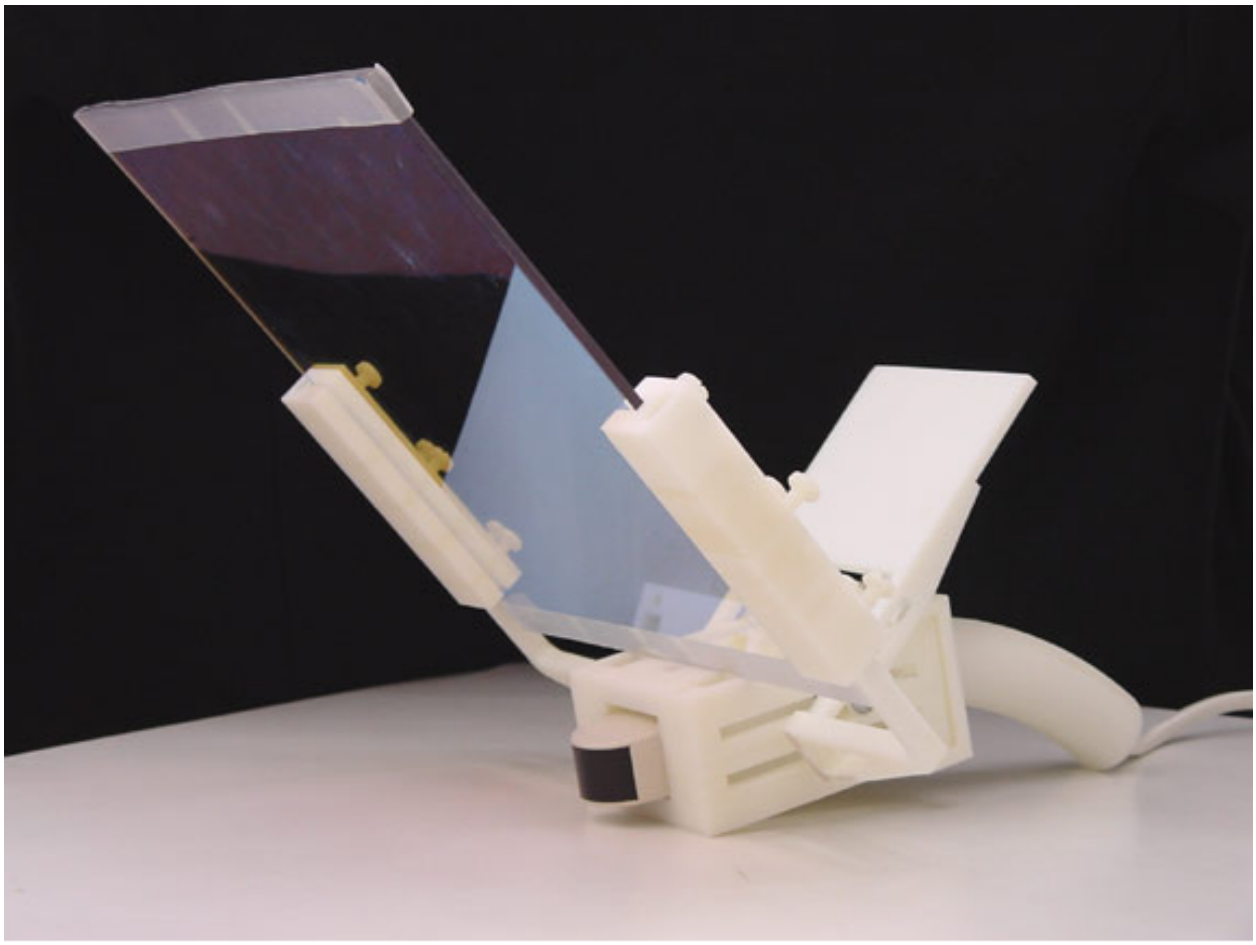


Figure 2.6 Pistol-grip Sonic Flashlight

Therefore the 4th generation SF had a "pencil grip" that could be manipulated more like a traditional US probe while scanning a patient. Unlike previous versions of the SF, this system was built around a portable US system, the 7MHz 50S Tringa Vet (Pie Medical, Maastricht, Belgium). It also featured a field emission display (FED) (FE524G1, PixTech, Santa Clara, CA), rather than a LCD display, which offered better brightness and off angle viewing properties. The analog real-time US video signal was sent through an analog-to-digital converter to a standard laptop computer (Dell Insipron 8200, Dell Inc., Round Rock, TX), which provided the

processing necessary to properly scale, orient, and locate the image on the screen. While this was the first system to be portable, in the sense that it could fit into a suitcase to be moved, the drawbacks of the system included relatively poor scanning resolution of the US machine, poor contrast of the FED display, a perceptible lag in the system, and significant weight. Despite these limitations, we performed the first SF guided invasive procedure, a retrobulbar injection, in a cadaver (7). The next and most recent prototype was developed as part of this dissertation, and will be described later.

2.4.3.2 Previous Methods of Calibration Clearly, an accurate method for calibration is required. Without it, the SF would be unable to guide interventional procedures (38). Calibration requires careful consideration of the degrees of freedom in the registration process. The challenge is to make each pixel in the virtual image occupy, and therefore appear to emanate from, its actual 3D location in the slice. We first consider only the *geometric transform* of a rigid body; i.e., we assume that the US slice is displayed without distortion at its correct scale on a perfectly flat monitor (Figure 2.7). The geometric transform that would be required to superimpose the virtual image onto the slice can be represented as two sets of translations and rotations, each of which has 3 degrees of freedom. The first (two rotations and one translation) allows the flat panel display to be moved into its correct plane, making the virtual image coplanar with the actual US slice. We can achieve this by physically moving the display, the US transducer, or the mirror. The second (two translations and one rotation) can be achieved simply by adjusting the image of the US slice on the flat panel monitor.

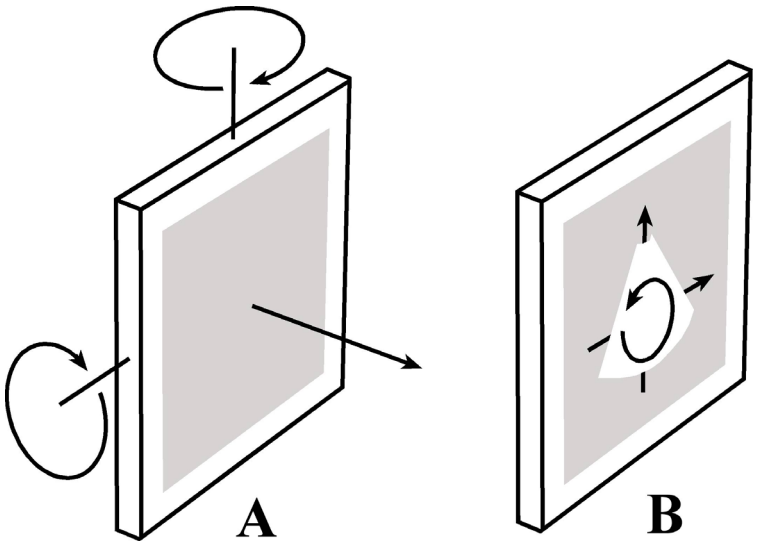


Figure 2.7 Geometric transforms for calibration
(A) physically moving the display, and (B) moving the image on the screen.

The assumption that both the US slice and the flat panel monitor are truly *flat* is a safe one, and makes a rigid-body (geometric) transform sufficient to move the monitor into the correct plane. However, correctly adjusting the image on the screen actually requires more degrees of freedom than those provided by the geometric transform, if only because the image must be scaled to its proper size. Extending the geometric transform to an *affine transform* allows us to scale the US slice to its correct size, as well as adjust its aspect ratio and correct for skewing. Most computers today are capable of mapping an image through an affine transform very rapidly, using a specialized piece of electronic hardware called *2D texture memory*. In our RTTR system, texture memory permits the displayed US slice to be translated, rotated, and anisotropically scaled. The calibration process becomes a matter of finding the optimal parameters for that affine transform. Mapping location (x, y) to (x', y') with an affine transform is accomplished by multiplying the homogeneous form of (x, y) by a 3 x 3 matrix **A**.

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ 0 & 0 & 1 \end{bmatrix}$$

where

$$\begin{bmatrix} x_1' & x_2' & x_3' \\ y_1' & y_2' & y_3' \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ 1 & 1 & 1 \end{bmatrix}$$

An affine transform is capable of mapping any triangle onto any other triangle. If the apices of both triangles are known (and not colinear), we can find an explicit solution for the 6 unknown elements of the matrix **A**.

The first calibration device based on this theory consisted of a phantom with 3 training beads, suspended in roughly an equilateral triangle, on threads, in the plane of the US slice. The threads were held in the plane of the slice by a frame, attached directly to the transducer (Figure 2.8). An uncalibrated US image of this phantom in a water tank was captured and displayed on the flat panel monitor. The water was subsequently drained to avoid refraction at the air-water interface during visual inspection. The translational errors in the location of each bead could then be used to solve for the affine transform matrix **A**. Assuming a coplanar slice and virtual image, the calibration process is independent of viewer location. Further calibration beyond the affine transform may be required to correct for non-linear geometry, in both the imaging system and the display by warping the image with a non-linear transform. To the extent that the slice geometry does not change with tissue type, and therefore remains constant as the transducer is moved relative to the target object, calibration of the system may only need to be performed initially, using a phantom such as that just described. This initial calibration will suffice for that

part of the image geometry that is fully definable, relative to the scanner. Further changes in image geometry may be due to changes in the speed of sound in different tissue types and to refraction of the US beam at discontinuities in acoustic impedance. It may be possible to correct for some of these using image analysis techniques that rely on tissue characterization, but that is beyond the scope of this dissertation.

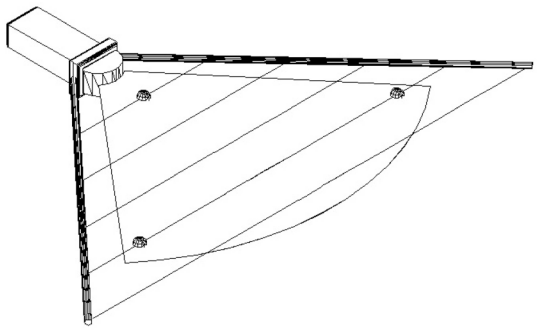


Figure 2.8 Water-tank calibration phantom
3 beads suspended on strings in the plane of the ultrasound slice

The bead phantom in a water tank (described above) has the disadvantage of requiring the water to be drained from the tank to avoid refraction at the air-water interface. For calibration purposes, a variety of gels have traditionally been used to fill US phantoms. A phantom was previously developed with the particular needs the SF in mind, which could be used to calibrate the display despite an inability to visually inspect the phantom's interior. The LED-tube calibration phantom is shown in Figure 2.9 with the earlier "pistol-grip" prototype SF scanning through a latex diaphragm (not shown) in the wall of the phantom. The LED-tube phantom

contained 3 small beads fixed on thin rods. Three corresponding hollow tubes ("virtual line-of-sight tubes") were mounted on the exterior of the phantom such that each tube pointed at its corresponding bead. A small LED was inserted into the base of each tube so that, even without seeing the bead, the viewer could determine that he was looking directly at the bead by seeing the LED through the tube. The US images of the beads could then be aligned, each bead with its corresponding LED, and the affine transform could be found, as before. This system has several drawbacks. Changing the configuration of the calibration targets requires refabricating the entire calibration device. This means that there is no simple way to use the calibration device on different target configurations to measure calibration accuracy. In addition, the system offers no direct visual confirmation of the location of the targets, in the sense that the user cannot actually see the targets that are being scanned.

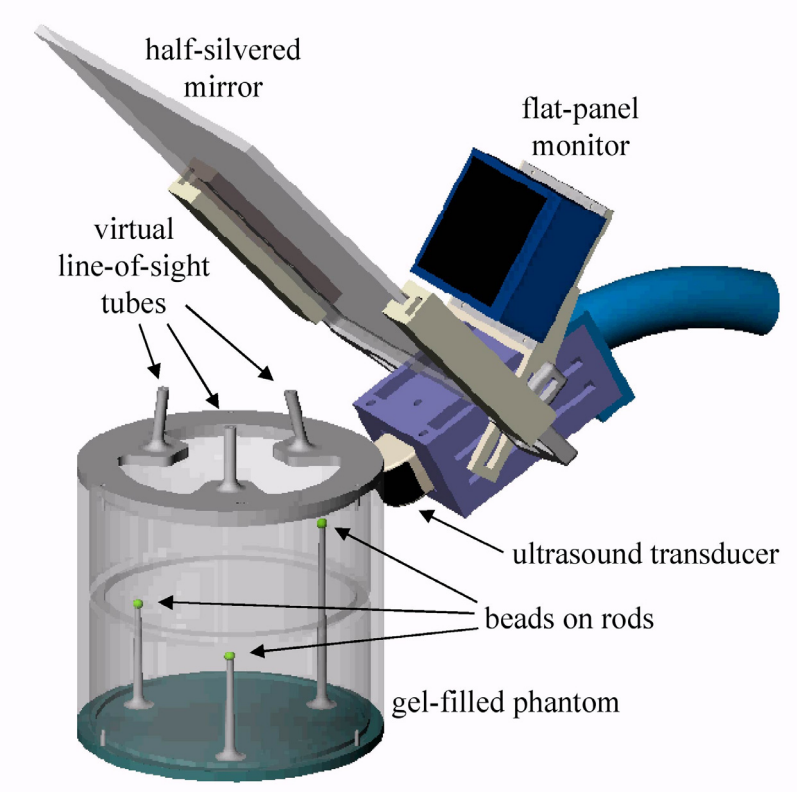


Figure 2.9 LED-tube calibration phantom

Spherical targets with virtual line of sight tubes emitting light collinear with each target

3.0 SONIC FLASHLIGHT PROTOTYPE VERSION 5

The major design goal for the 5th generation handheld SF was to make it usable for the first clinical trials. Based on observations and comments from our clinical consultants, a clinical device would require a combination of reduced size, weight, and system lag, with better scanning resolution, display resolution and contrast.

3.1 TERASON ULTRASOUND MACHINE

The US scanner used in the 5th generation SF is the Terason 2000 with a 10MHz probe (Terason, Burlington, MA), consisting of a transducer, a small (~6"x2"x1", 10 oz) electronic converter, and a standard laptop computer. The electronic converter executes the beam-forming while the scan-conversion is handled in software, a tribute to the power of current laptop computers. The Terason software runs on the laptop computer allowing software control over the US machine and access to the digital US image. Unlike previous SF systems, where the US image required an analog-to-digital conversion, the US image in the 5th generation SF always remains in the digital domain, which is inherently more accurate and stable. Terason has supplied a Software Development kit, which permits us to write custom software to use the digital US image in real–time (software code in Appendix A). The number of components in this SF system is reduced compared to previous SF systems since the Terason laptop computer can perform the task of the US machine as well as adjusting the size, orientation, and location of

the image on the SF display. By reducing the number of components in the system, lag is reduced to an unperceivable < 11 ms, significantly less than the estimated 100 ms lag in the previous systems. The US image contains 512 x 128 pixels of data per frame, and the system scans at ~22 frames per second.

3.2 SONIC FLASHLIGHT DISPLAY

The SF has essential and somewhat unusual display requirements. The display must be lightweight, small, bright when viewed off-angle, have high-resolution, and be able to display a high number of gray-scale and color levels. Good off-angle viewing is necessary since typical viewing angles of the SF's display are between 50 and 85 degrees from perpendicular. Few display technologies currently available meet these requirements.

Traditional CRT displays have excellent off-angle viewing but are far too large and bulky. Field Emission Display (FED) technology has an off-angle visibility similar to CRT displays, and can be made flat and thin. Unfortunately, currently available FED monitors have low resolution and generally poor ability to display a large number of gray levels. Also, they are not yet available in color. The FED in the 4th generation prototype (Pixtech FE524G1) is not acceptable for clinical use, and its production has actually been discontinued. Liquid crystal display (LCD) monitors, such as those used in laptop computers, are small, lightweight, high-resolution displays, with good gray-scale or color levels. The major drawback of LCD monitors is their poor off-angle viewing.

The display used in the 5th generation SF is an organic light-emitting-diode (OLED) display (AM550L, Kodak, Rochester, NY). OLED displays do not require a backlight, allowing

them to be thinner and brighter than LCD displays, with superb off-angle viewing properties. Images on these displays can be viewed at nearly 90 degrees. The AM550L OLED has a display area of 44 mm x 33 mm, weighs 8 g, and has a display resolution of 521 x 218 pixels. The digital US data contains 512 x 128 pixels, and can thus be displayed at 521 x 218 pixel resolution, with no loss of resolution or data.

3.3 GEOMETRY

The US probe and the flat-panel display are fixed at 80° on opposite sides of the 25 mm x 50 mm x 2 mm mirror (40% reflectance, Edmund Optics, Barrington, NJ) by a rigid mount (Figure 3.1, schematic in Appendix B). The Terason US probe upon which this version of the SF prototype is built measures approximately 16 mm x 54 mm x 92 mm with a 14 mm x 54 mm scanning footprint. The SF retains the same scanning footprint of 14 mm x 54 mm, with the entire device measuring approximately 44mm x 57 mm x 133 mm.

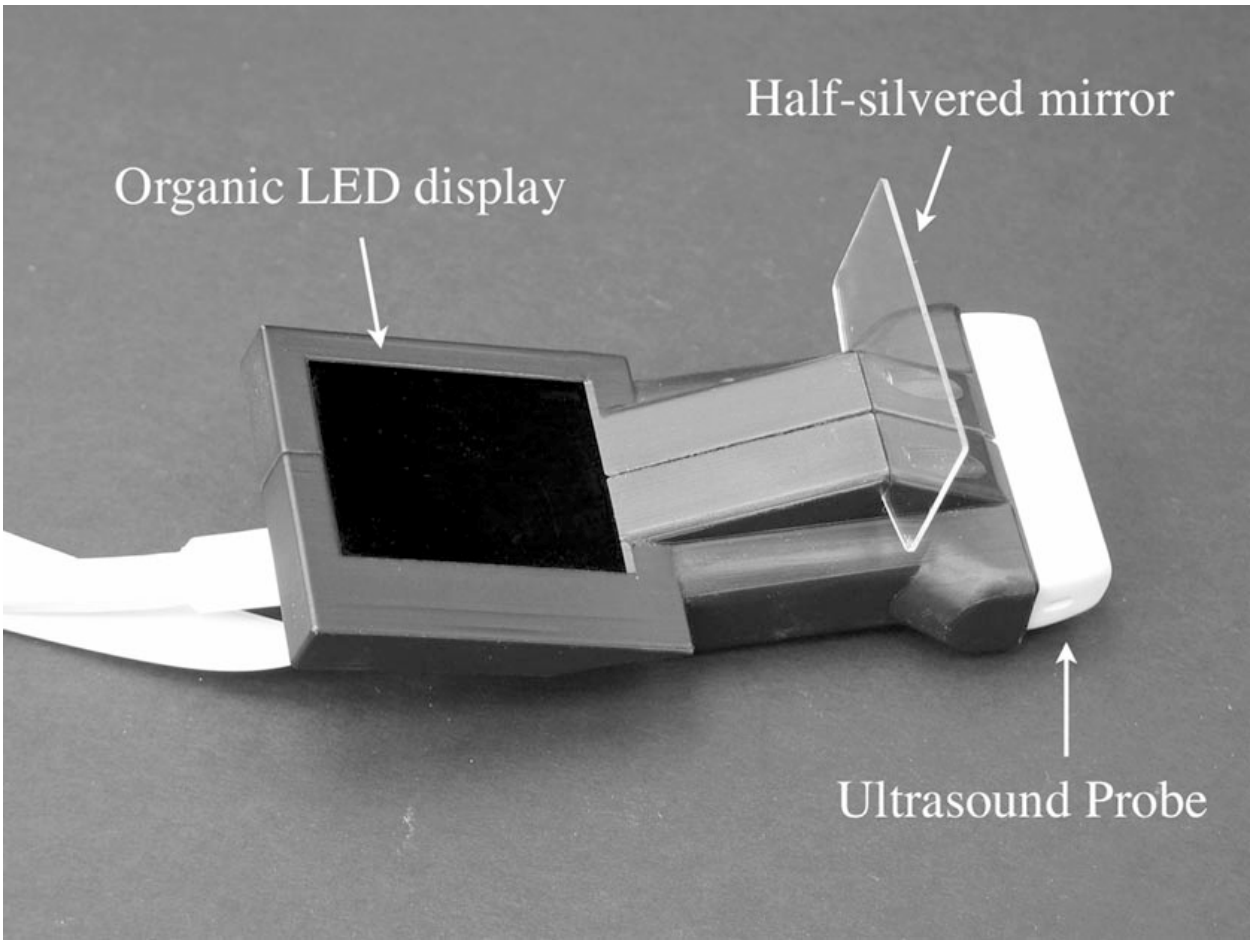


Figure 3.1 The Sonic Flashlight version 5

4.0 THIN-GEL CALIBRATION

The previous SF calibration methods required draining a water tank to avoid refraction at the air-water interface, or using LEDs within tubes to provide lines of site to the targets hidden within the phantom. Thin-gel calibration allows the user to directly see the targets and the US scan on the SF display simultaneously, unlike both the water-tank and LED-tube based systems. The thin-gel calibration system does not restrict the user to one viewpoint through the SF, as the LED-tube based system does. The thin phantom layer does not require capturing each US still frame and draining the water from a tank, and therefore allows calibration on a "live" US image. We will begin this chapter by discussing the design of the thin-gel calibration system, followed by a method to verify the US scanning plane and virtual image are coplanar using multiple viewpoints through the SF. Lastly, we will study the effect of target configuration within the thin-gel when performing the in-plane calibration.

4.1 THIN-GEL PHANTOM DESIGN

The thin-gel calibration system uses a thin layer of transparent phantom material along which the US machine scans, with small targets inserted transversely that are visible at the surface. When using the system to calibrate the SF, the user can simultaneously see these targets in the US image reflected by the SF as well as visually at the surface of the phantom. To

perform calibration, the user visually lines up the US image of the targets with the direct view of the targets by adjusting the location, scale, and orientation of the US image.

The transparent phantom material selected for the thin-gel calibration system is Tissue Mimicking Material™, the same material in the vascular phantom (Blue Phantom, Advanced Medical Technologies, Kirkland, WA) used in the validation studies (Chapter 5). A 1 mm layer of the phantom material sits on a 6 mm thick layer of clear acrylic plastic containing 17 1mm diameter holes placed in a specific pattern. Brass pins, 1 mm in diameter approximately 5 mm in length, are inserted through the thin-gel into 17 holes within the plastic, so that the tops of the pins sit in the middle of the thin-gel (Figure 4.1). The top surface of each brass pin, located within the middle of the thin-gel, is colored black for clear identification. A rigid mount holds the SF so that it's scanning plane is coplanar to the thin-gel phantom (Figure 4.2)

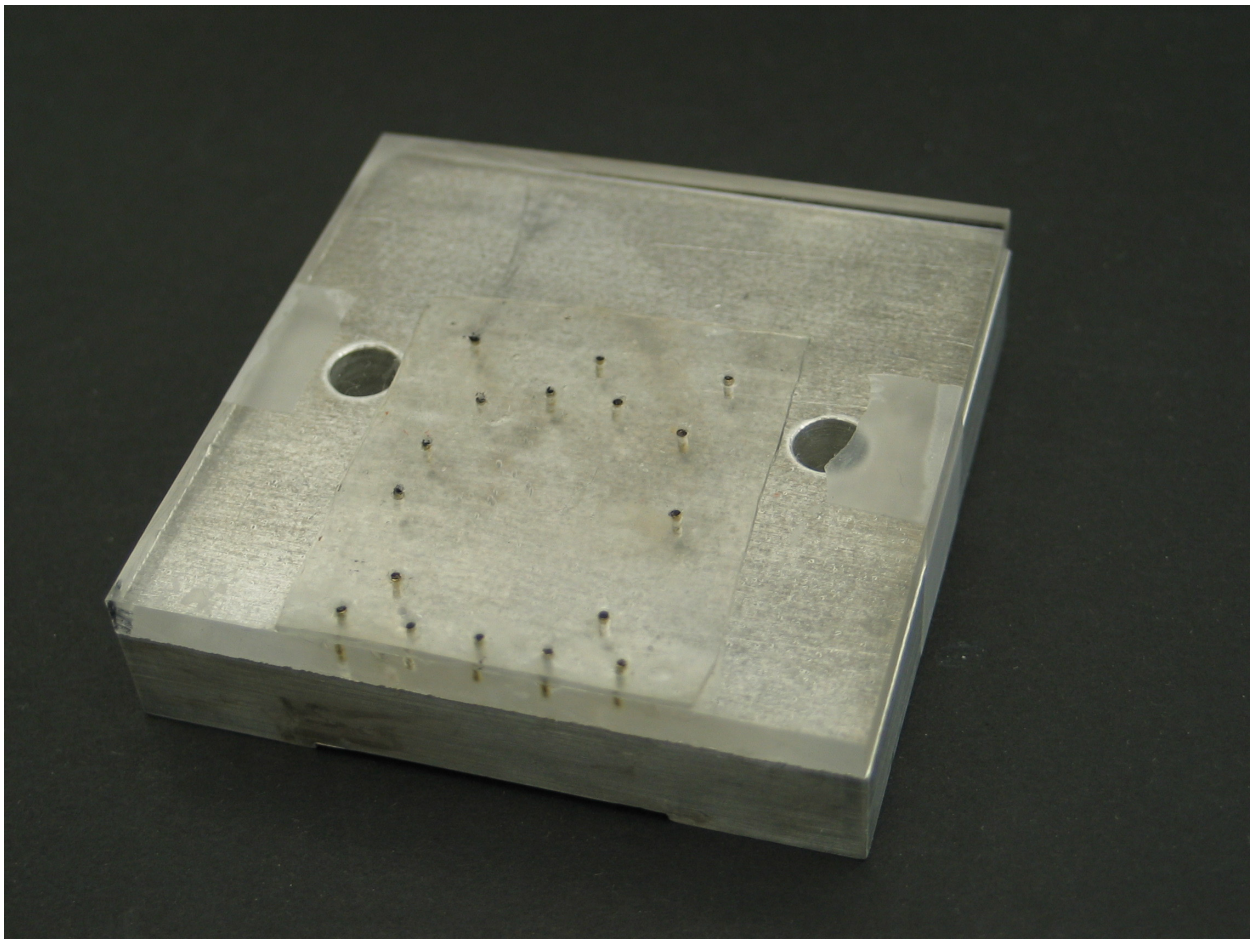


Figure 4.1 Thin-gel calibration phantom

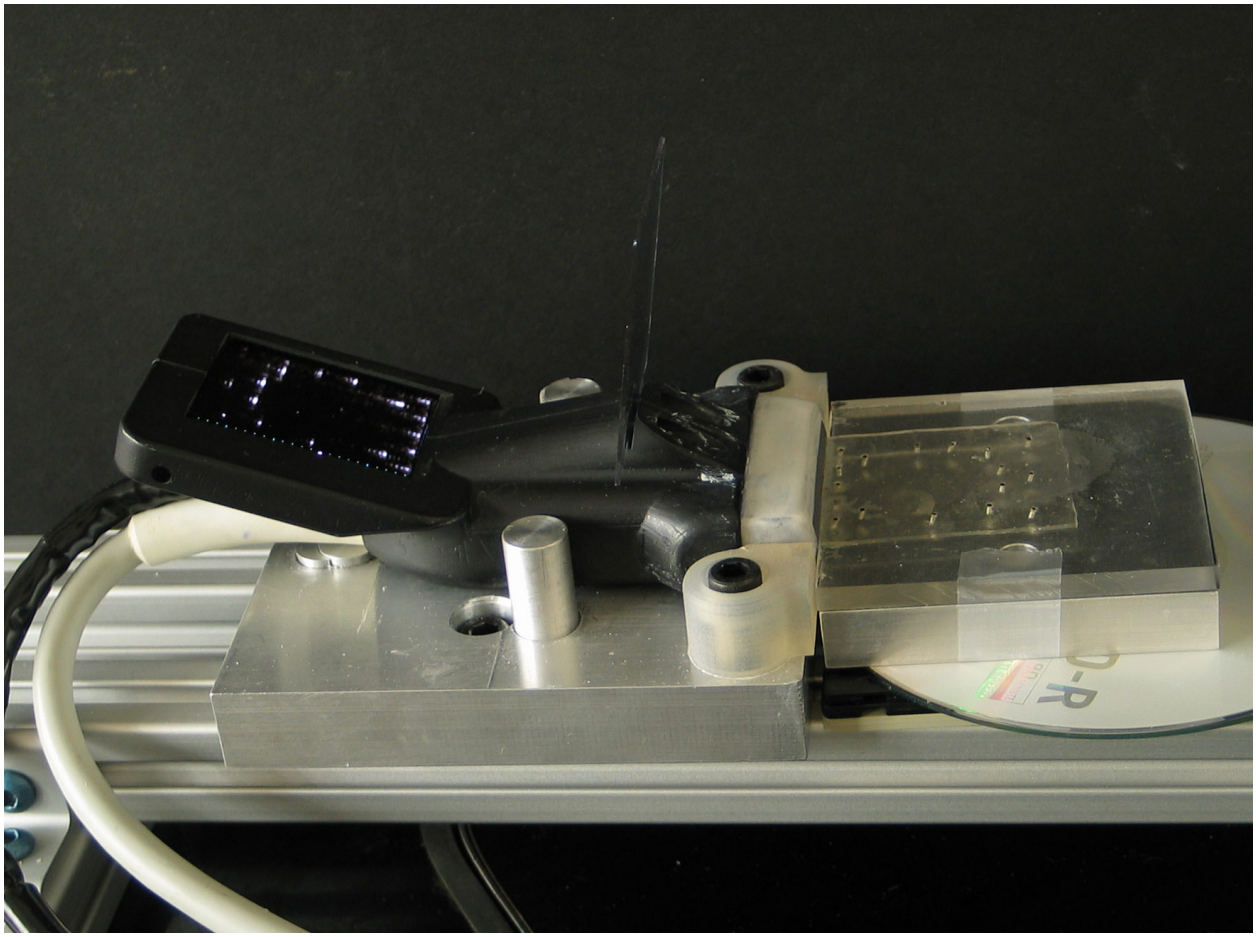


Figure 4.2 Thin-gel calibration setup

Three targets are necessary to generate an affine transform and calibrate the SF. Our previous calibration systems used a triangular configuration for the targets, but did not determine how target configuration affects calibration accuracy. To study the effect of target configuration on calibration accuracy, we arranged the targets in six unique configurations (A-F in Figure 4.3), with each configuration consisting of three pins to form a triangle (dotted lines in Figure 4.3). The six configurations fit into three categories: small equilateral triangle, isosceles, and large triangle.

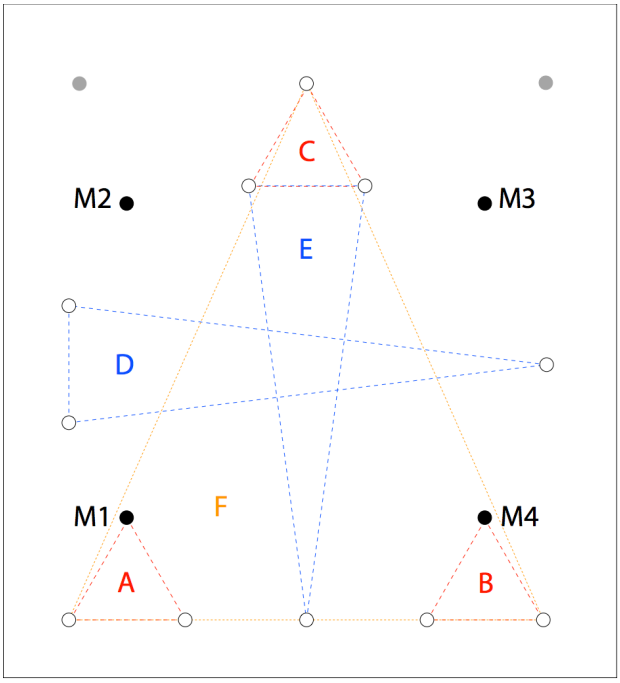


Figure 4.3 Thin-gel phantom schematic

The thin-gel contains three small equilateral triangles configurations (red triangles A through C in Figure 4.3), each with 7 mm sides, differing in their placement relative to the US probe. Two of the small equilateral triangles are located 4 mm from the probe, on the left and right side of the scanning plane, and are referred to as the left-proximal (A) and right-proximal (B) configurations. The third small equilateral triangle configuration, referred to as the distal configuration (C), is located 38 mm from the transducer, centered along the transducer.

The second category of configurations is the isosceles triangle (blue triangles D and E in Figure 4.3), measuring 7 mm on the short side, and 32 mm on the long sides. The thin-gel contains of two of these isosceles triangles with different orientations. Triangle (D) is oriented with its long axis parallel to the front of, and 17 mm away from, the US probe. Triangle (E) is

oriented with its long axis perpendicular to the US probe, with the long axis centered along the US probe, and located 4 mm from the US probe.

The thin-gel contains one large triangle configuration. Its dimensions are 30 mm x 43 mm x 43 mm, situated 4 mm away from the US probe, occupying the majority of the scanning plane (orange triangle F in Figure 4.3). The thin-gel contains four error measurement pins (solid black circles M1 through M4 in Figure 4.3), spaced approximately 30 mm apart in a square configuration. The M1 and M4 pins also serve as calibration pins in the two proximal small equilateral calibration triangles.

4.2 SONIC FLASHLIGHT CALIBRATION SOFTWARE

A custom software program was written to calibrate the SF. The procedure is as follows: Initially, the US image is turned off and the user is presented, instead, with three red crosshairs. The user looks through the SF and moves the crosshairs one at a time, visually lining each one up with a target pin. After marking the three target pins with crosshairs, the user is presented with an uncalibrated US image on the SF monitor. For each of the three marked pins, the user moves this uncalibrated US image on the SF monitor to visually line up the actual view of the pin with its US image. Based on the displacements necessary to line up the three pins, the calibration software calculates the affine transform, as discussed in chapter 2.4.3.2, and applies that affine transform to the US image. Once the affine transform is applied to the US image, the SF is considered calibrated, since the US image seen through the SF should line up with the view of the targets, assuming spatial linearity in the US image.

To measure the calibration error, the user moves the calibrated US image on the SF display to visually line up the actual view of each target pin with its US image. The calibration error at that target is the displacement necessary to make the US image and the actual target line up.

4.3 COPLANAR SCANNING PLANE AND VIRTUAL IMAGE PLANE

The viewpoint independent nature of the SF relies on the virtual US image and the US scanning plane being coplanar. This depends on the design of the rigid mount that holds the display, mirror, and probe at specific angles with the mirror bisecting the angle formed between the display and probe. The thin-gel calibration system can be used to verify that the virtual image and scanning plane are coplanar.

Our test technique relies on the fact that a virtual image is optically indistinguishable from a real image. Therefore, in a properly calibrated SF, objects displayed on the SF display will appear as if they are floating in the thin-gel phantom, and will remain in the same position within the thin-gel regardless of viewpoint. First, a roughly calibrated SF is placed in the calibration system, scanning through the thin-gel phantom. The appearance of the target pins in the US image verifies that the SF is scanning in the correct plane through the thin-gel. Next, the US image on the SF display is replaced with the three crosshairs used during calibration. Starting from one viewpoint through the SF, these crosshairs are moved to three separate target pins in the thin-gel. Then the thin-gel is viewed through the SF from a second viewpoint. To verify coplanar virtual image and scanning planes, the cross hairs must remain at the same location relative to the three target pins. This can be repeated from any number of viewpoints.

However, if the crosshairs do not remain at the three target pins from the second viewpoint, the virtual image and scanning plane are not coplanar.

Figure 4.4 shows the crosshairs lining up with the three target pins from the first viewpoint, located to the left of the center of the SF. Figure 4.5 is taken from the second viewpoint, located to the right of the center of the SF, and demonstrates that the crosshairs do not move relative to the pins. A third viewpoint, down the center of the SF and slightly higher than the first two viewpoints, also shows that the crosshairs do not move relative to the pins (Figure 4.6). These figures demonstrate that the SF is calibrated insofar as having a coplanar scanning and virtual image plane. In a further test, once the system is fully calibrated, including in-plane calibration described in Chapter 4.4, displaying the US on the SF display from multiple viewpoints shows the US image of the pins through the SF remains stable with the actual pins. These are shown in Figure 4.7, Figure 4.8.

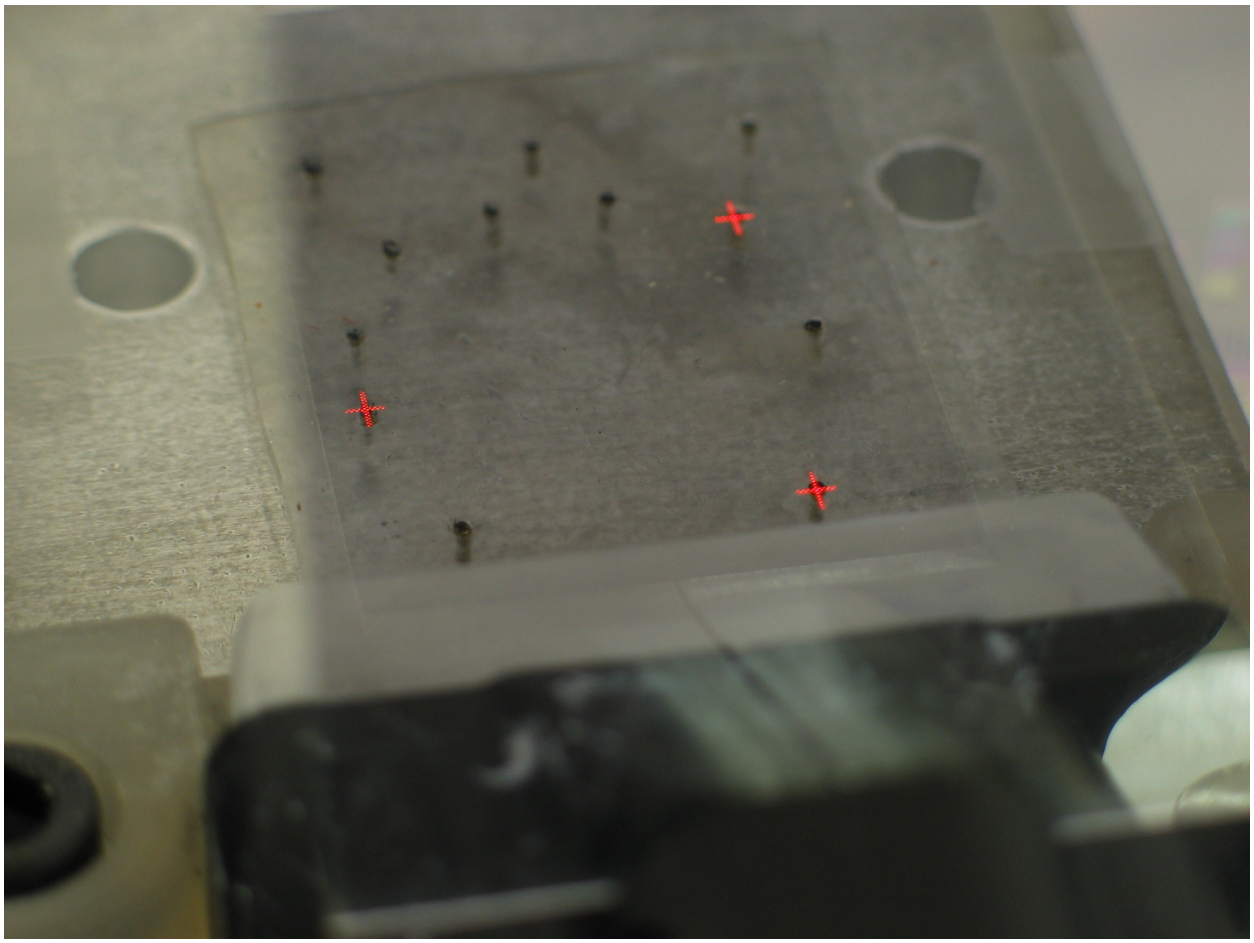


Figure 4.4 Viewpoint independence view 1

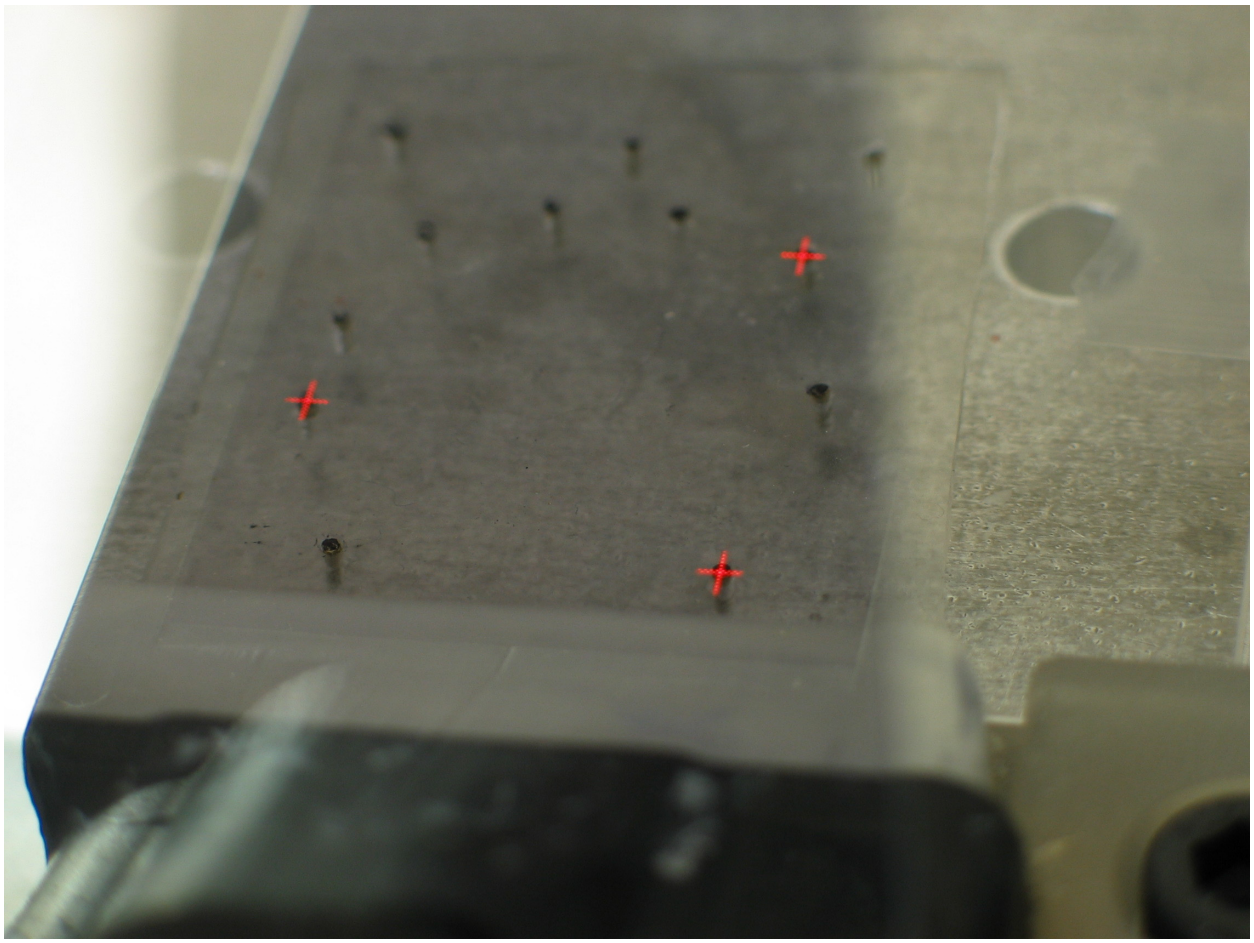


Figure 4.5 Viewpoint independence view 2

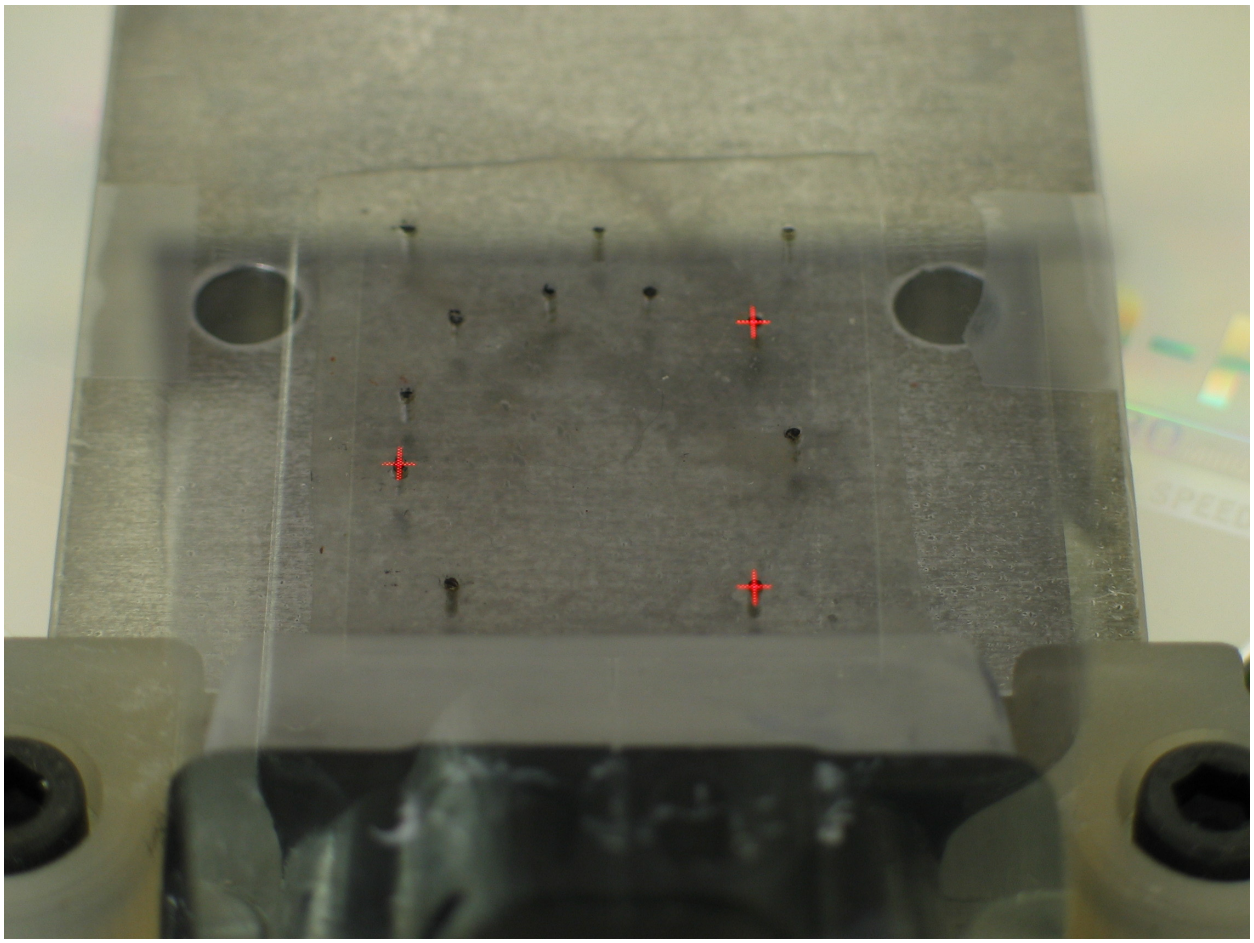


Figure 4.6 Viewpoint independence view 3

4.4 VERIFYING IN-PLANE CALIBRATION

Using the thin-gel calibration device, we tested the following hypothesis regarding calibration: Errors will increase with the distance from the calibration pins, and configurations with widely spaced pins will result in less error across the entire thin-gel.

Calibration was performed 5 times for each of the six configurations in Figure 4.3. The misalignment at each of the 4 measurement pins was determined 5 times per calibration. The complete results are compiled in Appendix C, with the mean distance errors summarized in Table 4.1, and the individual x and y-axis error components summarized in Table 4.2 and Table 4.3. When reporting errors, the x-axis was defined as parallel to the front of the US probe, and the y-axis as perpendicular to the front of the US probe.

Table 4.1 Mean Calibration Error Distances in mm

		Calibration Pin			
Calibration Configuration	*Mean Error*	*M1*	*M2*	*M3*	*M4*
(A) Left proximal small equilateral	0.56	0.00	0.90	0.87	0.49
(B) Right proximal small equilateral	0.65	0.82	0.72	1.08	0.00
(C) Distal small equilateral	0.68	1.14	0.30	0.30	1.00
(D) Parallel isosceles	0.42	0.43	0.60	0.26	0.39
(E) Perpendicular isosceles	0.37	0.52	0.30	0.30	0.37
(F) Large	0.12	0.02	0.22	0.20	0.05

Table 4.2 Mean Calibration Error Components (± std-dev) at M1 and M2 in mm

	M1		*M2*	
Calibration Configuration	*x*	*y*	*x*	*y*
(A) Left proximal small equilateral	0.00 ± 0.00	0.00 ± 0.00	-0.14 ± 0.55	0.72 ± 0.36
(B) Right proximal small equilateral	-0.24 ± 0.73	0.30 ± 0.27	0.07 ± 0.67	0.46 ± 0.15
(C) Distal small equilateral	1.00 ± 0.43	-0.06 ± 0.57	0.12 ± 0.14	0.00 ± 0.30
(D) Parallel isosceles	-0.10 ± 0.11	-0.39 ± 0.20	-0.23 ± 0.16	0.53 ± 0.21
(E) Perpendicular isosceles	0.37 ± 0.20	-0.30 ± 0.19	0.24 ± 0.20	-0.03 ± 0.15
(F) Large	0.01 ± 0.04	0.00 ± 0.03	-0.12 ± 0.10	0.17 ± 0.06

Table 4.3 Mean Calibration Error Components (± std-dev) at M3 and M4 in mm

	M3		*M4*	
Calibration Configuration	*x*	*y*	*x*	*y*
(A) Left proximal small equilateral	0.13 ± 0.54	0.59 ± 0.50	0.04 ± 0.17	0.44 ± 0.25
(B) Right proximal small equilateral	0.81 ± 0.81	-0.26 ± 0.25	0.00 ± 0.00	0.00 ± 0.00
(C) Distal small equilateral	0.21 ± 0.18	-0.12 ± 0.16	0.84 ± 0.49	0.29 ± 0.38
(D) Parallel isosceles	0.03 ± 0.12	0.14 ± 0.24	-0.08 ± 0.12	-0.34 ± 0.21
(E) Perpendicular isosceles	0.22 ± 0.13	0.03 ± 0.18	0.15 ± 0.14	0.30 ± 0.18
(F) Large	0.14 ± 0.13	-0.12 ± 0.06	0.03 ± 0.05	-0.01 ± 0.06

With the left proximal small equilateral configuration, triangle (A), the mean error across the four measurement pins was 0.56 mm. At the individual measurement pins, we see that at M1, the mean error was 0.00 mm, 0.90 mm at M2, 0.87 mm at M3, and 0.49 mm at M4. The 0.00 mm error at M1 was expected since M1 served the dual function of a calibration pin and measurement pin. With the right-proximal small equilateral configuration, triangle (B), the mean error across the four measurement pins was 0.65 mm. The mean error at M1 was 0.82 mm, 0.72 mm at M2, 1.08 mm at M3, and 0.00 mm at M4. The 0.00 mm error at M4 was expected since M4 served as both a calibration and measurement pin. With the distal small equilateral configuration, triangle (C), the mean error across the four measurement pins was 0.68 mm. The mean error at M1 was 1.14 mm, 0.00 mm at M2, 0.30 mm at M3, 1.00 mm at M4.

With the parallel isosceles configuration, triangle (D), the mean error across the four measurement pins was 0.42 mm. The mean error at M1 was 0.43 mm, 0.60 ± 0.20 mm at M2, 0.26 mm at M3, and 0.40 ± 0.16 mm at M4. With the perpendicular isosceles configuration, triangle (E), the mean error across all four measurement pins was 0.37 mm. The mean error at M1 was 0.52 mm, 0.30 mm at M2, 0.30 mm at M3, and 0.37 mm at M4. The large configuration, triangle (F), yielded a mean error across the four measurement pins of 0.12 mm. The mean error at M1 was 0.02 mm, 0.22 mm at M2, 0.20 mm at M3, and 0.05 mm at M4.

All three small equilateral triangle configurations have similar overall errors (0.56 vs. 0.65 vs. 0.68 mm). Since the left-proximal and right-proximal pin configurations are left-right mirror images relative to the center of the probe, we expect the errors at the measurement pins to show the same mirrored relationship. In other words, the left-proximal configuration should show similar errors at M1, M2, M3, and M4 that the right-proximal configuration shows at M4, M3, M2, and M1. M2 in the left-proximal configuration corresponds to M3 in the right-proximal

configuration, and both have similar errors at 0.9 and 1.08 mm, respectively. M3 in the left-proximal has an error of 0.87 mm, which compares very closely to the 0.72 mm error shown at M2. M4 in the left-proximal configuration has 0.49 mm of error, which compares closely to 0.65 mm of error at M1 in the right-proximal configuration.

The overall errors are reduced in the isosceles triangle configurations (0.42 and 0.37 mm) compared to the small equilateral configurations. According to our hypothesis, we expected the y-axis error to be greater in the x-axis error in the parallel configurations, and vice versa in the perpendicular configuration. In the parallel configuration, the y-axis error magnitude mean (± standard deviation) was 0.37 ± 0.23 mm. As expected, this is greater than the x-axis error magnitude mean of 0.14 ± 0.12 mm. However, in the perpendicular configuration, contrary to what we expected, the y-axis error magnitude mean was 0.22 ± 0.16 mm is very similar to the x-axis error magnitude mean of 0.25 ± 0.17 mm.

The large triangle configuration had the least amount of error, averaging 0.12 mm. This is a significant improvement over the three small equilateral and two isosceles triangle configurations. Therefore, the results clearly indicate that the large triangle configuration leads to a more accurately calibrated SF. Figure 4.7 and Figure 4.8 demonstrate a fully calibrated SF from two viewpoints.

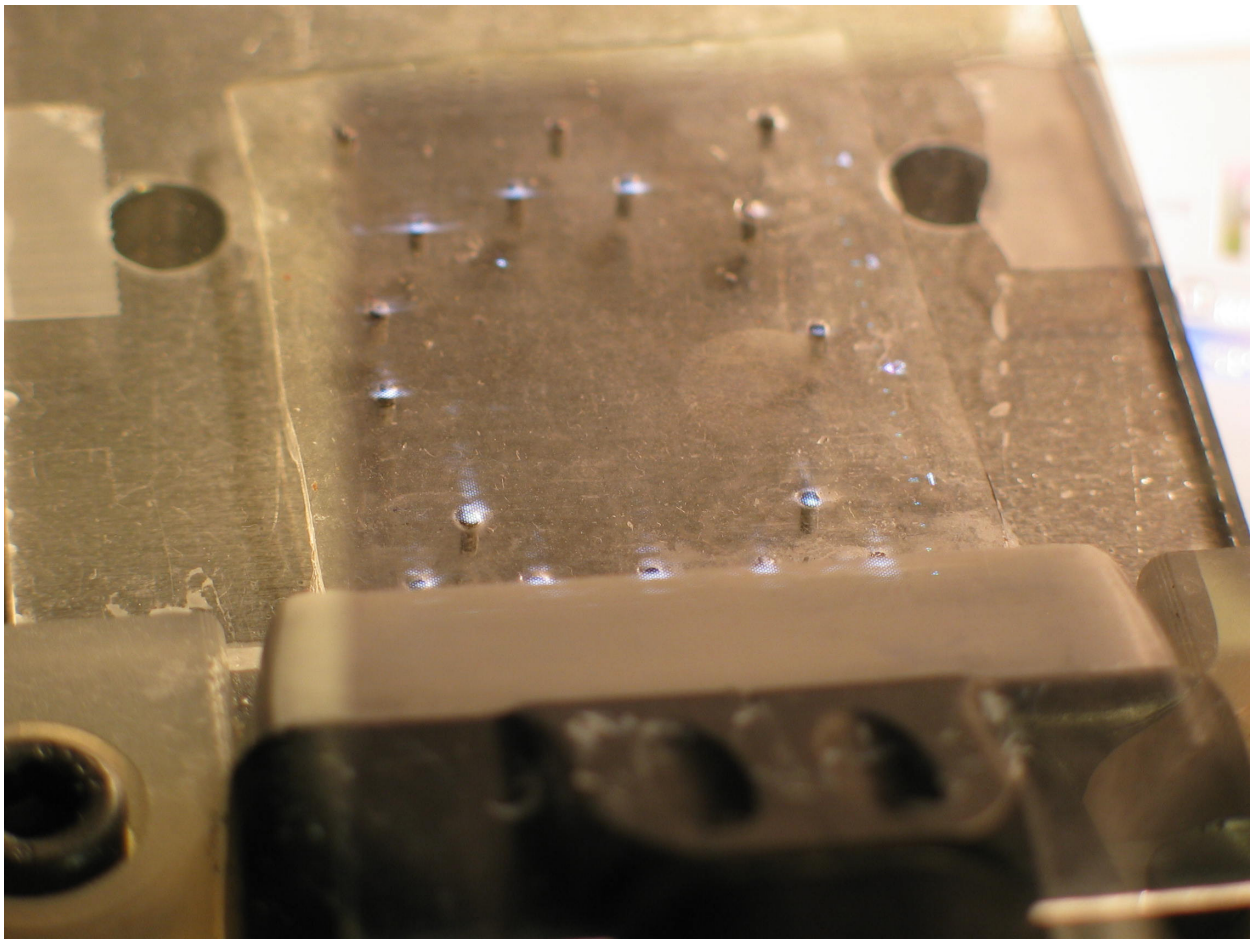


Figure 4.7 Fully calibrated Sonic Flashlight, view 1

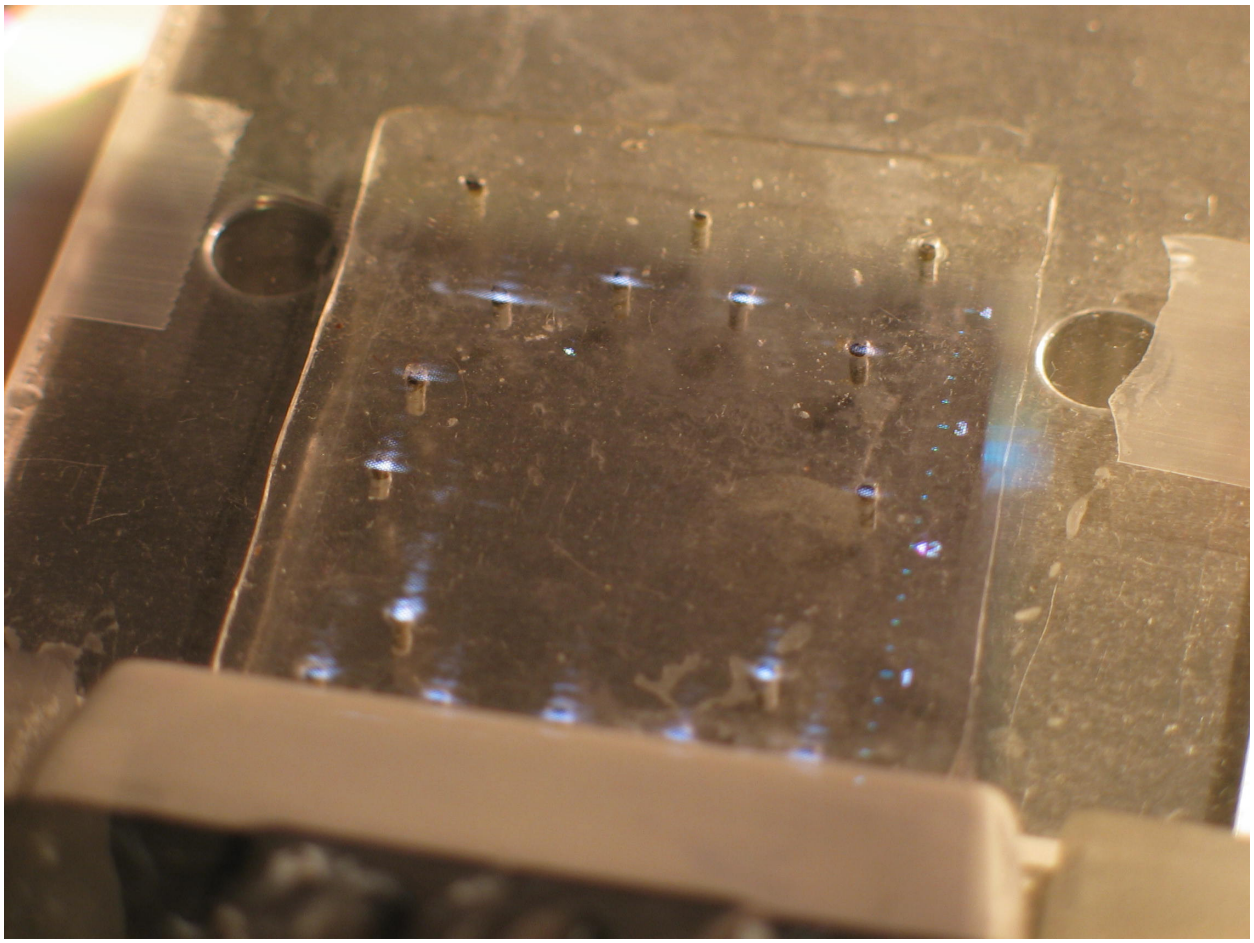


Figure 4.8 Fully calibrated Sonic Flashlight, view 2

5.0 VALIDATION

Experiments to analyze human performance are essential to an effective assessment of the SF before clinical testing. We believe the device taps into an existing reservoir of human skills that are not available with conventional US, and in particular, those skills based on direct hand-eye coordination. Some aspects of the SF are quite natural (aiming the needle directly at something, albeit, an image instead of a real target) and others are new to human perception (merging apparently translucent skin with an US image floating below it). Utilizing vascular US phantoms, we conducted a series of experiments comparing the use of the SF to CUS in two important population groups: US novices, and proficient CUS users. Finally, we demonstrated vascular access in a cadaver, operating the SF ourselves.

5.1 VASCULAR PHANTOMS

The US phantoms selected for the human performance studies in this dissertation contain fluid-filled tubes simulating vasculature. Our first vascular access phantoms were constructed by embedding thin-walled latex tubing within agar (30). Benefits of this system include low cost, and the ability to rapidly implement different designs and vascular configurations. Several drawbacks were noticed in early experimentation with the agar gels, however. First, an agar gel is easily crushed with moderate pressure from an US probe, essentially ruining the phantom. Second, the agar gel does not "heal," meaning that once a needle has entered the phantom, the

needle track remains visible on US for the life of the phantom. Finally, the combination of the agar drying out and being a good growth medium for bacteria makes long-term storage of the phantoms particularly difficult. Due to these drawbacks, agar phantoms were not used for the studies in this dissertation. Instead, three commercially available US phantoms were selected for the human performance studies. These phantoms do not suffer from the drawbacks of the agar phantoms, such as drying out, needle tracks, or bacteria growth. However, disadvantages of using commercial phantoms include expense and slow turn-around time for modifying vessel configuration.

5.2 NOVICE PILOT STUDY

The aim of this study was to compare US novices performing their first simulated vascular access using the SF vs. CUS on a phantom. A 9MHz Site-Rite 3 (Bard, Murray Hill, NJ) without a needle guide was the CUS scanner used in this study. Two commercially available vascular access phantoms were used in this study. Simulating large vasculature (e.g. IJV) the ATS phantom (ATS Laboratories Inc., Bridgeport, CT) contains a straight 9 mm diameter vessel located 10 mm below the surface, with a vessel wall thickness of ~1.5 mm. The standard Blue Phantom simulates smaller vasculature, containing a straight 4 mm diameter vessel located 6mm below the surface. The vascular phantoms are shown in Figure 5.1. The study population consisted of graduate students and medical residents associated with our institution with little or no US experience. Subjects were given a short tutorial on how to use both devices and how to perform the procedures before beginning. The CUS and SF tutorials consisted of a description of

the basic functioning of the device, what the US image represented, and how to locate, aim, and guide a needle in a cross-sectional US scan of a vessel, followed by a hands-off demonstration.

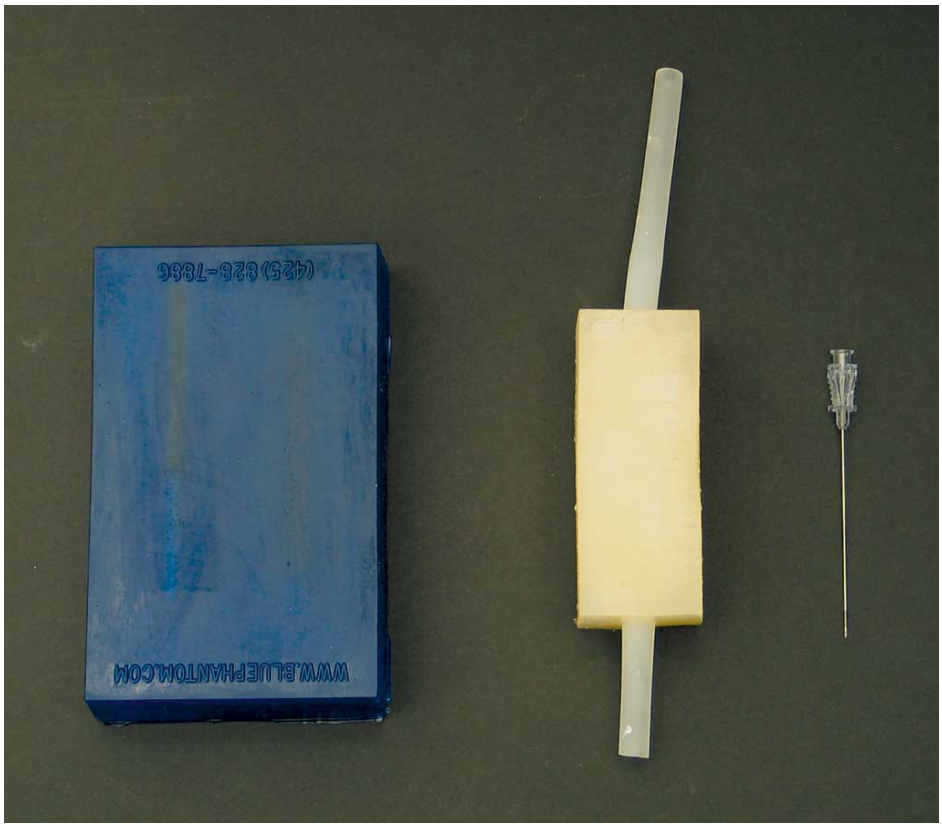


Figure 5.1 Vascular phantoms
Blue Phantom (left), ATS phantom (middle), 21-gauge micropuncture needle (right).

Trials by 37 subjects were recorded with the ATS phantom. In addition, a total of 16 separate subjects were recorded with the standard Blue Phantom. All subjects used the SF first and the CUS second, slightly biasing the study against the SF. Subjects used a 21-gauge 7 cm

needle to pierce the lumen of the particular vascular access phantom. The time between the probe touching the phantom to the needle piercing the lumen, including multiple attempts, was recorded. An observer experienced in US determined successful entry into the lumen by watching the US display. The times were analyzed using a two-tailed paired t-Test.

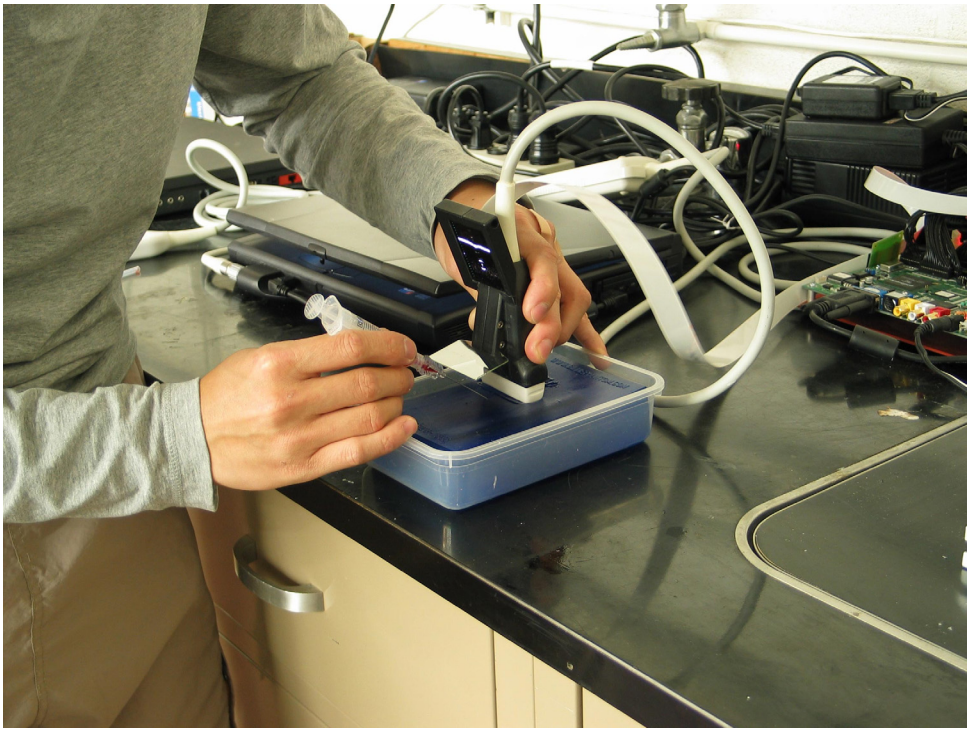


Figure 5.2 Overview of SF guided vascular access in a phantom

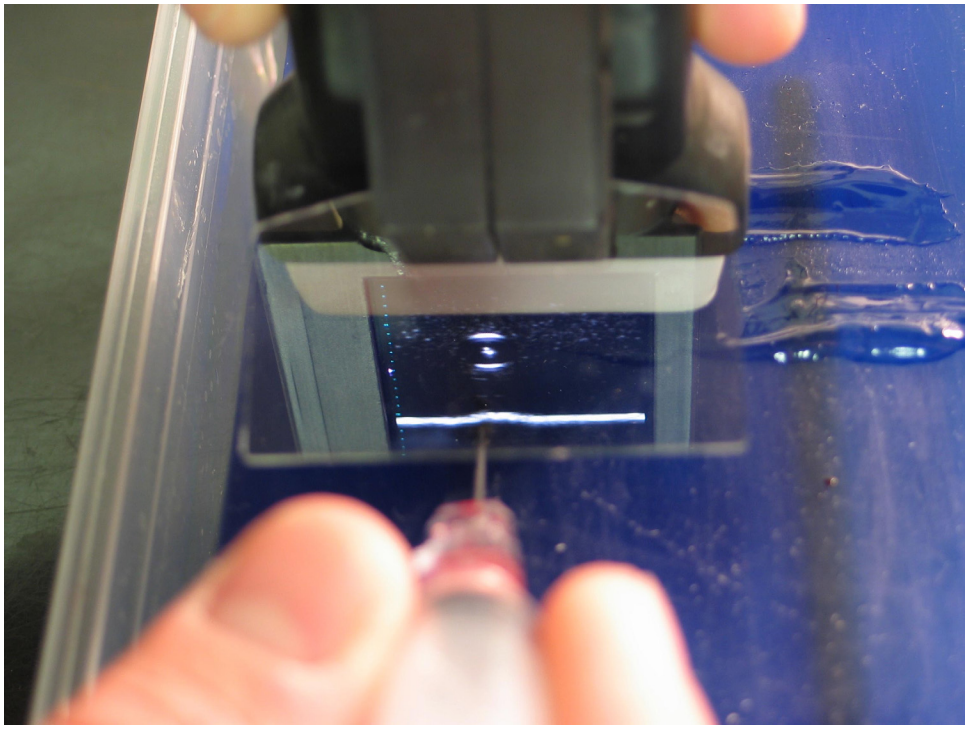


Figure 5.3 View through the SF during vascular access in a phantom.

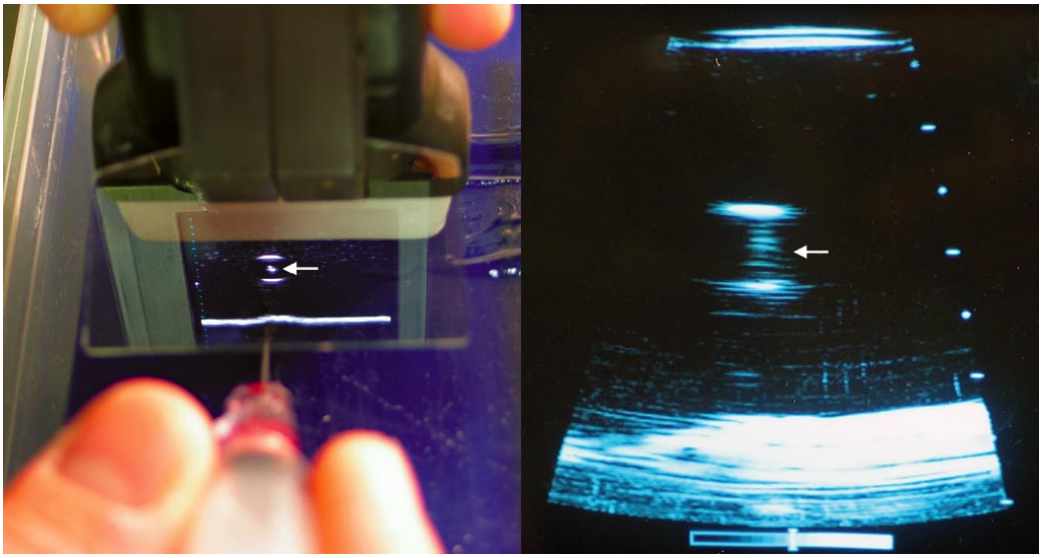


Figure 5.4 SF versus CUS image comparison

SF (left) and Site-Rite 3 (right). Arrow indicates the needle within the vessel.

Figure 5.2 and Figure 5.3 illustrate a simulated vascular access procedure on the Blue Phantom using the SF. Figure 5.4 shows a comparison of the US images between the SF and the Site-Rite 3. For both phantoms, significantly faster times were recorded using the SF for guidance as compared to CUS. On the ATS phantom, 37 subjects recorded an average time of 5.6 secs (SEM = 0.4) using the SF, versus 9.8 secs (SEM = 1.2) using CUS, with a difference of 4.2 secs (SEM = 1.1, $p < 0.0006$), shown in Figure 5.5 and Figure 5.6. 16 subjects recorded an average time on the standard Blue Phantom of 8.8 secs (SEM = 1.1 using the SF, versus 21.1 secs (SEM = 3.2) with a difference of 12.3 secs (SEM = 3.2, $p < 0.0016$), also shown in Figure 5.5 and Figure 5.6. The greater difficulty in using the Blue Phantom was due to the smaller diameter of the embedded vessel.

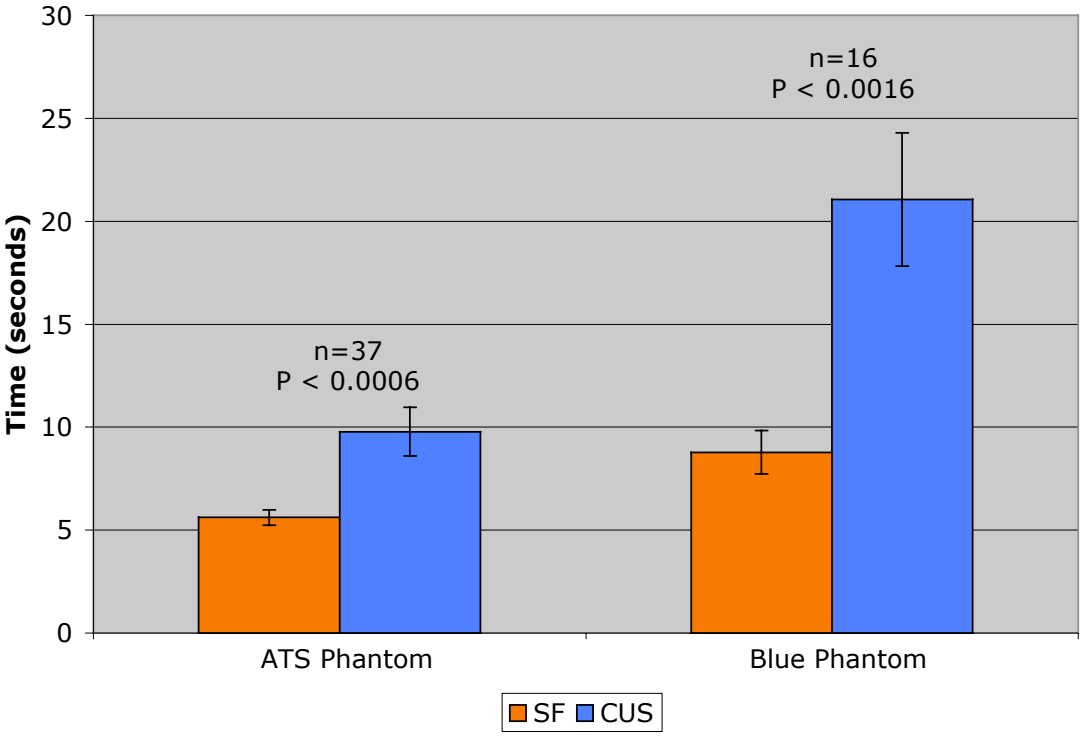


Figure 5.5 SF versus CUS guided vascular access times for novices

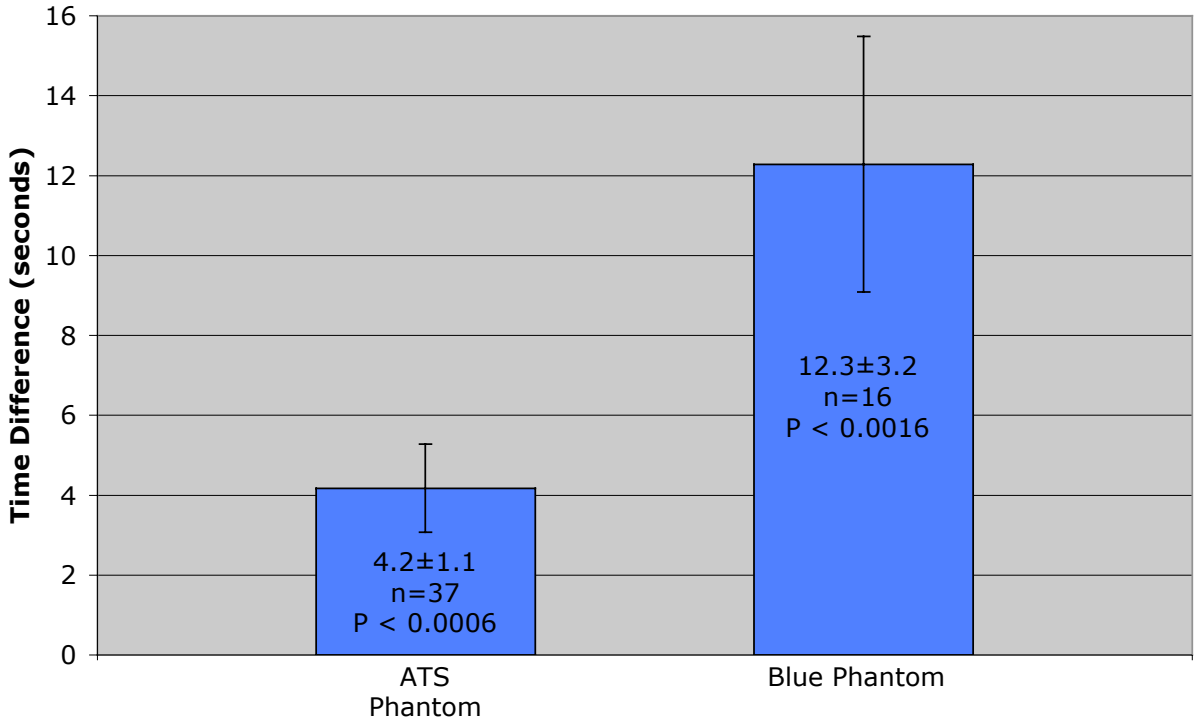


Figure 5.6 SF versus CUS vascular access time difference for novices (CUS - SF times)

While this pilot study suggests that novice US users performing vascular access are faster using the SF than CUS, there are several weaknesses in the design of this study. First, the study is biased since subjects use the SF first, and CUS second. Although the bias appears to be against the SF since subjects may learn the task and the phantom configuration before using CUS, randomly assigning subjects to begin with the SF or CUS would eliminate the dependence on that assumption. Another weakness to this study is the subjective determination of successful needle entry into the vessel. Since an independent US observer determined successful needle entry into the vessel lumen by watching the US screen, it would be possible to enter the vessel

lumen undetected with the needle outside the scanning plane. In such a case, the US observer would not have stopped the timer until the probe was moved to the correct scanning location. While this would not necessarily introduce a bias, it would add some inaccuracy to the study design. These design issues were addressed in two subsequent studies.

5.3 NOVICE LEARNING STUDY

The results of the pilot study in the previous section suggest easier vascular access by a novice using SF than CUS. However, that study did fully explore whether the SF is an easier device to learn. We believe there are three important questions about the SF that need to be answered: is it easier to use initially, is it easier to learn, and is it easier to use eventually after extensive training? To answer these questions, and address weaknesses in the previous study, novice US users were recruited to participate in a randomized study spanning two days.

The ATL and standard Blue Phantom phantoms in the previous study only contained straight vessels. Actual vasculature bifurcates, is not perfectly straight, and is often adjacent to other vessels and nerves. We designed a custom Blue Phantom containing multiple vessels, some with bifurcations, to increase the difficulty and add more realism to the procedure. The custom Blue Phantom contains 3 vessels, labeled vessel 1, vessel 2, and vessel 3, in Figure 5.7. Vessel 1 is a bifurcating 5 mm vessel, 9mm from the surface of the phantom. Directly beneath vessel 1, vessel 2 is 3 mm in diameter 20 mm from the surface of the phantom. Vessel 2 also bifurcates, but in the opposite direction from vessel 1. When viewed from above, the two vessels appear similar to a letter ‘Y’ stacked on top of an upside down ‘Y.’ Vessel 3 is a straight 4 mm vessel, 15 mm from the surface of the phantom. Intravenous bags containing saline dyed with

food coloring were hooked up to each vessel, with green in vessel 2 and vessel 3, and red in vessel 1. The IV bags were hung 6 inches above the phantom to provide positive pressure that caused a "flash" of colored fluid in the needle hub when successfully placed in the vessel. This eliminated the need for subjective judgment of success by an external observer looking at the US image.

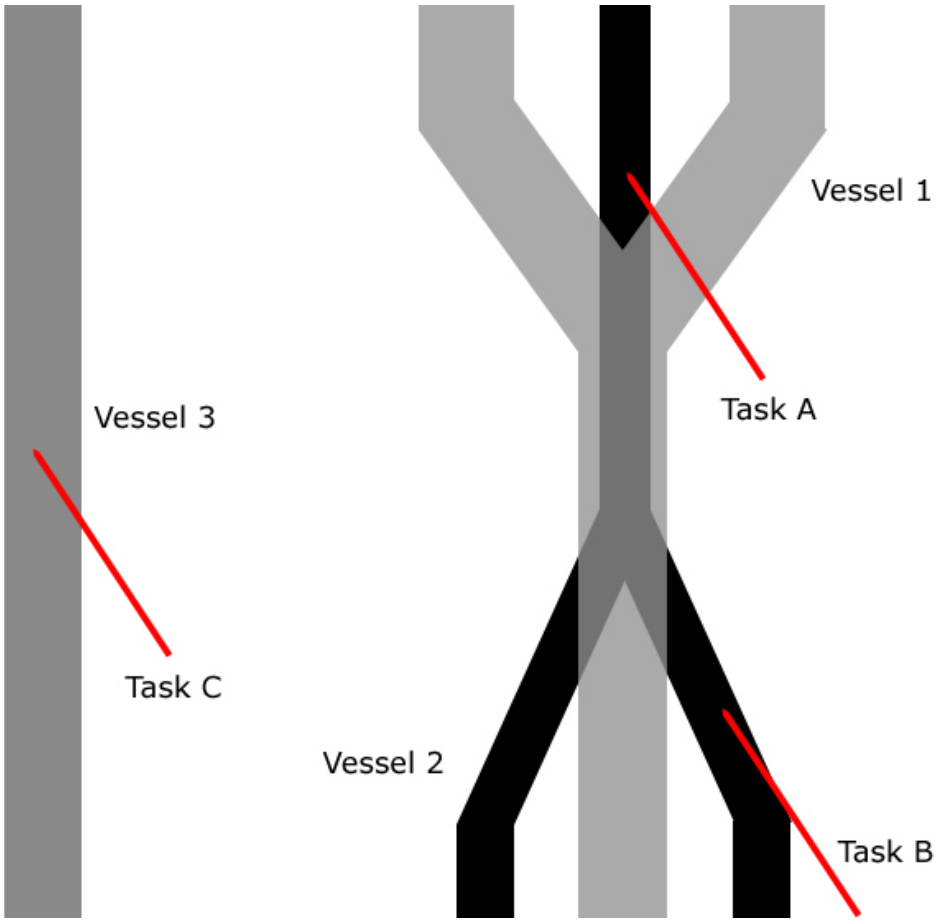


Figure 5.7 Custom Vascular Access Phantom

Red lines indicate approximate needle target locations for each task.

The subject population consisted of 16 medical students with no US experience, randomized to either the SF or CUS group, with 8 subjects per group. Subjects were given a short tutorial on how to use the SF or CUS, and how to perform the procedures before beginning. Again, the CUS and SF tutorials consisted of a description on how to use the device, what the US image represented, and how to locate, aim, and guide a needle in a cross-sectional US scan of a vessel, followed by a hands-off demonstration. The CUS machine used in this study was a Terason 2000 with a 10MHz probe, the same US machine used in the SF. This provided a fairer comparison between the SF and CUS, especially since the Terason scanner has superior image quality to the Site-Rite scanner used in the previous experiment.

Subjects were asked to perform three tasks, labeled A, B, and C in Figure 5.7. Task A: guide a needle between the bifurcation of vessel 1, without hitting vessel 1, and hit vessel 2. Task B: guide a needle into the right bifurcation branch of vessel 2. Task C: guide a needle into vessel 3. Subjects used a 21-gauge 7 cm micropuncture needle (Boston Scientific, Natick, MA) for all tasks. Task B was designed to be the most difficult, and task C the easiest. Total time from the probe first touching the phantom to a needle flash (i.e. colored fluid filling the needle hub) was recorded, including multiple attempts. If at any time the needle entered an incorrect vessel, the subject was asked to remove the needle completely from the phantom and reattempt guidance into the correct vessel. A trial was defined as the completion of one task, timed from the probe touching the phantom to needle flash. A block was defined as containing one trial each at task A, B, and C, in randomized order. Over the course of two days, separated by approximately one week (7-9 days), subjects were asked to perform ten blocks (or 30 trials) per day, or 60 trials total. By the end of the study, each subject had completed 20 trials on each task. The complete results are compiled in Appendix D and summarized below. Outliers were

replaced by a mean value of the other data points of other subjects at the same trial number, biased by the subjects' mean trial times.

The results from this study can be compared against the novice pilot study described in the previous section. Task C in this learning study involves a straight 4 mm vessel, 15 mm deep. This is similar to the novice pilot study with a straight 4 mm vessel, 6 mm deep. While the tasks are not identical (i.e. different vessel depths), we can compare the results of the novice pilot study with the subjects' first attempt at task C. In this study, 16 subjects recorded an average task C time of 10.0 secs (SEM = 1.4) using the SF, versus 16.7 secs (SEM = 1.2) using CUS, with a mean difference of 6.7 secs ($p < 0.003$, two-tailed t-test), shown in Figure 5.8. These results are very similar to the novice pilot study (Figure 5.5, right), where 16 subjects recorded an average time on the standard Blue Phantom of 8.8 secs (SEM = 1.1) using the SF, versus 21.1 secs (SEM = 3.2) using CUS, with a mean difference of 12.3 secs ($p < 0.0016$).

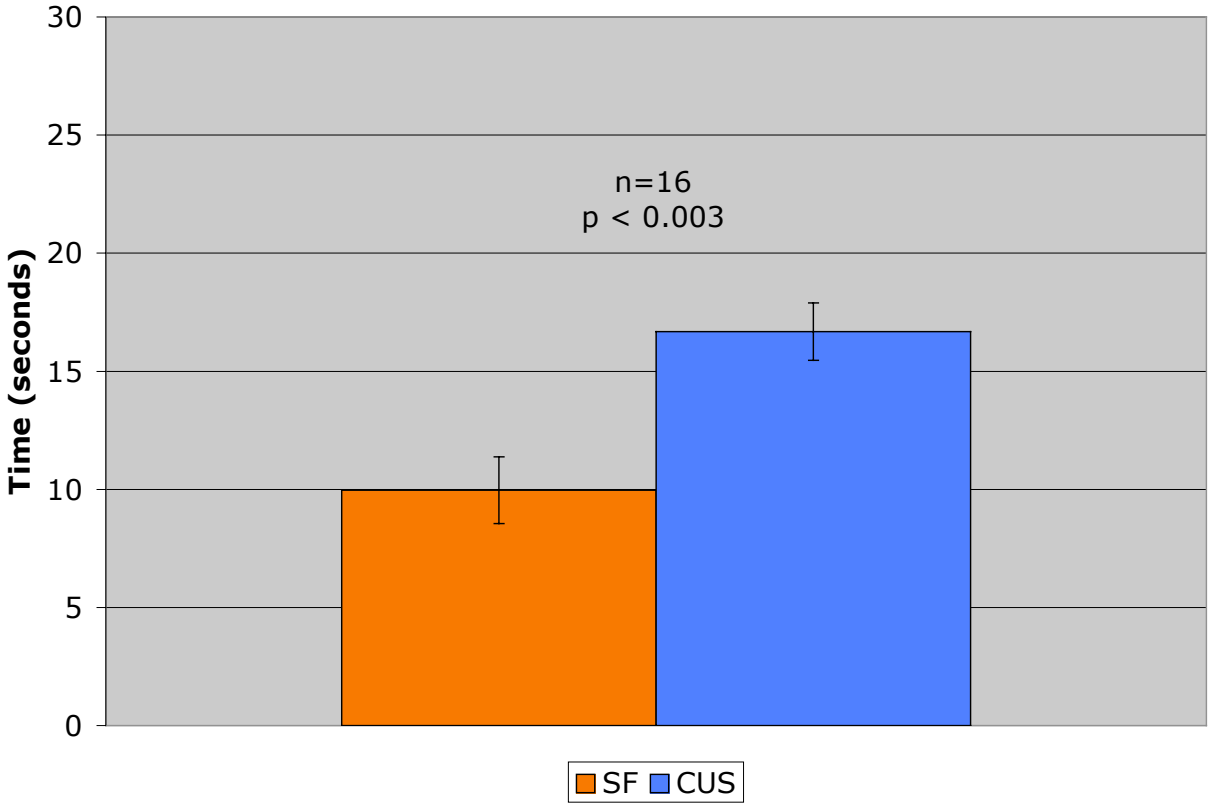


Figure 5.8 First trial of US novices using SF and CUS guidance at Task C

Subjects using the SF performed faster than subjects using the CUS (Figure 5.9, Figure 5.10, Figure 5.11) throughout all 20 trials at all three tasks. We performed an ANOVA using task time as the dependent variable, and task (A, B, C), group (SF vs CUS), and trial (trial number) as the factors. There was a significant group effect on task time for task A ($F(1,14)=5.501$, $p=0.0343$), task B ($F(1,14)=12.007$, $p=0.038$), and task C ($F(1,14)=7.584$, $p=0.0155$), reflecting faster times using the SF than CUS. There was also a significant effect of trial number on task time for task A ($F(19,14)=28.705$, $p<0.0001$), task B ($F(19,14)=11.587$, $p<0.0001$), and task C ($F(19,14)=18.964$, $p<0.0001$), due to the downward trend of the learning curves.

ANOVA trial x group interaction is a direct statistical comparison of learning rates between the SF and CUS. No significant trial x group interaction was seen in the more difficult tasks (task A, $F(19,266)=0.815$, $p=0.6891$; task B, $F(19,266)=0.897$, $p=0.5870$), while task C (the easiest task) did show a statistically significant interaction ($F(19,266)=1.647$, $p=0.0455$). This analysis alone indicates that there may be a difference in learning rate only in the easiest of the three tasks. Since the trial x group interaction analysis is sensitive to a single trial differing between the two groups, the significant trial x group interaction at task C is most likely due to the first trial differing, and not an actual difference in learning rate. We employed another method to measure learning rates that is not as sensitive to noise, fitting a power function. We found that fitting power functions to the data indicates the opposite about the learning rates, with tasks A and B showing the largest differences in learning rates between the two groups, and task C showing very little difference in learning rate (Table 5.1 and Figure 5.12). Based on these two analysis techniques, we can conclude that the mean difference in learning rate, as estimated by the power fit, do not reach significance for tasks A and B. For task C, the interaction is not due to an overall difference in learning rate, but rather to differences in one or more data points (i.e. trial 1) that are consistent across subjects, enough so to reach significance.

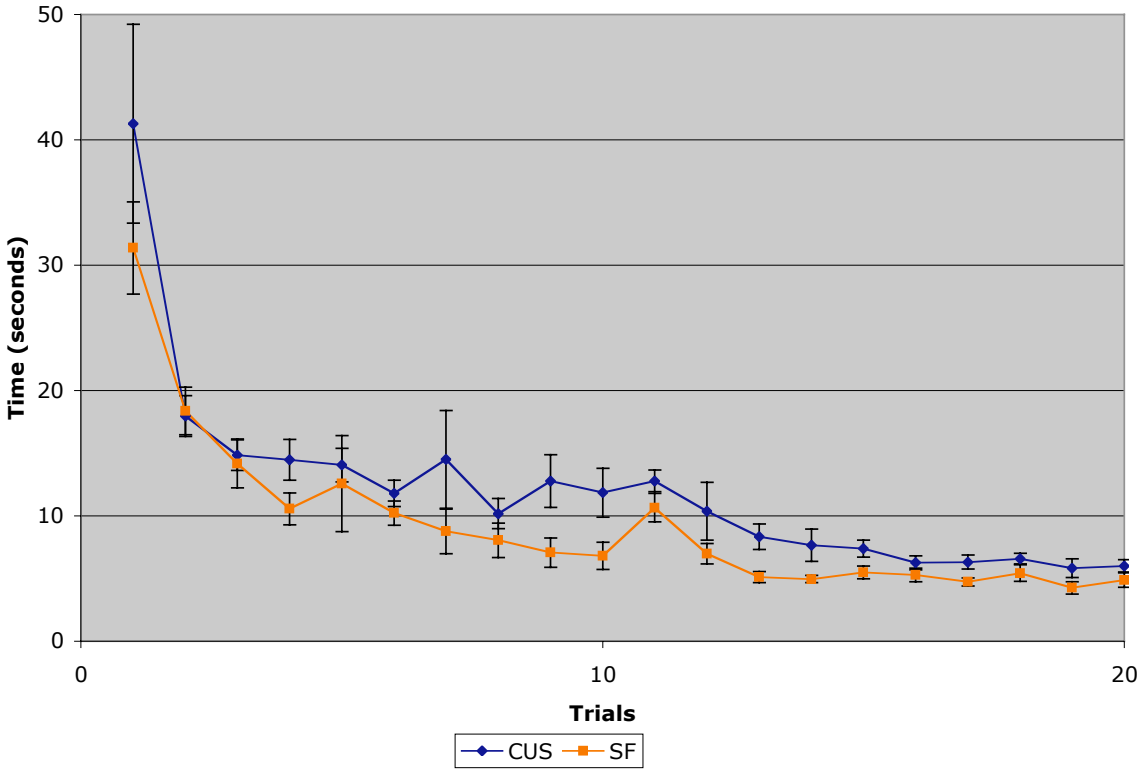


Figure 5.9 Task A learning curves

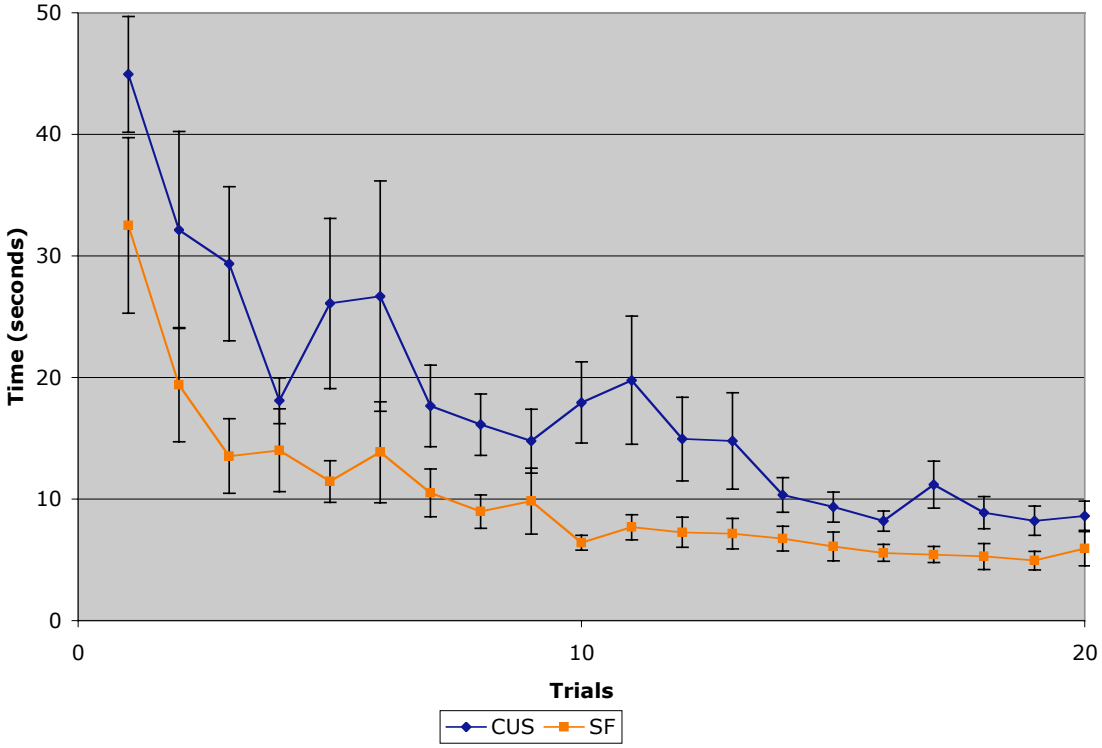


Figure 5.10 Task B learning curves

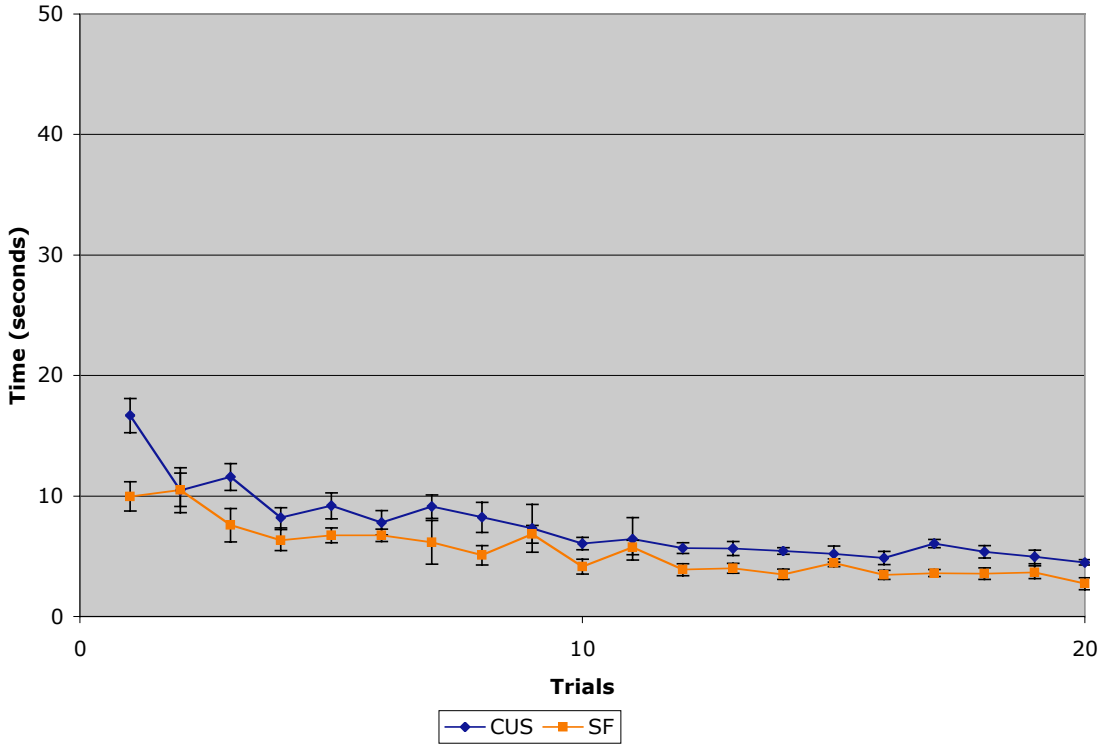


Figure 5.11 Task C learning curves

Table 5.1 Parameters of Power Functions Fit to Learning Curves

Task	*Modality*	*Slope (Learning Rate)*	*Intercept*	r^2 *(Fit)*
A	SF	-0.615	1.469	0.912
	CUS	-0.531	1.521	0.859
B	SF	-0.592	1.492	0.912
	CUS	-0.486	1.675	0.802
C	SF	-0.406	1.084	0.844
	CUS	-0.396	1.211	0.903

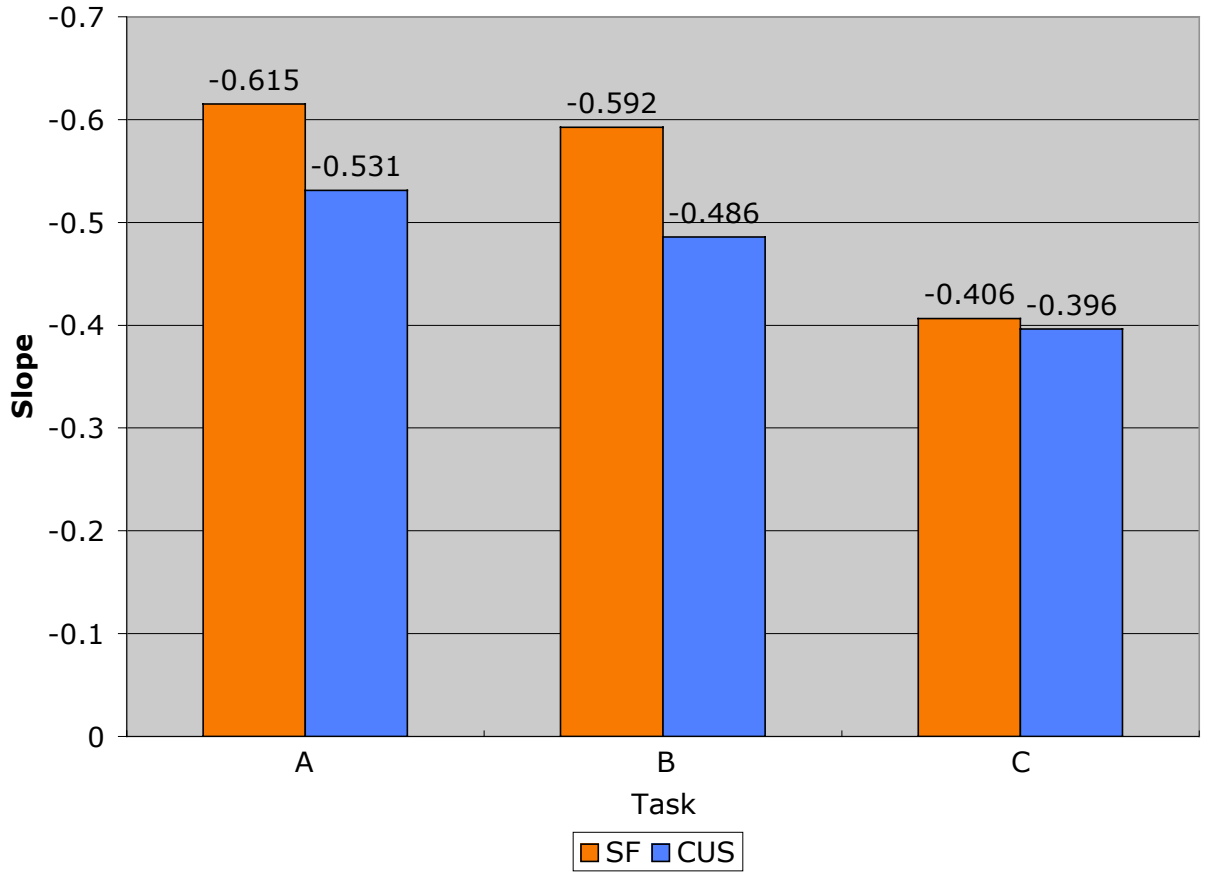


Figure 5.12 Learning rate (slope) of power functions fit to learning curves

A second way to compare learning between the SF and CUS is to select proficiency threshold times: 7, 8, and 9 secs for task A, 9, 10, and 11 secs for task B, and 4, 5, and 6 secs for task C. These threshold times were selected to clearly separate the SF and CUS data. From each of these thresholds, we compute the percentage of subjects finishing faster that the threshold, giving us an indication of how quickly subjects reach proficiency with each modality.

Results of this analysis are shown for task A in Figure 5.13, Figure 5.14, and Figure 5.15. At task A, on trial 13, 100% of the SF subjects, compared with 38% of the CUS subjects, were

able to perform the task in fewer than 7 secs (Figure 5.13). Notice, however, that subsequent trials were not guaranteed to stay at 100%, since sometimes a subject could still take longer than the threshold time to complete the task. With an 8 sec threshold on task A, at trial 13, 100% of the SF subjects, compared with 50% of the CUS subjects, were able to perform the task in under 8 secs (Figure 5.14). With a 9 sec threshold on task A, at trial 13, 100 % of the SF subjects, compared to 63% of the CUS subjects, are able to perform the task in under 9 secs (Figure 5.15). Similar results for tasks B and C are shown in Figure 5.16 through Figure 5.21.

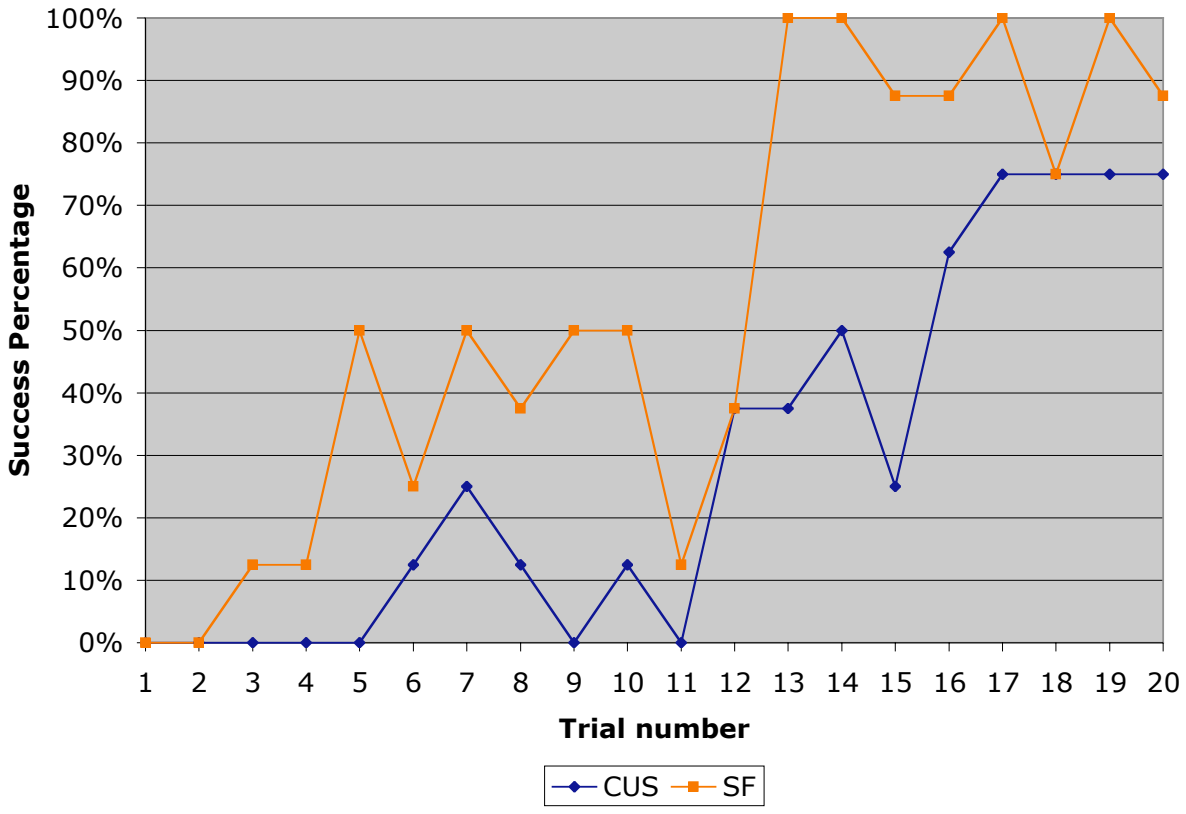


Figure 5.13 Percentages of subjects finishing under 7 secs at Task A

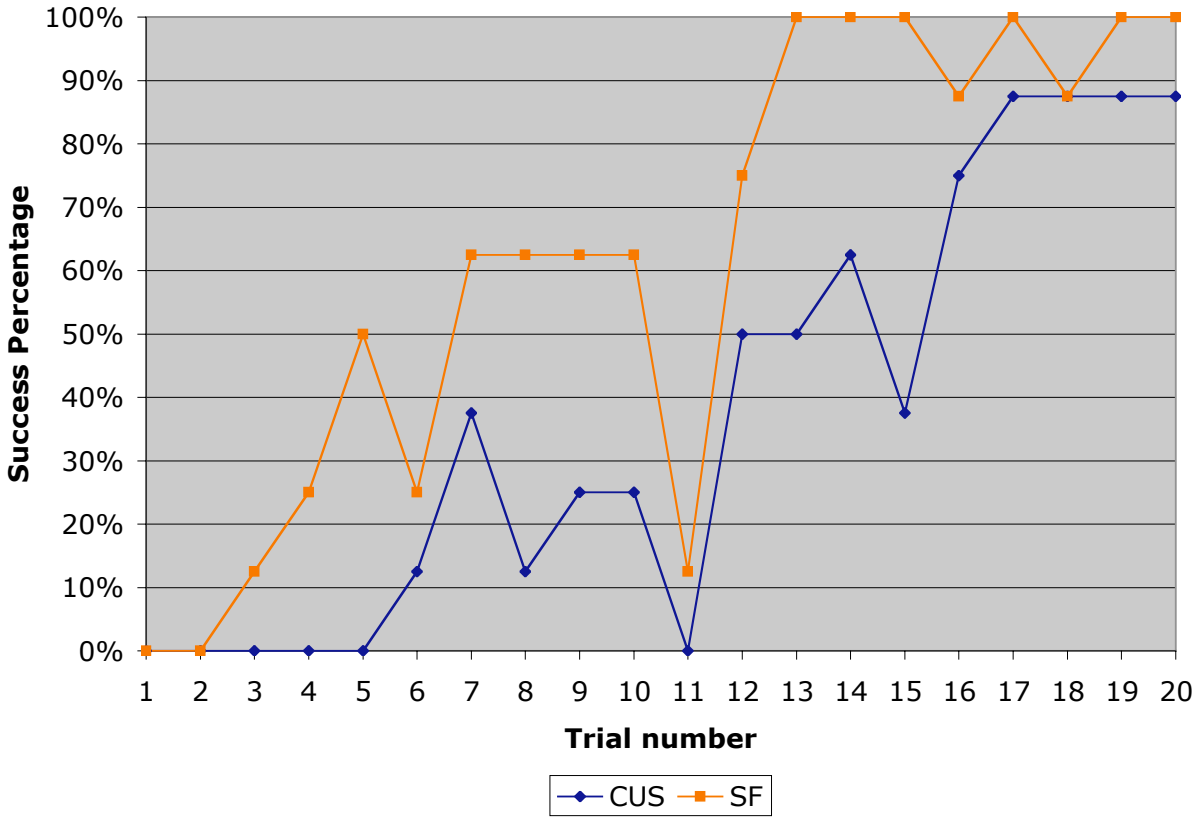


Figure 5.14 Percentages of subjects finishing under 8 secs at Task A

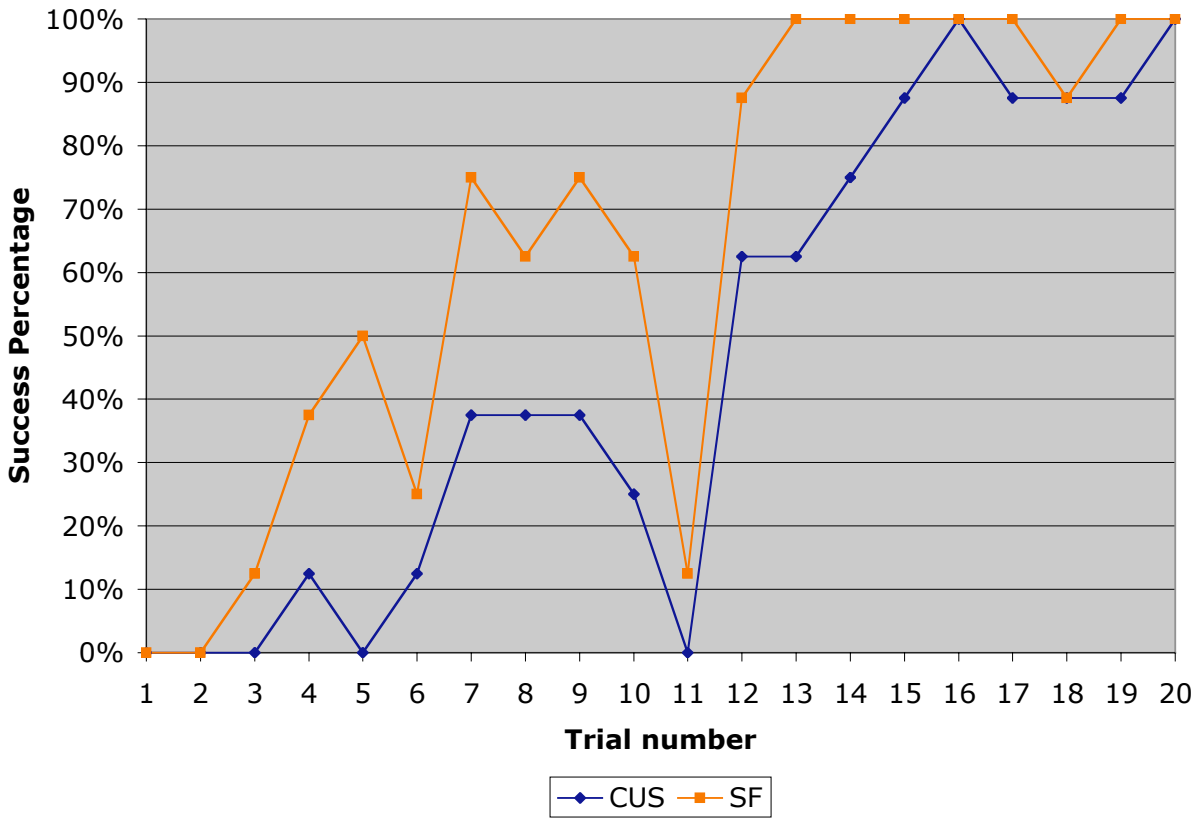


Figure 5.15 Percentages of subjects finishing under 9 secs at Task A

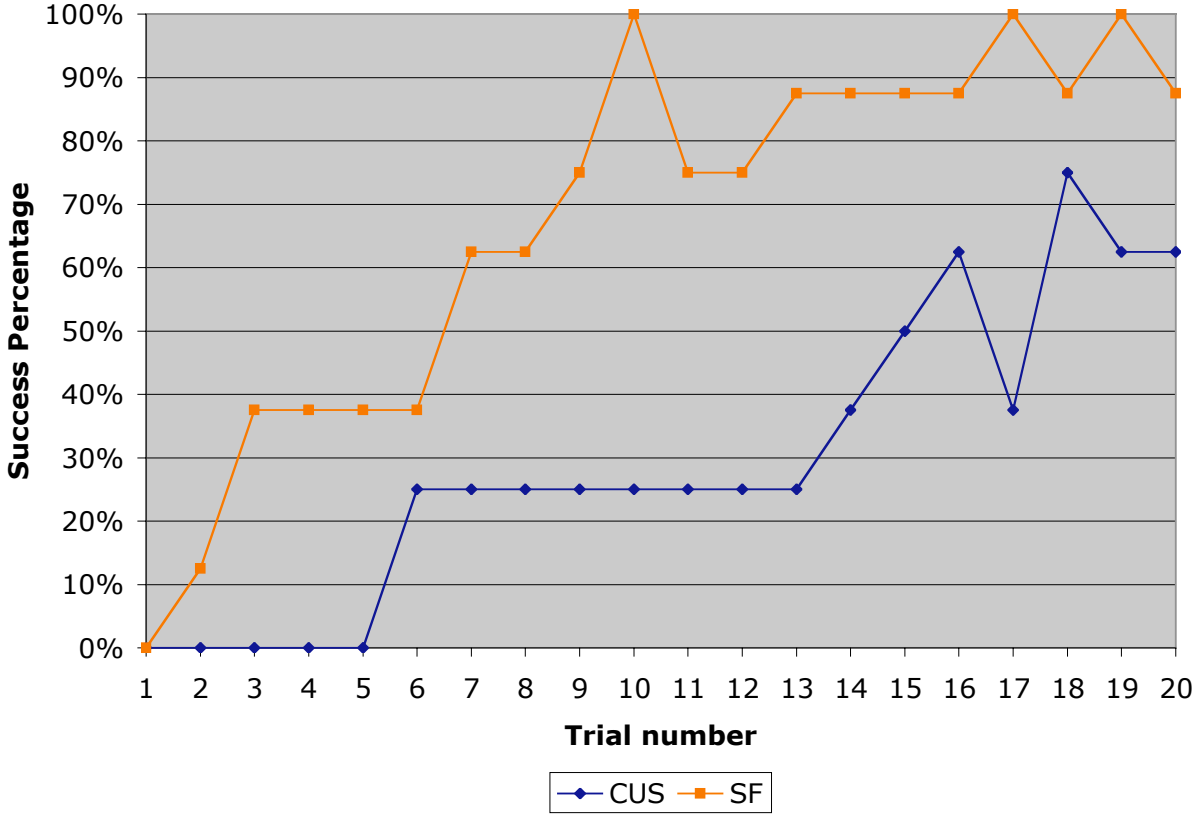


Figure 5.16 Percentages of subjects finishing under 9 secs at Task B

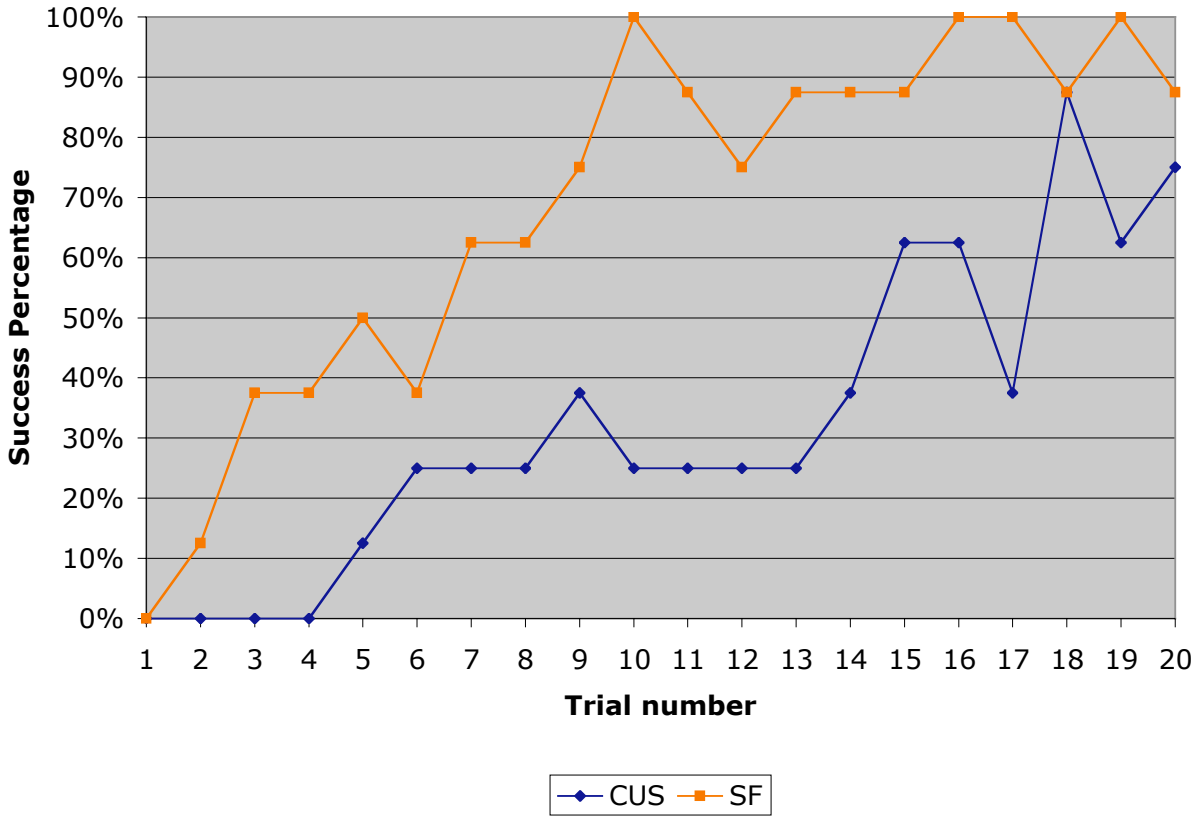


Figure 5.17 Percentages of subjects finishing under 10 secs at Task B

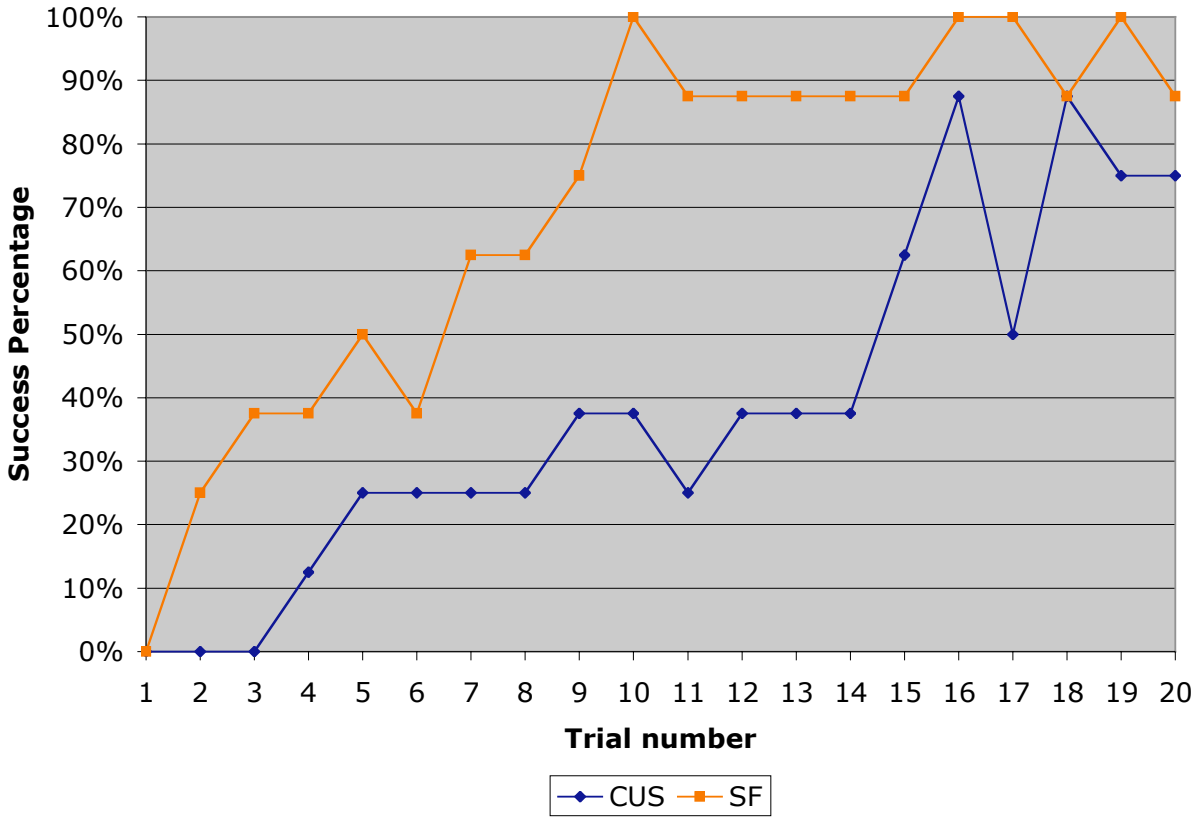


Figure 5.18 Percentages of subjects finishing under 11 secs at Task B

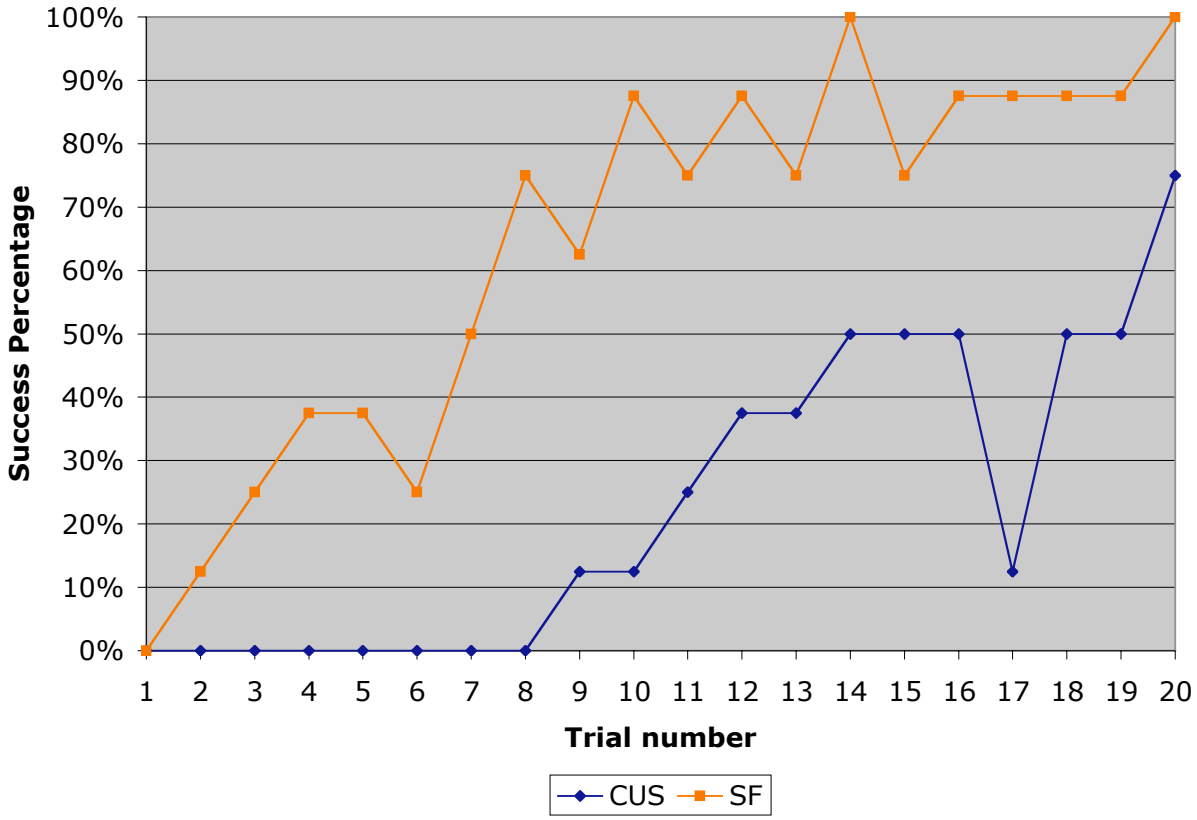


Figure 5.19 Percentages of subjects finishing under 5 secs at Task C

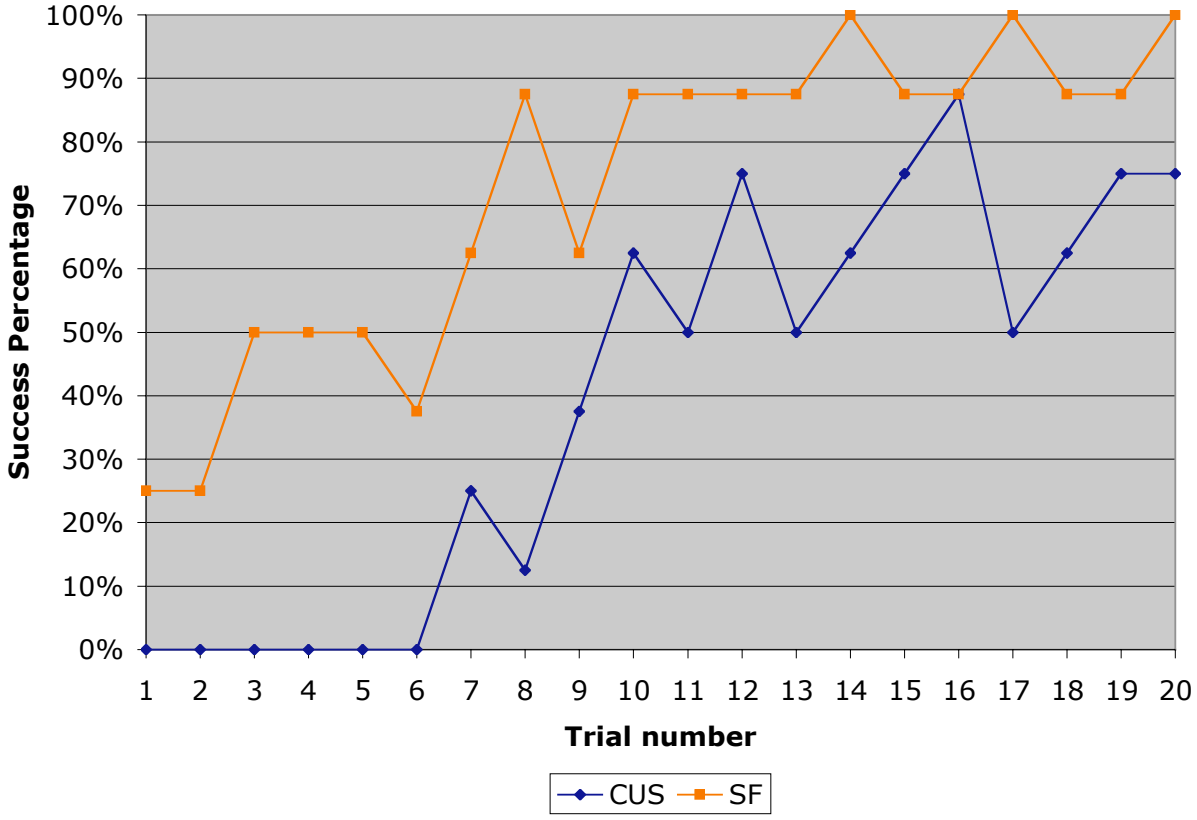


Figure 5.20 Percentages of subjects finishing under 6 secs at Task C

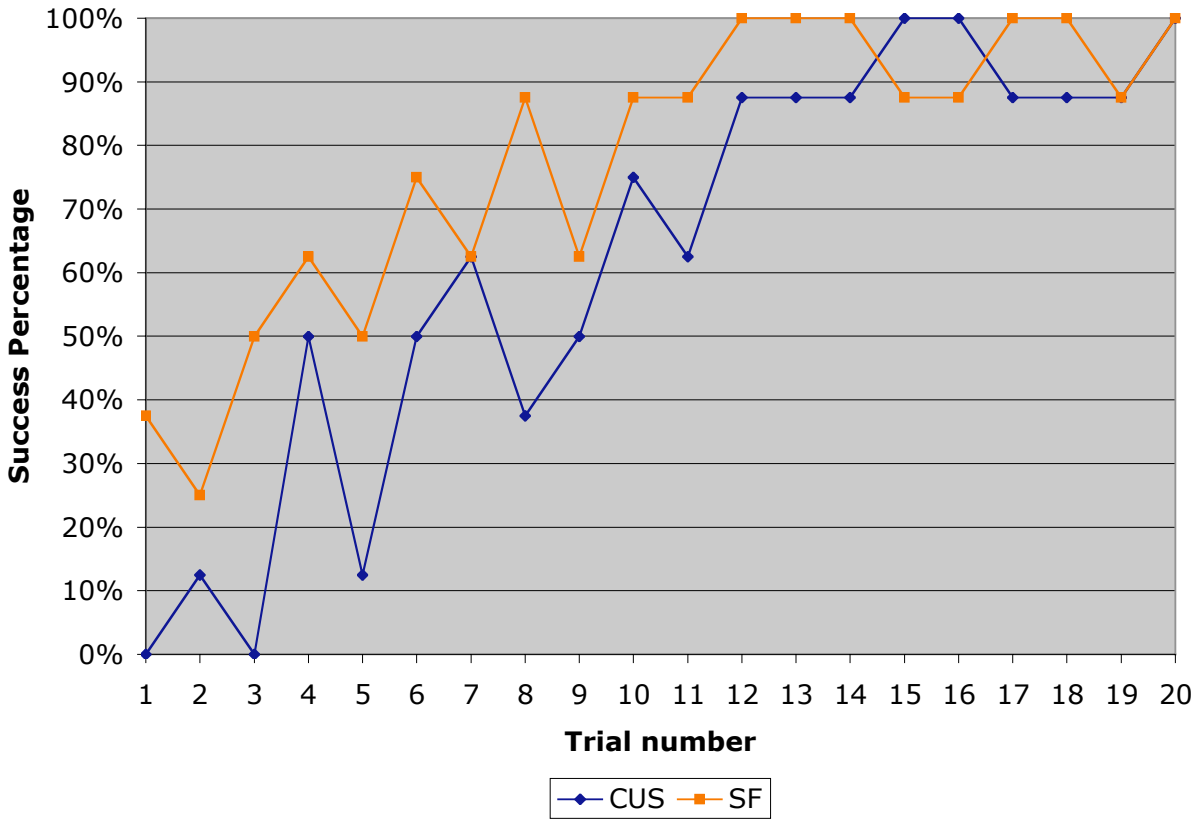


Figure 5.21 Percentages of subjects finishing under 7 secs at Task C

The SF and CUS curves generally parallel each other on all three tasks. At the start of day two for the subjects (i.e. trial 11), there is a drop in the number of subjects able to complete the task under the threshold times, most noticeable in task A. The most likely explanation for this drop is the subjects "unlearned," or forgot, the skills in the week between days 1 and 2 of the study. It is apparent from the curves that one to two trials at each task were required for the subjects to reach or surpass the previous proficiency level. This effect is also visible in Figure 5.9, Figure 5.10, and Figure 5.11 as an increase in average trial time at trial 11. It is unclear

why task A shows the greatest change (unlearning) at trial 11, even though task B could be considered the harder task (longer times).

A third method to compare learning between the SF and CUS is to plot the percentage of successful trials as a function of time. These curves show the percentage of all trials over all subjects completed below a certain time, starting at 0 secs with a 0% success rate and climbing to 100% success rate. The trials have been separated into two sets, trials 1 through 10, and 11 through 20, to illustrate how proficiency changes with more experience.

The SF curve is higher than the CUS curve during trials 1 through 10 at Task A (Figure 5.22), with both the SF and CUS curves steepening during trials 11-20 (Figure 5.23). The task B and C curves show similar findings, with flatter task B curves due to the increased task difficulty (Figure 5.24, Figure 5.25), and steeper task C curves due to the decreased task difficulty (Figure 5.26, Figure 5.27). The response curves of the SF and CUS quickly separate, and the overall tendency is that the SF curve is above the CUS curve.

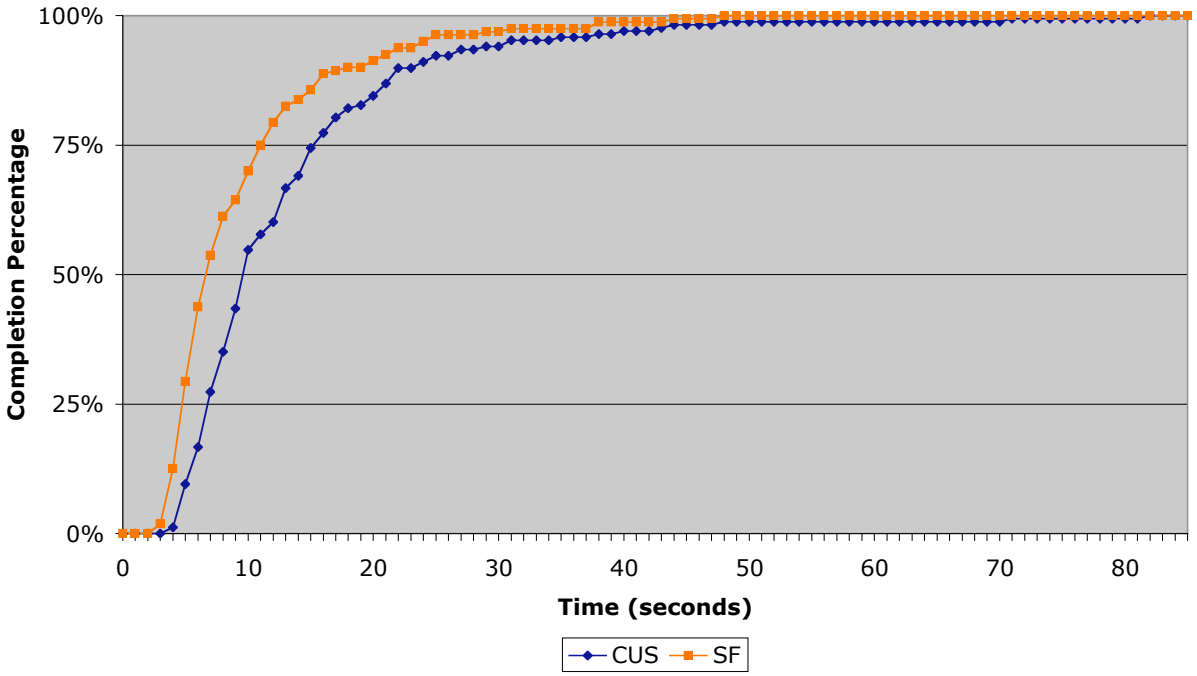


Figure 5.22 Pooled trials 1-10 completion rate vs time over all subjects at Task A

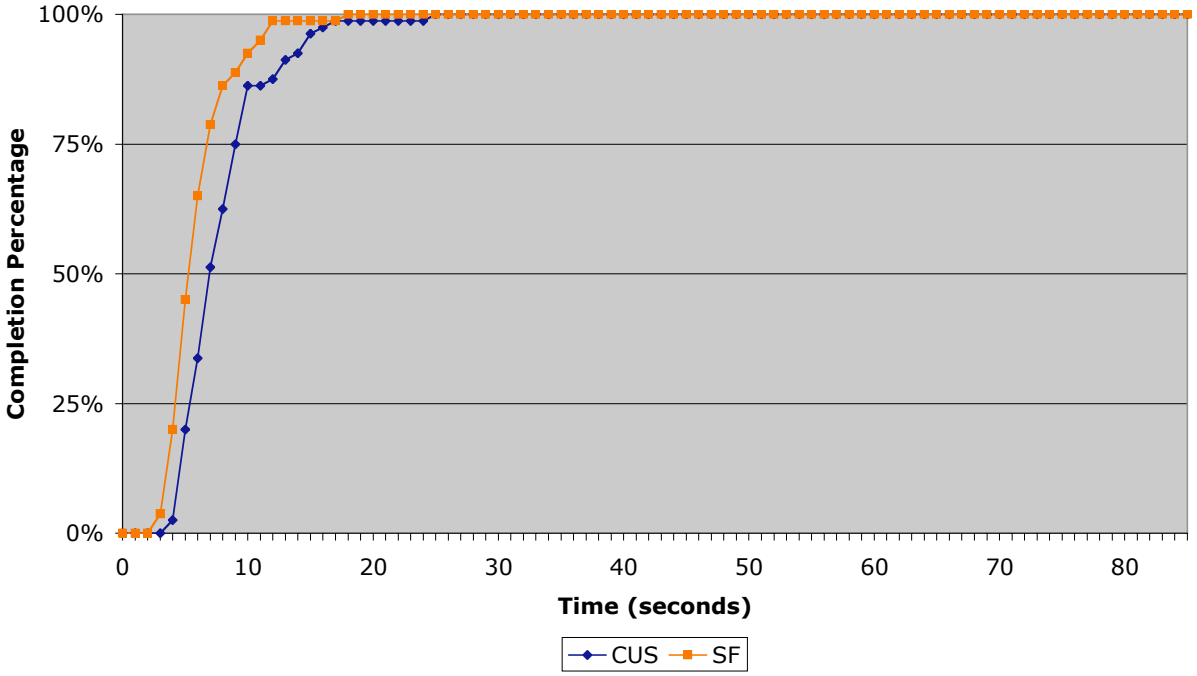


Figure 5.23 Pooled trials 11-20 completion rate vs time over all subjects at Task A

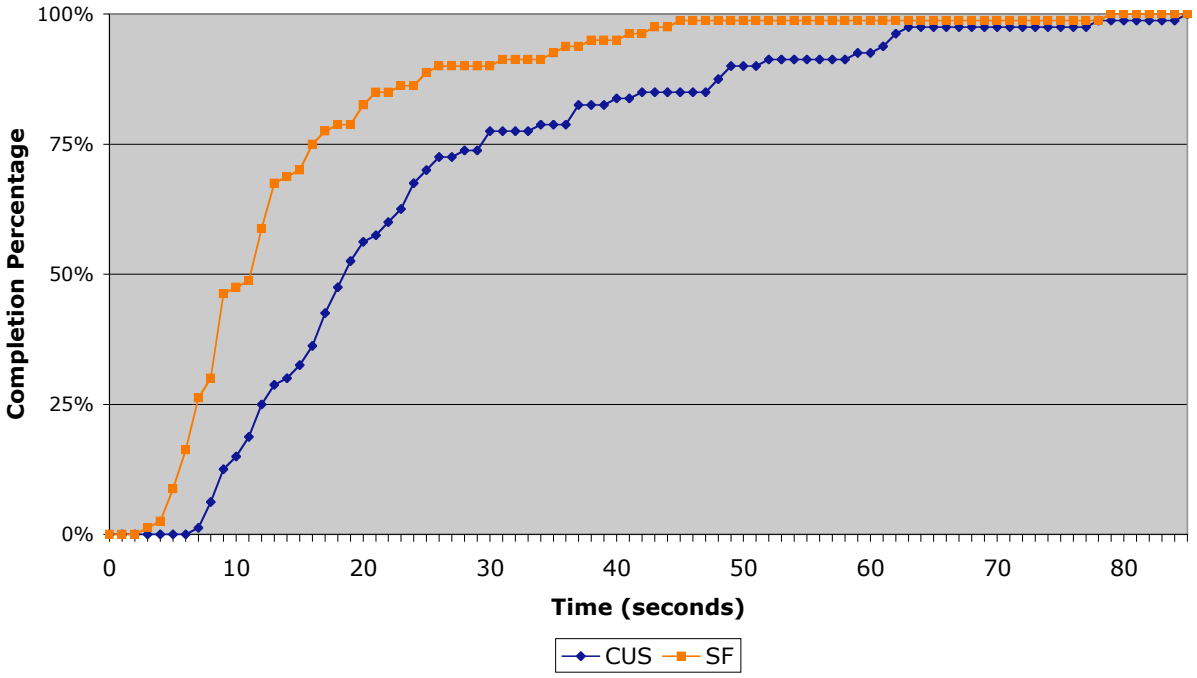


Figure 5.24 Pooled trials 1-10 completion rate vs time over all subjects at Task B

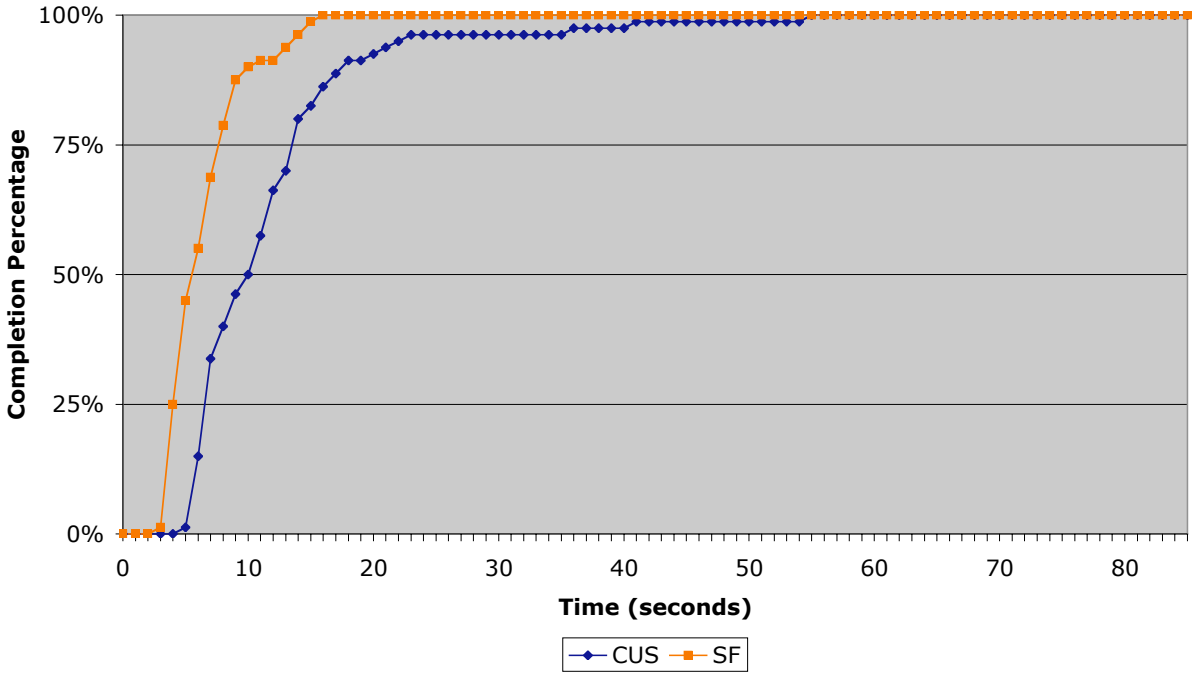


Figure 5.25 Pooled trials 11-20 completion rate vs time over all subjects at Task B

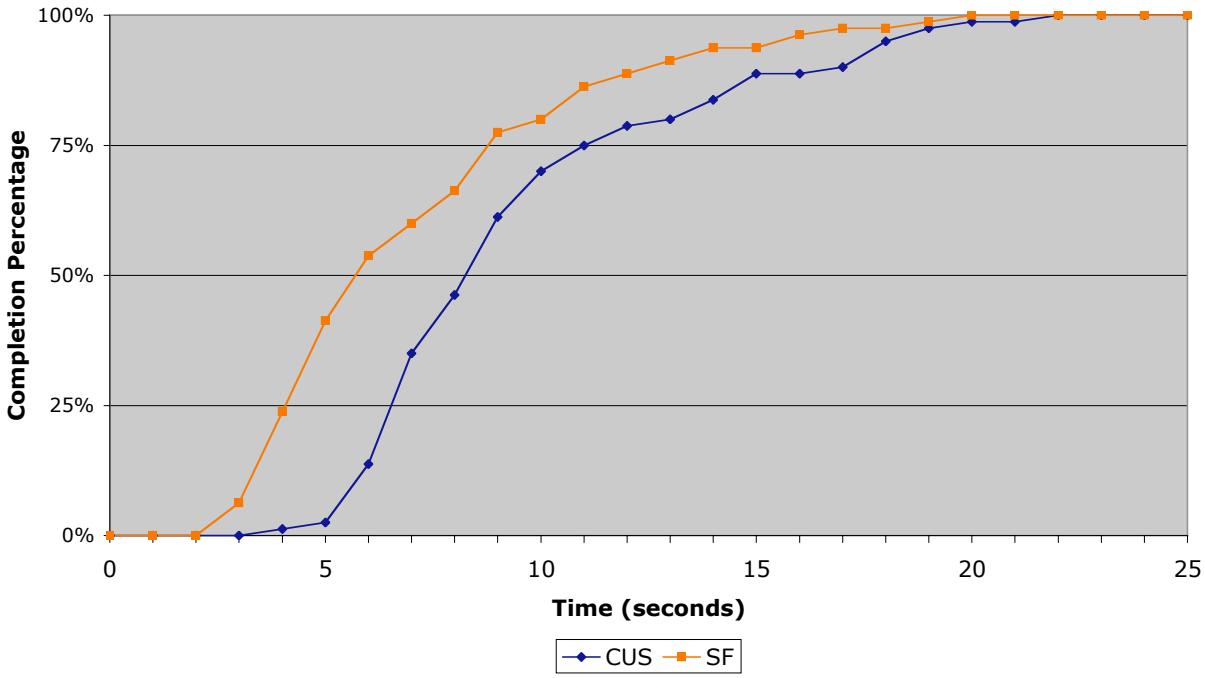


Figure 5.26 Pooled trials 1-10 completion ate vs time over all subjects at Task C

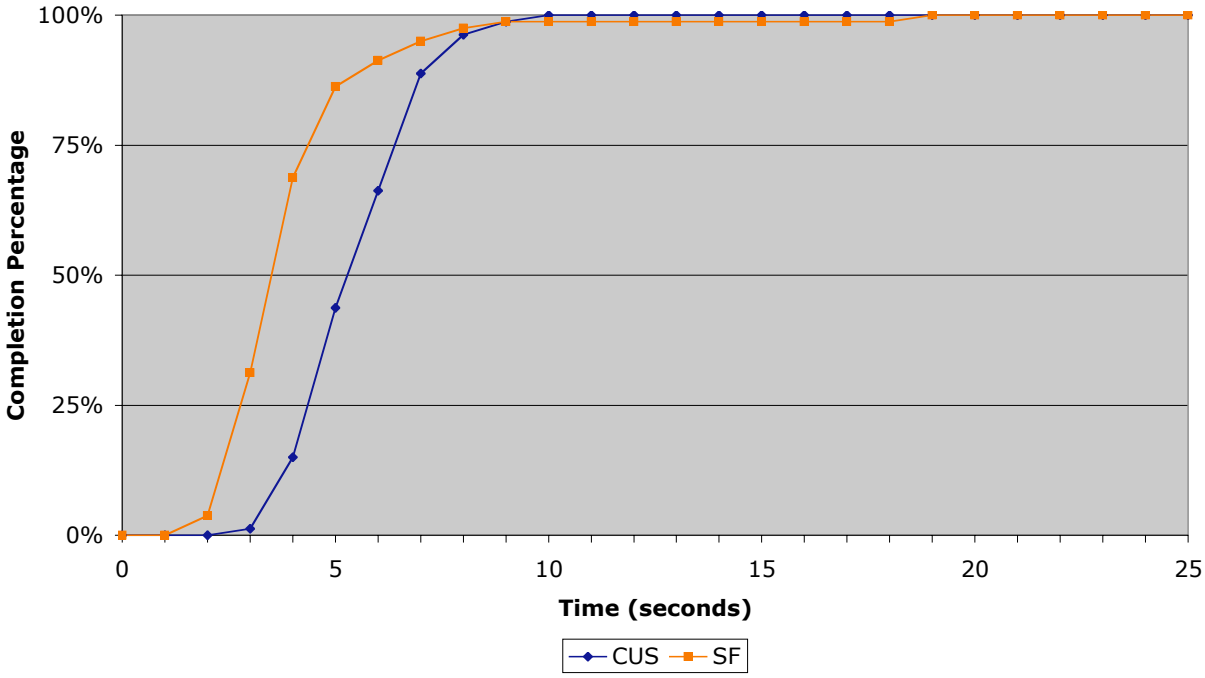


Figure 5.27 Pooled trials 11-20 completion rate vs time over all subjects at Task C

As the subjects learned the tasks and became more proficient, their rate of learning decreased and their times approached an asymptote. While a true asymptote is something that is only approached but never achieved, for this work, an asymptote time represents a level of proficiency at a task that is difficult to improve upon. A reasonable approximation of the asymptote time is the average time of the last five trials for each task, since for each task, these trials resulted in very similar times. The 95% confidence interval (CI) around the asymptote time is defined as the standard error of the mean (SEM) multiplied by 1.96. The trial (considering only the trials 16 and before, since trials 16-20 were used to estimate the asymptote time) following the earliest two successive trials with average times outside the asymptote's 95% CI was identified as the asymptote trial number.

The 95% CI for the mean asymptote time at task A using the SF was 4.9 ± 0.8 secs (range of mean times: 4.3 to 5.4 secs), compared to 6.2 ± 0.8 secs (range of mean times: 5.8 to 6.6 secs) for subjects using CUS (Figure 5.28, Figure 5.29). At task A, subjects reached the asymptote with the SF at trial 13, compared to trial 16 for the CUS group. Similar calculations were conducted for tasks B and C, as show in Figure 5.30 to Figure 5.33, with the results summarized in Table 5.2. A two-tailed T-test comparison of asymptote times showed that there was a significant difference at all three tasks between the SF and CUS (Task A $p < 0.0008$; Task B $p < 0.0003$; Task C $p < 0.0007$)

Table 5.2 Learning Asymptote Time and Trial Number

Task	*Modality*	*Asymptote Time (95% CI, in secs)*	*Asymptote Trial Number*	*p <*
A	SF	4.9 ± 0.8	13	0.0008
	CUS	6.2 ± 0.8	16	
B	SF	5.4 ± 1.4	14	0.0003
	CUS	9.0 ± 1.9	14	
C	SF	3.4 ± 0.8	10	0.0007
	CUS	5.1 ± 0.6	12	

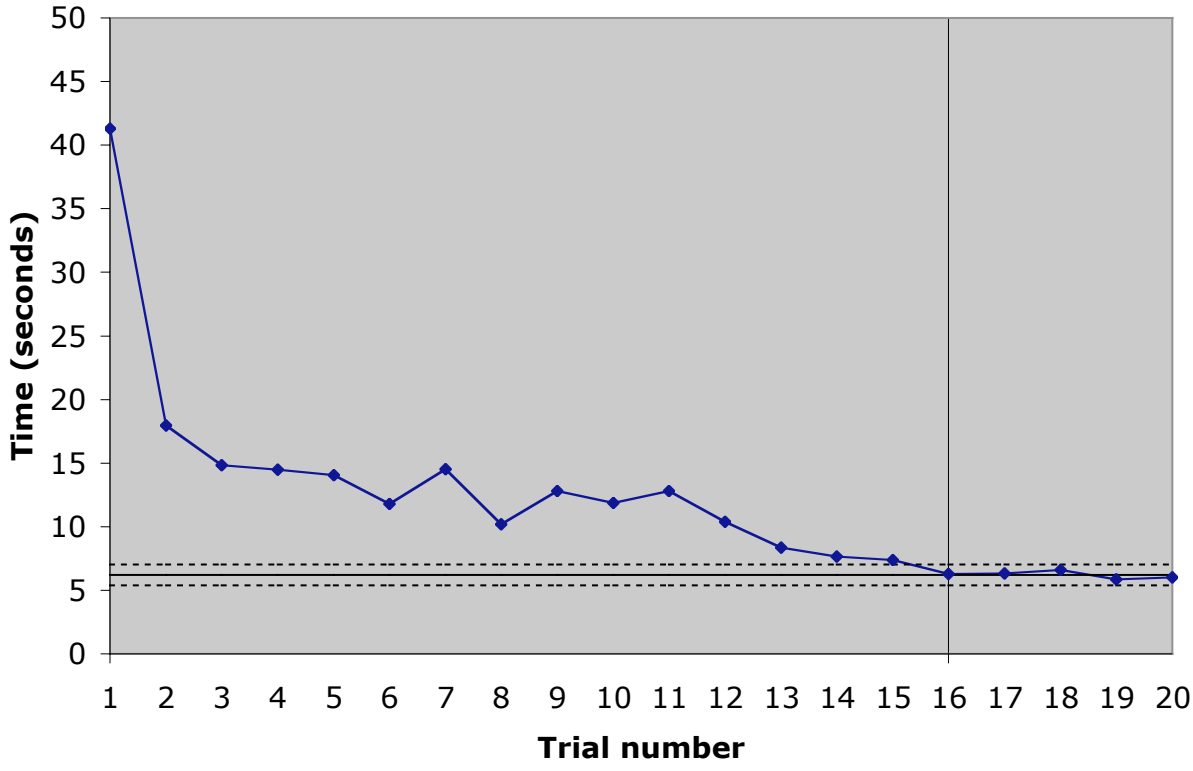


Figure 5.28 Task A learning asymptote for CUS (from Figure 5.9)
Asymptote (solid horizontal line), 95% Confidence interval (dashed line),
Trial reaching asymptote (solid vertical line)

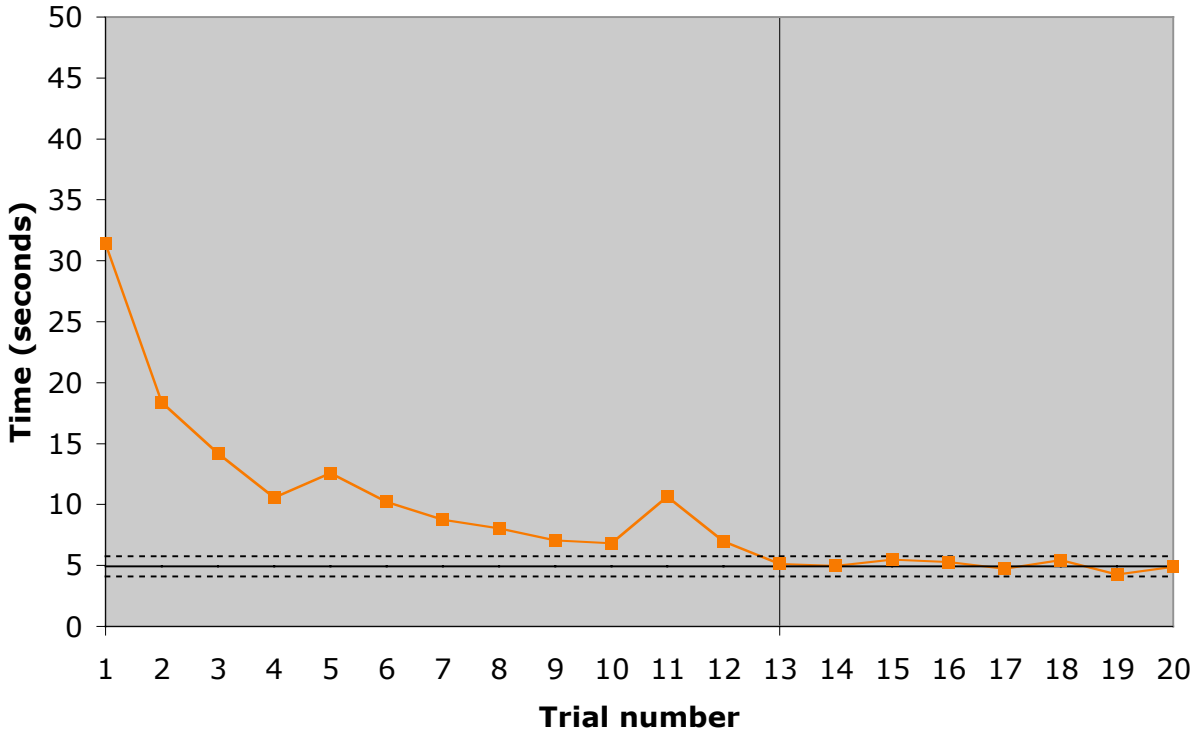


Figure 5.29 Task A learning asymptote for the SF (from Figure 5.9)
Asymptote (solid horizontal line), 95% Confidence interval (dashed line),
Trial reaching asymptote (solid vertical line)

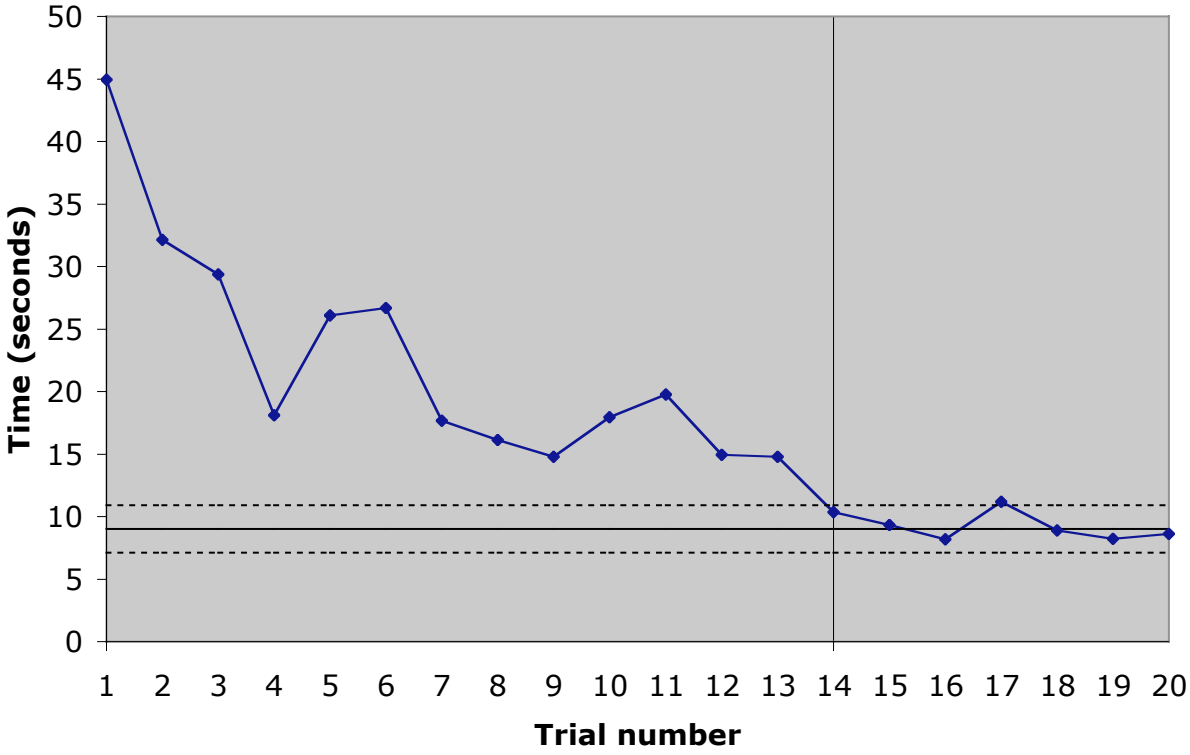


Figure 5.30 Task B learning asymptote for CUS (from Figure 5.10)
Asymptote (solid horizontal line), 95% Confidence interval (dashed line),
Trial reaching asymptote (solid vertical line)

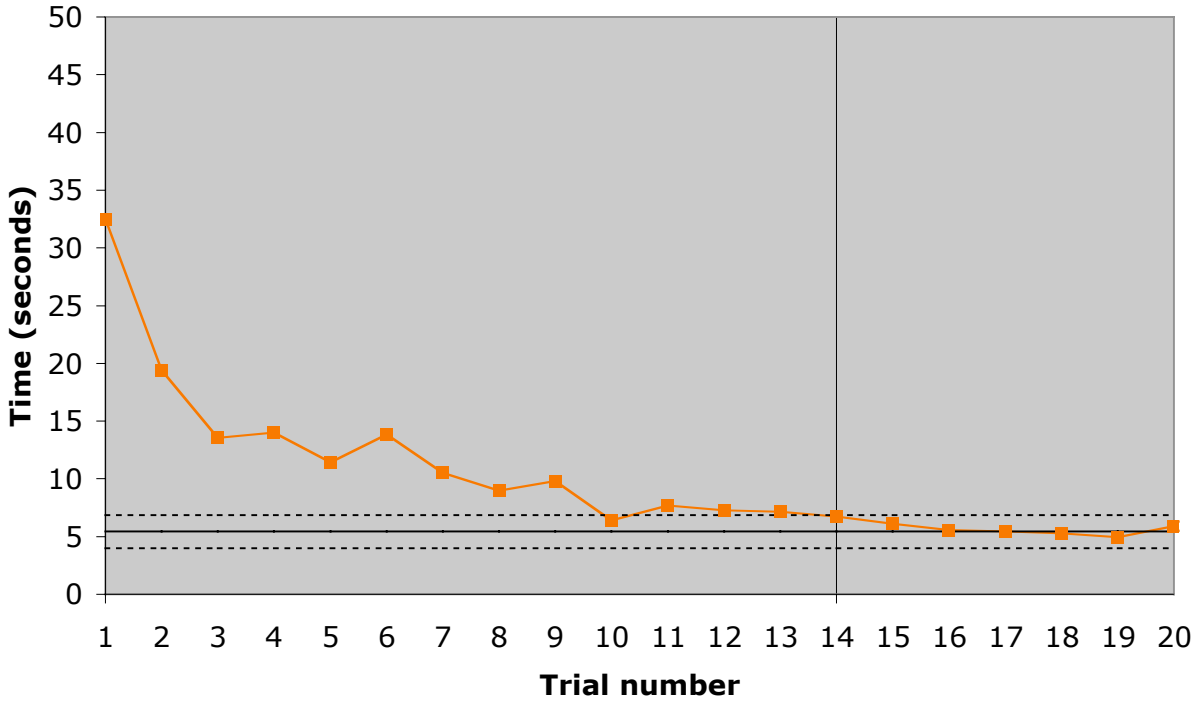


Figure 5.31 Task B learning asymptote for the SF (from Figure 5.10)
Asymptote (solid horizontal line), 95% Confidence interval (dashed line),
Trial reaching asymptote (solid vertical line)

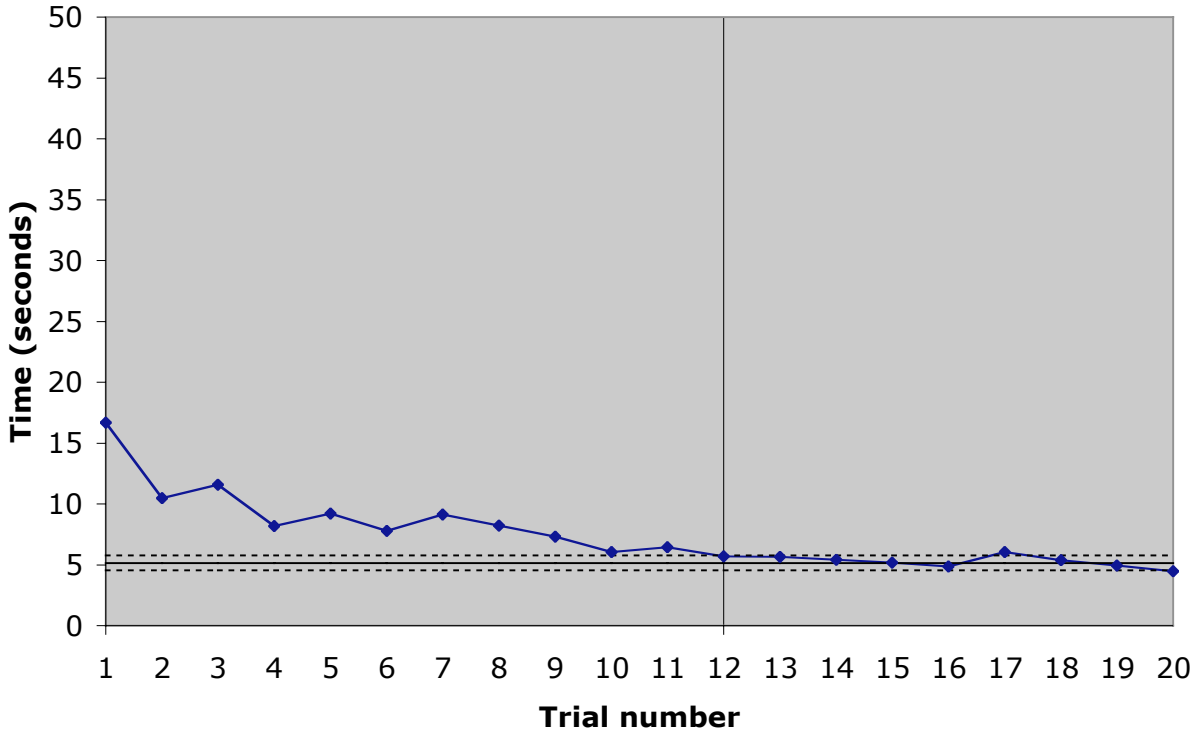


Figure 5.32 Task C learning asymptote for CUS (from Figure 5.11)
Asymptote (solid horizontal line), 95% Confidence interval (dashed line),
Trial reaching asymptote (solid vertical line)

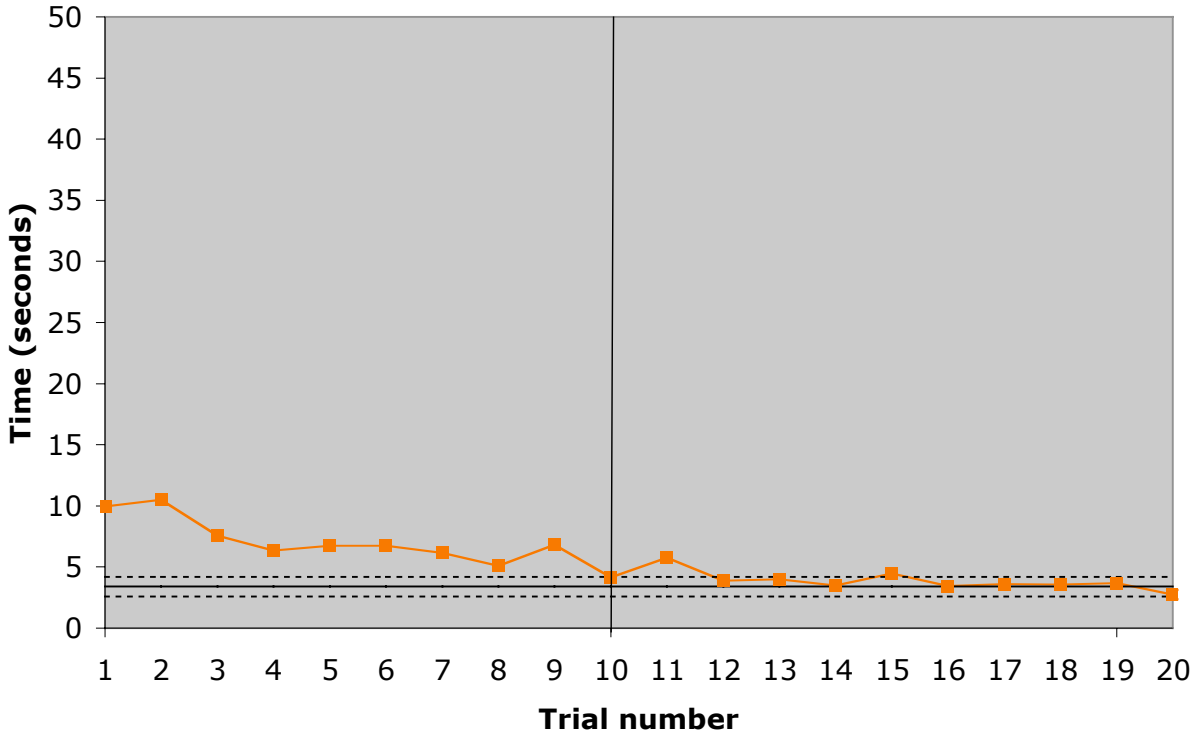


Figure 5.33 Task C learning asymptote for the SF (from Figure 5.11)
Asymptote (solid horizontal line), 95% Confidence interval (dashed line),
Trial reaching asymptote (solid vertical line)

The data collected in this study shows that learning vascular access on phantoms is significantly faster with the SF than the CUS. Significantly lower asymptote times were achieved for the SF group than the CUS group in all three tasks. Subjects reached asymptote in the same or fewer trials using the SF than the CUS group. This study also confirmed the results of the pilot study. These results strongly support the claim that the SF is not only easier to learn than CUS in novice users, but also easier to use once learning has been accomplished.

5.4 SF VS. CUS COMPARISON IN PROFICIENT CUS USERS

The previous studies showed that novice users are faster at learning and subsequently performing vascular access using the SF than CUS. However, there are a number of current CUS users that are not novices, such as IV access team nurses, that are potential SF users. Since it would be unreasonable to only train CUS novices on the SF, it is important to compare SF and CUS guidance in proficient CUS users. The following study makes this comparison for IV nurses already proficient in CUS guided PICC lines, without prior experience using the SF.

The study population consisted of 14 nurses from the University of Pittsburgh Medical Center IV access team. Using the same vascular phantom setup and same tasks as the novice learning study (section 5.4), each subject performed 24 vascular access tasks with each device. The subjects were randomized to begin with the SF or CUS, perform 24 trials, switch devices, and perform 24 more trials. Similar to the previous study, the tasks were randomized within blocks, with each block containing one task A, one task B, and one task C (Figure 5.7). The CUS machine used in this study was a Terason 2000 with a 10MHz probe, the same US machine used in the SF. Before using each device, the subjects were given a short tutorial on how to use the SF or CUS, and how to perform the procedures before beginning. The CUS and SF tutorials consisted of a description on how to use the device, what the US image represented, and how to locate, aim, and guide a needle in a cross-sectional US scan of a vessel, followed by a hands-off demonstration.

For each subject, the first 6 trials (2 trials at each task) were practice trials for familiarizing the subjects to the experimental procedures and equipment. Therefore, 18 trials per

modality per subject were used in the data analysis[1]. After completing all the trials on both devices, subjects were asked to complete a multiple answer questionnaire containing 6 questions. These questions asked their opinions on the usability of the SF and how it compared to CUS guidance. The results of this study and the questionnaire are compiled in Appendix E and summarized below.

The mean trial time for all 14 subjects to complete task A using the SF was 5.8 secs (SEM = 0.4). Using CUS guidance, the mean time was 7.1 secs (SEM = 0.5), as shown in Figure 5.34. The average time difference between SF and CUS guidance per subject was 0.3 secs (SEM = 0.5), as shown in Figure 5.35. A one-tailed paired t-test revealed did not show a significant difference between SF and CUS guidance times at task A ($p = 0.258$).

The mean trial time for all 14 subjects to complete task B using the SF was 5.9 secs (SEM = 0.5). Using CUS guidance, the mean time was 10.8 secs (SEM = 1.4), as shown in Figure 5.34. The mean time difference between SF and CUS guidance for each subject was 5.4 secs (SEM = 2.0), as shown in Figure 5.35. A one-tailed paired t-test revealed a significant difference between SF and CUS guidance at task B ($p = 0.009$).

The mean trial time for all 14 subjects to complete task C using the SF was 5.1 secs (SEM = 0.6). Using CUS guidance, the mean time was 7.3 secs (SEM = 0.6), as shown in Figure 5.34. The mean time difference between SF and CUS guidance for each subject was 1.7 secs (SEM = 0.9), as shown in Figure 5.35. A one-tailed paired t-test revealed a significant difference between SF and CUS guidance at task C ($p = 0.047$).

[1] The first three subjects to complete the study only completed 18 trials per device, including the 6 trials to familiarize themselves with the equipment. Therefore, only 12 trials per device were used in the analysis of these 3 subjects. This difference was due to a protocol change to increase the amount of data collected in the study. This protocol change was made after the first 3 subjects already completed the study.

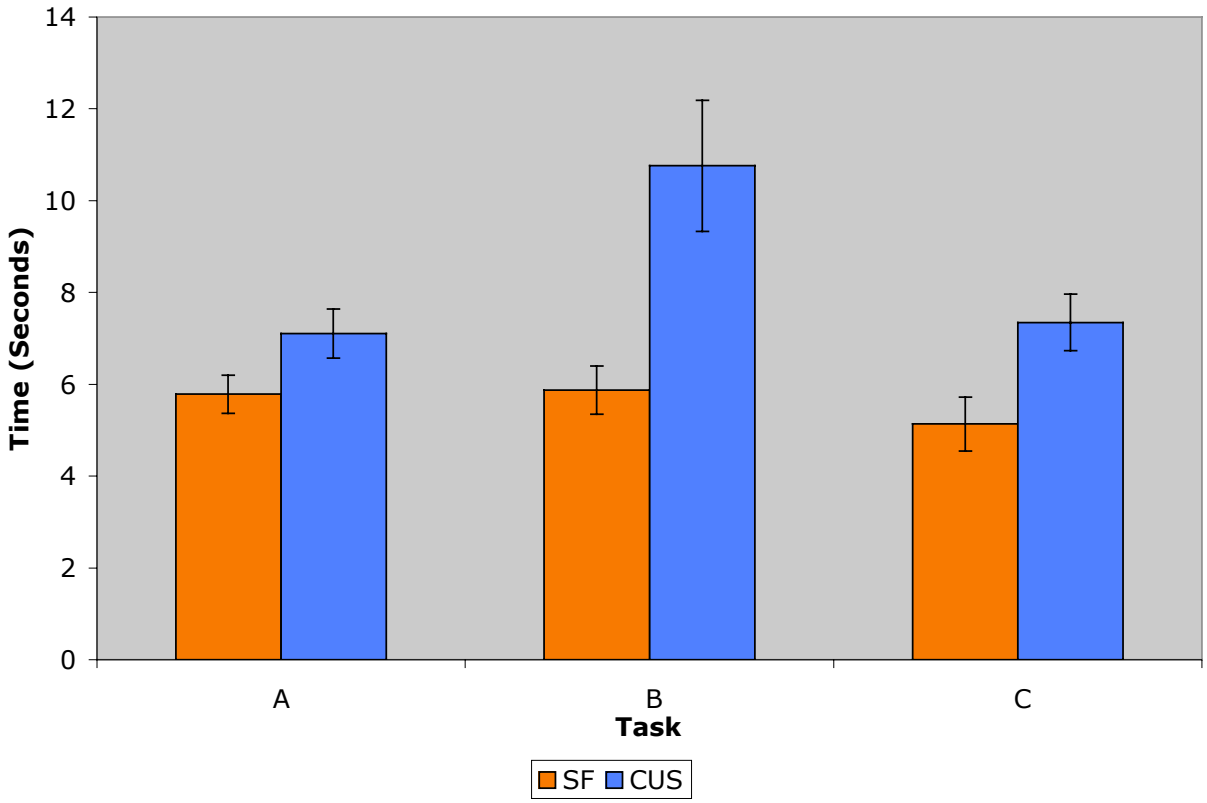


Figure 5.34 Vascular phantom access times with CUS proficient nurses

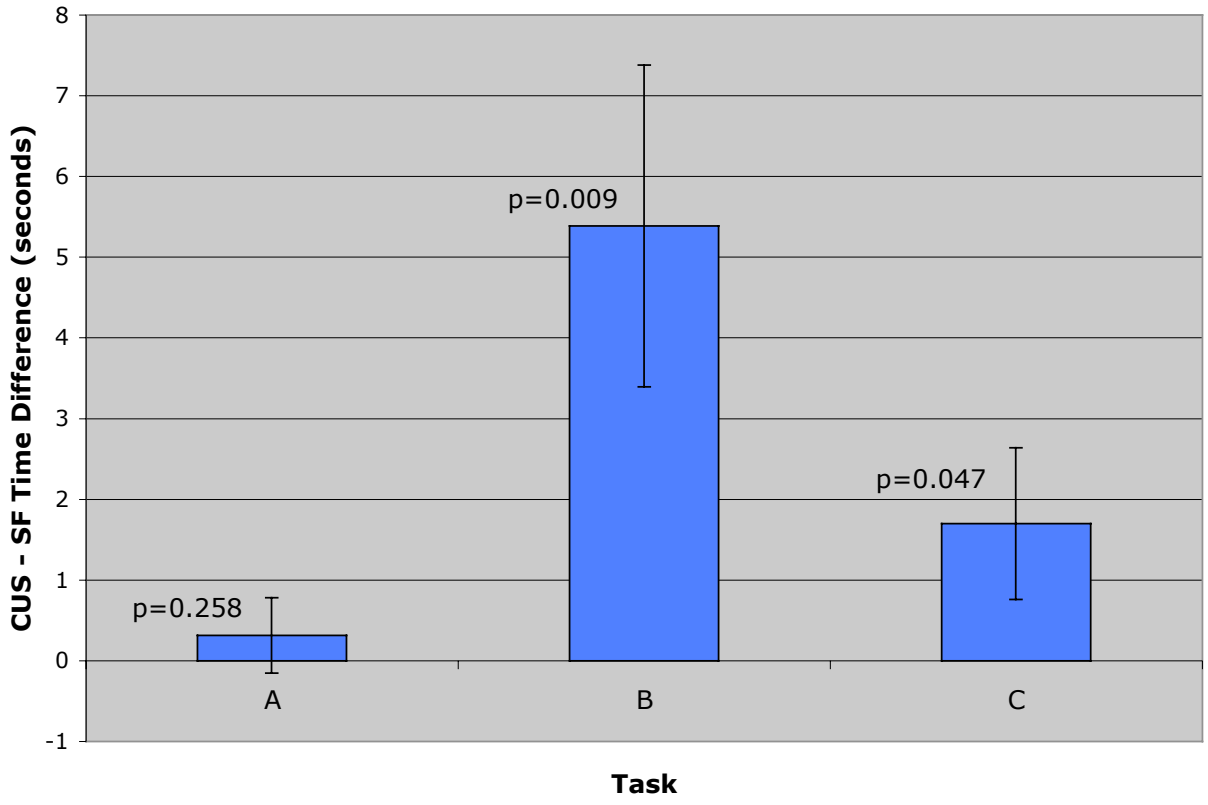


Figure 5.35 SF versus CUS vascular access time difference in CUS proficient nurses (CUS - SF times)

On question 1, the majority of subjects found the procedure "much easier" (64% or 9 of 14) or "somewhat easier" (29% or 4 of 14) to perform using the SF rather than CUS. 7% of the subjects, or 1 out of 14, found the procedure "about the same" to perform using the SF rather than the CUS. None of the subjects found the procedure harder using the SF rather than CUS guidance.

Question 2 asked, "The ultrasound image using the SF is _____ [to] interpret than on CUS." The majority (64%) of the subjects found the US image using the SF easier to interpret

than on the CUS, with 43% (6 of 14) answering it was "much easier," and 21% (3 of 14) answering that it was "somewhat easier." 35% (5 of 14) of the subjects stated that the US image was "about the same" to interpret.

One of our concerns with the SF was that users might find the mirror an obstruction, both physically and optically. Therefore, we asked whether the subjects found that the mirror physically got in the way of the procedure. Half the subjects disagreed strongly (21% or 3 of 14) or somewhat disagreed (29% or 4 of 14), while the other half of the subjects somewhat agreed (43% or 6 of 14) or agreed strongly (7% or 1 of 14). Asked whether "looking through the mirror made it more difficult to perform the procedure," the majority (78%) of the subjects disagreed, with 57% (8 of 14) strongly disagreeing, and 21% somewhat disagreeing. 14% of the subjects, or 2 of 14, responded that the neither agreed or disagreed. 7% of the subjects, or 1 of 14, somewhat agreed, and none of the subjects agreed strongly with the statement that looking through the mirror made the procedure more difficult.

Compared to the magnified US image displayed on most CUS monitors, the US image appearing on the SF display is generally much smaller, since it is, by definition, "life sized." On the questionnaire, subjects were asked whether, "compared to the size of the ultrasound image on the CUS, the ultrasound image on the SF made interpretation" easier or harder. 57% of the subjects reported that the size of the SF image made interpretation "much easier" (29% or 4 of 14) or "somewhat easier" (29% or 4 of 14) than the CUS image. Another 29% subjects (4 of 14) reported that the size made the SF image "neither easier nor harder." The final 14% (2 of 14) answered that it made it somewhat harder. No subjects responded that the SF image size made interpretation much harder than the CUS image size.

The final question asked whether "having the ultrasound image appear inside of the phantom [helped or hindered] in aiming and guiding the needle." 93% of the subjects (13 of 14) responded that having the US image inside of the phantom helped in aiming and guiding. 7% (1 of 14) responded that it neither helped nor hindered while none of the subjects responded that it hindered.

The results of this study show that proficient CUS users are faster overall at the three tasks using SF guidance compared to CUS guidance. This result is even more significant given that the subjects had no prior experience with the SF, compared to CUS guidance, which they use nearly everyday in their work. When the analysis is performed separately for the three tasks, two out of the three tasks, the easiest (task C) and hardest (task B), showed that the SF was significantly faster than CUS. It is not clear why subjects did not show a significant time difference at task A, the medium difficulty task. The questionnaire revealed that the majority of subjects found:

1. The procedures easier to perform using the SF.
2. The US image easier to interpret on the SF than CUS.
3. Looking through the mirror does not make the procedure more difficult to perform.
4. The size of the US on the SF display makes interpretation of the US easier.
5. Having the US image appear in the phantom helps in aiming and guiding the needle.

Subjects were also split 50/50 on whether the mirror physically got in the way of the procedure. Again, these results are significant because the subjects had no prior experience with the SF, and had to modify their guidance technique when using the SF.

5.5 VASCULAR ACCESS IN THE CADAVER

Prior to the first clinical trials using the SF to guide vascular access in humans, we have chosen to perform these procedures in an unpreserved cadaver. The cadaver was a female of unrevealed age and cause of death, heparinized prior to death. The neck and right upper arm were scanned using the SF to identify the internal structures, and the needle was aimed and inserted into the IJV and basilic vein, sites that would normally be used for central catheters and PICCs.

The internal anatomy was clearly visualized in situ using the SF, with the carotid artery and IJV easily identified in the neck, and basilic vein clearly identified in the arm. The needle was aimed and inserted into the IJV and basilic vein on the first attempts, and the needle tip visualized at its expected location (Figure 5.36 and Figure 5.37). When the needle entered the veins, blood freely flowed out of the needle hub, as can be seen in Figure 5.37.

This study shows that it is possible to gain central venous access guided by the SF. While not a thorough comparison between SF and CUS guidance in multiple cadavers with multiple operators, with the results of this study, and our extensive studies using phantoms, we feel that it is ready for clinical trials.

Figure 5.36 Sonic Flashlight guided vascular access into the internal jugular vein of a cadaver

Note the needle tip within the internal jugular vein

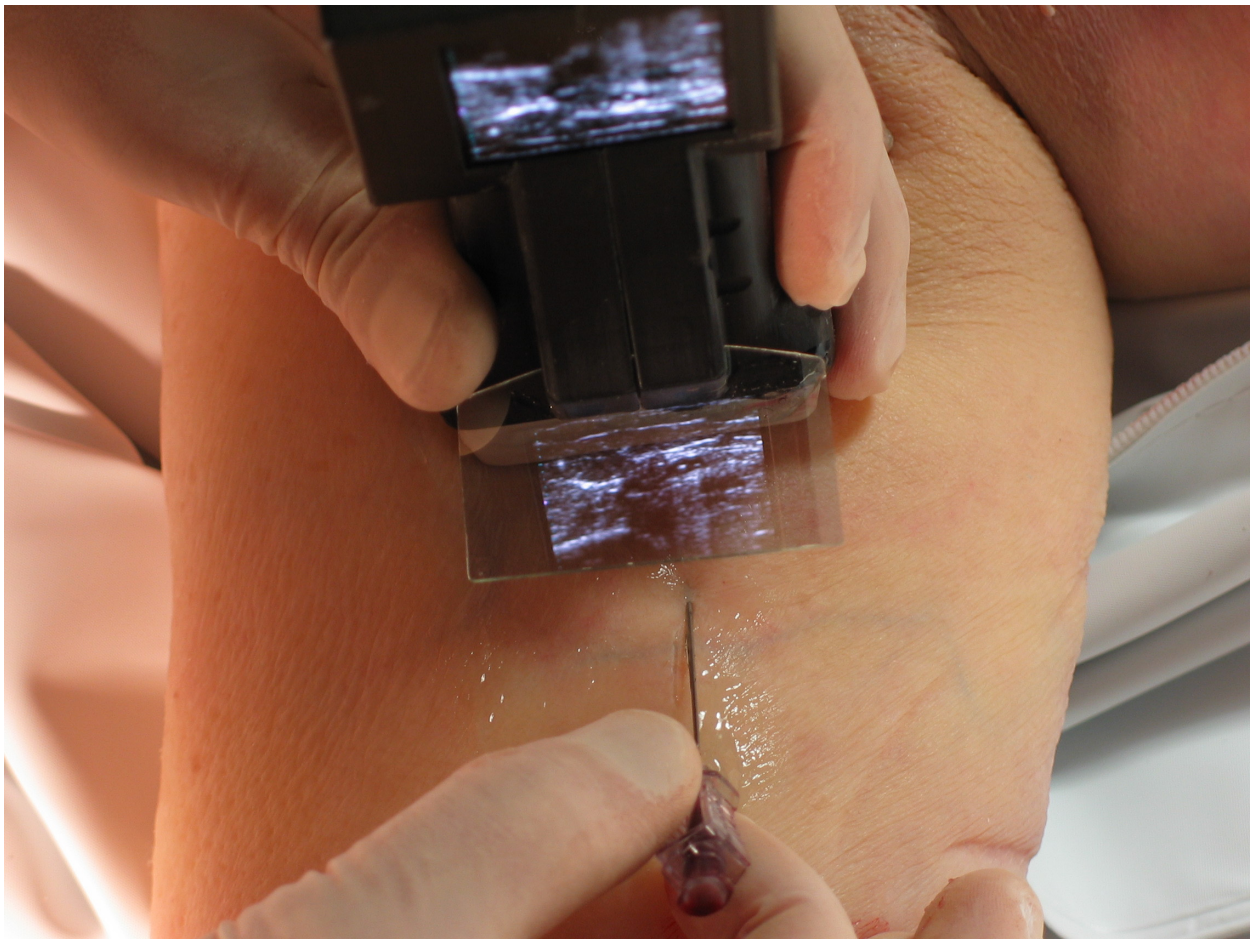


Figure 5.37 Sonic Flashlight guided vascular access into the right basilic vein of a cadaver

Note the needle tip within the basilic vein and free blood flow out of the needle

6.0 CONCLUSIONS

This dissertation has described the development, calibration, and pre-clinical validation of the 5th generation SF. The SF represents a simple augmented reality device to display an US image within the body, exactly where it is being scanned. To achieve this effect, accurate calibration is necessary and is made possible by the methods developed in this work. The SF ease of use and learning has been shown in a number of populations that are potential users of such a device. While potential applications of the SF are not limited to vascular access, this dissertation has focused on developing the SF towards vascular access, due to this application's relative importance and widespread use. In this chapter, we recap the key contributions of the dissertation and discuss future directions for the research.

6.1 KEY CONTRIBUTIONS

This dissertation makes three major contributions. The first is the development of the 5th generation SF. By developing this prototype around the Terason US system, it is lightweight, small, fast, and robust. The scanning and display resolutions have also been improved significantly over previous versions, with the US image remaining completely in the digital domain. With an OLED display, the fidelity of the US is preserved on the SF monitor. In nearly every respect, the SF has been improved, to the point that this version is considered ready to be used in the first clinical trials, once issues of sterility have been addressed.

The second major contribution of this dissertation is the development of a novel calibration method for the SF. Unlike previous methods, the thin-gel calibration system allows simultaneous merger of a live US image through the SF and an unrefracted, direct view of the calibration targets. This same system can also be used to measure the calibration accuracy, important especially in light of the aim toward clinical trials. A variety of target configurations were explored to find a configuration that minimized calibration error. Therefore, our confidence in this calibration method significantly exceeds that for the previous methods.

The third major contribution of this dissertation is validation of the SF. Using vascular phantoms, we have showed that the SF is significantly easier to learn and use then CUS for US novices. With proficient CUS users, we have also seen faster vascular access using the SF than CUS, even without any prior experience or training using the SF. Finally, in a cadaver, we have shown vascular access into the IJV and basilic vein, two of the most commonly used sites for central venous access.

6.2 FUTURE DIRECTIONS

We have made significant progress towards the first clinical trials of the SF for vascular access. The last remaining task before a clinical trial involves making the SF easier to use in a sterile field. We have designed a "hood" adaptation for the SF that encloses the monitor and mirror, with an opening, or viewport, through which the user looks to view the SF. This enclosure will make fitting a sterile sleeve over the entire SF easier, while helping with reflections and distraction from direct view of the OLED.

We believe that then next logical step following the modification for sterilization is the first clinical trial placing SF guided PICC lines. In addition to vascular access, there are numerous other applications where the SF may prove useful, including tumor biopsy, arthrocentesis, amniocentesis, drainage catheter placement, and intraoperative neurosurgical guidance. While some of these applications may use the same or very similar SF to that developed in this dissertation, applications such as neurosurgical guidance may require specialized versions of the SF based on unique US probes.

Other adaptations to the SF are planned to facilitate psychophysical studies of the way humans can make use of embedded virtual images. We are currently developing a version of the SF, which we call the *virtual SF* (33). Instead of a live US scanner, this system uses positional trackers to display virtual data on the SF, based on its position. To the user, the virtual SF functions almost identically to the SF developed in this dissertation, while providing a much more easily adaptable phantom. It will allow us to study aspects of human perception and SF usage not possible with the current version. By tracking the position of invasive instruments, the accuracy with which they can be placed relative to computer-generated phantoms will be studied.

We are also continuing development of a 3D version of the SF based on 3D US. Our first prototype of this SF displayed C-mode US slices, instead of the B-mode slices of the SF developed in this dissertation (39). We are planning a future version of the 3D SF that tracks the relative location of the 3D US probe and the monitor. This version would allow the user to move the monitor freely, while the SF system automatically displays the correct US slice.

Instead of the half-slivered mirror used in all of the previously discussed SFs, we are developing a holographic element replacement for the mirror. This holographic element would

allow us to create a SF that could magnify or shrink the virtual image, or even display non-planar virtual images.

Finally, the SF may be adapted to other imaging modalities other than US, including optical coherence tomography, confocal microscopy, MRI, CT, and sonar. With such a wide variety of imaging modalities, the SF may find applications and utility far into the future. In the near term, by making US guided procedures easier and more intuitive, the SF should allow health-care professionals to perform US guided procedures faster, with less specialized training, with increased patient safety, while lowering the cost of health care.

APPENDIX

APPENDIX A

SONIC FLASHLIGHT SOFTWARE

SONICFLASHLIGHTCONSTANTS.H

```
// Size of the texture to which the video is mapped
const double TEXTURE_SIZE_X = 512;
const double TEXTURE_SIZE_Y = 512;

// Terason ModifyImageDisplay constants
const long DEPTH_RULER = 0x00000001;
const long FRAME_RATE = 0x00000002;
const long MEASUREMENT_VALUE = 0x00000004;
const long ORIENTATION_LOGO = 0x00000008;
const long PATIENT_INFORMATION = 0x00000010;
const long PROBE_INFORMATION = 0x00000020;
const long REFERENCE_BAR = 0x00000040;
const long TGC_DISPLAY = 0x00000080;
const long ZOOM_THUMBNAIL = 0x00000100;
const long BODY_MARKER = 0x00000200;
const long NEEDLE_GRID = 0x00000400;

const short MIN_SCAN_DEPTH = 35;
const short MAX_SCAN_DEPTH = 80;

const short SCAN_DEPTH_STEP = 2;
const double ZERO = 1E-13;
```

SONICFLASHLIGHTMATH.H

```
/*
 *  SFMath.h
 *
 *
 *  Created by Wilson Chang on Sun Jan 26 2003.
 *  Copyright (c) 2003. All rights reserved.
 *
 *  Does all the matrix math necessary for the Sonic Flashlight
 */
#ifndef __SFCALIBRATEMATH_H__
#define __SFCALIBRATEMATH_H__

#include <iostream.h>
#include <fstream>
```

```
//the determinant of a 3x3
double det3x3(double a[][3]/* **a*/)
{
  return (a[0][0] * (a[2][2] * a[1][1] - a[2][1] * a[1][2]) -
          a[1][0] * (a[2][2] * a[0][1] - a[2][1] * a[0][2]) +
          a[2][0] * (a[1][2] * a[0][1] - a[1][1] * a[0][2]));
}

//the inverse of a 3x3
double** inverse3x3(double a[][3] /* **a*/)
{
  //the inverse determinant
  double id = 1/det3x3(a);
  double **b;
  b = new double*[3];
  for (int i = 0; i < 3; i++)
    b[i] = new double[3];

  b[0][0] = id * (a[2][2]*a[1][1] - a[2][1]*a[1][2]);
  b[1][0] = id * -1 * (a[2][2]*a[1][0] - a[2][0]*a[1][2]);
  b[2][0] = id * (a[2][1]*a[1][0] - a[2][0]*a[1][1]);

  b[0][1] = id * -1 * (a[2][2]*a[0][1] - a[2][1]*a[0][2]);
  b[1][1] = id * (a[2][2]*a[0][0] - a[2][0]*a[0][2]);
  b[2][1] = id * -1 * (a[2][1]*a[0][0] - a[2][0]*a[0][1]);

  b[0][2] = id * (a[1][2]*a[0][1] - a[1][1]*a[0][2]);
  b[1][2] = id * -1 * (a[1][2]*a[0][0] - a[1][0]*a[0][2]);
  b[2][2] = id * (a[1][1]*a[0][0] - a[1][0]*a[0][1]);
  return b;
}

double** transpose3x3(double a[][3])
{
  double **b;
  b = new double*[3];
  for (int i = 0; i < 3; i++)
    b[i] = new double[3];

  for (int j = 0; j < 3; j++)
    for (int k = 0; k < 3; k++)
      b[j][k] = a[k][j];

  return b;

}

//returns a dot b for 3x3 matricies
double** dotProduct3x3and3x3(double a[][3], double **b)
{

  double **s;
  s = new double*[3];
  for (int i = 0; i < 3; i++)
  {
    s[i] = new double[3];
    for (int j=0; j<3; j++)
      s[i][j] = 0.0;
  }

  for (int r = 0; r < 3; r++)
    for (int c = 0; c < 3; c++)
      for (int i = 0; i < 3; i++)
        s[r][c] += a[r][i] * b[i][c];
  return s;
}

//returns a dot b for 3x3 matricies
double** dotProduct3x3and3x3(double **a, double b[][3])
{

  double **s;
```

```
  s = new double*[3];
  for (int i = 0; i < 3; i++)
  {
    s[i] = new double[3];
    for (int j=0; j<3; j++)
      s[i][j] = 0.0;
  }

  for (int r = 0; r < 3; r++)
    for (int c = 0; c < 3; c++)
      for (int i = 0; i < 3; i++)
        s[r][c] += a[r][i] * b[i][c];
  return s;
}

//returns a dot b, where a is a 3x3 and b is a 3x4
double** dotProduct3x3and3x4(double a[][3], double b[][4])
{
  double **s;
  s = new double*[3];
  for (int i = 0; i < 3; i++)
  {
    s[i] = new double[4];
    for (int j = 0; j < 4; j++)
      s[i][j] = 0.0;
  }

  for (int r = 0; r < 3; r++)
    for (int c = 0; c < 4; c++)
      for (int i = 0; i < 3; i++)
        s[r][c] += a[r][i] * b[i][c];
  return s;
}

void dotProduct3x3and3x4(double a[][3], double b[][4], double
result[][4])
{
  double **s;
  s = new double*[3];
  for (int i = 0; i < 3; i++)
  {
    s[i] = new double[4];
    for (int j = 0; j < 4; j++)
      s[i][j] = 0.0;
  }

  for (int r = 0; r < 3; r++)
    for (int c = 0; c < 4; c++)
      for (int i = 0; i < 3; i++)
        s[r][c] += a[r][i] * b[i][c];

  for (int w = 0; w < 3; w++)
    for (int x = 0; x < 4; x++)
      result[w][x] = s[w][x];

  for (int y = 0; y < 3; y++)
    delete s[y];
  delete s;
}


void printArray3x3(double a[][3])
{
  for (int i = 0; i < 3; i++)
  {
    std::cout << "[ ";
    for (int j = 0; j < 3; j++)
    {
      std::cout << a[i][j] << " ";
    }
    std::cout << "]" << std::endl;
  }
```

```
}

void printArray(double **a, int n, int m)
{
  for (int i = 0; i < n; i++)
  {
    std::cout << "[ ";
    for (int j = 0; j < m; j++)
    {
      std::cout << a[i][j] << " ";
    }
    std::cout << "]" << std::endl;
  }
}
void printArray(double a[][4], int n, int m)
{
  for (int i = 0; i < n; i++)
  {
    std::cout << "[ ";
    for (int j = 0; j < m; j++)
    {
      std::cout << a[i][j] << " ";
    }
    std::cout << "]" << std::endl;
  }
}

void printArray(double a[][3], int n, int m)
{
  for (int i = 0; i < n; i++)
  {
    std::cout << "[ ";
    for (int j = 0; j < m; j++)
    {
      std::cout << a[i][j] << " ";
    }
    std::cout << "]" << std::endl;
  }
}
void printArray(double **a, int n, int m, std::ofstream &out)
{
  for (int i = 0; i < n; i++)
  {
//    out << "[ ";
    for (int j = 0; j < m; j++)
    {
      out << a[i][j] << " ";
    }
    out /*<< "]" */<< std::endl;
  }
  out << std::endl;
}
void printArray(double a[][3], int n, int m, std::ofstream &out)
{
  for (int i = 0; i < n; i++)
  {
//    out << "[ ";
    for (int j = 0; j < m; j++)
    {
      out << a[i][j] << " ";
    }
    out /*<< "]" */<< std::endl;
  }
  out << std::endl;
}
int factorial(int n)
```

```
{
  if (n <= 1) return 1;
  else return n*factorial(n-1);
}

//implements ( n ) as n! / ((n-m)! * m! )
//           ( m )
int NChooseM(int n, int m)
{
  return (factorial(n) / (factorial(n-m) * factorial(m)));
}

#endif
```

INVISIBLEDIALOG.H

```
//{{AFX_INCLUDES()
#include "ttframereceiver.h"
#include "ttautomate.h"
//}}AFX_INCLUDES
#if
!defined(AFX_INVISIBLEDIALOG_H__151B2405_62BB_40EE_81E3_330AEC895DC2__I
NCLUDED_)
#define
AFX_INVISIBLEDIALOG_H__151B2405_62BB_40EE_81E3_330AEC895DC2__INCLUDED_

#include "RGBTriple.h"

#if _MSC_VER > 1000
#pragma once
#endif // _MSC_VER > 1000
// InvisibleDialog.h : header file
//

/////////////////////////////////////////////////////////////////////////
//////
// InvisibleDialog dialog

class InvisibleDialog : public CDialog
{
// Construction
public:
  InvisibleDialog(CWnd* pParent = NULL);   // standard constructor
// Dialog Data
  //{{AFX_DATA(InvisibleDialog)
  enum { IDD = IDD_DIALOG1 };
  CTTFrameReceiver  m_FrameRx;
  CTTAutomate m_TTAutomate;
  //}}AFX_DATA


// Overrides
  // ClassWizard generated virtual function overrides
  //{{AFX_VIRTUAL(InvisibleDialog)
  protected:
  virtual void DoDataExchange(CDataExchange* pDX);        //  DDX/DDV
support
  //}}AFX_VIRTUAL

// Implementation
public:
  CGLRGBTRIPLE* m_TextureData;
  void RegisterTextureData(CGLRGBTRIPLE* tex, int texw, int texh);
  //this should find the width of the black border of the image;
  bool m_NewTextureFrameReady;
  int IncrementScanDepth();
  int DecrementScanDepth();
  void invertColors() { m_InvertColors = !m_InvertColors; }

  short m_ScanDepth;
```

```
//  void ResetOffset();
private:
  int m_texWidth, m_texHeight, m_StorageWidth, m_rawWidth, m_rawHeight;
  int m_Offset;
  int FindOffset(LPBITMAPINFO lpInfo, HBITMAP hBitmap);
  bool m_firstTimeOffset;
  bool m_InvertColors;
protected:

  // Generated message map functions
  //{{AFX_MSG(InvisibleDialog)
  afx_msg void OnFrameReady();
  DECLARE_EVENTSINK_MAP()
  //}}AFX_MSG
  DECLARE_MESSAGE_MAP()
};

//{{AFX_INSERT_LOCATION}}
// Microsoft Visual C++ will insert additional declarations immediately
before the previous line.

#    e    n    d    i    f                                                //
!defined(AFX_INVISIBLEDIALOG_H__151B2405_62BB_40EE_81E3_330AEC895DC2__I
NCLUDED_)
```

INVISIBLEDIALOG.CPP

```
// InvisibleDialog.cpp : implementation file
//   This is where the US data is copied from a DIB to the openGL
texture memory

#include "stdafx.h"
#include "resource.h"
#include "InvisibleDialog.h"
#include "SonicFlashlightConstants.h"

#ifdef _DEBUG
#define new DEBUG_NEW
#undef THIS_FILE
static char THIS_FILE[] = __FILE__;
#endif

/////////////////////////////////////////////////////////////////////////
//////
// InvisibleDialog dialog


InvisibleDialog::InvisibleDialog(CWnd* pParent /*=NULL*/)
  : CDialog(InvisibleDialog::IDD, pParent)
{
  //{{AFX_DATA_INIT(InvisibleDialog)
  //}}AFX_DATA_INIT
  m_ScanDepth = 74;
  m_firstTimeOffset = true;
  m_Offset = 0;
  m_InvertColors = true;
}


void InvisibleDialog::DoDataExchange(CDataExchange* pDX)
{
  CDialog::DoDataExchange(pDX);
  //{{AFX_DATA_MAP(InvisibleDialog)
  DDX_Control(pDX, IDC_TTFRAMERECEIVERCTRL1, m_FrameRx);
  DDX_Control(pDX, IDC_TTAUTOMATECTRL1, m_TTAutomate);
  //}}AFX_DATA_MAP
}


BEGIN_MESSAGE_MAP(InvisibleDialog, CDialog)
```

```
  //{{AFX_MSG_MAP(InvisibleDialog)
    // NOTE: the ClassWizard will add message map macros here
  //}}AFX_MSG_MAP
END_MESSAGE_MAP()

////////////////////////////////////////////////////////////////////////////
//////
// InvisibleDialog message handlers

BEGIN_EVENTSINK_MAP(InvisibleDialog, CDialog)
    //{{AFX_EVENTSINK_MAP(InvisibleDialog)
  ON_EVENT(InvisibleDialog, IDC_TTFRAMERECEIVERCTRL1, 1 /* FrameReady
*/, OnFrameReady, VTS_NONE)
  //}}AFX_EVENTSINK_MAP
END_EVENTSINK_MAP()

void InvisibleDialog::RegisterTextureData(CGLRGBTRIPLE* tex, int texw,
int texh)
{
  m_texWidth = texw;
  m_texHeight = texh;
  m_TextureData = tex;
}

int InvisibleDialog::IncrementScanDepth()
{
  if (m_ScanDepth <= MAX_SCAN_DEPTH - SCAN_DEPTH_STEP)
    m_ScanDepth += SCAN_DEPTH_STEP;
  m_TTAutomate.SetDepth(m_ScanDepth);
  m_firstTimeOffset = true;
  std::cout<<m_ScanDepth<<std::endl;
  return m_ScanDepth;
}


int InvisibleDialog::DecrementScanDepth()
{
  if (m_ScanDepth >= MIN_SCAN_DEPTH + SCAN_DEPTH_STEP)
    m_ScanDepth -= SCAN_DEPTH_STEP;
  m_TTAutomate.SetDepth(m_ScanDepth);
  m_firstTimeOffset = true;
  std::cout<<m_ScanDepth<<std::endl;
  return m_ScanDepth;
}



/*
void InvisibleDialog::ResetOffset()
{
  m_firstTimeOffset = true;
}
*/

//finds the offset from the left edge.
//TOO SLOW TO USE.  it casues OnFrameReady to have memory access errors
because it is way to slow
int InvisibleDialog::FindOffset(LPBITMAPINFO lpInfo, HBITMAP hBitmap)
{
      std::cout<< h    "<<lpInfo->bmiHeader.biWidth<<" ,    "<<lpInfo-
>bmiHeader.biHeight<<std::endl;

  m_firstTimeOffset = false;
  int lookwidth = lpInfo->bmiHeader.biWidth < TEXTURE_SIZE_X ? lpInfo-
>bmiHeader.biWidth : TEXTURE_SIZE_X;
  std::cout<<"lookwidth "<<lookwidth<<std::endl;

  const int LOOKHEIGHT = lpInfo->bmiHeader.biHeight / 2;
  int *sumArray = new int[lookwidth];
  for (int w = 0; w < lookwidth; w++)
  {
    sumArray[w] = 0;
```

```
  }
  if (lpInfo->bmiHeader.biBitCount == 8)
  {
    //DIBs are DWORD aligned meaning there is padding to the nearest 4
bytes
    //pointer that will move through the source image
    BYTE* pSrc = (BYTE*)m_FrameRx.GetPtrBitmapBits();
    RGBQUAD* pClrTab = (RGBQUAD*)lpInfo->bmiColors;
    int widthDiff = lpInfo->bmiHeader.biWidth - lookwidth;
    int counter = 0;
    for (int j = 0; j < LOOKHEIGHT; j++)
    {
      counter = 0;
      for (int i = 0; i < lookwidth; i++)
      {
        i f  (pClrTab[*pSrc].rgbBlue  |   pClrTab[*pSrc].rgbGreen  |
pClrTab[*pSrc].rgbRed)
        {
          sumArray[counter]++;
        }
        counter++ ;
        pSrc++ ;
      }
      pSrc += widthDiff;
    }
  }
  for (int k = 0; k < lookwidth; k++)
  {
    if ((sumArray[k]) && (k > 0))
    {
      delete sumArray;
      std::cout<<"offset "<<(k-1)<<std::endl;
      return (k-1);
    }
  }
  delete sumArray;
  std::cout<<" offset 0"<<std::endl;
  return 0;
}


void InvisibleDialog::OnFrameReady()
{
  //when a frame is ready, copy it out from DIB format to OpenGL
texture memory format

  LPBITMAPINFO lpInfo = (LPBITMAPINFO)m_FrameRx.GetPtrBitmapInfo();
  HBITMAP hBitmap = (HBITMAP)m_FrameRx.GetPtrBitmapBits();

  if (lpInfo && hBitmap)
  {

    if (m_firstTimeOffset)
      m_Offset = FindOffset(lpInfo, hBitmap);

    //This processing assumes biBitCount = 8.
    if (lpInfo->bmiHeader.biBitCount == 8)
    {
      //DIBs are DWORD aligned meaning there is padding to the nearest
4 bytes
      //pointer that will move through the source image
      int widthDiff = lpInfo->bmiHeader.biWidth - m_texWidth;
      BYTE* pSrc = (BYTE*)hBitmap;
      RGBQUAD* pClrTab = (RGBQUAD*)lpInfo->bmiColors;
      pSrc += m_Offset;
      int counter = 0;
      for (int j = 0; j < m_texHeight; j++)
      {
        for (int i = 0; i < m_texWidth; i++)
        {
          if (m_InvertColors)
```

```
            {
              m_TextureData[counter].rgbRed = ~pClrTab[*pSrc].rgbRed;
              m_TextureData[counter].rgbGreen =  ~pClrTab[*pSrc].rgbGreen
;
              m_TextureData[counter].rgbBlue = ~pClrTab[*pSrc].rgbBlue ;
            }
            else
            {
              m_TextureData[counter].rgbRed = pClrTab[*pSrc].rgbRed;
              m_TextureData[counter].rgbGreen = pClrTab[*pSrc].rgbGreen ;
              m_TextureData[counter].rgbBlue = pClrTab[*pSrc].rgbBlue ;
            }
            counter++ ;
            pSrc++ ;
          }
          pSrc += widthDiff;

        }
      }
    }
    m_NewTextureFrameReady = true;
    m_FrameRx.ReleaseFrame();
  }
```

SONICFLASHLIGHT.H

```
#if
!defined(AFX_SONICFLASHLIGHT_H__EBF70044_4F75_4F78_B0AB_6B6932DC4B73__I
NCLUDED_)
#define
AFX_SONICFLASHLIGHT_H__EBF70044_4F75_4F78_B0AB_6B6932DC4B73__INCLUDED_

#if _MSC_VER > 1000
#pragma once
#endif // _MSC_VER > 1000

#include "resource.h"
#include "InvisibleDialog.h"
#include "RGBTriple.h"
#include "SonicFlashlightConstants.h"
#include <iostream.h>
#include <fstream>


//using namespace _DSHOWLIB_NAMESPACE;
#pragma warning (disable : 4251)

// atexit() workaround for Win32
// Needed with Win32 platforms so that the application doesn't abort
without
// cleaning up after itself.
// Thanks to Luc-Eric Rousseau (whoever you are) for the code!
#ifdef _WIN32
    #define WIN32_LEAN_AND_MEAN
    #include <windows.h>

    HHOOK g_hMyHook;
    LRESULT CALLBACK MyGetMsgProc( int code, WPARAM wParam, LPARAM
lParam )
    {
    MSG * pMsg = (MSG *)lParam;
    if ( pMsg->message == WM_QUIT )
    {
      UnhookWindowsHookEx( g_hMyHook ); // futile..
      exit( 0 ); // force calling of atexit callback
    }
    return CallNextHookEx( g_hMyHook, code, wParam, lParam );
  }
```

```
  void HookGlut()
  {
    g_hMyHook = SetWindowsHookEx( WH_GETMESSAGE, MyGetMsgProc, NULL,
GetCurrentThreadId() );
  }
#else // Not on a Win32 platform, HookGlut() does nothing
  void HookGlut() { }
#endif


// Functions

  void CleanupApp(void);
  void HookGlut();
  void GrabFrame();
  void DisplayFunction();
  void ProcessNormalKeys(unsigned char key, int x, int y);
  void ProcessSpecialKeys(int key, int x, int y);
  void ArrowLeft();
  void ArrowRight();
  void ArrowUp();
  void ArrowDown();
  void Resize(int width, int height);
  bool InitializeTerason();
  void InitializeGL();
  void LoadManualSettings();
  void SaveManualSettings();
  void LoadCalibrationAffineTransform();
  void SaveCalibrationAffineTransform();
  bool CalculateAffineTransform();
  void ApplyAffineTransform();
  void OpenCalibrationLogs();
  void CloseCalibrationLogs();
  void LogCalibration();
  void ResetCalibration();

  // The invisible dialog window that contains the
  // TTFrameReceiver and TTAutomate ActiveX controls
  InvisibleDialog dlg;

  bool showTTAutomate = true;

  // The size of the video frame as displayed on screen
  // Note that the cropped image is some subset of the pixels
  // of the texture, with (0,0) in the cropped image being (0,0)
  // in the texture, and pixels in the texture not defined by the
  // cropped image set equal to black
  double croppedImageSizeX = TEXTURE_SIZE_X;
  double croppedImageSizeY = TEXTURE_SIZE_Y;

  int windowSizeX = 640;
  int windowSizeY = 480;

  // Image data arrays
  CGLRGBTRIPLE* textureData = NULL;

  enum calibrationMode{
    crosshair0,
    crosshair1,
    crosshair2,
    ultrasound0,
    ultrasound1,
    ultrasound2};
  const int NUM_CALIBRATION_STEPS = 6;

  int calibrationStep = -1;

  // The crop amounts are integers that specify how much to "indent"
each dimension
  int cropTop = 0;
  int cropBottom = 0;
  int cropLeft = 0;
```

```
  int cropRight = 0;

  // Scale and translations applied to the quad.  Used in the old
method of calibration
  double scaleX = 1;
  double scaleY = 1;
  double translateX = 0;
  double translateY = 0;

  //calibration variables
  double initTargetCoords [3][3] = {{0.0, 0.0, 0.0} , {0.1, 0.0, -0.1}
, {1, 1, 1}};
  double endingTargetCoords [3][3] = {{0, 0, 0} , {0.1, 0.0, -0.1} ,
{1, 1, 1}};

  double ultrasoundTargetOffset[2] = {0, 0};

  // the lengt of the cross hairs used in the calibration
  float m_TargetLength = 0.07;

  // the multiplier constants to convert from openGL coords to mm on
the OLED screen
  const double measurementMultiplier[2] = {16.615, 21.648} ;

  int measurementMode = false;
  double measurementTargetOffset[2] = {0, 0};

  //rotation
  int rotation = 0;

  // The size to "step" by when changing the scale or translation
  double stStep = 0.005;

  double calibrationAffineTransform [3][3] =  {{1,0,0},  {0,1,0},
{0,0,1}};
  double resetAffineTransform [3][3] = {{1,0,0}, {0,1,0}, {0,0,1}};

  //this defines where the corners of the texture (ultrasound) will be
displayed on the screen
  //column 1 (ie [][0]) is the lower left, default at (-1,-1, 1)
  //column 2 (ie [][1]) is the upper left, default at (-1, 1, 1)
  //column 3 (ie [][2]) is the upper right,default at ( 1, 1, 1)
  //column 4 (ie [][3]) is the lower right,default at ( 1,-1, 1)

  double currentDisplayCoordinates [3][4] = {{-1,-1,1,1}, {-1,1,1,-1},
{1,1,1,1}};
  double initializationDisplayCoords [3][4] = {{-1,-1,1,1}, {-1,1,1,-
1}, {1,1,1,1}};
  bool freeze = false;

  std::ofstream coordinateLog;
  std::ofstream transformLog;

#    e    n    d    i    f                                          //
!defined(AFX_SONICFLASHLIGHT_H__EBF70044_4F75_4F78_B0AB_6B6932DC4B73__I
NCLUDED_)
```

SONICFLASHLIGHT.CPP

```
// SonicFlashlight.cpp : Defines the entry point for the console
application.
//

#include "stdafx.h"
#include "SonicFlashlight.h"
#include "tt2000.h"
#include "RGBTriple.h"
#include "SonicFlashlightMath.h"
#include "math.h"
//#include <fstream.h>
```

```
#ifdef _DEBUG
#define new DEBUG_NEW
#undef THIS_FILE
static char THIS_FILE[] = __FILE__;
#endif

//////////////////////////////////////////////////////////////////////////
//////
// The one and only application object

CWinApp theApp;

using namespace std;

int _tmain(int argc, TCHAR* argv[], TCHAR* envp[])
{
  int nRetCode = 0;

  // initialize MFC and print and error on failure
  if (!AfxWinInit(::GetModuleHandle(NULL),   NULL,  ::GetCommandLine(),
0))
  {
    // TODO: change error code to suit your needs
    std::cerr << _T("Fatal Error: MFC initialization failed") << endl;
    nRetCode = 1;
  }
  else
  {

    AfxOleInit();
    AfxEnableControlContainer();
    CoInitialize(NULL);

    // Create the invisible dialog window that contains the
    // TTFrameReceiver and TTAutomate ActiveX controls.
    // Also, get the pointer to the texture memory stored in the dialog
    dlg.Create(IDD_DIALOG1);
    t  e  x  t  u  r  e  D  a  t  a                                    =
(CGLRGBTRIPLE*)malloc(TEXTURE_SIZE_X*TEXTURE_SIZE_Y*sizeof(CGLRGBTRIPLE
));
    dlg.RegisterTextureData(textureData,     TEXTURE_SIZE_X,
TEXTURE_SIZE_Y);
    dlg.m_NewTextureFrameReady = false;

    // Register the callback that cleans up our memory
    // when the application exits. atexit is called by
    // default on non-windows platforms and manually
    // by the HookGlut() function on win32 machines
    int i = atexit( CleanupApp );

    // Intercept windows messages to avoid atexit() problems
    // Does nothing on non-win32 platforms
    HookGlut();

    // Set up the Terason Scanner
    if( !InitializeTerason() )
      return 0;

    // Create a glut window
    glutInitWindowPosition(0, 0);
    glutInitWindowSize(windowSizeX, windowSizeY);

    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGB);
    glutCreateWindow("Sonic Flashlight(TM)");

    //load in the transform that calibrates the SF
    LoadCalibrationAffineTransform();
    OpenCalibrationLogs();
```

```
    // Setup the GL parameters
    InitializeGL();
    glutFullScreen();
    // Start the capture-display routine
    glutMainLoop();
  }
  return nRetCode;
}


// atexit callback - cleans up memory allocation prior to the
application exiting
void CleanupApp(void)
{
  // Close the frame receiver
  dlg.m_FrameRx.Close();
  // Stop transmitting images
  dlg.m_TTAutomate.StopTransmitting();
  // Close the TT2000
  dlg.m_TTAutomate.CloseTT2000();
  dlg.DestroyWindow();
  CoUninitialize();

  free(textureData);
  CloseCalibrationLogs();

  std::cout<< "clean exit" <<endl;
}

void DrawCrosshairs()
{

  for (int i = 0; i < 3; i++)
  {
    glBegin(GL_LINES);
      glColor4f(1.0, 0.0, 0.0, 0.5);
      glVertex3f(endingTargetCoords[0][i]  -       m_TargetLength,
endingTargetCoords[1][i], 0.5);
      glVertex3f(endingTargetCoords[0][i]  +       m_TargetLength,
endingTargetCoords[1][i], 0.5);
    glEnd();

    glBegin(GL_LINES);
      glColor4f(1.0, 0.0, 0.0, 0.5);
      glVertex3f(endingTargetCoords[0][i], endingTargetCoords[1][i] -
m_TargetLength, 0.5);
      glVertex3f(endingTargetCoords[0][i], endingTargetCoords[1][i] +
m_TargetLength, 0.5);
    glEnd();

  }
}

// Display the captured data on a quad patch
void DisplayFunction()
{
  glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);

  glColor3d(1.0,1.0,1.0);
   glTexImage2D(GL_TEXTURE_2D, 0, 3, TEXTURE_SIZE_X, TEXTURE_SIZE_Y, 0,
GL_RGB, GL_UNSIGNED_BYTE, textureData);

  // Apply scale and translate in that order
  glMatrixMode(GL_MODELVIEW);

  // Reset matrix prior to applying transforms
  // We want transforms relative to the original image, not
  // any existing transforms
  glLoadIdentity();

  //old code for hand calibrating... leave it in here in case we want
to hand tweak
```

```
  glScaled(scaleX, scaleY, 1.0);
  glTranslated(translateX, translateY, 0.0);

  glRotatef(270,0,0,1);

  if (calibrationStep == -1 && !measurementMode)
  { // the normal operational mode.
    glEnable(GL_TEXTURE_2D);
    glBegin(GL_QUADS);
      // Lower left
      glColor3d(1.0,1.0,1.0);

      glTexCoord2d(0.0, croppedImageSizeY/TEXTURE_SIZE_Y);
                              glVertex3d(currentDisplayCoordinates[0][0],
currentDisplayCoordinates[1][0], 0);

      // Upper left
      glTexCoord2d(0, 0);
                              glVertex3d(currentDisplayCoordinates[0][1],
currentDisplayCoordinates[1][1], 0);

      // Upper right
      glTexCoord2d( croppedImageSizeX/TEXTURE_SIZE_X, 0.0);
                              glVertex3d(currentDisplayCoordinates[0][2],
currentDisplayCoordinates[1][2], 0);

      // Lower right
             glTexCoord2d(croppedImageSizeX/TEXTURE_SIZE_X,
croppedImageSizeY/TEXTURE_SIZE_Y);
                              glVertex3d(currentDisplayCoordinates[0][3],
currentDisplayCoordinates[1][3], 0);
    glEnd();
  }
  else
  {  //calibration mode
    if (calibrationStep == ultrasound0 ||
        calibrationStep == ultrasound1 ||
        calibrationStep == ultrasound2 ||
        measurementMode)
    {
      glEnable(GL_TEXTURE_2D);
      glBegin(GL_QUADS);
        // Lower left
        glColor3d(1.0,1.0,1.0);
        glTexCoord2d(0.0, croppedImageSizeY/TEXTURE_SIZE_Y);
                             glVertex3d(currentDisplayCoordinates[0][0]  +
ultrasoundTargetOffset[0],
                             currentDisplayCoordinates[1][0]             +
ultrasoundTargetOffset[1], 0);
        // Upper left
        glTexCoord2d(0, 0);
                             glVertex3d(currentDisplayCoordinates[0][1]  +
ultrasoundTargetOffset[0],
                             currentDisplayCoordinates[1][1]             +
ultrasoundTargetOffset[1], 0);
        // Upper right
        glTexCoord2d( croppedImageSizeX/TEXTURE_SIZE_X, 0.0);
                             glVertex3d(currentDisplayCoordinates[0][2]  +
ultrasoundTargetOffset[0],
                             currentDisplayCoordinates[1][2]             +
ultrasoundTargetOffset[1], 0);
        // Lower right
                             glTexCoord2d(croppedImageSizeX/TEXTURE_SIZE_X,
croppedImageSizeY/TEXTURE_SIZE_Y);
                             glVertex3d(currentDisplayCoordinates[0][3]  +
ultrasoundTargetOffset[0],
                             currentDisplayCoordinates[1][3]             +
ultrasoundTargetOffset[1], 0);
      glEnd();
    }
    glDisable(GL_TEXTURE_2D);
    if (!measurementMode)  // draw these only if we are calibrating
```

```
      DrawCrosshairs();
  }
  glutSwapBuffers();
}

bool CalculateAffineTransform()
{
  if (det3x3((initTargetCoords)) == 0){
    std::cout<<"determinant is 0"<<std::endl;
    return false;
  }

  double** a = dotProduct3x3and3x3(endingTargetCoords,
                                   inverse3x3(initTargetCoords));

  //multiply the affines together here
  double** b = dotProduct3x3and3x3(a, calibrationAffineTransform);

  for (int i = 0; i < 3; i++){
    for (int j = 0; j < 3; j++){
      if    (b[i][j]    <=    ZERO    &&    b[i][j]    >=    -1*ZERO)
calibrationAffineTransform[i][j] = 0.0;
      else calibrationAffineTransform[i][j] = b[i][j];
    }
  }
  std::cout<<"affine transform"<<std::endl;
  printArray(calibrationAffineTransform, 3, 3);

  return true;
}

void ApplyAffineTransform()
{
  double **    c    =    dotProduct3x3and3x4(calibrationAffineTransform,
initializationDisplayCoords);
  for (int i=0; i<2; i++){
    for (int j=0; j<4; j++){
      currentDisplayCoordinates[i][j] = c[i][j];
    }
  }
}

// Keyboard callback for standard ASCII chars
void ProcessNormalKeys(unsigned char key, int x, int y)
{
  switch(key) {
    case 'w':
      SaveCalibrationAffineTransform();
      break;

    case 'C':
      calibrationStep = crosshair0;
      break;

    case 13: //enter takes you to the next step in calibration
//      if (calibrationStep == -1) break;
      if (calibrationStep == ultrasound0){
        initTargetCoords[0][0] =     endingTargetCoords[0][0] -
ultrasoundTargetOffset[0];
        initTargetCoords[1][0] =     endingTargetCoords[1][0] -
ultrasoundTargetOffset[1];
        ultrasoundTargetOffset[0] = 0;
        ultrasoundTargetOffset[1] = 0;
      }
      if (calibrationStep == ultrasound1){
        initTargetCoords[0][1] =     endingTargetCoords[0][1] -
ultrasoundTargetOffset[0];
        initTargetCoords[1][1] =     endingTargetCoords[1][1] -
ultrasoundTargetOffset[1];
        ultrasoundTargetOffset[0] = 0;
        ultrasoundTargetOffset[1] = 0;
      }
```

```
      if (calibrationStep == ultrasound2){
        initTargetCoords[0][2] =     endingTargetCoords[0][2] -
ultrasoundTargetOffset[0];
        initTargetCoords[1][2] =     endingTargetCoords[1][2] -
ultrasoundTargetOffset[1];
        ultrasoundTargetOffset[0] = 0;
        ultrasoundTargetOffset[1] = 0;
        CalculateAffineTransform();
        ApplyAffineTransform();
      }
      if (calibrationStep != -1) calibrationStep++;
      if  (calibrationStep  ==   NUM_CALIBRATION_STEPS){  //finished
calibration
        calibrationStep = -1;
        LogCalibration();
      }
      if (measurementMode){
        //print out the offset (in mm) of the ultrasound to get the
measurement targets to line up
        double  tempX     =      ultrasoundTargetOffset[0]      *
measurementMultiplier[0];
        double  tempY     =      ultrasoundTargetOffset[1]      *
measurementMultiplier[1];
        std::cout<<"measurement   offsets    (x,y):\t"<<tempX<<",
\t"<<tempY<<std::endl;
        ultrasoundTargetOffset[0] = 0;
        ultrasoundTargetOffset[1] = 0;
        measurementMode = false;
      }
      break;

    case 'R': //reset to the generic uncalibrated state.
      ResetCalibration();
      break;

    case 'm':
      //measures the calibration error
      //it allows the user to move the ultrasound image around.  After
the user presses enter
      //the total distance that the ultrasound image was shifted is
reported to the user.
      measurementMode = true;
      break;

    case 27:   //ESCAPE key.   kicks us out of the calibration or
measurement mode.
      if (calibrationStep != -1) calibrationStep = -1;
      if (measurementMode) {
        measurementMode = false;
        ultrasoundTargetOffset[0] = 0;
        ultrasoundTargetOffset[1] = 0;
      }
      break;

    case 'f': // rotate 180 degrees (flip)
      dlg.m_TTAutomate.InvertRightLeft();
      break;

    case 'a':
      showTTAutomate = !showTTAutomate;
      dlg.m_TTAutomate.ShowWindow(showTTAutomate);
      break;
    case 'r':
       rotation = (rotation + 90) % 360;
      break;
    case 'z': // change translate-scale step size
      if(stStep == 0.01)
        stStep = 0.001;
      else
        stStep = 0.01;
      break;
```

```
    case ' ': // freeze frame

      if (freeze == false){
        dlg.m_TTAutomate.FreezeImage();
        freeze = true;
      }
      else if (freeze = true){
        dlg.m_TTAutomate.ResumeLiveImaging();
        freeze = false;
      }
      break;

  }
}

// Keyboard callback for system commands
void ProcessSpecialKeys(int key, int x, int y)
{
  switch(key) {
    case GLUT_KEY_F5:
      dlg.invertColors();
    break;

    case GLUT_KEY_RIGHT:
      ArrowRight();
    break;

    case GLUT_KEY_LEFT:
      ArrowLeft();
    break;

    case GLUT_KEY_UP:
      ArrowUp();
    break;

    case GLUT_KEY_DOWN:
      ArrowDown();
    break;

    case GLUT_KEY_END:
      exit(1); // Exit the program, but cleanup first (calls C atexit
handler)
    break;
  }
}

void ArrowRight()
{
  if (measurementMode){
    ultrasoundTargetOffset[0] += stStep;
    return;
  }

  switch(calibrationStep)
  {
  case crosshair0:
    endingTargetCoords[0][0] += stStep;
    break;
  case crosshair1:
    endingTargetCoords[0][1] += stStep;
    break;
  case crosshair2:
    endingTargetCoords[0][2] += stStep;
    break;
  case ultrasound0:
  case ultrasound1:
  case ultrasound2:
    ultrasoundTargetOffset[0] += stStep;
    break;
  }
}
```

```
void ArrowLeft()
{
  if (measurementMode){
    ultrasoundTargetOffset[0] -= stStep;
    return;
  }

  switch(calibrationStep)
  {
  case crosshair0:
    endingTargetCoords[0][0] -= stStep;
    break;
  case crosshair1:
    endingTargetCoords[0][1] -= stStep;
    break;
  case crosshair2:
    endingTargetCoords[0][2] -= stStep;
    break;
  case ultrasound0:
  case ultrasound1:
  case ultrasound2:
    ultrasoundTargetOffset[0] -= stStep;
    break;
  }
}

void ArrowUp()
{
  if (measurementMode){
    ultrasoundTargetOffset[1] -= stStep;
    return;
  }

  switch(calibrationStep)
  {
  case crosshair0:
    endingTargetCoords[1][0] -= stStep;
    break;
  case crosshair1:
    endingTargetCoords[1][1] -= stStep;
    break;
  case crosshair2:
    endingTargetCoords[1][2] -= stStep;
    break;
  case ultrasound0:
  case ultrasound1:
  case ultrasound2:
    ultrasoundTargetOffset[1] -= stStep;
    break;
  }
}

void ArrowDown()
{
  if (measurementMode){
    ultrasoundTargetOffset[1] += stStep;
    return;
  }

  switch(calibrationStep)
  {
  case crosshair0:
    endingTargetCoords[1][0] += stStep;
    break;
  case crosshair1:
    endingTargetCoords[1][1] += stStep;
    break;
  case crosshair2:
    endingTargetCoords[1][2] += stStep;
    break;
  case ultrasound0:
  case ultrasound1:
```

```
  case ultrasound2:
    ultrasoundTargetOffset[1] += stStep;
    break;
  }
}

// GL resizing function
void Resize(int width, int height)
{
  windowSizeX = width;
  windowSizeY = height;

  glMatrixMode(GL_PROJECTION);
  glLoadIdentity();

  // Set the width of the new window
  glViewport(0,0,width,height);

  // Recompute the project matrix
  glOrtho(-1, 1, -1, 1, -1, 1);

  // Back to modelview
  glMatrixMode(GL_MODELVIEW);
}


bool InitializeTerason()
{

  // Start up the Terason using the TTAutomate ActiveX control
  if (dlg.m_TTAutomate.OpenTT2000() == FALSE)
  {
    std::cout << "m_TTAutomate.OpenTT2000() failed" <<endl;
    return false;
  }
  else std::cout << "TTAutomate open successful" <<endl;

   dlg.m_TTAutomate.OpenFrameStream(_T("TTSample1"), 3, 1048576, "LIO",
1);

  // Hook up the FrameReceiver ActiveX control to the TTAutomate
  if (dlg.m_FrameRx.Open(_T("TTSample1")) == FALSE)
  {
    std::cout << "Frame Receiver open failed" << endl;
    return false;
  }
  else std::cout << "Frame Receiver open"<<endl;

  // Start Transmitting ultrasound frames
  dlg.m_TTAutomate.ShowWindow(showTTAutomate);
  dlg.m_TTAutomate.StartTransmitting();

  // Must check the frame stream transmitter is open before opening the
receiver
  switch (dlg.m_TTAutomate.GetFrameStreamStatus())
  {
    case 0:
      std::cout << "Frame stream transmitter not open" << endl;
      return false;
    case 1:
      std::cout << "Frame stream transmitter open but not transmitting"
<< endl;
      return false;
    case 2:
      std::cout << "Frame stream transmitter open and transmitting" <<
endl;
  }

  // Hook up the FrameReceiver ActiveX control to the TTAutomate
  if (dlg.m_FrameRx.Open(_T("TTSample1")) == FALSE)
  {
    std::cout << "Frame Receiver open failed" << endl;
```

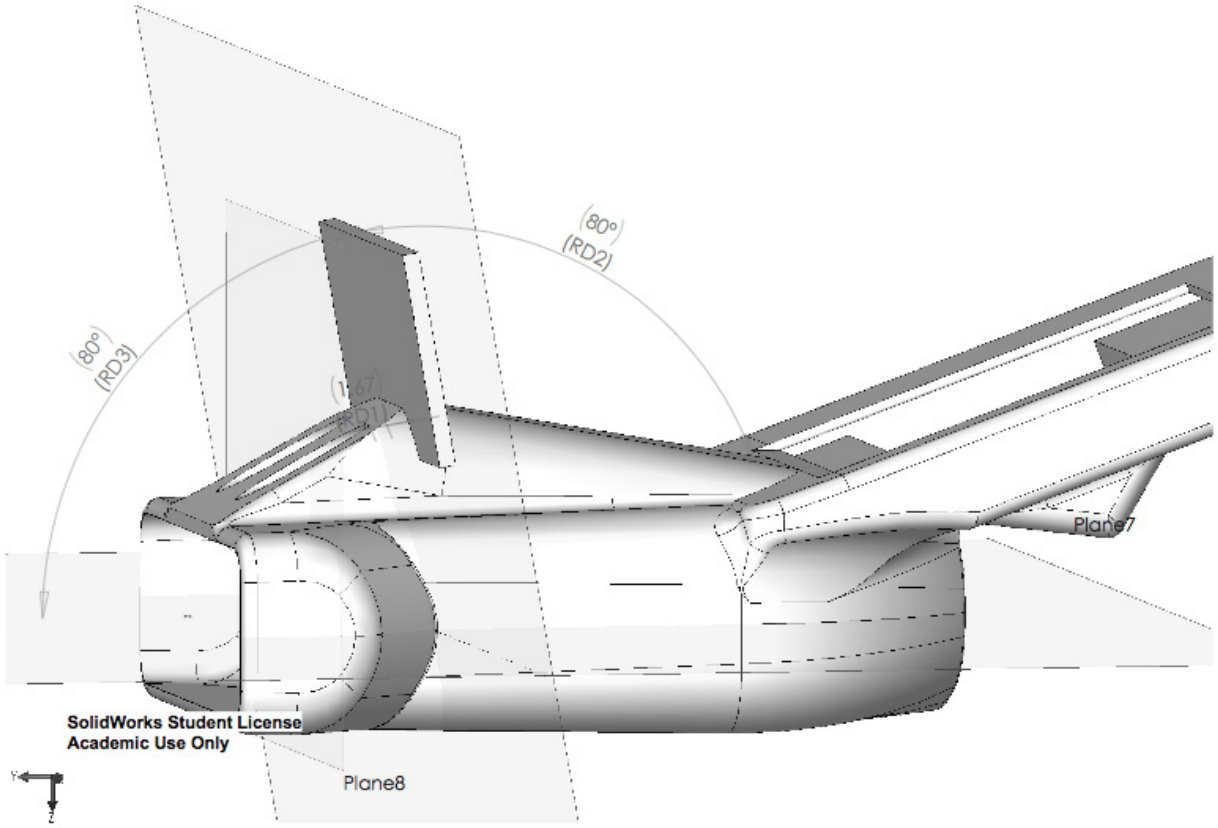
```
    return false;
  }
  else std::cout << "Frame Receiver open"<<endl;

   dlg.m_TTAutomate.SetFocusDepth(3);  //[06,  13,  20,  28,  35,  45,  55,
65]
  dlg.m_TTAutomate.SetDepth(dlg.m_ScanDepth);  //[30 thru 80]

  dlg.m_TTAutomate.ModifyImagingDisplay(0x00000FFF,  0);
  dlg.m_TTAutomate.ModifyImagingDisplay(NEEDLE_GRID,1);
  dlg.m_TTAutomate.LoadPreset("venous");
  dlg.m_TTAutomate.InvertUpDown();
  dlg.m_TTAutomate.InvertRightLeft();
//  dlg.m_TTAutomate.ResizeImage(512,512,0);
  return true;
}

void Redisplay()
{
  if (dlg.m_NewTextureFrameReady)
  {
    glutPostRedisplay();
    dlg.m_NewTextureFrameReady = false;
  }
}


// Set up GL parameters
void InitializeGL()
{
  glutIdleFunc(Redisplay);
  glutReshapeFunc(Resize);
  glutDisplayFunc(DisplayFunction);
  glutKeyboardFunc(ProcessNormalKeys);
  glutSpecialFunc(ProcessSpecialKeys);

  // Set viewing perspective
  glMatrixMode(GL_PROJECTION);
  glOrtho(-1, 1, -1, 1, -1, 1);

  glMatrixMode(GL_MODELVIEW);
  glLoadIdentity();

  // When the texture wants to repeat we clamp the texture,
  // rather than wrapping around
  glTexParameterf(GL_TEXTURE_2D, GL_TEXTURE_WRAP_S, GL_CLAMP);
  glTexParameterf(GL_TEXTURE_2D, GL_TEXTURE_WRAP_T, GL_CLAMP);

  // How do we interpolate between pixels?
  glTexParameterf(GL_TEXTURE_2D, GL_TEXTURE_MAG_FILTER, GL_LINEAR);
  glTexParameterf(GL_TEXTURE_2D, GL_TEXTURE_MIN_FILTER, GL_LINEAR);

  // How are textures applied to surfaces?
  glTexEnvf(GL_TEXTURE_ENV, GL_TEXTURE_ENV_MODE, GL_BLEND);

  // Enable 2D texturing
  glEnable(GL_TEXTURE_2D);
  glPixelStorei(GL_UNPACK_ALIGNMENT, 1);

  glClearColor(0.0, 0.0, 0.0, 1);
  glClearIndex(0);
  glClearDepth(1);
  glLineWidth(3.0);
}


void ResetCalibration()
{
  for (int i=0; i<3; i++)
  {
    for (int j=0; j<4; j++)
      c u r r e n t D i s p l a y C o o r d i n a t e s [ i ] [ j ]          =
```

```
initializationDisplayCoords[i][j];
    for (int k=0; k<3; k++)
      calibrationAffineTransform[i][k] = resetAffineTransform[i][k];
  }
  ApplyAffineTransform();
}


//Load the affine transform calibration from disk
void LoadCalibrationAffineTransform()
{
  std::fstream iStream;
  iStream.open("SFAffineTransform.txt", ios::in);
  if (!iStream.is_open()) {
    std::cout<<"No calibration file found"<<std::endl;
    return;
  }
  iStream >> calibrationAffineTransform[0][0];
  iStream >> calibrationAffineTransform[0][1];
  iStream >> calibrationAffineTransform[0][2];

  iStream >> calibrationAffineTransform[1][0];
  iStream >> calibrationAffineTransform[1][1];
  iStream >> calibrationAffineTransform[1][2];

  iStream >> calibrationAffineTransform[2][0];
  iStream >> calibrationAffineTransform[2][1];
  iStream >> calibrationAffineTransform[2][2];

  iStream.close();

  std::cout<<"transform:"<<endl;
  printArray(calibrationAffineTransform,3,3);

  std::cout<<"coords before "<<currentDisplayCoordinates[0][0]<<endl;
  printArray(currentDisplayCoordinates, 3, 4);

                  dotProduct3x3and3x4(calibrationAffineT
currentDisplayCoordinates, currentDisplayCoordinates);
  std::cout<<"/ncoords after"<<endl;
  printArray(currentDisplayCoordinates, 3, 4);
}


void SaveCalibrationAffineTransform()
{
  std::ofstream oStream;
  oStream.open("SFAffineTransform.txt", ios::out);
  if (!oStream.is_open()) {
    std::cout<<"Unable to open file for writing"<<std::endl;
    return;
  }
  printArray(calibrationAffineTransform,3,3,oStream);
  oStream.close();
  std::cout<<"calibration file written"<<std::endl;
}


void OpenCalibrationLogs()
{
     coordinateLog.open("CalibrationLog/coordinates.log", ios::out |
ios::app);
  if (!coordinateLog.is_open()){
    std::cout<<"Unable to open coordinates.log file for
writing"<<std::endl;
  }
     transformLog.open("CalibrationLog/transforms.log", ios::out |
ios::app);
  if (!transformLog.is_open()){
    std::cout<<"Unable to open transforms.log file for
writing"<<std::endl;
  }
```

```
}

void CloseCalibrationLogs()
{
  //put "-----" into the both the coordinate and transform log files to
indicate that we are
  //finished calibrating.  This is because all calibrations are going
to be logged into these
  //files, and we would like to know which set represents one complete
calibration.
  coordinateLog<<"-----"<<std::endl;
  coordinateLog.close();
  transformLog<<"-----"<<std::endl;
  transformLog.close();
}

void LogCalibration()
{
  //log the coordinates for this iteration
  coordinateLog<<std::endl;
  for (int i=0; i<3; i++)
    coordinateLog<<initTargetCoords[0][i]<<"
"<<initTargetCoords[1][i]<<"  ";
  coordinateLog<<std::endl;
  for (i=0; i<3; i++)
    coordinateLog<<endingTargetCoords[0][i]<<"
"<<endingTargetCoords[1][i]<<"  ";
  coordinateLog<<std::endl;

  //log the overall affine transform to get to these coordinates from
initializationDisplayCoords
  printArray(calibrationAffineTransform,3,3,transformLog);
  transformLog<<std::endl;
}
```

APPENDIX B

SONIC FLASHLIGHT VERSION 5 SCHEMATIC



APPENDIX C

CALIBRATION ERROR RESULTS

Affine Transforms

Affine Transform #	*Configuration Type*	*Affine Transform*		
1	Left-Proximal Small Equilateral	1.53577 0.0154472 0	0.0227642 1.05528 0	0.353254 0.00578455 1
2	Left-Proximal Small Equilateral	1.49679 0.0120017 0	0.0600086 1.1093 0	0.29626 -0.0377047 1
3	Left-Proximal Small Equilateral	1.44957 0.0159071 0	0.0240757 1.1092 0	0.321432 -0.0404708 1
4	Left-Proximal Small Equilateral	1.5269 -0.0051238 0	0.00683177 1.09394 0	-0.359389 -0.0396456 1
5	Left-Proximal Small Equilateral	0.1525 -0.005 0	0.0416667 1.09167 0	0.328 -0.03276 1
6	Right-Proximal Small Equilateral	1.57664 -0.0137095 0	0.00373832 1.17092 0	0.3685 -0.0825962 1
7	Right-Proximal Small Equilateral	1.59672 -0.0474121 0	-0.0072993 1.11681 0	0.374133 -0.0436396 1
8	Right-Proximal Small Equilateral	1.56416 -0.0418015 0	0.018875 1.12012 0	0.358982 -0.0420257 1

9	Right-Proximal Small Equilateral	1.56617 -0.0421714 0	-0.0035891 1.11754 0	0.375787 -0.0445357 1
10	Right-Proximal Small Equilateral	1.53656 0.0602041 0	0.0274174 1.05808 0	0.357974 -0.0149314 1
11	Distal Small Equilateral	1.50909 0.0181818 0	-0.0050909 1.04982 0	0.356505 -0.0475891 1
12	Distal Small Equilateral	1.55556 -0.0015595 0	0 1.08421 0	0.374444 -0.0440019 1
13	Distal Small Equilateral	1.50055 0.0368324 0	0.0309392 1.06262 0	0.375409 -0.042744 1
14	Distal Small Equilateral	1.49982 -0.0007212 0	0.0100974 1.04039 0	0.357418 -0.0653282 1
15	Distal Small Equilateral	1.54614 0.0348837 0	-0.0375094 1.0814 0	0.346545 -0.0289535 1
16	Parallel Isosceles	1.55046 0.0116691 0	0.0550459 1.0845 0	0.365734 -0.0196189 1
17	Parallel Isosceles	1.54558 0.00897293 0	0.0197052 1.10166 0	0.369757 -0.0203315 1
18	Parallel Isosceles	1.54411 0.0135884 0	0.0304232 1.10669 0	0.369258 -0.0218646 1
19	Parallel Isosceles	1.54929 0.0118427 0	0.0465496 1.08575 0	0.366934 -0.0193315 1
		1.55126	0.0857394	0.36366

20	Parallel Isosceles	0.00672122 0	1.08966 0	-0.0230666 1
21	Perpendicular Isosceles	1.53551 0.016184 0	0.0112698 1.0937 0	0.36568 -0.0380664 1
22	Perpendicular Isosceles	1.52743 0.0562525 0	0.00840785 1.09389 0	0.365745 -0.0266842 1
23	Perpendicular Isosceles	1.49081 -0.0033406 0	0.0118206 1.09521 0	0.351783 -0.0402608 1
24	Perpendicular Isosceles	1.46552 0.0172414 0	-0.0003803 1.09394 0	0.352662 -0.0325811 1
25	Perpendicular Isosceles	1.48276 0.0172414 0	0.00164807 1.09394 0	0.355963 -0.0325811 1
26	Large	1.55974 0.00826927 0	0.00455258 1.09057 0	0.376655 -0.0266897 1
27	Large	1.55974 0.00360158 0	0.00451037 1.08196 0	0.37667 -0.0233969 1
28	Large	1.55967 0.00821992 0	0.00944766 1.09401 0	0.372365 -0.0297038 1
29	Large	1.55228 0.00827817 0	0.00687133 1.08719 0	0.376405 -0.0237181 1
30	Large	1.55233 -0.0022351 0	0.00205633 1.09388 0	0.372859 -0.0325324 1

Calibration Error with X-Y Components (mm)

	M1			M2			M3			M4		
Affine Transform #	x	y	dist	x	y	dist	x	y	dist	x	y	dist
1	0.00	0.00	0.00	-1.30	0.42	1.36	-0.87	0.17	0.88	-0.22	0.17	0.27
1	0.00	0.00	0.00	-1.19	0.50	1.29	-0.97	0.17	0.99	-0.22	0.08	0.23
1	0.00	0.00	0.00	-1.19	0.42	1.26	-0.97	0.08	0.98	-0.11	0.17	0.20
1	0.00	0.00	0.00	-1.08	0.50	1.19	-0.87	0.08	0.87	-0.11	0.08	0.14
1	0.00	0.00	0.00	-0.97	0.42	1.06	-0.97	0.17	0.99	-0.22	0.17	0.27
2	0.00	0.00	0.00	0.22	1.33	1.35	0.43	1.33	1.40	-0.11	0.75	0.76
2	0.00	0.00	0.00	0.32	1.25	1.29	0.43	1.41	1.48	-0.11	0.66	0.67
2	0.00	0.00	0.00	0.22	1.16	1.18	0.43	1.33	1.40	0.00	0.75	0.75
2	0.00	0.00	0.00	0.32	1.16	1.21	0.43	1.41	1.48	0.00	0.75	0.75
2	0.00	0.00	0.00	0.22	1.25	1.26	0.43	1.41	1.48	-0.11	0.75	0.76
3	0.00	0.00	0.00	0.32	0.66	0.74	0.43	0.75	0.86	0.00	0.75	0.75
3	0.00	0.00	0.00	0.43	0.66	0.79	0.43	0.75	0.86	0.11	0.66	0.67
3	0.00	0.00	0.00	0.32	0.75	0.82	0.43	0.66	0.79	0.00	0.66	0.66
3	0.00	0.00	0.00	0.32	0.66	0.74	0.43	0.75	0.86	0.00	0.66	0.66
3	0.00	0.00	0.00	0.32	0.75	0.82	0.43	0.75	0.86	0.00	0.75	0.75
4	0.00	0.00	0.00	-0.11	0.25	0.27	0.32	0.00	0.32	0.11	0.25	0.27
4	0.00	0.00	0.00	-0.11	0.25	0.27	0.32	0.00	0.32	0.22	0.25	0.33
4	0.00	0.00	0.00	0.00	0.33	0.33	0.43	0.00	0.43	0.22	0.25	0.33
4	0.00	0.00	0.00	0.00	0.17	0.17	0.32	0.00	0.32	0.32	0.25	0.41
4	0.00	0.00	0.00	-0.11	0.25	0.27	0.43	0.08	0.44	0.32	0.25	0.41
5	0.00	0.00	0.00	0.00	1.00	1.00	0.32	0.66	0.74	0.11	0.42	0.43
5	0.00	0.00	0.00	-0.22	0.83	0.86	0.32	0.66	0.74	0.22	0.42	0.47
5	0.00	0.00	0.00	-0.11	1.00	1.00	0.32	0.75	0.82	0.22	0.33	0.40
5	0.00	0.00	0.00	-0.11	0.91	0.92	0.32	0.66	0.74	0.32	0.33	0.46
5	0.00	0.00	0.00	0.00	1.08	1.08	0.32	0.66	0.74	0.11	0.33	0.35
mean	0.00	0.00	0.00	-0.14	0.72	0.90	0.13	0.59	0.87	0.04	0.44	0.49
6	-0.22	0.50	0.54	1.30	0.42	1.36	1.95	-0.33	1.98	0.00	0.00	0.00
6	-0.32	0.42	0.53	1.41	0.50	1.49	1.95	-0.33	1.98	0.00	0.00	0.00
6	-0.32	0.42	0.53	1.41	0.50	1.49	1.95	-0.33	1.98	0.00	0.00	0.00
6	-0.22	0.42	0.47	1.41	0.50	1.49	1.95	-0.33	1.98	0.00	0.00	0.00
6	-0.22	0.42	0.47	1.30	0.58	1.42	2.06	-0.33	2.08	0.00	0.00	0.00
7	-0.87	0.58	1.04	-0.54	0.58	0.79	0.87	-0.42	0.96	0.00	0.00	0.00
7	-0.76	0.75	1.06	-0.32	0.50	0.59	0.87	-0.50	1.00	0.00	0.00	0.00
7	-0.76	0.66	1.01	-0.43	0.50	0.66	0.76	-0.50	0.91	0.00	0.00	0.00
7	-0.87	0.66	1.09	-0.43	0.58	0.73	0.87	-0.50	1.00	0.00	0.00	0.00
7	-0.76	0.66	1.01	-0.32	0.58	0.67	0.97	-0.58	1.13	0.00	0.00	0.00
8	-0.65	0.33	0.73	-0.32	0.58	0.67	0.76	0.00	0.76	0.00	0.00	0.00
8	-0.65	0.25	0.70	-0.32	0.66	0.74	0.87	0.00	0.87	0.00	0.00	0.00
8	-0.65	0.17	0.67	-0.32	0.58	0.67	0.87	-0.08	0.87	0.00	0.00	0.00
8	-0.76	0.33	0.83	-0.32	0.66	0.74	0.87	-0.08	0.87	0.00	0.00	0.00
8	-0.76	0.42	0.86	-0.22	0.66	0.70	0.97	0.00	0.97	0.00	0.00	0.00

9	-0.54	0.17	0.57	-0.32	0.25	0.41	0.87	-0.50	1.00	0.00	0.00	0.00
9	-0.65	0.33	0.73	-0.32	0.25	0.41	0.87	-0.50	1.00	0.00	0.00	0.00
9	-0.65	0.25	0.70	-0.22	0.25	0.33	0.87	-0.50	1.00	0.00	0.00	0.00
9	-0.54	0.25	0.60	-0.32	0.25	0.41	0.87	-0.50	1.00	0.00	0.00	0.00
9	-0.54	0.25	0.60	-0.32	0.25	0.41	0.87	-0.50	1.00	0.00	0.00	0.00
10	1.08	-0.08	1.09	0.00	0.42	0.42	-0.54	0.08	0.55	0.00	0.00	0.00
10	1.19	-0.08	1.19	0.00	0.33	0.33	-0.54	0.08	0.55	0.00	0.00	0.00
10	1.08	-0.17	1.10	0.00	0.42	0.42	-0.54	0.00	0.54	0.00	0.00	0.00
10	1.19	-0.17	1.20	0.00	0.25	0.25	-0.43	0.17	0.46	0.00	0.00	0.00
10	1.19	-0.17	1.20	0.11	0.33	0.35	-0.54	0.08	0.55	0.00	0.00	0.00
mean	-0.24	0.30	0.82	0.07	0.46	0.72	0.81	-0.26	1.08	0.00	0.00	0.00
11	0.97	-0.08	0.98	-0.11	0.00	0.11	0.22	0.08	0.23	0.97	0.42	1.06
11	1.19	-0.08	1.19	0.11	-0.08	0.14	0.11	0.08	0.14	1.08	0.42	1.16
11	1.19	-0.08	1.19	0.11	-0.08	0.14	0.11	0.00	0.11	1.08	0.42	1.16
11	1.19	0.00	1.19	0.22	0.00	0.22	0.11	0.08	0.14	1.08	0.33	1.13
11	1.30	0.00	1.30	0.00	0.00	0.00	0.11	0.00	0.11	1.19	0.42	1.26
12	0.22	0.08	0.23	-0.02	0.17	0.17	0.32	-0.33	0.46	0.43	0.00	0.43
12	0.32	0.17	0.36	0.00	0.25	0.25	0.43	-0.42	0.60	0.43	0.08	0.44
12	0.32	0.08	0.34	0.00	0.33	0.33	0.32	-0.33	0.46	0.54	0.00	0.54
12	0.32	0.17	0.36	-0.11	0.25	0.27	0.43	-0.33	0.55	0.43	0.08	0.44
12	0.32	0.17	0.36	-0.11	0.25	0.27	0.32	-0.42	0.53	0.54	0.08	0.55
13	1.08	-0.83	1.36	0.32	-0.42	0.53	0.11	-0.17	0.20	0.76	-0.25	0.80
13	1.19	-0.83	1.45	0.22	-0.50	0.54	0.11	-0.17	0.20	0.76	-0.17	0.78
13	1.19	-0.83	1.45	0.32	-0.50	0.59	0.11	-0.08	0.14	0.76	-0.25	0.80
13	1.30	-0.83	1.54	0.22	-0.42	0.47	0.11	-0.17	0.20	0.76	-0.17	0.78
13	1.30	-0.91	1.59	0.11	-0.50	0.51	0.11	-0.17	0.20	0.87	-0.25	0.90
14	1.41	-0.42	1.47	0.00	-0.17	0.17	0.43	0.00	0.43	1.52	0.33	1.55
14	1.52	-0.33	1.55	0.00	-0.08	0.08	0.43	0.00	0.43	1.62	0.25	1.64
14	1.52	-0.42	1.57	0.22	-0.08	0.23	0.43	0.08	0.44	1.62	0.33	1.66
14	1.41	-0.33	1.45	0.11	-0.17	0.20	0.54	0.08	0.55	1.52	0.33	1.55
14	1.62	-0.33	1.66	0.11	-0.08	0.14	0.43	0.00	0.43	1.73	0.33	1.76
15	0.76	0.91	1.19	0.22	0.33	0.40	-0.11	-0.17	0.20	0.22	0.83	0.86
15	0.76	0.75	1.06	0.22	0.42	0.47	0.00	-0.17	0.17	0.22	0.91	0.94
15	0.87	0.83	1.20	0.32	0.33	0.46	0.00	-0.17	0.17	0.32	0.91	0.97
15	0.87	0.83	1.20	0.22	0.33	0.40	0.00	-0.17	0.17	0.22	0.91	0.94
15	0.76	0.83	1.12	0.22	0.42	0.47	0.00	-0.17	0.17	0.32	0.91	0.97
mean	1.00	-0.06	1.14	0.12	0.00	0.30	0.21	-0.12	0.30	0.84	0.29	1.00
16	-0.11	-0.42	0.43	-0.32	0.58	0.67	-0.11	0.17	0.20	0.00	-0.42	0.42
16	0.00	-0.42	0.42	-0.32	0.66	0.74	-0.11	0.17	0.20	0.00	-0.42	0.42
16	0.00	-0.50	0.50	-0.32	0.58	0.67	-0.11	0.17	0.20	0.00	-0.42	0.42
16	-0.11	-0.50	0.51	-0.43	0.66	0.79	-0.22	0.08	0.23	0.11	-0.42	0.43
16	0.00	-0.50	0.50	-0.43	0.66	0.79	-0.11	0.08	0.14	0.11	-0.50	0.51
17	-0.11	-0.17	0.20	-0.32	0.25	0.41	0.11	-0.08	0.14	-0.22	-0.08	0.23
17	0.00	-0.17	0.17	-0.22	0.33	0.40	0.00	-0.08	0.08	-0.22	-0.08	0.23

17	-0.11	-0.17	0.20	-0.32	0.33	0.46	0.11	-0.17	0.20	-0.22	-0.08	0.23
17	-0.22	-0.17	0.27	-0.32	0.33	0.46	0.11	-0.17	0.20	-0.22	-0.08	0.23
17	-0.22	-0.08	0.23	-0.32	0.33	0.46	0.22	-0.17	0.27	-0.22	-0.08	0.23
18	-0.32	-0.25	0.41	0.00	0.33	0.33	0.11	0.00	0.11	-0.22	-0.25	0.33
18	-0.22	-0.17	0.27	0.00	0.42	0.42	0.11	0.00	0.11	-0.22	-0.17	0.27
18	-0.22	-0.33	0.40	0.11	0.33	0.35	0.11	0.00	0.11	-0.22	-0.17	0.27
18	-0.22	-0.25	0.33	0.11	0.33	0.35	0.22	0.00	0.22	-0.22	-0.17	0.27
18	-0.32	-0.25	0.41	0.00	0.33	0.33	0.22	-0.08	0.23	-0.22	-0.17	0.27
19	-0.11	-0.33	0.35	-0.32	0.50	0.59	-0.11	0.17	0.20	0.00	-0.33	0.33
19	0.00	-0.42	0.42	-0.32	0.50	0.59	0.11	0.17	0.20	0.00	-0.33	0.33
19	-0.11	-0.42	0.43	-0.32	0.50	0.59	0.00	0.25	0.25	-0.11	-0.33	0.35
19	0.00	-0.42	0.42	-0.32	0.50	0.59	-0.11	0.25	0.27	0.00	-0.33	0.33
19	0.00	-0.42	0.42	-0.32	0.50	0.59	-0.11	0.17	0.20	0.00	-0.33	0.33
20	-0.11	-0.75	0.76	-0.22	0.83	0.86	0.11	0.58	0.59	0.00	-0.66	0.66
20	0.00	-0.66	0.66	-0.22	0.83	0.86	0.11	0.58	0.59	0.11	-0.66	0.67
20	0.00	-0.75	0.75	-0.11	0.91	0.92	0.11	0.58	0.59	0.00	-0.66	0.66
20	-0.11	-0.75	0.76	-0.11	0.91	0.92	0.00	0.42	0.42	0.00	-0.66	0.66
20	0.00	-0.50	0.50	-0.22	0.83	0.86	0.00	0.50	0.50	0.00	-0.66	0.66
mean	-0.10	-0.39	0.43	-0.23	0.53	0.60	0.03	0.14	0.26	-0.08	-0.34	0.39
21	0.22	0.00	0.22	0.11	0.25	0.27	0.32	-0.08	0.34	0.11	0.08	0.14
21	0.32	-0.08	0.34	0.11	0.08	0.14	0.22	-0.17	0.27	0.22	0.08	0.23
21	0.32	-0.08	0.34	0.32	0.17	0.36	0.22	-0.08	0.23	0.22	0.08	0.23
21	0.32	-0.08	0.34	0.11	0.08	0.14	0.22	-0.17	0.27	0.22	0.08	0.23
21	0.32	-0.17	0.36	0.11	0.17	0.20	0.22	-0.25	0.33	0.32	0.08	0.34
22	0.76	-0.08	0.76	0.54	0.08	0.55	0.00	-0.17	0.17	-0.11	0.17	0.20
22	0.65	-0.08	0.65	0.54	0.00	0.54	0.00	-0.17	0.17	0.00	0.08	0.08
22	0.65	-0.08	0.65	0.65	0.08	0.65	0.00	-0.17	0.17	0.00	0.17	0.17
22	0.76	-0.17	0.78	0.54	0.17	0.57	0.11	0.00	0.11	-0.11	0.08	0.14
22	0.65	-0.17	0.67	0.54	0.08	0.55	0.00	-0.08	0.08	-0.11	0.08	0.14
23	0.11	-0.25	0.27	0.00	0.00	0.00	0.32	0.08	0.34	0.22	0.33	0.40
23	0.11	-0.25	0.27	0.00	0.00	0.00	0.43	0.08	0.44	0.32	0.33	0.46
23	0.11	-0.33	0.35	0.00	-0.17	0.17	0.43	0.17	0.46	0.32	0.33	0.46
23	0.11	-0.33	0.35	0.00	-0.08	0.08	0.43	0.17	0.46	0.32	0.33	0.46
23	0.11	-0.33	0.35	-0.11	0.00	0.11	0.43	0.08	0.44	0.32	0.33	0.46
24	0.43	-0.50	0.66	0.32	-0.25	0.41	0.22	-0.33	0.40	0.11	0.58	0.59
24	0.32	-0.66	0.74	0.22	-0.25	0.33	0.11	0.25	0.27	0.11	0.58	0.59
24	0.32	-0.58	0.67	0.32	-0.25	0.41	0.22	0.25	0.33	0.11	0.50	0.51
24	0.43	-0.58	0.73	0.22	-0.25	0.33	0.22	0.33	0.40	0.11	0.50	0.51
24	0.43	-0.58	0.73	0.22	-0.25	0.33	0.22	0.25	0.33	0.11	0.50	0.51
25	0.32	-0.42	0.53	0.22	-0.17	0.27	0.22	0.17	0.27	0.11	0.42	0.43
25	0.22	-0.42	0.47	0.22	-0.08	0.23	0.22	0.17	0.27	0.22	0.42	0.47
25	0.32	-0.42	0.53	0.22	-0.08	0.23	0.32	0.17	0.36	0.11	0.50	0.51
25	0.43	-0.42	0.60	0.22	-0.08	0.23	0.22	0.17	0.27	0.22	0.42	0.47
25	0.43	-0.42	0.60	0.32	-0.08	0.34	0.22	0.17	0.27	0.22	0.42	0.47
mean	0.37	-0.30	0.52	0.24	-0.03	0.30	0.22	0.03	0.30	0.15	0.30	0.37

26	0.00	-0.08	0.08	-0.22	0.17	0.27	0.11	-0.17	0.20	0.00	-0.08	0.08
26	0.00	0.00	0.00	-0.11	0.25	0.27	0.11	-0.08	0.14	0.00	-0.08	0.08
26	0.11	-0.08	0.14	-0.22	0.17	0.27	0.00	-0.17	0.17	0.00	-0.08	0.08
26	0.00	0.00	0.00	-0.11	0.25	0.27	0.11	-0.08	0.14	0.11	-0.08	0.14
26	0.11	0.00	0.11	-0.11	0.17	0.20	0.11	-0.08	0.14	0.11	-0.17	0.20
27	0.11	0.08	0.14	-0.32	0.17	0.36	0.11	-0.17	0.20	0.00	0.00	0.00
27	0.00	0.00	0.00	-0.11	0.17	0.20	0.11	-0.08	0.14	0.00	0.00	0.00
27	0.00	0.00	0.00	-0.11	0.17	0.20	0.00	-0.17	0.17	0.00	0.00	0.00
27	0.00	0.00	0.00	-0.11	0.08	0.14	0.11	-0.08	0.14	0.00	0.00	0.00
27	0.00	0.00	0.00	-0.22	0.08	0.23	0.11	-0.08	0.14	0.00	0.00	0.00
28	0.00	0.00	0.00	0.00	0.08	0.08	0.00	-0.08	0.08	0.00	0.00	0.00
28	0.00	0.00	0.00	0.00	0.17	0.17	0.00	0.00	0.00	0.00	0.00	0.00
28	0.00	0.00	0.00	0.00	0.17	0.17	0.00	0.00	0.00	0.00	0.00	0.00
28	0.00	0.00	0.00	-0.11	0.17	0.20	0.11	-0.17	0.20	0.00	0.00	0.00
28	0.00	0.00	0.00	0.00	0.17	0.17	0.22	-0.17	0.27	0.00	0.00	0.00
29	0.00	0.00	0.00	-0.22	0.08	0.23	0.00	-0.17	0.17	0.00	0.00	0.00
29	0.00	0.00	0.00	-0.22	0.08	0.23	0.11	-0.08	0.14	0.00	0.00	0.00
29	0.00	0.00	0.00	-0.32	0.25	0.41	0.11	-0.17	0.20	0.00	0.00	0.00
29	0.00	0.00	0.00	-0.11	0.25	0.27	0.11	-0.25	0.27	0.00	0.00	0.00
29	0.00	0.00	0.00	-0.11	0.25	0.27	0.11	-0.25	0.27	0.00	0.00	0.00
30	0.00	0.00	0.00	0.00	0.17	0.17	0.32	-0.17	0.36	0.11	0.08	0.14
30	0.00	0.00	0.00	0.00	0.17	0.17	0.43	-0.17	0.46	0.11	0.08	0.14
30	0.00	0.00	0.00	0.00	0.17	0.17	0.32	-0.08	0.34	0.11	0.00	0.11
30	0.00	0.00	0.00	-0.11	0.17	0.20	0.43	-0.08	0.44	0.11	0.08	0.14
30	0.00	0.00	0.00	-0.11	0.25	0.27	0.32	-0.08	0.34	0.11	0.08	0.14
mean	0.01	0.00	0.02	-0.12	0.17	0.22	0.14	-0.12	0.20	0.03	-0.01	0.05

APPENDIX D

NOVICE LEARNING STUDY RESULTS

CUS subjects (time in seconds)									
Subject #	*2*	*4*	*6*	*8*	*10*	*12*	*14*	*16*	
Task									*Mean*
A	81.39	42.08	70.23	34.31	30.8	23.05	21.75	26.68	41.29
A	16.56	13.4	26.9	17.5	16.46	21.05	12.46	19.34	17.96
A	16.59	9.95	19.23	12.95	19.36	15.9	12.53	12.28	14.85
A	20.37	9.17	15.95	14.08	21.08	14.42	8.95	11.77	14.47
A	18.37	13.67	20.11	14.43	14.92	9.89	9.53	11.58	14.06
A	16.2	12.05	10.71	10.14	14.73	12.14	6.4	11.98	11.79
A	14.53	20.5	10.95	6.64	39.33	9.67	6.83	7.62	14.51
A	17.48	12.81	8.39	9.68	8.18	9.11	6.84	9	10.19
A	24.3	15.27	12.02	7.75	17.67	9.17	8.33	7.8	12.79
A	23.21	15.53	10.23	10.5	13.02	9.21	7.65	5.5	11.86
A	13.2	16.71	9.43	14.17	9.96	14.69	11.5	12.71	12.80
A	7.06	24.98	6.02	9.53	5.9	14.3	6.73	8.48	10.38
A	12.08	12.31	4.87	9.84	4.93	6.81	7.17	8.71	8.34
A	6.59	15.43	5.3	7.37	4.14	9.7	8.17	4.55	7.66
A	8.21	8.05	8.37	7.28	4.73	9.5	8.86	4.14	7.39
A	4.68	5.9	5.02	8.36	4.2	6.61	8.2	7.18	6.27
A	6.27	9.43	4.7	7.09	5.48	6.37	6.83	4.36	6.32
A	6.27	9.11	6.33	6.05	6.77	7.05	6.09	4.98	6.58
A	7.43	9.92	3.81	6.08	3.45	5.75	5.55	4.7	5.84
A	5.39	8.87	4.77	4.93	5.8	7.33	5.25	5.7	6.01
B	48.9	61.61	61.28	41.52	51.83	29.05	25.53	39.78	44.94
B	77.56	25.2	19.48	16.64	58.47	22.73	19.61	17.43	32.14
B	36.43	12.95	19.73	47.67	62.02	21.67	11.21	23.14	29.35
B	21.31	24.03	14.43	23.39	22.34	13.08	10.14	16.02	18.09
B	36.67	48.42	10.58	18.83	60.3	11.25	9.68	12.98	26.09
B	84.7	47.93	17.34	7.96	23.39	12.06	8.84	11.31	26.69
B	24.36	15.15	18	8.84	36.64	15.39	6.23	16.64	17.66
B	27.86	15.12	16.64	8.61	11.65	23.92	7.31	17.92	16.13
B	18.3	9.09	16	7.92	18.45	11.05	7.89	29.45	14.77
B	29.87	17.1	8.65	33.43	8.45	10.9	14.96	20.2	17.95
B	22.73	19.73	13.42	54.52	7.68	15.95	8.89	15.21	19.77
B	11.96	11.17	6.02	35.65	6.61	10.06	21.03	17.02	14.94
B	13.18	13.96	5.89	11.21	5.78	16.61	10.71	40.89	14.78
B	11.9	11.03	5.06	13.14	5.02	8.45	11.39	16.71	10.34
B	12.11	9.27	5.5	6.39	5.33	8.78	13.64	13.7	9.34

B	6.05	10.36	6.83	10.09	6.2	7.28	12.27	6.48	8.20
B	15.93	10.28	5.25	13.21	5.46	6.25	20.5	12.62	11.19
B	6.34	17.71	6.45	7.87	8.18	9.42	8.71	6.37	8.88
B	14.2	10.18	5.77	7.3	4.83	5.05	11.56	6.86	8.22
B	14.15	9.02	6.58	6.43	5.27	6.77	13.55	7.15	8.62
C	18.67	19.83	18.03	17.23	21.05	12.75	14.37	11.5	16.68
C	17.52	9.43	11.83	14.95	6.92	7.73	7.73	7.73	10.48
C	14.27	7.25	13.62	17.73	14.4	7.3	7.89	10.25	11.59
C	13.28	6.27	8.36	6.4	8.12	6.48	10.11	6.52	8.19
C	10.64	11.81	9.03	8.87	10.56	7.56	8.46	6.62	9.19
C	8.78	9.68	6.58	8.77	9.17	6.78	6.15	6.39	7.79
C	7.2	17.95	6.67	5.93	16.68	5.06	6.86	6.58	9.12
C	13.09	9.12	7.05	6.73	8.65	5.67	8.58	6.96	8.23
C	9.8	8.53	9.27	6.4	5.42	5.97	8.96	4.23	7.32
C	5.9	6.06	5.66	8.93	3.52	5	8.23	5.03	6.04
C	5	7.68	4.7	6.5	4.86	8.09	5.55	9.18	6.45
C	4.52	8.56	4.9	6.48	5.95	5.37	3.96	5.65	5.67
C	4.27	7.3	4.31	4.71	5.05	6.02	6.87	6.53	5.63
C	7.23	6.4	4.64	4.56	4.28	5.21	6.83	4.28	5.43
C	6.02	6.18	4.65	5.89	4.73	4.9	5.64	3.48	5.19
C	5.73	5.9	4.77	4.37	3.34	5.02	6.18	3.61	4.87
C	7.37	6.9	4.77	5.3	6.09	5.87	6.33	5.87	6.06
C	6.14	6.5	4.59	4.89	3.81	3.96	5.25	7.78	5.37
C	6.45	7.65	3.59	4.11	3.65	5.05	5.37	3.73	4.95
C	4.8	6.17	6.52	2.71	4.17	3.08	4.92	3.48	4.48

SF subjects (time in seconds)

Subject #	*1*	*3*	*5*	*7*	*9*	*11*	*13*	*15*	
Task									*Mean*
A	43.34	37.45	28.98	47.52	21.84	20.61	30.09	21.23	31.38
A	23.61	24.52	20.48	14.83	12.15	14.09	24.33	12.98	18.37
A	23.89	6.46	13.11	14.17	19.39	11.46	15.67	9.34	14.19
A	16.36	9.84	7.83	9.17	8.92	11.09	15.36	5.93	10.56
A	15.46	4.65	37.34	10	5.78	6.34	15.06	6	12.58
A	11.71	13.58	12.96	10.2	9.77	6.87	10.78	5.93	10.23
A	12.46	5.34	8.37	7.27	5.53	6.62	19.65	4.92	8.77
A	7.56	11.11	15.27	7.53	4.03	5.06	9.8	4.09	8.06
A	8.39	12.8	10.28	4.4	5.17	3.95	7.8	3.78	7.07
A	10.71	7.52	10.42	4.84	3.8	4	9.45	3.7	6.81
A	17.52	11.3	9	10.62	6.2	11.2	10.06	9.33	10.65
A	7.52	11.21	8.08	7.56	4.75	7.59	4.92	4.19	6.98
A	5.92	4.78	4.84	4.33	6.89	4.77	6.31	3.06	5.11
A	5.03	4.45	5.34	4.77	5.4	5.73	5.67	3.3	4.96
A	3.65	3.84	7.39	6.3	6.18	5.53	6.86	4.12	5.48
A	5.45	3.87	8.06	4.97	5.43	4	6.81	3.61	5.28
A	5.5	4.55	5.42	6.09	4.34	4.09	3.18	4.75	4.74
A	5.23	4.15	9.09	4.43	7.06	4.53	5.7	3.27	5.43
A	6.23	3.43	6.06	2.89	5.28	3.83	3.48	2.87	4.26

A	6.36	2.95	7.53	5.64	3.96	4.15	5.37	3.12	4.89
B	78.8	15.15	25.9	22.32	20.28	30.77	24.05	42.8	32.51
B	35.17	8.08	44.32	15.18	11.39	11	19.7	10.33	19.40
B	34.28	8.36	11.27	15.11	11.39	8.56	11.34	8	13.54
B	37.27	8.68	13.48	12.86	12.78	8.39	11.14	7.53	14.02
B	19.58	8.15	11.89	12.58	16.9	6.92	9.8	5.71	11.44
B	40.33	14.05	12.25	2.66	12.12	17.33	6.92	5.18	13.86
B	20.39	8.89	12.71	5.4	6.8	8.91	16.46	4.52	10.51
B	8.93	8.05	15.2	5.14	12.23	6.03	11.93	4.28	8.97
B	24.3	6.55	6.65	5.59	19.73	4.46	6.93	4.37	9.82
B	8.02	8.42	7.14	6.86	7.34	4.28	5.65	3.56	6.41
B	7.58	7.42	4.9	8.25	9.73	6.11	4.17	13.31	7.68
B	14.05	4.23	6.33	4.42	5.15	7.68	5.39	10.89	7.27
B	8.67	3.59	5.46	5.17	5.12	15	6.2	8.05	7.16
B	7.52	5.23	6.11	3.68	12.47	8.75	4.08	6.05	6.74
B	5.84	4.2	6.64	4.12	6.62	3.7	3.95	13.8	6.11
B	7.05	3.78	5.55	9.09	6.7	4.89	4.06	3.43	5.57
B	8.2	3.05	4.45	4.8	3.93	7.83	4.58	6.59	5.43
B	12.13	4.5	6.72	3.05	4.09	4.39	4.5	2.8	5.27
B	8.53	7.36	3.73	3.65	3.37	6.3	3.31	3.25	4.94
B	14.36	3.42	7.14	3.96	3.48	8.56	3.34	3.11	5.92
C	13.68	5.98	16.89	10.56	6.73	8.03	11.84	5.95	9.96
C	18.27	15.27	15.48	7.02	5.14	7.28	10.89	4.7	10.51
C	10.78	10.11	11.9	5.87	4.3	5.92	8.31	3.42	7.58
C	9.08	8.82	6.25	4.46	3.27	5.05	9.28	4.52	6.34
C	12.5	8.34	8.06	3.34	4.55	5.64	7.73	3.75	6.74
C	8.45	12.62	6.75	6.46	4.5	5.27	6.62	3.17	6.73
C	8.23	10.98	5.65	3.75	4.56	3.55	8.73	3.86	6.16
C	4.52	13.45	4.34	2.86	2.81	4.65	5.65	2.4	5.09
C	8.08	19.92	4.65	4.56	3.48	3.4	7.68	2.86	6.83
C	4.7	4.87	3.92	3.21	3.78	3.56	7.03	2.05	4.14
C	18.06	4.62	3.46	3.56	4.33	3.3	5.05	3.56	5.74
C	6.92	4.02	3.42	3.34	3.73	3.18	3.65	2.83	3.89
C	5.82	4.23	1.73	2.48	3.67	2.83	6.55	4.7	4.00
C	3.96	4.36	2.62	2.46	4.55	3.36	3.77	2.93	3.50
C	4.05	3.5	2.93	4.2	4.87	8.58	5.18	2.39	4.46
C	7.09	2.86	3.02	2.4	3.7	3.05	3.28	2.23	3.45
C	5.52	3.59	3.75	3.73	2.87	3.05	4.05	2.21	3.60
C	6.56	3.67	4.17	2.05	3.27	4.08	2.27	2.43	3.56
C	7.27	2.92	3.34	3.37	3.12	3.21	4.36	1.81	3.68
C	3.96	2.37	3.05	2.86	2.7	2.66	2.46	1.78	2.73

APPENDIX E

PROFICIENT CUS USERS STUDY RESULTS

			Subject Number													
Modality	*Trial #*	*Task*	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*14*
CUS	-	*A*	6.39	11.17	5.05	4.61	9.77	5.27	3.90	4.33	6.64	8.77	3.64	10.2	7.92	11.48
CUS	-	*A*	5.59	13.27	9.86	4.18	9.75	5.95	4.33	3.18	4.84	9.64	1.96	13.84	7.75	10.43
CUS	*1*	*A*	5.27	8.89	10.02	3.90	11.3	6.02	3.84	3.64	4.87	10.15	4.53	5.98	8.43	13.55
CUS	*2*	*A*	7.25	9.64	6.83	5.70	5.59	14.62	3.53	2.89	4.17	7.64	8.06	7.59	3.87	8.18
CUS	*3*	*A*	5.92	9.80	4.98	1.90	10.96	4.77	3.80	2.65	6.75	9.83	7.00	5.48	5.39	14.92
CUS	*4*	*A*	2.71	8.86	12.55	7.17	5.3	5.08	2.36	2.42	4.08	9.23	6.84	5.11	8.70	17.59
CUS	*5*	*A*				3.67	12.42	38.5	7.06	3.67	8.42	5.78	4.21	8.36	14.46	7.23
CUS	*6*	*A*				4.53	5.8	8.95	4.83	3.67	4.11	14.25	4.09	8.12	13.3	7.56
CUS	-	*B*	12.06	12.67	9.67	10.37	7.00	15.37	10.98	17.65	4.64	21.89	28.67	25.23	8.05	25.3
CUS	-	*B*	26.39	14.5	8.9	12.31	8.55	6.37	4.37	8.84	5.27	11.98	7.53	14.4	4.78	25.25
CUS	*1*	*B*	9.55	6.84	5.62	5.84	6.28	7.5	4.31	2.42	5.34	6.95	17.21	20.12	4.52	11.37
CUS	*2*	*B*	11.25	37.94	5.34	6.71	7.08	11.06	3.59	6.89	4.68	10.37	13.25	11.92	6.40	40.36
CUS	*3*	*B*	8.83	22.87	7.17	4.92	7.30	17.3	3.59	4.15	4.28	8.12	26.84	9.25	6.06	8.71
CUS	*4*	*B*	5.9	28.37	17.71	5.25	7.45	37.77	3.20	2.50	6.05	18.18	6.13	19.89	21.55	16.87
CUS	*5*	*B*				6.19	6.27	18.27	4.45	2.30	29.08	11.89	30.05	7.62	9.71	9.61
CUS	*6*	*B*				3.75	5.40	4.21	13.33	2.71	5.64	9.73	11.11	7.48	17.53	13.71
CUS	-	*C*	4.98	22.8	5.67	21.37	5.40	12.37	3.56	12.58	2.98	13.55	8.46	26.81	7.73	24.17
CUS	-	*C*	4.05	13.78	5.39	3.31	4.52	3.4	2.17	3.64	5.00	11.17	9.52	10.58	5.37	11.15
CUS	*1*	*C*	4.86	23.78	8.92	5.46	6.33	7.09	2.93	7.90	3.00	17.3	3.89	8.58	3.93	47.75
CUS	*2*	*C*	5.03	9.37	4.42	3.92	4.27	5.45	3.53	3.45	4.75	15.95	22.87	5.56	4.84	16.21
CUS	*3*	*C*	5.46	9.37	9.05	4.83	3.96	31.77	3.55	5.65	3.11	8.03	14.03	8.30	3.81	83.89
CUS	*4*	*C*	6.08	9.98	4.90	3.56	4.70	16.05	2.36	6.12	4.75	6.27	16.00	7.73	4.23	11.17
CUS	*5*	*C*				2.80	7.48	4.96	3.64	2.00	3.08	8.92	9.65	8.17	4.37	11.15
CUS	*6*	*C*				3.12	5.05	5.09	3.64	3.75	3.31	10.37	7.21	5.68	6.00	10.36
SF	-	*A*	9.8	23.34	15.87	5.42	17.37	9.55	4.62	3.59	4.49	3.62	8.95	7.1	5.12	8.4
SF	-	*A*	5.6	13.17	12.12	5.68	12.75	9.59	5.34	4.39	5.89	18.21	5.64	6.15	10.34	10.37
SF	*1*	*A*	4.52	11.5	17.92	7.84	10.4	15.81	6	3.73	4.87	8.23	3.45	6.43	5.2	8.33
SF	*2*	*A*	5.6	9.58	8.42	5.12	9.73	8.52	4.36	4.15	13.9	7.48	3.28	6.28	5.81	8.6
SF	*3*	*A*	5.21	8.68	5.28	6.46	11.17	9.39	4.17	3.08	6.3	7.02	3.27	12.03	6.53	16.62
SF	*4*	*A*	7.28	14.14	6.12	2.89	10.58	5.71	5.28	1.96	6.03	5.65	3.15	3.45	6.37	4.84
SF	*5*	*A*				3.73	8.09	7.43	3.86	3.03	2.63	6.9	2.81	3.89	5.87	8.59
SF	*6*	*A*				3.56	6.96	13.23	4.14	3	3.68	8.96	2.25	4.21	4.86	16.34
SF	-	*B*	8.66	10.17	9.02	5.28	17.56	23.48	8.25	4.84	4.71	18.25	8.11	9.28	9.65	9.48

SF	-	*B*	18.37	14.48	6.53	13.09	13.31	16.34	6.75	5.89	5.34	12.7	5.7	4.98	7.73	15.25
SF	*1*	*B*	11.1	14.7	7.46	3.31	35.55	7.43	6.43	3.18	8.11	5.5	4.52	3.55	4.53	5.95
SF	*2*	*B*	7.68	10.08	7.02	4.62	5.7	16.55	5.62	3.67	5.3	6.7	4.1	5.96	6.08	12.05
SF	*3*	*B*	5.33	8.12	5.65	6.27	8.77	7.33	5.15	2.11	14.12	4.48	4.28	3.89	4.42	5.84
SF	*4*	*B*	9.77	11.11	4.17	4.78	8.75	11.65	3.61	2.31	13.4	5.71	4	4.27	6.7	8.78
SF	*5*	*B*				3.06	6.61	7.53	3.83	3	6.31	5.43	4.06	5.89	4.96	11.84
SF	*6*	*B*				3.64	7	7.68	5.12	3.5	4.86	9.95	3.15	8.8	3.21	8.67
SF	-	*C*	14.02	23.14	9.34	10.8	8.52	12.34	3.71	7.67	3.31	12.25	5.05	12.3	5.4	12.31
SF	-	*C*	6	10.08	5.95	3.5	8.12	10.37	3.73	2.86	3.02	5.39	2.34	3.83	2.87	7.8
SF	*1*	*C*	4.2	16.95	11.42	6.59	7.5	9.59	2.89	3.62	3.98	7.92	8.93	4.9	3.48	7.14
SF	*2*	*C*	23.15	10.12	7.67	3.31	4.83	7.33	3.87	2.92	3.6	6.61	3.75	3.87	3.36	6.17
SF	*3*	*C*	4.27	9.37	4.05	4.28	4.95	37.78	2.4	3.5	2.89	7.21	2.15	7.73	2.85	6.95
SF	*4*	*C*	3.9	8.36	3.9	6.27	5.64	20.28	2.83	2.55	2.4	4.87	2.3	4.7	4.34	18.18
SF	*5*	*C*				3.01	5.09	4.17	2.34	2.1	1.56	7.03	2.77	3.77	3.37	4.87
SF	*6*	*C*				2.75	5.67	6.18	1.89	1.87	5.27	4.36	2.9	2.84	2.58	5.84

Note: Odd numbered subjects (ie 1, 3, 5, etc) used the SF first and CUS second. Even numbered subject used CUS first and the SF second. A trial number of '-' was a practice trial.

Questionnaire Results

	Subject Number													
Task	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*14*
A	6.39	11.17	5.05	4.61	9.77	5.27	3.90	4.33	6.64	8.77	3.64	10.2	7.92	11.48
A	5.59	13.27	9.86	4.18	9.75	5.95	4.33	3.18	4.84	9.64	1.96	13.84	7.75	10.43
A	5.27	8.89	10.02	3.90	11.3	6.02	3.84	3.64	4.87	10.15	4.53	5.98	8.43	13.55
A	7.25	9.64	6.83	5.70	5.59	14.62	3.53	2.89	4.17	7.64	8.06	7.59	3.87	8.18
A	5.92	9.80	4.98	1.90	10.96	4.77	3.80	2.65	6.75	9.83	7.00	5.48	5.39	14.92
A	2.71	8.86	12.55	7.17	5.3	5.08	2.36	2.42	4.08	9.23	6.84	5.11	8.70	17.59

QUESTIONNAIRE

1. The procedure is _____ to perform using the SF rather than CUS

1) much easier
2) somewhat easier
3) about the same
4) somewhat harder
5) much harder

2. The ultrasound image using the SF is _____ interpret than on CUS

1) much easier
2) somewhat easier
3) about the same
4) somewhat harder
5) much harder

3. The mirror physically got in the way of the procedure

1) disagree strongly
2) somewhat disagree
3) neither agree or disagree
4) somewhat agree
5) agree strongly

4. Looking through the mirror made it more difficult to perform the procedure

1) disagree strongly
2) somewhat disagree
3) neither agree or disagree
4) somewhat agree
5) agree strongly

5. Compared to the size of the ultrasound image on the CUS, the ultrasound image on the SF made interpreting the ultrasound

1) much easier
2) somewhat easier
3) neither easier or harder
4) somewhat harder
5) much harder

6. When using the SF, having the ultrasound image appear inside of the phantom _____ in aiming and guiding the needle

1) helped
2) neither helped or hindered
3) hindered

BIBLIOGRAPHY

1. Azuma RT, Bishop G: Improving static and dynamic registration in an optical see-through HMD. Presented at SIGGRAPH, 1994.

2. Bajura M, Neumann U: Dynamic registraion correction in video-based augmented reality systems. IEEE Computer Graphics and Applications 15:52-60, 1995.

3. Blackwell M, Nikou C, DiGioia AM, Kanade T: An image overlay system for medical data visualization. Med Image Anal 4:67-72, 2000.

4. Bold RJ, Winchester DJ, Madary AR, Gregurich MA, Mansfield PF: Prospective, randomized trial of Doppler-assisted subclavian vein catheterization. Arch Surg 133:1089-1093, 1998.

5. Branger B, Zabadani B, Vecina F, Juan JM, Dauzat M: Continuous guidance for venous punctures using a new pulsed Doppler probe: efficiency, safety. Nephrologie 15:137-140, 1994.

6. Brass P, Volk O, Leben J, Schregel W: Central venous cannulation - always with ultrasound support? Anasthesiol Intensivmed Notfallmed Schmerzther 36:619-627, 2001.

7. Chang WM, Stetten GD, Lobes Jr. LL, Shelton DM, Tamburo RJ: Guidance of retrobulbar injection with real time tomographic reflection. J Ultrasound Med 21:1131-1135, 2002.

8. DiGioia AM, Simon D, Jaramaz B, Blackwell M, Morgan FM, O'Toole RV, Colgan B, Kischell E: HipNav: Pre-operative Planning and Intra-operative Navigational Guidance for Acetabular Implant Placement in Total Hip Replacement Surgery. Presented at Computer Assisted Orthopaedic Surgery Symposium, November, 1995.

9. Fuchs H, State A, Pisano ED, Garrett WF, Hirota G, Livingston MA, Whitton MC, Pizer SM: Towards Performing Ultrasound-Guided Needle Biopsies from within a Head-Mounted Display. Presented at Visualization in Biomedical Computing, Hamburg, Germany, September 22-25, 1996, 1996.

10. Gann M Jr, Sardi A: Improved results using ultrasound guidance for central venous access. Am Surg 69:1104-1107, 2003.

11. Gilbert TB, Seneff MG, Becker RB: Facilitation of internal jugular venous cannulation using an audio-guided Doppler ultrasound vascular access device: results from a prospective, dual-center, randomized, crossover clinical study. Crit Care Med 23:60-65, 1995.

12. Gratz I, Afshar M, Kidwell P, Weiman DS, Shariff HM: Doppler-guided cannulation of the internal jugular vein: a prospective, randomized trial. J Clin Monit 10:185-188, 1994.

13. Gualtieri E, Deppe SA, Sipperly ME, Thompson DR: Subclavian venous catheterization: Greater success rate for less experienced operators using ultrasound guidance. Crit Care Med 23:692, 1995.

14. Han D, Lan Seo Y, Soon Choi C, Chul Kim H, Young Yoon D, Hoon Bae S, Hee Moon J, Hyup Kim S, Soo Kim S, Han H: A steerable guiding device: the new method in ultrasound guidance. Invest Radiol 37:626-631, 2002.

15. Hatfield A, Bodenham A: Portable ultrasound for difficult central venous access. Br J Anaesth 82:822-826, 1999.

16. Hind D, Calvert N, McWilliams R, Davidson A, Paisley S, Beverley C, Thomas S: Ultrasonic locating devices for central venous cannulation: meta-analysis. BMJ 327:361, 2003.

17. Janin AL, DW M, TP C: Calibration of head-mounted displays for augmented reality applications. Presented at IEEE Virtual Reality Annual International Symposium, 1993.

18. Keenan SP: Use of Ultrasound to Place Central Lines. Journal of Critical Care 17:126-137, 2002.

19. Lindgren PG: Ultrasonically guided punctures. A modified technique. Radiology 137:235-237, 1980.

20. Lindgren PG, Hemmingsson A: Percutaneous nephropyelostomy. A new technique. Acta Radiol Diagn (Stockh) 21:759-761, 1980.

21. MacIntyre PA, Samra G, Hatch DJ: Preliminary experience with the Doppler ultrasound guided vascular access needle in paediatric patients. Paediatr Anaesth 10:361-365, 2000.

22. Masamune K, Fichtinger G, Deguet A, Matsuka D, Taylor R: An Image Overlay System with Enhanced Reality for Percutaneous Therapy Performed Inside CT Scanner. Presented at Medical Image Computing and Computer-Assisted Intervention - MICCAI 2002, Tokyo, Sept 25-28, 2002.

23. Masamune K, Masutani Y, Nakajima S, Sakuma I, Dohi T, Iseki H, Takakura K: Three-dimensional Slice Image Overlay System with Accurate Depth Perception for Surgery. Presented at Medical Image Computing and Computer-Assisted Intervention, Pittsburgh, 2000.

24. Miller AH, Roth BA, Mills TJ, Woody JR, Longmoor CE, Foster B: Ultrasound Guidance versus the Landmark Technique for the Placement of Central Venous Catheters in the Emergency Department. Academic Emergency Medicine 9:800-805, 2002.

25. Oishi T, Tachi S: Calibraation method of visual parameters for see-through head-mounted display. Presented at IEEE International Conference on Multisensor Fusion and Integration For Intelligent Systems, 1994.

26. Parkinson R, Gandhi M, Harper J, Archibald C: Establishing an ultrasound guided peripherally inserted central catheter (PICC) insertion service. Clin Radiol 53:33-36, 1998.

27. Polak JF, Anderson D, Hagspiel K, Mungovan J: Peripherally inserted central venous catheters: factors affecting patient satisfaction. Am J Roentgenol 170:1609-1611, 1998.

28. Reich A, Booke M: Does internal jugular vein cannulation in infants require a SMART needle rather than a smart anaesthetist? Paediatr Anaesth 12:287, 2002.

29. Reid MH: Real-time sonographic needle biopsy guide. Am J Roentgenol 140:162-163, 1983.

30. Rickey D, Picot P, Christopher D, Fenster A: A wall-less vessel phantom for Doppler ultrasound studies. Ultrasound Med biol 21:1163-1175, 1995.

31. Rosenthal M, State A, Lee J, Hirota G, Ackerman J, Keller K, Pisano E, Jiroutek M, Muller K, Fuchs H: Augmented reality guidance for needle biopsies: an initial randomized, controlled trial in phantoms. Med Image Anal 6:313-320, 2002.

32. Sauer F, Khamene A, Bascle B, Schimmang L, Wenzel F, Vogt S: Augmented reality visualization of ultrasound images: System description, calibration, and features. Presented at 4th International Symposium on Augmented Reality - ISAR, New York, NY, October, 2001.

33. Shelton DM, Klatzky RL, Stetten GD: Method for assessing augmented reality needle guidance using a virtual biopsy task. Presented at IEEE International Symposium on Biomedical Imaging, Arlington, VA, April, 2004.

34. Sofocleous C, Schur I, Cooper S, Quintas J, Brody L, Shelin R: Sonographically guided placement of peripherally inserted central venous catheters: review of 355 procedures. AJR Am J Roentgenol 170:1613-1616, 1998.

35. State A, Livingston MA, Garrett WF, Hirota G, Whitton MC, Pisano ED, Fuchs H: Technologies for Augmented Reality Systems: Realizing Ultrasound-Guided Needle Biopsies. Presented at SIGGRAPH, New Orleans, LA, 1996.

36. Stetten GD: System and Method for Location-Merging of Real-Time Tomographic Slice Images with Human Vision, in. USA, 2003.

37. Stetten GD, Chib V: Overlaying Ultrasound Images on Direct Vision. J Ultrasound Med 20:235-240, 2001.

38. Stetten GD, Chib V, Hildebrand D, Bursee J: Real time tomographic reflection: Phantoms for calibration and biopsy. Presented at Second IEEE and ACM International Symposium on Augmented Reality, Columbia University, NYC, 2001.

39. Stetten GD, Cois A, Chang WM, Shelton DM, Tamburo RJ, Castelucci J, von Ramm O: C-mode Virtual Image Display for a Matrix Array Ultrasound Sonic Flashlight. Presented at Medical Image Computing and Computer Assisted Intervention – MICCAI, 2003.

40. Teichgraber UK, Benter T, Gebel M, Manns MP: A sonographically guided technique for central venous access. Am J Roentgenol 169:731-733, 1997.

41. Troianos CA, Jobes DR, Ellison N: Ultrasound-guided cannulation of the internal jugular vein. A prospective, randomized study. Anesth Analg 72:823, 1991.

42. Tseng M, Sadler D, Wong J, Teague K, Schemmer D, Saliken J, So B, Gray R: Radiologic placement of central venous catheters: rates of success and immediate complications in 3412 cases. Canadian Association of Radiologists Journal 52:379-384, 2001.

43. Vucevic M, Tehan B, Gamlin F, Berridge JC, Boylan M: The SMART needle. A new Doppler ultrasound-guided vascular access needle. Anaesthesia 49:889-891, 1994.

44. Yeh HC, Mitty HA, Wolf BS: A simple ultrasound guide for needle puncture. J Clin Ultrasound 4:53-54, 1976.

45. Yonei A, Nonoue T, Sari A: Real-time ultrasonic guidance for percutaneous puncture of the internal jugular vein. Anesthesiology 64:830-831, 1986.