Development of Novel Haptic Tools for Virtual Navigation

by

Diana Sim

B.S., Engineering as recommended by the Department of Mechanical Engineering
Massachusetts Institute of Technology, 2007

Submitted to the Department of Mechanical Engineering in
Partial Fulfillment of the Requirements for the Degree of

Master of Science in Mechanical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2013

Signature of Author: ______________________________
Department of Mechanical Engineering
June 1, 2013

Certified by: ______________________________
Dr. Mandayam A. Srinivasan
Senior Research Scientist, Mechanical Engineering
Thesis Supervisor

Accepted by: ______________________________
David E. Hardt
Professor of Mechanical Engineering
Chairman, Department Committee for Graduate Students

(This page intentionally left blank)

Development of Novel Haptic Tools for Virtual Navigation

by

Diana Sim

Submitted to the Department of Mechanical Engineering on
June 1, 2013 in Partial Fulfillment of the Requirements for the Degree of
Master of Science in Mechanical Engineering

ABSTRACT

Learning how to navigate a real space using haptic virtual environments can be challenging. One major issue is the inefficient rate of exploration due to the single point interface of haptic devices. The use of haptic fields such as repulsive and attractive force fields was studied to determine their ability to enable global sensing to improve haptic navigation.

Repulsive Force Fields for Global Haptic Sensing

Repulsive force fields were designed to help users understand their environments more quickly through global sensing. Two experiments were conducted using repulsive force fields to provide information about indoor and outdoor spaces. In both experiments, repulsive force fields were found to be usable but not more effective in teaching the user about the environment than no force field.

Attractive Force Fields for Global Haptic Sensing Applied to Route Guidance

Attractive force fields were studied in the context of providing route guidance. Several haptic guides were designed and evaluated in a developmental experiment. The most promising haptic guide was selected and compared to conventional alternatives (using no guide and an audio guide) in a main experiment to determine its ability to effectively aid route learning. The haptic guide fared poorly in the initial main experiment and was re-designed. Following this, a final main experiment was conducted with ten subjects. The results suggest that the haptic guide is, in fact, an effective tool for route learning.

(This page intentionally left blank)

TABLE OF CONTENTS

1 INTRODUCTION

The over-encompassing goal of our research and BlindAid is to give blind people tools to build cognitive maps of unknown spaces to support independent travel. BlindAid is a haptic-aural virtual environment (VE) system that enables blind people to learn about new environments through touch and sound. Haptics refers to the study of sensing and manipulation through touch. BlindAid uses a state-of-the-art haptic device called the Phantom (Sensable). The Phantom is a desktop device that allows for virtual exploration through a stylus; this mode of exploration can be compared to exploring the world with the tip of a miniature white cane. Unfortunately, learning spaces through a single-point interface can be challenging. The goal of this thesis was to develop tools to improve haptic exploration of virtual spaces. The development process included proposing various concepts, using design criteria to narrow down the ideas, building program code for the most promising concepts using C++, and performing experiments with human subjects to test usability.

1.1 MOTIVATION FOR BLINDAID

The ability to travel freely is a critical component of personal independence (Passini, Dupre, & Langlois, 1986). This ability is significantly compromised without vision. However, tools and strategies have been developed over the years to compensate and provide spatial information through other means. Some commonly used tools for obstacle detection and navigation include white canes, dog guides, and GPS (Global Positioning System) devices. However, these tools are intended for in-situ travel, or during motion. Navigation has two major components: in-situ travel and planning. Planning which includes building cognitive maps and learning routes, is especially important without sight. Up until recently, written and verbal descriptions (Golledge, Klatsky, & Loomis, 1996) and tactile maps which are difficult to come by, were the only planning tools available to blind people (Wang, Li, Hedgpeth, & Haven, 2009). However, haptics, a relatively new area of research has introduced a new solution, haptic virtual environment systems such as BlindAid. Studies have shown that spatial information from a virtual environment is transferable to a real environment

(Klatzky, Loomis, Beall, Chance, & Golledge, 1998; Lahav, Schloerb, Kumar, & Srinivasan, 2008; Péruch, Vercher, & Gauthier, 1995; Picard & Pry, 2009; Schmelter, Jansen-Osmann, & Heil, 2008).

1.2 MOTIVATION

Learning about an environment using a haptic map implemented by BlindAid can be more difficult than learning about a space using a visual map, because the eye can process spatial data at a much faster rate than the finger. While a sighted person can often look at a map and within seconds, understand the information being conveyed; a blind person using a single-point haptic device, might need minutes to learn the same space. This highlights the fact that vision is a global sense, allowing a person to learn in parallel, while state-of-the-art haptics is serial. Hence, there is a need to improve BlindAid to educate users about spaces more efficiently by making haptic learning more global.

1.3 OVERVIEW

This thesis focuses on making haptic sensing with the Phantom more global to improve BlindAid. Like vision, audio is a global sense, and while we could have explored combinations of haptics and audio to improve BlindAid, we chose to focus on haptics to avoid overloading the audio channels. Although overloading audio in a virtual system is very tempting, prior work on audio VEs suggest that conveying too much information through audio can be detrimental to learning. Furthermore, haptics is the primary mode in which users interact with the VE.

Global sensing for haptics can be implemented by creating haptic fields around objects in the environment so that the Phantom can detect them at a distance. The idea of using haptic fields to enable global haptic sensing grew out of a preliminary experiment. In that experiment, the subject attempted to use BlindAid to learn about a VE representing a 3D cityscape with streets and buildings. The subject suggested that it might be helpful if he could feel the locations of the buildings as he passed them while following the street. There are a number of ways to create haptic fields using repulsive forces, attractive forces, and alternating forces. Repulsive and attractive forces can vary in different ways with respect to distance – linearly and non-linearly. Alternating forces can be dynamic, varying in time to produce a buzzing effect, or static, varying in space to create a texture which can differ in amplitude, period, and shape of waveform.

Repulsive Force Fields for Global Haptic Sensing

We decided to begin with a repulsive force field. This was a simple idea that we believed could be helpful and had received little attention from the research community. The goal was to investigate using repulsive force fields to enable global sensing to help users learn about the environment more quickly. We experimented with using repulsive force fields to provide information about indoor and outdoor spaces. Preliminary tests indicated that repulsive force fields, while not a hindrance to exploration, were not more effective in teaching the user about the environment than no force field.

Attractive Force Fields for Global Haptic Sensing Applied to Route Guidance

We then decided to explore attractive forces. However, during our research, we came up with the idea to apply attractive forces to the problem of developing a haptic guide. There is a fundamental issue with using haptics to teach routes. Simply leading a person by hand (passive exploration) does not facilitate proper route learning (Lécuyer et al., 2003; Farrell et al., 2003). Thus, we focused on using an attractive force field for global sensing, allowing users to simultaneously sense the route, while exploring surrounding objects, and applied this to developing a haptic guide. In order to facilitate active exploration, we designed a guide that tethered users to a route to encourage navigation in the correct direction, while still allowing for the exploration of the neighboring areas. We tested a few different renditions of the haptic guide to determine whether haptics could be used for route learning. The final concept for the guide provided directions (on command) by nudging the user in the correct direction and used textures to indicate off-route areas. Ultimately, the results suggested that a haptic guide could be an effective tool for route learning.

2 BACKGROUND

2.1 NEEDS OF PEOPLE WHO ARE BLIND

While sighted people use their eyes to gather most of the information needed to navigate, people with visual impairments are faced with significant difficulties when exploring new spaces. Passini, Dupre, & Langlois (1986) postulated that the ability to travel freely is critical to one's personal independence and integration into society. Hence, research on blindness has focused considerable energy on finding methods to improve independent navigation (Espinosa & Ochaita, 1998).

Blindness, "defined as visual acuity of less than 20/400 (6/120), or corresponding visual field loss to less than 10 degrees," ("Blindness", n.d., para. 3) and a lack of mobility have been found to have a negative impact on different aspects of health. With regards to physical health, blindness is not only associated with lower levels of fitness in children, but also delays in the development of motor skills (O'Connell, Lieberman, & Petersen, 2006). One study concluded that "individuals who are visually impaired have an increased risk of chronic health problems and difficulty with functional mobility tasks that require strength and speed" (Ray, Horvat, Williams, & Blasch, 2007, p. 112). Furthermore, a lack of independence and social isolation from late-onset blindness may be associated with depression (O'Donnell, 2005). These studies stress the importance of personal independence, mobility, and their impact on quality of life.

2.2 TOOLS FOR BLIND NAVIGATION

The most commonly used orientation and mobility tool is the white cane. A smaller percentage of people use guide dogs and GPS devices. There are approximately 130,000 white cane users (Russell, Hendershot, LeClere, Howie, & Adler, 1997) and 7,000 dog guide users in the United States (Journal of Visual Impairment & Blindness, 1995). Research and development of mobility tools for people who are blind typically fall into one of two categories: obstacle detection and navigation.

2.2.1 OBSTACLE DETECTION

Obstacle detection tools include the white cane, haptic or audio-enabled "augmented" white canes, guide dogs, and portable obstacle detectors ("Blindness", n.d.). Handheld obstacle detectors typically consist of acoustic or optic sensors that detect objects and convey information to the user through touch (e.g., vibration) or sound. Similar technologies have been adapted for the white cane; "augmented" white canes are designed to detect obstacles and drops offs in the environment, and communicate through audio signals, vibrations, and/or physical forces (Borenstein, n.d.; Gallo et al., 2010; Julius, 2010; Yu, Yoon & Jeong, 2009). Mobile phones, wearable devices and robots have also been adapted to function as obstacle detectors (Abdulrasool & Sabra, 2011; Akhter, Mirsalahuddin, Marquina, Islam, & Sareen, 2011, Pradeep, Medioni, & Weiland, 2010, Shoval, Ulrich, & Borenstein, 2003).

2.2.2 NAVIGATION

Navigational tools can be divided into two categories: in-situ tools (used during travel) and planning tools. Electronic mobility aids (EMA) like GPS devices fall under the in-situ tools category; virtual haptic systems like BlindAid and tactile maps are considered planning tools, although tactile maps can also be mobile.

In-situ Tools

Electronic mobility aids can provide information about the user's current position and instructions on how to get to a target destination (Havik, Steyvers, van der Velde, Pinkster, & Kooijman, 2010). Some of the benefits of using an EMA include "improved wayfinding performance; the detection of obstacles, objects, landmarks, and travel path alignment; and feelings of safer, more comfortable, and less stressful travel accompanied by a higher quality and increased frequency of travel" (Roentgen, Gelderblom, Soede, & de Witte, 2009, p. 751). However, EMAs often result in slower than desired travel speeds and suffer from high rates of discontinued use (Roentgen et al., 2009).

As with obstacle detection, mobile devices such as PDAs and cell phones, and wearable devices such as vests and belts have been adapted for blind navigation (Loomis, Marston, Golledge & Klatsky, 2005; Ross & Blasch, 2000; Heuten, Henze, Boll, & Pielot, 2008; Sánchez, 2009; Sáenz and Sánchez, 2010). Another type of in-situ navigation aid modifies the environment to include location identifiers. Systems have used infrared transmitters, RFID (radio-frequency identification) tags, and audio beacons in public spaces to provide information about locations or objects (Blenkhorn & Evans, 1997; D'Atri et al., 2007; Landau, Wiener, Naghshineh, & Giusti, 2005; Loomis, Golledge, Klatzky, & Marston, 2007; Na, 2006; Shiizu, Hirahara, Yanashima, & Magatani, 2007).

Planning Tools

The traditional alternative to planning aids now being developed is to "make use of sighted assistance to describe an environment prior to visiting it, and thereby memorize a mental model to assist them when they are there on their own" (White, Fitzpatrick, & McAllister, 2008, p. 5). Planning navigation aids supply the information a person needs to create such a mental model, making them valuable for independent navigation. Tactile maps are among the most basic planning aids. Multiple studies have demonstrated the effectiveness of tactile maps for navigation (Papadopoulos, 2004; Picard & Pry, 2009). Ungar (2000) reported two important benefits of tactile maps: in the short-term, being introduced to a space and in the long-term, improving the user's "abstract level spatial thought [by providing experience with] relating a map to the environment it represents" (p. 10). However, there are limitations; tactile maps are not widely available due to production costs, and struggle with information density – the level of detail of a given tactile map is fixed.

Virtual environment (VE) systems have demonstrated potential as planning tools. VEs, also known as "virtual reality," are computer-simulated environments that can model real world spaces. A major advantage of virtual environment systems, especially in comparison to tactile maps, is the ease with which maps can be reproduced and transmitted. Virtual maps can also display different amounts of detail and information depending on zoom levels, and can relay this information dynamically. VEs are interactive: users can "explore environment[s] actively and control what they experience" (Schmelter, Jansen-Osmann, & Heil, 2009, p. 4). In addition to serving as a general aid for experienced visually impaired travelers, haptic VE technology may be used to support an Orientation and Mobility (O&M) curriculum. Haptic technologies such as BlindAid can help visually impaired students learn about new spaces without the aid of an instructor. This type of tool can be very valuable since instructor time is often a limiting factor. Additionally, users can interact and collect spatial information without constraints on exploration time and space, and physical effort needed. Virtual environments can also be used for O&M diagnostic tools since they can provide efficient ways to monitor and record behavioral responses and navigation strategies of users.

A number of researchers have studied the efficacy of audio-only feedback in a virtual environment. Such systems have utilized non-speech and speech sounds to educate the user about map objects and routes (Giudice, Bakdash, Legge, & Roy, 2010; Heuten, Henze, & Boll, 2007; Sánchez, Tadres, Pascual-Leone, & Merabet, 2009; Seki & Sato, 2011). However, the audio channels can quickly become overloaded and audio alone may not be as effective as audio and haptics together; thus systems that have both haptic and audio capabilities have received more attention.

2.3 HAPTIC-AURAL VIRTUAL NAVIGATION PLANNING SYSTEMS

Haptic-aural virtual navigation systems allow users to explore a virtual space that may model a real space, through sound and haptic stimuli, using a device like the Phantom. Researchers have conducted feasibility studies of haptic-aural navigation systems with favorable results. One such system, developed by Feintuch, Haj, & Weiss (2006), utilizes an off-the-shelf haptic joystick and provided information about objects in 2D space via vibrations and sounds. Initial tests with blind children found that subjects were able to translate map information gathered virtually to the navigation of real spaces. Kostopoulos, Moustakas, Tzovaras, & Nikolakis (2007) designed a method for map image analysis that could create virtual haptic-aural maps from existing map data. The system deciphered street names and conveyed location information relative to streets and intersections through audio. Kaklanis, Votis, Moschonas, & Tzovaras (2011) took virtual mapping one step further and created HaptiRiaMaps. HaptiRiaMaps is a free web-based map application that allows users to build virtual haptic-aural maps from OpenStreetMaps. In both studies, preliminary tests with blind subjects were promising.

A couple of systems have attempted to work in three-dimension. The HOMERE system, developed by Lécuyer et al. (2003), is comprised of a virtual white cane for haptic interactions, a virtual sun to sense cardinal direction, and spatialized ambient and event-related environmental sounds. Hara et al. (2010) explored the use of life-size virtual environments. The system consists of a white cane that can interact with virtual components and relay information audibly and tactually, an optical tracking system, and a computer to create the virtual environments. While both systems were relatively well-received, the space requirements for such devices significantly limit usability. These prior studies are important for validating the use of virtual haptic-aural maps for blind navigation, and providing insight into useful system features (many of these features are described below).

While we were unable to find any prior work related to the use of force fields to enable global haptic sensing, a few studies did explore the use of haptic guide agents. Of note are two systems, HOMERE and the Haptic Walk-Guide simulator (HAWG). Both systems use active guide agents which force the user to become passive in route learning. HOMERE's guide agent moves the user at a constant speed through the route from start to finish, while HAWG uses a teacher-student feature. This feature enables one haptic device to be passively led by another within the same virtual environment. The primary result of these two studies and a study on route learning via virtual environments by Farrell et al. (2003) suggests that the mode in which the space is learned is important. HOMERE users expressed a clear preference for navigation that was active and variable in route and speed, rather than passive. Farrell et al. found that active exploration resulted in more accurate cognitive maps than passive exploration. Preliminary studies illustrate some of the challenges faced by guide agents but have yet to present a good solution.

2.4 BLINDAID

BlindAid is comprised of many of the most promising tools explored, as well as a number of new ideas that have not been explored. The system allows for exploration of a virtual 3D space with spatialized sounds and contains virtual objects with varying haptic properties and surface textures (Schloerb, Lahav, Desloge, & Srinivasan, 2010). BlindAid uses headphones to convey audio and the Sensable Phantom for haptics. Users can explore the virtual space by manipulating the Phantom's stylus; the stylus maps to a virtual object that represents the user's position in the VE called the proxy. It also incorporates other features such as haptic zooming, restarting (returns the proxy to the start position) and recording of traveled paths for instructor analysis or research purposes. The use of sounds, haptic sensations, and zooming is important in helping users learn about virtual spaces quickly. Adding tools for global sensing and route guidance may provide even more assistance in this endeavor.

Features of BlindAid

Some of the main features that have been developed and included in the BlindAid system include:

Audio

- *Spatialized sound* gives information that allows the user to directly perceive the distance and direction of sound sources in the VE. Note, directional information is effectively limited to azimuth with the current system and it is difficult for the user to differentiate between sounds ahead or behind.
- *Contact sounds* provide information about the type of object touched by using different identifiable short sounds (brief so as not to slow exploration).
- *Identification sounds* give more detail about an object than a contact sound. Identification sounds, which are also called long sounds, are typically verbal descriptions that can be played on command when touching an object.
- *Background sounds* are similar to ambient noise and can give information about a location and the boundaries of that location.
- *Landmarks* are useful in route planning as they can serve as audio beacons. Landmarks are important in blind navigation so the ability to create and access landmarks in a virtual environment is essential. The audio files associated with landmarks are played using spatialized sound. This means the user hears the sound as if he/she is facing forward in the VE and the sound source is at the relative distance and direction specified.

Haptics

BlindAid uses a number of different haptic objects that can vary in color and sound. Some objects such as walls, public doors, ground textures, and rectangle objects can vary in haptic properties and texture as well. Haptic properties include stiffness, damping, static friction, and dynamic friction as described in the OpenHaptics Toolkit API Reference Manual (Sensable Technologies, 2008). The OpenHaptics API is a C++ library that gives access to and control of the Phantom. In BlindAid, texture is achieved by applying a force tangential to the normal force of the user's hand. The texture simulates a series of individual ridges that can be varied by type (saw tooth, sinusoidal), amplitude (height), period (length), and other parameters (see Schloerb, Lahav, Desloge, & Srinivasan, 2010)). Haptic objects include:

- *Walls* are solid surfaces that are defined by two end points in the horizontal plane and extend vertically from floor to ceiling. Walls have contact and long sounds.
- *Public Doors* are penetrable surfaces that are defined by two end points and extend from floor to ceiling like walls. Public doors feel solid like walls until a certain threshold force is applied, at which point, the proxy passes through to the other side. The "pop-through" force is adjustable. Public doors have contact and long sounds.
- *Areas* are polygonal horizontal regions in the VE, extending from floor to ceiling that play background sounds when the proxy is within their boundaries. Areas can be used to represent indoor and outdoor spaces such as restaurants and parks.
- *Ground Textures* are polygonal floor objects that simulate textures on the floor.
- *Rectangle Objects* are 3D rectangular prisms that are defined by two front end points in the horizontal plane, the horizontal depth and vertical height, and the vertical coordinate of the bottom. Rectangle objects can be defined to have any height and vertical location, as opposed to walls which extend from floor to ceiling.

3 GENERAL METHODS

Five experiments were conducted over the course of this thesis. The experimental procedures share a number of commonalities that are explained in this section.

All experiments were conducted using version v1.50 of BlindAid with the addition of features under test. v1.50 was adapted from an earlier version, v1.02, by upgrading the program to run under Win 7 x64, and developed using Visual Studio 2010. In addition, all experiments utilized a Desktop Phantom[1], which has a physical workspace of 160 W x 120 H x 120 D mm, corresponding to the region of the VE that the user explores (virtual workspace). The graphical displays shown in the map layout figures that follow correspond to the horizontal plane of the virtual workspace (top down view), such that the VE boundaries correspond to the physical W & H dimensions. The VE is measured in meters; this affects the audio since the volume and orientation of spatialized sounds are calculated based on these units (Schloerb, Lahav, Desloge, & Srinivasan, 2010). All of the maps used for experimentation in this thesis fit in the virtual workspace and the zoom feature was not used.

The total force applied by the Phantom was capped at 0.875 N, the maximum allowable continuous force divided by a safety factor of two. During initial tests, we discovered that the Phantom would shut down after just ten minutes of use. The problem appeared to be due to software designed to protect the device from overheating. To remedy the issue, we limited the continuous force by a safety factor of two. This extended the time before the device shut down to one to two hours. Finally, to accommodate test sessions that could last up to two hours, we swapped in a second Phantom when the first one showed signs of shutting down.

Each experiment is comprised of tasks performed in training and test settings for all conditions. All of the experimental tasks were 2D (e.g., involved a map layout that was effectively only in the horizontal plane) except for the second repulsive force field experiment. The order of trial conditions and maps were randomized in the experiment

---

[1] The original manufacturer of the Phantom, SensAble Technologies, Inc., was recently bought by Geomagic, 3D Systems Corp., and is now the Sensable Group at Geomagic. The "Desktop Phantom" has also been renamed "Phantom Touch X."

using a pseudo-random algorithm based on the rand function in Excel. The algorithm assigned each condition (or map) a value between 0 and 1 and then sorted these assigned values in ascending order to determine the condition (or map) order. In the training trials, the subjects were asked to explore a virtual map until they had an accurate understanding of the space as need for the particular test. Then, the subjects were tested on their knowledge of the space by performing identification tasks in which they were asked to illustrate (draw or build a physical model) or verbally identify the locations of certain objects in the VE. Alternatively or in addition, the subjects were asked to perform navigation tasks (finding their way along a specified path).

When the subject first began, the keyboard and Phantom was positioned to accommodate his/her dominant hand and overall comfort. The subject was then given a brief verbal introduction to BlindAid, the Phantom and the details of the experiment including the purpose, the design of the task (i.e. instructions), the commonalities of the test maps (i.e. controlled variables), and any other considerations for the test (i.e. aspects of the map to remember). In addition, subjects were introduced to using the Phantom and some basic features of BlindAid via a simplified map. In some experiments, the subject was introduced to additional tools needed for the task such as a physical modeling kit.

Before each training or test trial, the experimenter identified the condition to be used (e.g., force field or no force field) and gave the subject the option to hear the instructions again. For the trainings, subjects were encouraged to take their time learning to use BlindAid and exploring the map layouts. For the tests, subjects were encouraged to complete them as quickly and accurately as possible. Once the instructions were administered, the subject was blindfolded and given the BlindAid headphones to wear.

Subject explorations of the virtual maps were recorded and saved as data files that could be replayed by the BlindAid program for later analysis. Identification tasks that did not use BlindAid (e.g., the modeling task and drawing task) were photographed for the data record. Also, the times to complete the exploration, navigation, and/or identification segments of the test were recorded. Once the subject had completed the entire experiment, he/she was asked a set of questions to better understand his/her preferences for the conditions tested. The data is presented in graphical form in the results sections; in all graphs, a lower value corresponds to better rank/performance. The experimental procedure was approved by the MIT Committee on the Use of Humans as Experimental Subjects (COUHES).

3.1 SUBJECT SELECTION

The subjects used in this thesis were blindfolded sighted subjects that ranged in age between 23 and 33. Blindfolded sighted subjects are believed to be sufficient for the preliminary experiments described in this thesis because these tests are intended to guide the technical development of the system. Future tests aimed at evaluating the usability of the system will be conducted with blind subjects from the ultimate user population.

4 REPULSIVE FORCE FIELDS FOR GLOBAL HAPTIC SENSING

We designed and implemented a repulsive force field to emanate from BlindAid wall objects (see Section 2.4) and conducted two developmental experiments to evaluate the effectiveness of this feature in enabling global sensing. Experiment #1 tested the repulsive force field in an enclosed space and Experiment #2 was modeled after the original cityscape VE referred to in Section 1.3.

4.1 DESIGN

The process of implementing force fields in the BlindAid software began with brainstorming strategies to identify a suitable approach. Ideas included making the repulsive force a material property of the virtual object using an OpenHaptics parameter, HL_EFFECT_SPRING, and flipping the direction of the force produced. We also considered using the HL_CONSTRAINT parameter that specifies an attractive force to points, lines and planes. Descriptions of HL_EFFECT_SPRING and HL_CONSTRAINT can be found in Addendum B-13 and B-12, respectively, of the OpenHaptics Toolkit API Reference Manual (Sensable Technologies, 2008).

A third, more complicated approach was to do the low-level calculation of the combined forces acting on the proxy based on the distance and direction from each object. Unfortunately, the OpenHaptics library did not allow us to adapt HL_EFFECT_SPRING or HL_CONSTRAINT to be used in the desired manner.

Thus, we chose to develop the third idea, employing an algorithm that loops through a set of virtual objects and calculates the component force for each. We started with a sample OpenHaptics program that modeled an attractive force between two particles; when we flipped the sign of the force, we were able to create a repulsive force. We then incorporated the repulsive force code into BlindAid through a class called vmap_forcefield. This class was modeled after an existing BlindAid class (vmap_texture) that produced textures because that class generates forces associated with the surfaces of BlindAid objects as desired for the vmap_forcefield class.

The repulsive force field feature went through a number of iterations to address different design considerations. These considerations included determining the strength and variability of the force, and deciding from which objects the force should emanate. In the case of wall objects, the algorithm needed to determine whether the proxy was in front of the wall since walls are one-sided, and if the proxy was within a pre-determined distance from the wall. Additionally, it needed to calculate the force associated with wall corners correctly.

4.2 IMPLEMENTATION

The force field feature produces a repulsive force from designated objects. In this implementation, the repulsive force fields are limited to walls and 2D (independent of the vertical axis), so force calculations are only concerned with the horizontal plane. The force (spring) constant and the radius of influence (the distance from which the force field can be felt) can be varied depending on the environment. Additionally, the repulsive force can be toggled on and off, and the radius of influence can be adjusted manually during navigation in the VE.

Figure 1 illustrates the force vectors at different points along a grid in a simple map of rooms and hallways. The length of each vector represents the magnitude of the repulsive force one would feel at that specific point on the map in the given direction. The repulsive force field algorithm is part of the haptic loop which runs at 1000 Hz. This is necessary because the calculation is highly dependent on the real-time location of the proxy (user's position).

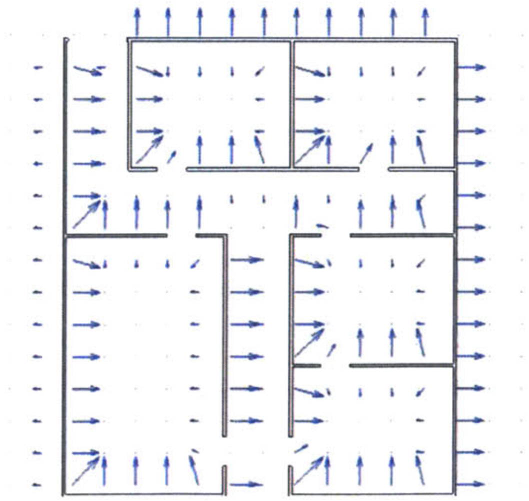


Figure 1. Illustration of force vectors in an example map

BlindAid maps are created by specifying the features of the VE in a text file called a vmap that may be read by the program. For example, a wall is specified in the vmap by its end coordinates, short and long sound files, minimum zoom level, color, and haptic properties. Each time the force field algorithm iterates, it searches through each wall contained in the map and returns the closest point on the wall to the proxy; this point is referred to as the "force point." This is either the perpendicular point on the wall if the proxy lies in front (or back) of it, or one of the end points. The program then determines if the distance between the force point and the proxy is within the radius of influence. If the force point is within range, a repulsive force that is dependent on distance is calculated. The following two sections present a general description of the algorithm. The actual program code is presented in Appendix A.

4.2.1 REPULSIVE FORCE ALGORITHM

The repulsive force is calculated based on the distance the proxy is away from the force point. It is greatest when the proxy is touching the object and decreases linearly as the proxy moves away. The repulsive force is zero outside of the radius of influence. If the calculated repulsive force is larger than the maximum force of the Phantom divided by a safety factor (in this application, we used a safety factor of 2), the force is capped at this safety value. The walls may be considered as line segments in the horizontal plane due to the 2D nature of the algorithm.

4.2.2 FORCE POINT ALGORITHM

The force point, the point from which the repulsive force is calculated, is either the point perpendicular to the proxy on the wall or one of the end points. These are the steps used to calculate this point for each wall:

1. Find the point at the intersection of an infinite line containing the wall segment and the perpendicular line which passes through the proxy. Check if this point lies on the wall segment or if it is outside of the two endpoints. If it does not lie on the wall, a "no value" coordinate (-1000000, -1000000, -1000000) is provided such that the program knows that a perpendicular point does not exist.
2. If there is a perpendicular point, check if it lies in front of the wall. Walls are only solid on the front side so it is important to keep the repulsive force restricted to this

side. If it is, set the force point to this point and skip the remaining steps, otherwise move to Step 3.

3. If there is either no perpendicular point or the proxy is not in front of the wall, find the end point on the wall that is closest to the proxy and the angle between the proxy and the wall. Each end point of a wall touches the end point of another wall so walls always meet at corners (see point B in Figure 2). If the angle between the proxy and the wall is smaller than half of the total corner angle, set the closest corner point as the "force point"; else, this wall does not contribute a "force point." To explain this further: when two walls meet at a corner, only one wall can be responsible for contributing the force point. Otherwise, the force calculated from this corner point will be doubled. Thus each wall has a certain angle range in which its corner point is valid (see Figure 2).

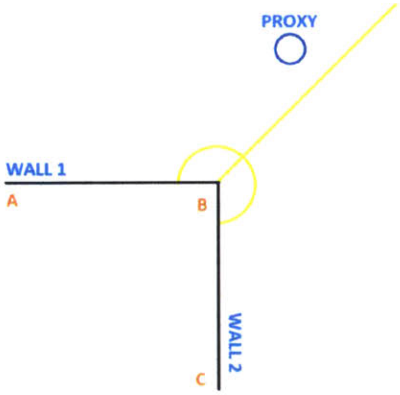


Figure 2. Illustration of how to determine which wall contributes the corner point (B). The figure shows that the angle from the proxy to WALL 1 (ABProxy) is less than half the total angle subtended by the two walls (yellow line) so the endpoint of WALL 1 contributes the force.

4.3 EXPERIMENT #1: FORCE FIELDS IN ENCLOSED SPACES

The goal of this experiment was to determine if repulsive force fields could be used to help learn about indoor spaces. In this experiment, subjects attempted to learn a floor plan of three rooms interconnected by three hallways. Then subjects were asked to build a physical model representing the space to evaluate their understanding of the layout (identification task).

4.3.1 METHODS

Subjects

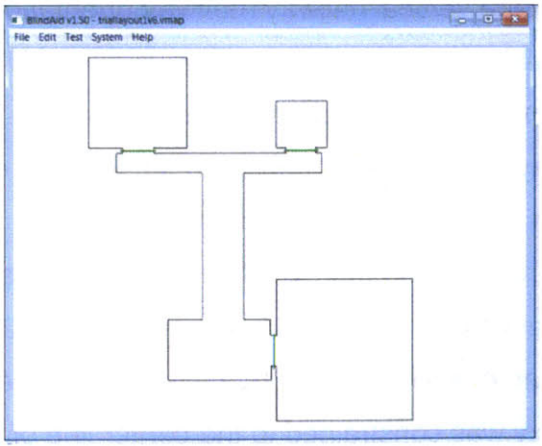
Subjects ranged from 23 to 33 years of age. Three subjects were male, one subject was female. Three subjects were right-handed, one subject was left-handed. All had normal or corrected normal vision, hearing, and sense of touch. Table 1 provides a summary of the tested subjects.

Subject ID	Gender	Age	Height	Weight (lb)	Profession	Hand	Vision	Hearing	Touch
1	M	23	5'8"	135	Student	Right	Normal	Normal	Normal
2	M	33	5'7"	134	Student	Right	Glasses	Normal	Normal
3	F	24	5'6"	126	Design engineer	Right	Normal	Normal	Normal
4	M	27	6'0"	190	Student	Left	Normal	Normal	Normal

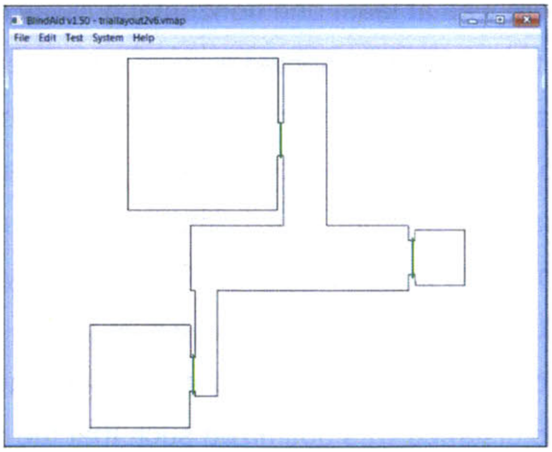
Table 1. Repulsive force field experiment #1 subjects

Arrangement

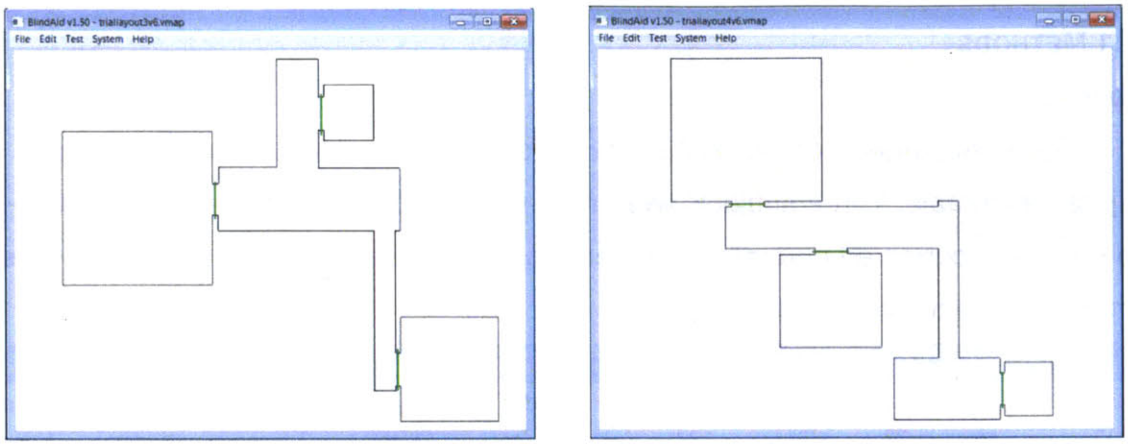
Four different layouts of similar complexities were used in the experiment (Figure 3). The x and y dimensions of the virtual workspace shown in the figure are 21.3 m by 16.0 m, where x is horizontal on the page and y is vertical on the page. In order to maintain a comparable level of complexity, the VE layouts consisted of alternating the same set of basic components (e.g., rooms and hallways) and relations (e.g., hallway to room, hallway to hallway).



Map 1



Map 2



Map 3 *Map 4*

Figure 3. Repulsive force field experiment #1 map layouts

Each map had three square rooms of varying sizes (small, medium, large) with each size represented once. Note that the corresponding virtual dimensions of the rooms were 2.1 m, 3.8 m, 5.7 m, respectively. Rooms were connected by three rectangular hallways of three potential widths (thin, medium, wide: 0.85 m, 1.7 m, 2.5 m) and three potential lengths (short, middle, long: 4.2 m, 6.1 m, 8.5 m), with each potential width and length represented once. For example, one map could contain a long thin hallway, a short wide hallway, and a middle medium hallway, but another map might consist of a middle thin hallway, a short medium hallway, and a long wide hallway. There was at least one horizontal hallway and one vertical hallway (hallway orientation). Each hallway intersected at least one other hallway and one hallway always ended in a dead-end (hallway parameters). Each room connected to a hallway but no more than two rooms were connected to the same hallway; one room connected to the left side of a hallway, another room connected to the right side of a hallway and the third room connected to the end of a hallway (hallway-to-room parameters). Additionally, the total distance between rooms – the sum of the distances from Room 1 to Room 2, Room 2 to Room 3, and Room 3 to Room 1 – was equal across all four layouts. Finally, the force field distance was optimized such that the proxy was constrained to the center line of the thinnest hallway but allowed to be move side-to-side with ease in the widest hallway. Subjects were not given the ability to adjust the force constant or radius of influence.

To simplify the map, all intersecting angles were 90 degrees and there were exactly eight potential connection points for a hallway. This meant that a room or hallway could only connect to another hallway at one of eight points: top left, top middle, top right, bottom left, bottom middle, bottom right, left and right (Figure 4).

Figure 4. The orange segments represent the eight potential hallway connection points

Procedure

The experimental task was to explore a floor plan of three rooms interconnected by three hallways until the subject believed he/she had a good understanding of it, and then identify different aspects of the layout using a physical modeling kit immediately after each trial. Aspects the user was asked to focus on included: room size, hallway widths and lengths, placement of rooms in relation to hallways, and hallway intersections. The experimental task was performed as a training trial and a test trial under each condition (No Forcefield, Forcefield), for a total of four trials. Table 2 presents the nominal order of the trials which were counter balanced across subjects. Either the No Forcefield condition was presented first in the Training #1 and Test #1 trials, and then the Forcefield condition in the second set of trials, or vice versa. The four maps were randomly assigned to the four trials for each subject. See Appendix B.1 for the actual ordering for each subject.

Trial	Condition	Map
Training #1	No Forcefield	1
Test #1	No Forcefield	2
Training #2	Forcefield	3
Test #2	Forcefield	4

Table 2. Nominal set of trials for repulsive force field experiment #1

In addition to the general procedure described in Section 3, subjects were introduced to a physical modeling kit that consisted of magnetic laser-cut ¼ inch thick acrylic replicas of the hallways and rooms to be used in the identification task (see Figure 5 for an example completed model).

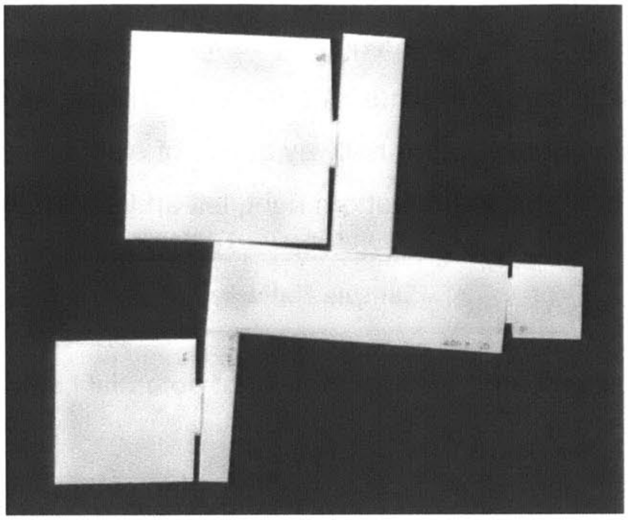


Figure 5. Example of a completed model. This is a model of Map 2 completed by Subject 4. The width of the region shown is eight inches.

Once the subject had completed the entire experiment, he/she was asked the following questions:

a. Was the task difficult? What parts were difficult? What parts were easy?
b. Did one condition seem more difficult than the other? Why?
c. What information (e.g., room size, hallway size) was made easier or more difficult with force fields?
d. Was using the force field easy or difficult?
e. Did using the force field help? If yes, in what way?
f. Do you have any other comments?

Variables

The dependent variables in this experiment included preference for the feature (i.e. the force field), build error (the number of errors in the physical model), build time (the time needed to build the physical model of the map), and exploration time (the time needed to learn the map layout). Build errors included selecting the wrong length or width hallway, placing a room in the wrong general location (e.g., placing the small room where the large room should be), and connecting a hallway to another hallway or room at the incorrect intersection point (of eight potential points on a hallway). See Table 3.

Independent Variable	Dependent Variables	Controlled Variables	
Use of force field	Preference for force field	Number of hallways	Hallway orientation
	Build error	Size of hallways	Hallway parameters
	Build time	Number of rooms	Total room-to-room distance
	Exploration time	Size of rooms	Hallway-to-room parameters

Table 3. Repulsive force field experiment #1 variables

4.3.2 RESULTS / DISCUSSION

Figure 6 shows the mean exploration time, physical model build time, and build error averaged across subjects for the two conditions, with lower values corresponding to better performance (error bars in the figure are ± 1 standard deviation). The complete set of data is included in Appendix B.1. The mean times for exploring the map and building the physical model, and the mean number of errors in the model were approximately equal for both conditions. These results suggest that while repulsive force fields can be used, they do not add to the user's ability to navigate and understand the VE.

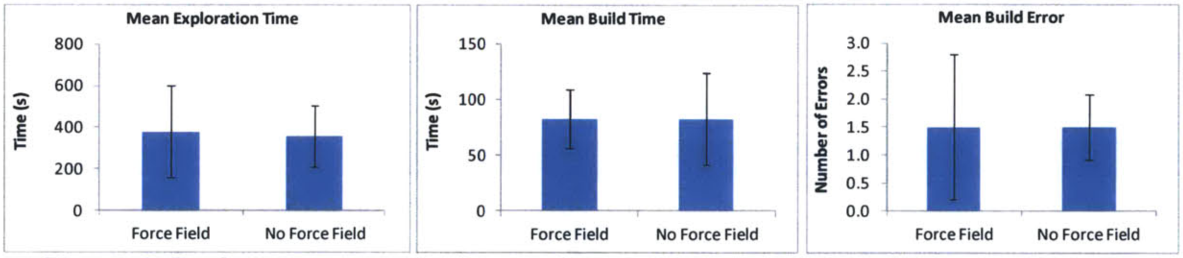


Figure 6. Performance metrics by condition of repulsive force field experiment #1

Indeed, the post experiment questionnaire revealed that users either did not notice the repulsive force field or actively ignored it. Observations of subject exploration during testing confirmed this. Additionally, all subjects expressed an affinity for physically touching the walls as opposed to relying on the force fields emanating from the walls. Without vision, subjects looked for solid objects to ground them during exploration; the force field, with its variable "surface," complicated the task of pinpointing and retaining the user's position in space.

Interestingly, the force field condition had a much larger standard deviation for build error as compared to the no force field condition, despite having approximately equal means. This may have been due to the issue that the force field was not intuitive. Perhaps with practice the build accuracy might improve.

While the application of repulsive force fields to all surfaces may not be ideal, it may be useful in a smaller scope to provide information about specific objects, such that it does not affect the user's ability to ground himself/herself. This leads us into the second repulsive force field experiment.

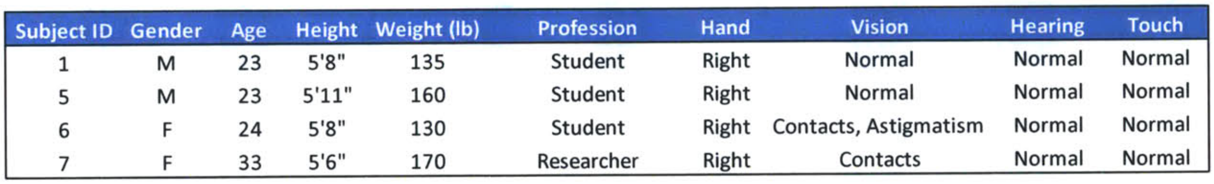
4.4 Experiment #2: Force Fields in Open Spaces

The goal of this experiment was to determine if repulsive force fields could be used to learn about outdoor spaces. In this experiment, the virtual objects of interest represented buildings and the task involved learning their locations while traveling along a virtual street.

4.4.1 Methods

Subjects

Subjects ranged from 23 to 33 years of age. There were two male subjects and two female subjects. All subjects were right-handed and had normal or corrected normal vision, hearing and sense of touch. Only subject 1 took part in the previous experiment. See Table 4.



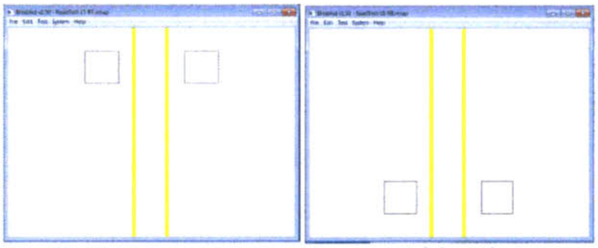
Subject ID	Gender	Age	Height	Weight (lb)	Profession	Hand	Vision	Hearing	Touch
1	M	23	5'8"	135	Student	Right	Normal	Normal	Normal
5	M	23	5'11"	160	Student	Right	Normal	Normal	Normal
6	F	24	5'8"	130	Student	Right	Contacts, Astigmatism	Normal	Normal
7	F	33	5'6"	170	Researcher	Right	Contacts	Normal	Normal

Table 4. Repulsive force field experiment #2 subjects

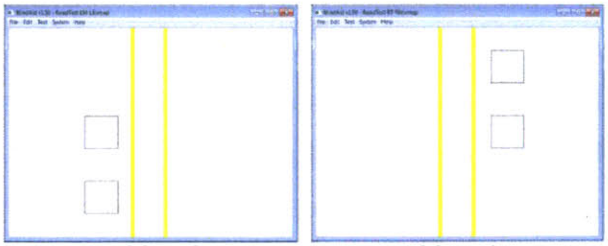
Arrangement

A total of 12 different layouts of similar complexities (Figure 7) were used for the 22 tasks; two maps were used for the training trials and the remaining ten maps were used for the test trials in both conditions (each test map was used in the force field condition and the no force field condition). Again, we designed the maps to be similar in complexity. The virtual workspace was 8.3 m in width and 6.5 m in length. Each map had a centered road 0.9 m wide, bounded by curbs on each side and two square buildings (1.0 m on a side) located somewhere outside of the road. Curbs were rectangle objects, rendered in yellow in the figure that extended from the floor up 1/10 of the distance to the ceiling. Buildings were made of four walls that extended from floor to ceiling. The subject was able to move along the edge of the curb as a guide if the proxy was touching the ground, but could also jump over the curb and explore the rest of the map if the proxy was moved up slightly in the vertical axis (out of the page). There were six potential locations for the two buildings – bottom left, middle left, top left, bottom right, middle right, or top right.

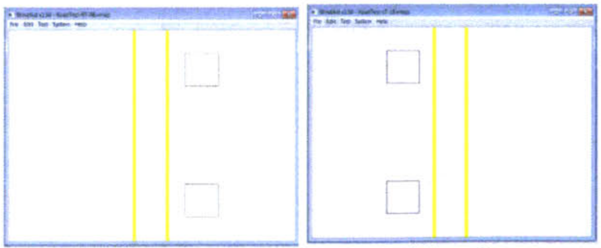
While the maps were believed to be similar in complexity as they were comprised of the same components, the subject could still believe one map was more challenging to explore than another based on the placement of the two buildings. Thus, we aimed to create a uniform distribution of different map arrangements. Specifically, the test maps consisted of five categories of building arrangements with two maps in each category as shown in Figure 7. The training maps were a sixth category. Lastly, the force fields emanating from the buildings were designed so that users could sense the buildings while moving along the road; the force field's radius of influence was approximately the length of one side of a building in the horizontal plane.


Map 1 Map 2

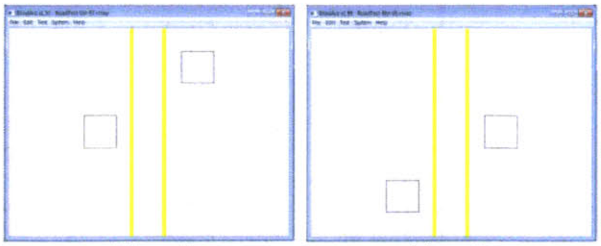
(i) Directly across from each other


Map 3 Map 4

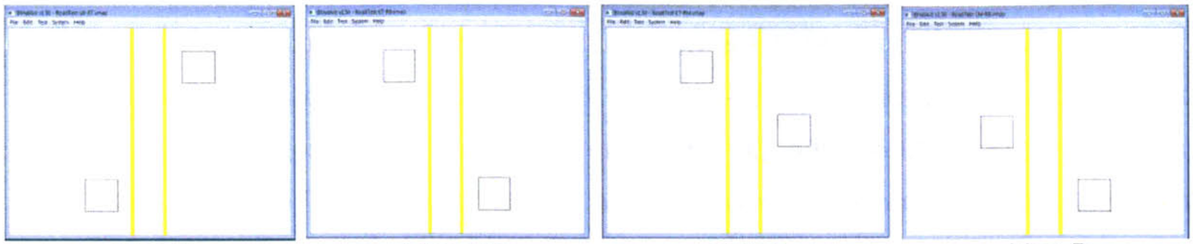
(ii) Next to each other on the same side


Map 5 Map 6

(iii) Separated by one unit on the same side


Map 7 Map 8

(iv) Across from each other, offset by one unit



Map 9 Map 10 Map A Map B

(v) Across from each, offset by two units (vi) Training maps

Figure 7. Repulsive force field experiment #2 map layouts

Procedure

The experimental task was to explore a virtual street with buildings on either sides of it and identify the locations of the buildings verbally. Overall, the procedure was very similar to that of Experiment #1. Specifically, in terms of counterbalancing the conditions, either the No Forcefield condition was presented first with its training trial followed by ten test trials, and then the Forcefield condition with its training and ten test trials, or vice versa. Table 5 presents the nominal order of the trials which were counter balanced across subjects. See Appendix B.2 for the actual ordering for each subject. The two training maps were randomly assigned to the two conditions and the ten test maps were randomly assigned to the ten tests of each condition. The main difference in Experiment #2 was that subjects gave verbal responses to indicate the locations of the buildings rather than build a physical model. The maps were also simpler, enabling more trials in a test session. The post experiment questionnaire was identical.

Trial	Condition	Map	Trial	Condition	Map
Training #1	Force Field	A	Training #2	No Forcefield	B
Test #1	Force Field	1	Test #11	No Forcefield	1
Test #2	Force Field	2	Test #12	No Forcefield	2
Test #3	Force Field	3	Test #13	No Forcefield	3
Test #4	Force Field	4	Test #14	No Forcefield	4
Test #5	Force Field	5	Test #15	No Forcefield	5
Test #6	Force Field	6	Test #16	No Forcefield	6
Test #7	Force Field	7	Test #17	No Forcefield	7
Test #8	Force Field	8	Test #18	No Forcefield	8
Test #9	Force Field	9	Test #19	No Forcefield	9
Test #10	Force Field	10	Test #20	No Forcefield	10

Table 5. Repulsive force field experiment #2 example setup

Variables

The dependent variables in this experiment were: preference for the feature, identification error (the number of buildings the subject located incorrectly), and exploration time (the time needed to locate the buildings). See Table 6.

Independent Variable	Dependent Variables	Controlled Variables	
Use of force field	Preference for force field	Size of road	Size of buildings
	Identification error	Location of road	Number of buildings
	Exploration time	Size of curbs	Location of buildings
		Location of curbs	

Table 6. Repulsive force field experiment #2 variables

4.4.2 RESULTS / DISCUSSION

Figure 8 shows the mean exploration time and identification error by condition averaged across subjects, with lower values corresponding to better performance (error bars in the figure are ± 1 standard deviation). The exploration time of the two conditions was essentially equal; and on occasion, the force field condition took longer (see Appendix B.2). In the eighty test trials conducted, only two trials, both in the force field condition, resulted in an identification error. This may imply that mistakes are more easily made in the force field condition. Similar to the prior experiment, the results suggest that the repulsive force field is usable but does not increase learning efficiency.

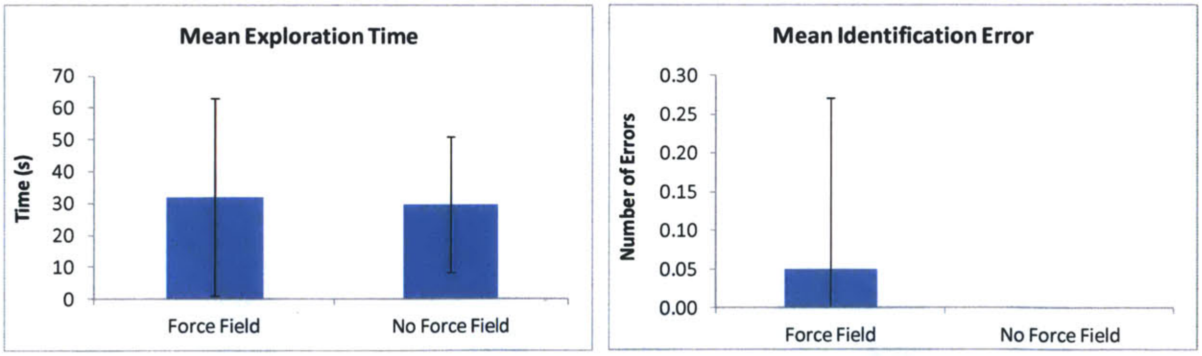


Figure 8. Performance metrics by condition of repulsive force field experiment #2

Discussions with the subjects revealed that they found the force field feature easy-to-use because it required less overall exploration; however, they also expressed a few issues. Comments concerning issues included:

- The force field was helpful until you tried to touch the building, at which point it pushed you away.

- It took longer to translate and verify the locations of the buildings because you could only sense that it was there. However, practicing made the ability to locate easier.
- You had to know exactly where you were in the map for it to be helpful; otherwise there was no clear location reference.

The results of both experiments were insufficient to support the use of repulsive force fields to improve global sensing. While more experimentation to support or refute this claim was possible, we believed that the preliminary tests were convincing enough to switch gears and consider another type of force field.

4.5 DISCUSSION OF LEARNING CURVES IN THE TWO EXPERIMENTS

Before moving on to development and testing of the new force field, we did the following analysis and identified a useful change to the initial experimental procedure. Specifically, in the first two force field experiments, the training and test trials were grouped together for each condition. This meant that the subject was trained and tested in the first condition, and then trained and tested in the second condition. In our initial design of the experiment, having one training was thought to be sufficient as the subject was given as much time as he/she needed to feel comfortable with the apparatus. Analysis of the performance metrics with respect to the first and second tests of Experiment #1, independent of the force field condition, shows a modest improvement across all three metrics (Figure 9).

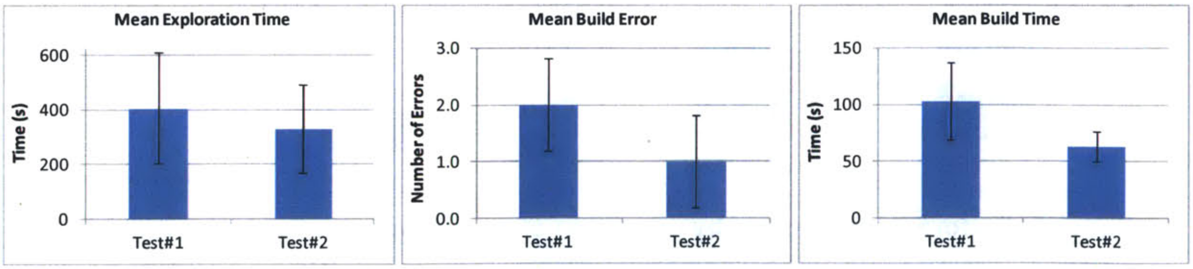


Figure 9. Performance metrics by test number of force field experiment #1 independent of condition

A similar trend can be observed for the second experiment (Figure 10). The first few tests show large variability in performance but quickly approach an asymptote.

Based on these observations, we adjusted the procedures of all following experiments. The procedure conducts trainings before any tests to provide sufficient time to stabilize the learning curve.

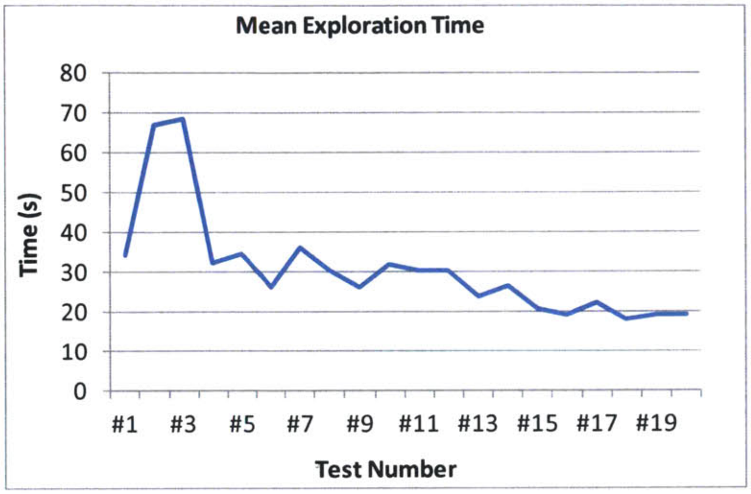


Figure 10. Exploration time by test number of force field experiment #2

5 ATTRACTIVE FORCE FIELDS FOR GLOBAL HAPTIC SENSING APPLIED TO ROUTE GUIDANCE

Route guidance is an important aspect of navigation for all people. Route guidance can provide users with the most efficient route to a specific destination as well as alternate routes. Depending on the size of the map, finding places like the nearest grocery store without guidance can be very difficult and even impossible. While there are many ways to implement a guide agent, the most commonly used guide strategies are to provide instructions visually and/or verbally. With blind people, the audio channels can become overloaded because much more information is presented aurally. This issue is also apparent in the BlindAid program; audio is an easy tool to use to convey information so using haptics when possible can alleviate the burden placed on the audio channels.

Because simply being lead through a route is insufficient for route learning, we thought to use attractive force fields for global sensing so that users could keep track of objects (or the route), while still learning the space around them. We first devised a concept to set tether points while exploring so that users could explore without losing track of where they were. From this, stemmed the idea of creating a path of tether points that would comprise a route to guide the user (by pulling him/her along it) while still allowing the user to explore around the route. We decided to investigate global sensing with an attractive force field in the context of this application.

The remainder of the chapter presents our development of the haptic route guide chronologically, beginning with discussions of the initial design and implementation of the guide. We then discuss the first developmental experiment in which we evaluated four potential haptic guide designs and selected the most promising one. The plan was to compare the most promising haptic guide to conventional alternatives in a main experiment. Before moving on to the main experiment, we developed an audio guide to provide a standard for comparison in the main experiment along with the no guide condition. The audio guide development involved an initial design and implementation phase in which four candidate audio guides were created, and then experimentally evaluated to select the best one similar to the initial haptic guide development. The experimental procedures were also refined in the first two experiments for the main experiment. The results of initial tests in the main experiment suggested further improvements to the haptic guide were needed. The haptic guide design was revised accordingly and the final main experiment was performed. The final results showed the haptic guide was indeed effective for global sensing and route guidance.

5.1 INITIAL DESIGN

GPS devices are commonly used by drivers for route guidance in the real world. These devices use audio to verbally convey instructions about the route at key points, called way points, where the user must act to stay on the route. Typically, a GPS will first inform the user that a new route action will be required ahead and then instruct the action. GPS devices for blind persons typically follow this methodology, but can also convey additional audio information that is normally presented visually for sighted users. Such additional information includes the user's current location with respect to addresses, streets and intersections. Because these types of audio features can always be added to the haptic guide later, we focused primarily on developing the haptic guide.

5.1.1 HAPTIC GUIDE COMPONENTS

There are three components of the initial haptic guide concept that we proposed: (1) guide line, (2) tether, (3) forward force.

Guide Line

The guide line (initial concept) is a set of line segments going from one way point to the next along a route. While other shapes could be used to specify the route (e.g., a surface representing a sidewalk or street), a set of line segments was thought to be sufficient and certainly the simplest way to start. Initially, the concept represented line segments haptically as snap-to objects. A snap-to object produces a strong attractive force to the proxy when the two are very close to each another. We had also considered representing way points and curved segments haptically. However, since curve segments could be represented by straight line segments and were not needed in the actual experiments, we decided to work with line segments.

Tether

The "tether," which allows the user to explore the region around the route while keeping track of the guide line, is an attractive force between the proxy and another point on the guide line – this point is called the tether point or anchor point. Anchor points may be located at any point along the guide line, nominally shifting in the forward direction as the user progresses along the route. The placement of anchor points can be determined using a number of different strategies. These strategies include setting the anchor point to: the last touched point on the guide line, the closest point to the proxy on the guide line (moves with the user), and important way points such as turns that can be selected manually or automatically to progress in the forward or reverse direction. Additionally, the attractive force can be constant or vary by distance, linearly or non-linearly.

Forward Force

The "forward force" concept is a force that can push/pull the user in the forward direction along the route, toward the destination. A few strategies were devised for how to use the forward force: (1) the anchor point could move along the guide line, pulling the user along via the tether (2) the forward force acts only when the user is snapped to the guide line, (3) the tether to way points acts as a forward force (by placing the tether point ahead of the user). Again, the force can be constant, linear, or non-linear.

5.1.2 SELECTION OF CANDIDATE HAPTIC GUIDES

We explored many alternative design concepts of the proposed haptic guide, involving different implementation and combinations of the three components.

- *Guide line only* gives more information than no assistance but does not provide information about the forward direction.
- *Tether to last touched point on guide line* produces no forces when the user is in contact with the guide line, but creates an attractive force to the last touched point on the guide line when the user moves away from it.
- *Tether to closest point on guide line* produces an attractive force to the closest point on the guide line only when the user is not in contact with it. The closest

point strategy is not ideal because the tether point can switch abruptly in concave segments of the route.

- *Manual forward force* uses the idea of attaching a tether to the next way point. Specifically, the user manually selects the next way point to be the tether anchor point so that he/she is pulled in the desired direction along the route. This concept may be implemented with or without a guide line and the tether enables the user to explore the region around the current way point without losing contact with the route.
- *Automatic forward force* is similar to the manual version, except the user does not have manual control over which way point is selected to be the anchor point. Instead, the anchor point automatically changes to the next way point when the user arrives at the current way point.
- Various other forward force concepts were considered (e.g., a force acting in the forward direction when the proxy is on the guide line; or a forward force acting on the anchor point of a tether, pulling the user forward along the guide line via the tether), but all cases in which the forward and tether forces functioned independently, were found to be confusing or caused stability issues with the Phantom.

5.2 INITIAL IMPLEMENTATION

Based on preliminary testing in the lab (without subjects), the following four concepts were selected to implement and test in the first developmental experiment. The haptic guide program code can be found in Appendix A.4.

5.2.1 GUIDE LINE

The concept for this implementation was to use the guide line only design to demark the route, with no tether and no indication of the forward direction. This is the simplest of the four concepts and is used in combination with other features in the other three haptic guides. The entire guide route is made up of multiple guide lines in succession. A guide line is defined by two end points much in the way walls and doors are defined in the program. However, unlike walls, guide lines are straight line segments lying on the floor as opposed to defining a plane. Additionally, wall entities are considered contact objects meaning that the proxy cannot pass through them, while guide lines are called constraint objects; constraint objects force the proxy to their surface when it is within the snap distance. In this implementation, the snap distance was 1.0 mm in the physical workspace of the Phantom. The OpenHaptics API allows this feature to be added with a couple lines of code (see Sensable Technologies, 2008, p. 9-3 and 9-4):

```
hlTouchModelf(HL_SNAP_DISTANCE, 1.0);
hlTouchModel(HL_CONSTRAINT);
```

Other haptic properties such as friction and damping could also be specified using the hlMaterialf function (see Sensable Technologies, 2008, p. 9-2).

5.2.2 TETHER TO LAST TOUCHED POINT

This guide, which is also referred to as the "tether" guide, adds a tether feature to the guide line to keep track of the last location visited on the route. When the user is on the guide line, no forces are produced, but when the user leaves the guide line, the program stores the user's last location and produces an attractive force to this anchor point on the guide while the user explores the neighboring area. Again, there is no information about the forward or backward direction along the route. In this implementation, the anchor point is updated every time the proxy touches the guide line. The program code loops through the array of guide lines (the array is created when the map first loads) and determines whether the proxy is on a guide line; if the proxy is found to be touching a guide line, the anchor point is reset to the proxy's new location. The tether force is always on, but while the proxy is on a guide line, the distance between it and the anchor point is too small to produce a detectable force.

Additionally, the user may increase or decrease the strength of the force (by adjusting the force constant) by pressing the up and down arrow keys. This option allows the user to adjust the strength of the tether to his/her comfort level as well as exercise the option to explore freely when desired. The lower limit of the force constant is zero; at this setting, the force is effectively turned off.

5.2.3 MANUAL FORWARD FORCE

The manual forward force guide (also referred to as the "manual" guide) uses guide lines in conjunction with a forward force that is implemented by a tether to the next way point. The design enables the user to explore the region around the current way point without losing contact with the route through the use of a static variable, iAnchorID, which keeps track of the current way point. When a map loads, the anchor is set to the starting route way point. The user can then move the anchor point forward or backward by increasing or decreasing the value stored in iAnchorID by pressing the right or left arrow keys, respectively. The user is tethered to the selected anchor point whether he/she is on or off the guide line. All way points are associated with a unique ID number and the range of ID numbers are limited by the number of way points. When the value stored in iAnchorID changes, the algorithm reassigns the anchor to the way point corresponding to the new value and recalculates the attractive force. If the user reaches the destination and presses the forward (right) key or presses the backward (left) key while at start, the value stored in iAnchorID and the anchor point do not change. The user can also increase or decrease the strength of the force as described in the previous section (5.2.2).

5.2.4 AUTOMATIC FORWARD FORCE

The automatic forward force guide (also referred to as the "auto" guide) is implemented in essentially the same way as the manual guide described in the previous section, except the location of the anchor point is updated automatically. Specifically, the proxy is initially anchored to start. Once the proxy touches start (the first way point), the anchor point moves to the second way point on the route. If the proxy reaches the second way point, the anchor point then switches to the next way point. This process continues until the user reaches the end of the route. The user can also switch the travel direction by pressing the left or right arrow key, such that he/she can go towards the guide-start instead of the guide-end, and adjust the strength of the force as mentioned in the previous two guide implementations.

5.3 HAPTIC ROUTE GUIDANCE EXPERIMENT

The focus of this experiment was to first, determine whether haptic guides could be used to facilitate route learning and second, to find the best haptic guide among the designs considered. The experiment consisted of learning and evaluation tasks completed eight times: the first four trials were for training in each condition, the last four trials were for testing in each condition (Table 7). As discussed in Section 4.5, we chose to group the four trainings together to give the user more time to become acclimated with using BlindAid.

5.3.1 METHODS

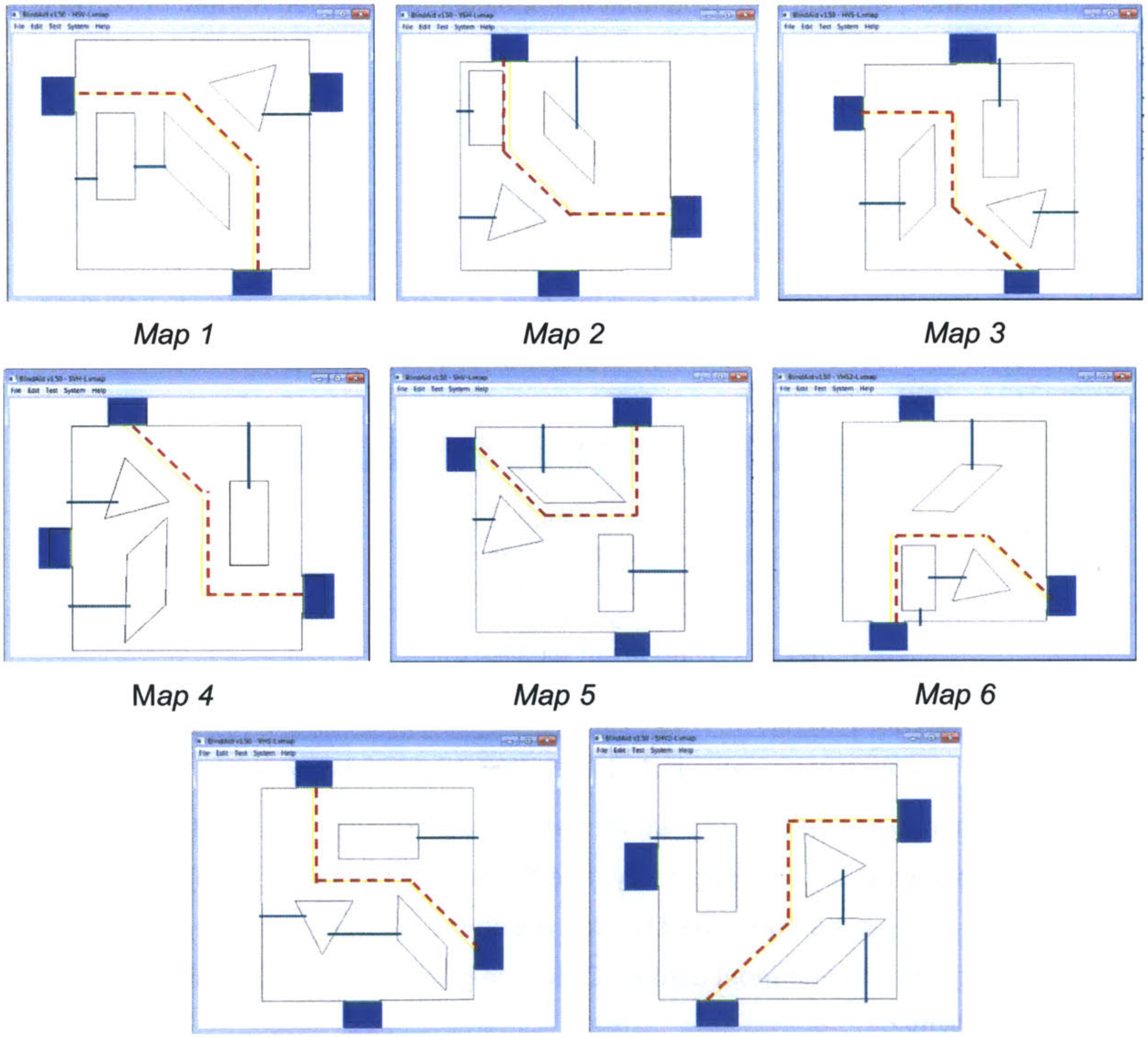
Subjects

The subjects used in this experiment were the same subjects from the second force field experiment. See Section 4.4.1.

Arrangement

A total of eight different layouts of similar complexities were used (Figure 11). In order to maintain a comparable level of complexity, the VE layouts consisted of alternating the same set of basic components. Each map had a guide route consisting of three guide lines in random order: one horizontal, one vertical and one with a 45 degree slant. The guide route had two turns and each guide line was 4.5 m long. The map itself consisted of a room with three doors (one entrance, one exit and one extra), three objects of different shapes (triangle, rectangle, and parallelogram) constructed of the same type of virtual "wall" object used to construct the room (see Section 2.4), and three barriers to obstruct any other complete routes to the exit besides the one indicated by the guide route. The triangle was 3.0 m long on each side, the rectangle was 3.8 m by 1.7 m, and the parallelogram was 4.0 m in width and 1.7 m in height. The VE workspace varied from 15.4 m by 11.6 m to 17.5 m by 13.1 m to fit the different configurations of objects and guide lines. Walls, doors and barriers had different contact sounds to help the user identify touched objects; shapes used wall contact sounds since they were constructed of wall objects. The beginning and end of the route were placed such that the forward direction of the route was always left to right. Each door lead to a small contained space that played a background sound ("Entrance," "Exit," and "Wrong

Door") to indicate location. Barriers allowed subjects to enter incorrect paths but not reach the exit door using these paths. This configuration provided opportunities to observe errors in the subject's navigation without allowing him/her to complete the exercise by reaching the exit through an alternate path. Additionally, shapes were included to test whether subjects could explore and learn about objects that were not directly on the path.



Map 1 *Map 2* *Map 3*

Map 4 *Map 5* *Map 6*

Map 7 *Map 8*

Figure 11. Haptic guide experiment map layouts

Procedure

The experimental task was to explore a room layout and a specific route through it (as many times as needed to learn), draw certain aspects of the room and route on a piece of paper, and then navigate through the room without the guide, as quickly and accurately as possible. Subjects were asked to focus on the locations of the following objects when drawing the layout: guide lines (route), shapes with respect to the guide lines, the entrance, and the exit. Note that the subjects, who all had normal or corrected normal vision, were allowed to take the blindfold off while making the drawing.

The experimental task was performed as a training trial and a test trial under each condition (Guide Line, Tether to Last Touched Point, Manual Forward Force, Automatic Forward Force), for a total of eight trials. Table 7 presents the nominal order of the trials which were counter balanced across subjects. The eight maps were randomly assigned to the eight trials, and the four conditions were randomly assigned separately to the training and test trials for each subject. See Appendix B.3 for the actual ordering for each subject.

Trial	Condition	Map
Training #1	Guide Line	1
Training #2	Tether to Last Touched Point	2
Training #3	Manual Forward Force	3
Training #4	Automatic Forward Force	4
Test #1	Guide Line	5
Test #2	Tether to Last Touched Point	6
Test #3	Manual Forward Force	7
Test #4	Automatic Forward Force	8

Table 7. Nominal set of trials for haptic guide experiment

In addition to the general procedure described in Section 3, subjects were given instructions on how to use the restart key (see Section 2.4). Once the subject had completed the entire experiment, he/she was asked the following questions:

a. Which guide was the most useful? Order the guides from most useful to least useful. Why?
b. Which guide was the least confusing to use? Order the guides from easiest to most difficult to use. Why?
c. Which guide gave you the most confidence in knowing how to navigate from start to end? Order the guides from most confident to least confident. Why?

d. Do you have any other comments?

Variables

Independent Variable	Dependent Variables	Controlled Variables	
Haptic guide type	Preference for guide	Number of guide lines	Number of shapes
	Navigation error	Orientation of guide lines	Type of shapes
	Navigation time	Size of guide lines	Size of shapes
	Identification error	Number of turns	Number of barriers
	Exploration time	Number of doors	Number of complete paths

Table 8. Haptic guide experiment variables

The variables in this test are listed in Table 8. The dependent variables were:

- *Preference for the guide* was an average of three subject rankings; the rankings measured helpfulness, ease of use, and the ability to instill confidence in navigating the route after learning. This method of determining subject preference was used for all route guidance experiments.
- *Navigation error* referred to the number of wrong turns made when navigating without the guide. An error was logged every time the proxy passed the corner of a side (of an object) that was not in the direction of the guide line. In the example map below, entering the areas enclosed by the pink dotted lines would have resulted in a navigation error (Figure 12).
- *Navigation time* was the total time needed to navigate from start to finish.
- *Identification error* was the number of errors in the drawing. Errors included incorrect placement of the guide lines (orientation), the start and finish doors, and the shapes with respect to the guide lines (the shape needed to be next to and on the correct side of the correct guide line).
- *Exploration time* was the time needed to learn the route and layout.

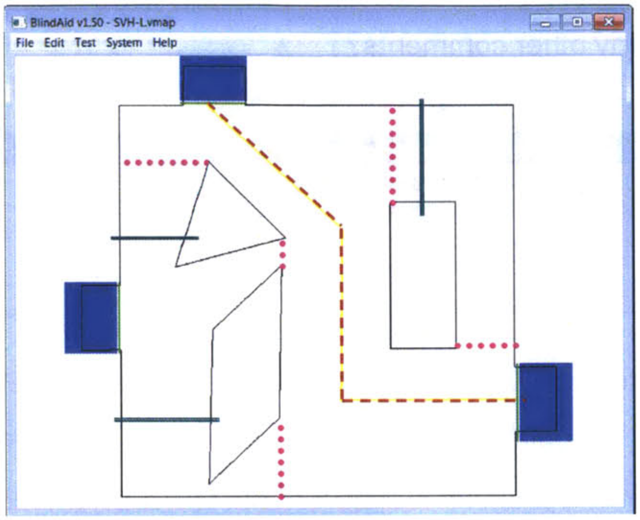


Figure 12. Entering the areas enclosed by the pink dotted lines would result in a navigation error

5.3.2 RESULTS / DISCUSSION

The following figures (13, 14, 15) present the mean results averaged across all subjects, with lower values corresponding to better performance (error bars in the figure are ± 1 standard deviation). The complete set of data is included in Appendix B.3. Subjects communicated a preference for the manual guide (Figure 13) which the navigation time and error data supported (Figure 14). The manual guide resulted in the highest preference as well as the fastest time for navigation and the least number of navigation errors. The automatic guide was the next preferred guide, but its navigation performance was the least favorable (Figure 14). The line guide and tether guide were the least preferred (Figure 13).

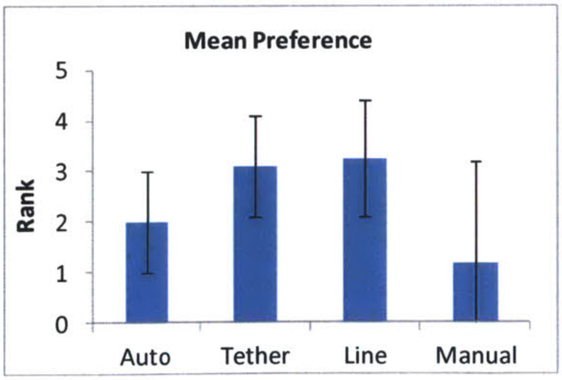


Figure 13. Mean preference of haptic guides

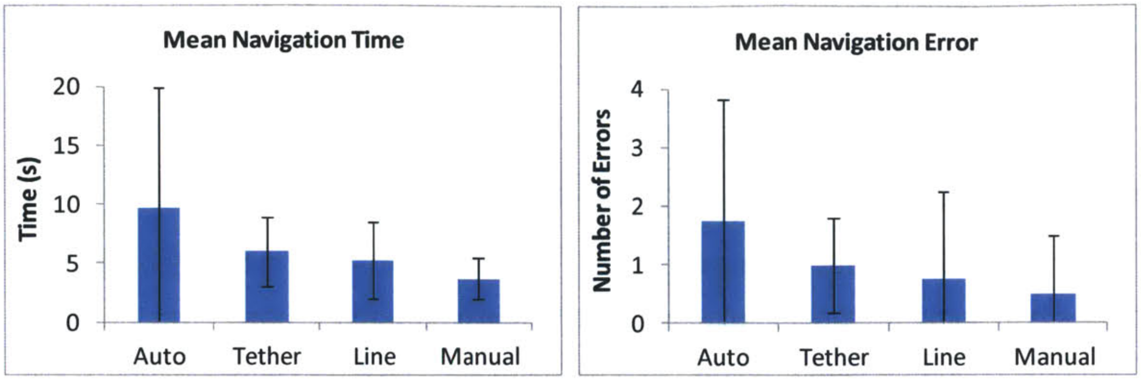


Figure 14. Mean navigation time and error of haptic guides

Exploration times were comparable (Figure 15), suggesting that users were capable of using all four guides. The mean number of identification errors was also comparable; however, the manual guide had the largest standard deviation.

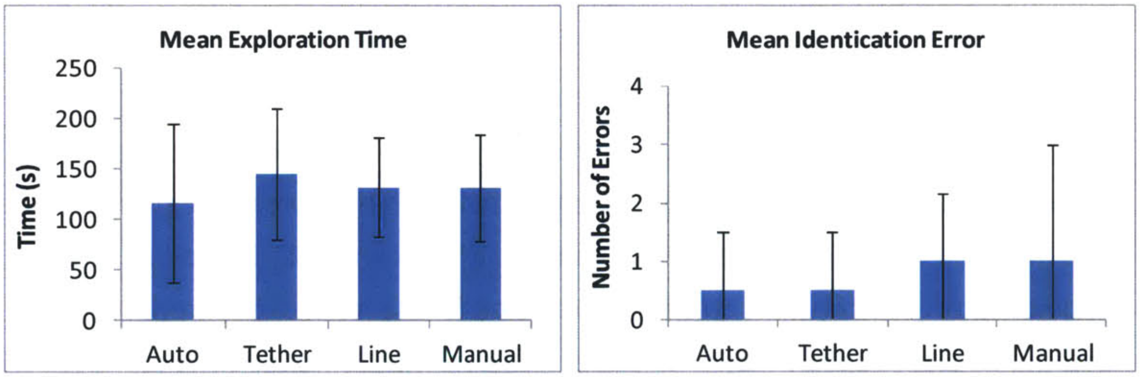


Figure 15. Mean exploration time and identification error of haptic guides

For the manual guide, users liked that they could move at their own pace and choose to turn off the guide as desired (by setting the force constant to zero). Both the automatic and manual guides could be used to give an overall sense of the route, but some users struggled with the "forcefulness" of the automatic guide as it pulled the proxy along the entire path without pause. The data also suggested that users had trouble with retaining the navigation information presented using the automatic guide. Conversations with subjects provided some insight: subjects believed that the automatic guide provided too much assistance, making it easier to rely on the guide and not on one's actual knowledge of the space. On the other hand, users felt that the guide line gave too little information, while the tether guide gave confusing and unintuitive information. While there isn't a full proof case in favor of one guide over the others, the manual guide appeared to be the best option to compare with conventional alternatives in the main experiment.

5.4 AUDIO ROUTE GUIDE DEVELOPMENT

The purpose of developing the audio route guide was to create a good standard with which to compare the usefulness of the haptic guide. At the base level, all audio guides provide information about when a decision such as a turn must be made. We looked into existing audio guides such as the Trekker Breeze GPS and Sendero Maps to identify "standard features." Features included:

- Providing step by step instructions to a specific destination
- Announcing the user's current location on command ("Where am I?" feature)
- Announcing street and intersection information as the user encounters them
- Recording specific routes while traveling
- Recording landmarks along a route that play automatically when within range

We developed four alternate designs to simulate the features of a standard audio guide in the VE (except for the recording features which can easily be added to the haptic guide later) and the best guide was selected for the main experiment in a developmental test (Section 5.6). One guide was designed to behave like an audio GPS device, giving step by step instructions automatically. Another guide, a manual guide, was modeled after the most successful haptic guide, since we believed that an audio guide that mirrored the haptic guide could provide a fair comparison.

One potential advantage of an audio guide in the BlindAid VE over existing audio guides is the ability to use spatialized sound. Spatialized sound can provide the user with more information about the location of the way point without additional message content. To take advantage of this feature, we introduced two additional audio guides. The third guide adapted the manual guide to use spatialized sound – we could have also added spatialized sound to the automatic guide, but did not, to limit the number of test scenarios. The fourth guide used spatialized audio in a way that allowed users to hear the entire path through "audio landmarks" (see Section 2.4).

The implementation of the audio guides employed the use of a class called vmap_locations that is modeled after another BlindAid class called vmap_areas. Vmap_areas play background sounds when the proxy is within a specified horizontal region of the VE (see Section 2.4). For simplicity, the guide divides the map into a grid and treats each unit of the grid as a separate location. When the proxy is contained within the perimeter of a way point location (a location where the proxy must turn to stay on the route), an identification sound is played. The ID sound plays once every time the proxy enters or re-enters the location. For the manual guide, the ID sound is the way point ID number, and for the automatic guide, it is the next set of instructions. The design and implementation of each guide is described in depth below and the program code is presented in Appendix A.5.

5.4.1 MANUAL AUDIO GUIDE

The manual audio guide consists of way points that announce themselves automatically when the user first enters them. The first way point after start has an ID number of one, the second way point has an ID number of two, and so forth. Off-route locations announce themselves as “off-route” so the user is always aware of whether he/she is on or off the route. Users can toggle the “auto-announce” feature of the guide on and off by pressing the Ctrl key; this can be used to test their knowledge of the map. The location ID sound (e.g., "start," "way point 5," "between way points 4 and 5," etc.) will also play if the user presses the down arrow key to provide a "where am I" feature. While on the route, instructions to the prior or next way point can be accessed by pressing the left or right arrow key, respectively. While off of the route, pressing the left or right arrow key will only provide the general direction in which to find the route.

5.4.2 MANUAL SPATIALIZED AUDIO GUIDE

The manual spatialized audio guide differs from the manual audio guide in that it utilizes spatialized sound when providing instructions. When the user presses the left or right arrow key, the instruction sound will originate from the destination way point (e.g., “left to way point 5” will originate from way point 5 on the left), giving the user an additional sense of the direction.

5.4.3 AUTOMATIC AUDIO GUIDE

The automatic audio guide also consists of way points that announce themselves when entered, but instead of playing the way point ID number, it plays the instructions to the next or prior way point depending on whether the direction of motion is set to forward or backward. The down arrow key still functions as a "where am I" tool and the left and right arrow keys change the direction of motion to backward and forward, respectively. Off-route locations also announce instructions to the route rather than saying "off-route." Off-route instructions only indicate the general direction in which the user should move to exit the dead-end. For example, if the dead-end is to the left of the route, then the instructions may say "right to route."

5.4.4 LANDMARKS GUIDE

Similar to the other guides, the landmark guide consists of way points that announcement themselves automatically and includes the "where am I" tool. Additionally, there is a separate set of way point audio files that can be accessed independent of where the proxy is located. The user can scroll through the way point ID sounds using the left and right arrow keys and repeat an ID sound by pressing the up arrow key. This independent set of way point ID sounds begins at "start." Pressing the right arrow key will cause the "way point 1" spatialized audio file to play (sound originates from way point 1 – see Section 2.4), pressing the right arrow key again will then cause the "way point 2" spatialized audio file to play, and so forth. This guide is unique in that it allows the user to get a sense of the entire route through spatialized sound, but also navigate locally by playing the way points around the user's current location to determine the direction in which to move next.

5.5 AUDIO ROUTE GUIDANCE EXPERIMENT

The design of this experiment mirrored that of the haptic route guide experiment except the VE arrangement was a 4 x 4 grid maze. The task consisted of exploring the route and space using the audio guide and then navigating through the same maze without the guide and with a focus on time and accuracy.

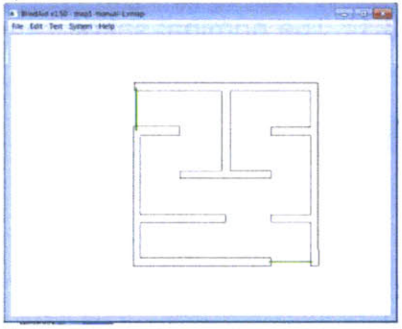
5.5.1 METHODS

Subjects

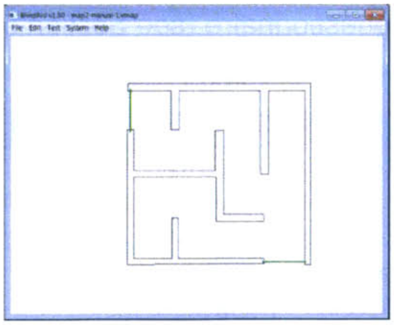
The subjects used in this experiment were the same subjects from the second force field experiment. See Section 4.4.1.

Arrangement

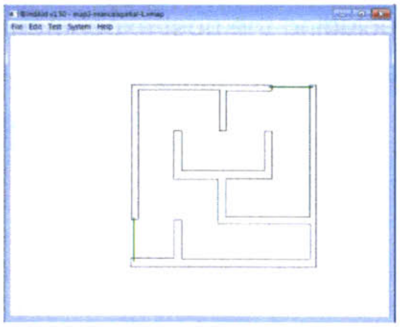
A total of eight different layouts of similar complexities were used (Figure 16). Each map was a closed 4x4 maze with only one route from the entrance to the exit that was always the same length, moved from left to right, and had the same number of turns (seven). Each grid square was 2.1 m in length and the virtual workspace was 18.8 m by 14.1 m. The entrance and exit areas announced themselves when entered by playing distinct background sounds. Additionally, the maze always had two dead-ends. A maze design was chosen for this task because its grid-like structure was well-suited for the audio guide implementation. Moreover, navigation errors could be more easily assessed with a maze as each unit of space was dedicated to either the route or a dead-end.



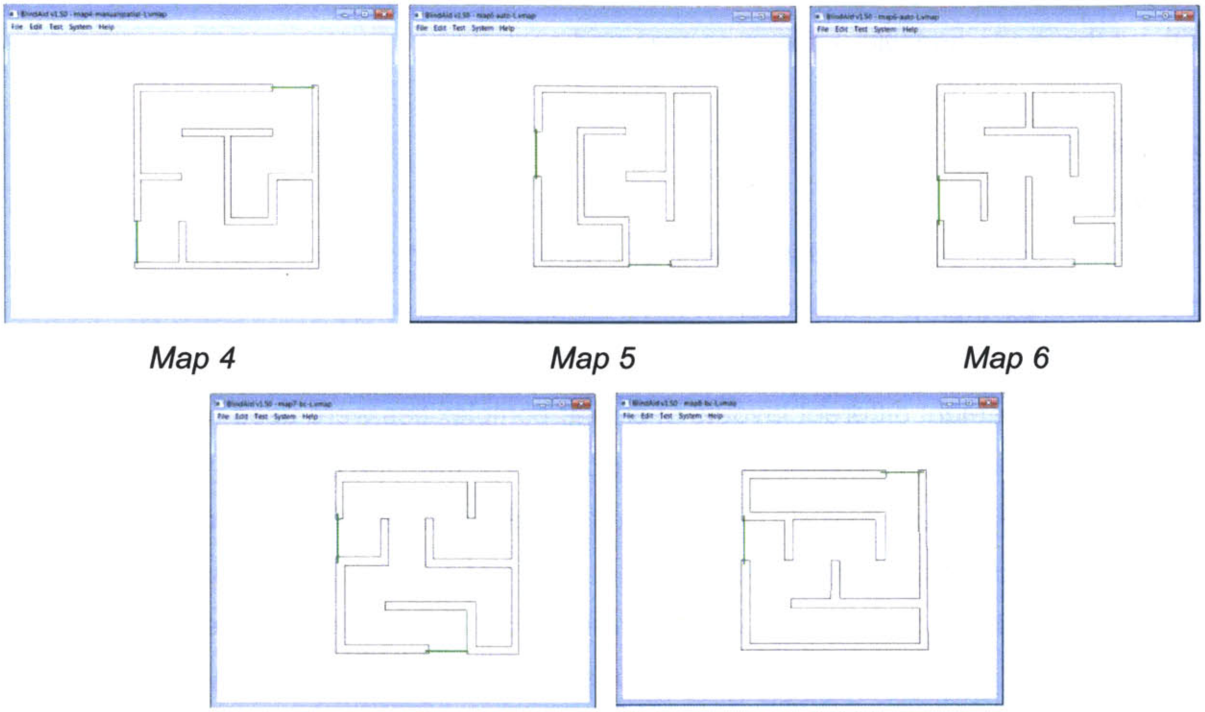
Map 1



Map 2



Map 3



Map 4 *Map 5* *Map 6*

Map 7 *Map 8*

Figure 16. Audio guide experiment map layouts

Procedure

The procedure was identical to the one used for testing the haptic guides including the post experiment questionnaire (see Section 5.3.1). The nominal set of trials for the audio guide experiment is listed in Table 9.

Trial	Condition	Map
Training #1	Manual	1
Training #2	Manual Spatial	2
Training #3	Automatic	3
Training #4	Landmarks	4
Test #1	Manual	5
Test #2	Manual Spatial	6
Test #3	Automatic	7
Test #4	Landmarks	8

Table 9. Nominal set of trials for audio guide experiment

Variables

The dependent variables in this test were: preference for the guide (calculation described in Section 5.3.1), navigation error (the number of times the subject entered a dead-end while navigating without the guide), navigation time (the time needed to navigate through the maze without the guide) and exploration time (the time needed to learn the route and space). See Table 10.

Independent Variable	Dependent Variables	Controlled Variables	
Audio guide type	Preference for guide	Size of maze	Number of dead-ends
	Navigation error	Length of route	Length of dead-end
	Navigation time	Number of turns	Number of doors
	Exploration time		

Table 10. Audio guide experiment variables

5.5.2 RESULTS / DISCUSSION

The following figures (17, 18) present the mean results averaged across all subjects, with lower values corresponding to better performance (error bars in the figure are ± 1 standard deviation). The complete set of data is included in Appendix B.4. User preference was similar across the manual, spatialized manual and automatic, but the preference for the landmark guide was noticeably worse (Figure 17). The difference in user preference was statistically significant with a P-value of 0.04 based on an independent group ANOVA test. Such an analysis can indicate if there is difference in the averages across a group but not between the specific variables. That being said, the graphical representation of the mean preference shows almost no overlap in rank values between the landmark guide and the other guides. The exploration time data presented a similar story. Navigation times were similar across all of the guides. Navigation errors occurred rarely, and only in the spatialized manual and automatic guide conditions (Figure 18).

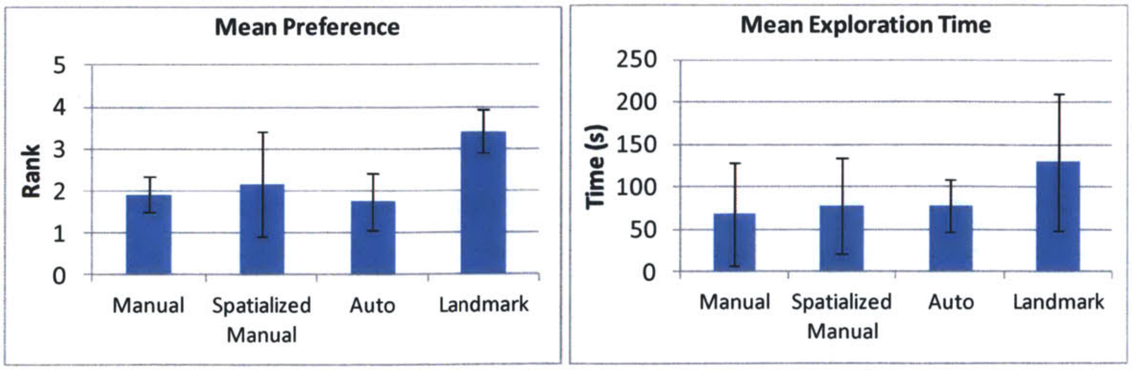


Figure 17. Mean preference and exploration time of audio guides

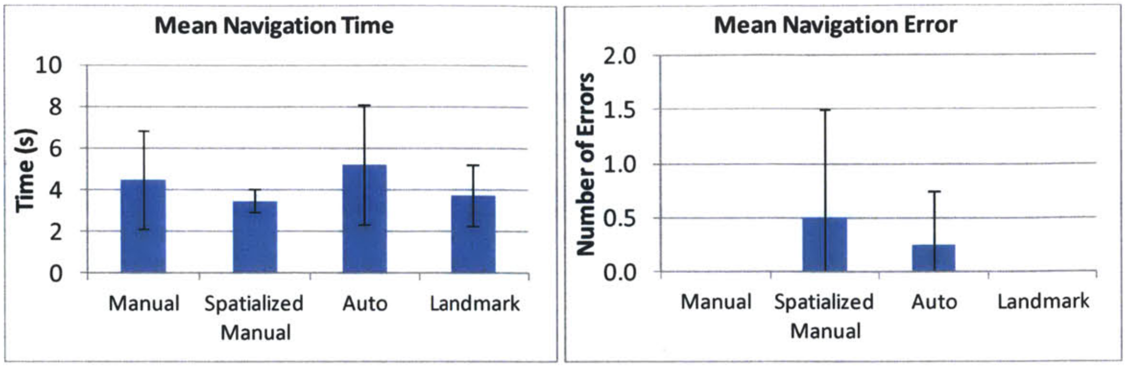


Figure 18. Mean navigation time and error of haptic guides

All four users expressed distaste for the landmark guide as it gave the least amount of assistance. Similar to the haptic guide experiment, the automatic audio guide was thought to have provided too much assistance and thus hindered learning, while the manual guide was desirable because it allowed the user to digest the information and learn at his/her own pace. The difference between the manual and manual spatial guide was seen as negligible by most users.

The lack of differentiation in navigation performance across the guides may be related to the ease of the task; thus for the main experiment, the task was made more challenging. A case could be made to choose any of the guides save the landmark guide; however, we chose to use the manual guide in the main experiment because it produced no navigation errors and matched the guide of choice in the haptic guide experiment, potentially minimizing confounding factors.

5.6 MAIN HAPTIC ROUTE GUIDANCE EXPERIMENT - INITIAL

The main experiment focused on determining the usability of a haptic guide. In this experiment, a haptic guide was compared with an audio guide, which represented the current standard, and free exploration (i.e., no guide) as the control condition. Specifically, the manual haptic and audio guides used were chosen based on the results of the previous two experiments. The haptic guide consisted of an attractive force that anchored the user to a manually selected way point along a snap-to guide line (located in the center of the grid squares). The audio guide consisted of way points and off-route locations that announced themselves automatically when entered and allowed the user to play instructions to the prior or next way point on command. In free exploration, the user was given no assistance in learning the route. This experiment compared the effectiveness of using a haptic guide, an audio guide, and free exploration for route learning.

5.6.1 METHODS

Subjects

All four subjects in the initial main experiment were right-handed males. The subjects ranged in age from 23 to 28 and all had corrected or corrected normal vision, and normal hearing and sense of touch. None of the subjects had participated in any of the prior tests. See Table 11.

Subject ID	Gender	Age	Height	Weight (lb)	Profession	Hand	Vision	Hearing	Touch
8	M	25	5'9"	171	Student	Right	Glasses, Astigmatism	Normal	Normal
9	M	28	5'4"	140	Software Developer	Right	Astigmatism	Normal	Normal
10	M	25	6'3"	187	Student	Right	Normal	Normal	Normal
11	M	23	5'9"	155	Student	Right	Normal	Normal	Normal

Table 11. Initial main experiment subjects

Arrangement

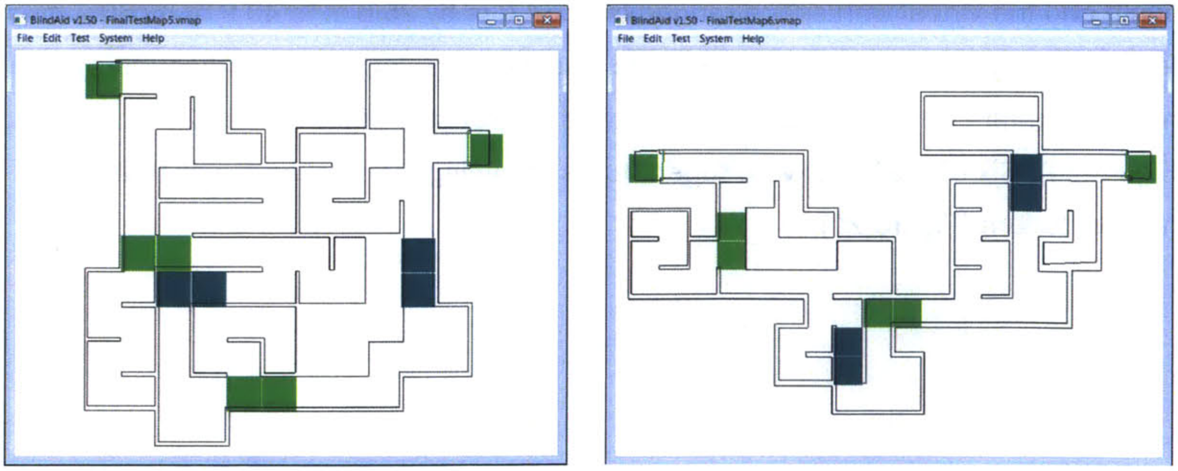
A total of six layouts of similar complexities were used (Figure 19). Each map was a closed maze with only one complete route from the entrance to the exit; the forward direction of the route was always left to right.). Each grid square was 2.1 m in length and the virtual workspace ranged from 34.0 m by 25.6 m to 40.9 m by 30.7 m, depending on the configuration. The route was the same length (34 grid squares), had the same number of turns (12 turns), and was constructed of the same 12 pieces (Figure 20). Six of the turns were "open" turns, meaning they lacked a wall that forced the user to turn (Figure 21).

Map 1

Map 2

Map 3

Map 4



Map 5

Map 6

Figure 19. Main experiment map layouts

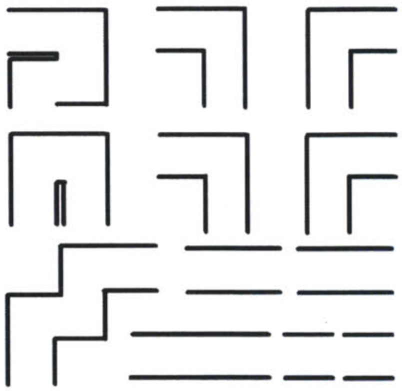


Figure 20. Route components

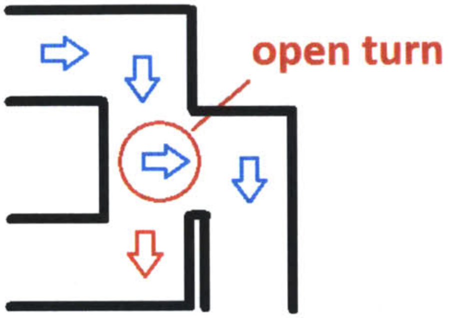


Figure 21. Example of an open turn. Red arrow indicates a wrong turn.

There were six dead-ends that were identical in each map (Figure 22); three of which were connected to the route via "open" turns and three which were connected to the side of the route (chosen at random). There were also three dead-end 2x2 rooms that were connected to the route via "open" turns – two rooms were created using two units from the route (Room A) and one room used no route units (Room B) (Figure 23).

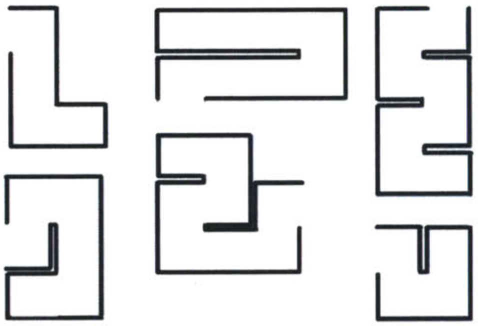
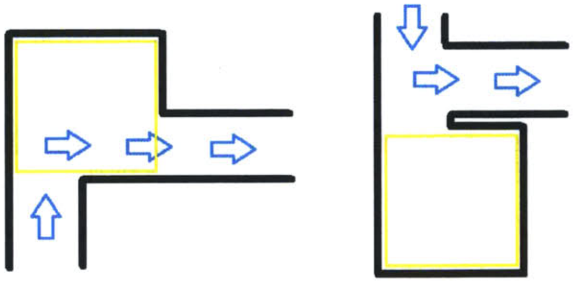


Figure 22. Dead-end components



Room A

Room B

Figure 23. Open turn examples

Additionally, there were four reference areas that were each two units long. The reference areas acted as the 1/5, 2/5, 3/5 and 4/5 markers of route (plus or minus two units). In the initial four trials of the main experiment, the first marker was a cafeteria sound, the second marker was a soft texture, the third was a chirping sound, and the forth was a rougher texture; one sound and one texture were always placed at way points. The second marker was a smooth texture with 0.0 mm peak amplitude, 160.0 mm period, 0.5 up/period ratio, and 0.0 deadband/period ratio. The fourth marker was a sinusoidal texture with 0. 3 mm peak amplitude, 4.0 mm period, 0.5 up/period ratio, and 0.0 deadband/period ratio (Schloerb, Lahav, Desloge, & Srinivasan, 2010).

Procedure

The procedure was identical to the one used for testing the haptic guides including the post experiment questionnaire, except for the number of conditions & maps (see Section 5.3.1). The nominal set of trials for the main experiment is listed in Table 12.

Trial	Condition	Map
Training #1	None	1
Training #2	Audio	2
Training #3	Haptic	3
Test #1	None	4
Test #2	Audio	5
Test #3	Haptic	6

Table 12. Nominal set of trials for main experiment

Variables

The dependent variables in this test were: preference for the guide, navigation error, navigation time, and exploration time. See Table 13. The calculation for the guide preference is described in Section 5.3.1 and the details for all other dependent variables are presented in Section 5.5.1.

Independent Variable	Dependent Variables	Controlled Variables	
Guide agent type	Preference for guide	Length of route	Type of route segments
	Navigation error	Type of route segments	Number of rooms
	Navigation time	Number of doors	Type of rooms
	Exploration time	Number of turns	Number of reference areas
		Number of "open" turns	Type of reference areas
		Number of dead-ends	Placement of reference areas
		Type of dead-ends	

Table 13. Main experiment variables

5.6.2 RESULTS / DISCUSSION

As in the previous experiments, the following figures present the mean results averaged across all subjects, with lower values corresponding to better performance (error bars in the figure are ± 1 standard deviation). The complete set of data is included in Appendix B.5. Figure 24 presents the mean user preference for each guide, showing the audio guide was the most favored and the haptic guide least favored. If we take the null hypothesis to be that the preferences across the three guides were the same, we get a P-value of 0.0002. This gives strong support for the alternate hypothesis that the preferences are unique. If we drill down further and do a pair-wise comparison of the haptic guide to the audio guide and the haptic guide to no guide and apply the Bonferonni adjustment (since multiple t-tests increase the chance of finding an incorrect significance), we get P-values of 0.0002 and 0.04, respectively. This means that the finding that the haptic guide is the least preferred is statistically significant. The mean exploration time data (Figure 24) also suggests that it was harder to learn to use the haptic guide. Although the mean navigation performance was similar across all conditions (Figure 25).

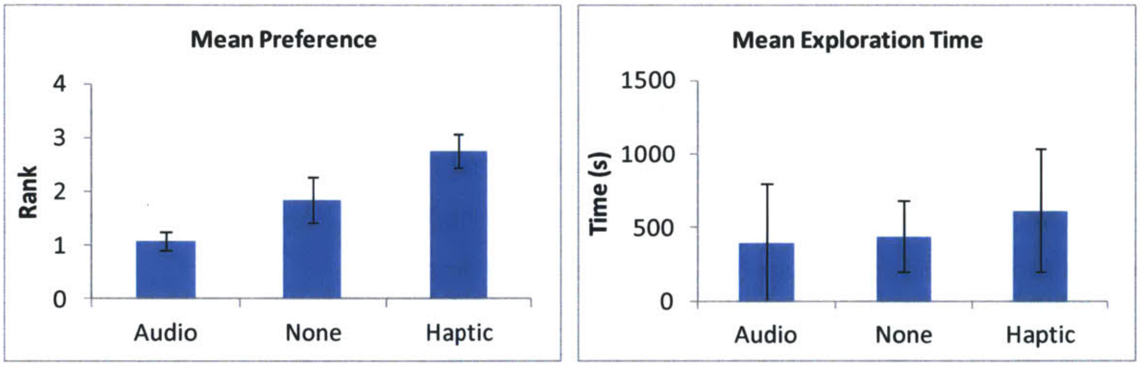


Figure 24. Mean preference and exploration time of route guides by type

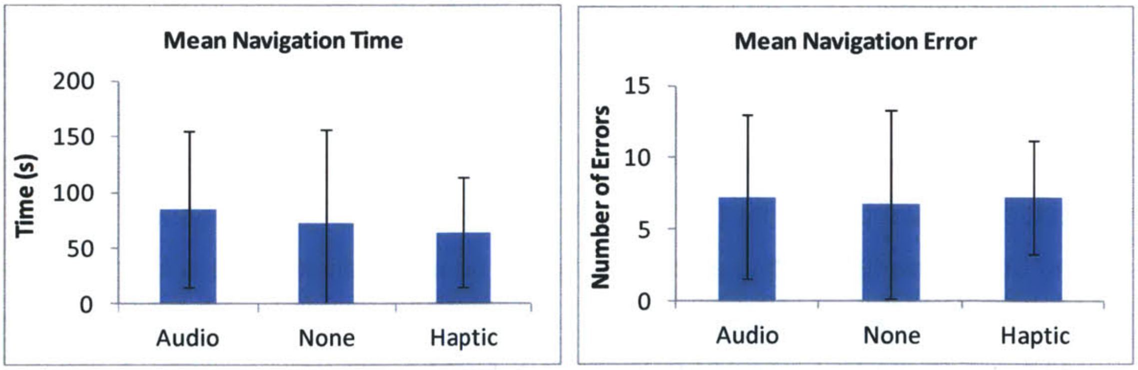


Figure 25. Mean navigation time and error of route guides by type

Based on observations and comments by subjects, issues with using the haptic guide included:

- Guide lines could be confused for walls.
- It was easy to get lost in the dead-end segments.
- Once lost, the anchor continued to pull in a direction that was not intuitive.

- The anchor could also continue to move away from the user if the user presses the forward or backward keys in an attempt to regain his/her bearings.
- The guide could pull too quickly and forcefully, making it difficult to remember the path.
- When the way points were closer together, it was difficult to tell that they were separate points.

5.6.3 REVISED HAPTIC GUIDE DESIGN/IMPLEMENTATION

The first four subjects struggled with the haptic guide and found that it hindered their ability to learn the route. When compared with other haptic guides, the manual haptic guide was well-received and thought to be very helpful. However, when compared with a more standard audio guide, the haptic guide fell short. Thus, we decided to stop the experiment and re-think the design of the haptic guide.

The first question we asked was "What information do I need when navigating?" We need to be able to answer two questions at all times:

1. Am I on or off the route?
2. How do I get to where I need to go? If on the route, what is my next step? If off the route, how do I get back to the route?

The current implementation of the haptic guide could only give vague answers to these questions. Open space along a guide line (on the route) could easily be confused for open space in a dead-end. Guide lines were intended to indicate the route; however, at times, guide lines were confused for walls. It was also easy to get lost behind a wall of a dead-end but still be tethered to a point that gave no information about how to exit the dead-end. Furthermore, frustrated users would press the arrow keys in hopes of finding the anchor, only to have it move further and further away. Keeping the most important navigation questions in mind, we proceeded in re-designing the guide.

We removed the guide lines altogether since it seemed that this feature only caused confusion. The tether was also removed because it became increasingly clear that the freedom to explore was extremely important for learning and retaining information. In the new design of the guide, the user is able to move freely along the path without being pulled unless he/she specifically asks for directions by pressing the left or right key. Users receive directions in the form of a gentle constant force (nudge) in the right direction. The nudge instruction is always dependent of the proxy location since having access to way points further along the path has little value. Pressing the left or right key pushes the user in the direction of the previous or next way point while on the route. The force is active for as long as the user holds the key down, so a user could use this capability to get an overview of the entire route. The force strength is constant as a distance-dependent one was found to be confusing. The default force strength is 0.4 N but can be increased to 0.5 N or decreased to 0.3 N using the up and down arrow keys. Texture is used to denote dead-ends, providing immediate haptic feedback about whether the user is on or off the route. Finally, pressing the left or right key while off-route nudges the user in the direction of the exit to the route from the dead-end. Based on this version of the guide, we conducted ten trials of the main experiment.

5.7 MAIN HAPTIC ROUTE GUIDANCE EXPERIMENT - FINAL

The main experiment was repeated with the revised haptic guide using the same methods as in the initial experiment.

5.7.1 SUBJECTS

There were six female and four male subjects ranging between the ages of 23 and 28. Three were left-handed and seven were right-handed (Table 14). All subjects had corrected or corrected normal vision and normal hearing and sense of touch. None of the subjects had participated in any of the prior tests.

Subject ID	Gender	Age	Height	Weight (lb)	Profession	Hand	Vision	Hearing	Touch
12	F	24	5'3"	115	Student	Right	Glasses	Normal	Normal
13	F	28	5'9"	125	Mech Engineer	Right	Contacts	Normal	Normal
14	F	23	5'1"	105	Student	Right	Contacts	Normal	Normal
15	M	24	6'3"	193	Student	Right	Contacts	Normal	Normal
16	F	25	5'4"	140	Student	Right	Contacts	Normal	Normal
17	M	23	5'8"	155	Student	Left	Glasses	Normal	Normal
18	F	27	5'6"	125	Student	Right	Normal	Normal	Normal
19	M	25	5'11"	155	Student	Right	Normal	Normal	Normal
20	F	28	5'7"	115	Student	Left	Contacts	Normal	Normal
21	M	28	5'10"	145	Student	Left	Glasses	Normal	Normal

Table 14. Final main experiment subjects

Arrangement

The map arrangements were identical to those used in the initial main experiment, except the texture reference areas were replaced by sounds (the “soft” texture by a playground sound and the “rough” texture by an ocean sound) and the “soft” texture defined in Section 5.7.1 was repurposed to represent off-route areas.

Procedure

The procedure was identical to the one used for testing the haptic guides including the post experiment questionnaire. The nominal set of trials for the main experiment is shown in Table 12.

Variables

The dependent variables are also identical to the initial main test. See Table 13.

5.7.2 RESULTS / DISCUSSION

Figure 26 presents the mean preference and the mean times for exploring (learning) the route for each of the experimental conditions. As in previous sections, the figure presents the mean results averaged across all subjects, with lower values corresponding to better performance (error bars in the figure are ± 1 standard deviation). The complete set of data is included in Appendix B.6.

On average, subjects found each guide to be equally satisfactory. In other words, qualitatively, the haptic guide was as well received as the audio guide, which we designed to approximate the type of guides that are currently in use. Also, the exploration times were approximately equal across the three guides, with slightly less time spent using the haptic guide. A single factor ANOVA test produced a P-value of 0.61, providing support for the null hypothesis: the time needed to learn the route using the three guides is equal.

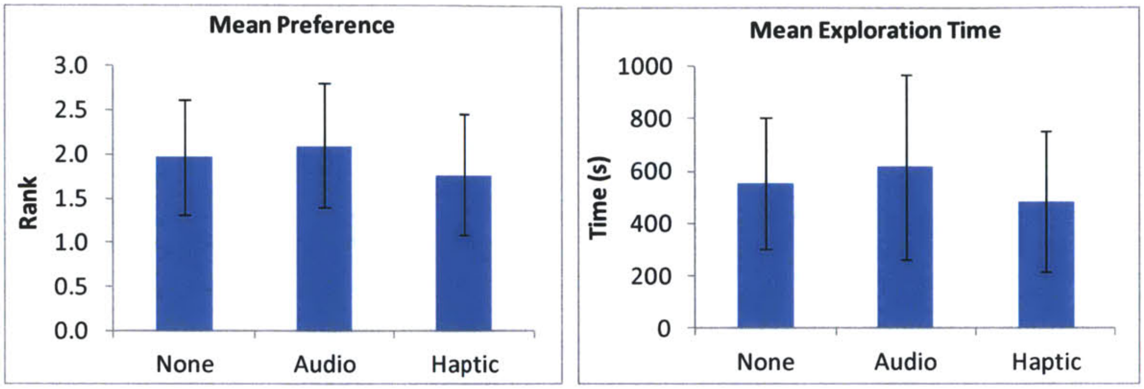


Figure 26. Mean preference and exploration time of guides

It was initially surprising to see that having no guide was equally favored in preference and that the no guide exploration time was essentially the same as with a guide. Subjects liked the no guide condition because it typically resulted in good retention of the layout – there was no confusion on whether they knew the map or not. However, it is important to note that in a closed maze like the ones designed for this experiment, the no guide case works well because the person can eventually find his/her way out. However, real maps are much more complex and typically continuous like the landscapes they model, making it much more difficult, if not impossible, to find a destination without any guidance. For experimental purposes, it was important to create closed maps that subjects could finish to keep the exploration times within reason.

Figure 27 presents the mean navigation times and errors for each of the experimental conditions, averaged across all subjects, with lower values corresponding to better performance. To simplify the discussion, the navigation error will not be considered further because it was strongly correlated to the navigation times, with a correlation coefficient of 0.97 (Figure 28). A correlation coefficient of 1 indicates that the two data sets are perfectly positively correlated. Navigation times were more variable with the haptic guide, requiring the most amount of time and having the greatest standard deviation. The no guide condition had the fastest navigation time; this result was not surprising since learning with less help typically results in the better retention of information. That being said, the ANOVA test still supported the null hypothesis that the navigation times were not different, with a P-value of 0.08.

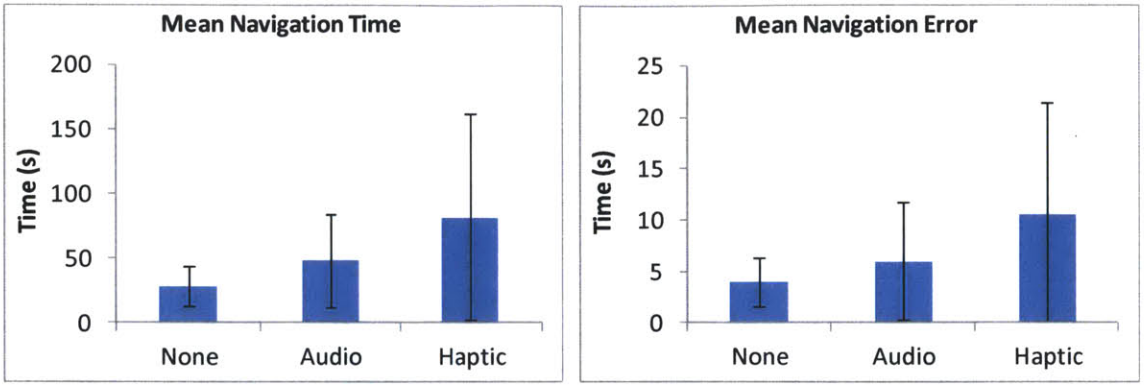


Figure 27. Mean navigation time and error of guides

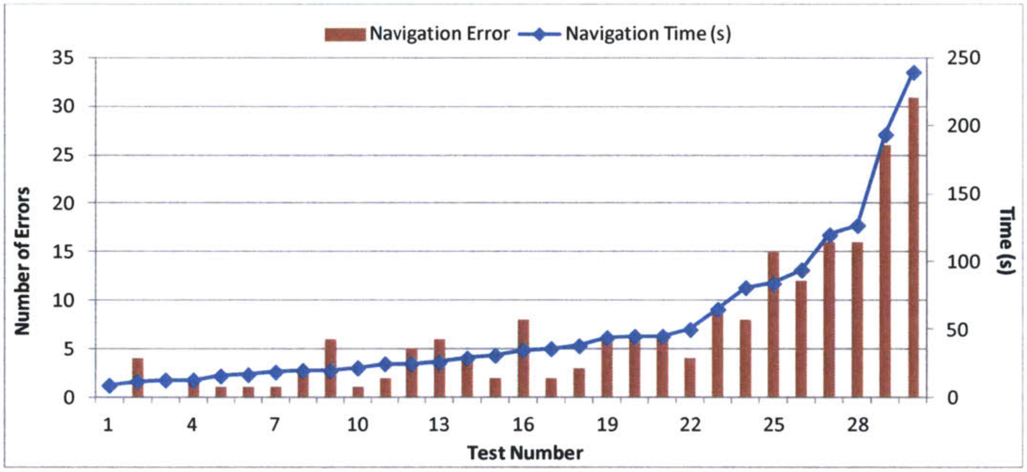


Figure 28. Correlation between navigation time and error

5.7.3 OUTLIER ANALYSIS

Closer analysis of the data (Figure 29) shows one data point in the exploration time dataset and two data points in the navigation time dataset that could be statistical outliers (greater than three standard deviations from the mean). Note that the histograms in Figure 29 sum the data for all of the subjects.

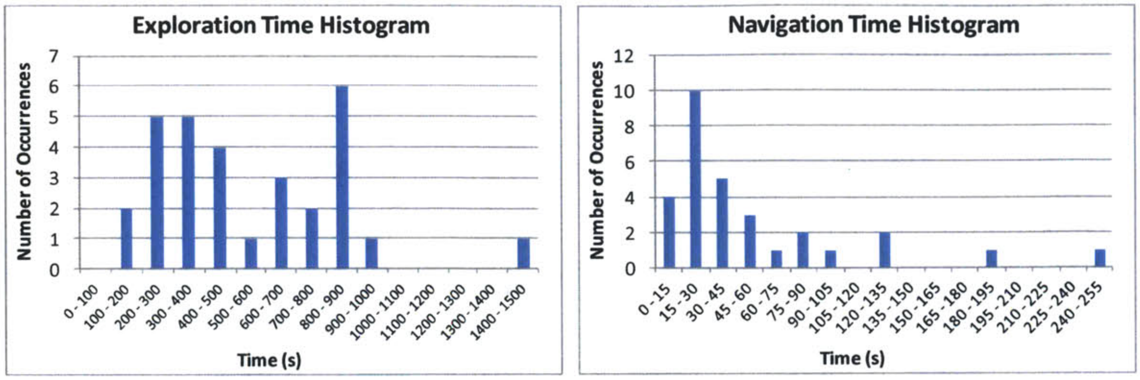


Figure 29. Exploration and navigation time histograms

Without the three outliers, the mean exploration time for the audio guide increases to a level roughly equal to the mean haptic exploration time, and the mean haptic navigation time decreases to a level below that of the audio guide (Figure 30). Additionally, the alternate hypothesis that the times are not different becomes even less likely (P-value for exploration time = 0.82; P-value for navigation time = 0.31).

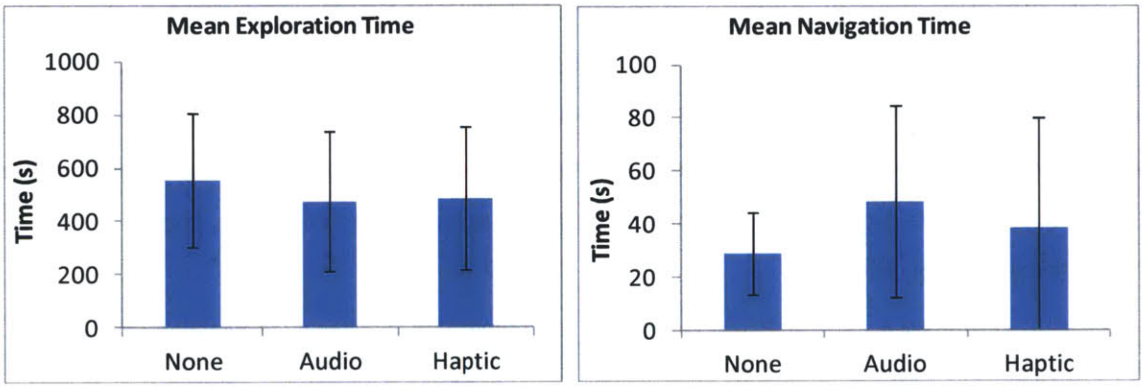


Figure 30. Mean exploration and navigation time less outliers

5.7.4 DISCUSSION OF LEARNING CURVE

All ten subjects had no experience with haptics so the learning curve, which could make the results of the last test more favorable than the first, could skew the data. The experimental procedure was configured to help avoid this issue, and the results of experiment suggest that the learning curve did not pose a problem as can be seen in Figure 31. The figure presents the mean exploration and navigation times averaged across all subjects and conditions (including the outliers), for each of the three tests. The results from the first test (#1) in the experiment are essentially the same as the results for second and third tests (#2 & #3). Given that the conditions were distributed evenly between the tests, this means that on average, performance did not improve over time.

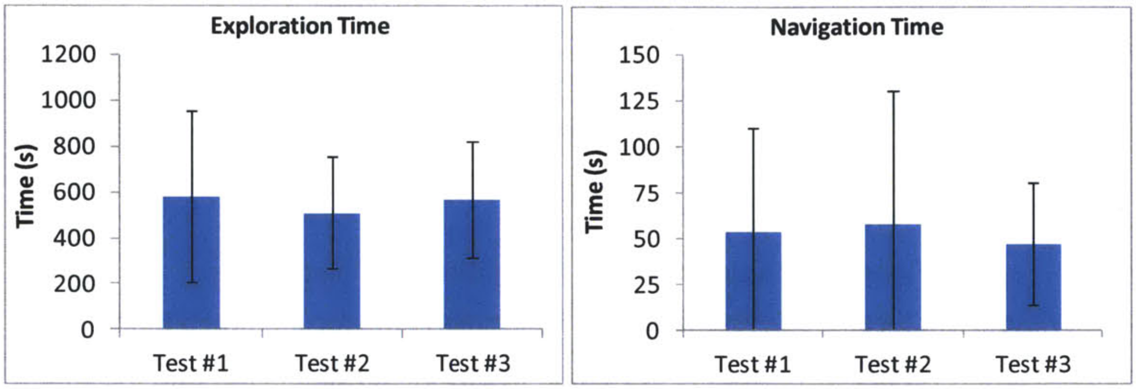


Figure 31. Mean exploration and navigation time of tests

5.7.5 DISCUSSION OF PERFORMANCE BASED ON PREFERENCE

Preference for a guide may be related to the user's proficiency with it. If we separate the exploration and navigation times by preference, we see that a correlation exists between preference and exploration time but not necessarily navigation time (Figure 32 – 34). Five subjects ranked the haptic guide the highest, two subjects ranked the audio guide the highest, two subjects ranked no guide the highest, and one subject ranked them all equally.

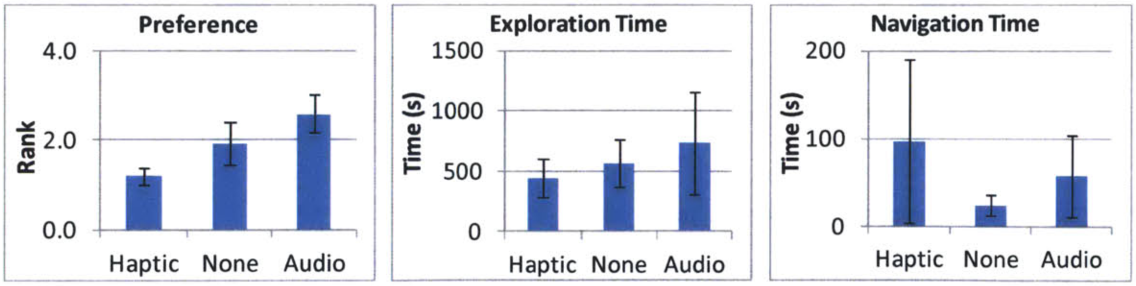


Figure 32. Mean results averaged across the subjects who preferred the haptic guide

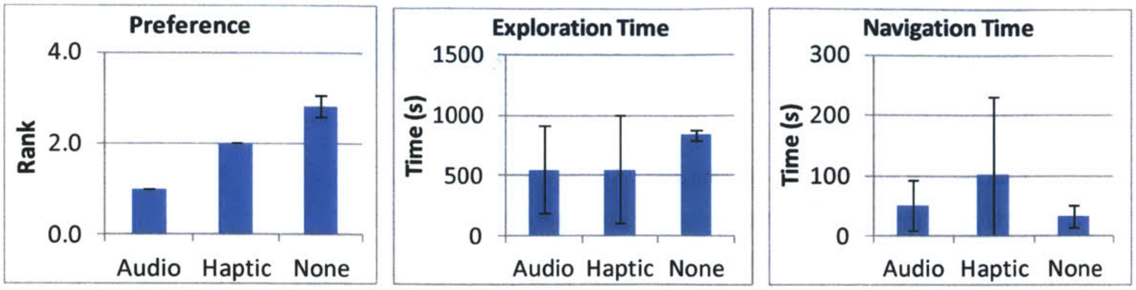


Figure 33. Mean results averaged across the subjects who preferred the audio guide

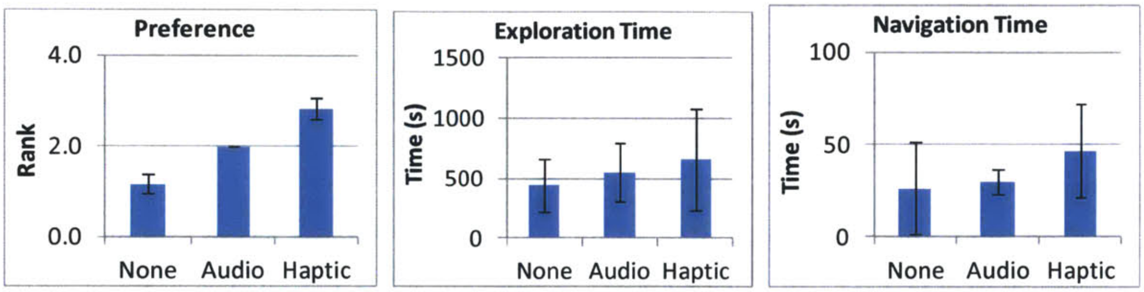


Figure 34. Mean results averaged across the subjects who preferred no guide

5.7.6 DISCUSSION OF SUBJECT COMMENTARY

General Strategy

Every subject implemented a different set of strategies; however, there were several prominent strategies (Table 15). Most users relied heavily on the reference areas, since it was the only tool that carried over to the navigation task. Some users stuck close to the walls to guide their turns. Another popular strategy was to segment the route into multiple pieces and learn each segment separately. After realizing that the guide would not be available in the navigation task, many users learned to use the guide as a supplemental tool to avoid over dependence on it.

General Strategy	No. of Subjects
Used reference areas	9
Followed the walls	6
Learned in segments by retracing steps	6
Explored areas around reference sounds to determine the next step	5
Tried not to rely on the guide too much	5
Learned to switch to the other side of the path to avoid dead-ends	4
Got an overview of the route using the guide	3
Verbalized instructions	3
Used restart option often when lost	3

Table 15. General strategy

Haptic Guide

The haptic guide consists of textures that denote off-route areas and an instruction tool that uses a constant force to push users in the correct direction when activated. Table 16 illustrates the perceived positives and negatives of the feature. The ability to receive instructions in the form of a gentle push was thought to be helpful and easy to understand, however, it was not used very often. Most subjects relied heavily on the textures and only asked for instructions when lost (Table 17). The reception of textures was mixed. Half of the subjects thought that they were intuitive in indicating areas to avoid and were able to use them effectively. The other half thought that textures were confusing and difficult to use because textures made traveling in the learning stage easy, allowing the user to become less active in his/her learning. This issue – that the guide is helpful until it is removed – is one that affects both the haptic and audio guides. However, subjects that were able to moderate their dependence on the guide found that the additional information provided by the guide was very helpful. This result is important because it highlights the need for ensuring that the user has truly learned the route when using a guide agent.

Pros	No. of Subjects	Cons	No. of Subjects
Nudge instruction was helpful because the direction was easy to follow and it reinforced the path	7	Guide was helpful but felt lost when guide was removed	5
Textures were intuitive and gave clear indication to turn away	6	Didn't like texture because it was easy to become inactive in learning and at times confusing	4
Ability to toggle the texture on and off was helpful	1	Nudge instruction was difficult to understand	1

Table 16. Pros and cons of the haptic guide

Haptic Guide Strategy	No. of Subjects
Instructions	
Didn't ask for instructions often, relied mostly on the textures	7
Asked for instructions primarily when lost	5
Asked for instructions often	1
Instruction Keys	
Tapped the instruction key	6
Held instruction key down to learn longer segments of the route	4
Textures	
Turned around when came in contact with texture to avoid dead-end	6
Toggled texture on and off often	2

Table 17. Strategies used with the haptic guide

Audio Guide

The audio guide consists of route way points and off-route points that can announce themselves automatically and an instruction tool that provides verbal directions on command. The sequential ordering of the way points was thought to be logical and useful in learning (Table 18). The other aspects of the guide returned mixed reviews. Some users thought the auto-announce feature was very helpful; others thought it was distracting and discouraging, particularly when in the off-route areas. Some users believed having to translate the verbal instructions into physical motion was unnecessarily time-consuming, while others found it useful for remembering the path.

Like the haptic guide, most users did not ask for instructions often and instead relied on the auto-announce feature (Table 19).

Pros	No. of Subjects	Cons	No. of Subjects
Sequential ordering of way points was logical and made learning easier	3	Guide was helpful but felt lost when it was removed	6
Could use the guide as a supplement so removing it was not an issue	2	Disliked the off-route feeback	4
Liked the immediate off-route feedback	1	Translating the audio was time-consuming	3
Translating the audio helped with remembering the route	1	Too many sounds to keep track of	2
		Didn't like the auto-announce feature	1

Table 18. Pros and cons of the audio guide

Audio Guide Strategy	No. of Subjects
Instructions	
Didn't ask for instructions often, relied mostly on the auto-announce feature	8
Used instructions often	2
Auto-announce	
Toggled the auto-announce feature on and off often	6
Disliked the auto-announce feature	1

Table 19. Strategies used with the audio guide

No Guide

While having no guide was thought to be the most difficult to learn with initially, it often gave users a greater sense of confidence in their understanding of the route (Table 20). Not having a guide agent in a real life implementation of BlindAid is not realistic, so it is important to be able to instill the same level confidence in the user's understanding of the route while using a guide. One way to achieve this is to wean the user off of using the guide agent as he/she learns the route.

Pros	No. of Subjects	Cons	No. of Subjects
Gave me the most confidence since I had to learn it on my own	4	Didn't give as much information about the route and off path areas as the guides	3
Was easy to use	2	Difficult to learn the route	2

Table 20. Pros and cons of no guide

6 CONCLUSIONS

This thesis focuses on improving global sensing for haptics through the use of force fields. The application of repulsive and attractive force fields to objects in the VE was studied through a total of six experiments.

Repulsive Force Fields for Global Haptic Sensing

Repulsive force fields were designed to enable global sensing to help users learn about the environment more quickly. We experimented with using repulsive force fields in both indoor and outdoor spaces; however, they were unable to produce an improvement in navigation ability in either environment. In both experiments, the times to explore and acquire the desired information with the force field and without were essentially equal. On occasion, the repulsive force field required more exploration time and resulted in additional errors. There are a few important conclusions to derive from these experiments.

First, it is difficult to ground oneself using variable surfaces. While the purpose of the force field was to provide information with less exploration, the user first needed to know where he/she was before processing any additional information. Hence, we observed subjects pushing through the force fields in an effort to gain their bearings using the walls. This leads us to the second point - knowing that an object exists is only useful if you know where you are currently. Additionally, applying a repulsive force field to an object of interest may not be ideal, since the repulsive force pushes the user away if he/she tries to approach the object. Next, sensing an object rather than actually touching it can be challenging. This method of learning about an object may cause navigation errors because it is more difficult to gauge the object's exact location. Furthermore, subjects may need to spend more time translating and verifying the locations of objects (e.g., moving past the object multiple times or seeking it out to physically touch it).

Attractive Force Fields for Global Haptic Sensing Applied to Route Guidance

The attractive force field is another tool that was designed to improve VE exploration. In this thesis, we focused specifically on using attractive force fields for route guidance. Previous attempts at using haptics for route guidance (by actively leading the user) had proved unsuccessful, so our goal was to improve global sensing during route learning.

In the first developmental experiment, we evaluated four potential haptic guide designs and selected the most promising one. Our intention was to compare the most promising haptic guide to conventional alternatives (using no guide and an audio guide) in a main experiment. Similar to the haptic guide development, we designed and evaluated four audio guides and selected the best one. In both developmental experiments, the manual guide was deemed the most favorable. The haptic manual guide consisted of an attractive force that anchored the user to a manually selected way point along a snap-to guide line. The audio manual guide comprised of way points and off-route locations that announced themselves automatically and allowed the user to play instructions to the prior or next way point.

The manual guides were preferable because they allowed users to move at their own pace and test their knowledge by turning the guides off as needed. While these aspects helped to improve map retention, all of the guides were susceptible to incomplete learning due to an over reliance on the feature. On the other hand, subjects that were able to moderate their dependence on the guide found that the additional information provided by the guides was beneficial to learning and did not hinder retention. This was one of the most important findings.

The main experiment compared using the best haptic guide and audio guide to using no guide. Despite strong reception for the manual haptic guide in the first experiment, the haptic guide fared poorly when compared to a more standard audio guide in the initial main experiment. Thus, we re-designed the haptic guide to address the issues it faced. In the final main experiment, we found that all three guides were equally favorable in terms of user preference and performance. This meant that the haptic guide was as well received as the audio guide, which was designed to approximate the type of guides that are currently in use. While it was surprising to find that the no guide condition performed as well as the audio and haptic guides, it is important to note that in a closed maze like the ones designed for this experiment, the no guide case works because the person can eventually find his/her way out. Based on user preference and performance, the haptic guide is as capable and useful as the current standard of route guidance. The results of this experiment validate its use as a guide agent in virtual environments.

REFERENCES

Abdulrasool, D., & Sabra, S. (2011). Mobile-Embedded Smart Guide for the Blind. Paper presented at the DICTAP '11, International Conference on Digital Information and Communication Technology and Its Applications, Dijon, France.

Akhter, S., Mirsalahuddin, J., Marquina, F. B., Islam, S., & Sareen, S. (2011). A Smartphone-Based Haptic Vision Substitution System for the Blind. Paper presented at the 2011 IEEE 37th Annual Northeast Bioengineering Conference (NEBEC), Troy, NY.

Blenkhorn, P., & Evans, D. G. (1997). A System for Enabling Blind People to Identify Landmarks: The Sound Buoy. IEEE Transactions on Rehabilitation Engineering, 5(3), 276-278. doi: 10.1109/86.623019

Blindness. (n.d.). Wikipedia. Retrieved April 7, 2013, from http://en.wikipedia.org/wiki/Blindness

Borenstein, J. (n.d.). GuideCane by Johann Borenstein. Retrieved October 27, 2010, from http://www.arrickrobotics.com/robomenu/guidecan.html

D'Atri, E., Medaglia, C. M., Serbanati, A., Ceipidor, U. B., Panizzi, E., & D'Atri, A. (2007). A System to Aid Blind People in the Mobility: A Usability Test and its Results. Paper presented at the Second International Conference on Systems (ICONS'07), Sainte-Luce, Martinique, France.

Espinosa, M. A., & Ochaita, E. (1998). Using tactile maps to improve the practical spatial knowledge of adults who are blind. *Journal of Visual Impairment and Blindness, 92*(5), 338-345.

Farrell, M. J., Arnold, P., Pettifer, S., Adams, J., Graham, T., & MacManamon, M. (2003). Transfer of Route Learning From Virtual to Real Environments. *Journal of Experimental Psychology: Applied, 9*(4), 219-227. doi: doi:10.1037/1076-898X.9.4.219

Feintuch, U., Haj, J., & Weiss, P. L. (2006). *Low-cost Haptic-aural Virtual Environments for Way-finding by Children with Congenital Blindness: Feasibility Study.* Paper presented at the International Workshop on Virtual Rehabilitation, 2006 New York, NY.

Gallo, S., Chapuis, D., Santos-Carreras, L., Kim, Y., Retornaz, P., Bleuler, H., & Gassert, R. (2010). Augmented White Cane with Multimodal Haptic Feedback. Paper presented at the 2010 3rd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics, Tokyo, Japan.

Giudice, N. A., Bakdash, J. Z., Legge, G. E., & Roy, R. (2010). Spatial Learning and Navigation Using A Virtual Verbal Display. ACM Transactions on Applied Perception, 7(1), 3:1-3:22. doi: 10.1145/1658349.1658352

Golledge, R. G., Klatzky, R. L., & Loomis, J. M. (1996). Cognitive Mapping and Wayfinding by Adults Without Vision. In J. Portugali (Ed.), The Construction of Cognitive Maps (Vol. 32, pp. 215-246). Dordrecht, Netherlands: Kluwer Academic Publishers.

Hara, M., Shokur, S., Yamamoto, A., Higuchi, T., Gassert, R., & Bleuler, H. (2010). *Virtual Environment to Evaluate Multimodal Feedback Strategies for Augmented Navigation of the Visually Impaired.* Paper presented at the 32nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2010, Buenos Aires, Argentina.

Havik, E. M., Steyvers, F. J. J. M., van der Velde, H., Pinkster, J. C., & Kooijman, A. C. (2010). Design and Evaluation of a Protocol to Assess Electronic Travel Aids for Persons Who Are Visually Impaired. Journal of Visual Impairment & Blindness, 104(2), 84-94.

Heuten, W., Henze, N., & Boll, S. (2007). Interactive Exploration of City Maps with Auditory Torches. Paper presented at the CHI EA '07, CHI '07 Extended Abstracts on Human Factors in Computing Systems San Jose, CA.

Heuten, W., Henze, N., Boll, S., & Pielot, M. (2008). TactileWayfinder: A Non-Visual Support System for Wayfinding. Paper presented at the NordiCHI '08, 5th Nordic Conference on Human-computer Interaction: Building Bridges Lund, Sweden.

Julius. (2010). Electronic Mobility Devices for Persons Who are Blind or Visually Impaired. Retrieved December 10, 2010, from http://www.evengrounds.com/blog/electronic-mobility-devices-for-persons-who-are-blind-or-visually-impaired

JVIB News Service. (1995). Demographics Update: Alternative Estimate of the Number of Guide Dog Users. *Journal of Visual Impairment & Blindness, 89*(2), 4-6.

Kaklanis, N., Votis, K., Moschonas, P., & Tzovaras, D. (2011). *HapticRiaMaps: Towards Interactive exploration of Web World maps for the Visually Impaired.* Paper presented at the W4A '11, International CrossDisciplinary Conference on Web Accessibility, Hyderabad, India.

Klatzky, R. L., Loomis, J. M., Beall, A. C., Chance, S. S., & Golledge, R. G. (1998). Spatial Updating of Self-Position and Orientation During Real, Imagined, and Virtual Locomotion. Psychological Science, 9(4), 293-298. doi: 10.1111/1467-9280.00058

Kostopoulos, K., Moustakas, K., Tzovaras, D., & Nikolakis, G. (2007). Haptic Access to Conventional 2D Maps for the Visually Impaired *Journal on Multimodal User Interfaces, 1*(2), 13-19. doi: 10.1007/BF02910055

Lahav, O., Schloerb, D. W., Kumar, S., & Srinivasan, M. A. (2008). BlindAid: a Virtual Exploration Tool for People who are Blind. Annual Review of Cybertherapy and Telemedicine, 6, 126-132.

Landau, S., Wiener, W., Naghshineh, K., & Giusti, E. (2005). Creating Accessible Science Museums With User-Activated Environmental Audio Beacons (Ping!). Assistive Technology, 17(2), 133–143. doi: 10.1080/10400435.2005.10132103

Lecuyer, A., Mobuchon, P., Megard, C., Perret, J., Andriot, C., & Colinot, J. P. (2003, March 22-26). *HOMERE: a multimodal system for visually impaired people to explore virtual environments.* Paper presented at the IEEE Virtual Reality, Los Angeles, CA, USA.

Loomis, J. M., Golledge, R. G., Klatzky, R. L., & Marston, J. R. (2007). Assisting wayfinding in visually impaired travelers. In G. L. Allen (Ed.), Applied Spatial Cognition: From research to cognitive technology. Mahwah NJ: Lawrence Erlbaum Associates.

Loomis, J. M., Marston, J. R., Golledge, R. G., & Klatzky, R. L. (2005). Personal Guidance System for People with Visual Impairment: A Comparison of Spatial Displays for Route Guidance. Journal of Visual Impairment & Blindness, 99(4), 219-232.

Murai, Y., Tatsumi, H., & Miyakawa, M. (2007). *A Haptic Walk-Guide Simulator for the Visually Impaired.* Paper presented at the IEEE International Conference on Systems, Man and Cybernetics, 2007, Montréal, Québec, Canada.

Na, J. (2006). The Blind Interactive Guide System Using RFID-Based Indoor Positioning System. Paper presented at the ICCHP '06, Computers Helping People with Special Needs, Linz, Austria.

O'Connell, M., Lieberman, L. J., & Petersen, S. (2006). The Use of Tactile Modeling and Physical Guidance as Instructional Strategies in Physical Activity for Children Who Are Blind. *Journal of Visual Impairment & Blindness, 100*(8), 471-477.

O'Donnell, C. (2005). The Greatest Generation Meets Its Greatest Challenge: Vision Loss and Depression in Older Adults. *Journal of Visual Impairment & Blindness, 99*(4), 197-208.

Papadopoulos, K. S. (2004). A School Programme Contributes to the Environmental Knowledge of Blind People *British Journal of Visual Impairment, 22*(3), 101-104. doi: 10.1177/0264619604050046

Passini, R., Dupre, A., & Langlois, C. (1986). Spatial Mobility of the Visually Handicapped Active Person: A Descriptive Study. Journal of Visual Impairment and Blindness, 80(8), 904-907.

Péruch, P., Vercher, J.-L., & Gauthier, G. M. (1995). Acquisition of Spatial Knowledge Through Visual Exploration of Simulated Environments. Ecological Psychology, 7(1), 1-20. doi: 10.1207/s15326969eco0701_1

Picard, D., & Pry, R. (2009). Does Knowledge of Spatial Configuration in Adults with Visual Impairments Improve with Tactile Exposure to a Small-scale Model of

Their Urban Environment? *Journal of Visual Impairment & Blindness, 103*(4), 199-209.

Pradeep, V., Medioni, G., & Weiland, J. (2010, Aug. 31 2010-Sept. 4 2010). A Wearable System for the Visually Impaired. Paper presented at the 32nd Annual International Conference of the IEEE EMBS, Buenos Aires, Argentina.

Ray, C. T., Horvat, M., Williams, M., & Blasch, B. B. (2007). Clinical Assessment of Functional Movement in Adults with Visual Impairments. *Journal of Visual Impairment & Blindness, 101*(2), 108-113.

Roentgen, U. R., Gelderblom, G. J., Soede, M., & de Witte, L. P. (2009). The Impact of Electronic Mobility Devices for Persons Who Are Visually Impaired: A Systematic Review of Effects and Effectiveness. Journal of Visual Impairment & Blindness, 103(11), 743-753.

Ross, D. A., & Blasch, B. B. (2000). Wearable Interfaces for Orientation and Wayfinding. Paper presented at the ASSETS '00, Arlington, VA.

Russell, J. N., Hendershot, G. E., LeClere, F., Howie, L. J., & Adler, M. (1997). *Trends and Differential Use of Assistive Technology Devices: United States, 1994. Advance Data from Vital and Health Statistics of the Centers for Disease Control and Prevention; No. 292.* Hyattsville, Maryland: National Center for Health Statistics.

Sáenz, M., & Sánchez, J. (2010). Indoor Orientation and Mobility for Learners Who are Blind. Annual Review of Cybertherapy and Telemedicine 2010 - Advanced Technologies in Behavioral, Social and Neurosciences, 154, 131-134. doi: 10.3233/978-1-60750-561-7-165

Sánchez, J. (2009, July 19-24, 2009). Mobile Audio Navigation Interfaces for the Blind. Paper presented at the UAHCI '09, 5th International on Conference Universal Access in Human-Computer Interaction, San Diego, CA.

Sánchez, J., Tadres, A., Pascual-Leone, A., & Merabet, L. (2009, June 29 2009-July 2 2009). Blind Children Navigation through Gaming and Associated Brain Plasticity. Paper presented at the Virtual Rehabilitation International Conference, 2009 Haifa, Israel.

Schloerb, D. W., Lahav, O., Desloge, J. G., & Srinivasan, M. A. (2010, 25 - 26 March, 2010). *BlindAid: Virtual Environment System for Self-Reliant Trip Planning and Orientation and Mobility Training.* Paper presented at the IEEE Haptics Symposium 2010, Waltham, Massachusetts, USA.

Schmelter, A., Jansen-Osmann, P., & Heil, M. (2008). Empirical Evaluation of Virtual Environment Technology as an Experimental Tool in Developmental Spatial Cognition Research. *The European Journal of Cognitive Psychology, 21*(5), 724-739. doi: 10.1080/09541440802426465

Seki, Y., & Sato, T. (2011). A Training System of Orientation and Mobility for Blind People Using Acoustic Virtual Reality. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 19(1), 95-104 doi: 10.1109/TNSRE.2010.2064791

Sensable Technologies (2008). OpenHaptics® Toolkit API Reference Manual, v3.0 Sensable Technologies, Inc., Woburn, MA.

Shiizu, Y., Hirahara, Y., Yanashima, K., & Magatani, K. (2007, 22-26 Aug. 2007). The development of a white cane which navigates the visually impaired. Paper presented at the 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2007 Lyon, France.

Shoval, S., Ulrich, I., & Borenstein, J. (2003, Mar 2003). Robotics-Based Obstacle-Avoidance Systems for the Blind and Visually Impaired: NavBelt and the GuideCane. IEEE Robotics & Automation Magazine, 10, 9-20.

Ungar, S. (2000). Cognitive Mapping without Visual Experience. In R. Kitchin & S. Freundschuh (Eds.), *Cognitive Mapping: Past Present and Future*. London: Psychology Press.

Wang, Z., Li, B., Hedgpeth, T., & Haven, T. (2009, October 25-28, 2009). *Instant Tactile-Audio Map: Enabling Access to Digital Maps for People with Visual Impairment.* Paper presented at the ASSETS'09, Pittsburgh, Pennsylvania, USA.

White, G. R., Fitzpatrick, G., & McAllister, G. (2008). *Toward Accessible 3D Virtual Environments for the Blind and Visually Impaired.* Paper presented at the DIMEA '08, 3rd International Conference on Digital Interactive Media in Entertainment and Arts Athens, Greece.

Yu, K.-H., Yoon, M.-J., & Jeong, G.-Y. (2009). 3D Detection of Obstacle Distribution and Mapping for Tactile Stimulation. Paper presented at the IEEE International Conference on Mechatronics, 2009, Málaga, Spain.

APPENDIX A: PROGRAM CODE

A.1 NOTE ABOUT PROGRAM CODE ILLUSTRATIONS

The BlindAid program is written in C++. The program code makes use of a C++ structure called VECTOR_TYPE that consists of an x, y and z component. Thus to access these components we use a period: Point.x, Point.y and Point.z. This syntax will be used often in the explanation of the code in this thesis. Additionally, some of the variable names have been changed or the code has been simplified for illustrative purposes. Finally, some variable initializations are outside of the scope of the function.

A.2 COMMUNICATING WITH THE PHANTOM

The OpenHaptics API associated with the Phantom provides the framework for communicating with the Phantom via code. Accessing the real-time position of the proxy is made possible through a function provided by the API: hdGetDoublev(HD_CURRENT_POSITION, Proxy). The program code associated with finding the force point is shown below.

A.3 FORCE FIELD PROGRAM CODE

Repulsive Force

The program calculates the repulsive force based on the location of the force point.

```
//_______________________________________________________________ComputeForcefieldEffect
void HLCALLBACK ComputeForcefieldEffect(HDdouble force[3], HLcache *cache, void *userdata)
{
        //Typecast the pointer passed in appropriately
        vmap_forcefield* pvforcefield = (vmap_forcefield*)userdata;

        //Variables are set in the header of the vmap file
        HDdouble ForceMultiplier = pvforcefield->dForceMultiplier; //low multiplier = weaker force
        double MaxRadius = pvforcefield->dMaxRadius; //forcefield radius (range)

        //Get proxy position in physical workspace frame
        static HLdouble Pp[3]; //proxy in physical (phantom) workspace
        static VECTOR_TYPE Pw; //proxy in world workspace

        //Transform the proxy to worldspace keeping in vmap units
```

```
    hdGetDoublev(HD_CURRENT_POSITION, Pp);
    pvforcefield->PWTransformVmap(&Pw, Pp);

    //Compute the wall points from which forces should be calculated (called force points)
    VECTOR_TYPE forceP;
    pvforcefield->ComputeForcePoint(&forceP, Pw);

    //Calculate force between the force point and the proxy for each wall and aggregate the forces
    if (forceP.z != -1000000)           //But only do so if a valid force point exists
    {
        HLdouble forceVec[3] = {0,0,0}; //Vector to hold temporary force values

        //Calculate difference vector between the proxy and the force point
        HLdouble diffVec[3];
        diffVec[0] = forceP.x - Pw.x;
        diffVec[1] = forceP.y - Pw.y;
        diffVec[2] = forceP.z - Pw.z;

        //Find the magnitude of the distance
        double dist = sqrt(pow(diffVec[0], 2) + pow(diffVec[1], 2) + pow(diffVec[2], 2));

        //If the proxy is close enough to the force point, calculate a repulsive force
        if (dist <= MaxRadius)
        {
            for (int i=0;i<3;i++)
                {
                diffVec[i]=diffVec[i]/dist;         //Normalize vector
                forceVec[i] = -ForceMultiplier * diffVec[i] * (MaxRadius-dist)/MaxRadius;
                }
        }
        //If proxy is too far away from the force point, there is no repulsive force
        else if (dist > MaxRadius)
        {
            pvforcefield->ResetpbWallTouchedAndInRange(FALSE);

            for (int i=0;i<3;i++)
            {forceVec[i] = 0;}
        }

        //Find the max continuous force that the device is capable of (error handling)
        HDdouble nominalMaxContinuousForce;
        hdGetDoublev(HD_NOMINAL_MAX_CONTINUOUS_FORCE,
&nominalMaxContinuousForce);

        //Limit force calculated to Max continuous to avoid exceeding value
        for (int i=0;i<3;i++)
        {
            if(forceVec[i]>nominalMaxContinuousForce/2)
            forceVec[i] = nominalMaxContinuousForce/2;

            if(forceVec[i]<-nominalMaxContinuousForce/2)
            forceVec[i] = -nominalMaxContinuousForce/2;
        }

        //Rotate to physical (phantom) workspace (-90 deg about x) and add to the existing force
        force[0] += forceVec[0];
```

```
        force[1] += forceVec[2];
        force[2] -= forceVec[1];
    }

    return;
}
```

Force Point

The force point, the point from which the repulsive force is calculated, is either the point perpendicular to the proxy on the wall or one of the end points.

```
//______________________________________________________________ComputeForcePoint
void vmap_forcefield::ComputeForcePoint(VECTOR_TYPE* forceP, VECTOR_TYPE proxyP)
{
    //Assign variables values from wall and proxy to use for calculations
    double X0, X1, X2, Y0, Y1, Y2, Z0, Z1, Z2;
    X0 = proxyP.x; Y0 = proxyP.y; Z0 = proxyP.z;    //proxy location
    X1 = wallX1; Y1 = wallY1; Z1 = 0;               //wall endpoint 1
    X2 = wallX2; Y2 = wallY2; Z2 = 0;               //wall endpoint 2

    VECTOR_TYPE perpP = CheckforPerpPoint(proxyP);

    //if there is a perpendicular point and the proxy is in front of the wall, assign perpP as forceP
    if((X2-X1)*(Y0-Y1)-(Y2-Y1)*(X0-X1)<0 && perpP.z!=-1000000)
        {*forceP = perpP;}
    else    //otherwise get the corner point of the wall
    {
        VECTOR_TYPE cornerP=CalcProxyWallAngle(proxyP);

        //if there is cornerP exists, set this to forceP
        if (((X2-X1)*(Y0-Y1)-(Y2-Y1)*(X0-X1)<0) &&
                 perpP.z==-1000000 && cornerP.z!=-1000000)
        {        *forceP = cornerP;}
        else     //otherwise set forceP to the no value setting
        {        forceP->x = -1000000; forceP->y = -1000000; forceP->z = -1000000;}
    }

    return;
}

//______________________________________________________________CheckforPerpPoint
VECTOR_TYPE vmap_forcefield::CheckforPerpPoint(VECTOR_TYPE proxyP)
{
    VECTOR_TYPE perpP = {-1000000, -1000000, -1000000};

    double X0, X1, X2, Y0, Y1, Y2, Z0, Z1, Z2;
    X0 = proxyP.x; Y0 = proxyP.y; Z0 = proxyP.z;    //proxy location
    X1 = wallX1; Y1 = wallY1; Z1 = 0;               //wall endpoint 1
    X2 = wallX2; Y2 = wallY2; Z2 = 0;               //wall endpoint 2

    //Find the point on the wall that is perpendicular to proxy depending on the orientation
    if (Y2-Y1==0)                       //If the line segment is horizontal
```

```
        {       perpP.x = X0;
                perpP.y = Y1;
                perpP.z = Z0;
        }
        else if (X2-X1==0)                  //If the line segment is vertical
        {       perpP.x = X1;
                perpP.y = Y0;
                perpP.z = Z0;
        }
        else                                //If the line segment has a non-zero real slope
        {
                perpP.x = (Y0 + ((X2-X1)/(Y2-Y1))*X0 - Y1 + ((Y2-Y1)/(X2-X1))*X1)/
                        (((Y2-Y1)/(X2-X1))+((X2-X1)/(Y2-Y1)));
                perpP.y = Y0 - ((X2-X1)/(Y2-Y1))*perpP.x + ((X2-X1)/(Y2-Y1))*X0;
                perpP.z = Z0;
        }

        //Check if perpendicular point is on the wall
        If (perpP.x>=min(X1,X2) && perpP.x<=max(X1,X2) && perpP.y>=min(Y1,Y2) &&
perpP.y<=max(Y1,Y2))
        {       ;}                  //Do nothing
        else                        //If not, then set the perpendicular point to the no value point
        {       perpP.x=-1000000; perpP.y=-1000000; perpP.z=-1000000;}

        return perpP;

}

//________________________________________________________________________CalcProxyWallAngle
VECTOR_TYPE vmap_forcefield::CalcProxyWallAngle(VECTOR_TYPE proxyP)
{
        double X0, X1, X2, Y0, Y1, Y2, Z0, Z1, Z2;
        X0 = proxyP.x; Y0 = proxyP.y; Z0 = proxyP.z;     //proxy location
        X1 = wallX1; Y1 = wallY1; Z1 = 0;                //wall endpoint 1
        X2 = wallX2; Y2 = wallY2; Z2 = 0;                //wall endpoint 2

        //Calculate the distance from the proxy to each end point of the wall and determine the closest
        //corner point. Keep track of the corner point and the angle associated with this wall's corner (half
        //of the total corner angle)
        double distcorner1 = sqrt(pow(X0-X1,2)+pow(Y0-Y1,2)); //dist from proxy to (X1, Y1)
        double distcorner2 = sqrt(pow(X0-X2,2)+pow(Y0-Y2,2)); //dist from proxy to (X2, Y2)

        double angle = 0;
        double theta;                       //Wall angle where force is applicable
        VECTOR_TYPE vec1, vec2, cornerP;

        if(distcorner1<=distcorner2)        //If proxy is closer to (X1, Y1)
        {
                //Create vectors to determine the angle
                vec1.x=(X0-X1);
                vec1.y=(Y0-Y1);
                vec2.x=(X2-X1);
                vec2.y=(Y2-Y2);

                //Assign cornerP to (X1, Y1) and theta to the angle for this corner
```

```
                cornerP.x = X1;
                cornerP.y = Y1;
                cornerP.z = Z0;
                theta = cornerangle1;      //CornerAngle1 value set when map was first loaded
        }
        else if(distcorner2<distcorner1)   //If proxy is closer to (X2, Y2)
        {
                vec1.x=(wallX1-wallX2); vec1.y=(wallY1-wallY2);
                vec2.x=(proxyP.x-wallX2); vec2.y=(proxyP.y-wallY2);

                //Assign cornerP to (X2, Y2) and theta to the angle for this corner
                cornerP.x=wallX2; cornerP.y=wallY2; cornerP.z=Z0;
                theta=cornerangle2;        //CornerAngle2 value set when map was first loaded
        }

        //Find the angle between the proxy and wall and determine if this wall is responsible for
        //contributing the corner point
        angle = AngleBtwnVectors(vec1, vec2);

        //If the proxy is in the area outside of the corner that this wall is responsible for then the corner
        point is //set to this corner's coordinates
        if (angle*(180/PI)<=90+theta)
        {       ;}                         //Leave the corner point as it was set above
        else                               //Otherwise, set the corner point to the no value setting
        {       cornerP.x=-1000000; cornerP.y=-1000000; cornerP.z=-1000000;}

        return cornerP;
}

//_____________________________________________________________________________AngleBtwnVectors
double base_texture::AngleBtwnVectors(VECTOR_TYPE v1, VECTOR_TYPE v2)
{
        double angle;

        //Normalize the vectors
        VECTOR_TYPE norVec1, norVec2;
        norVec1.x = v1.x / sqrt(pow(v1.x, 2) + pow(v1.y, 2));
        norVec1.y = v1.y / sqrt(pow(v1.x, 2) + pow(v1.y, 2));
        norVec2.x = v2.x / sqrt(pow(v2.x, 2) + pow(v2.y, 2));
        norVec2.y = v2.y / sqrt(pow(v2.x, 2) + pow(v2.y, 2));

        //Calculate the dot product of the normalized vectors
        double dotProd = (norVec1.x * norVec2.x) + (norVec1.y * norVec2.y);

        if ( abs(dotProd - 1.0) <= epsilon )
                angle = 0;
        else if ( abs(dotProd + 1.0) <= epsilon )
                angle = PI;
        else
        {
                double cross = 0;
                angle = acos(dotProd);
                cross = (norVec1.x * norVec2.y) - (norVec2.x * norVec1.y);

                if (cross < 0) // vec1 rotate counter clockwise to vec2
                        angle = 2 * PI - angle;
```

```
        }

        return angle;
}
```

A.4 HAPTIC GUIDE PROGRAM CODE

Tether to Last Touched Point

```
//________________________________________________________________________TouchedTetherForce
void vmap_anchorforce::TouchedTetherForce (VECTOR_TYPE proxyP)
{
        VECTOR_TYPE P1, P2, PointOnLine;

        for (int i = 0; i < iNumGuides; i++)
        {
                //pdGuidesArray stores all the guide lines in order of the route
                //Set P1 to the first end point of the first guide line and P2 to the second end point
                //Set the first potential anchor point to P1, the start of the route
                setP (&P1, pdGuidesArray[4*i], pdGuidesArray[(4*i)+1], 0);
                setP (&P2, pdGuidesArray[4*i+2], pdGuidesArray[(4*i)+3], 0);
                PointOnLine = P1;

                //Determine whether the proxy is on the guide line
                if (P2.x-P1.x == 0)                                     //If the guide line is vertical
                {
                        //If the proxy is in line with the last touched position on the guide line and within
                        //the bounds of the guide line, record the proxy's new position as the last touched
                        //position and set it as the anchor
                        if (abs(PointOnLine.x-proxyP.x) < 0.1  &&
                                proxyP.y >= min(P1.y,P2.y) && proxyP.y <= max(P1.y,P2.y))
                        {
                                setP (&PointOnLine, P1.x, proxyP.y, 0);
                                AnchorP = PointOnLine;
                                return;
                        }
                }
                else if (P2.y - P1.y == 0)                              //If the guide line is horizontal
                {
                        if (abs(PointOnLine.y - proxyP.y) < 0.1 &&
                                 proxyP.x >= min(P1.x,P2.x) && proxyP.x<=max(P1.x,P2.x))
                        {
                                setP (&PointOnLine, proxyP.x, P1.y, 0);
                                AnchorP = PointOnLine;
                                return;
                        }
                }
                else if ((P1.x-P2.x) != 0  && (P1.y-P2.y) != 0 )        //All sloped guide lines
                {
                        double slope = CalcSlope(P1, P2);
                        VECTOR_TYPE PerpP = CalcPerpPoint(P1, P2, proxyP);
                        double length = CalcMagnitude(PerpP, proxyP);
```

```
			if (Length < 0.1 &&
				PerpPoint.x >= min(P1.x,P2.x) && PerpPoint.x <= max(P1.x,P2.x) &&
				PerpPoint.y >= min(P1.y,P2.y) && PerpPoint.y <= max(P1.y,P2.y))
			{
				setP (&PointOnLine, proxyP.x, proxyP.y, 0);
				AnchorP = PointOnLine;
				return;
			}
		}
	}

	return;
}

//_____________________________________________________________________________________CalcSlope
double base_texture::CalcSlope(VECTOR_TYPE p1, VECTOR_TYPE p2)
{
	return (p2.y-p1.y)/(p2.x-p1.x);
}

//_________________________________________________________________________________CalcMagnitude
double base_texture::CalcMagnitude(VECTOR_TYPE p1, VECTOR_TYPE p2)
{
	return sqrt(pow(p1.x-p2.x, 2) + pow(p1.y-p2.y, 2));
}

//_____________________________________________________________________________________CalcPerpP
VECTOR_TYPE base_texture::CalcPerpPoint(VECTOR_TYPE p1, VECTOR_TYPE p2, VECTOR_TYPE
pw)
{
	VECTOR_TYPE PerpP;
	double slope = (p2.y-p1.y)/(p2.x-p1.x);

	PerpP.x = (pw.y + (1/slope)*pw.x - p1.y + (slope)*p1.x)/(slope+(1/slope));
	PerpP.y = pw.y - (1/slope)*PerpP.x + (1/slope)*pw.x;
	PerpP.z = 0;

	return PerpP;
}
```

Key Press for Tether to Last Touched Point

```
//xdSpringStiffness() gives access the value stored in SpringStiffness
//setdSpringStiffness() sets the value stored in SpringStiffness
//MessageBeep() plays an error sound
//StiffnessIncrement is the amount that the stiffness (which directly impacts the strength since the force
//is calculated like a spring force) can increase with a single button press. It is initialized to 0.05.
//MaxStiffness and MinStiffness are initialized to 1.00 and 0.00, respectively.
case VK_UP://---------------------------------------------------------------------------------increase spring stiffness
    Stiffness = xdSpringStiffness();
    if ( (Stiffness + StiffnessIncrement) > MaxStiffness )
    {
        setdSpringStiffness(MaxStiffness);
        MessageBeep(uiNON_APPLICATION_KEY);
```

```
    }
    else
    {     setdSpringStiffness(Stiffness + StiffnessIncrement);}
    break;
case VK_DOWN://---------------------------------------------------------------------------------decrease spring stiffness
    Stiffness = xdSpringStiffness();
    if ( (Stiffness - StiffnessIncrement) < MinStiffness )
    {
        setdSpringStiffness(MinStiffness);
         MessageBeep(uiNON_APPLICATION_KEY);
    }
    else
    {   setdSpringStiffness(Stiffness - StiffnessIncrement);}
    break;
```

Manual Forward Force

```
//_______________________________________________________________________________________ManualAnchorForce
void vmap_anchorforce::ManualAnchorForce (void)
{
        if (iAnchorID == -1)
        {     setP (&AnchorP, pdGuidesArray[0], pdGuidesArray[1], 0);}
        else if (iAnchorID > -1 && iAnchorID < iNumGuides )
        {     setP (&AnchorP, pdGuidesArray[4*iAnchorID+2], pdGuidesArray[4*iAnchorID+3], 0);}
        else
        {     setP (&AnchorP, -1000000, -1000000, -1000000);}

        return;
}
```

Key Press for Manual Forward Force

```
//xiAnchorID() gives access the value stored in iAnchorID
//setiAnchorID() sets the value stored in iAnchorID
case  VK_LEFT://-----------------------------------------------------------------------------------------select prev anchor
    if (xiAnchorID() > -1)
    {   setiAnchorID(xiAnchorID() - 1);}
    else
    {   MessageBeep();}
    break;
 case VK_RIGHT://---------------------------------------------------------------------------------------select next anchor
    if (xiAnchorID() < xiNumGuides() - 1)
    {   setiAnchorID(xiAnchorID() + 1);}
    else
    {   MessageBeep();}
    break;
```

Automatic Forward Force

```
//__________________________________________________________________AutoAnchorForce
void vmap_anchorforce::AutoAnchorForce (VECTOR_TYPE Pw)
{
        //SnapDistance is set to 0.1. The proxy is thought to be touching a way point if it is less than the
        //SnapDistance away.
        //bDecreaseAnchorID is a static Boolean variable that keeps track of direction of travel (forward or
        //backward)
        double DistancefromAnchor = CalcMagnitude(AnchorPoint, Proxy);
        if (DistancefromAnchor < SnapDistance && iAnchorID >= -1 && iAnchorID < iNumGuides)
        {
        if (bDecreaseAnchorID == TRUE && iAnchorID > 0)
        {       //If travel direction is backward
                iAnchorID--;
                setP (&AnchorP, GuidesArray[4*iAnchorID], GuidesArray[4*iAnchorID+1], 0);
        }
        else if (bDecreaseAnchorID == FALSE && iAnchorID < iNumGuides-1)
        {       //If travel direction is forward
                iAnchorID++;
                setP (&AnchorP, GuidesArray[4*iAnchorID+2], GuidesArray[4*iAnchorID+3], 0);
        }
}
```

Attractive Force Used for Guide Agent

```
//_______________________________________________________________ComputeAnchorForce
void HLCALLBACK ComputeAnchorForce(HDdouble force[3], HLcache *cache, void *userdata)
{
        //Typecast the pointer passed in appropriately
        vmap_anchorforce* pvanchorforce = (vmap_anchorforce*)userdata;

        //Get proxy position in physical workspace frame
        static HLdouble Pp[3]; //Proxy in physical (phantom) workspace
        static VECTOR_TYPE Pw; //Proxy in world workspace

        //Transform the proxy to worldspace keeping in vmap units
        hdGetDoublev(HD_CURRENT_POSITION, Pp);
        pvanchorforce->PWTransformVmap(&Pw, Pp);

        //Compute the anchor point from which spring force should be calculated
        //Variable is static so that it holds the point until intentionally reassigned
        //Otherwise anchor point(s)  will reset everytime function is called
        pvanchorforce->ComputeAnchorPoint(Pw);

        static VECTOR_TYPE anchorP;
        pvanchorforce->SetAnchorPoint(&anchorP);

        if (anchorP.z != -1000000)
        {
                //Calculate force between the anchor point and the proxy
                HLdouble forceVec[3] = {0,0,0}; //vector to hold temporary force values

                //Calculate distance vector between the proxy and the anchor point
```

```
			HLdouble diffVec[3];
			diffVec[0] = anchorP.x - Pw.x;
			diffVec[1] = anchorP.y - Pw.y;
			diffVec[2] = anchorP.z - Pw.z;

			HDdouble springStiffness = pvanchorforce->xdSpringStiffness();

			for (int i=0;i<3;i++)
			{	forceVec[i] = springStiffness * diffVec[i];}

			//Find the max continuous force that the device is capable of (error handling)
			HDdouble nominalMaxContinuousForce;
			hdGetDoublev(HD_NOMINAL_MAX_CONTINUOUS_FORCE,
&nominalMaxContinuousForce);

			//Limit force calculated to Max continuous to avoid exceeding value
			for (int i=0;i<3;i++)
			{
				if(forceVec[i]>nominalMaxContinuousForce/2)
				forceVec[i] = nominalMaxContinuousForce/2;

				if(forceVec[i]<-nominalMaxContinuousForce/2)
				forceVec[i] = -nominalMaxContinuousForce/2;
			}

			//Rotate to physical (phantom) workspace (-90 deg about x) and add to the existing force
			force[0] += forceVec[0];
			force[1] += forceVec[2];
			force[2] -= forceVec[1];
		}

		return;
}
```

A.5 AUDIO GUIDE PROGRAM CODE

This function is responsible for executing all four audio guides depending on the user designated condition (e.g., manual audio, auto audio).

```
//________________________________________________________________ExecuteAudioGuide
void vmap::ExecuteAudioGuide(void)
{
		char* szTempPrevID = NULL;
		static DATA_TYPE* pdt = NULL;
		int iID = 1;

		vmap_location* pLocation = NULL;		//Pointer to vmap_location object
		if ( locations.ListIsNotEmpty() )		//If locations exist, initialize the sound variables
		{
			szCurrentIDSnd = NULL;
			szPrevIDSnd = NULL;	szNextIDSnd = NULL;
			szPrevDirSnd = NULL;	szNextDirSnd = NULL;
```

```
PrevIDCoord.x = -1000000;   PrevIDCoord.y = -1000000;   PrevIDCoord.z = -1000000;
NextIDCoord.x = -1000000;   NextIDCoord.y = -1000000;   NextIDCoord.z = -1000000;

pLocation = locations.GetNextObj(TRUE);
while ( pLocation != NULL )                 //Loop through the all the locations
{
        if (pLocation->xBIn() == TRUE)  //If the proxy is contained in a specific location
        {

                //Set the location ID sound to that of the object the proxy is in
                szCurrentIDSnd = pLocation->xszIDSnd();
                szPrevDirSnd = pLocation->xszPrevSnd();
                szNextDirSnd = pLocation->xszNextSnd();

                //Automatically play the ID sound if it's the first time in the way point
                if(bAudioGuideAutoPlay == TRUE && pLocation->xBWasIn() == FALSE
                        && pLocation->xiLocationType() == 1)
                {
                        if (vAnchorForce.xiTestMode() == 2 )      //Auto audio mode
                        {
                                if (vAnchorForce.xbDecAnchorID() == FALSE)
                                {
                                        pdt = Data->RecCommand( iCMD_PLAY_ID_SOUND );
                                        vAud->ID(szNextDirSnd, Px, Py, Pz, AUD_bNOW,
                                                AUD_iDIST_SCALE_ON, AUD_iVOL, pdt);
                                }
                                else
                                {       //Play directions to the next way pt
                                        pdt = Data->RecCommand( iCMD_PLAY_ID_SOUND );
                                        vAud->ID(szPrevDirSnd, Px, Py, Pz, AUD_bNOW,
                                                AUD_iDIST_SCALE_ON, AUD_iVOL, pdt);
                                }
                        }
                        else                                            //All other modes
                        {       //Play the way point ID number
                                pdt = Data->RecCommand( iCMD_PLAY_ID_SOUND );
                                vAud->ID(szCurrentIDSnd, Px, Py, Pz, AUD_bNOW,
                                        AUD_iDIST_SCALE_ON, AUD_iVOL, pdt);
                        }
                }

                //Iterate to the next location on the route and save the coordinates for the
                //manual spatialized guide
                pLocation = locations.GetNextObj();
                if ( pLocation != NULL && iID < iNumWayPoints-1)
                {
                        NextIDCoord.x = pLocation->xXs();
                        NextIDCoord.y = pLocation->xYs();
                        NextIDCoord.z = pLocation->xZs();
                }
                else
                {
                        NextIDCoord.x = -1000000;
                        NextIDCoord.y = -1000000;
                        NextIDCoord.z = -1000000;
                }
```

```
					return;
				}

				//If proxy was not found in the location, save the location's coordinates and set as
				//the previous way pt's coordinates for the manual spatialized guide and iterate to
				//the next way pt to check
				if ( iID < iNumWayPoints-1)			//iNumWayPoints is equal to 11
				{
					PrevIDCoord.x = pLocation->xXs();
					PrevIDCoord.y = pLocation->xYs();
					PrevIDCoord.z = pLocation->xZs();
				}
				else
				{
					PrevIDCoord.x = -1000000;
					PrevIDCoord.y = -1000000;
					PrevIDCoord.z = -1000000;
				}

				pLocation = locations.GetNextObj();
				iID++;
			}
		}
		return;
}
```

APPENDIX B: EXPERIMENTAL DATA

B.1 FORCE FIELD EXPERIMENT #1 DATA

Subject ID	Trial	Condition	Map No.	Explore Time (s)	Build Time (s)	Build Errors
1	Training #1	No Force Field	2	253	153	4
1	Test #1	No Force Field	4	325	139	1
1	Training #2	Force Field	1	152	183	0
1	Test #2	Force Field	3	225	76	0
2	Training #1	Force Field	2	186	92	3
2	Test #1	Force Field	4	707	122	3
2	Training #2	No Force Field	1	450	90	0
2	Test #2	No Force Field	3	560	57	1
3	Training #1	No Force Field	1	248	108	4
3	Test #1	No Force Field	4	340	89	2
3	Training #2	Force Field	2	1120	100	0
3	Test #2	Force Field	3	329	72	1
4	Training #1	Force Field	1	183	66	3
4	Test #1	Force Field	3	255	62	2
4	Training #2	No Force Field	2	259	50	1
4	Test #2	No Force Field	4	207	47	2

B.2 FORCE FIELD EXPERIMENT #2 DATA

Subject #1

Test No.	Condition	Map No.	Explore Time (s)	Identification Error
Training #1	Force Field	B	60	0
Test #1	Force Field	2	16	0
Test #2	Force Field	9	20	0
Test #3	Force Field	3	19	0
Test #4	Force Field	7	18	0
Test #5	Force Field	6	13	0
Test #6	Force Field	10	15	0
Test #7	Force Field	1	11	0
Test #8	Force Field	8	12	0
Test #9	Force Field	5	18	0
Test #10	Force Field	4	10	0
Training #2	No Force Field	A	30	0
Test #11	No Force Field	9	16	0
Test #12	No Force Field	5	15	0
Test #13	No Force Field	2	17	0
Test #14	No Force Field	4	14	0
Test #15	No Force Field	8	17	0
Test #16	No Force Field	3	14	0
Test #17	No Force Field	1	9	0
Test #18	No Force Field	7	14	0
Test #19	No Force Field	10	8	0
Test #20	No Force Field	6	14	0

Subject #5

Test No.	Condition	Map No.	Explore Time (s)	Identification Error
Training #1	No Force Field	A	161	0
Test #1	No Force Field	5	51	0
Test #2	No Force Field	4	72	0
Test #3	No Force Field	8	105	0
Test #4	No Force Field	6	40	0
Test #5	No Force Field	1	35	0
Test #6	No Force Field	3	32	0
Test #7	No Force Field	9	67	0
Test #8	No Force Field	10	44	0
Test #9	No Force Field	7	45	0
Test #10	No Force Field	2	65	0
Training #2	Force Field	B	63	0
Test #11	Force Field	7	35	0
Test #12	Force Field	3	36	0
Test #13	Force Field	5	17	0
Test #14	Force Field	8	48	0
Test #15	Force Field	2	20	0
Test #16	Force Field	1	29	0
Test #17	Force Field	6	29	0
Test #18	Force Field	4	19	0
Test #19	Force Field	9	27	0
Test #20	Force Field	10	25	0

Subject #6

Test No.	Condition	Map No.	Explore Time (s)	Identification Error
Training #1	Force Field	B	82	0
Test #1	Force Field	5	16	0
Test #2	Force Field	2	20	0
Test #3	Force Field	8	19	0
Test #4	Force Field	9	18	0
Test #5	Force Field	10	13	0
Test #6	Force Field	7	15	0
Test #7	Force Field	6	11	0
Test #8	Force Field	1	12	0
Test #9	Force Field	3	18	0
Test #10	Force Field	4	10	0
Training #2	No Force Field	A	45	0
Test #11	No Force Field	9	16	0
Test #12	No Force Field	5	15	0
Test #13	No Force Field	2	17	0
Test #14	No Force Field	4	14	0
Test #15	No Force Field	8	17	0
Test #16	No Force Field	3	14	0
Test #17	No Force Field	1	9	0
Test #18	No Force Field	7	14	0
Test #19	No Force Field	10	8	0
Test #20	No Force Field	6	14	0

Subject #7

Test No.	Condition	Map No.	Explore Time (s)	Identification Error
Training #1	No Force Field	B	270	0
Test #1	No Force Field	5	51	0
Test #2	No Force Field	4	72	0
Test #3	No Force Field	8	105	0
Test #4	No Force Field	6	40	0
Test #5	No Force Field	1	35	0
Test #6	No Force Field	3	32	0
Test #7	No Force Field	9	67	0
Test #8	No Force Field	10	44	0
Test #9	No Force Field	7	45	0
Test #10	No Force Field	2	65	0
Training #2	Force Field	A	63	0
Test #11	Force Field	7	35	0
Test #12	Force Field	3	36	0
Test #13	Force Field	5	17	0
Test #14	Force Field	8	48	0
Test #15	Force Field	2	20	0
Test #16	Force Field	1	29	0
Test #17	Force Field	6	29	0
Test #18	Force Field	4	19	0
Test #19	Force Field	9	27	0
Test #20	Force Field	10	25	0

B.3 HAPTIC GUIDE EXPERIMENT DATA

				Completion Time (s)			Number of Errors		Subject Ranking			
Subject ID	Trial	Map No.	Condition	Exploration	Identification	Navigation	Identification	Navigation	Helpful	Easiest	Confidence	Preference*
1	Training #1	1	Guide Line	176	27	1	1	0	-	-	-	-
1	Training #2	3	Manual	93	19	7	0	2	-	-	-	-
1	Training #3	5	Tether	125	20	2	0	1	-	-	-	-
1	Training #4	8	Automatic	130	22	2	0	0	-	-	-	-
1	Test #1	2	Manual	75	19	2	0	0	1	1	1	1
1	Test #2	4	Automatic	76	33	3	0	0	2	2	1	2
1	Test #3	6	Tether	61	24	3	0	0	4	4	1	4
1	Test #4	7	Guide Line	160	24	3	0	0	3	3	1	3
5	Training #1	1	Tether	525	30	16	2	3	-	-	-	-
5	Training #2	4	Guide Line	275	25	3	0	0	-	-	-	-
5	Training #3	6	Manual	412	20	5	2	2	-	-	-	-
5	Training #4	7	Automatic	374	25	3	2	0	-	-	-	-
5	Test #1	2	Automatic	110	18	2	0	0	2	2	1	2
5	Test #2	3	Tether	149	25	5	2	1	3	3	3	3
5	Test #3	5	Manual	200	16	6	0	2	1	1	2	1
5	Test #4	8	Guide Line	80	20	3	2	0	4	4	4	4
6	Training #1	1	Automatic	593	20	9	1	1	-	-	-	-
6	Training #2	3	Guide Line	145	25	13	0	2	-	-	-	-
6	Training #3	5	Tether	239	39	5	0	0	-	-	-	-
6	Training #4	7	Manual	510	16	22	1	0	-	-	-	-
6	Test #1	2	Tether	219	16	6	0	1	4	4	4	4
6	Test #2	4	Manual	111	21	4	0	0	1	1	1	1
6	Test #3	6	Guide Line	186	15	5	0	0	3	3	3	3
6	Test #4	8	Automatic	230	10	10	0	3	2	2	2	2
7	Training #1	2	Guide Line	176	27	1	1	0	-	-	-	-
7	Training #2	4	Manual	93	19	7	0	2	-	-	-	-
7	Training #3	5	Tether	125	20	2	0	1	-	-	-	-
7	Training #4	7	Automatic	130	22	2	0	0	-	-	-	-
7	Test #1	1	Guide Line	102	91	10	2	3	4	3	4	4
7	Test #2	3	Automatic	51	145	24	2	4	3	4	1	3
7	Test #3	6	Tether	152	3	10	0	2	2	2	3	2
7	Test #4	8	Manual	139	102	3	4	0	1	1	2	1

*Preference is the average of the Helpful, Easiest, and Confidence rankings.

B.4 AUDIO GUIDE EXPERIMENT DATA

				Completion Time (s)		Number of Navigation	Subject Ranking			
Subject ID	Trial	Condition	Map No.	Exploration	Navigation	Errors	Helpful	Easiest	Confidence	Preference*
1	Training #1	Automatic	5	125	3	0	-	-	-	-
1	Training #2	Spatialized Manual	3	77	3	0	-	-	-	-
1	Training #3	Landmark	7	97	3	0	-	-	-	-
1	Training #4	Manual	1	57	4	0	-	-	-	-
1	Test #1	Spatialized Manual	4	38	3	0	2	1	1	1.3
1	Test #2	Automatic	6	95	3	0	1	3	1	1.7
1	Test #3	Manual	2	47	3	0	2	1	1	1.3
1	Test #4	Landmark	8	117	3	0	4	4	4	4.0
5	Training #1	Manual	1	207	6	0	-	-	-	-
5	Training #2	Landmark	7	285	7	1	-	-	-	-
5	Training #3	Automatic	5	100	3	0	-	-	-	-
5	Training #4	Spatialized Manual	3	101	3	0	-	-	-	-
5	Test #1	Automatic	6	113	3	0	1	1	3	1.7
5	Test #2	Spatialized Manual	4	158	4	0	2	2	2	2.0
5	Test #3	Manual	2	159	3	0	2	2	2	2.0
5	Test #4	Landmark	8	247	2	0	4	4	1	3.0
6	Training #1	Landmark	7	201	25	2	-	-	-	-
6	Training #2	Manual	1	126	5	0	-	-	-	-
6	Training #3	Automatic	5	126	5	0	-	-	-	-
6	Training #4	Spatialized Manual	3	156	8	1	-	-	-	-
6	Test #1	Automatic	6	55	9	1	1	1	1	1.0
6	Test #2	Manual	2	31	4	0	2	2	2	2.0
6	Test #3	Landmark	8	77	5	0	3	3	3	3.0
6	Test #4	Spatialized Manual	4	72	4	2	4	4	4	4.0
7	Training #1	Manual	1	167	12	1	-	-	-	-
7	Training #2	Spatialized Manual	3	300	7	1	-	-	-	-
7	Training #3	Landmark	7	102	8	0	-	-	-	-
7	Training #4	Automatic	5	47	11	0	-	-	-	-
7	Test #1	Landmark	8	77	5	0	4	4	3	3.7
7	Test #2	Spatialized Manual	4	39	3	0	2	1	1	1.3
7	Test #3	Automatic	6	48	6	0	1	3	4	2.7
7	Test #4	Manual	2	36	8	0	3	2	2	2.3

*Preference is the average of the Helpful, Easiest, and Confidence rankings.

B.5 INITIAL MAIN EXPERIMENT DATA

				Completion Time (s)		Number of Navigation	Subject Ranking			
Subject ID	Trial	Condition	Map No.	Learn	Navigation	Errors	Helpful	Easiest	Confidence	Preference*
8	Training #1	Audio	6	1021	55	5	-	-	-	-
8	Training #2	Haptic	5	580	151	17	-	-	-	-
8	Training #3	None	4	1300	32	4	-	-	-	-
8	Test #1	None	2	385	30	2	2	2	1	1.7
8	Test #2	Audio	1	218	138	10	1	1	2	1.3
8	Test #3	Haptic	3	390	50	6	3	3	3	3.0
9	Training #1	Audio	1	1005	44	9	-	-	-	-
9	Training #2	None	6	274	147	32	-	-	-	-
9	Training #3	Haptic	2	1096	55	17	-	-	-	-
9	Test #1	Haptic	3	1162	56	6	2	3	3	2.7
9	Test #2	Audio	5	990	38	3	1	1	1	1.0
9	Test #3	None	4	790	38	9	3	2	2	2.3
10	Training #1	Haptic	5	1110	205	16	-	-	-	-
10	Training #2	None	4	566	27	4	-	-	-	-
10	Training #3	Audio	1	248	352	36	-	-	-	-
10	Test #1	None	3	225	199	15	2	1	1	1.3
10	Test #2	Haptic	2	198	135	13	3	1	3	2.3
10	Test #3	Audio	6	133	152	14	1	1	1	1.0
11	Training #1	Haptic	6	1340	133	16	-	-	-	-
11	Training #2	Audio	1	539	20	3	-	-	-	-
11	Training #3	None	4	411	16	4	-	-	-	-
11	Test #1	Audio	5	237	13	2	1	1	1	1.0
11	Test #2	None	1	354	24	1	2	2	2	2.0
11	Test #3	Haptic	3	720	18	4	3	3	3	3.0

*Preference is the average of the Helpful, Easiest, and Confidence rankings.

B.6 FINAL MAIN EXPERIMENT DATA

				Completion Time (s)		Number of Navigation	Subject Ranking			
Subject ID	Trial	Map #	Condition	Learn	Navigation	Errors	Helpful	Easiest	Confidence	Preference*
12	Training #1	6	Audio	1295	69	5	-	-	-	-
12	Training #2	5	Haptic	738	34	2	-	-	-	-
12	Training #3	4	None	939	19	3	-	-	-	-
12	Test #1	2	None	821	26	6	1	2	2	1.7
12	Test #2	1	Audio	428	127	16	3	3	3	3.0
12	Test #3	3	Haptic	526	120	16	2	1	1	1.3
13	Training #1	1	Audio	620	26	1	-	-	-	-
13	Training #2	6	None	494	40	5	-	-	-	-
13	Training #3	2	Haptic	598	24	2	-	-	-	-
13	Test #1	3	Haptic	308	13	0	1	2	1	1.3
13	Test #2	5	Audio	604	19	1	3	3	3	3.0
13	Test #3	4	None	344	20	6	2	1	1	1.3
14	Training #1	5	Haptic	1650	17	3	-	-	-	-
14	Training #2	4	None	870	58	7	-	-	-	-
14	Training #3	1	Audio	1106	34	4	-	-	-	-
14	Test #1	3	None	281	9	0	1	1	1	1.0
14	Test #2	2	Haptic	960	29	4	3	3	3	3.0
14	Test #3	6	Audio	725	35	8	2	2	2	2.0
15	Training #1	6	Haptic	1929	188	32	-	-	-	-
15	Training #2	1	Audio	1110	195	20	-	-	-	-
15	Training #3	4	None	489	64	9	-	-	-	-
15	Test #1	3	Audio	270	36	2	1	2	3	2.0
15	Test #2	2	None	144	50	4	3	1	2	2.0
15	Test #3	5	Haptic	195	38	3	2	3	1	2.0
16	Training #1	4	None	510	62	12	-	-	-	-
16	Training #2	6	Haptic	1050	90	15	-	-	-	-
16	Training #3	2	Audio	840	44	5	-	-	-	-
16	Test #1	3	Haptic	360	65	9	3	2	3	2.7
16	Test #2	1	None	600	44	6	2	1	1	1.3
16	Test #3	5	Audio	375	25	2	1	3	2	2.0
17	Training #1	1	Audio	1195	35	5	-	-	-	-
17	Training #2	3	None	546	14	3	-	-	-	-
17	Training #3	4	Haptic	525	43	7	-	-	-	-
17	Test #1	6	None	499	13	2	3	2	3	2.7
17	Test #2	2	Audio	396	25	5	2	3	2	2.3
17	Test #3	5	Haptic	263	94	12	1	1	1	1.0
18	Training #1	2	None	1259	24	4	-	-	-	-
18	Training #2	6	Audio	730	46	9	-	-	-	-
18	Training #3	5	Haptic	553	41	4	-	-	-	-
18	Test #1	4	Audio	1429	84	15	3	2	1	2.0
18	Test #2	1	Haptic	470	240	31	1	1	1	1.0
18	Test #3	3	None	732	45	6	2	3	1	2.0

*Preference is the average of the Helpful, Easiest, and Confidence rankings.

				Completion Time (s)		Number of Navigation	Subject Ranking			
Subject ID	Trial	Map #	Condition	Learn	Navigation	Errors	Helpful	Easiest	Confidence	Preference*
19	Training #1	3	Audio	345	12	1	-	-	-	-
19	Training #2	5	Haptic	1285	22	2	-	-	-	-
19	Training #3	2	None	828	18	4	-	-	-	-
19	Test #1	4	Haptic	870	194	26	2	2	2	2.0
19	Test #2	1	Audio	810	22	1	1	1	1	1.0
19	Test #3	6	None	866	20	3	3	3	3	3.0
20	Training #1	2	None	867	18	4	-	-	-	-
20	Training #2	3	Audio	570	41	6	-	-	-	-
20	Training #3	6	Haptic	1314	87	14	-	-	-	-
20	Test #1	1	Audio	296	81	8	1	1	1	1.0
20	Test #2	5	Haptic	235	12	4	2	2	2	2.0
20	Test #3	4	None	809	45	6	3	3	2	2.7
21	Training #1	2	None	500	35	5	-	-	-	-
21	Training #2	5	Audio	685	20	3	-	-	-	-
21	Training #3	1	Haptic	750	28	4	-	-	-	-
21	Test #1	6	Haptic	660	17	1	1	1	2	1.3
21	Test #2	3	None	452	16	1	3	2	1	2.0
21	Test #3	4	Audio	825	31	2	2	3	3	2.7

*Preference is the average of the Helpful, Easiest, and Confidence rankings.