Road Stakeout In Wearable Outdoor Augmented Reality

A thesis
submitted in partial fulfilment
of the requirements for the Degree
of
Doctor of Philosophy
in the
University of Canterbury
by
Volkert "Oakley" Buchmann

Examining Committee	
Mark Billinghurst	Supervisor
Andy Cockburn	Co-Supervisor

University of Canterbury
2008

Declaration

All contributions described in this thesis are entirely my own, with some exceptions where I collaborated with other researchers. The work described in each chapter was primarily my own. The following list identifies those parts of my research where I collaborated with other researchers and clarifies the role of the contributors.

Chapter 3 – Directional Interfaces: I wrote the majority of a paper that described the experiment with the assistance of my supervisors, Mark Billinghurst and Andy Cockburn [Buchmann et al. 2008].

Chapter 4 – Depth Cues: I wrote the majority of a paper that described the experiment with the assistance of my supervisors, Mark Billinghurst and Andy Cockburn [Jürgens et al. 2006][1].

Chapter 5 – Obscured Information Visualisation: I wrote the majority of a paper that described the work from sections 5.3.2 and 5.3.3 with the assistance of Trond Nilsen and my supervisor Mark Billinghurst [Buchmann et al. 2005].

Chapter 6 – A WOAR Stakeout Application: The WOAR system described in section 6.1 was primarily developed by Kevin Sharp, Andy Evans, and Nick Mein. The idea and the implementation of the rotated camera described in section 6.1.1, the visualisation described in section 6.1.2, and the lag measurement described in section 6.1.3 were primarily my own. The rest of the chapter describes my own work.

[1] Jürgens is my birth name

Dedicated to
Christel and Uwe

Abstract

This thesis advances wearable outdoor augmented reality (WOAR) research by proposing novel visualisations, by consolidating previous work, and through several formal user studies. Wearable outdoor augmented reality combines augmented reality (AR) and wearable computing to enable novel applications. AR allows the user to perceive virtual objects as part of their real environment. Using wearable computers as a platform for AR allows users to see the real and the virtual world combined in a mobile environment. This combination enables new and exciting applications that bring with them new challenges for interface and usability research.

The research described in this thesis advances the field of WOAR research by developing a WOAR version of a commercial road stakeout application. This case study makes possible the first formal direct comparison of the performance of a WOAR application and its conventional counterpart. Road stakeout is the process of locating points in the real world and marking them with stakes. This process is not only relevant for road construction, but also to construction and surveying in general. A WOAR stakeout application can visualise stakeout targets on their location in the real world, while conventional stakeout systems can only guide users to these locations using indirect displays. The formal comparison found significant differences in performance, and showed that the WOAR system performed twice as fast at the same accuracy level as the conventional system. The study also identified a number of usability issues and technical problems related to WOAR systems that still need to be overcome.

The thesis examines usability problems of the WOAR road stakeout application in detail, proposes solutions, and compares their efficiency in formal user studies. The basic stakeout tasks are navigating to a target location and then placing a stakeout pole on that location. Original research in the fields of directional interfaces and depth cues determined solutions for efficient navigation and pole placement in the WOAR stakeout application.

Further, the presented work includes explorative implementations of obscured information visualisations. The thesis proposes interaction with artificially transparent stakeout poles and hands, and examines their feasibility with respect to perceptual and technical issues. A visualisation of a road model investigated the preservation of context while automatically providing detail when needed.

The thesis presents working WOAR implementations of navigation and depth cue support, a road model visualisation, and an artificially transparent stakeout pole.

In conclusion, the thesis consolidated WOAR interface research and extended the field with empirical research. The presented research is the first that allows a WOAR application to compete directly with a commercial conventional system, demonstrating the strong potential that WOAR systems already have.

Table of Contents

List of Figures

List of Tables

Acknowledgments

I would like to thank all who helped and supported me while I worked on this thesis.

My supervisors Mark Billinghurst and Andy Cockburn supported me greatly for over four years. It was because of Andy that I came to the University of Canterbury, and it was Mark who invited me to begin this PhD thesis.

The research presented in this thesis was made possible through the support by Trimble Navigation NZ Ltd. Kevin Sharp improved the robustness and accuracy of the WOAR system greatly. I would like to thank Stuart Ralston and Bruce Graham for their support, and everybody at Trimble who helped out.

I would like to thank everybody at the HIT Lab NZ who helped me. Thanks to Julian Looser and Matt Keir for their proofreading and style suggestions. Thanks also to Trond Nilsen and Christina Dicke.

My awesome friends supported me through the good and the bad times of my PhD. Special thanks go to my proofreaders Angela Canton, Sally McLennan, Emma Cunliffe, and Fi Bennetts. I would also like to thank the members of the facebook group "Oakley, do your thesis" for their continued support and nagging.

My family supported me through my decade of university study with their love and their resources.

I would like to thank Steven Feiner, Blair MacIntyre, Tobias Hllerer from the Department of Computer Science, Columbia University; Anthony Webster from the Graduate School of Architecture, Planning and Preservation, Columbia University; Wayne Peikarski, Bruce Thomas, and Aaron Stafford from the Wearable Computer Laboratory, School of Computer and Information Science, University of South Australia; Andreas Schmeil and Wolfgang Broll from Fraunhofer FIT for the permission to use some of their images in this thesis.

This research was mainly funded by a Technology for Industry Fellowship from Technology New Zealand. My research would not have been possible without them.

Chapter I

Introduction

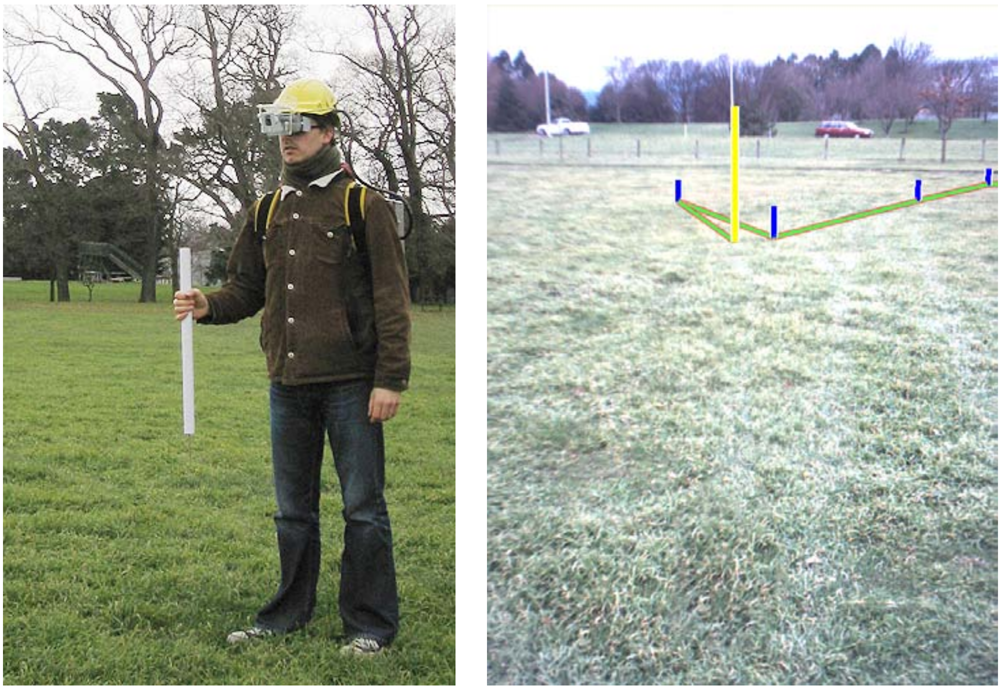
This thesis describes the development and the evaluation of a wearable outdoor augmented reality (WOAR) stakeout application.

Augmented reality (AR) typically overlays virtual imagery onto the user's view of the real environment in order to create the illusion of virtual objects being part of the real world. Augmented reality is commonly defined as having the following properties [Azuma et al. 2001] :

- "combines real and virtual objects in a real environment;
- runs interactively, and in real time; and
- registers (aligns) real and virtual objects with each other."

Using wearable computers as a platform for augmented reality enables a completely new set of applications. Wearable computing is a new research field that explores how computers can be worn on the body to provide constant access to computing power. By adding a head-mounted display and trackingsensors to a wearable computer, users can move through the real world, and have an augmented reality experience wherever they are.

Current WOAR system typically use backpack computers combined with a head mounted display, GPS, and an inertial tracking system, and run custom developed graphics and application software (figure 1.1). Once the underlying hardware and software is ready to be commercialized a range of innovative applications that were previously science fiction could become reality. For example, tourists would be able to see information overlaid on historic buildings, prospective house owners would be able to see and enter their house on site before it is built, and construction workers would be able to use X-ray vision to see underground pipes and cables.



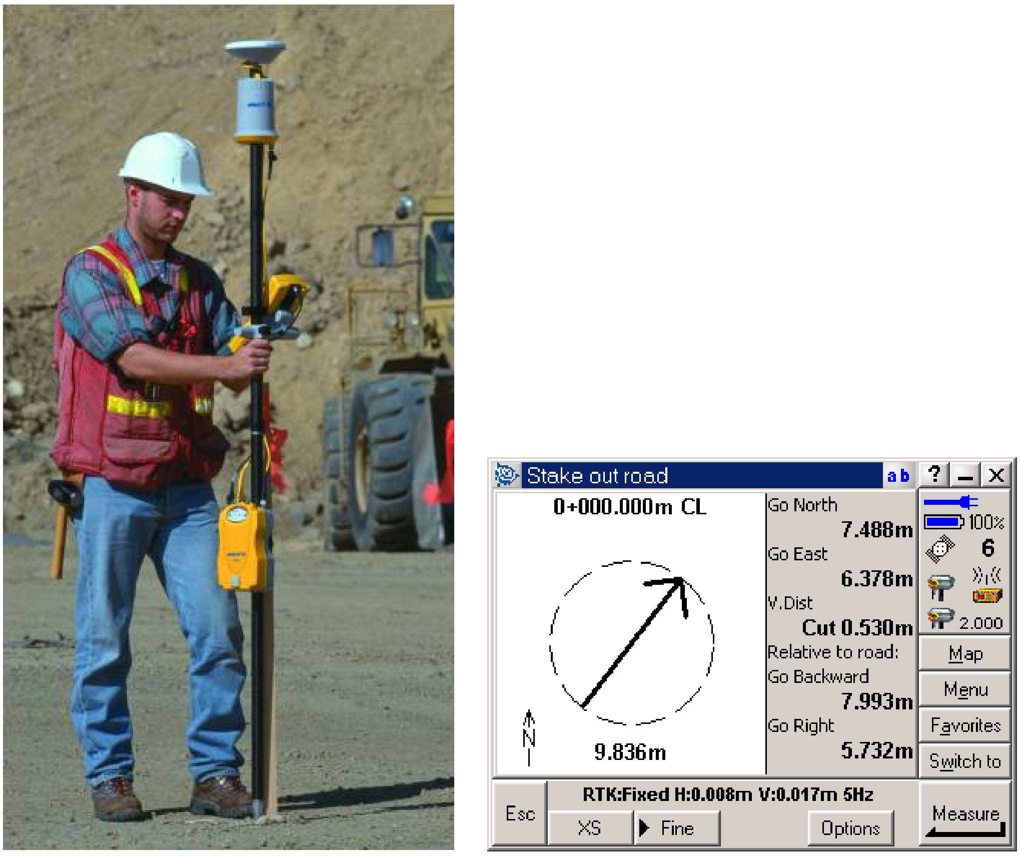
(a) A user wearing the WOAR system

(b) A visualisation of stakeout targets connected by a path. The virtual yellow pole marks the current target.

Figure 1.1: The WOAR system described in this thesis.

The application focus of this research is a WOAR road stakeout system. Road stakeout is the process of transferring a road design into the real world by placing wooden stakes in the ground at points defined in the road design. Road builders use these stakes for guidance when building the road. A stakeout task requires the user to quickly and accurately place a pole at a location defined by a road design point. A commercial conventional stakeout system, the Trimble Survey Controller (figure 1.2(a)), displays the direction to and the distance from the point on a small handheld screen (figure 1.2(b)). In contrast, a WOAR system is able to display the point as if it were part of the real world (figure 1.1(b)). Thus, the user no longer has to mentally map between the information display and the real world. This should make the stakeout process faster and easier. Formal studies are required to confirm this, and this thesis presents such a user study.

WOAR road stakeout is a good case study for several relevant interface prob-



Stake out road

0+000.000m CL

9.836m

Go North
7.488m
Go East
6.378m
V.Dist
Cut 0.530m
Relative to road:
Go Backward
7.993m
Go Right
5.732m

RTK:Fixed H:0.008m V:0.017m 5Hz

(a) The TSC in the field

(b) A screen capture of the TSC interface, guiding the user to a stakeout location

Figure 1.2: The Trimble Survey Controller (TSC)

lems that need to be addressed to develop commercially usable WOAR systems. A successful road stakeout system needs to provide

- guidance for navigation to a particular point,
- depth cues for placing the pole accurately on a point,
- a visualisation of a complex road model that does not clutter the screen.

This thesis describes original research in each of these fields, the integration of the resulting interface components into a WOAR road stakeout application, and a formal comparison of it with the TSC.

1.1 Scope

This thesis aims to find out whether a WOAR stakeout application can be developed and if the performance of such a system is good enough to eventually replace conventional stakeout systems. In order to build a commercialisable WOAR system, a number of hardware and software problems need to be investigated and solved. This includes the AR tracking quality, the weight of the wearable system, the ergonomics of the hardware, menu interfaces, and possible input devices. The majority of these problems cannot be covered in a single PhD thesis, and so this thesis is limited to supporting the actions that are most fundamental to a road stakeout application: visualising the road and the stakeout points, navigating to the next stakeout point, and placing the stakeout pole onto a point. If these actions cannot be performed satisfactorily with a WOAR system, then a WOAR stakeout application cannot be built. Once these fundamental problems of WOAR stakeout have been overcome, broader usability factors such as hardware ergonomics or input devices will have to be considered.

1.2 Problem Statement and Approach

The main motivation for this PhD research is to build a WOAR stakeout system that performs at least as well as a conventional stakeout system in a typical task. In order to develop an efficient WOAR stakeout application, three different user interface problems were investigated: navigation, depth cues and the viualisation of obscured information. The best interface solution found for each of the three areas was implemented in a WOAR system. Then, in a formal user study, the performance of the resulting WOAR system was compared to the performance of a conventional stakeout system.

The three interface components explored were navigation, depth cues, and the visualisation of obscured information:

- *Navigation* support can help the user to walk quickly between stakeout locations. The thesis reviews previous work on WOAR navigation and presents a formal user study that compared the performance of directional interfaces.

- *Depth cues* are relevant when placing a real pole on a virtual marker. The thesis presents a formal user study that compared the performance of depth cues for stakeout in AR.

- *Obscured information visualisation* techniques were investigated in a series of explorative implementations. For example, one of the implementations visualised both the shape of the road and the complex structure inside the road at the same time. Another implementation explored perceptual issues of using an artificially transparent stakeout pole.

The best solutions for these problems were then combined into a working WOAR application. A formal user study evaluated the application to identify further usability problems and to find out how well it performed in comparison to the conventional Trimble Survey Controller.

1.3 Contributions

The main contributions of the thesis are:

- The first implementation of a WOAR stakeout application.

- The first formal user study comparing the performance of a WOAR application and its conventional counterpart. This is also the first formal user study to assess the objective performance of a WOAR system.

- A formal user study comparing several depth cues for a stakeout task. This is the first study of depth cues in augmented reality at a distance of two meters from the observer's eyes.

- A review and a formal comparison of the most relevant directional interfaces for wearable AR in a user study.

- The first formal user study to compare the performance of a haptic belt to that of other navigation aids. A haptic belt is a novel directional interface that indicates direction through haptic feedback.

- Explorative studies on the use of transparent hands and tools, and the use of transparent stakeout poles with respect to real and virtual markers.

- A road visualisation that combines techniques of obscured information visualisation and information filtering.

1.4 Overview of the Thesis

Chapter 2, "Background", describes the relevant concepts for the presented work and reviews previous research for wearable outdoor augmented reality as well as the three areas of interest: navigation, depth cues, and obscured information visualisation. The stakeout process with the Trimble Survey Controller is also described.

Chapter 3, "Directional Interfaces for WOAR Navigation", describes the implementation of a selection of directional interfaces for WOAR systems and a formal comparison of the interfaces in a user study. With a significant difference between interfaces, a circular compass was found to be the best interface and a WOAR implementation of it is presented.

Chapter 4, "Depth Cues for AR Stakeout", describes the implementation and evaluation of a selection of depth cues for stakeout in an AR application. The formal user study that compared the relative performance of these depth cues for a stakeout task showed that users did not use the visual depth cues presented to them and instead relied on kinesthetic depth knowledge. However, occlusion was found to be fundamental for WOAR stakeout, and the chapter describes an optical tracking algorithm for a stakeout pole that ensured correct occlusion of virtual stakeout markers by the real pole in a WOAR system.

Chapter 5, "Obscured Information Visualisation for WOAR Road Stakeout", describes explorative implementations of four different types of obscured information visualisation. The chapter describes the implementation of a road visualisation for WOAR stakeout that automatically hides or reveals obscured detail and it explores the use of artificially transparent stakeout poles and hands.

Chapter 6, "A WOAR Stakeout Application", describes a formal user study that compared the performance of the WOAR stakeout application to that of the Trimble Survey Controller (TSC). The study found that the WOAR application performed significantly faster and allowed users to stake out twice as fast as with the TSC at an accuracy of about 4cm. The study identified several usability issues with the WOAR system, and the chapter proposes solutions for these issues.

Chapter 7, “Conclusion”, discusses the results of the research, proposes future work, and provides a concise summary of the contributions in the thesis.

Chapter II

Background

This chapter describes the basic concepts of wearable outdoor augmented reality (WOAR) that are relevant for this thesis. It then introduces the concepts of depth cues, navigation and obscured information visualisation in wearable outdoor augmented reality and reviews previous work from each of these areas. This section also describes the stakeout process that the WOAR application described in this thesis aims to facilitate.

2.1 Wearable Outdoor Augmented Reality

Wearable outdoor augmented reality is a combination of augmented reality and wearable computing. AR is introduced in section 2.1.1 before WOAR itself is introduced in section 2.1.2.

2.1.1 Augmented Reality

Augmented reality typically overlays virtual imagery onto the user's view of the physical environment in order to create the illusion of virtual objects being part of the real world [Azuma 1997]. While most AR applications are of a visual nature, a computer system may also augment a user's auditory or haptic environment. AR applications do not necessarily add virtual content to the real environment. They can also erase real objects from the user's view [Mann & Fung 1996] and manipulate the appearance of real objects [Mann 1994]. Removing real objects from the observer's view is called diminished reality, and manipulating the appearance of real objects is called mediated reality.

Augmented reality is commonly defined as having the following properties [Azuma et al. 2001]:

- “combines real and virtual objects in a real environment;

- runs interactively, and in real time; and

- registers (aligns) real and virtual objects with each other.”

[Milgram et al. 1994] situate AR between reality and virtual reality on their mixed reality continuum (figure 2.1). This continuum represents how much of the user’s environment is computer generated. For example, in the real world the user’s environment is not controlled by a computer, while in a virtual environment the user’s environment is completely created by a computer. Augmented reality lets users experience the real world as well as computer generated and controlled objects in one environment, allowing for completely new applications and interfaces.

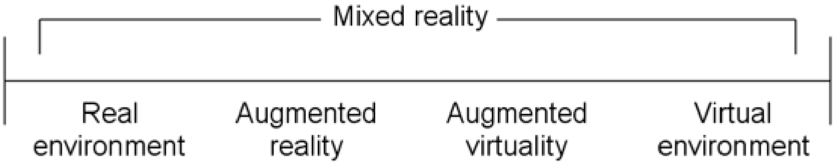


Figure 2.1: Milgram’s mixed reality continuum (after the adapted version from Azuma et al. [2001])

Since the first AR prototype demonstrated by Sutherland [Sutherland 1968] few AR applications have left the research labs. However prototype AR interfaces have been developed for domains as diverse as medical sciences [Peuchot et al. 1995], video conferencing [Billinghurst & Kato 2000], machine assembly and maintenance [Feiner et al. 1993], entertainment [Nilsen 2005], and education [Kaufmann & Schmalstieg 2002].

AR is mainly used to add virtual objects to the user’s real visual environment. To achieve this, the AR system tracks the user’s head position and orientation with respect to his or her environment. This information is then used to generate a corresponding view of the virtual objects. The virtual imagery is then overlaid on

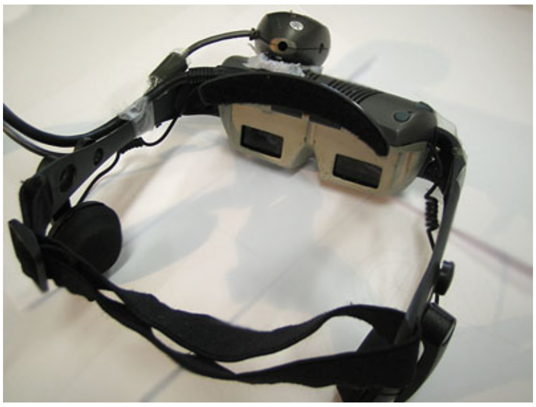


Figure 2.2: The i-glasses, a video HMD, with a webcam attached at the top.

top of the real world imagery in a head mounted display (HMD) or other display system through which the user sees their environment.

The two main HMD types for augmented reality are optical see-through and video see-through. Optical see-through HMDs typically use half-silvered mirrors to overlay computer generated imagery onto the user's view of the real world. Video see-through HMDs capture a view of the real world with a camera and combine it with computer generated imagery before displaying it to the user on small screens in front of the user's eyes (figure 2.2). The work described in this thesis is based on video see-through HMDs.

In an AR system with a video see-through HMD the system tracks the location and orientation of the video camera with respect to the real environment. It then uses this information to position and orientate the virtual camera in a 3D graphics environment to render a matching view of the virtual objects in real-time. This view is then combined with the camera image before it is presented to the user through the HMD (figure 2.3).

Different tracking technologies have been used by AR applications; all with a variety of underlying technologies, resolutions, accuracy and operating ranges. For example, **Magnetic tracking** systems such as the Flock of Birds [1] (figure 2.4(b)) generate a strong magnetic field. The corresponding Flock of Birds sensors can then sense their location and orientation within this field at an accuracy

[1] http://www.ascension-tech.com/products/flockofbirds.php

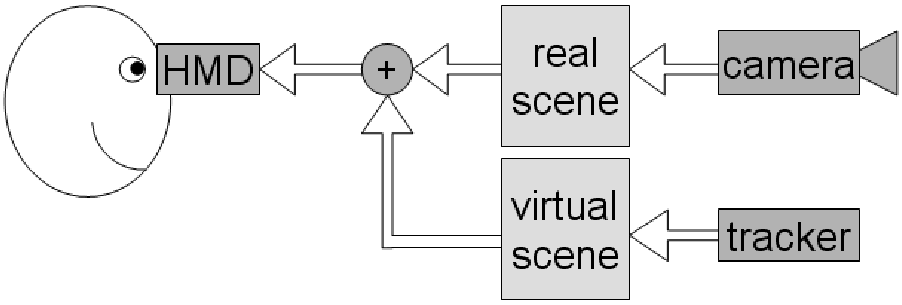


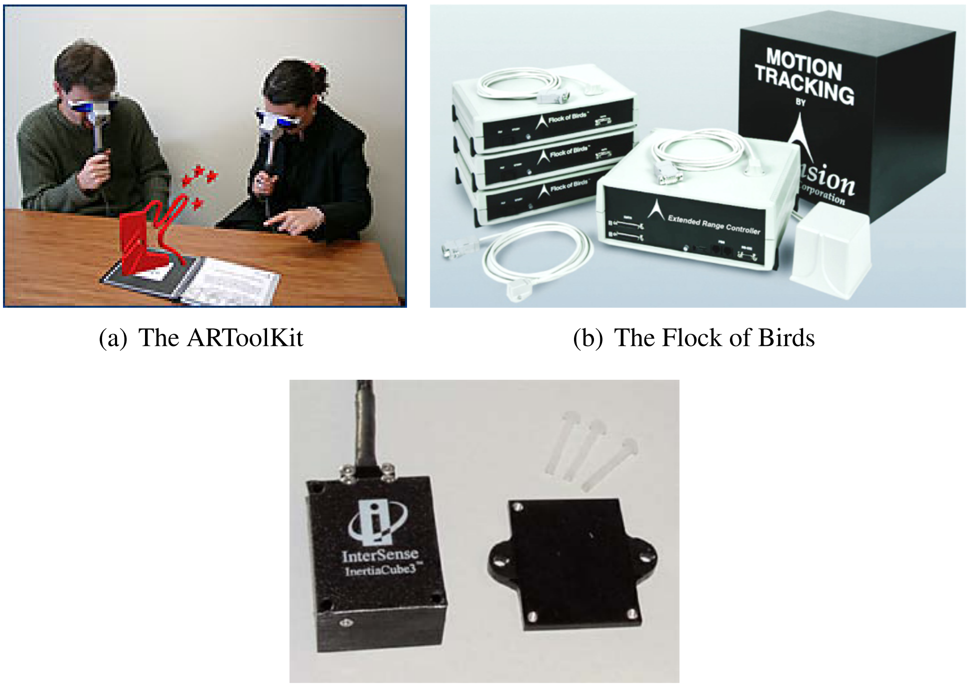
Figure 2.3: The basic components of visual augmented reality

of 1.8mm for position and 0.5° for orientation at a radius of up to 3m around the transmitter. **Inertial trackers** such as the Intersense IS 900 [2] can provide a positional accuracy of 3cm and an angular accuracy of 1° when used in combination with **ultrasonic tracking**. Inertial trackers must use dead-reckoning methods for position tracking, which means that they estimate the current position based on the previous position. This method accumulates measurement errors over time, introducing drift. The IS 900 relies on ultrasonic pulses sent out from devices placed in the environment to correct for this. The most common and affordable method for tracking is **vision based tracking**, as it generally does not require additional costly hardware. A popular vision based tracking system is ARToolKit [Kato & Billinghurst 1999]. ARToolKit uses the camera image captured to identify fiducial markers in the environment in order to determine the camera's position and orientation with respect to these markers (figure 2.4(a)). The accuracy for ARToolKit depends on the camera's distance and angle from the tracked marker [Malbezin et al. 2002, Abawi et al. 2004].

2.1.2 *Wearable Outdoor Augmented Reality*

This section describes the basic features and requirements for wearable outdoor augmented reality systems and reports on the range of research since the *Touring*

[2] http://www.isense.com/



(a) The ARToolKit

(b) The Flock of Birds

(c) The InertiaCube3

Figure 2.4: Tracking technologies

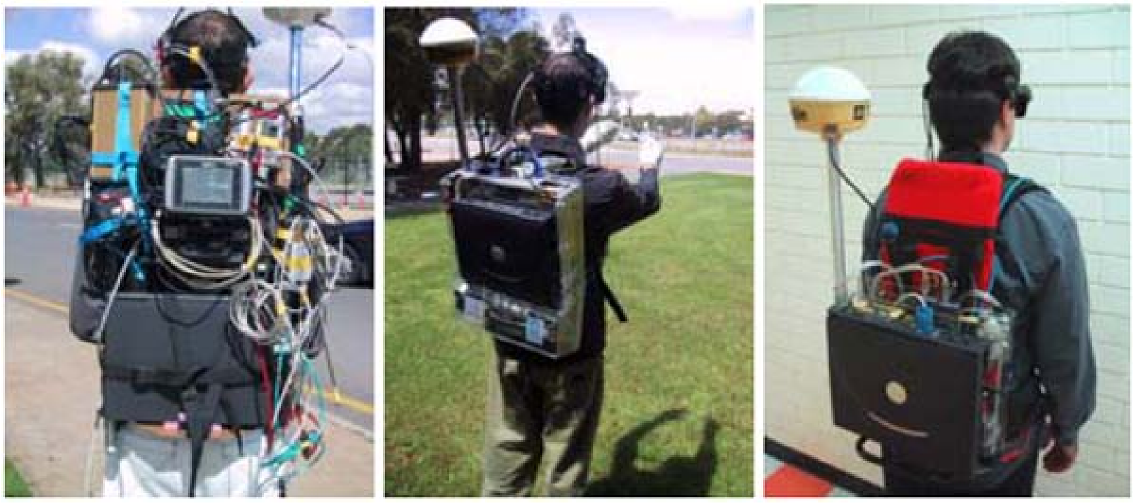
Machine [Feiner et al. 1997], the first WOAR system. The section first introduces wearable computing, describes the unique hardware and tracking requirements for WOAR systems, and finally gives an overview of previous WOAR implementations.

This thesis focuses on wearable AR systems as opposed to other mobile AR systems. Mobile AR is possible on platforms such as a handheld PC [Tsuda et al. 2005] or cell phones [Henrysson & Billinghurst 2007]. Wearable computing is a new research field that explores how computers can be worn on the body to provide constant access to computing power [Mann 1998]. Wearable computers are computers which are “subsumed into the personal space of the user, controlled by the user and [...] always on and always accessible.” [Mann 1998]. Wearable computers constantly assist the user or augment the user’s environment with information. An example of such a system is the *Remembrance Agent* [Starner et al. 1995] that was used to manually and automatically retrieve relevant textual information.

Using wearable computers as a platform for augmented reality enables a completely new set of applications. By adding a head-mounted display and tracking sensors to a wearable computer, users can move through the real world and have an augmented reality experience wherever they are. This produces an AR system that is not necessarily a tool built for a special purpose and fixed to a location. It becomes a commodity that enables novel applications and also brings new challenges with it [Starner et al. 1997].

Hardware Requirements

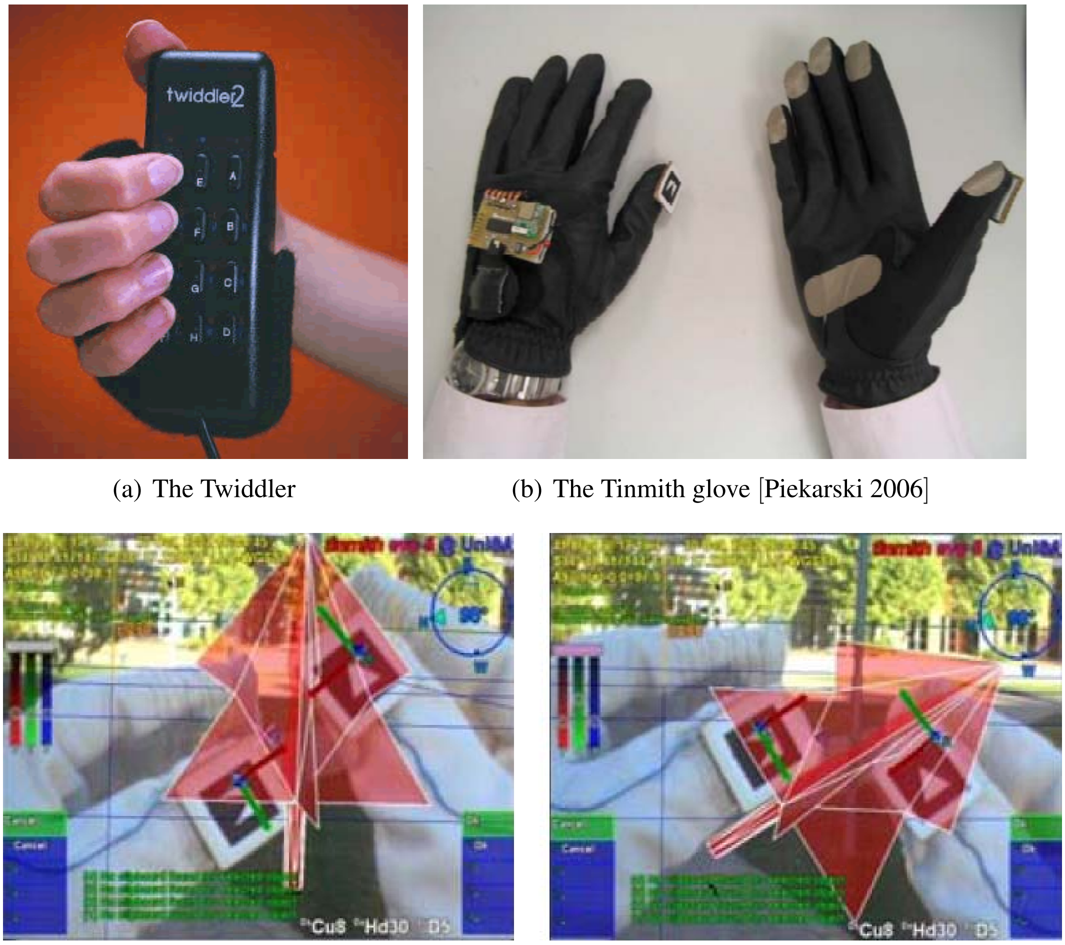
Wearable AR systems have different hardware requirements than desktop AR systems. Suitable computers are a compromise between size, weight, computing power and energy consumption. Figure 2.5 shows different versions of a typical hardware platform: the Tinmith system [Piekarski 2004]. To date, WOAR systems are heavy and bulky research tools that consist of a number of components. To truly become wearable and usable, the hardware will have to be reduced in size and weight.



(a)

Figure 2.5: The evolution of the Tinmith system [Piekarski et al. 2004]

In addition, traditional desktop computer input devices like the keyboard or the mouse have to be replaced or adapted. Custom input devices such as the Twiddler chording keypad (figure 2.6(a)) have been developed. Hand tracking that supports direct gesture based interaction with virtual objects has also been explored for wearable AR systems [Buchmann et al. 2004, Piekarski 2004, Smith et al. 2005] (figure 2.6(b)). Piekarski [2004] and Piekarski & Smith [2006] also used a glove based keyboard interface (figure 2.6(c)). Kölsch et al. [2006] used a combination of speech, hand gestures, trackball, and head pose for interaction with content in a WOAR application. Other possible input devices are touch pads [Thomas, Grimmer, Zucco & Milanese 2002] and gyroscopic mice [Zucco et al. 2005].



(a) The Twiddler

(b) The Tinmith glove [Piekarski 2006]

(c) Object manipulation with the Tinmith glove [Piekarski 2004]

Figure 2.6: WOAR input devices

In WOAR, special consideration has to be paid to the choice of head-mounted display. Navigating through an uncontrolled outdoor environment can be hazardous even in normal situations. Wearing a relatively heavy WOAR system with a head-mounted display raises a number of safety issues. An optical see-through HMD typically reduces the field of view (FOV) and reduces contrast. A video see-through HMD additionally:

- displays the environment at a drastically reduced resolution
- introduces lag
- removes depth cues (such as stereo-vision)

Optical see-through displays are potentially safer than video see-through displays, as they do not remove natural depth cues and they are less likely to completely block the user's vision – for example during a power failure. However, at this time, video see-through HMDs have a greater potential for high-quality augmented reality. This is because video see-through HMDs make it easier to achieve good registration and to avoid depth cue conflicts between real and virtual objects. With video see-through HMDs, it is possible to synchronise the imagery of the real world and virtual imagery better than with an optical video see-through system. This is because with optical see-through HMDs, there is no system delay for the real world imagery, while there is a delay for the virtual imagery due to tracking, processing, and rendering. The higher the system latency, the more virtual and real imagery may appear out of sync. With video see-through AR, the system can synchronise sensor data to ensure that real and virtual imagery match each other. In addition, some types of AR such as diminished reality are not possible with optical see-through HMDs. For these reasons, the presented original work is based on video see-through HMDs.

Tracking Technologies

Early wearable AR systems were not necessarily aware of their location. Instead they recognised markers on objects of interest to overlay virtual objects. Wearable AR systems such as Boeing's prototype wire bundle assembly application [Azuma 1997] allowed the user to move within their work area, tracking the user's position with respect to the workspace. A wider range of operation was supported by the *Remembrance Agent* system [Starner et al. 1997] that recognised LED tags placed in the environment. These tags could be used to overlay textual annotations or three-dimensional, correctly registered, maintenance instructions for printers. However, all virtual objects would only be directly associated with physical objects.

One great advantage of wearable AR systems is navigational support. A system that guides the user to a room is superior to a system that only informs the user that a meeting will held be held in that room. The approach of placing markers in the environment can be extended to navigation by placing markers not only on single items of interest but also on parts of the actual building such as walls [Reitmayr & Schmalstieg 2003]. This allows the system to not only guide the user through a buildingbut also allows for placing virtual objects inside a building, independent of individually tagged objects [Thomas et al. 2000].

The problem with this approach is to ensure that enough markers are visible to the system's camera or cameras at any one time. However, this means that a vast amount of individually different markers must be placed and calibrated in the environment. This is not only invasive and possibly a disruptive change in the environment, but also there is usually a limit of the number of individual markers that a system can process [Reitmayr & Schmalstieg 2003].

Wearable outdoor augmented reality is different from indoor wearable AR in several aspects. Typically, WOAR systems have to use different tracking techniques, have unique hardware requirements, operate in less confined areas, may require new interaction techniques, and enable different applications.

Indoor wearable AR systems typically use vision based tracking [Reitmayr & Schmalstieg 2003], sometimes combined with inertial tracking [Höllerer et al. 2001]. Vision based tracking significantly limits the area that an AR system can be used in as the environment needs to be prepared first. This means that either markers need to be placed in the environment [Thomas et al. 2000], or the system needs to be trained to recognise features of the environment [Vlahakis et al. 2002]. Typically, WOAR systems use GPS for position tracking and a combination of a compass and an inertial system for orientation tracking.

A WOAR system with a video see-through HMD needs to track the location and orientation of the camera in order to present the user with properly registered virtual objects. Accordingly, the orientation sensor is mounted rigidly onto

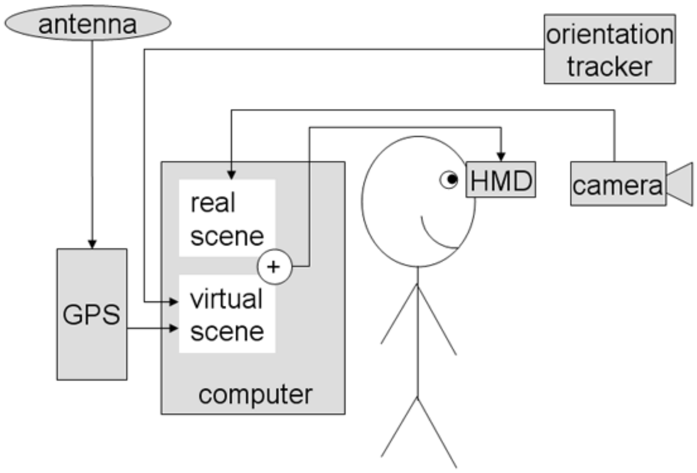


Figure 2.7: A schematic WOAR system with a video see-through HMD

the camera. The HMD must not move relatively to the camera, as the presented real imagery must match the user's head movements. To satisfy these requirements, WOAR systems mount the camera and the sensor rigidly on the HMD. The diagram seen in figure 2.7 is a typical setup for a WOAR system with a video see-through HMD.

For 3 degree of freedom (DOF) orientation tracking the TCM2 series [3] has been used in WOAR systems [Thomas et al. 1998, Suomela & Lehikoinen 2000, Gleue & Dähne 2001] as well as the InterSense IS-300 [4] [Höllerer, Pavlik & Feiner 1999, Julier et al. 2000, Thomas, Close, Donoghue, Squires, Bondi & Piekarski 2002]. Newer systems tend to employ the InterSense InertiaCube2 and 3 series [Cheok, Wan, Goh, Yang, Liu, Farzbiz & Li 2003, Reitmayr & Schmalstieg 2004, Avery et al. 2005, Schmeil & Broll 2007], notable for their tracking resolution and accuracy due to a combination of solid state magnetometers, accelerometers, and gyroscopes (figure 2.4(c)).

The majority of WOAR systems used differential GPS (DGPS) position tracking with accuracies ranging from 5 [Piekarski et al. 1999] to 1 meter [Feiner

[3] http://pnicorp.com

[4] http://www.isense.com/

et al. 1997], depending on the differential corrections service purchased. Exceptions are Höllerer, Pavlik & Feiner [1999] and Julier et al. [2000] who used real-time kinematic (RTK) differential GPS at centimeter level accuracy. Differential GPS requires the use of a base station that sends radio corrections to the GPS receiver. These radio corrections can be purchased as a service in many countries.

An accuracy of one meter is good enough to navigate through a city, and to overlay virtual objects on far away objects. For some applications – such as stakeout – greater accuracy is required. Some systems have therefore combined GPS and other position trackers to achieve greater accuracy at an affordable price. Stricker [2001] and Reitmayr & Drummond [2006] combined vision based feature-recognition with GPS data in order to accurately overlay virtual objects onto the real world imagery, and Cheok, Wan, Goh, Yang, Liu, Farzbiz & Li [2003] used a combination of GPS and inertial tracking.

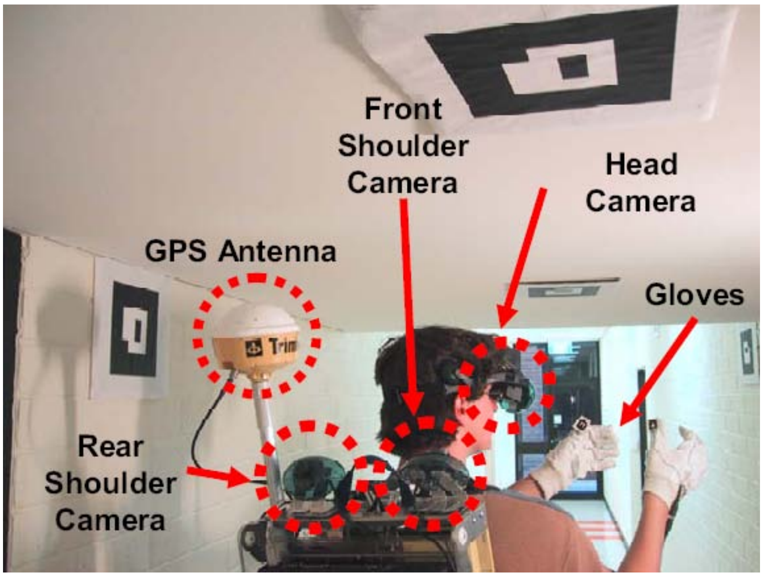


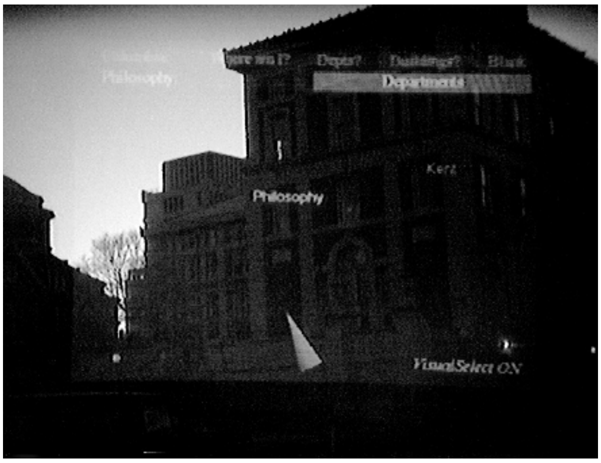
Figure 2.8: A system with combined indoor and outdoor tracking sensors [Piekarski 2004] (annotations by the original author)

Theoretically, the GPS antenna needs to be mounted rigidly onto the helmet just like the orientation sensor, as the system needs to track the camera's location. However, WOAR systems commonly mount the GPS antenna on a backpack (figure 2.5(a)) to minimize the weight on the helmet. At a GPS accuracy of about 1 meter, this will not make a significant difference.

Due to the different approaches to tracking, wearable AR systems usually work exclusively either indoors or outdoors. Notable exceptions are *ARQuake* [Thomas, Close, Donoghue, Squires, Bondi & Piekarski 2002] (figure 2.8), *GameCity* [Cheok et al. 2002] and *Human Pacman* [Cheok, Wan, Goh, Yang, Liu, Farzbiz & Li 2003]. These systems combine indoor and outdoor tracking sensors and work in both environments.

Applications and Features

A variety of WOAR applications have been built on the technologies described above. Since the *Touring Machine* (figure 2.9) [Feiner et al. 1997], the first WOAR system, several other systems have been implemented focusing on different applications – from campus tour guides to immersive games.

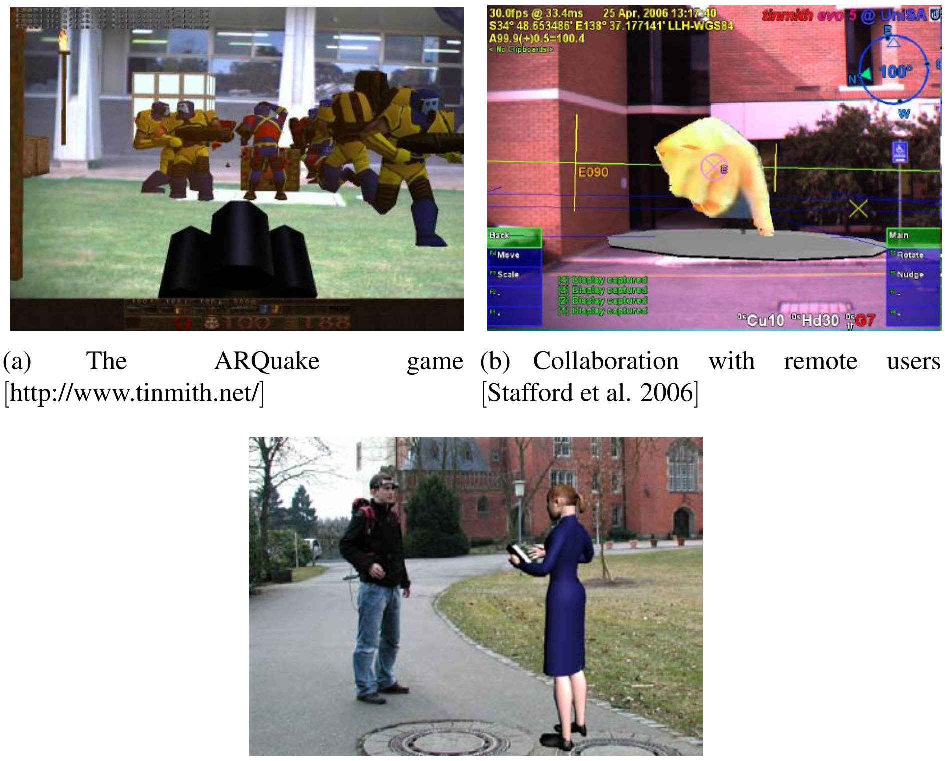


(a)

Figure 2.9: Screen capture of the Touring Machine [Feiner et al, 1997]

Navigation in the real environment is an obvious application for WOAR systems. The *Touring Machine* [Feiner et al. 1997], *map-in-the-hat* [Piekarski et al. 1999] and the *context compass* [Suomela & Lehikoinen 2000] guided users to a location using a compass metaphor. *WalkMap* [Suomela et al. 2003], based on the context compass project, used a map view for navigation. Non-visual cues such as audio are also possible, especially for helping visually impaired users navigate [Sephton 2001]. Navigation techniques will be discussed in depth in section 2.3.

Information browsing is supported by systems that make information about the user's real environment accessible. As users navigate through their environment they also navigate a vast information space that can be made accessible through their WOAR system. For example, the *Touring Machine* provided information about the buildings on the campus of Columbia University. The *situated documentaries* system was an extension of the same system that provided information about the history of the campus [Höllerer, Pavlik & Feiner 1999]. The tourist guide system by Reitmayr & Schmalstieg [2004] guided users to locations of interest in the City of Vienna and offered information on those locations on demand.



(a) The ARQuake game [http://www.tinmith.net/]

(b) Collaboration with remote users [Stafford et al. 2006]

(c) An avatar acting as a personal assistant [Schmeil et al. 2007]

Figure 2.10: Examples of WOAR applications

Architectural visualisations were part of these and other WOAR systems. With accurate registration, WOAR system could be used for architectural visualisations of hypothetical buildings [Satoh et al. 2001], proposed buildings [Thomas et al. 2000] or historic buildings [Höllerer, Pavlik & Feiner 1999]. For example, the *ARCHEOGUIDE* system visualised Greek temples on the site of their ruins at ancient Olympia [Vlahakis et al. 2001, Gleue & Dähne 2001].

Games have also been implemented on several WOAR systems – some only to demonstrate the capabilities of WOAR systems, others to explore the possibilities of WOAR gaming. *ARQuake* (figure 2.10(a)) was the first WOAR game, which ported the first person shooter Quake from the desktop computer into a real environment with virtual monsters [Thomas, Krul, Close & Piekarski 2002]. *Human Pacman* implemented the game Pac Man in a real environment with WOAR users taking over the roles of the Pac Man and the ghosts [Cheok, Wan, Goh, Yang, Liu, Farzbiz & Li 2003]. While these games served as demonstrations of WOAR capabilities on expensive systems, others addressed the trade-off between costly hardware and low-quality tracking and implemented simple games that are designed specifically for poor tracking conditions, such as Moon Lander [Avery et al. 2005] or Sky Invaders [Avery et al. 2006].

Collaboration has been explored in a few WOAR systems. Both the *ARQuake* and the *Human Pacman* games allowed for multiple users to play the game at the same time and communicate through an audio channel. The systems supported the collaboration of both WOAR and desktop users [Thomas, Close, Donoghue, Squires, Bondi & Piekarski 2002, Cheok, Fong, Goh, Yang, Liu & Farzbiz 2003]. The tourist guide system by Reitmayr and Schmalstieg allowed WOAR users to follow, guide and meet one another [Reitmayr & Schmalstieg 2004]. The "god-like interaction" interface by Stafford et al. [2006] allowed indoor tabletop users to place virtual versions of real objects into a WOAR user's environment in real-time (figure 2.10(b)).

Avatars are sometimes used for story telling or as an interface to the wearable system. They were used as virtual tour guides in some systems [Sephton et al. 1999, Herbst et al. 2007]. The *ARCHEOGUIDE* system showed virtual athletes competing in the ruins of the stadium of ancient Olympia [Vlahakis et al. 2002], while the *GEIST* system supported interactive story telling [Kretschmer et al. 2001]. In the GEIST system, users interacted with avatars who acted out historic scenes. The *LIFEPLUS* system used avatars to act out scenes from ancient Roman everyday life in the actual ruins of Pompeii [Papagiannakis et al. 2002]. *MARA* (figure 2.10(c)) was a virtual agent in the form of an avatar that could manage the user's appointments and provided the user with information about objects in the environment [Schmeil & Broll 2007].

Military applications as well as military funded systems also exist. For example, the *BARS* system explored techniques such as annotations for better situational awareness in urban environments [Julier et al. 2000]. Azuma et al. [2006] analysed the possible quality of outdoor AR tracking in combat situations.

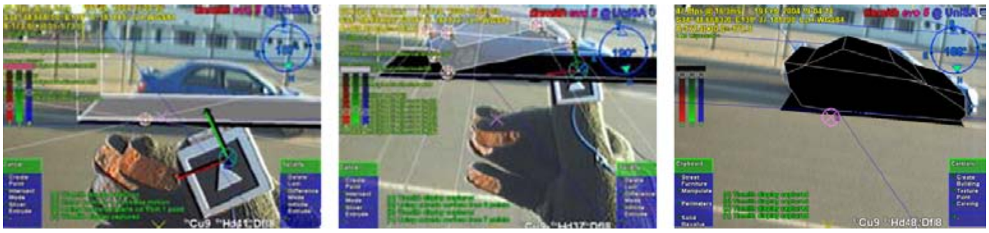


Figure 2.11: WOAR modelling with the Tinmith system [Piekarski 2004]

Modelling and Data acquisition in outdoor situations were also explored. Piekarski [2004] described a WOAR system that could be used to model objects in an outdoor environment, either to capture real objects or to create virtual objects from scratch (figure 2.11). Sephton [2002] described a system that may be used to capture and visualise ancient Maya cities.

User Studies

User studies are relatively rare in the field of WOAR applications. Only two informal and two formal evaluations of WOAR applications have been reported, while there were several formal evaluations of WOAR components such as navigational aids or depth perception.

Several formal user studies evaluated interface components for wearable or outdoor AR. The covered topics were navigation with visual, audio, and haptic cues [Billinghurst et al. 1998, Van Erp et al. 2005, Ross & Blasch 2000, Ross & Blasch 2002]; map views [Suomela et al. 2003]; and the visualisation of hypermedia structures and paths [Guven & Feiner 2006]. Depth cues and depth perception for mobile and outdoor AR were researched in formal user studies that covered depth cues and depth perception in the far field [Livingston et al. 2003, Wither & Höllerer 2005, Swan et al. 2006], and techniques for interaction at a distance [Wither & Höllerer 2004]. Furmanski et al. [2002] formally evaluated visualisation techniques for combining visible and obscured information in wearable AR.

The two reported informal studies of WOAR systems assessed the accuracy of the map-in-the-hat system [Thomas et al. 1998] and explored perceptual issues with the ARQuake system [Thomas, Krul, Close & Piekarski 2002].

The two formal user studies that evaluated WOAR applications focused exclusively on subjective measures. Cheok et al. [2004] performed a formal user study to find out positive and negative aspects of the Human Pacman system, and the overall enjoyment of playing the game. They found that users found the backpack too bulky and heavy, but that players enjoyed the game. Compared to a traditional "catch me" game, users preferred the WOAR game. Avery et al. [2006] formally evaluated their WOAR Sky Invaders game by measuring subjective feedback. Users enjoyed playing the game, and their enjoyment increased the more rounds they played. Users enjoyed playing the AR version of the game more than a PC version of it.

In conclusion, no rigid formal user study that assessed the objective performance of a WOAR application was described before.

2.1.3 WOAR Research Challenges

A variety of applications and interface features were explored for WOAR systems, but most of them were demonstrations or proofs of concept rather than robust working systems. This is mainly because of two reasons: the relatively low amount of research done in this area and the immaturity of the hardware used.

While WOAR systems have been around since at least 1997 [Feiner et al. 1997], most of the related research focused on inventing and exploring new interfaces for WOAR. None of these interfaces have emerged as a standard for WOAR, with most new WOAR systems implementing their own interface components from scratch. Standards or guidelines for WOAR user interfaces do not exist, and for most interface components they will not exist for several years as research on them has only just begun.

Using the *BRETAM* model [Gaines 1991] as an analogy, WOAR research is still in its infancy. The original BRETAM model described the process that the information sciences underwent on their way to maturity. "BRETAM" stands for the phases that new technology goes through:

- **B**reakthrough
- **R**eplication period
- **E**mpirical period
- **T**heoretical period
- **A**utomation period
- **M**aturity

Currently, most new WOAR systems explore new interfaces and applications, placing their work in the B phase. Some systems replicate the work of other WOAR systems, but empirical studies are very rare in the field of WOAR when for example compared to research on computer desktop interfaces. Following Gaines [1991], invention happens at the BR interface, while research is located at the RE interface. Product lines do not appear until the TA interface, and he locates low-cost products at the AM interface.

This thesis touches on the first three phases of the BRETAM model. It presents novel applications and interfaces such as WOAR stakeout and interaction with transparent tools. It replicates and builds on previous work such as information filtering, and presents empirical research to consolidate the research on navigation with WOAR systems.

Hardware quality and cost is a major factor in the success of WOAR systems. The relatively high cost of WOAR tracking systems is prohibitive for many research labs. RTK grade GPS systems typically cost at least US$ 10,000, while the *InertiaCube3* costs US$ 2,295 [5]. These devices are in the high-end range of WOAR tracking, but only deliver a positional accuracy of 1cm and an angular

[5] `http://www.i-glassesstore.com/inertiacube3.html`

accuracy of 1° with added jitter. To this, the cost of a head-mounted display – for example the *i-glasses*, a video HMD at a resolution of 800x600 pixels for US$ 999 [6] – and the cost of a laptop have to be added. Thus, the total system cost is approaching US$ 15,000 at least.

This thesis describes the most ambitious WOAR system so far with respect to tracking and calibration accuracy and compares the performance of a sample stakeout application to the performance of a commercially available stakeout system: the Trimble Survey Controller (see section 2.5). This study established the performance of a current WOAR system as compared to a real world industrial application.

The research presented in this thesis focuses on depth cues, navigation and obscured information visualisation. These concepts are described in detail in sections 2.2, 2.3, and 2.4.

2.2 Depth Cues in AR

Augmented Reality applications have to provide depth cues so that users can quickly, accurately and safely interact with virtual and real content. Incorrect depth cues may cause false depth perception [Drascic & Milgram 1996]. The perceived depth of objects can vary widely from their real depth, making AR systems less efficient and usable.

Providing correct depth cues is important in AR not only because the system needs to generate depth cues for the virtual content, but it often also reduces depth cues for the real environment. For example, many WOAR systems with video see-through HMDs remove stereo vision of the real environment, a very important depth cue. Thus, the task of an AR system is not only to generate appropriate depth cues for the virtual content, but where possible also to re-create depth cues for the real environment.

Besides factors such as cost and feasibility of implementation there are three main properties that have to be considered when discussing depth cues: resolution, scale and distortion.

- **Depth Resolution**: Each depth cue has its own resolution of depth, often

[6] `http://www.i-glassesstore.com/iglasses-pc-hr.html`

varying over distance. Cutting [1997] used the term "*just noticeable difference* (JND) in depth for two objects" to describe depth resolution. The term *zone of uncertainty* used by Drascic & Milgram [1996] is similar in that it describes a range where participants in their experiments were unable to judge which object was closer when presented with two objects at different depths.

- **Measurement Scale**: All depth cues are at least *ordinal*, meaning that they can be used to determine which of two objects is closer. Some may allow measurement at an *unanchored ratio*, meaning that they can be used to determine that object A is twice as far from object B than C is. Very few depth cues allow for *absolute* measures that indicate how far away an object is from an observer [Cutting 1997].

- **Distortion**: Perceived space can be distorted. While space within 2m distance from the observer is *Euclidian*, it is *affine* beyond that: distance along the visual axis (depth) is perceived differently from distance orthogonal to the visual axis [Cutting 1997, Wagner 1995]. In addition, haptic space within arms reach is *anisotropic*; it is distorted in several directions [Klatzky & Lederman 2003].

2.2.1 Visual Depth Cues

Figure 2.12 plots the depth resolution of several visual depth cues over distance. The plot shows that the quality of depth cues varies widely, especially over distance.

Ware [2000] used three categories for visual depth cues: *pictorial* depth cues are those that may be present in a static image, *kinetic* depth cues stem from movement of either the observer or objects, and *binocular* depth cues may be present if the observer's eyes are presented with different images.

Pictorial Depth Cues

The following are the most relevant pictorial depth cues:

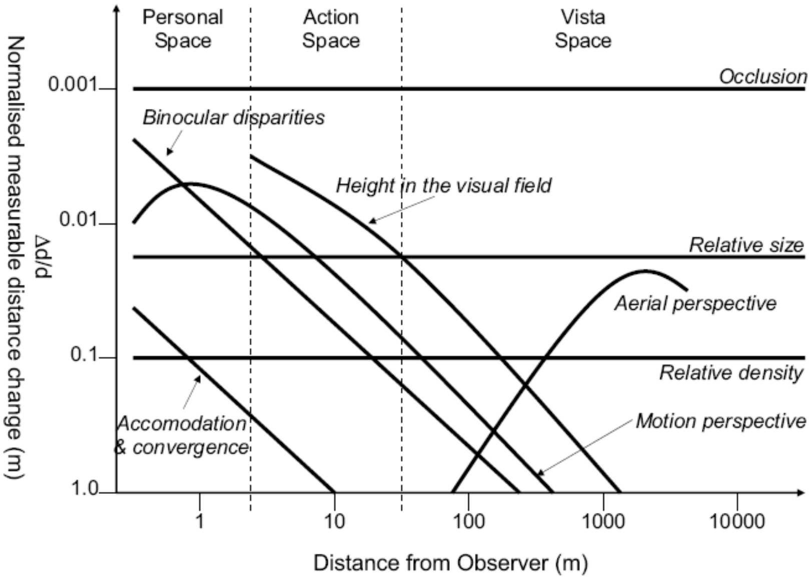


Figure 2.12: Depth cue resolutions according to Cutting [1997]. Figure taken from Piekarski [2004].

- *Occlusion* is the most important depth cue [Cutting 1997] and the most common source of depth cue conflict in AR [Drascic & Milgram 1996]. Although only of ordinal measurement, occlusion has the best just noticeable difference (JND) in figure 2.12.

- *Height in the visual field* is an ordinal or better depth cue [Cutting 1997] that is based on an assumption that the base of objects, when standing on the ground plane, will have a perceived height in the visual field according to their distance from the observer. Ware [2000] suggested that for floating objects, droplines should be drawn to the ground plane to artificially create this cue.

- *Cast shadows* have been shown to be an efficient depth cue when the light source was fixed and the shadow borders were soft [Kersten et al. 1997]. In the experience of the author, adding shadows to a scene greatly enhances the user's depth perception [Buchmann et al. 2004].

- *Relative size, relative density* and *texture gradients* are expected to be ordinal [Cutting 1997]. Relative size and relative density require a set of objects either at the same size or at a uniform spatial distribution. Texture gradients are a combination of the two first cues [Cutting 1997]. Note that in order to use these cues the actual size or distance of the objects does not need to be known.

- *Aerial perspective* is visible when away objects converge to the colour of the atmosphere, an effect that is imitated in computer graphics using fog [Ware 2000]. This cue is only effective at great distances and is hard to calibrate to match the real world aerial perspective at any given time.

- *Depth of focus* is used in photography and some paintings to give a sense of depth by blurring objects that are not close to the point of interest. The observer's focus changes constantly, making this depth cue expensive to implement [Ware 2000].

Kinetic Depth Cues

Kinetic depth cues arise from motion of an observer's head: objects in an observer's field of view appear to move at different speeds depending on their distance from the observer. Kinetic depth cues are at least as important as stereo vision [Ware 2000]. Ware [2000] reports that there are three relevant kinetic depth cues: *forward motion*, *motion parallax* and *kinetic depth*. The forward motion depth cue reveals depth information as the observer walks through a scene. Motion parallax reveals depth when the observer looks at an angle of 90 degrees to the direction of movement, for example as a passenger in a train. Kinetic depth refers to depth information resulting from object movement rather than observer movement.

Cutting [1997] said that kinetic depth will only reveal object shape but not their position in space. Accordingly, he included only forward motion and motion parallax under the term motion perspective. In AR, motion parallax may be of interest since the observer can easily move his or her head from side to side to get a better understanding of the scene. However, the observer would have to actively use this cue. Natural forward motion would be present in outdoor AR where the observer walks through an immersive AR scene. However, McCandless et al. [2000] found that a delay in video see-through HMDs deteriorated depth perception from motion parallax.

Binocular Depth Cues

Binocular Disparity “is the difference in relative position of an object as projected on the retinas of the two eyes” [Cutting 1997]. Small binocular disparity gives the strongest impression of depth and has been extensively researched. However, if the disparity becomes too big, it no longer results in *stereopsis* but in *diplopia*: the observer can no longer fuse the two impressions of the object.

Binocular disparity is a compelling depth cue for AR. It works well up to a distance of 30 meters and can be produced with stereoscopic HMDs. However, producing stereopsis in an HMD is not without problems. The distance between the HMD screens as well as the distance between the cameras will often not match the user’s interpupillary distance, and accommodation, a depth cue linked to stereopsis [Drascic & Milgram 1996] is not supported by modern HMDs.

2.2.2 Haptic Depth Cues

Haptic depth cues are relevant for directly interacting with real or virtual content. Haptic space is experienced due to *kinesthetic* feedback that indicates how joints are bent. However, there is indication that *cutaneous* feedback is also relevant. Dizio & Lackner [2002] found that visual feedback combined with touch helped participants to better learn an objects position than visual feedback alone. While kinesthetic feedback cannot be simulated in an AR system, cutaneous feedback can be. An example is given with the fingARtips system [Buchmann et al. 2004].

As Klatzky & Lederman [2003] pointed out, there is “no agreed-upon definition of haptic space”. They distinguish between *manipulatory space* which is in reach of the hand and *ambulatory space* which is beyond reach of the hand. Klatzky & Lederman [2003] state that manipulatory space is *anisotropic*. It is distorted according to several motor patterns, meaning the distortions are not solely dependent on the distance from the observer but also by how the observer’s hand was moved to a location.

2.2.3 Depth Cue Conflicts

The combination of real and virtual imagery in Augmented Reality can result in conflicting depth cues. With AR applications, it is not possible to control all depth cues, which may result in incorrect depth perception [Drascic & Milgram 1996].

Drascic & Milgram [1996] described four possible outcomes of conflicting depth cues: conflicts can be resolved either by one cue taking *precedence*, by *averaging* between them or by using *contingent information* such as the user's experience. If the conflict cannot be resolved the conflicting depth cues are *rivalrous* and the observer's depth perception is unstable, resulting in inaccuracy. Drascic & Milgram [1996] said that if one cue takes precedence, accurate positioning would be possible. Averaging would cause a constant error and using contingent information would cause inconsistent behaviour over longer periods. In addition, there can be calibration mismatches, where depth cues for real and virtual objects do not match up.

2.2.4 Depth Cues in Augmented Reality

Previous studies on AR and VR depth cues have mainly focused on space within arm's reach. Binocular disparity is especially important within this "personal space" [Cutting 1997]. For most depth cues, the JND increases (gets worse) with distance. During stakeout, interaction takes place just beyond the personal space, an area that has not yet been extensively researched in AR.

Wither & Höllerer [2005] are the exception to the rule, as they formally compared a set of pictorial depth cues for WOAR in the far field. They were interested in providing depth information so that users could accurately annotate far objects such as trees. They found that shadow planes [Wither & Höllerer 2004], and a map and colour coding of depth all improved the participants' depth judgment.

Hendrix & Barfield [1995] compared the performance of several depth cues and their combinations for a VR alignment task. They found that the addition of droplines had the most significant effect on accuracy. Droplines caused a 200% improvement in depth judgment, while virtual shadows only improved performance by 30%. Adding binocular disparity as a depth cue did not increase the depth judgment performance. However, it did increase the consistency of spatial judgments.

The experiment described by [Hubona et al. 1999] is different from most other VR depth experiments in that they measured both completion time and accuracy. Participants were asked to accurately place or resize objects in space, using shadows, stereoscopic viewing and different scene backgrounds. The experimenters found that stereoscopic viewing was both faster and more accurate than using virtual shadows. However, the experiment was flawed since they randomly changed the light source position between tasks, creating inconsistent shadow cues. The authors thus suggested using a stationary light position for future experiments.

Kersten et al. [1997] have researched cast shadows as a depth cue. They found that dark shadows had a significantly stronger effect than light shadows and that fuzzy shadows performed better than sharp shadows. They also found that the visual system relies on the assumption that the light source is stationary.

Ellis & Menges [1998] and Livingston et al. [2006] investigated the effects of calibration errors and the rivalry between real and virtual depth cues. Their results emphasize the importance of good calibration of an AR system that combines real and virtual depth cues.

In conclusion, AR systems should provide correct depth cues, as incorrect or conflicting depth cues will make it hard for the user to interpret spatial relationships in the image presented to them. Several depth cues of different nature, resolution and cost are available. Correctly interpreting spatial relationships of objects in the environment is relevant for navigation as well as interaction with objects such as aligning real and virtual objects.

2.3 Navigation

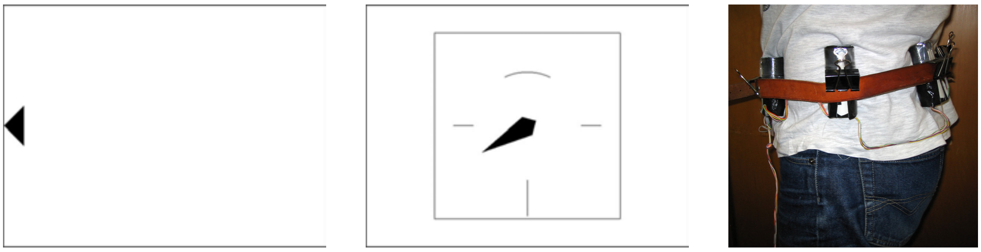
A WOAR system is used in a mobile context, with the user navigating the real environment. Thus, providing good navigation support is not only an opportunity for a WOAR system to assist the user but also a requirement. HMDs with a small field of view, low resolution and potential lag make it harder for the user to orient themselves in the environment. To avoid potential disorientation and to increase navigation efficiency, WOAR systems need an interface that supports the user's navigational tasks.

For navigation, this thesis focuses on interfaces that help the user turn themselves towards the direction of the next target. This is a subtask of navigation that has not yet been researched in detail by other researchers.

For the purpose of this thesis, it is assumed that route finding is computed by the system, and that the user is guided from one target location to another. Complex paths to targets may be broken down into segments with virtual or real waypoints that the user can easily walk to as for example in [Höllerer, Pavlik & Feiner 1999] and [Reitmayr & Schmalstieg 2004]. This is a natural and efficient way to navigate in known and unknown terrain. Virtual waypoints have been used in WOAR [Reitmayr & Schmalstieg 2004, Newman et al. 2001], while indoor AR applications have made use of highlighting real waypoints such as doors [Reitmayr & Schmalstieg 2003, Schmalstieg & Reitmayr 2005]. Waypoints can be combined with a path or trail on the ground to lead the user to the next waypoint [Höllerer, Feiner, Terauchi, Rashid & Hallaway 1999, Julier et al. 2002, Cheok, Wan, Goh, Yang, Liu, Farzbiz & Li 2003, Schmalstieg & Reitmayr 2005]. Most of these visualisations only show the next few path segments to give the user contextual information without cluttering the screen. Maps [Suomela & Lehikoinen 2000, Suomela et al. 2003, Vlahakis et al. 2002] or world-in-miniature representations [Reitmayr & Schmalstieg 2003, Schmalstieg & Reitmayr 2005] can also be used to give the user an overview of the route and the environment. Waypoints and paths connecting them are a natural and efficient way of providing navigation aid to the user.

In the case that the current waypoint is not in the user's field of view – for example behind them – previous WOAR systems used a variety of interfaces to aid the user to orient themselves towards the target. A comprehensive comparison of these interfaces is needed to establish which one is preferable. A factor that contributes to the use of directional interfaces in WOAR is the limited field of view that head mounted displays provide. For example, the video see-through HMDs used for the research described in this thesis provided only a 45 to 50 degrees horizontal field of view, significantly smaller than the typical human field of view of 180 degrees. This small virtual window into the real world causes the loss of context and makes it difficult for the user to see the next virtual waypoint as they walk towards the current one.

A variety of directional interfaces were previously described for wearable AR. They can be classified into three categories:



(a) No angular information: left/right arrows

(b) Indirect angular information: compass

(c) Direct angular information: haptic belt

Figure 2.13: Examples of interfaces that provide no, indirect or direct angular information.

- no angular information
- indirect angular information
- direct angular information

An example of an interface that provides no angular information are left and right arrows (see figure 2.13(a)) that only indicate if the user needs to turn clockwise or counter-clockwise [Thomas et al. 1998]. An interface that provides indirect angular information is a compass (see figure 2.13(b)) that informs the user how far they have to turn [Feiner et al. 1997]. Direct angular information can be provided by a haptic belt (see figure 2.13(c)) that taps the user from the correct real world direction removing the need for the user to map from a visual representation to the real world [Van Erp et al. 2005]. While it is easy to assume that the efficiency of interfaces will improve when they are more direct and provide more angular information, the experiment presented in chapter 3 explored if this is indeed the case.

Most directional interfaces are *head-stabilised*, meaning that their position on the screen is fixed and independent of the user's head or body movements. This thesis distinguishes between two types of head stabilised interfaces: *head-up displays* (HUDs), which are two-dimensional interfaces drawn directly on the screen as for example used in [Suomela & Lehikoinen 2000], and *perspective interfaces*, which are presented to the user at an angle as used in [Suomela et al. 2003]. Objects are *body-stabilised* if they appear fixed to a person's body instead of the person's head. A haptic belt is an example of a body-stabilised interface. Normal AR objects are *world-stabilised*; they are correctly registered in 3D and appear as part of the real environment. A head-stabilised interface has the advantage of always being visible, regardless of head orientation. A world-stabilised interface has the advantage of directly showing a location in the real world.

Left/right arrows are the most simple directional visualisation (see figure 2.13(a)). These arrows are a head-up display and indicate whether the user has to rotate left or right to find the target [Billinghurst et al. 1998, Thomas et al. 1998, Suomela & Lehikoinen 2000, Suomela et al. 2003]. However, the arrows do not provide angular information. They could for example convey angular information through their length. However, this would make them equivalent to a horizontal compass.

Turning signals are arrows that indicate whether the user has to turn left, right, go forward or turn around 180° [Reitmayr & Schmalstieg 2003, Newman et al. 2001]. Left and right turns are traditionally visualised as bent arrows. Turning signals are more appropriate for navigating a grid-like environment with 90° turns such as buildings or streets, while left/right arrows may be used for arbitrary directions.

Maps have been used in WOAR applications to display the user's position and orientation. A *forward-up* map [Suomela et al. 2003, Höllerer, Feiner, Terauchi, Rashid & Hallaway 1999] is automatically rotated so that the user's direction is aligned with the "up" direction of the map. A *north-up* map [Vlahakis et al. 2002] is aligned with North and is not rotated. Darken & Cevik [1999] found that forward-up maps are more efficient for simple target finding where efficiency os measure din search time and errors, while north-up maps are more efficient for tasks that require context. Maps are usually head-stabilized although there have also been hand-stabilized implementations [Höllerer, Feiner, Terauchi, Rashid & Hallaway 1999, Schmalstieg & Reitmayr 2005]. Some of the systems use a HUD map, while others use a perspective map. All of these map types provide indirect angular information. Suomela et al. [2003] compared HUD and perspective maps and found that HUD maps were more efficient. However, they hypothesized that this was due to the low resolution of their display. Some of the maps used are two-dimensional [Vlahakis et al. 2002], while others followed the World-In-Miniature metaphor, and used three-dimensional maps [Schmalstieg & Reitmayr 2005].

A *compass* has been used in several WOAR applications [Schmalstieg & Reitmayr 2005, Newman et al. 2001] and models a familiar tool. In the context of this thesis, a compass is forward-up, and the needle does not point to North but to the target. It provides indirect angular information. The compass needle described by Feiner et al. [1997] turned red if the target was more than 90° away from the user's orientation.

A *horizontal compass* maps a part or the whole range of a compass onto a horizontal line with the centre of the line being aligned with the user's line of sight [Thomas et al. 1998, Suomela & Lehikoinen 2000]. The compass needle moves horizontally along the compass and provides indirect angular information. The map-in-the-hat system's compass [Thomas et al. 1998] only covered the camera's field of view, while the context compass [Suomela & Lehikoinen 2000] covered about twice the camera's field of view. Targets outside the compass were indicated with left/right arrows.

Audio cues fall into two distinct categories; spatial audio which simulates a sound beacon at the location of the target [Billinghurst et al. 1998] and speech interfaces with pre-recorded [Sephton 2003] or synthesized [Ross & Blasch 2000] speech. Billinghurst et al. [1998] used a continuous white noise loop as their audio beacon. Ross & Blasch [2000] used an audio beacon as well as synthesized speech to inform severely visually impaired users about the heading of the target in degrees or clock face positions. The beacons were played back every 700ms while the speech output was played back every 1700ms.

Haptic cues have the advantage of not taking up screen real estate while not being as susceptible to interference by environmental noise as sound cues are [Ross & Blasch 2000]. Ross & Blasch [2000] described a turning signals interface based on the "rabbit" display [Tan & Pentland 1997]. The interface tapped the user's shoulders to indicate left and right turns. It tapped the user's neck to indicate that the target was straight ahead. In each case, double taps were made every 700ms. After user testing, Ross & Blasch [2000] suggested that the tapping frequency for the neck should be reduced. Van Erp et al. [2005] tested a belt with eight vibrators and found that their participants were able to use it for waypoint navigation. Zelek & Holbein [2006] proposed a haptic belt with four haptic actuators to indicate forward, backward, left and right turning signals.

There has not yet been a comprehensive description or comparison of these different directional interfaces and most authors do not say how well they performed. The studies by Billinghurst et al. [1998] and Ross & Blasch [2000] are noteworthy exceptions. Billinghurst et al. [1998] compared the efficiency of HUD left/right arrows and an audio beacon for a rotation task. Each of these two cues alone, as well as in combination helped participants perform faster in a search task than when no cue was present. There was no difference between the performance of either cue or their combination.

Ross & Blasch [2002] compared three directional interfaces for users with severe visual impairments: a haptic interface and two audio interfaces. The haptic interface tapped the user on the shoulder for changes in direction and tapped the neck for forward movements. This was compared to a spatial audio beacon and a speech interface that informed the user every 1.7 seconds how far off course the target was, using either degrees or clock face positions to do so. All three interfaces were compared for their performance as aids for crossing the road, with dependent measures such as walking pace, veering, and hesitations. Users preferred the haptic interface, which also performed best for most objective measures. The audio interfaces did not work well in noisy environments.

Haptic belts have not yet been compared to other directional interfaces, but there was a study evaluating the feasibility of such a device. The study found that participants were able to use the belt to follow a complex path without any visual or acoustic help [Van Erp et al. 2005].

As can be seen, the developers of wearable AR systems have found it necessary to integrate directional interfaces into their systems. A wide range of possible interfaces have been used with little or no assessment of their efficiency. Unlike previous work, this thesis describes several different modalities of interfaces for providing directional cues and provides a rigorous user study evaluating these interfaces in chapter 6.

2.4 Obscured Information Visualisation

Obscured Information Visualisation (OIV) enables WOAR users to see obscured or hidden objects in their environment [Furmanski et al. 2002]. For example, allowing a person to see a representation of an object that is obscured by a building. This is especially relevant for WOAR applications, as users navigate large areas with overlapping real and virtual information. MacIntyre & Feiner [1996] introduced the concept of *environment management* as opposed to the desktop concept of window management. An environment manager organises the information displayed to the user so that the display is not cluttered and important information is not occluded. This can for example be achieved by making sure that object labels do not overlap or that virtual objects do not occlude real people. In a WOAR application the user's environment becomes part of the user interface, and the system has to manage it to present the user with an optimal view of the information that surrounds them. This view is free of clutter or information overload. To make it explicit, "virtual and real objects alike are considered part of the interface" [Höllerer 2004, page 3]. Much of that environment is not controlled by the WOAR application as it is the user's real environment. WOAR systems cannot manage the real environment by physically altering it. However, they can control how the environment is displayed to the user, and OIV is one approach that may be used for this.

There are four types of information occlusion that can occur in augmented reality:

1: Virtual objects occluding real objects

2: Virtual objects occluding other virtual objects

3: Real objects occluding virtual objects

4: Real objects occluding other real objects

Note that the last two are often interchangeable, as sometimes only approximations of the occluded real objects can be displayed in the form of virtual objects. An example of this is the 3D reconstruction of a foetus in the mother's womb [State et al. 1994].

There have been several OIV implementations for AR, with only some WOAR implementations. State et al. [1994] and Bajura et al. [1992] visualised ultrasound imagery within a patient to guide needle biopsies. Webster et al. [1996] visualized hidden architectural structures in indoor AR.

OIV have also been used in WOAR applications such as the ones described in this paragraph. Tsuda et al. [2005] used imagery from a surveillance camera to make an area occluded by a building visible to the user in a mobile outdoor AR system. Bane & Höllerer [2004] created tools that allowed users of their WOAR system to see virtual representations of rooms and heat distributions in buildings. Avery et al. [2007] used a remote controlled robot to capture live images of occluded objects that were then mapped onto 3D models of the occluded objects to synthesize the user's point of view. This system did not use transparency to visualise the obscured data, as this would "clutter the display" [Avery et al. 2007]. Instead, the system used the hand-tracking functionality of the Tinmith system [Piekarski 2004] to allow the user to define a tunnel cutaway. Tunnel cutaways are discussed in section 2.4.1. Kalkofen et al. [2007] investigated obscured information visualisation in combination with magic lenses.

The main challenges that OIV has to overcome are:

- Perceptual depth ordering problems [Furmanski et al. 2002, Livingston et al. 2003]
- "Superman's X-ray vision" problem [Livingston et al. 2003]
- Recognition of occluding real objects and reconstruction of occluded real objects [State et al. 1994, Mourgues et al. 2001, Tsuda et al. 2005]

These problems are discussed in the following sections.

2.4.1 Perception of Depth Ordering

Perceptual depth ordering problems are inherent in the visualisation of obscured information. As previously mentioned in section 2.2.1, occlusion is one of the most important depth cues in human perception. Simply making obscured objects visible can make them appear to be in front of the occluding object, as Furmanski et al. [2002] have found in a preliminary study. They found that virtual objects are most often perceived as overlaying real objects, even when they are partially occluded by real objects.

There have been two main approaches to this problem: transparency and cutaways. This section first discusses transparent objects, and then cutaways.

Livingston et al. [2003] investigated how best to visualise a virtual object in front of or behind one or two real obstacles in the far field. They found that the addition of a ground plane greatly helped the participants to correctly resolve the depth ordering correctly. In the absence of a ground plane, a combination of wireframe graphics, solid fill, and decreasing opacity and intensity with distance was as powerful. They believed that the wireframe helped their participants to understand the shape of the occluded objects, while the solid fill of the object aided in understanding depth ordering. Tsuda et al. [2005] investigated perception in the depth ordering of two real objects for mobile outdoor AR. They found that a combination of wireframe, solid fill, and a ground grid was best at conveying the location of an occluding object. However, their application was tablet based and thus not a WOAR system. They argued that for a tablet based application the occluding object could be made completely invisible, as it was still visible to the user in the real world.

Cutaways come in two main varieties: stationary and as a tunnel. Stationary cutaways cut a volume out of the real object that stays stationary with respect to the object when the user moves. This allows the user to see the borders of the intersection. Tunnel cutaways move with the observer and do not show any intersection borders. The advantage of stationary cutaways is that they give the user a clear perception of depth and a clear relation between the different objects. The advantage of the tunnel view is that it moves with the user, allowing for easier inspection of occluded objects. The perception of depth ordering for stationary cutaways has been investigated in a preliminary study [Furmanski et al. 2002]. Participants perceived a virtual object to be in front of a real object, even if it was not. Stationary cutaway views have been used to visualise ultrasound in patients [State et al. 1994, State et al. 1996, Bajura et al. 1992], while tunnel views have been explored in WOAR to see inside a building [Bane & Höllerer 2004].

2.4.2 “Superman’s X-ray vision” problem

“Superman’s X-ray vision” problem deals with the problem of how many and which layers of occluded objects should be presented to the user [Livingston et al. 2003]. While the appropriate use of pictorial cues such as wireframe and solid fill for transparency based OIV can reduce this problem, the WOAR application still has to make decisions on which objects to display to provide the user with context and which objects to suppress so as to avoid clutter and information overload.

One way to do this is by giving the user tools that they can use to actively select cutaways. For example, Bane & Höllerer [2004] developed an outdoor AR system that allows for X-ray vision of a building. The system provided users with a tunnel tool and a room selector tool. The tunnel tool produced a tunnel cutaway as described in the previous section. The cutaway was head-stabilized and centred in the user's view. Users could control the depth of the tunnel to limit the amount of X-ray information presented to them. However, this resulted in clipping artifacts with the rooms' walls when the end of the tunnel intersected with them. The room selector tool eliminated this problem by letting users select single rooms that would then be displayed in their entirety. Bane & Höllerer [2004] found that the tunnel tool was appropriate for exploring volumetric data, while the room selector was much more appropriate for the domain of the building.

Automated methods, such as knowledge-based AR [Feiner et al. 1993] or information filtering [Julier et al. 2000], could be used to display obstructed objects automatically. In an outdoor AR environment the system should take as much load off the user as possible so that the user can interact with the environment and not with the actual system. In a WOAR stakeout application, for example, the user will already carry a stake or a similar tool. This means that the system should work as automatically as possible without requiring the user to manipulate input devices. Julier et al. [2004] described a software architecture for adaptive user interfaces with urban WOAR systems and X-ray vision in mind that is able to manage such visualisations automatically.

2.4.3 Recognition of Occluding Real Objects and Reconstruction of Occluded Real Objects

The two major implementation problems for artificial transparency of real objects are

(1) the identification of the relevant objects in the current frame and

(2) the reconstruction of the obstructed background.

A range of diminished reality (DR) papers have studied how to hide real objects from the user's view by replacing them with their estimated background. Diminished reality is a type of augmented reality that aims at removing real objects from the observer's view. Most authors are concerned with making real objects completely transparent, while very few aim at partial transparency. They are also concerned with hiding static objects such as buildings [Klinker et al. 2001] or monuments [Lepetit & Berger 2001], but the possibility of removing instruments from a surgeon's view has also been explored [Mourgues et al. 2001].

If occluded real objects should be presented to the user, then they need to be reconstructed first. A relatively simple method is to have a second camera placed in the environment that captures the occluded object in real-time [Tsuda et al. 2005]. However, such an approach is limited to those locations covered by the camera. State et al. [1994] reconstructed a 3D model of a foetus from a set of ultrasound images. However, this approach did not work in real-time. Methods used for generating the occluded background include additional cameras [Zokai et al. 2003, Kameda et al. 2004], dense temporal sequences of images captured from a camera [Lepetit & Berger 2001] and stereoscopic endoscopic views [Mourgues et al. 2001]. Modelling occluded objects in advance as completely virtual objects does not solve this problem either, as no real-time information can be presented to the user. For example, it is possible to model the rooms of a building so the rooms can be seen in X-ray vision [Bane & Höllerer 2004]. However, the models will not represent the current state of the rooms such as the current furniture configuration or the people in it.

In conclusion, the visualisation of obscured information can help users understand their environment and interact with it better. While OIV promises easier interaction and better understanding of the user's environment, it brings with it non-trivial challenges of perceptual and technical nature. While some types of OIV are simple to solve – such as the occlusion by virtual objects – other types – such as the occlusion of real objects by other real objects – are significantly harder to implement. Chapter 5 presents implementations for each of the four types of OIV that are solutions for WOAR stakeout.

2.5 Stakeout With the Trimble Survey Controller

This thesis is motivated by implementing a stakeout application in a WOAR system. The Trimble Survey Controller (TSC) is a state-of-the-art commercial stakeout system that this research compared performance with the WOAR system against.

This section describes the general process of staking out a road design with the TSC with focus on the most relevant processes and interface components. The TSC guides the surveyor through the process of staking out a road and also provides functionality for other surveying tasks.

Road stakeout is the process of transferring a virtual three-dimensional road model into the real world in order to guide road workers. These road designs are typically created in an office and are based on the topology of the site previously surveyed with a resolution of 20 meters. During the stakeout process the surveyor places stakes at positions indicated by the design. These stakes are then used by the builders to form the terrain and build the road correctly.

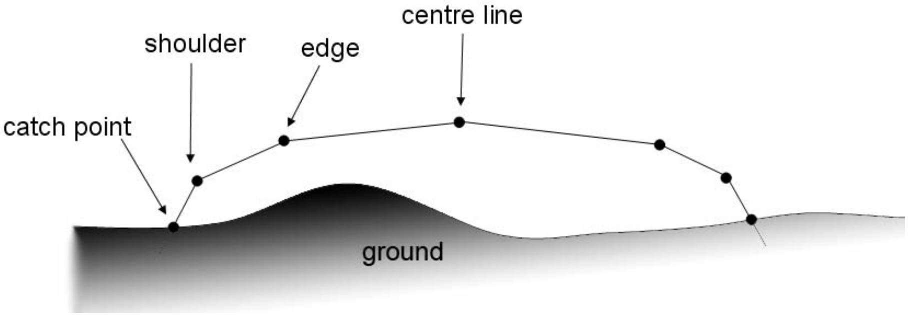
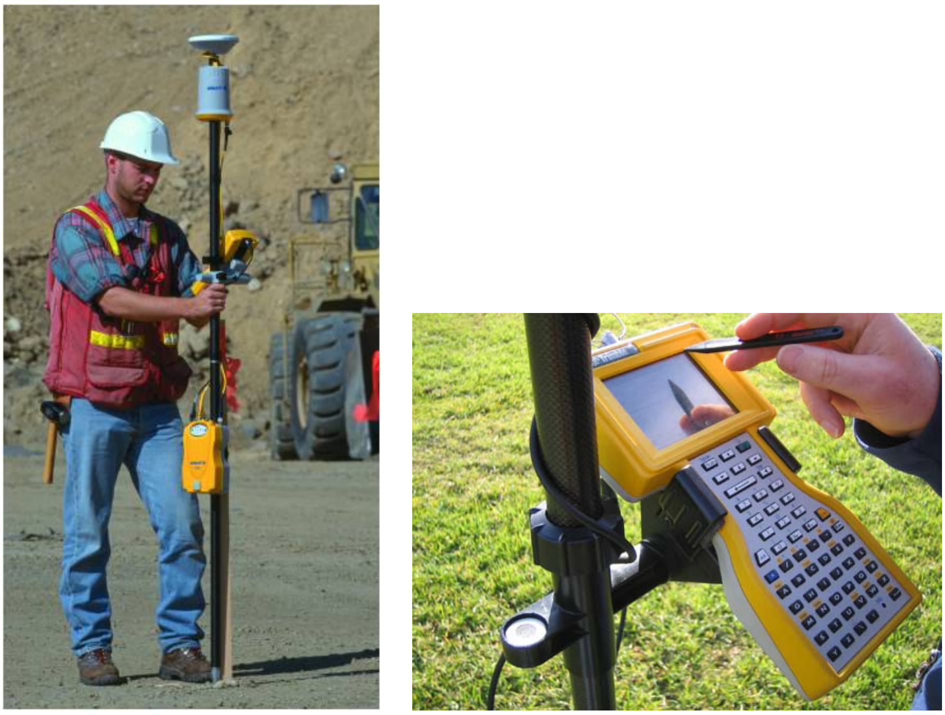


Figure 2.14: Components of a road cross section

A road design consists of several key elements. The *centre line* is a chain of straight and curved lines. *Cross-sections* of the road are defined at *stations* on the centre line, usually at two meter distance from each other. Depending on the type and complexity of a road, its cross-sections will consist of a set of *offsets* such as centre line (offset 0), edge, shoulder, and catch point (figure 2.14).

2.5.1 The Device

The Trimble Survey Controller runs on a custom made mobile PC (figure 2.15(b)) using the Windows CE operating system. For surveying, the TSC mobile PC is attached to a surveying pole (figure 2.15(a)) along with the GPS antenna at



(a) The TSC in the field

(b) The TSC

Figure 2.15: The Trimble Survey Controller hardware

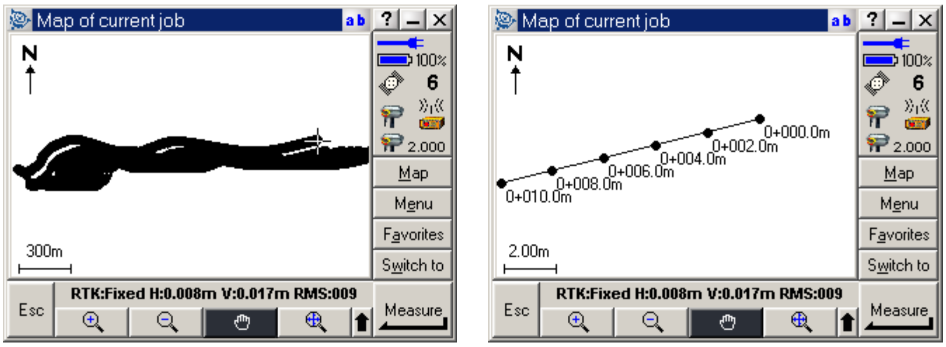
the top of the pole and the Trimble RTK GPS receiver. The equipment in this configuration weighs 4.28kg.

2.5.2 The User Interface

The road's centre line could be displayed graphically as shown in figure 2.16(a). Figure 2.16(b) shows a zoomed in version of the same map.

A station can be selected either on the overview map by clicking on it, or by selecting it from a drop-down list (figure 2.17(a)). In the sample road used for the screenshots, a cross section is placed every two meters along the centre line. An offset on a cross section can be selected from a drop-down list in a different form (figure 2.17(b)).

The TSC then displays a compass view that directs the user to the selected position (figure 2.18(a)). This visualisation indicates the direction and distance of



RTK:Fixed H:0.008m V:0.017m RMS:009

RTK:Fixed H:0.008m V:0.017m RMS:009

(a) A road. The station labels do not zoom well in SC and form “parallel roads” below the actual centre line.

(b) A magnified view of the same road. The first six stations are visible.

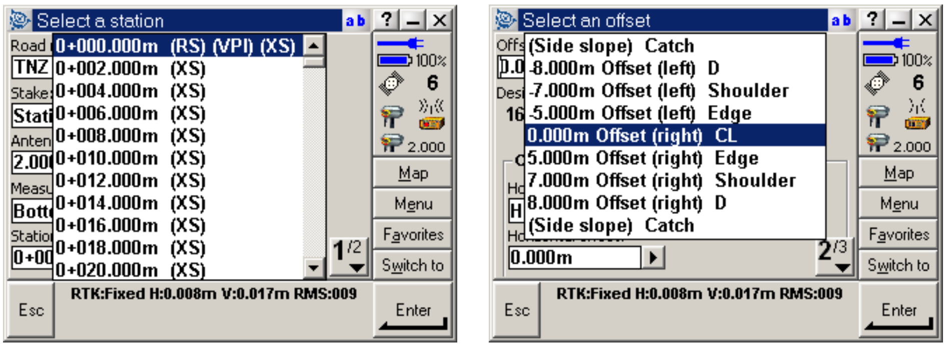
Figure 2.16: A road model displayed in the Survey Controller

the selected point. On the right hand side of the compass, distances in two different coordinate systems are displayed. One is aligned with North and West, while the other is aligned with the road. The directions *Backward* and *Forward* follow the centre while *Right* and *Left* are identical with a person’s right and left when standing on the centre line facing Forward. When the user is within 2 meters of the point, the screen changes as shown in figure 2.18(b). The cross in the centre of the screen is the user’s current position, and the two concentric circles indicate the target position. The long line in the circles points in the road’s Forward direction and the short line points to the centre line. This map is forward-up, meaning it is rotated according to the user’s heading

2.5.3 Usability Issues

There are a number of usability issues associated with using the TSC for stakeout. The surveying pole has to be held accurately in a vertical position. The GPS position is measured for the antenna, which is mounted on the top of the pole. To calculate the actually surveyed position, the TSC subtracts the length of the pole from the GPS location, assuming that the pole is held perfectly vertical. A small water-level embedded in a pole helps the surveyor to keep it vertical (see figure 2.19)

The TSC has to infer the user’s heading from GPS data, which is only posi-



Select a station

0+000.000m (RS) (VPI) (XS)
0+002.000m (XS)
0+004.000m (XS)
0+006.000m (XS)
0+008.000m (XS)
0+010.000m (XS)
0+012.000m (XS)
0+014.000m (XS)
0+016.000m (XS)
0+018.000m (XS)
0+020.000m (XS)

RTK:Fixed H:0.008m V:0.017m RMS:009

Select an offset

(Side slope) Catch
-8.000m Offset (left) D
-7.000m Offset (left) Shoulder
-5.000m Offset (left) Edge
0.000m Offset (right) CL
5.000m Offset (right) Edge
7.000m Offset (right) Shoulder
8.000m Offset (right) D
(Side slope) Catch

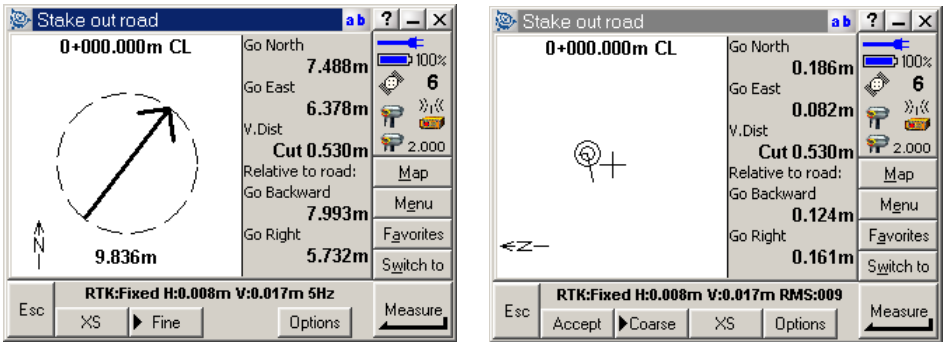
0.000m

RTK:Fixed H:0.008m V:0.017m RMS:009

(a) Stations can be selected from a drop-down list

(b) Selecting an offset.

Figure 2.17: Drop-down lists are used to select a point for stakeout.



Stake out road

0+000.000m CL

Go North 7.488m
Go East 6.378m
V.Dist Cut 0.530m
Relative to road:
Go Backward 7.993m
Go Right 5.732m

9.836m

RTK:Fixed H:0.008m V:0.017m 5Hz

Stake out road

0+000.000m CL

Go North 0.186m
Go East 0.082m
V.Dist Cut 0.530m
Relative to road:
Go Backward 0.124m
Go Right 0.161m

RTK:Fixed H:0.008m V:0.017m RMS:009

(a) A compass directs the user to a far away point

(b) At a closer distance, the user and the target point are visualised on a map.

Figure 2.18: The two screens that support navigation to a target and accurate pole placement.

tional. Accordingly, the directional information shown in figure 2.18(a) and the rotation of the map shown in figure 2.18(b) are no longer accurate when the user changes direction or rotates the surveying pole. The displayed direction is corrected after a meter or two of forward motion. This is not a problem when the user is far away from the target point, as the user will only swerve a little off-course. However, close to the target point, this can lead to the “RTK dance”: back and forth movement due to overshooting the target.

These restrictions mean that the user has to hold the pole vertically when mea-

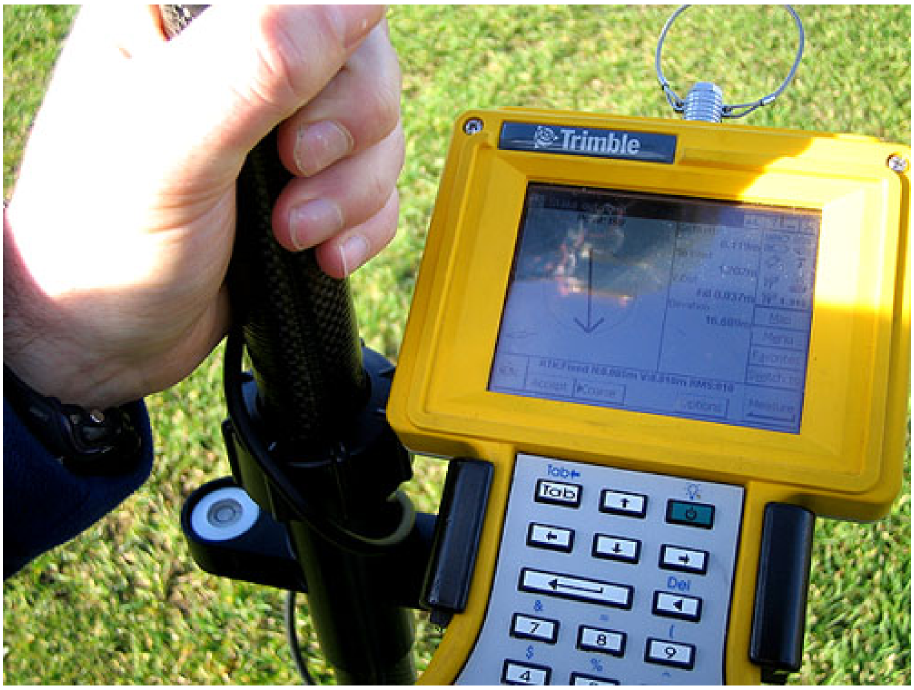


Figure 2.19: The TSC attached to a surveying pole seen from the user's perspective. The water level is the white circle to the left of the pole.

suring a point, and they cannot turn the pole with respect to world coordinates during stakeout. During navigation, the surveyor has to monitor the display for guidance, the ground for safe navigation and the environment for possible hazards. During staking, the surveyor has to monitor both the water level and the display.

2.6 Conclusion

This chapter describes the basic concepts of wearable outdoor augmented reality and introduces depth cues, directional interfaces and obscured information visualisation in WOAR. It also describes the stakeout process with the Trimble Survey Controller, a leading commercial system for outdoor surveying.

WOAR technology allows the development of novel applications. Current applications are explorative implementations that were not formally compared to their commercial conventional equivalents. This thesis describes the development of a WOAR stakeout application and a formal user study that compared the performance of the developed application to that of the conventional Trimble Survey Controller.

Appropriate depth cues and directional interfaces may make a WOAR stakeout application more efficient. No depth cue study at a distance of two meters distance from the user's eye in AR was described in the literature. Similarly, no comprehensive study of directional interfaces had been undertaken. This thesis describes formal user studies that evaluated the efficiency of depth cues for stakeout and the efficiency of directional interfaces for WOAR navigation.

The next chapter investigates navigation with WOAR systems.

Chapter III

Directional Interfaces for WOAR Navigation

This chapter presents implementations of directional interfaces and an experiment to formally compare the performance of these interfaces.

Being able to locate survey points is an important step in the stakeout process. This chapter compares different ways that a WOAR interface can help the user to orient himself or herself in the direction of the next survey point. The chapter researches these interfaces in the context of navigational tasks which require the user to move from waypoint to waypoint on a predetermined route. Once a waypoint is reached, the user needs to turn towards the next waypoint. These waypoints are very easy to recognise once they are in the user's field of view, as they are highlighted by the AR system.

However, when they are outside the user's field of view, a cue may be beneficial to help the user face the correct direction. This is mainly due to the limited field of view that AR video see-through HMDs provide. For example, the camera of the video see-through HMD used in this experiment provided a horizontal field of view of only 50°, significantly smaller than the typical human field of view of 180°. This small virtual window into the real world can cause a loss of context and can make it difficult for the user to see the next virtual waypoint as they walk towards the current one.

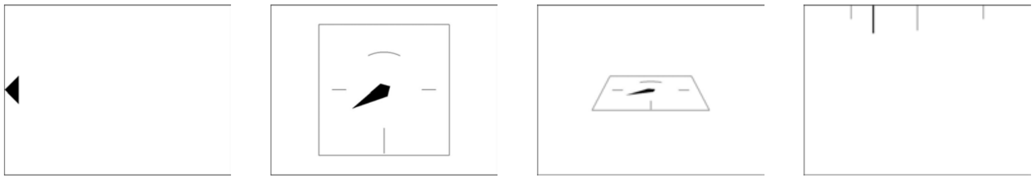
Orientation cues are relevant when the user wants to begin walking towards a target and needs to orient himself or herself towards the target first. These cues can also be relevant while navigating around an obstacle that the AR path finding system did not know about.

While navigation in AR itself has been the subject of previous research, interfaces for direction finding have mostly been mentioned in passing, and formal comparisons between them are the exception. For example, [Billinghurst et al. 1998] formally compared a limited selection of visual and audible directional cues.

Section 2.3 provided a survey of directional interfaces. This chapter describes implementations of a selection of these interfaces and compares them in a formal user study.

3.1 The Implemented Directional Interfaces

Six directional interfaces were implemented for the evaluation, based on the most promising cues described in the research literature. Four of them were visual interfaces based on a circular compass, a horizontal compass, and left/right arrows (see figure 3.1). The non-visual interfaces were a haptic belt (figure 3.2(c)) and a spatial audio beacon.

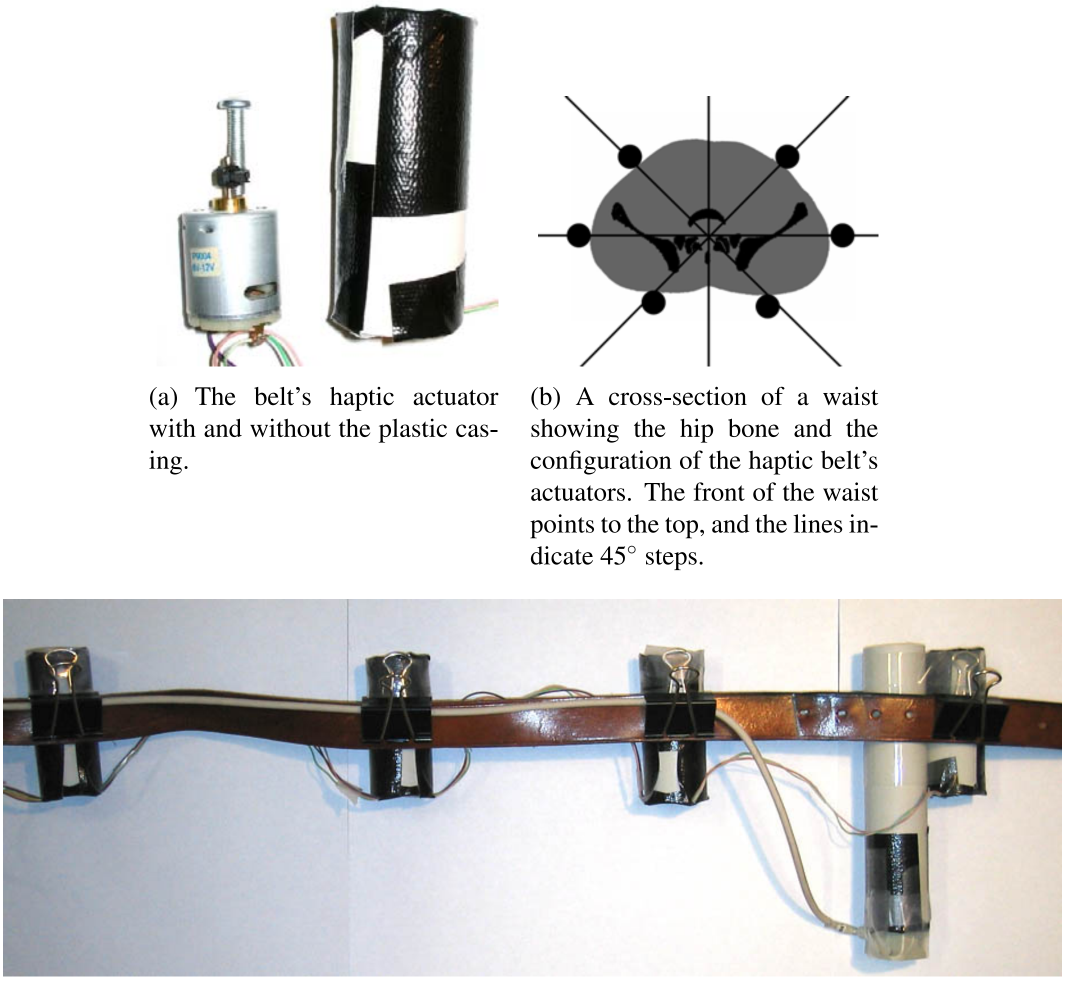


(a) Left/right arrows (b) The HUD compass (c) The perspective compass (d) The horizontal compass

Figure 3.1: The four implemented directional visualisations. All are indicating a target at $-120°$

The evaluation did not include a map interface, as maps are more concerned with providing contextual rather than directional information. As described in the introduction, the experiment looks at directional interfaces in the context of virtual paths and waypoints. These features provide navigational context during the stakeout process. The wealth of information presented by a map would not make it efficient as a directional interface. If the amount of information displayed on the map were reduced to a minimum – the user's current location and the location of the next target – the interface would be too similar to the circular compass to be included in the experiment.

The visual interfaces were implemented as head-stabilised. Before the evaluation, a small pilot study was carried out that addressed the concern whether head roll (leaning the head to the side) would make the head-stabilised visualisations



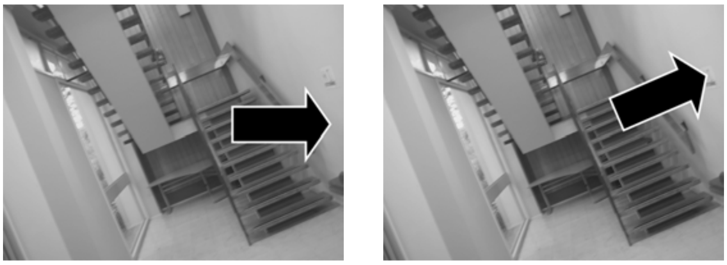
(a) The belt's haptic actuator with and without the plastic casing.

(b) A cross-section of a waist showing the hip bone and the configuration of the haptic belt's actuators. The front of the waist points to the top, and the lines indicate 45° steps.

(c) The haptic belt with four of its haptic actuators and the magnetic tracker.

Figure 3.2: The haptic belt.

confusing. As long as the user held their head level, the virtual imagery aligned well with the real world. However, if the head was tilted, an interface supposed to point to the side would then also point up or down at an angle (figure 3.3(a)). This was corrected for by making the visualisations world-stabilised in relation to camera roll but head-stabilised with respect to the other 5 degrees of freedom. This meant that the virtual imagery would always appear level with the real world (figure 3.3(b)). However, pilot testers found this to be more confusing than helpful. Directional interfaces with world-stabilised roll appeared to be tilted rather



(a) A head-stabilised arrow.

(b) An arrow with world-stabilised roll.

Figure 3.3: Directional arrows with head- and world-stabilised roll

than balanced horizontally. This became more apparent the more the head was tilted. Thus, the visualisations were fully head-stabilised as exemplified in figure 3.3(a).

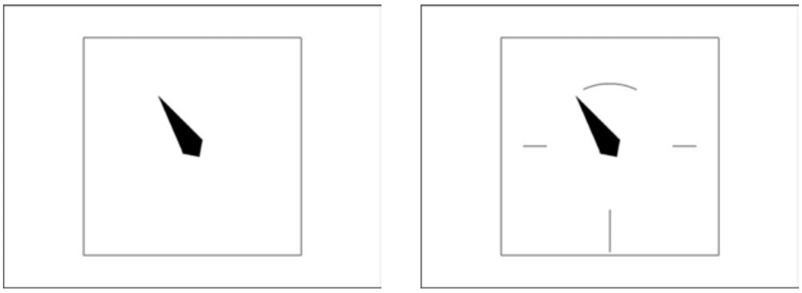
The following section describes the implementation of the interfaces evaluated in the experiment.

3.1.1 Left/Right Arrows

The experiment included a simple interface with left/right arrows as shown in figure 3.1(a). In order to avoid rapid flipping between opposing arrows for targets directly behind the user, the interface used a different threshold depending on which arrow was currently displayed. Instead of using a single 180° threshold to switch between left and right arrows, it used a 170° threshold to change to a left pointing arrow and a 190° threshold to change to a right pointing arrow. This meant that once the arrow had flipped, the user's head had to turn at least 20° in the opposite direction to flip the arrow back.

3.1.2 Circular Compasses

Two versions of the circular compass were evaluated: a HUD version (figure 3.1(b)) and a perspective head-stabilised version (figure 3.1(c)). Both were based on the same drawing algorithm. The square seen in figures 3.1(b) and 3.1(c) was intended to help the participants to easily distinguish between the two compass visualisations.



(a) A plain compass pointing at a target, leading the user to believe that the target should be visible to them.

(b) The modified compass pointing at the same target, showing that the target is outside of the camera's field of view.

Figure 3.4: A target in the phantom field of view

The circular compasses had to consider the difference between the human field of view of 180° and the field of view of the HMD of 50°, and the resulting potential directional confusion in the participant. When using the system, a target placed in this *phantom field of view* resulted in distorted perception of the target's location, especially when the target was close to the edge of the HMD's field of view. In early trials with the circular compass interface, users sometimes perceived the compass needle to point to an object "in front of them" while the target was not actually visible in the HMD (see figure 3.4). This confused participants, as they assumed that the camera image covered a larger part of their natural field of view, and that targets in front of them would be visible in the HMD. Some users were not able to get accustomed to the differences in field of view, and, even after repeated use of the system, the compass visualisation remained confusing to them. To address this problem, the visualisation explicitly incorporated the two different field of views in the compass visualisation (see figure 3.4(b)): the camera's field of view was highlighted as an arc in the compass, while the human field of view was highlighted as compass markers at 90° and $-90°$. In addition to this, a compass marker was placed at 180°. This visualisation eliminated the confusion between the field of view of the camera and the human field of view.

3.1.3 Horizontal Compass

The horizontal compass as shown in figure 3.1(d) was inspired by the context compass [Suomela & Lehikoinen 2000]. However, while the context compass visualised targets within the user's field of view, the horizontal compass implemented for this experiment visualised targets that the user could not see. It worked similarly to a rear-view mirror, with the needle moving left as the user turned left. This visualisation was included since most people are accustomed to rear-view mirrors. A horizontal visualisation might also be more easily integrated into a HMD interface than a circular compass, as it could be placed at the edge of the screen.

The horizontal compass covered all directions that were not within the user's field of view. It had the same $90°$, $-90°$ and $180°$ markings as the circular compass. For example, in figure 3.1(d), a target is indicated at $-120°$, requiring the user to turn $120°$ to the left. Since the compass needle is drawn right of the $-90°$ marker, the user can see that the target is behind them. As the user turned left, the needle moved to the left as well, just as in a mirror. As the needle disappeared left off the screen, the target appeared in the user's field of view, and moved from left to right.

3.1.4 Audio Beacon

A spatial audio beacon to indicate the position of a target was also evaluated. Descriptive audio cues such as the one described by [Ross & Blasch 2000] will probably be slow due to the duration of the spoken instructions, and thus were not used. The audio beacon was a continuous white noise loop similar to the one used by [Billinghurst et al. 1998]. It was presented to the user as a stationary spatial sound using the OpenAL library[1]. The beacon was head-stabilised except for yaw, which was world-stabilised. A generic sound card and headphones were used.

3.1.5 Haptic Belt

Based on the positive results reported in the literature as described in section 2.3, the evaluation also included a haptic belt (see figures 3.2(c) and 3.5). The belt

[1] http://www.openal.org/

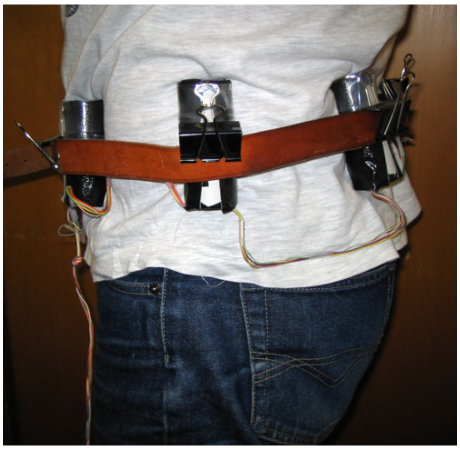


Figure 3.5: A user wearing the haptic belt

had six actuators so as to compromise between cost and efficiency. The actuators were distributed around the user's waist as shown in figure 3.2(b). The chosen configuration spread the actuators evenly around the user's waist with the exception of the front. Note that the irregular shape of the human body meant that an even distribution on the body's surface resulted in an uneven angular distribution of the actuators. In figure 3.2(b), the straight lines through the centre of the body indicate 45° angles, showing that there was a 90° distance between the two back actuators, while they were evenly spaced across the body's surface. For directions that fell between two actuators, the software interpolated and triggered both actuators.

For the actuators, the belt initially used small electronic buzzers of the same type as used for the fingARtips system [Buchmann et al. 2004]. The fingARtips system provided haptic feedback to the user's fingertips, allowing user's to "feel" virtual objects with their hands. These buzzers vibrated at around 400 Hz and were muted while preserving their vibrotactile properties. However, some users were not able to feel the belt's buzzers when they were placed on soft tissue areas, while the vibrations were felt strongly when the actuators were placed close to a bone. See figure 3.2(b) for an indication of the bone distribution in the waist area. As a result, the belt used a tapping signal rather than vibration. The tapping signal was generated by motors with eccentric weights (see figure 3.2(a)), as they are used for vibration actuators. The motors were only switched on for short periods of time, just long enough for one revolution, so that they produced successive tapping motions rather than vibrations. When triggered, the actuators were continuously activated for 100ms and then switched off for 67ms.

The orientation of the belt was tracked independently of the HMD with a dedicated Flock of Birds sensor, and the belt only signalled the target's direction when the target was not in the HMD's field of view.

3.2 Experiment

This section presents an experiment that compared the performance of the selected directional interfaces in an AR environment. In this experiment, participants had to orient themselves using the different directional interfaces. The dependent measures were task completion time and accuracy.

The experiment followed a within subjects design with the factor "directional interface" and seven conditions:

- no help
- left/right arrows
- HUD compass
- perspective compass
- horizontal compass
- haptic belt
- spatial sound

The dependent measures were task completion time and the number of times the user overshot the target direction. Participants also gave subjective feedback and answered selected questions from the NASA TLX questionnaire [Hart & Staveland 1988].

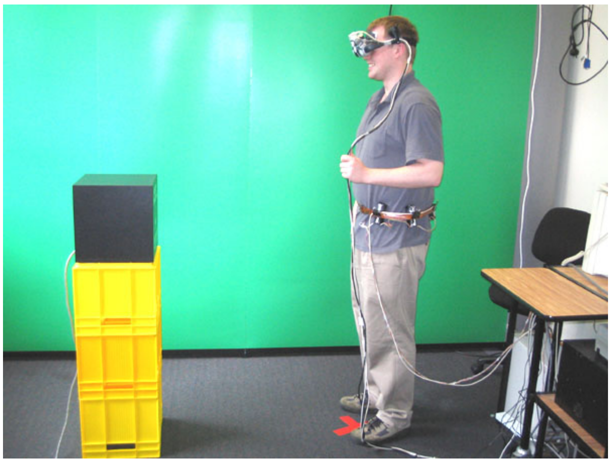


Figure 3.6: The experiment setup for the haptic belt condition. Participants wore a video see-through HMD and were tracked with a Flock of Birds magnetic tracker.

3.2.1 Procedure

Each participant completed seven direction finding tasks for each interface type. The target directions were symmetrical with respect to the user's left and right hand side to eliminate bias. Targets were placed at 50°, 90°, 130°, 180° and at $-50°$, $-90°$ and $-130°$ (figure 3.7). Considering that the camera's field of view was 50°, the participants only had to turn a minimum of 25°, 65°, 105° and 155°, respectively. The order of the target directions between the interfaces and the order of the interfaces between participants were each counter-balanced using a Latin square design. Experiment sessions took an average of 30 minutes including the completion of the questionnaires.

3.2.2 Apparatus

Participants wore an 800*600 pixel video see-through i-glasses HMD connected to a desktop PC. The HMD had a single USB 2.0 web camera with a resolution matching the HMD's resolution and a field of view of 50°. Target directions were displayed as vertical red lines. Participants were asked to stand in a predetermined location and roughly orient themselves towards the same start location before each task (see figure 3.6). The target locations were then placed relative to the participants' actual orientation to ensure comparable measurements. For each task, the

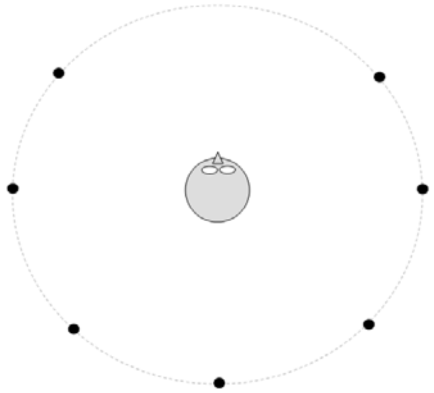


Figure 3.7: The virtual targets distributed around the participant

direction of the target was displayed using the current interface. The participants were then asked to orient themselves as quickly and accurately as possible towards the target while remaining in the same location. A task trial ended when the target was anywhere in the participant's field of view for one second. This second was then subtracted from the measured task completion time.

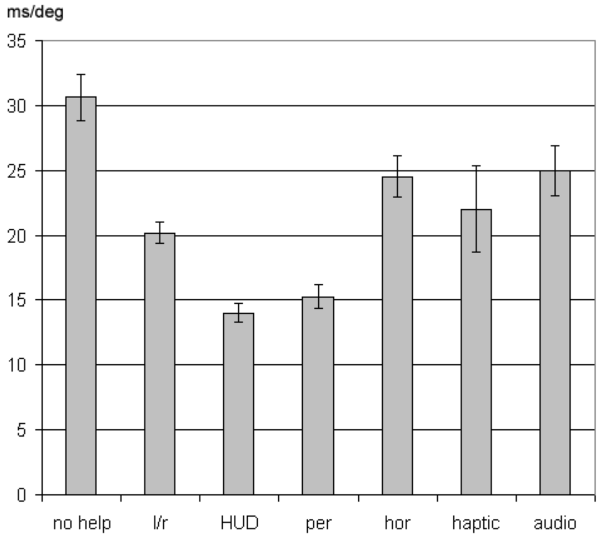
The users' head orientation and the haptic belt were tracked with a Flock of Birds magnetic tracker with a resolution of $0.1°$ and an accuracy of $0.5°$. Only the yaw information was used to overlay AR content onto the HMD.

3.2.3 Participants

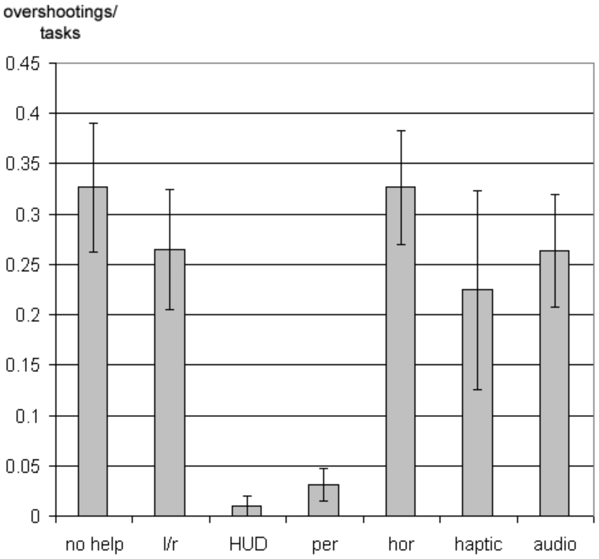
Fourteen participants ranging from 20 to 45 years old completed the experiment. All were students at the University of Canterbury. 11 were male, 3 were female. Three of them had used virtual reality systems before, but none of them had experienced an AR interface before.

3.3 Results

The two circular compass conditions performed the best of all the interfaces. They were faster, resulted in fewer overshootings, and were preferred by the participants.



(a) Normalized means for milliseconds per degree for each interface.



(b) Average number of overshootings per task for each interface.

Figure 3.8: Task completion time and accuracy of the directional cues.

Task Completion Time

Task completion times are normalised as ms/deg to allow the comparison between targets. Please note that this is not the speed at which the participants

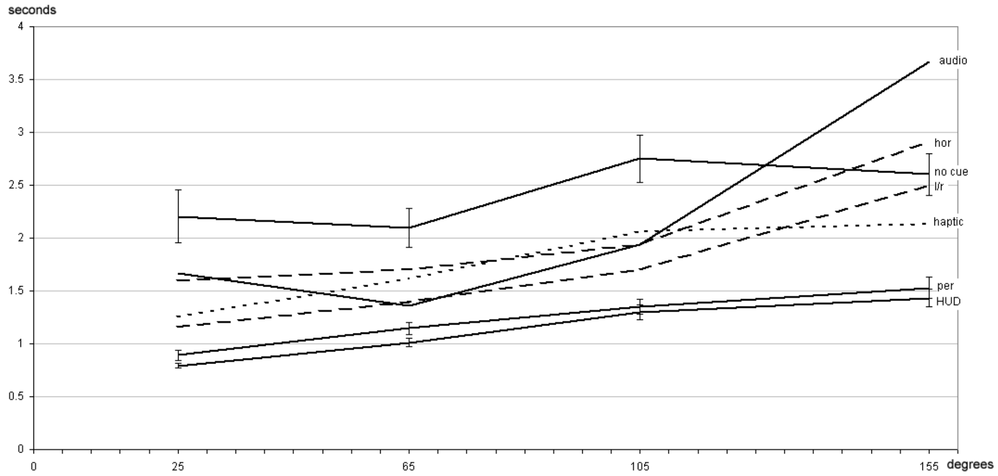


Figure 3.9: A time/degree diagram for the directional interfaces. Only some interfaces have error bars to avoid clutter.

Table 3.1: The mean values for task completion times normalised in milliseconds per degree and overshootings. Standard deviations are listed in parentheses.

Condition	Task Completion Time	Overshootings
no help	30.6 (6.7)	0.33 (0.24)
left/right arrows	20.2 (3.1)	0.27 (0.22)
HUD compass	14.0 (2.6)	0.01 (0.04)
perspective compass	15.2 (3.4)	0.03 (0.06)
horizontal compass	24.5 (6.0)	0.33 (0.21)
haptic belt	22.0 (12.5)	0.22 (0.37)
spatial sound	25.0 (6.9)	0.26 (0.20)

turned. The fastest condition was the HUD compass with 14ms/deg, closely followed by the perspective compass with 15.2ms/deg. The next best condition was left/right arrows at 20.2ms/deg, taking 33% longer than the perspective compass. As expected, the “no help” condition showed the slowest performance at 30.6ms/deg. The HUD compass performed twice as fast as the “no help” condition. An analysis of variance shows a significant difference between the interfaces ($F(6,90) = 10.44, p < 0.001$). Table 3.1 and figure 3.8(a) show the average turning speed for each interface in milliseconds per degree. A post-hoc analysis with

Table 3.2: Significant differences in task completion time at the 0.01 level according to a post-hoc test with Bonferroni adjustment.

	no help	left/right	HUD	perspective	horizontal	haptic	audio
no help		✔	✔	✔			
left/right	✔		✔	✔			
HUD compass	✔	✔			✔		✔
perspective	✔	✔			✔		✔
horizontal			✔	✔			
haptic belt							
audio			✔	✔			

Table 3.3: Significant differences in overshootings at the 0.05 level according to a post-hoc test with Bonferroni adjustment.

	no help	left/right	HUD	perspective	horizontal	haptic	audio
no help			✔	✔			
left/right			✔	✔			
HUD compass	✔	✔			✔		✔
perspective	✔	✔			✔		✔
horizontal			✔	✔			
haptic belt							
audio			✔	✔			

Bonferroni adjustment showed that there was a significant difference between the two circular compasses and the other interfaces – except for the haptic belt (see table 3.2). There was no significant difference between the two circular compasses.

Figure 3.9 plots the average task completion times split up by target distance and interface. The two compass interfaces both performed about a second faster for each target direction than the "no help" condition.

Overshooting

The results were even more dramatic for overshooting (figure 3.8(b) and table 3.1). Again, the two circular compasses performed best, and there was a significant difference between the interfaces ($F(6,90) = 5.11, p < 0.001$). These results mean that a user would overshoot once in 100 tasks when using the HUD compass compared to overshooting 27 times in 100 when using left/right arrows. Similar to the task completion time, a post-hoc analysis with Bonferroni adjustment showed that there was a significant difference between the two circular compasses and the other interfaces – except for the haptic belt (see table 3.3). There was no significant difference between the two circular compasses.

It is desirable to minimize unnecessary head movement in WOAR applications. To stay focused and avoid disorientation – especially with a significantly reduced field of view – the AR system needs to guide the user in such a way as to avoid unnecessary head movement. The two circular compasses achieve excellent performance for this. Unlike the two circular compasses, there was high variance in the performance of the other interfaces.

Post-hoc tests with a Bonferroni adjustment showed similar results for task completion time and overshooting: There was no significant difference between the performances of the two circular compasses, while their performance times were faster and significantly different from the performance of all other conditions except for the haptic belt. There was no significant difference between the performance of the belt condition and the performance of the other conditions due to the belt's high variance.

Subjective Measures

The subjective measures generally matched the objective results.

The participants ranked the interfaces from 1 ("liked best") to 7 ("liked least"). Figure 3.10 shows the average preference ranking for each of the conditions. A Friedman test showed that there was a significant difference between the interfaces ($\chi^2_r = 47.18, df = 6, N = 13, p < 0.001$), with the two circular compasses being preferred most.

The participants also answered four questions for each interface on a Likert scale from 1 ("disagree") to 7 ("agree"). Friedman tests showed that there was a significant difference between the interfaces for each of the questions, with the HUD compass always rated best and the perspective compass always rated second best. The questions were "I performed well" ($\chi^2_r = 39.24, df = 6, N = 13, p < 0.0001$), "I found the task easy to complete with this interface." ($\chi^2_r = 40.45, df =$

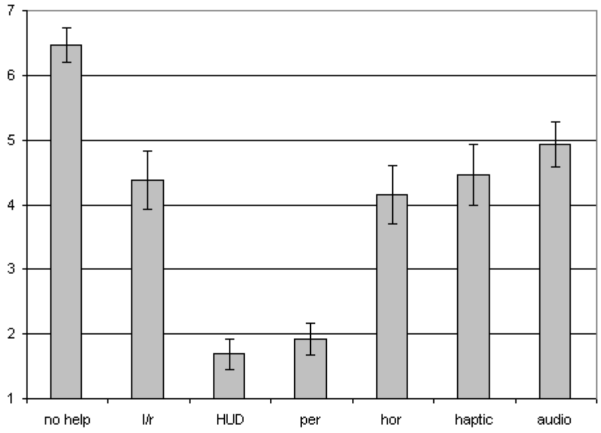


Figure 3.10: Ranking from 1 “liked best” to 7 “liked least”.

$6, N = 12, p < 0.0001$), “It was always easy to understand and follow the directions from the interface.” ($\chi_r^2 = 32.74, df = 6, N = 12, p < 0.0001$) and “I felt comfortable using the interface.” ($\chi_r^2 = 26.49, df = 6, N = 12, p < 0.0001$).

The participants’ comments agreed with the statistical data. Several users noted the sudden ease of direction finding when switching to one of the circular compasses for the first time, making comments such as “oh, this is much better”. The circular compasses received overwhelmingly positive comments such as “[it] tells you exactly where to turn before you turn”, “[it was] very easy to follow”, and “perfect.” The markings on the circular compasses were also well received with participants saying that “the field of view [marking] was very helpful and stopped me from overshooting” and “having the 90° and 180° markers was helpful”.

Some participants also highlighted issues with the perspective compass as compared to the HUD version. They said that “it was distorted a bit, so [they] found it harder to tell the exact point [they were] turning to” and that “[they] had to ‘read’ the perspective a little more”. One participant noted that the HUD compass took up a lot of screen real estate as compared to the other interfaces.

The other visual interfaces did not receive such positive comments. Several participants noted that, with the left/right arrows, they “did not know how far to turn”. The horizontal compass was called “confusing” and “troublesome” by participants, with some noting that they “might get better with practice,” or that it was “good practice for [their] brain”. One of the participants strongly preferred the horizontal compass and even ranked it the highest. Coincidentally, he was the second slowest participant with this interface.

The non-visual cues received mixed comments. Spatial audio cues were not perceived as an efficient interface, with several stating that “it was hard to tell where exactly the noise was coming from,” and others making comments such as “confusing when target is behind”. Several participants also noted that they did not like the audio sample or that this interface would interfere with conversations or listening to music. However, one of the participants noted that this was the least obtrusive interface.

Opinions on the haptic belt were more divided. While some found the interface “fun” and “easy to follow”, several participants said that they felt “uncomfortable”, or “slightly uncomfortable”. Others noted the problems that were already encountered while building the belt: “some parts of the body feel [the] motor [more strongly] than others. So [the] signal is stronger in different directions.” A few participants commented on the low resolution of the device, as “it didn’t tell [them] exactly where the target was, it only guided [them] in the general direction” and that there were “ambiguous vibrations when the target is close”.

3.4 Discussion of the Results

The two circular compass interfaces performed faster and were preferred to the other interfaces. The performance of the haptic belt, the spatial audio, and the horizontal compass was surprisingly low. They all conveyed more information than the left/right arrows, which performed faster. This section discusses possible reasons for this result.

3.4.1 Haptic Belt

It is notable that the haptic belt showed a relatively large variance in its performance. There are four possible reasons which can account for this. Firstly, the belt had by far the lowest resolution of the interfaces that provided angular information. Compared to the belt’s six actuators, the horizontal compass, had a 133 times better resolution. Considering its low resolution, the belt performed surprisingly well. Secondly, the haptic actuators had to be adjusted for each participant, so that they were positioned at the same angular direction. The largest participant needed a belt 20cm longer than the thinnest participant did. The actuators were positioned according to the diagram shown in figure 3.2(b), but an accurate calibration was not performed. This better reflects how the belt would be used in practice. In hindsight, accurate calibration would have been preferable. A third possible contributing factor was the relative inexperience of the participants with such a device. For example, the participants would not have had much practice in quickly orienting themselves towards the direction of a haptic signal. Thus, the perceived relation between their waist and the real world coordinates might have been distorted. This is a factor that could be reduced by training. The fourth factor that may account for the mutable efficiency of the belt is the resolution of haptic sensors in the waist area. It is possible that users could not physically differentiate between the signals sufficiently, making an accurate perception of the signal impossible. This is especially true of vibration, which can be felt over a larger area than other stimuli such as tapping. While the application tried to drive the actuators so that they tapped rather than vibrated, vibration could still be felt, which may have made perception of the stimuli less precise.

The performance of the haptic belt could be improved by making the modifications suggested above. However, to provide a practical solution, the device will need an accurate calibration procedure that is not prohibitive in everyday use. Based on the experiment results and the costs incurred by the additional tracking and calibration requirements, the belt is not a viable alternative if a system is able to provide visual directional cues.

3.4.2 Audio Beacon

As shown in figure 3.9, the audio beacon was most efficient for 90° and -90° turns, while it was very inefficient for a 180° turn. The audio beacon was even the least efficient condition for a 180° turn. This is consistent with our observations of participant behaviour during the experiment. Most of the participants initially did not turn when audio cues were provided. Instead, they concentrated on the sound and tried to find out which side the sound was coming from before they started to turn their head. This indicates that either spatial sound is not a very efficient cue, or that the audio signal provided to the participants was not good enough.

For targets at 90° and $-90°$, the ambiguity of the signals provided to the left and right ear was the lowest, resulting in comparative performance of the audio beacon. For audio beacons located at 180°, the signals presented to the left and right ear were initially identical, causing some participants to stall. In addition to the left/right ambiguity of this signal, participants could have also been affected by the so-called "front-back ambiguity" [Wightman & Kistler 1999], meaning that they could not hear the difference between a spatial audio beacon originating from directly in front of and directly behind them. The participants' behaviour is especially curious since the participants were explicitly told before the experiment that the target would be behind them if they could not perceive a difference in volume between their left and right ear.

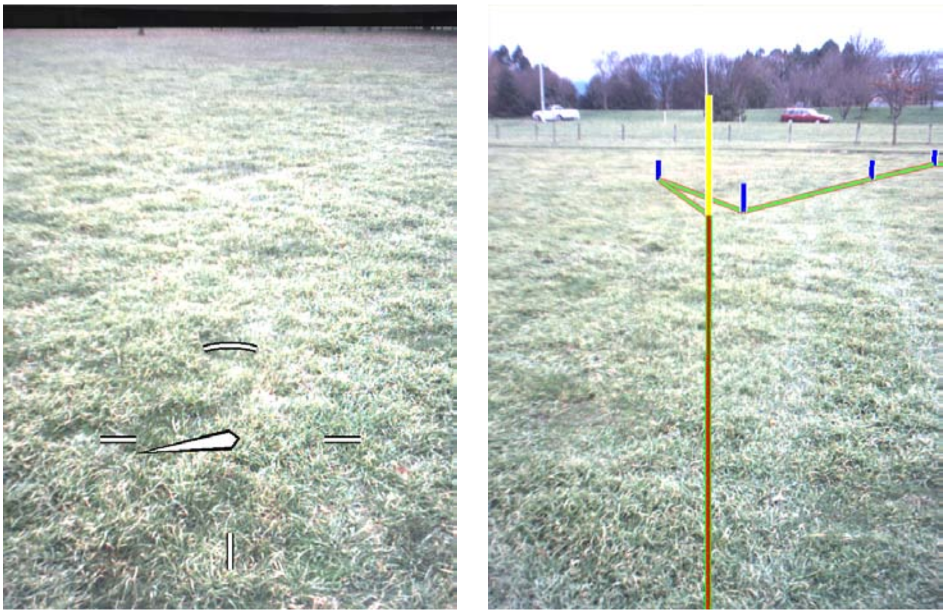
It should be noted that the data collected for the 180° direction may not be as reliable as for the other directions, as the sample size here was only half the size as for the other directions. This is because this direction was not mirrored as the other ones.

3.4.3 Horizontal Compass

While the horizontal compass performed slightly worse than most other interfaces, its performance could be greatly improved. The experiment focused on visualizing targets outside the user's field of view, and thus was only concerned with the portion of the horizontal compass that presented what was outside the user's field of view. While it was straightforward to explain the other interfaces to the participants, the horizontal compass was difficult to explain and hard to master. A more traditional horizontal compass such as the one originally described by [Suomela & Lehikoinen 2000] might be more efficient. The scale could also be extended to cover the entire 360° around the user. Further research will be necessary to compare the performance of such an interface to that of the circular compasses.

3.5 The Implemented WOAR Interface

The navigation interface described in this section was implemented based on the evaluation reported in this chapter. The interface supported navigational tasks which require the user to move from waypoint to waypoint on a predetermined route. Once the user reached a waypoint, the interface directed the user towards the next waypoint.



(a) A compass led the user in the direction of the target if it was not currently in their field of view

(b) A red path led the user to the location which was marked with a yellow pole.

Figure 3.11: The navigation interface of the WOAR system.

Figure 3.11(b) shows a sample route between stakeout locations. Stakeout locations were visualised as vertical waypoints. The points were linked by a path, with the path to the next stakeout location being highlighted.

A circular compass indicated target directions when the next stakeout point was currently not in view (figure 3.11(a)).

As seen in figure 3.11, the drawing algorithm for both the stakeout route and the compass drew a high contrast border around the objects. This was to ensure that they could be seen in both high and low lighting conditions that occur in outdoor Augmented Reality.

A version of this interface was used in the formal evaluation of the WOAR system, as reported in chapter 6.

3.6 Conclusion

This chapter presents a formal evaluation of selected directional interfaces for navigation in WOAR. There was a significant difference in performance, with a circular compass being the most efficient and the most preferred option.

Two versions of a circular compass were investigated, with the HUD compass performing marginally better and being preferred more than the perspective compass. The two circular compasses used a visualisation that eliminated the problem of the phantom field of view that is caused by the mismatch between the human field of view and the field of view of the video see-through display.

The evaluation was the first to formally compare the performance of a haptic belt. The performance of the belt was low as compared to the circular compass. However, the experiment showed that the participants were able to follow the directions of the belt. Systems for visually impaired users or systems that do not provide a head-mounted display may use a haptic belt as an effective tool for navigation.

At the end of the chapter, an implementation of the HUD circular compass for a WOAR navigation interface is demonstrated.

The next chapter compares depth cues far AR stakeout in a formal user study and presents a WOAR implementation for correct occlusion of a real stakeout pole.

Chapter IV

Depth Cues for AR Stakeout

This chapter describes implementations and a formal comparison of depth cue techniques for the WOAR stakeout application.

In the stakeout process, depth cues are relevant for two sub tasks: (1) walking towards a target and (2) placing a stake on that target. Different depth cues may be used for these two tasks, as movement and distance to the observed objects are relevant to the usefulness of depth cues. This chapter focuses on the task of placing a stake on a target while standing still.

Testing early prototypes of the WOAR stakeout application showed that placing a real pole on a virtual marker was not straightforward. With depth cues such as stereovision removed, users could not easily estimate the height of the pole over the ground. The pole usually touched the ground before or after the user anticipated it, making stakeout awkward. Users tended to first place the pole on the ground and then slowly move it closer to the target. This chapter describes an experiment that investigated the use of depth cues that aimed at helping users to place the stake directly on a virtual marker by providing depth cues.

With this task, interaction takes place on the ground at a distance of about two meters from the user's eyes, a distance that has not been well studied before by AR depth perception research. Most previous research has been conducted on near field depth perception, such as [Mason et al. 2001], and some in the far field, such as [Wither & Höllerer 2005]. As shown in figure 2.12, the quality of depth cues can vary greatly over distance.

This chapter describes an experiment that attempted to fill this gap by comparing the efficiency of selected depth cues for an AR stakeout application. First, section 4.1 describes depth cues for WAOR stakeout. Then, section 4.2 describes the implemented depth cues for the experiment which is described from section 4.3 on. Sections 4.7 and 4.8 report the results and discuss them. Finally, section

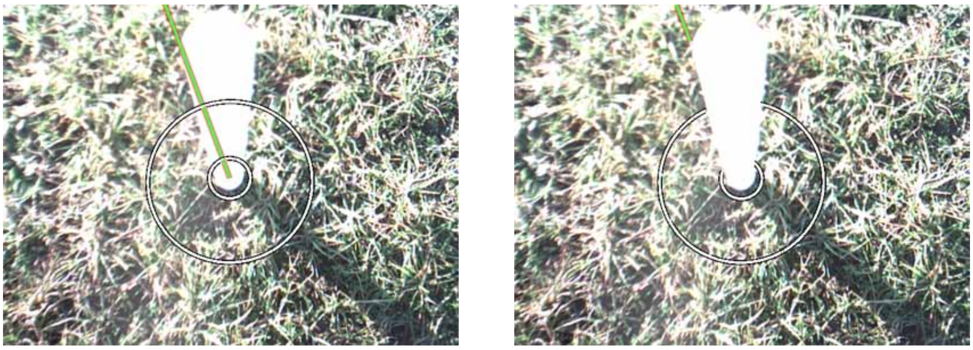
4.10 describes and discusses an implementation of correct occlusion for a real stakeout pole in WOAR.

4.1 Depth Cues for WOAR Stakeout

This section assesses the depth cues described in section 2.2 with respect to stakeout in augmented reality, and discusses which cues are relevant to be included in the experiment. The task that the experiment investigated was placing a stakeout pole on a stakeout marker. The marker was placed on the ground at a distance of about two meters from the user's eyes.

4.1.1 Pictorial Depth Cues

- *Occlusion* was described as the most important depth cue [Cutting 1997] and the most common source of depth cue conflict in AR [Drascic & Milgram 1996]. It is relevant for AR stakeout, as figure 4.1 illustrates: If the AR system does not provide correct occlusion, the virtual stakeout marker will simply be overlaid on top of the real stakeout pole, presenting conflicting depth cues to the user (figure 4.1(a)).



(a) The pole being wrongly occluded by virtual objects

(b) The real pole correctly occluding the virtual objects.

Figure 4.1: Occlusion of virtual objects drawn on the ground by a real stakeout pole.

In early trials of the experiment interface, pilot users found correct occlusion of the pole to be so important that it was always provided in the experiment. With incorrect occlusion, pilot testers would see a virtual target marker that appeared fixed to the ground but also floated on top of the real pole that was supposedly above the marker. With the imagery not matching the user's concept of what he or she should have been seeing, the user had to actively interpret the visualisation to match what was happening. Interaction became tedious as users now had to concentrate harder and work around the imagery presented to them. User feedback for incorrect occlusion was very negative. Based on these early trials, correct occlusion is fundamental for a good user experience for AR stakeout.

- *Height in the visual field* is not a relevant depth cue for stakeout, as it requires the observer to look ahead instead of down. Accordingly, it is only effective beyond the personal space (see figure 2.12).

- *Cast shadows* are relevant for stakeout, as they map the vertical height of the pole to the horizontal ground. While cast shadows do exist in real life, they are often not visible in a video see-through HMD due to the low contrast of the HMD. For example, cloud cover can result in diffuse lighting, making real shadows too faint to be seen in a video see-through HMD. Virtual cast shadows may be used to remedy this.

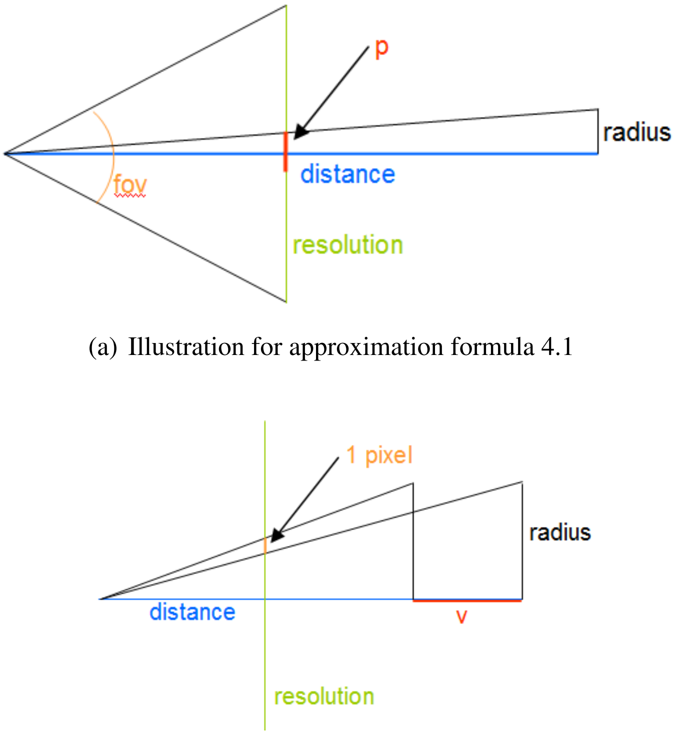
- *Relative size* could be relevant, as the radius of the pole and the target marker could be made to be of the same size. However, this cue is probably only ordinal [Cutting 1997]. In addition, the resolution of the video see-through display that was used in the experiment downgraded the resolution of this cue considerably.

  The approximation formula 4.1 shows that the pole, with a radius of 1.25cm, at a distance of 2m, with a horizontal field of view of 45°, and a horizontal resolution of 800 pixels, would have appeared about 12 pixels wide in the HMD. Figure 4.2(a) illustrates how the formula was derived: p are the pixels on the captured image that a pole with *radius* would occupy at a certain *distance*. The formula is based on the angular resolution of the camera measured in pixels per degree and calculated as $resolution/fov$, in our case

$800 pixels/45°$. Additionally, the approximation formula 4.2 shows that the pole would have to be brought to a distance of 1.85m from the eye to appear one pixel wider. A depth resolution of 15cm is not good enough for stakeout. This formula is illustrated in figure 4.2(b): again, based on the angular resolution of the camera, we calculate the distance v that an object needs to be moved closer to the camera to occupy one pixel more than before.

$$p = atan\left(\frac{radius}{distance}\right)\frac{resolution}{fov} \tag{4.1}$$

$$v = distance - \frac{radius}{tan\left(\frac{(p+0.5)fov}{resolution}\right)} \tag{4.2}$$



(a) Illustration for approximation formula 4.1

(b) Illustration of the approximation formula 4.2.

Figure 4.2: Illustration of the two approximation formulas.

- *Aerial perspective* is not relevant for stakeout, as it is only effective at a much further distance than two meters (see figure 2.12).

- *Depth of focus* could be simulated in AR. This cue did not appear as relevant because in the real world, small increments in focus blur indicate relatively large differences in distance. Additionally, deliberately degrading the visual representation of objects that the user interacts with might can an effect on performance.

4.1.2 Kinetic Depth Cues

Since kinetic depth cues arise from motion of the observer's head, they are probably not relevant for AR stakeout applications. A system that requires the user to sway their head during stakeout will not be successful. Kinetic depth cues were not included as a condition in the experiment.

4.1.3 Binocular Depth Cues

Binocular disparity is a strong depth cue at a distance of two meters or less, and is highly relevant for stakeout. It may, however, not be as effective in AR due to the reduced resolution of the video-see through display. It is also the most expensive cue of the experiment, and the most difficult to implement. This means that the performance of stereo vision has to be very good in order to justify its inclusion in a WOAR system.

4.1.4 Haptic Depth Cues

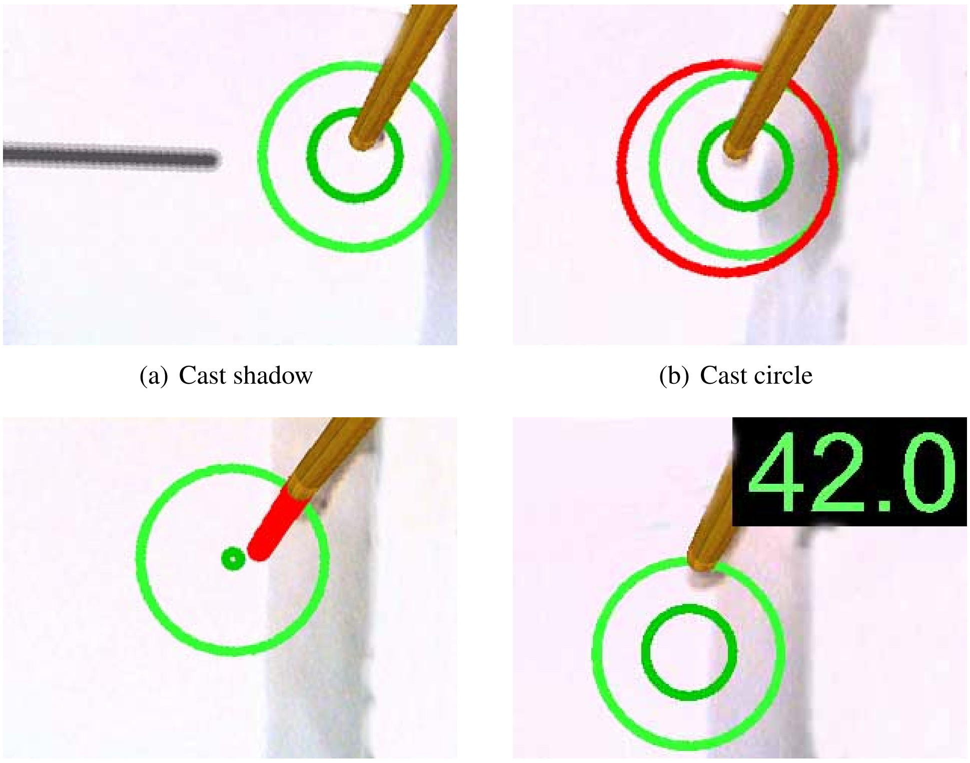
Haptic depth cues are relevant for stakeout. They are always present in both real and augmented reality stakeout, and do not need to be simulated. Haptic space is experienced due to kinesthetic feedback that indicates how joints are bent. For example, when we reach out to an object, we know in which direction and how far we have to bend the separate joints involved. In addition, there is indication that cutaneous feedback is also relevant for learning an object's location Dizio & Lackner [2002]. This feedback is provided when the pole hits the ground.

Since haptic depth cues are always present for both real and augmented reality stakeout, they were not controlled in the experiment.

4.1.5 Summary

In summary, the depth cues that are relevant for stakeout are occlusion, cast shadows, binocular disparity, and haptic depth cues. In the experiment, correct occlusion and haptic depth cues were always provided.

4.2 The Implemented Depth Cues



(a) Cast shadow

(b) Cast circle

(c) Dropline

(d) Number

Figure 4.3: The monocular AR depth cues. The wide grey line that runs vertical through these images is a shadow that was removed for the actual experiment sessions.

The experiment evaluated five different visual depth cues for AR stakeout. Two of these cues were based on the review of depth cues in the previous section: cast shadow and binocular disparity. The other three depth cues were artificial depth cues: cast circles, droplines, and a symbolic number cue. Correct occlusion was provided in each experiment condition. This was achieved by using either a real pole with a real marker, or a virtual pole with a virtual marker.

Cast Shadow

The experiment software provided virtual soft black shadows (see figure 4.3(a)) as recommended by Kersten et al. [1997]. The literature did not mention the influence of the lighting angle on the performance, but if the angle is too high or too low, the shadow will move too fast or too slow respectively. The implementation used a lighting angle of 45°. A disadvantage of cast shadows is that they may be occluded by the pole itself when the light source is behind the user. To amend this, the virtual light source could be made body-stabilised so that, for example, it always shines from the user's right. However, moving light sources have been shown to have a negative effect on depth perception [Kersten et al. 1997]. In the experiment, the user did not turn, and the artificial light always came from their right so that the shadow was unlikely to be occluded by the pole or the user's arm. If implemented in a WOAR system, the light source should match the sun to avoid confusion between the real and the virtual shadow that otherwise may move at different speeds.

Cast Circle

Cast circles may be used as an alternative depth cue in order to avoid the potential problems of cast shadows. This artificial depth cue is generated by drawing a virtual circle on the ground centred directly under the tip of the pole (see figure 4.3(b)). In the experiment, the radius of the circle was equal to the elevation of the tip of the pole, so that the radius of the circle decreased as the pole is lowered to the ground. This meant that when the pole was lowered, the cast circle moved as fast on the ground as a cast shadow with a lighting angle of 45°.

Dropline

A dropline is a vertical line drawn between the ground and an elevated object (figure 4.3(c)). It is an artificial cue that normally aids in the use of height in the visual field for floating objects [Ware 2000]. While height in the visual field may not be efficient for stakeout, the length of a dropline indicates the height of an object above ground. This means that droplines directly show the height of the pole above ground, and are therefore relevant for stakeout. Droplines have been shown to be very efficient depth cues, even more efficient than cast shadows [Hubona et al. 1999]. They have the advantage of projecting the position of the pole directly on the ground, and they also provide a direct cue for the elevation of the pole. A narrow virtual line is easily occluded by the real pole when the pole is close to the ground, so the virtual dropline cue was implemented as a cylinder with the same radius as the pole (see figure 4.3(c)).

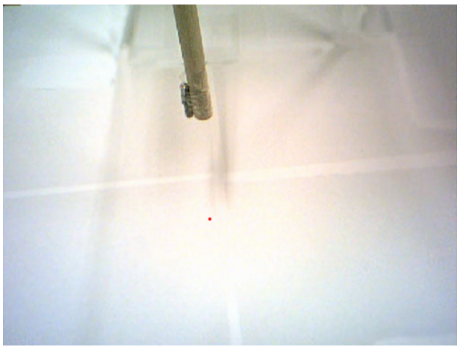


Figure 4.4: A pole with a laser mounted at the bottom. The laser dot projected on the ground has been enhanced in the photo to make it more visible.

A variation of droplines can be easily implemented without tracking the pole. A laser mounted to the tip of the pole projects a point on the ground and can give the observer similar information as the dropline cue (see figure 4.4). This cue was omitted from the experiment since the combination of cameras and HMD was not able to display the red dot of the laser to the observer in a satisfactory quality. In an outdoor situation, lighting is uncontrolled and may make the laser dot harder to see through the HMD. Some surfaces such as grass may also make the dot hard to see, as grass blades may occlude the dot.

Binocular Disparity

This dominant cue was included in the experiment. However, binocular disparity is an expensive cue with respect to hardware requirements as well as to possible sources of errors. For example, stereoscopic displays cause an accommodationvergence conflict by forcing the observer to focus independently from eye convergence. Another problem is the difficult calibration required for stereoscopic AR displays. For example, while the average interpupillary distance (IPD) is 6.3cm [Rosenberg 1993], even slight differences between the observer's IPD and the IPD simulated by the system will lead to dramatic depth perception errors [Drascic & Milgram 1996]. The implementation for the experiment circumvented the alignment problems of real and virtual objects as described in [Milgram & Drasic 1997]: in each condition, pole and marker were either both real or both virtual.

Number

In addition to these depth cues, the experiment included an artificial symbolic depth cue. The height of the pole above the ground was rendered as a number close to the tip, displaying the current height in centimetres and millimetres (see figure 4.3(d)). Unlike the other pictorial cues, this cue did not graphically connect the ground and the tip of the pole, making it potentially difficult for the observer to perceive the pole's location in the environment. However, this cue was the only one in the experiment that provided exact information about the distance of the pole to the ground. It could also be implemented without tracking the pole in 6 degrees of freedom.

In summary, the experiment compared four pictorial depth cues and binocular disparity. Of the pictorial depth cues, only the cast shadow modelled a real depth cue. Cast circle, dropline and number were artificial depth cues. Correct occlusion of the pole was always provided.

4.3 Experiment Design

The experiment compared the depth cues described in the previous section, with the addition of control conditions with and without the use of an HMD. A second factor modelled the varying accuracy requirements of the stakeout application, which can range from 0 to 10cm [Trimble 2005].

The experiment followed an 8×2 repeated measures factorial design. The dependent measures were placement error and task completion time. The two factors were

- **Depth Cue** with the levels
  - non-AR, both eyes
  - non-AR, one eye closed
  - AR mono, no depth cue support
  - AR stereo
  - AR virtual shadow
  - AR dropline
  - AR cast circle
  - AR number
- **Accuracy** with the levels
  - exact
  - within 10cm radius

Participants filled out a questionnaire with selected questions from the NASA TLX questionnaire [Hart & Staveland 1988] to assess the subjective workload for each condition. At the end of the experiment, participants were asked which depth cue they preferred and if they had any problems or suggestions for improvement.

4.4 Experiment Setup

Participants wore a video see-through i-glasses HMD for the AR conditions but not for the two non-Ar conditions. They held a pole in their right hand. Both the HMD and the pole were tracked using the Ascension Flock of Birds magnetic tracker. The Ascension tracker and the cameras were dynamically calibrated using an ARToolKit marker attached to the pole. This was to ensure good alignment of the real and virtual world. ARToolKit is an optical tracking library that recognises black and white markers in a video frame and calculates their position with respect to the camera [Kato & Billinghurst 1999]. The experiment software used the image plane position of the marker to ensure good alignment of real and virtual objects. A switch on the bottom tip of the pole connected to the computer was triggered when the pole tip touched the ground. The location and orientation of the pole were logged continuously during the experiment tasks. Only the FOB data was used for logging, and both the AR and the non-AR conditions were measured the same.

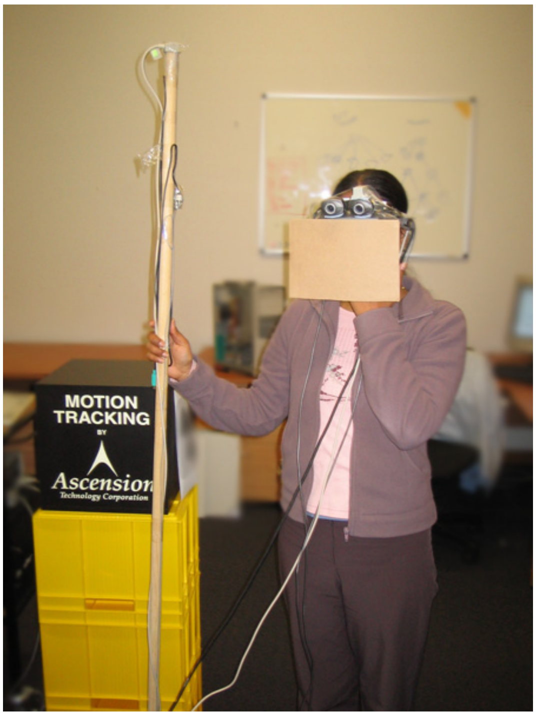
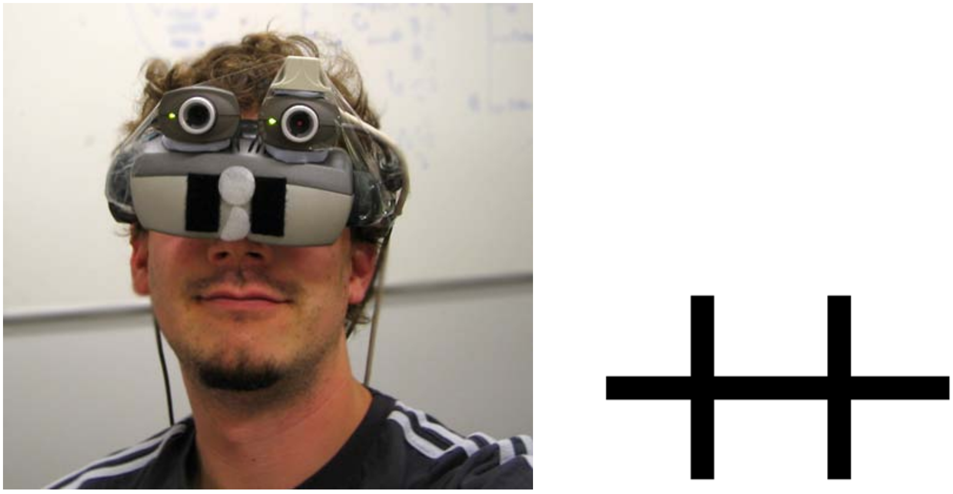


Figure 4.5: A participant holding the pole and wearing the stereo HMD

The i-glasses HMD was operated in stereo mode, with a resolution of 800 × 600 pixels at 100Hz. Each display was refreshed at 50Hz. Two Aplux USB 2.0 web cameras were mounted on the HMD with a resolution of 640 × 460 pixels at 30fps and a field of view of 45°. The system displayed frames to the user at 27fps in mono mode and at 15fps in stereo mode. A cardboard screen attached to the HMD blocked the participants peripheral vision and ensured that they could only see through the HMD in all conditions (see figure 4.5).

The stereo video see-through HMD used the stereo version of the i-glasses and two USB 2.0 web cameras (figure 4.6(a)). The cameras were attached to the top of the i-glasses with adhesive putty and tape. This allowed for easy adjustment of camera alignment and was stable for the duration of each experiment session.



(a) The stereo video see-through HMD (b) The stereo calibration image

Figure 4.6: The stereo video see-through HMD

Stereo vision was achieved by using OpenGL's LEFT and RIGHT buffers in conjunction with the NVIDIA stereo drivers[1].

Before each experiment session, the stereo HMD had to be calibrated. The HMD was fixed to the table at a distance of 60cm from the calibration pattern seen in figure 4.6(b). The cameras were adjusted so that each pointed directly at the center of one of the crosses with the viewing axes perpendicular to the pattern. For this, the two camera images were overlaid on top of each other on a computer screen to show the alignment of the two crosses. The operator then adjusted the cameras so that the left and right cross would overlay as perfectly as possible. Usually, the alignment would be several pixels out. The operator then rotated and translated the two images against each other in the software to reach pixel perfect alignment. The distance between the two camera lenses would be measured and used as the offset between the two virtual cameras in the OpenGL scene.

To prevent a calibration mismatch between real and virtual stereopsis, inter-camera distance and camera orientation were constantly corrected for by tracking ARToolKit markers on the pole and ground to ensure that virtual objects were registered correctly for both eyes. The ARToolKit was used only to track mark-

[1] `http://www.nvidia.com`

ers in the two-dimensional image plane of the camera. No 3D information was gathered through optical tracking. Information from the ARToolKit was used to translate and rotate the generated virtual images so that their location on the image presented to the user matched the location of the real objects. The physical calibration process between the cameras, the ground, the pole, and the FOB tracker was not accurate enough for a perfect match. A virtual pole was overlaid on top of the real pole to avoid a mismatch in stereo calibration between the real and the virtual world.

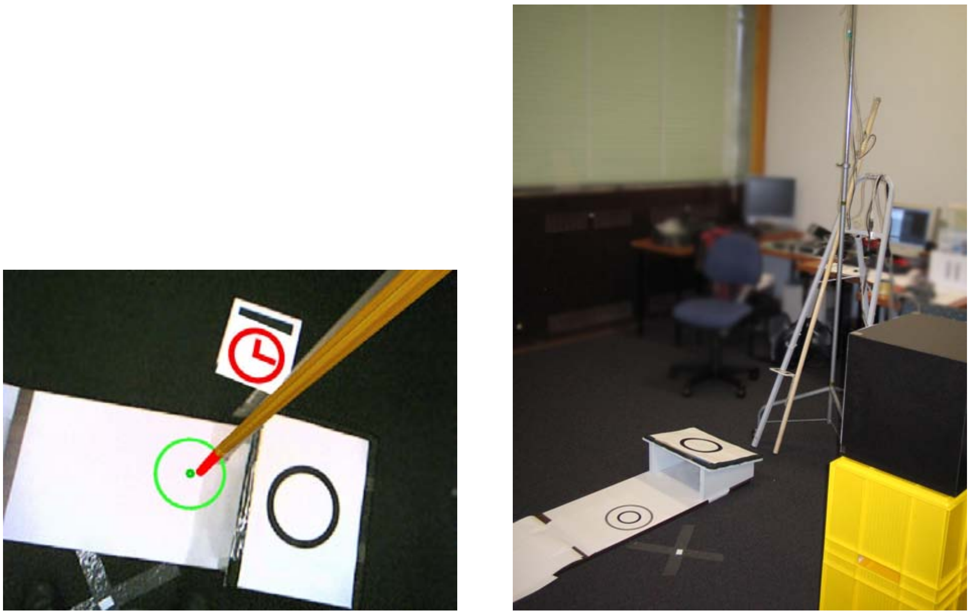
4.5 Procedure

At the beginning of each experiment session, a random dot test (similar to the RANDOT stereo test by Stereo Optical Co. Inc., Illinois) established whether the participants had stereo vision. In addition, the dominant eye was identified for the real world monocular condition.

A random dot test presents images of seemingly random dots to each of the user's eye. The images for the left and right eyes are mainly identical except for a region that has been translated, resulting in a 3D effect when both images are presented to the user's eyes in a stereoscopic viewer. The difference between the left and the right image can only be discovered if the user possesses stereo vision.

Each participant completed 4 stakeout tasks each in 16 different conditions, undertaken in randomized order, since strong learning effects were expected. For each condition, the participants finished a set of training tasks. Participants were asked to sit down and relax between tasks if they felt fatigued.

During the experiment, the participants stood upright and held a pole in their right hand. A task consisted of placing the pole in a resting area on the right hand side of the participant. Participants would not require the help of depth cues if the ground was completely level, as this would allow participants to lift the pole minimally and drag it across to the target. To prevent this, the resting area was elevated 12cm in order to force the participants to use the depth cues for the pointing tasks (see figure 4.7(a)). A virtual marker was then displayed in a random location within a small predefined area in front of the participant. In the case of those conditions that did not entail the use of an HMD, a printed marker was placed before them at a previously surveyed location (figure 4.7(b)).



(a) The experiment area with the virtual pole and the dropline cue (in red) as seen through the HMD: the participant stood on the cross, the elevated pole resting area was in the black circle on the right.

(b) The experiment area with the Flock of Birds tracker in the foreground. The metal easel and stand on the right was used to hold the cables leading to the pole. A real marker is shown on the ground.

Figure 4.7: The experiment setup

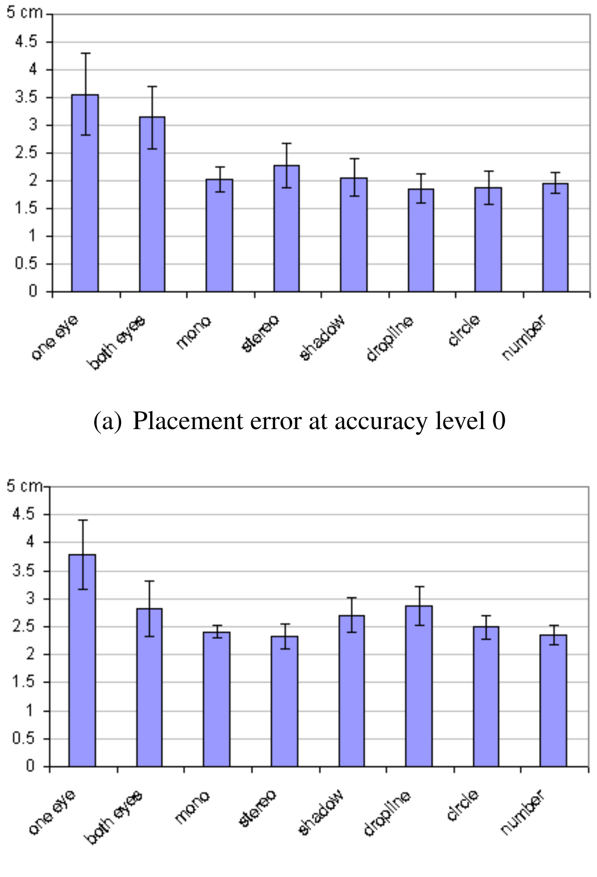
Markers were displayed as two concentric circles on the ground, with the inner circle visualising the task's accuracy level. For the "exact" accuracy level, the inner circle was as wide as the pole (see for example figure 4.3(c)), and for the "10cm" accuracy level, the inner circle was 10 centimetres in diameter (see for example figure 4.3(a)). The participants could study the marker's position before each task and then placed the pole as quickly and as accurately as possible on the virtual marker. Completion time was measured from lifting the pole from the resting area to placing the pole on the virtual marker. The virtual red clock attached to the pole as seen in figure 4.7(a) indicated to the participants that the stopwatch was running.

Experiment sessions lasted on average 36 minutes including the training phase and filling out the questionnaires.

4.6 *Participants*

Ten male students participated in the experiment. All were right handed, had normal or corrected to normal vision and stereo vision, and had previous experience with AR and HMDs. Two had brief experience with stereo AR, and none had previous experience with any activity similar to the stakeout task.

4.7 *Results*

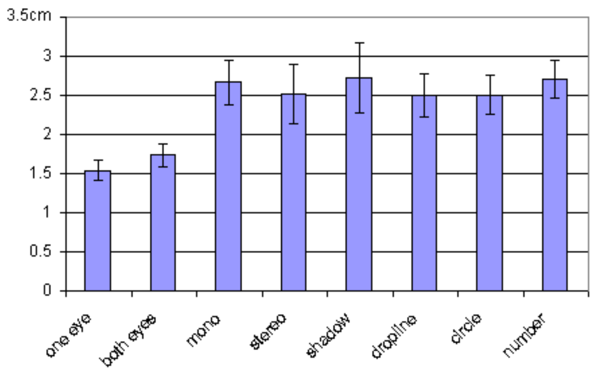


(a) Placement error at accuracy level 0

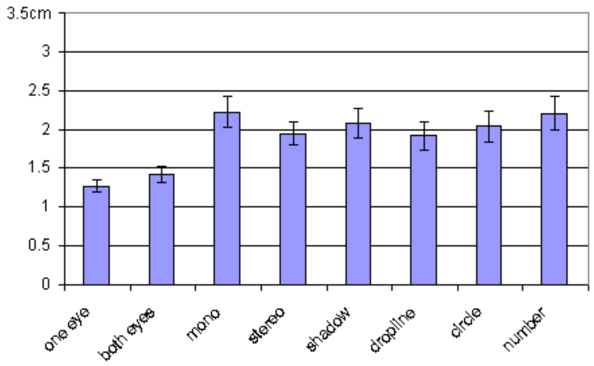
(b) Placement error at accuracy level 10

Figure 4.8: The means for the dependent measure "placement error".

The performance of the AR depth cues was remarkably similar across all the conditions with respect to both task completion time (figure 4.9 and table 4.2) and



(a) Task completion time at accuracy level 0



(b) Task completion time at accuracy level 10

Figure 4.9: The means for the dependent measure "task completion time" in seconds.

placement error (figure 4.8 and table 4.1), with a maximum difference of 5mm and 0.3s between condition means, respectively. In contrast, the non-AR conditions were faster but less accurate.

There was a significant difference between the accuracy levels and between the depth cue conditions with respect to both task completion time and placement error. See table 4.3 for the ANOVA results. There was no interaction between the factors for task completion time or placement error. Post-hoc comparisons with a Tukey test with an HSD of 2.37 seconds for task completion time and an HSD of 3.1 centimetres for placement error could not show a significant difference between the depth cue conditions.

Table 4.1: Mean and standard deviation results for displacement error in centimetres

Interface Condition	Accuracy 0	St. Dev.	Accuracy 10	St. Dev.
one eye	3.56	2.35	3.79	1.95
both eyes	3.14	1.78	2.83	1.56
mono	2.03	0.70	2.41	0.35
stereo	2.27	1.26	2.32	0.73
shadow	2.06	1.06	2.70	0.99
dropline	1.86	0.83	2.87	1.09
circle	1.87	0.98	2.49	0.64
number	1.95	0.58	2.35	0.54

Table 4.2: Mean and standard deviation results for task completion time in seconds

Interface Condition	Accuracy 0	St. Dev.	Accuracy 10	St. Dev.
one eye	1.54	0.42	1.27	0.24
both eyes	1.73	0.48	1.42	0.35
mono	2.67	0.90	2.23	0.64
stereo	2.52	1.21	1.95	0.48
shadow	2.73	1.42	2.08	0.60
dropline	2.51	0.88	1.92	0.59
circle	2.51	0.80	2.04	0.63
number	2.71	0.78	2.21	0.67

As expected, there was a significant difference between the two accuracy levels. When participants were given the more accurate targeting requirements, they took longer to complete the task and were more accurate. This was due to the tradeoff between time and accuracy for targeting tasks.

There was no interaction between the depth cue conditions and the accuracy levels for both task completion time and displacement error, meaning that the depth cues did not perform differently based on the two accuracy levels.

The data suggested different behaviours for the AR and non-AR conditions. The non-AR conditions were faster and less accurate. The accuracy of real stereo even dropped by 3mm when the participants staked out a smaller target, while the accuracy for the AR conditions increased for smaller targets. The accuracy

Table 4.3: ANOVA results for the two dependent measures "task completion time" and "displacement error".

dependent measure	**significant difference between accuracy levels**
task completion time	yes, $(F(1,7) = 15.89, p = 0.001)$
displacement error	yes, $(F(1,7) = 3.88, p = 0.05)$
	significant difference between conditions
task completion time	yes, $(F(1,7) = 5.8, p < 0.001)$
displacement error	yes, $(F(1,7) = 3.81, p < 0.001)$
	interaction
task completion time	no, $(F(1,7) = 0.15, p = 0.99)$
displacement error	no, $(F(1,7) = 0.54, p = 0.80)$

variance for the non-AR conditions was higher than that of the others. The "both eyes" condition was more accurate than the "one eye" condition. However, it was also slower.

As already mentioned, the AR conditions performed nearly equally well. Averaged over all samples for both accuracy levels, there was a maximum difference of 2.3mm between the means of the AR conditions. It may appear that some AR depth cues were more appropriate for staking out small targets. For example, AR stereo performed best and droplines performed worst for low accuracy requirements, while the reverse was the case for high accuracy requirements. However, the differences in placement error were small, and an analysis of variance showed that there was no interaction between the conditions.

It is especially notable that the plain AR condition performed as well as any of the other AR conditions. This comes as a surprise not only because the absence of depth cues did not deteriorate performance, but also because the subjective feedback, as described later in this section, suggested that the participants thought that not all depth cue conditions performed equally well.

4.7.1 Pole Movement

The position of the pole was logged throughout the experiment whenever a new frame was displayed to the participant. Plotting the pole movement during stakeout tasks provided surprising insights into the participants' targeting strategy.

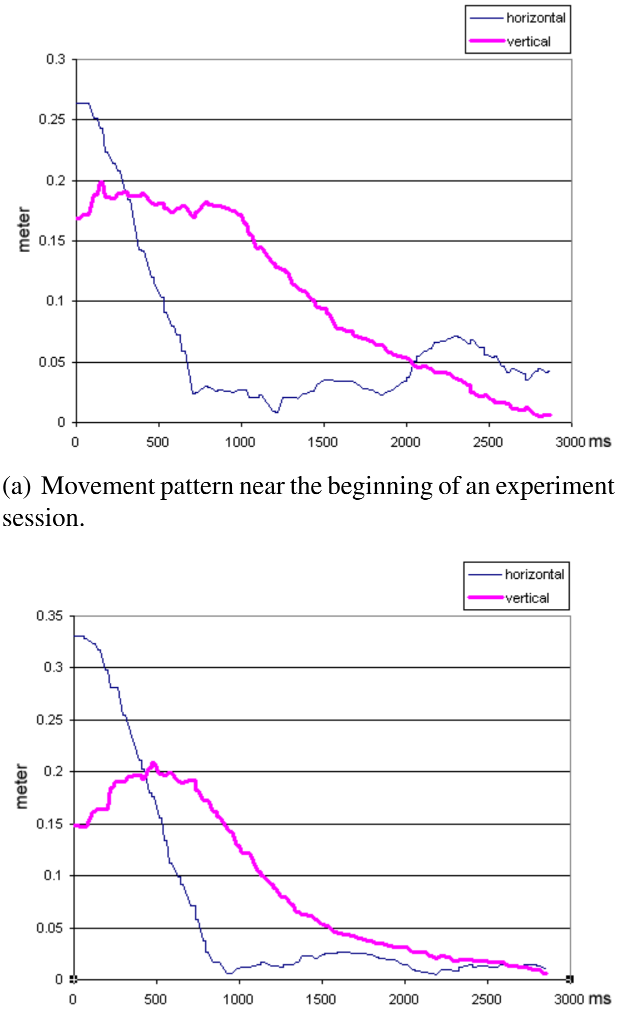
Depth cues would have been most relevant for vertical movement in the experiment, and analysis of the movement patterns of the pole showed that the participants handled vertical movement differently from horizontal movement. This is illustrated by graphs that plot the horizontal distance and the vertical distance of the bottom tip of the pole from the target over time (figure 4.10). The horizontal distance was measured on the horizontal plane, and the vertical distance was measured on the vertical axis.

The movement patterns were the same for all participants and all AR conditions. The pole elevation first increased due to an obstacle between the resting area and the targeting area. It then smoothly decreased without corrective movements. The horizontal distance rapidly decreased and then oscillated about the target. The graphs show an obvious learning effect, with figure 4.10(b) – a representative trial by a participant towards the end of an experiment session – showing a much more controlled, smooth targeting approach compared to figure 4.10(a), which represents a trial towards the beginning of an experiment session. The graphs for all AR conditions including plain AR and the number cue showed the same behaviour.

The movement patterns for the non-AR conditions differed from this in that the descent of the pole did not slow down as the pole approached the ground. Figure 4.11(a) shows a representative pattern for trials towards the beginning of an experiment session and figure 4.11(b) shows a representative pattern for trials towards the end of a session.

4.7.2 Subjective Results

Although the AR depth cues performed similarly, the participants had strong, consistent opinions on which cue was best. The participants were asked to rate their subjective experience of depth cue levels on a Likert scale from 1 to 5, using selected questions from the NASA TLX questionnaire [Hart & Staveland 1988]. The data was analysed with a Friedman test. Table 4.4 shows list of the measures and their analysis. Performance was rated from 1="poor" to 5="good", and frustration, mental demand, physical demand, and overall effort were rated from 1="low" to 5="high". Questions for these measures were of the format "Physical demand: How much mental and perceptual activity was required?"

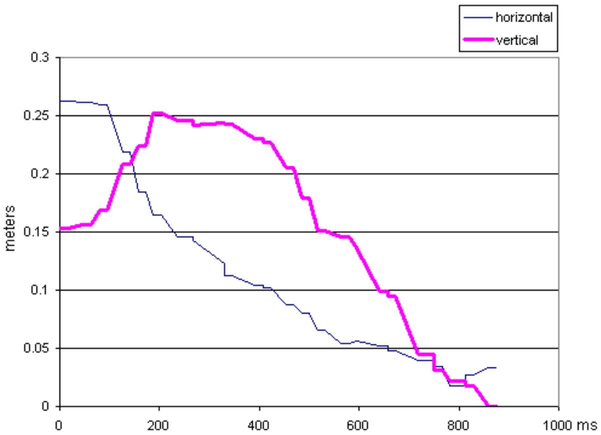


(a) Movement pattern near the beginning of an experiment session.

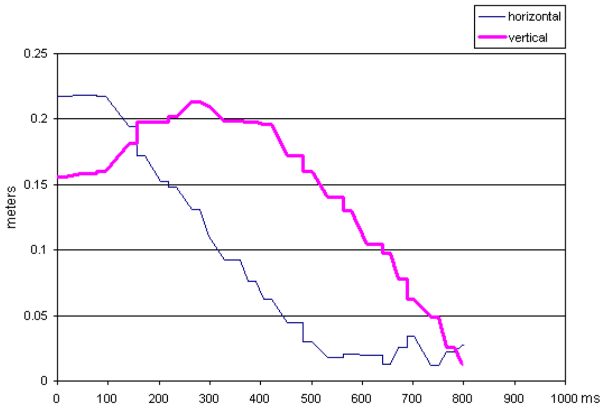
(b) Movement pattern towards the end of an experiment session.

Figure 4.10: Typical pole movements for a targeting task with HMD. The graphs show the distance of the pole to the target from lifting the pole off the start location to placing the pole on the target. Horizontal movement and the height of the pole are plotted separately to show the different movement patterns.

There was a significant difference between the depth cue conditions for each question, with non-AR stereo always rated best. Of the AR conditions, the circle cue was always rated best. It tied with AR stereo in mental demand and with the shadow in physical effort. Conditions such as plain AR or the number cue were consistently rated badly. For example, with the number cue receiving an average



(a) Movement pattern near the beginning of an experiment session.



(b) Movement pattern towards the end of an experiment session.

Figure 4.11: Typical pole movements for a targeting task without HMD. The graphs plot the elevation of the pole and the horizontal distance of the pole from the target in meters over time in milliseconds.

of 1.9 in performance as compared to an average of 4.4 for the circle (table 4.4 and figure 4.12(a)), or participants giving plain AR and the number cue a 3.1 average score for frustration, while the circle only rated 1.9 on average. The results showed that the participants found some depth cues easier to use than others.

The participants were also asked to rank the AR depth cues from ”liked best” to ”liked least” (see figure 4.12(b)). Again, there was a significant difference in how the depth cues were ranked, with the circle ranked best (Friedmann, $\chi_r^2 =$

Table 4.4: Results of the Friedman Test analysis of the subjective measures. The first row for each measure shows the means, and the second row shows the standard deviation.

Measure	**one eye**	**both eyes**	**mono**	**stereo**	**shadow**	**dropline**	**circle**	**number**
Performance	3.4	4.6	2.8	3.3	4.2	3.3	4.4	1.9
	0.7	0.7	0.9	0.7	0.4	1.2	1.0	1.0
Friedman	$\chi_r^2 = 41.59, df = 7, N = 10, p < 0.0001$							
Frustration	2.5	1.5	3.1	2.6	2	2.9	1.9	3.1
	1.0	0.5	1.0	0.8	0.8	1.0	0.9	1.2
Friedman	$\chi_r^2 = 30.197, df = 7, N = 10, p < 0.0001$							
Mental	2.3	1.4	3.4	2.5	2.7	3.3	2.5	3.7
	0.9	0.5	1.0	0.7	0.8	1.1	0.8	0.8
Friedman	$\chi_r^2 = 35.637, df = 7, N = 10, p < 0.0001$							
Physical	2.8	2.3	3.0	3.2	2.9	3.2	2.9	3.3
	1.0	1.3	0.9	0.6	0.7	0.6	0.7	0.7
Friedman	$\chi_r^2 = 19.639, df = 7, N = 10, p < 0.01$							
Effort	2.6	1.7	3.2	2.9	2.6	3.1	2.5	3.5
	1.0	0.7	1.1	0.9	0.8	1.0	1.0	1.0
Friedman	$\chi_r^2 = 29.066, df = 7, N = 10, p < 0.0001$							

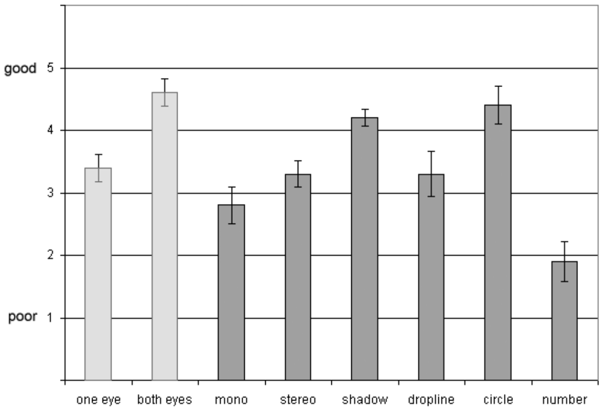
$19.31, df = 5, N = 10, p < 0.01$).

4.7.3 Comments By the Participants

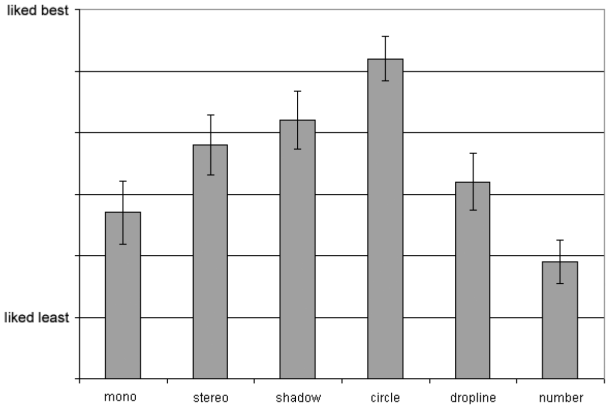
The *plain AR* condition was described as “not fun”. The HMD itself was disliked by the participants, with one complaining about a stiff neck, and another saying that the real monoscopic condition was “great” because it had “no HMD”.

Several participants were enthusiastic about the *shadow* when they first used this condition, especially when it followed one of the least preferred conditions. They said it was “much better”, that one “cannot imagine how much easier [the shadow] is”, and that it was “more natural” than the conditions they had previously completed. Two participants, however, did not perceive the shadow as a shadow but rather as a “moving line” or a “separate object”.

The comments about the *cast circle* conditions were even more enthusiastic. Participants said that the circle was best because it provided “big and obvious”



(a) Subjective performance rated on a Likert scale.



(b) Ranking of AR conditions

Figure 4.12: Subjective performance and ranking of the AR conditions.

feedback, that it was “heaps easier than the shadow” or that they liked it as much as the shadow, that it “helped greatly in speed and accuracy”, that it was “easy to figure out” and that it was easier to infer the location on the ground directly under the pole than it was with the shadow. However, one participant said that it “sometimes was a bit tricky to establish [the] centre of the circle” and suggested a combination with the dropline.

While identifying the position directly under the pole was a distinctive quality of the *dropline* cue, participants commented that this cue was “a bit annoying”, that it was irritating, and that a dropline at this steep angle “did not help so much”.

One participant said that “joining the ground and the pole did more harm than good for depth perception” while another said he was “not sure how exactly it worked”. One participant suggested that “possibly casting the line in the same direction as the pole might be easier.” This would be equivalent in functionality to the laser seen in figure 4.4.

The *number* cue was described as “fairly useless” and as “the most unnatural” cue which required focus switching and concentration for “interpreting the numbers”. The biggest problem with the number cue was that the digits often changed too rapidly to be read. Two participants stated in their questionnaire that they did not use the numbers as a depth cue because of this. In hindsight, problems with this cue should have been identified before the experiment, and it should not have been included.

One participant said that he did not know why, “but [AR *stereo*] seemed to make targeting easier without thinking about it” while one participant said it “felt a bit strange” and was “not quite natural enough to feel real...” Another participant said it had a “slight advantage over [AR] mono” while yet another suggested it should be combined with the shadow cue.

One participant suggested that “transparent cues would be better” so as not to occlude the marker.

In conclusion, the main results of the experiment were:

- There was no statistically significant difference between the efficiency of the depth cues, with the AR depth cues all performing similar to each other.

- The non-AR conditions were faster but less accurate than the AR conditions.

- Participants strongly preferred some depth cues over others, with the cast circle rated best.

4.8 Discussion of the Results

There were two surprising findings: there was no notable difference in performance results between the AR conditions, and the non-AR conditions performed worse than the AR conditions with respect to placement error.

First, section 4.8.1 shows that participants must have used a depth cue in the AR conditions, even when no visual depth cue was presented to them. This depth cue was most likely kinesthetic. Then, section 4.8.2 discusses why the presented visual depth cues were outperformed by a kinesthetic depth cue. Lastly, section 4.8.3 discusses the performance of the non-AR conditions.

4.8.1 Reliance on Kinesthetic Depth Cues

The similar performance of the AR depth cue conditions was surprising, as the participants' comments indicated that the depth cues were not equal, with some of the cues being perceived as helpful while others were perceived as being a distraction. Yet participants performed quickly and accurately, indicating that they did rely on some sort of depth cues.

Analysis of the data suggested that the participants may have relied on kinesthetic knowledge rather than visual feedback, as there was an obvious learning effect. Figure 4.10(a) shows a representative movement pattern during one of the first tasks of a participant. Elevation decreased nearly linearly. Figure 4.10(b) shows a representative movement pattern during one of the last tasks of a participant. The movement was much smoother, as expected. Interestingly, the pole elevation no longer decreased linearly. Instead, the downward velocity of the pole decreased, and the pole hovered closely to the ground while the horizontal position was adjusted. A non-linear downwards movement like this shows that participants must have been aware of the pole's elevation during the stakeout process.

Real pictorial depth cues that participants may have picked up through the HMD, such as real cast shadows or relative size, can be ruled out. Real cast shadows were eliminated by using diffuse lighting and adjusting the camera contrast accordingly. They were too faint to see in the video see-through HMD. The theoretical depth resolution of relative size was 15cm as shown in section 4.1. This was clearly not good enough to account for the pole hovering less than 5cm above the ground as seen in figure 4.10(b).

Having ruled out all plausible visual depth cues, the vertical movement of the pole is an open loop task that is aided by a kinesthetic learning effect, while the horizontal movement appears to be more of a closed loop approach with obvious corrective movements. This means that the provided visual depth cue information was less attractive to the participants than their kinesthetic sense.

4.8.2 *Ineffectiveness of the Visual Depth Cues*

Visual depth cues may not have had an effect on performance because of the following reasons:

(1) The depth resolution of the evaluated depth cues was not good enough at a distance of two meters to outperform kinesthetic knowledge.

(2) The lag or the low frame rate of the system made the depth cues inefficient.

Reason (1) can be ruled out for some cues, as the cast shadow or cast circle would theoretically have a very good depth resolution at a distance of 2 meters. Using the approximation formula 4.3, at a distance of two meters, a pixel represented about 1.9mm horizontal distance on the ground. This was a sufficient resolution for the pictorial depth cues such as the shadow or the circle to be efficient, as users were able to see changes in the pole's elevation at a resolution of 1.9mm.

$$d = distance\, tan(\frac{fov}{resolution}) \tag{4.3}$$

Reason (2) cannot be ruled out, as the system had significant lag and a relatively low frame rate. Several participants suggested that the low frame rate and the system lag might have had an adverse effect on their performance. The system ran at frame rates of 27Hz for the monocular conditions and at 15Hz for the stereo condition. An update of visual information of the moving limb at 10Hz has been found to be a minimum requirement for an accuracy gain [Heuer 2003]. This reflects human processing delays as well as a delay in muscle reaction. Ware & Balakrishnan [1994] believe that a frame rate higher than 10 Hz is required for accurate limb control. The system's lowest frame rate of 15Hz was well above 10Hz.

The lag of the system was not measured, but there was a significant, easily noticeable lag. There have been several studies on the relation of frame rate, lag and pointing performance. For example, Ware & Balakrishnan [1994] have shown that introducing system lag is a major factor in reducing the speed of 3D target selection, while McCandless et al. [2000] found that introduced time delay in a depth cue judgment task was linearly related to judgment error.

In conclusion, the low frame rate and the lag of the system will have had an adverse effect on the participants' performance. While all other HMD conditions would have been affected by the same low frame rate, it is not known if the lower frame rate of the AR stereo condition had a significant effect on the performance. It is possible that the lag and the frame rate made the visual depth cues less efficient than kinesthetic depth cues. An experiment using faster and more modern equipment is necessary to confirm this.

4.8.3 The Performance of the Non-AR Conditions

As can be seen in figures 4.8 and 4.9, users performed faster and with lower accuracy in the conditions without an HMD. The relatively poor accuracy performance of the non-AR conditions as shown in figure 4.8 and 4.9 may be due to the participants' over-confidence with these conditions. The participants did not take as much care as in the AR conditions and thus performed faster and less accurately.

The targeting strategy for the non-AR conditions was similar to that used in the HMD condition. However, the decrease of the pole's elevation appears to be much more linear and steeper here, regardless of stereo or monocular vision. Figure 4.11 shows two representative graphs of the pole movement for conditions without an HMD early during an experiment session (figure 4.11(a)) and towards the end of a session (figure 4.11(b)).

There are two possible explanations for the difference between the movement of the pole during conditions with an HMD and conditions without an HMD: (a) the participants knew exactly where the pole and the ground were at any time and were able to place the pole on the target without the need for long correctional movements, and (b) the participants were overly confident with these conditions and did not take enough care. The relatively inaccurate performance suggests that (a) is not the correct answer, while it supports (b). As a result, it is hard to compare the performance of these two control conditions to that of the others.

The over-confidence of the participants may have been avoided by giving immediate feedback on their performance during the experiment. Participants would have realised that their accuracy with the non-AR conditions was worse than with the AR conditions, and they might have adjusted their accuracy.

4.9 Experiment Conclusion

In conclusion, it appears that visual depth cues other than occlusion are of relatively low importance for efficiency in this AR stakeout task. Still, the participants clearly preferred some depth cues to others. The movement patterns found in this study give valuable insight into the participants' targeting strategy.

There were three main findings:

(1) Correct occlusion and depth ordering is essential for a stakeout task that requires the user to align real and virtual objects, and it should always be provided.

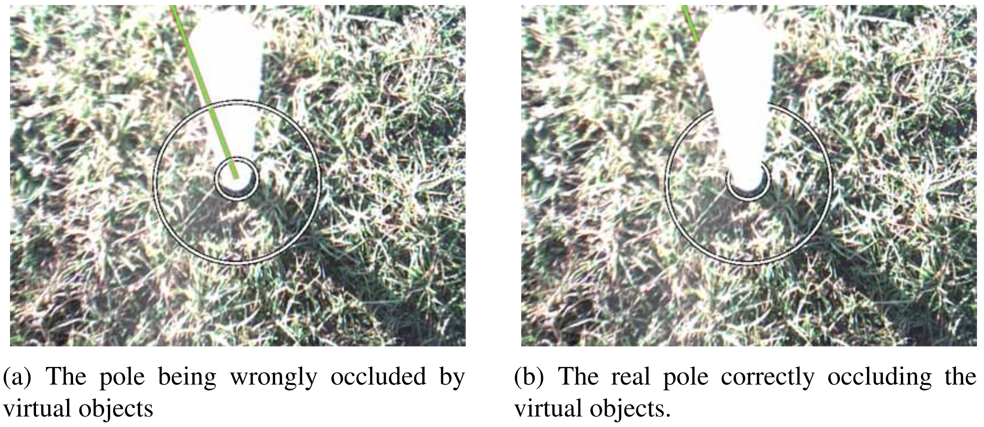
(2) At a distance of 2m, the evaluated visual depth cues were not effective – at least with the setup used in the experiment – and participants relied on kinesthetic knowledge instead.

(3) Users still felt that some of the evaluated depth cues aided them greatly, while other depth cues were perceived as detrimental. This means that providing appropriate depth cues can make users feel more in control and increase user satisfaction. The participants consistently preferred the cast circle as proposed in this chapter.

4.10 The Implemented WOAR Interface

Based on the results of the evaluation, correct occlusion will make stakeout with a WOAR system easier and more efficient. While the experiment showed that adding visual depth cues such as the circle increased user satisfaction, the implementation of such a cue would be expensive in WOAR, as the pole or stake would have to be tracked in 6 degrees of freedom. As a result, the WOAR system only provided correct occlusion of the real stakeout pole.

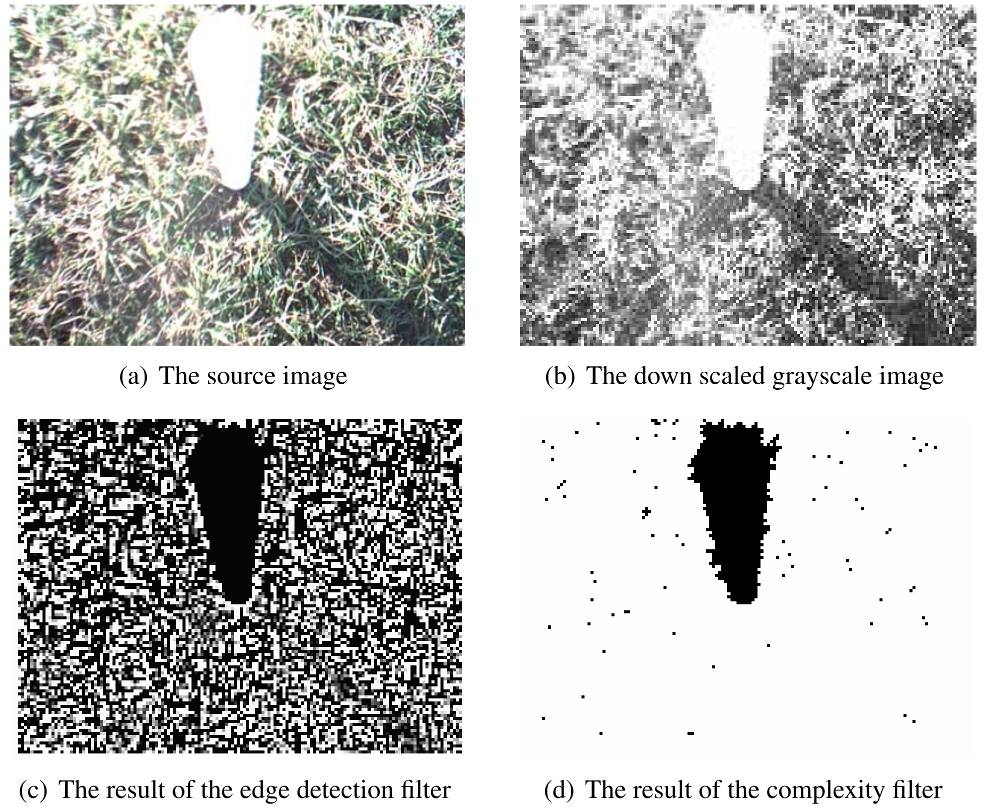
A vision based tracking algorithm tracked the stakeout pole to prevent the virtual markers from occluding the pole. Figure 4.13 shows the pole with correct and incorrect occlusion. The tracking algorithm was based on the assumption that the ground's visual complexity was greater than the stake's visual complexity. Figure 4.14 illustrates the tracking algorithm:



(a) The pole being wrongly occluded by virtual objects

(b) The real pole correctly occluding the virtual objects.

Figure 4.13: Occlusion of virtual objects drawn on the ground by a real stakeout pole.



(a) The source image

(b) The down scaled grayscale image

(c) The result of the edge detection filter

(d) The result of the complexity filter

Figure 4.14: The steps of the pole tracking algorithm

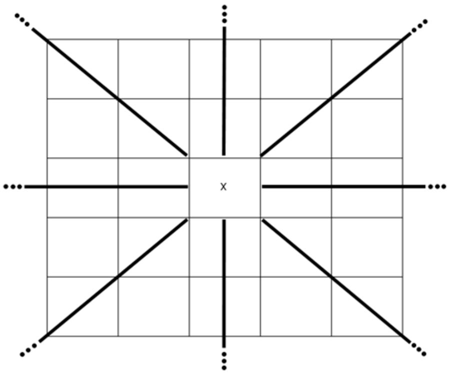


Figure 4.15: The complexity filter searched along the black lines for edges.

(a) The source image was captured through the system’s camera.

(b) The image was converted to grayscale and reduced to a fourth of its original resolution.

(c) An edge detection algorithm was applied.

(d) The complexity filter marked parts of the image with low visual complexity as black, and parts of the image with high visual complexity as white.

The implementation used the following convolution matrix as a single-pass edge detection filter:

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & -8 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

The complexity filter examined the neighbourhood of each pixel in the image. If there was an edge detected in the central point, then the point was marked as having high complexity. Otherwise, the algorithm checked for each branch of the star structure shown in figure 4.15 to see if it contained an edge or a pixel that was already marked as high complexity. If this was false for at least two neighbouring branches, then the central point was marked as low complexity. Otherwise, it was marked as having high complexity. The small areas in the grass marked as having low complexity in figure 4.14(d) could have been eliminated by applying a second pass of the filter. However, a single pass was good enough for the system, as the small occlusion errors were hardly noticeable to the user.

The implemented algorithm took on average 31ms to process a camera frame on the WOAR system with a 1.5 GHz Intel Pentium CPU and 1GB RAM.

The resulting mask was then used to avoid rendering virtual content over the stake as seen in figure 4.13(b). This simple approach worked for the stakeout application, as all virtual objects were placed directly on the ground, meaning that they should not occlude the pole.

This algorithm does not work in every setting, as the implementation was fine-tuned to the grass in the park that was used for evaluating the system. However, it provided a robust solution in this case, and the algorithm performed reliably during the formal evaluation of the WOAR stakeout application as described in chapter 6. It is desirable to create a general algorithm to track a specially prepared stakeout pole against any typical background encountered during stakeout (similar to the work presented by [Smith et al. 2005]). This would mean that every WOAR stakeout system with a video see-through display can and should provide correct occlusion during the process of placing the real stake on a virtual marker. However, this research is beyond the scope of this thesis.

4.11 Conclusion

This chapter described implementations and an evaluation of depth cue techniques for WOAR stakeout.

The main findings of this chapter are:

- Correct occlusion was important for stakeout in AR.

- Adding other visual depth cues did not have an effect on performance of AR stakeout.

- Participants most likely relied on kinesthetic depth cues for efficient stakeout instead.

- Adding some visual depth cues such as the cast circle increased participant satisfaction.

- The chapter presented an optical tracking algorithm that provided correct occlusion of a real pole in a WOAR stakeout application.

The next chapter explores obscured information visualisation techniques for the WOAR stakeout application and introduces interaction with artificially transparent stakeout poles and hands.

Chapter V

Obscured Information Visualisation for WOAR Road Stakeout

This chapter describes explorative implementations of obscured information visualisation (OIV) that may be used to make WOAR road stakeout more efficient. These visualisations include an overview of the road and an artificially transparent stakeout pole.

Obscured information visualisation enables AR users to see obscured or hidden objects in their environment. In a WOAR application, the system has to present the user with an optimal view of the information that surrounds them, free of clutter or information overload. To make it explicit, "virtual and real objects alike are considered part of the interface" [Höllerer 2004, page 3]. The user's real environment is not controlled by the application, and so, WOAR systems cannot manage the real environment by physically altering it. However, the system can control how the environment is displayed to the user, and OIV is one approach that may be used for this.

This chapter describes explorative implementations for each of the four types of OIV as described in section 2.4:

- *Virtual objects occluding other virtual objects*: section 5.1 presents a WOAR implementation that visualised a virtual road in the real world. The visualisation used OIV techniques to present road design details while preserving context.

- *Virtual objects occluding real objects*: section 5.2 presents an explorative implementation that aimed at recognising potentially hazardous real objects and ensured that they were not occluded by virtual objects.

- *Real objects occluding other real objects*: section 5.3 introduces the idea of interaction with artificially transparent objects and presents the results of an explorative implementation of a system that let users interact with artificially transparent stakeout poles in the lab.

- *Real objects occluding virtual objects*: section 5.4 presents an explorative implementation that investigated the possibility of artificially transparent stakeout poles in WOAR.

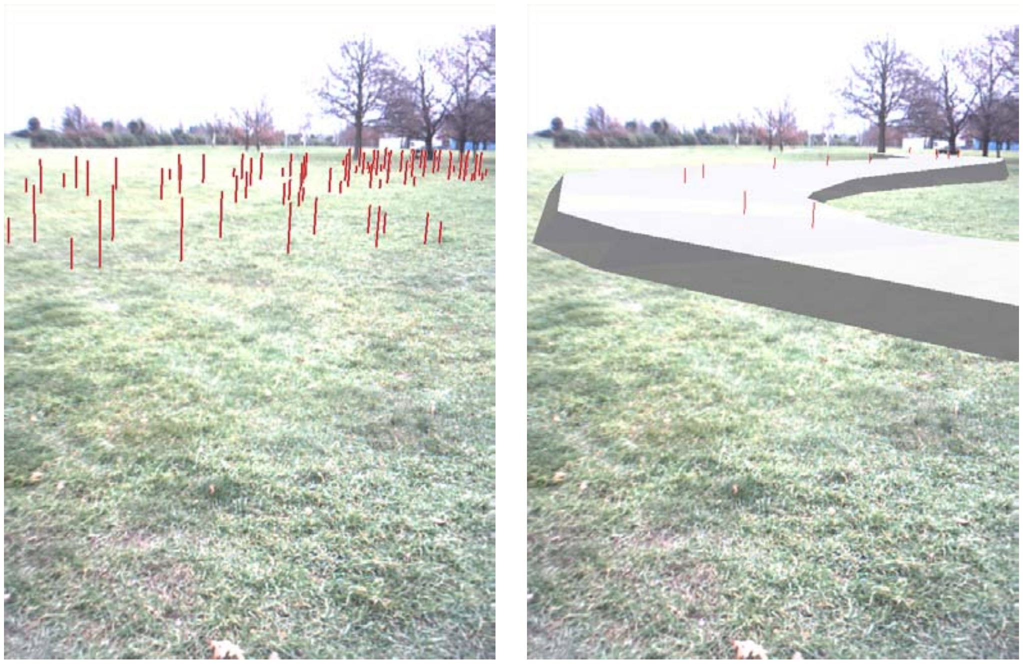
Unlike the work presented in the other chapters, this chapter describes explorative implementations rather than rigid user studies. The implementations and mock-ups described in this chapter have been developed with user feedback, but no rigid evaluation has been performed. While the previous chapters deal with depth cues and navigation, two essential issues of WOAR stakeout, this chapter puts WOAR stakeout in a wider context, for example road visualisation and safety.

5.1 Virtual Objects Occluding Other Virtual Objects

Road stakeout, the sample application for the WOAR system described in this thesis, does not only require users to locate and stake out road design points as investigated in chapters 3 and 4. It also needs to provide an overview of the road structure in the real world. This section describes how the WOAR application displays a road model while automatically giving the user detailed information about single stakeout locations.

Figure 5.1 illustrates the problem that this implementation solved. Each road design consists of a large number of stakeout points that the surveyor needs to stake out (figure 5.1(a)). This means that each point needs to be visible to the surveyor. A road visualisation such as the one shown in figure 5.1(a) becomes cluttered quickly and does not provide a good overview of the shape and position of the road in the environment. In contrast, a road visualisation like that shown in figure 5.1(b) provides a good overview of the road while suppressing the stakeout points. This section describes an implementation that used OIV techniques in order to combine these two extremes.

Informal interviews with stakeout experts, who used an earlier version of the WOAR application, established that a road visualisation needs to



(a) The stakeout points of a road design do not convey the road design well.

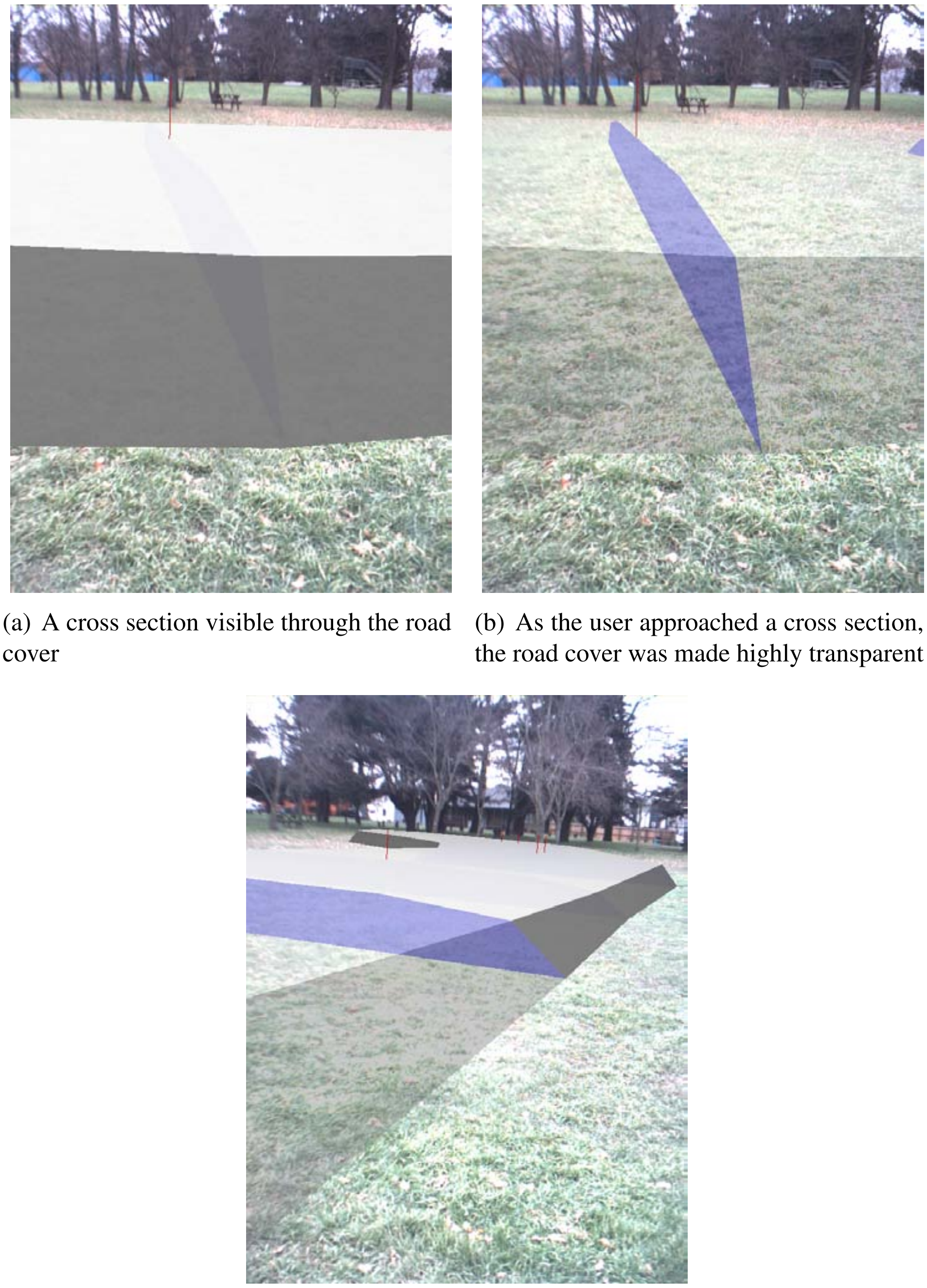
(b) The road model hides the stakeout points.

Figure 5.1: Stakeout points versus road model. These video see-through HMD screen captures show the same road design seen from the same location.

- provide an overview of the road to see how it fits into the real environment
- visualise the cross sections of the road
- highlight the centre line points of the road

The visualisation presented in this section fulfilled these objectives and let surveyors inspect all the major features of a road design in the real world comfortably without the need for input devices.

Figure 5.1(b) shows an overview of the road as provided by the implementation. While the shape and location of the road were clearly visible, the centre line points of each cross section were marked by short red lines, and the cross sections were visible through the slightly transparent road cover. This allowed surveyors to see both the shape of the road as well as the inner structure at the same time. Cross sections need to be visible at all times, as they are most relevant for a surveyor.

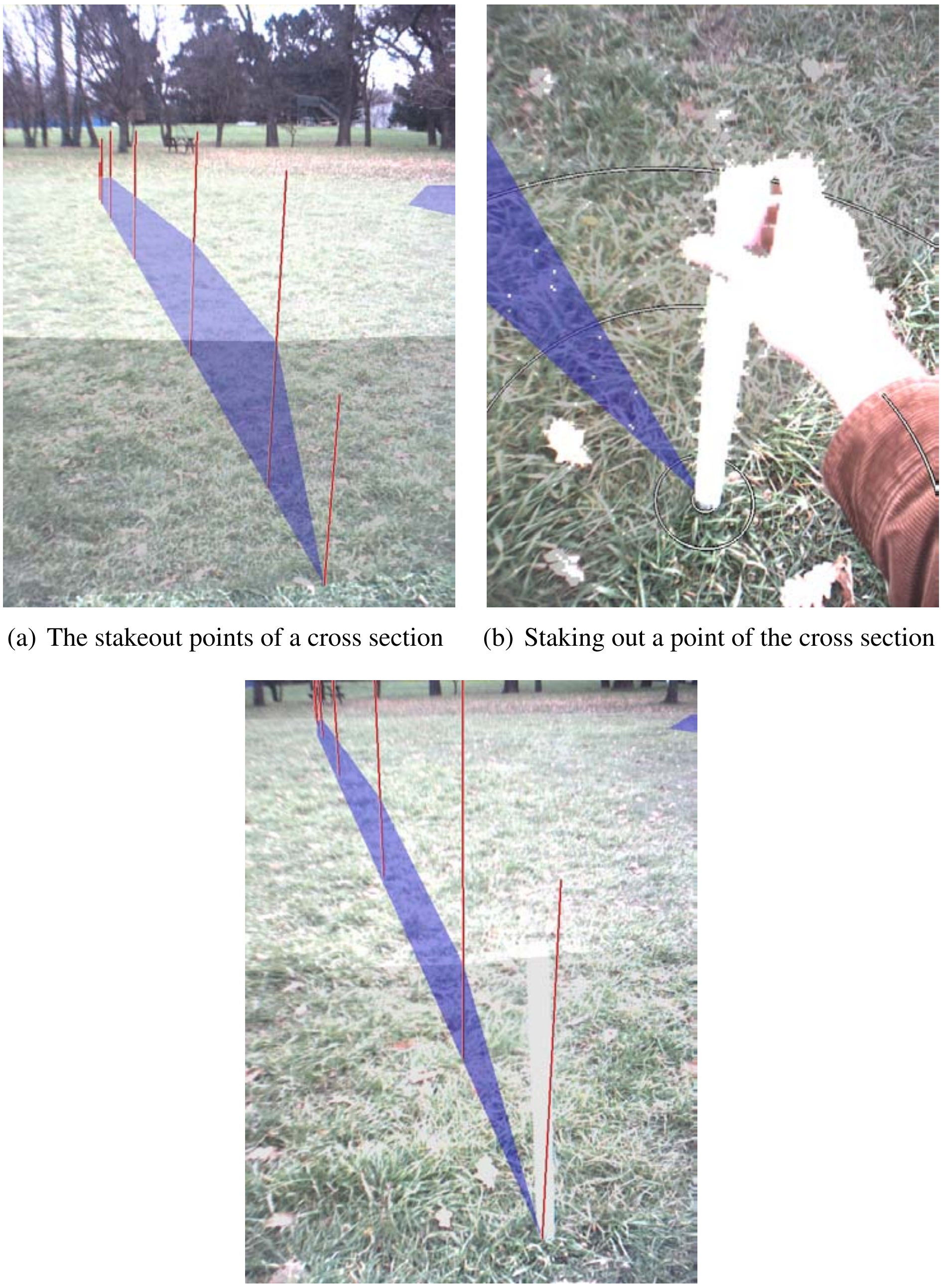


(a) A cross section visible through the road cover

(b) As the user approached a cross section, the road cover was made highly transparent

(c) Far away cross sections were still hidden under the road cover

Figure 5.2: When the user approached a cross section, more detail was revealed.



(a) The stakeout points of a cross section (b) Staking out a point of the cross section

(c) A staked out point

Figure 5.3: The visualisation provided enough detail to stake out the points.

When the user walked up to a cross section (figure 5.2(a)) and reached a distance of 2 meters, the road cover adjacent to the cross section was made highly transparent to reveal the shape of the cross section (figure 5.2(b)). Far away cross sections were still hidden under the road cover so as not to clutter the screen with currently irrelevant information (figure 5.2(c)). This meant that the road visualisation combined transparency and cutaway views as described in section 2.4. A stationary cutaway view revealed the cross section. Such a visualisation is preferable to a simple tunnel cutaway if prior semantic knowledge of the virtual model exists. The cutaway did not simply cut out the road cover to reveal the inner structure of the road. Instead, the road cover was still visible at very high transparency, providing contextual information about the shape of the road. Both the light gray part as well as the dark gray part of the road cover were rendered at 40% opacity in figure 5.2(b). The dark part was more visible at high transparency. The parameters for this visualisation – such as distance of the observer and opacity – were based on an iterative design that involved not only the researcher but also stakeout experts from Trimble.

It should be noted that the road in figures 5.1(b) and 5.2(a) was rendered at 95% opacity. This was to ensure that the virtual road did not completely occlude real world hazards such as holes.

As the user walked closer to an individual cross section, the individual stakeout points were shown (figure 5.3(a)). These stakeout points were equivalent to the offsets shown in figure 2.14. When the user approached a point, the virtual stake turned into a bull's eye visualisation that let the user stake out the location (figure 5.3(b)). Note how the vision based tracking algorithm that provided correct occlusion of the pole left artefacts in the transparent road cover due to tracking noise. The algorithm was originally written for the correct occlusion of the bull's eye visualisation which did not show tracking errors so prominently. Figure 5.3(c) shows the stake's location with respect to the cross section.

The implementation used OpenGL to render virtual objects transparently. Polygons were blended in hardware at negligible cost to performance. To achieve correct transparency effects in OpenGL, polygons have to be drawn according to their depth order from back to front. Otherwise, polygons occluded by transparent polygons will be discarded based on depth buffer values. An example of this can be seen in figure 5.2(b), where the dark gray backside of the road was not rendered. This was because the top part of the road was drawn first, filling OpenGL's depth buffer and resulting in the polygons for the back section being discarded. A future implementation of the visualisation will have to render transparency correctly.

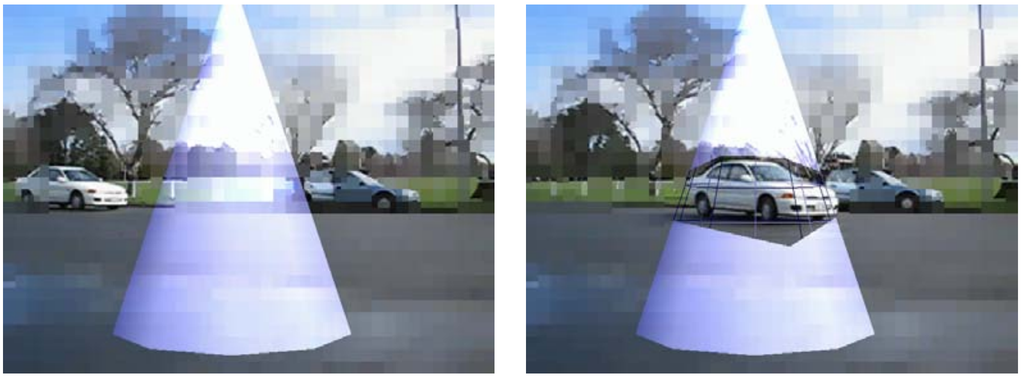
As can be seen in figure 5.3(b), the occlusion algorithm of the real stake will have to be improved to reduce tracking noise. In addition, the algorithm will need to be able to recognise narrower stakes as seen in figure 5.3(c).

In summary, the road visualisation showed both the road design as well as the details of every cross section without the need for user input. The visualisation revealed the detail obscured by the road cover without losing the context provided by the road cover.

Informal interviews with stakeout experts who tested the road visualisation showed that the visualisation conveyed the road structure and the road components well, and made it easy to access detail information.

5.2 Virtual Objects Occluding Real Objects

Virtual objects may occlude large parts of the user's field of view, as mentioned in the previous section. The road visualisation used different degrees of transparency not only to make virtual objects visible, but also to not obstruct the user's view of the real world completely (figure 5.1(b)). In WOAR, virtual objects occluding the real world may be a safety concern. For example, the road visualisation may occlude holes or other hazardous terrain. To address this problem, this section explores the use of selective transparency to make hazards visible.



(a) A virtual object occluding a part of the user's field of view.

(b) A fast moving object is recognised and given priority on the screen.

Figure 5.4: Avoiding occlusion of moving cars.

Inspired by the idea of the environment management system [MacIntyre & Feiner 1996], this section describes an application that aimed to ensure that fast moving real objects would not be obscured by virtual targets. Figure 5.4 shows how the application tracked fast moving objects such as cars and cut a hole in virtual objects to make sure the user is notified of the potential danger. A wireframe view of the virtual object replaced the solid view of the object in the cutout part of the image. This provided a nearly unobstructed view of the moving object while maintaining the integrity of the virtual object.

The exploratory implementation used the OpenCV [1] library to compute the optical flow in the user's field of view. The implemented algorithm identified areas of large uniform flow and used OpenGL's stencil buffer to provide a tunnel cutaway through any virtual object that might obscure the fast moving image areas.

To find fast moving areas in the image, the algorithm first subdivided the image into 64 regions. It then tagged a region of the image as fast moving if the optical flow algorithm showed that enough points moved over a certain threshold. This meant that the algorithm recognised all types of fast movement, not just uniform movement. The algorithm then created convex hulls around connected fast moving image parts to create areas that were cut out of the virtual imagery using OpenGL's stencil buffer. The use of a convex hull was to ensure that fast moving objects would be shown completely, even if only parts of it were recognised. However, it also meant that the area could include stationary parts of the image. The algorithm did not run in real-time, and worked on video files for input and output instead.

The simple algorithm required the background to be stable. This meant that the camera had to be stationary. An implementation for a WOAR system would have to support a moving camera. However, differentiating between camera movement and object movement is not trivial. Points in the image may move at different speeds depending on their distance from the camera and the camera movement. This means that the WOAR system would need to have a model of the real world in order to calculate the expected camera movement for each tracked point in the image. For further reading see for example Thakoor et al. [2004].

[1] `http://opencvlibrary.sourceforge.net/`

Apart from the technical challenges, there were several usability problems with the approach. To begin with, it would be impossible for a WOAR system to accurately classify each object into a potential hazard or a harmless object. The described algorithm only recognised fast moving objects. Not all fast moving objects are dangerous, and slowly moving objects may be hazardous as well. Considering that a WOAR stakeout system may be used at a construction site, many dangers would be unrecognised. This means that the system would be able to only highlight some potential hazards, which could result in the user ignoring hazards that the system did not recognise. For example, since the implementation was not able to classify moving objects by their potential danger, it blindly highlighted each fast moving object, including shadows. Lastly, the tunnel cutaway visualisation might not be the most appropriate solution. As discussed in section 2.4, it does not necessarily provide the user with good depth ordering cues.

In order to automatically recognise all possible hazards in an environment – for example a construction site – a collection of sensors could be used. Trucks, cranes, excavators, etc. could be tracked using GPS. Holes, trenches, string lines, and other tripping hazards could be manually surveyed and registered as they are created. However, this seems neither realistic nor productive.

In summary, a usable implementation of a tracking algorithm that recognises and classifies hazards is complex, and a bigger, integrated system is likely required. Additionally, a system that only visualises some dangers but ignores others may give users a false sense of security, thereby creating potentially dangerous situations.

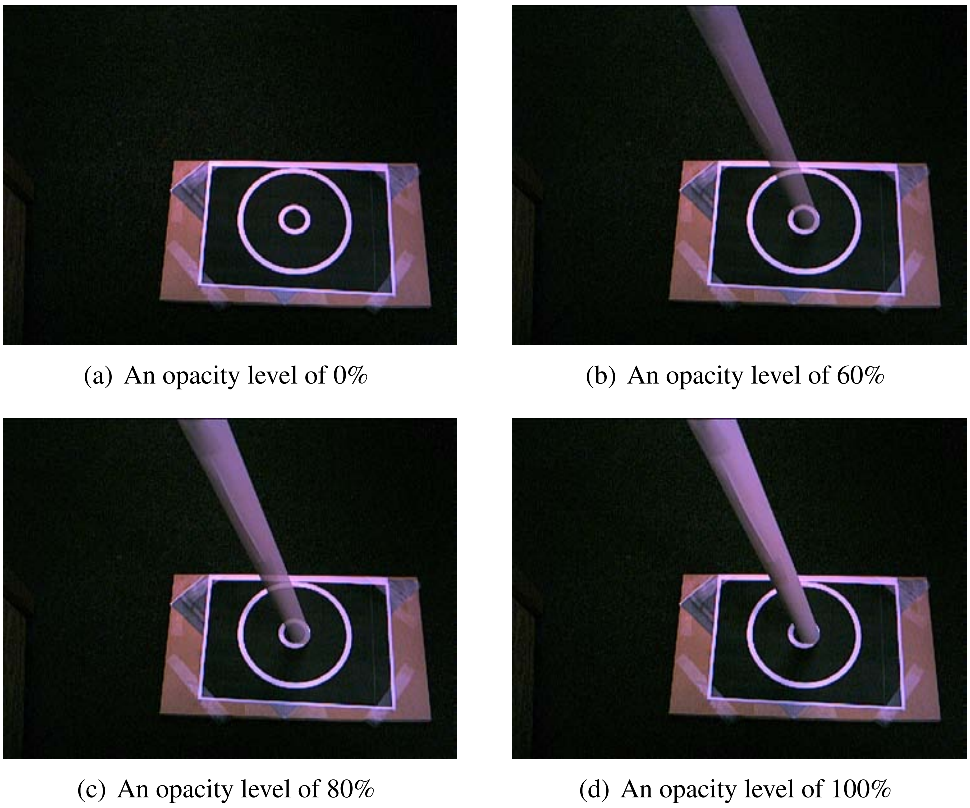
5.3 Real Objects Occluding Other Real Objects

During the process of placing the stakeout pole on a stakeout marker, the pole obscures the marker (figure 5.5(d)). In the experiments described in chapter 4 and 6, participants noted that placing the pole accurately might be easier if the pole was made transparent, as it obscured the stakeout marker when it was most important.

This section describes a system that controlled the level of transparency of stakeout poles, hands, and tools in real-time. Several perceptual issues with artificial transparency are identified, and possible solutions proposed. The results of this implementation will again be used in section 5.4 to make a real stakeout pole transparent with respect to a virtual stakeout marker.

Artificially transparent stakeout poles are to be preferred over stakeout poles made from transparent material. Transparent materials will cause distortion, making the obscured marker harder to see. A pole made from transparent material would possibly also have to be cleaned between stakeout tasks.

5.3.1 An Artificially Transparent Stakeout Pole



(a) An opacity level of 0%

(b) An opacity level of 60%

(c) An opacity level of 80%

(d) An opacity level of 100%

Figure 5.5: A real pole made artificially transparent

A simple, desk based AR application with a static video see-through interface explored how well artificially transparent poles may be used for stakeout with real poles and markers. The implementation avoided the need for identifying the pole in the current camera frame and used a very simple approach to background restoration.

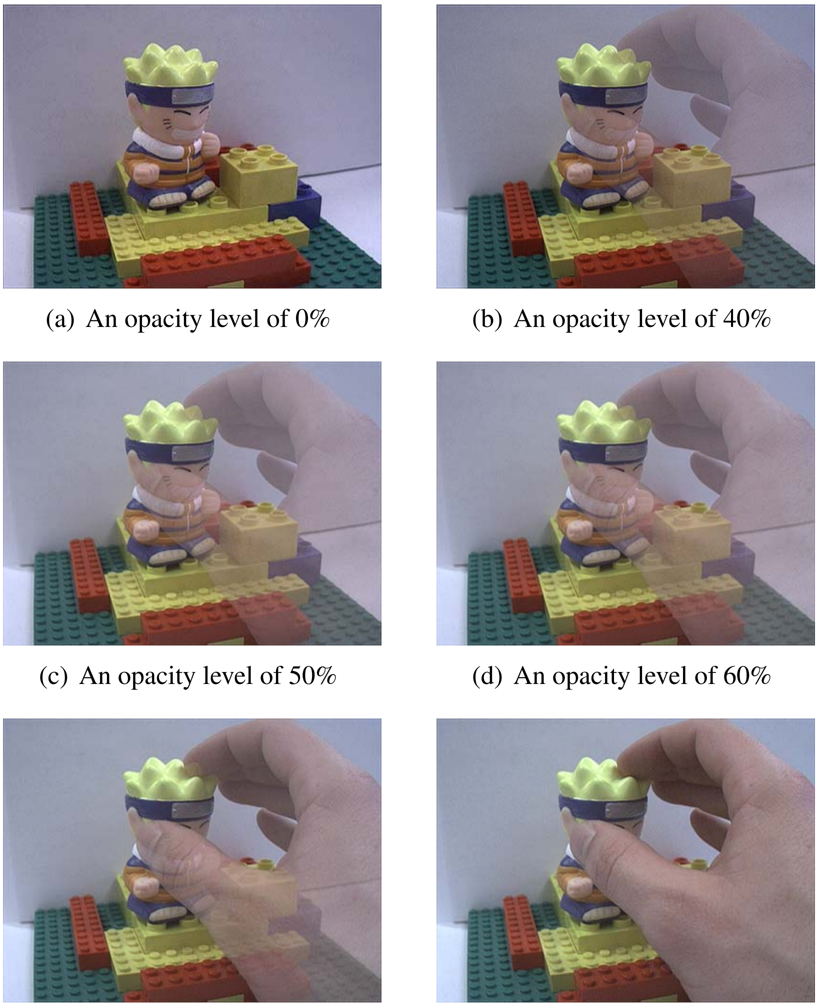
The camera was fixed and could not be moved by the user, and the background had to be static. The users viewed the generated image through a head-mounted display at roughly the same position and pose as the fixed camera. The HMD did not touch the camera so as not to accidentally move it. At startup, the application stored a picture taken by the camera as the background image (figure 5.5(a)) and then blended each subsequent frame against this background picture. The user could then freely move the pole, resulting in the system presenting a transparent pole to the user. Figure 5.5 shows the result of several opacity levels.

Alpha blending was done in hardware, using the OpenGL library. Video was displayed at a resolution of 640*480 pixels at 15 frames per second in an i-glasses HMD.

The setup imposed several restrictions on what the user perceived. Stereoscopic depth cues were removed, and cast shadows were gradually removed by making them as transparent as the user's hand. Making cast shadows less salient may have a negative effect on the shadows' efficiency [Kersten et al. 1997]. The indoor setup with a static background and constant lighting would also be different from outdoor situations. In addition, decoupling the HMD and the camera sometimes confused users who expected to be able to use parallax as a depth cue. Although the viewpoint could not be changed, this simple setup provided valuable insights in how users could use such a transparent interface.

This implementation visualised an artificially transparent pole in real-time at different levels of opacity (figure 5.5). This system revealed several possible perceptual problems with artificially transparent stakeout poles: if the pole was made too transparent, placing the pole on the marker seemed harder. If the pole was made too opaque, it occluded the marker again.

An informal user study explored perceptual issues with artificial transparency for interaction. The explorative study used artificially transparent hands, tools, and Lego bricks instead of stakeout poles, as such a setup was more accessible for the users. For example, users could explore interaction with a pen or a Lego brick longer than with a comparatively large and heavy stakeout pole. This also allowed for faster prototyping of possible solutions. At the end of the explorative study, the findings were applied to a visualisation of the stakeout pole.



(a) An opacity level of 0%

(b) An opacity level of 40%

(c) An opacity level of 50%

(d) An opacity level of 60%

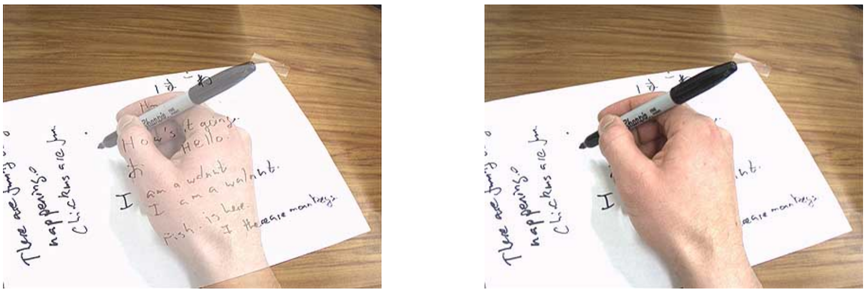
(e) An opacity level of 80%

(f) An opacity level of 100%

Figure 5.6: Grasping an object with different levels of opacity of the hand.

5.3.2 Uniform Transparency

The first explorative implementation rendered a user's reaching hand at the opacity levels 20%, 40%, 60%, 80% and 100% (see figure 5.6). An informal study evaluated the implementation. Students from the HIT Lab NZ gave their opinion on the usability of the system for reaching towards and touching objects (figure 5.6), writing (figure 5.7), putting a screwdriver on a screw and inserting a PCI card into a computer. The pen, screwdriver and PCI card were rendered at the same transparency level as the user's hand. Each of these tasks started with an opacity level of 100% to let the users get accustomed to the task and then increased the level of transparency. Since this was an informal pilot study, no data was recorded automatically and there were no subjective questionnaires.



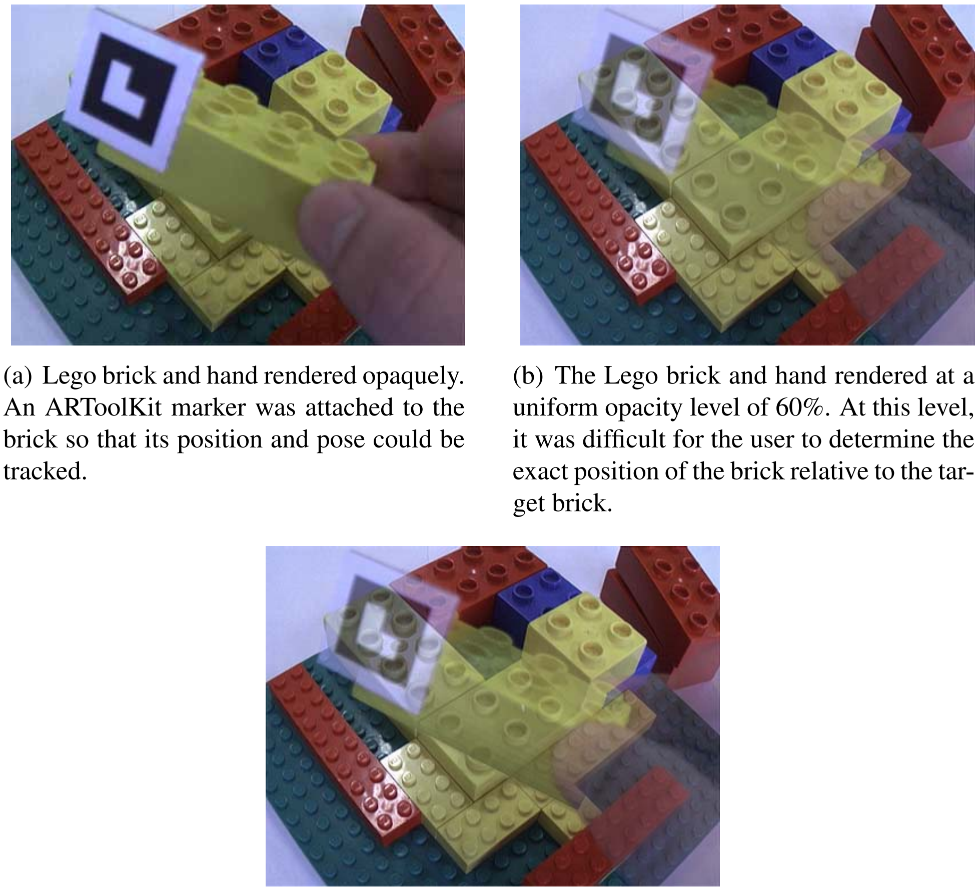
(a) An opacity level of 50%

(b) An opacity level of 100%

Figure 5.7: Writing with and without transparent rendering of the hand. Note how the writing appears to be on the transparent hand rather than on the paper.

All users found the system fascinating to use and were certain that it would help them in situations where their hands or tools would cover relevant background structures. Students generally preferred opacity levels 60% and 80%. They said that interaction was about as easy as with 100% opacity, while providing a "nice balance" between the hand and the background. Opacity levels of 50% and below were regarded as too transparent for most tasks. Typical comments on 50% opacity were "it felt a bit more alien", "kind of weird. . . your mind was playing tricks on you" and "it felt like your hand was not really there", while levels 40% and 20% were completely dismissed. When comparing it to opacity level 50%, one student said of 60% opacity that it "felt more like I'm actually fiddling with the object". One student found that an opacity level of 80% was not transparent enough to reveal information about the background.

A general observation was that the perceived level of transparency varied depending on the background colour. If the background was darker, the hand appeared more transparent, while if the hand was darker than the background, it appeared less transparent (see for example figure 5.6(c)). This dependency of observed transparency is a potential problem for accurate perception. If this affects performance, transparency should adapt to the background light conditions.



(a) Lego brick and hand rendered opaquely. An ARToolKit marker was attached to the brick so that its position and pose could be tracked.

(b) The Lego brick and hand rendered at a uniform opacity level of 60%. At this level, it was difficult for the user to determine the exact position of the brick relative to the target brick.

(c) The bottom edge of the brick rendered with 90% opacity. Now it was much easier for the user to place the brick.

Figure 5.8: Rendering the relevant edge of a Lego brick at high opacity improved interaction.

The users reported that they focused on the edge regions of the hand, pen or PCI card that were relevant for the current task. With increasing transparency, the focus shifted from the edge regions to the actual edge. One user said that he became distracted by the background visible through his hand if he did not concentrate on the PCI card's edge.

There were two perceptual problems for opacity levels of 50% and lower. The first was that high transparency of the hand diminished occlusion as a depth cue. Users reported that it seemed as if the hand might be behind the background objects. For example, letters in the background seemed to be painted on the hand (figure 5.7(a)). See figure 5.6 for how the index finger seems to disappear behind the figurine at low opacity. Occlusion is one of the most important depth cues, as for example shown in chapter 4. Bingham et al. [2001] found that elimination of occlusion had no effect when binocular vision is present. This suggests that stereoscopic vision might allow for higher transparency of the hand. The second problem was that users felt that they had less control of their hand when it was very transparent, especially for precise movements. Mason et al. [2001] found that in an AR environment, reaching movements took longer when vision of the reaching limb was removed.

5.3.3 Selective Transparency

Based on these results, an object might be treated as having regions of different task relevance. Most of the hand is not directly relevant for precise interaction, but is used for perceiving overall hand pose. Only those edges of the hand that touch other objects are relevant for precise interaction. In the case of the PCI card, users mainly concentrated on the edge of the card that was inserted into the slot. To make these relevant edges more visible to the user, a modified application used the ARToolKit to track the position of a Lego brick (figure 5.8(a)) and rendered its bottom edge at very low transparency (figure 5.8(c)).

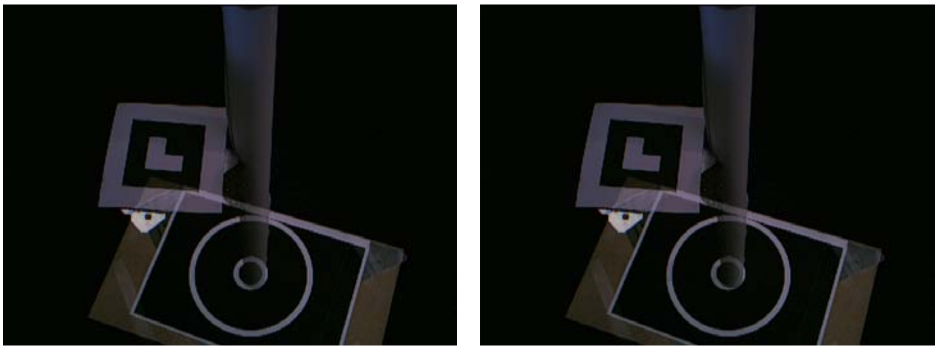
This implementation explored if such a visualisation allowed for greater levels of transparency of the object's main regions. However, users still did not like low opacity levels, and 60% and 80% were again picked as the favourite levels. Overall, users liked the clearer view of the brick's edge that this interface provided and they were able to place the brick faster.

However, some noted that making the brick's edge less transparent made it more difficult to see the relevant part of the target brick. Future implementations should explore how thin such a low transparency region can be in order to provide better visibility of the occluded target structures.

It might also be helpful to investigate the outlining of the brick with a wireframe model. This was a technique used by Livingston et al. [2003] and Tsuda et al. [2005] to resolve occlusion problems with transparent layering.

5.3.4 An Improved Artificially Transparent Stakeout Pole

Based on the results of the explorative study, the transparent stakeout pole was revisited. Figure 5.9 shows the pole with uniform transparency (figure 5.9(a)) and with selective transparency (figure 5.9(b)). With the bottom tip of the pole being most relevant for stakeout, lower transparency values were applied here (figure 5.9(b)).



(a) A uniform opacity level of 60%

(b) An opacity level of 60% with selective transparency

Figure 5.9: A real pole made artificially transparent with uniform and selective transparency. The two images are only different in the opacity level at the bottom edge of the pole.

The results did not match the success with selective transparency of the Lego block. This is in part due the high contrast of the stakeout marker used. The pole was easy to see over the black part of the marker at low opacity values, while higher opacity values were necessary over the white parts of the marker. In figure 5.9(a), the bottom edge of the pole is hard to see, while the inner circle of the marker is highly visible. With selective transparency applied, the bottom edge of the pole is highly visible, while the inner circle is harder to see (figure 5.9(b)).

Additionally, the area on the screen that is relevant for interaction is much smaller than with the Lego block. The bottom edge of the pole would need to be enhanced very thinly, for example with a wireframe outline. The tracking capabilities of the explorative implementation were not accurate enough to make such an augmentation believable.

5.3.5 Conclusion

The section presents an implementation of artificial transparency for hands and stakeout poles, and reported perceptual issues with the interface. Users found that transparency helped them, but they also saw drawbacks. At higher transparency levels, the lack of occlusion led to conflicting depth cues. Furthermore, the lack of visual feedback at high transparency caused some users to report feeling reduced control over their hands.

The implementations provided both uniform and selective transparency according to task relevance of different regions of a Lego block. Users liked the decreased transparency of important regions in the selective rendering, but found that with this reduction of transparency, important background details were once again obscured. The implementation of selective transparency for a stakeout pole was not as successful. This was partially because the stakeout marker's relevant area for interaction was much smaller.

The initial results were promising, and the next section describes a WOAR implementation that explores how real poles can be made artificially transparent so as not to obscure virtual markers.

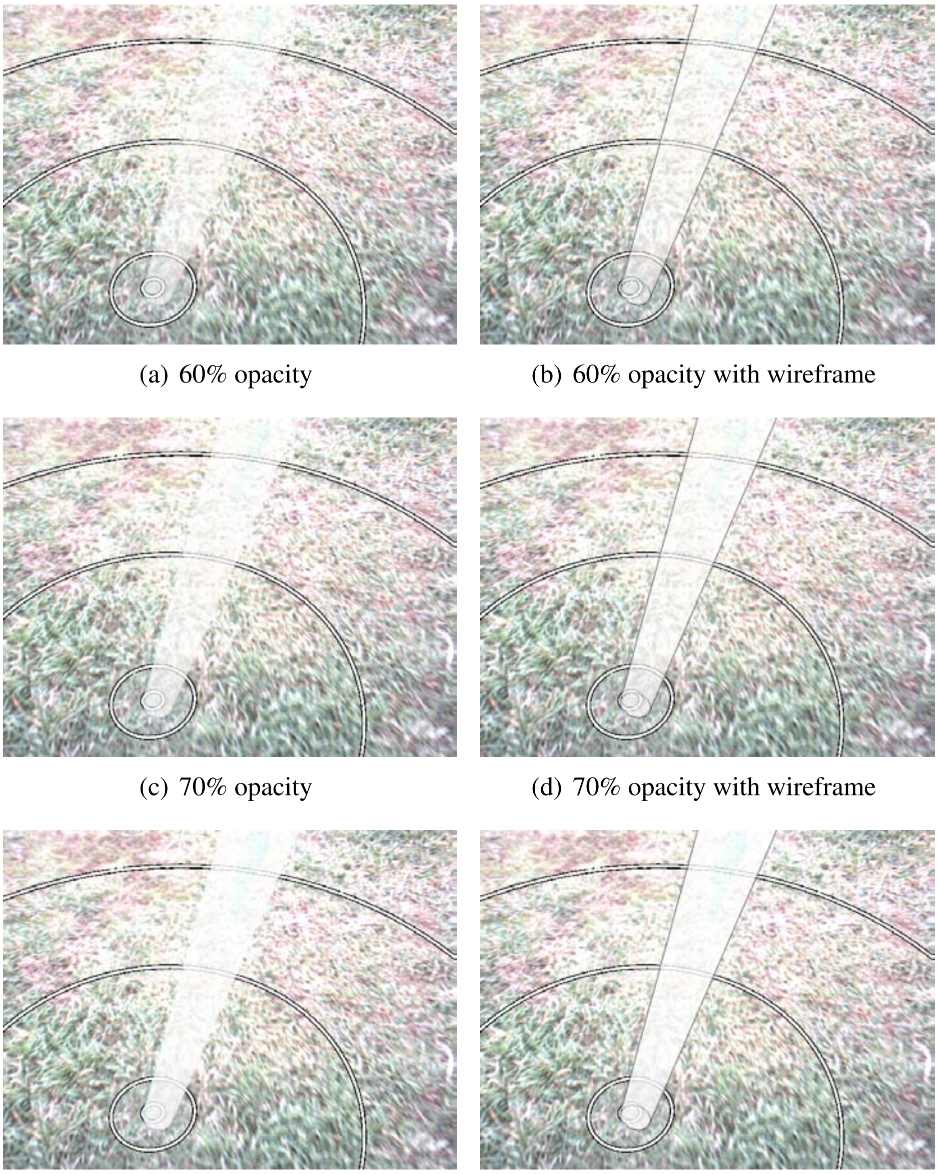
5.4 Real Objects Occluding Virtual Objects

This section describes a WOAR implementation that made a real stakeout pole transparent to prevent it from completely obscuring virtual stakeout markers.

First, this was explored with static mock-up images, and then with a real-time WOAR implementation. Prototyping with static images was chosen, as this allowed the exploration of interface options such as wireframe outlines without the need for accurate tracking.

5.4.1 Static Prototypes

An informal study asked students at the HIT Lab NZ for their opinion on static mock-ups that showed a real pole occluding a virtual target marker at different levels of opacity (figure 5.10). The study aimed at finding the lowest opacity setting for the pole that made the observer perceive it to still be in front of the marker. Based on the findings from the previous section, an opacity level of 60%



(a) 60% opacity

(b) 60% opacity with wireframe

(c) 70% opacity

(d) 70% opacity with wireframe

(e) 80% opacity

(f) 80% opacity with wireframe

Figure 5.10: A static mock-up of a transparent pole at different opacity levels with and without a wireframe border. Note how the real background can be seen through the pole.

could be expected to be ideal for a transparent pole (figure 5.10(a)). However, as previously reported, a light foreground object will appear more transparent than a dark foreground object at the same opacity level. Since the camera captured the pole as nearly completely white most of the time, the pole seemed very transparent at this opacity level. Rendering the pole at an opacity level of 80% (figure 5.10(e)) made the pole appear more solid. However, the virtual markers were now also rather faint. An opacity level of 70% worked best (figure 5.10(c)).

Next, the students looked at static mock-ups of a transparent pole with wireframe outlines (figure 5.10(b), 5.10(d), and 5.10(f)). It was hoped that adding outlines to the pole would make the location of the pole easier to understand and would not obscure the stakeout marker, as selective transparency did in the previous section.

Similar to the results from using Lego bricks with selective transparency, adding outlines did not cause students to tolerate a lower level of opacity. However, it did aid them in understanding the pole's shape and location. Note that the wireframe outline also revealed the occluded bottom edge. This was seen as a significant improvement.

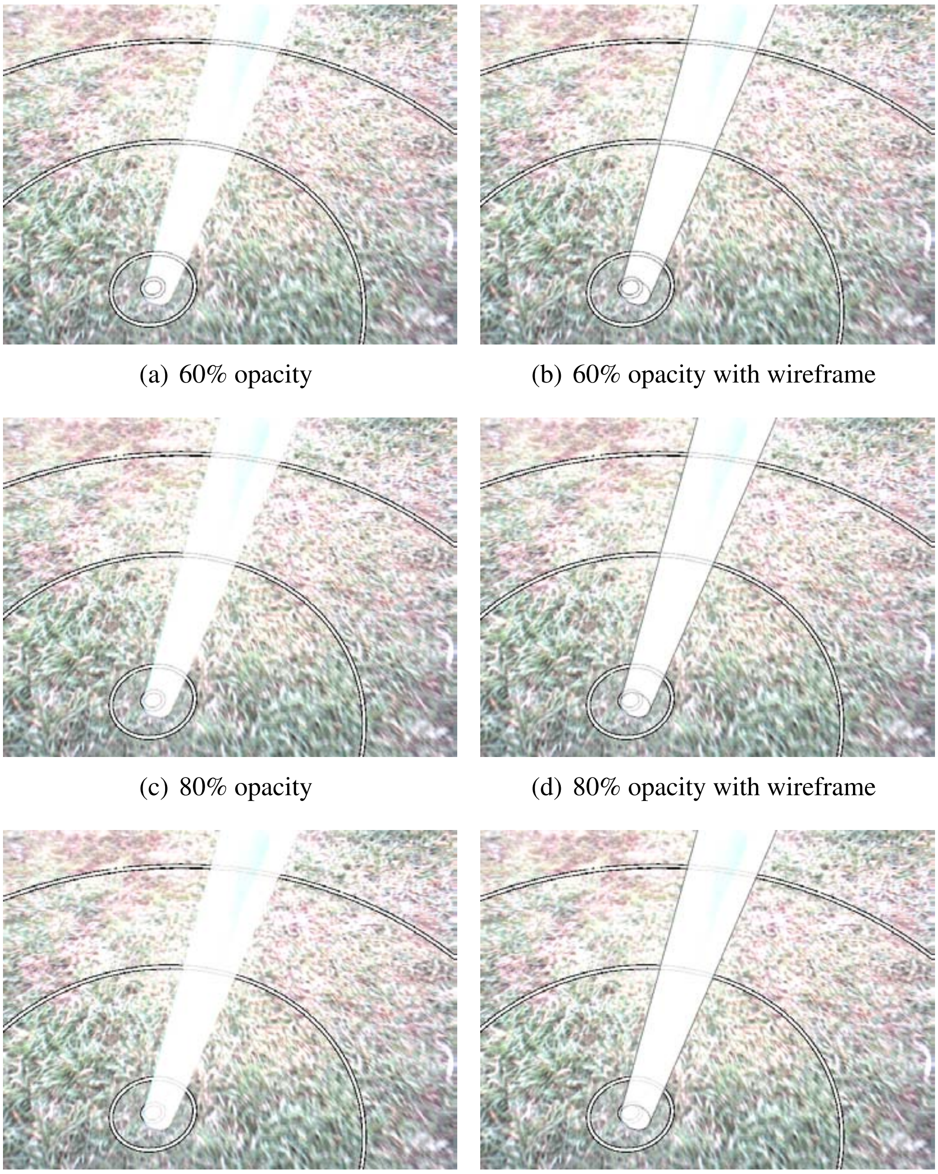
The students thought that an opacity level of 70% with an added wireframe outline was the best visualisation of the ones shown in figure 5.10.

The real-time implementation of such an interface for a WOAR system could be expensive, as it would require background reconstruction to provide pole transparency as well as accurate pole tracking in at least 5 degrees of freedom to provide a wireframe outline. The next section describes the exploration of a method that is easier to implement.

5.4.2 Transparency with Respect to Virtual Objects

This section explores the possibility of making the pole transparent only for the virtual target while it is fully opaque with respect to the real background (figure 5.11). A WOAR system could easily implement this solution, as it only required 2-dimensional tracking of the pole in the image plane. An appropriate tracking algorithm is described in section 4.10.

Static mock-ups showed that the pole appeared to be occluded by the target at opacity level 60% (figure 5.11(a)). At 80% (figure 5.11(c)), observer opinions were divided as to how helpful the system was. At 90% opacity (figure 5.11(e)), most observers agreed that the pole occluded the virtual target. However, at this level of opacity, the virtual target was relatively hard to see.



(a) 60% opacity

(b) 60% opacity with wireframe

(c) 80% opacity

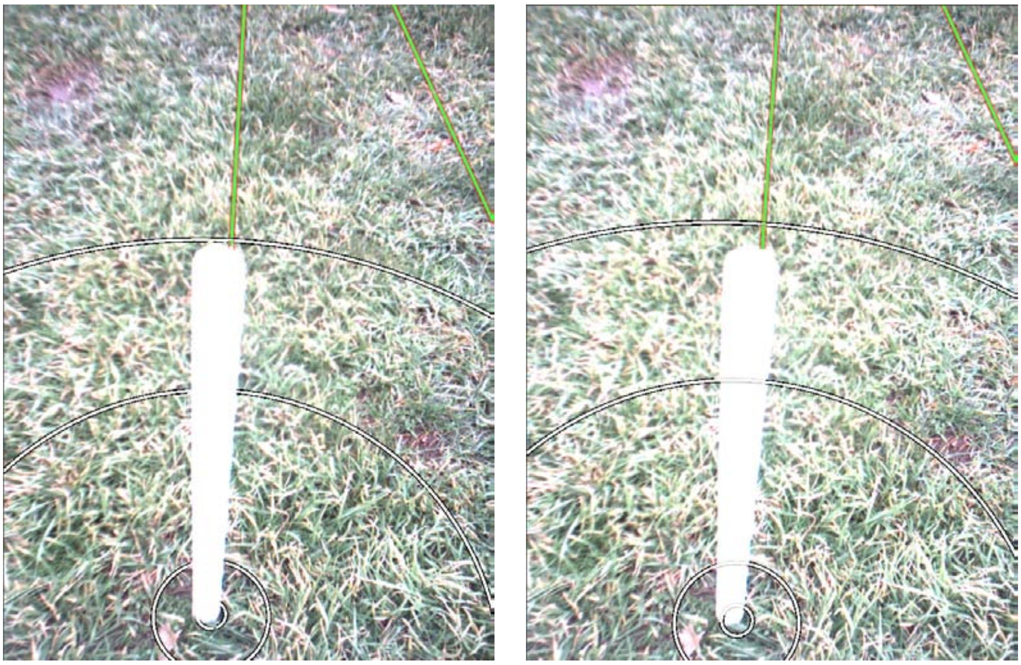
(d) 80% opacity with wireframe

(e) 90% opacity

(f) 90% opacity with wireframe

Figure 5.11: A static mock-up of a pole with different opacity levels with respect to the virtual target with and without a wireframe border. Note how the pole is opaque with respect to the background.

Adding a wireframe outline (figures 5.11(b), 5.11(d), and 5.11(f)) made depth ordering less confusing for opacity level 80%. However, most observers still preferred an opacity level of 90%.



(a) The real pole rendered in WOAR with correct occlusion

(b) The real pole rendered transparent with respect to the virtual stakeout marker

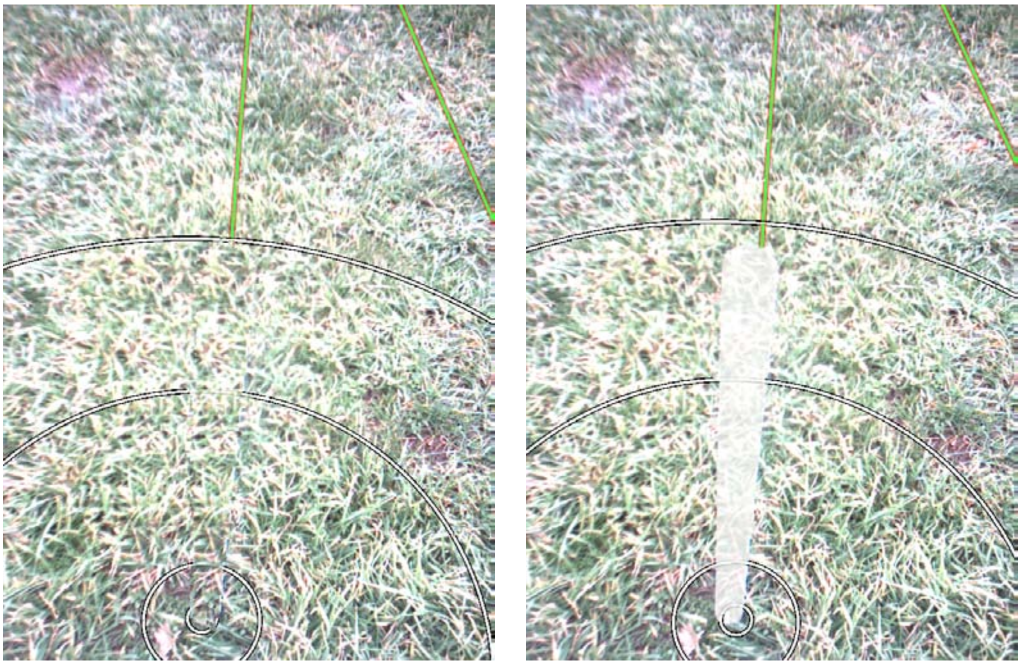
Figure 5.12: The real pole in WOAR rendered transparent with respect to the virtual stakeout marker

5.4.3 WOAR Implementations

This section describes algorithms that rendered a real pole transparent in WOAR with respect to virtual objects and the real background based on the findings of the previous section.

First, the optical pole tracking algorithm described in section 4.10 was used to change the transparency level of the virtual target marker over those pixels identified as being part of the pole to achieve an effect as seen in figure 5.11(e). This visualisation (figure 5.12) did not work as well as the feedback for the static mock-ups suggested, as the pole appeared to be occluded by the marker. A good transparency level that allowed for the visualisation to work satisfactorily could not be found.

It is unclear why there was a difference between the mock-ups and the WOAR implementation. It could be that the movement of the pole had an impact on depth ordering perception. However, it is more likely that with the static images, users had ample time to interpret the presented imagery, while in the actual process of stakeout, depth cues must not be ambivalent or in need of interpretation. A formal user study is needed to explain this.



(a) Background substitution of the real pole

(b) The real pole rendered transparent

Figure 5.13: Making the real pole artificially transparent in WOAR

A second algorithm rendered the pole transparent with respect to both the real and virtual parts of the user's environment (figure 5.13(b)). This implementation made use of the fact that the real background was not relevant to the interaction, as the user's task was to align the real pole with the virtual marker. This meant that a rough approximation sufficed for background reconstruction.

The simple algorithm relied on the background being textured with random noise, as this is the case for grass. The algorithm cut out a part of the image that was not recognised as being part of the pole and copied it into the tracked location of the pole (figure 5.13(a)). For each pixel that had been identified as being part of the pole, the algorithm copied the pixel 90 pixels to the left into the current one. If that pixel was also part of the pole, the algorithm probed the pixel 91 pixels to the left of the current one instead, and so on.

The algorithm is only appropriate for prototyping of artificially transparent stakeout poles. In real applications, target locations may by obstructed by objects such as rocks. This implementation would render these obstructions invisible.

The replacement action was mainly noticeable through seams at the edge of the pole. When the user moved their head, the copied part of the background moved in roughly the same direction and at roughly the same speed as the background surrounding the pole. This meant that coherent motion was presented to the user, and the illusion of background reconstruction worked well.

This algorithm was used to render the stakeout pole transparent with respect to both to real and the virtual parts of the environment (figure 5.13(b)). This visualisation was more convincing than a pole that is only transparent with respect to the stakeout marker, and worked well for stakeout at an opacity level of 70%.

In conclusion, a fully transparent pole (figure 5.13(b)) is preferable to a pole that is only transparent with respect to virtual objects (figure 5.12(b)). A fully transparent pole with wireframe outlines (figure 5.10(d)) is likely to be an even better option, but this could not be tested in the WOAR application, as it would have required tracking of the pole in five degrees of freedom.

5.5 Conclusion

This chapter explores visualisations of obscured information for all four types of obscured information visualisation.

Section 5.1 demonstrates a successful visualisation of a road design that enabled surveyors to see both the overall shape of the road as well as the internal structure. Using OIV techniques such as transparency and stationary cutaways, the road visualisation let users stake out the points of a road design. Using information filtering techniques, the visualisation automatically changed based on the user's location. This allowed users to concentrate on the stakeout task rather than on controlling the interface.

Section 5.2 demonstrated a system that aimed at recognising and visualising potentially hazardous objects in the user's field of view. However, the research found that recognising hazardous objects is a non-trivial problem that is outside the scope of this thesis. Providing the user with warnings for some hazards but not for others may be more dangerous than providing no warnings at all. With no warnings provided, the user will stay more alert.

Section 5.3 demonstrated implementations that let users interact with artificially transparent stakeout poles and hands. Test users found that interaction with transparent stakeout poles and hands was possible at a transparency level that lets users see occluded objects. However, an optimal visualisation of a real stakeout pole occluding a real marker could not be found. This was most likely due to the high contrast of the stakeout marker.

Section 5.4 demonstrated a WOAR implementation that made a real stakeout pole transparent in order to not occlude virtual stakeout markers. The implementation indicated that the pole had to be made transparent with respect to both the real world and the virtual markers for users to resolve depth ordering correctly.

The best opacity levels found with the implementation are not universally applicable, as appropriate opacity levels depend on the intensity of the foreground and background object. Further research is required to predict the best opacity level depending on the intensity of the foreground and background objects.

In conclusion, OIV techniques can be used to enhance WOAR interfaces. Further research is needed to establish good transparency levels for interaction with transparent objects to assess the effect of wireframe outlines and to formally evaluate user performance with artificially transparent stakeout poles. As expected, the visualisation of obscured objects becomes more complex the more it relies on tracking or reconstructing real objects. Where these problems have been solved, WOAR interfaces should use OIV techniques in order to present relevant information to the user in an optimal way.

The next chapter describes the developed WOAR stakeout application and presents a formal user study that compared the performance of the application to the Trimble Survey Controller.

Chapter VI

A WOAR Stakeout Application

This chapter describes a WOAR stakeout system and a formal comparison of the system with the state-of-the-art commercial stakeout system Trimble Survey Controller (TSC). Section 6.1 describes the hardware and software of the WOAR system and the stakeout application software that runs on it. Then, section 6.2 and following present a formal user study that compared the performance of the WOAR application that of the TSC. The study found a significant difference in performance, and showed that the WOAR application performed twice as fast as the TSC at an accuracy of 4cm for a specific stakeout task. The chapter also identifies usability issues that users had when using the WOAR system and it identifies different causes of reduced accuracy in the WOAR system.

6.1 The WOAR Stakeout System

The WOAR system was built in cooperation between Trimble Navigation NZ Ltd. and the HIT Lab NZ. Trimble Navigation is a provider of high-end positioning equipment, and manufactures the RTK grade GPS receiver used in the WOAR system as well as the Trimble Survey Controller described in section 2.5. An earlier working version of the system was completed before the author began his research. The current, improved version had several modified components which are described in detail in this chapter.

6.1.1 Hardware

The system consisted of two main parts: the backpack (figure 6.1), which contained a GPS receiver and a laptop; and the helmet (figure 6.3), which contained the sensors and the head-mounted display. The complete system (figure 6.2) weighed 11.2kg, and consisted of the following hardware:

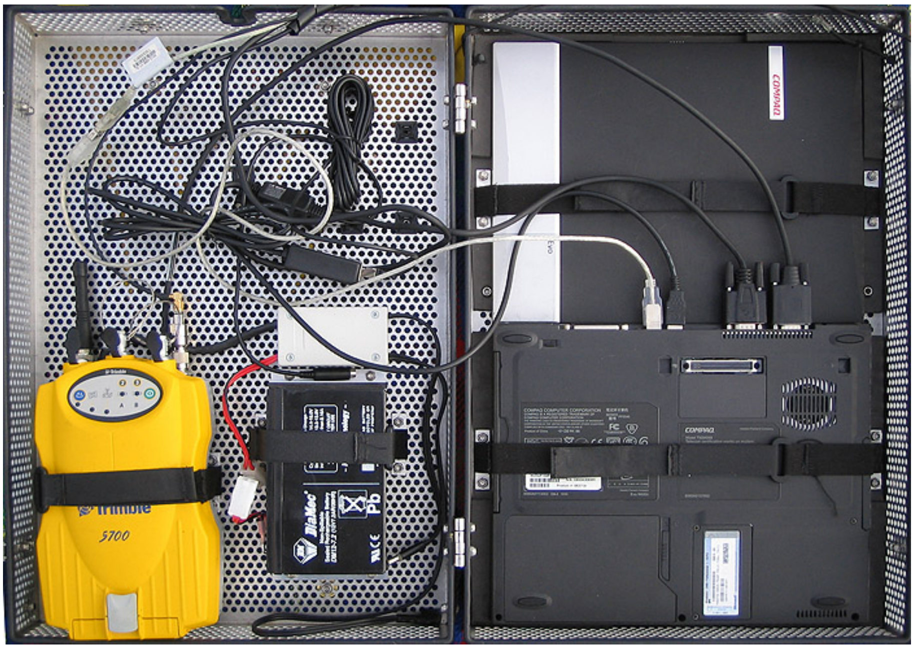


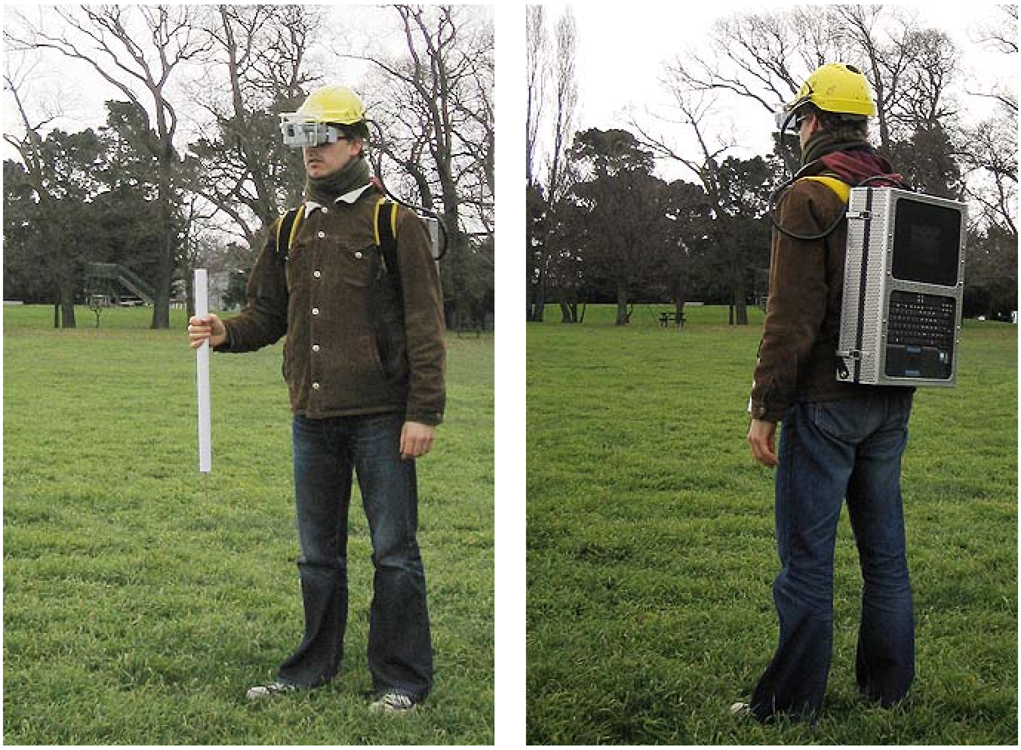
Figure 6.1: The backpack contained an RTK GPS receiver, a battery and a laptop (from left to right)

- A Compaq Evo N620c laptop with a 1.5 GHz Intel Pentium CPU and 1GB RAM
- A Trimble R7 RTK grade GNSS and GPS receiver
- An InterSense InertiaCube 3 three degrees of freedom orientation sensor
- An Aplux USB 2.0 webcam with a resolution of 800*600 pixels at a frame rate of 15 fps and a horizontal field of view of 45 degrees
- An i-glasses HMD with a resolution of 640*480 pixels

See table 6.1 for the sensor specifications.

Sensor Calibration

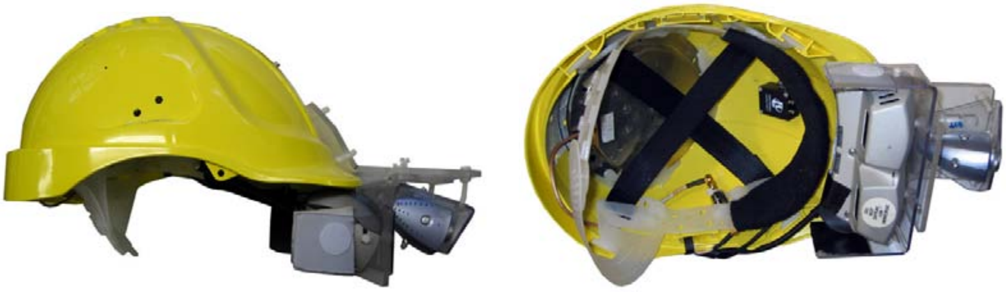
For stakeout, the tracking of the WOAR system needed to be as accurate as possible. This required that the tracking sensors had to be calibrated as accurately as



(a) The WOAR system from the front

(b) The WOAR system from the back

Figure 6.2: The WOAR system in the field



(a) The helmet from the side, showing that the camera was angled at the ground

(b) The inside of the helmet showing the round GPS antenna, the small InertiaCube, the i-glasses and the camera.

Figure 6.3: The helmet

Table 6.1: The sensor specifications

Sensor	Resolution	Accuracy	Sampling Rate	Latency
Trimble R7	1mm	10mm horizontal, 20mm vertical	10Hz	20ms
InertiaCube 3	0.03°	1° yaw, 0.25° pitch and roll	180Hz	2ms
camera	800*600	n.a.	15Hz	155ms

possible. Both the GPS receiver and the InertiaCube orientation sensor provided their own calibration methods that had to be applied independently of one another. The challenge of calibrating the sensor configuration on the helmet remained.

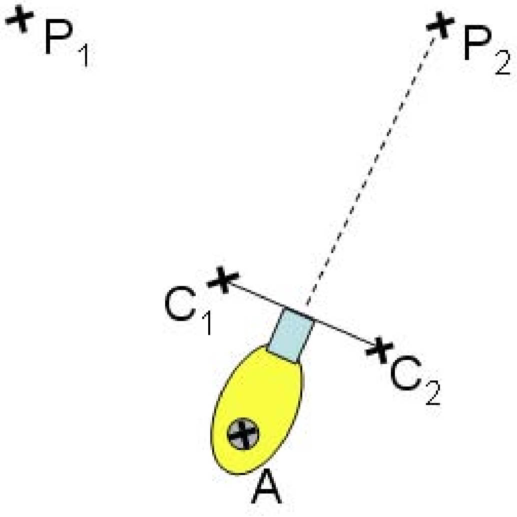
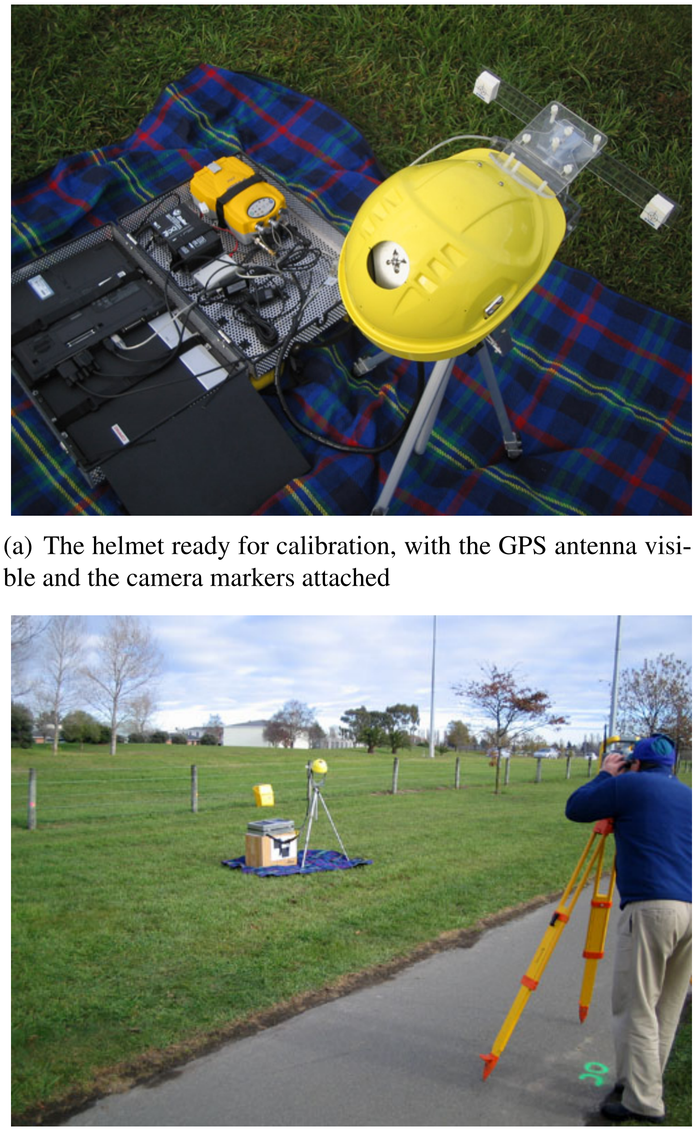


Figure 6.4: The helmet calibration points

To calibrate the sensors on the HMD, a Trimble optical total station was used to survey a set of points that would allow for an accurate calibration of the helmet. Figure 6.4 shows the necessary points for this. Point *A* is the location of the GPS antenna. A hole in the helmet as seen in figure 6.5(a) enabled direct optical measurement of the antenna's location with the total station. All points were



(a) The helmet ready for calibration, with the GPS antenna visible and the camera markers attached

(b) Surveying the calibration points with an optical total station.

Figure 6.5: Helmet calibration

measured without the total station being moved. This meant the location of the helmet and the camera could not both be measured directly. The location of the camera was measured indirectly using the points C_1 and C_2. Figure 6.5(a) shows the target markers used for this. The visible markers P_1 and P_2 were placed on a fence as shown in figure 6.5(b). The camera was centred on P_2, with P_1 lying on the central horizontal axis of the camera image. While the points were measured with the optical total station, data from the WOAR system's InertiaCube and GPS was recorded.

These measurements were used to calculate the matrices C_r and C_t that were needed to calculate the camera's location and orientation as $C_t \cdot C_r \cdot I \cdot R \cdot G$ based on the real-time data provided by the sensors:

G: The current GPS coordinates, already translated into a range that was easier for OpenGL to handle.

R: The rotation from the GPS coordinate frame into the coordinate frame of the InertiaCube. This rotation included the local magnetic declination.

I: The current orientation of the InertiaCube.

C_r: The rotation from the InertiaCube to the camera coordinate frame.

C_t: The translation from the GPS antenna into the camera coordinate frame.

InertiaCube Problems

The relatively low accuracy of the InertiaCube 3, with a possible error of 1° for yaw (see table 6.1), may result in a noticeable horizontal displacement of virtual objects on the ground. For an observer looking at an object on the ground at a distance of 2 metres, an angular error of 1° translates to $tan(1^\circ)2m \approx 3.5cm$. Together with the horizontal accuracy of RTK GPS, this gave the system a maximum accuracy of 4.5cm, to which errors introduced by the HMD and helmet calibration had to be added. This did not meet the accuracy requirements for stakeout. To improve the accuracy of the system, users were required to crouch when staking out a point, thereby halving the horizontal position error introduced by the InertiaCube. The resulting maximum accuracy of 2.8cm was still not ideal but acceptable for some stakeout tasks.

In addition, the InertiaCube is susceptible to magnetic distortion, for example by large ferrous objects. This could be a problem at construction sites where a stakeout system might be used. We experienced problems with distortion introduced by the HMD at certain screen resolutions and refresh rates. This was resolved by moving the HMD and the InertiaCube as far apart as possible and by choosing a screen resolution and refresh rate that did not appear to have an effect on the InertiaCube.

RTK GPS Problems

RTK GPS requires at least five satellites in good geometric distribution to be visible in the sky. There are a number of possible bad geometric distributions, such as when all of the five satellites are too close together. While there usually is a good geometric distribution of five or more satellites throughout the day in New Zealand, this is only true when the whole sky is visible to the GPS antenna. This was often not the case for the WOAR system.

In the WOAR system, the GPS antenna was mounted in the helmet to ensure that it measured the location of the camera accurately. This meant that the antenna would be tilted whenever the user looked up or down. This also tilted the antenna's ground plane, below which it could not receive satellites. To give an extreme example, a 90° inclined antenna can only "see" half the sky. In a WOAR stakeout application, users tend to incline their heads frequently – for example when looking down at a target location. This is a considerable problem for RTK GPS based WOAR stakeout and for WOAR applications in general.

To solve this problem, the helmet was modified to keep the GPS antenna as level as possible. As can be seen in figure 6.3(b), the antenna sat at an angle in the back of the helmet. This meant that it was already inclined circa 20° when the user kept their head level, it was level when the user looked down 20°, and it was inclined 20° again when the user looked down 40°. This gave the user a much wider range of motion with an application driven bias towards looking down.

To reduce head motion for looking up and down, the camera was rotated about 90°, so that its wider field of view was now vertical and it had a resolution of 600*800 pixels. This meant that a 600*450 pixels sized section of the camera image was stretched and displayed as the background image on the 640*480 pixel display. Figure 6.6 shows how a part of the camera image was used for the HMD. The red frame indicates the part of the image that the user could see on their HMD. Note how the original camera image covered both near and far parts of the real environment.

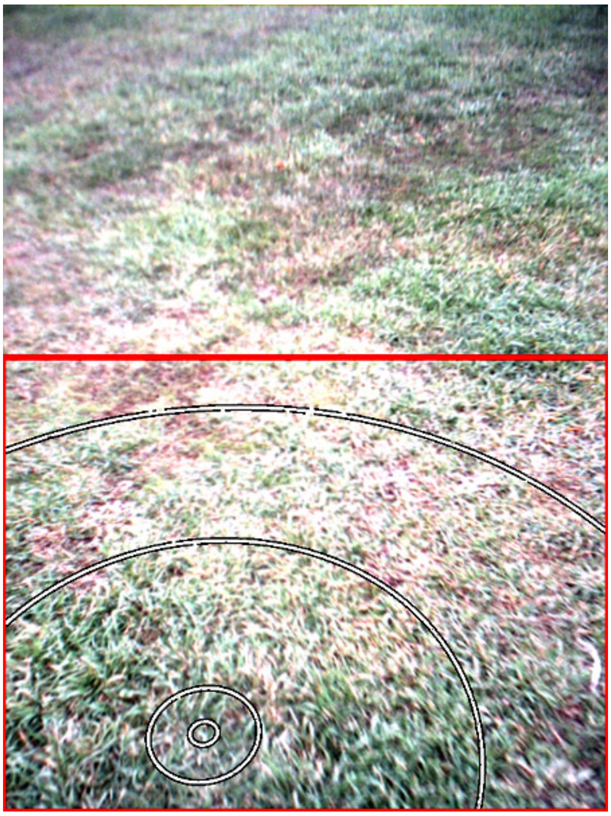


Figure 6.6: A screen capture showing how the HMD only displayed a selection of the rotated camera image, marked here with a red frame.

The camera image area displayed on the HMD was selected based on the user's head inclination. If the camera pitch was less than 15°, the top part of the camera image was selected. If the camera pitch was more than 30°, the bottom part of the camera image was selected. If the pitch was in-between these values, a location for the image section was chosen by linear interpolation between the two extremes. This sliding image selection made sure that it was easy to look down at a target but also allowed users to look ahead without lifting their head too much.

A Trimble R7 GNSS and GPS receiver provided further reliability of position tracking. This receiver used satellites from both the Russian Global Navigation Satellite System (GLONASS) and the American GPS system to determine its position, making it more reliable.

In addition to this, users were asked to crouch down when staking out a target as already mentioned in the previous section. This kept the user's head more level as the user now looked at the target from a less steep angle.

6.1.2 Software

The software was divided into two main components: the tracking system and the visualisation system.

The Tracking System

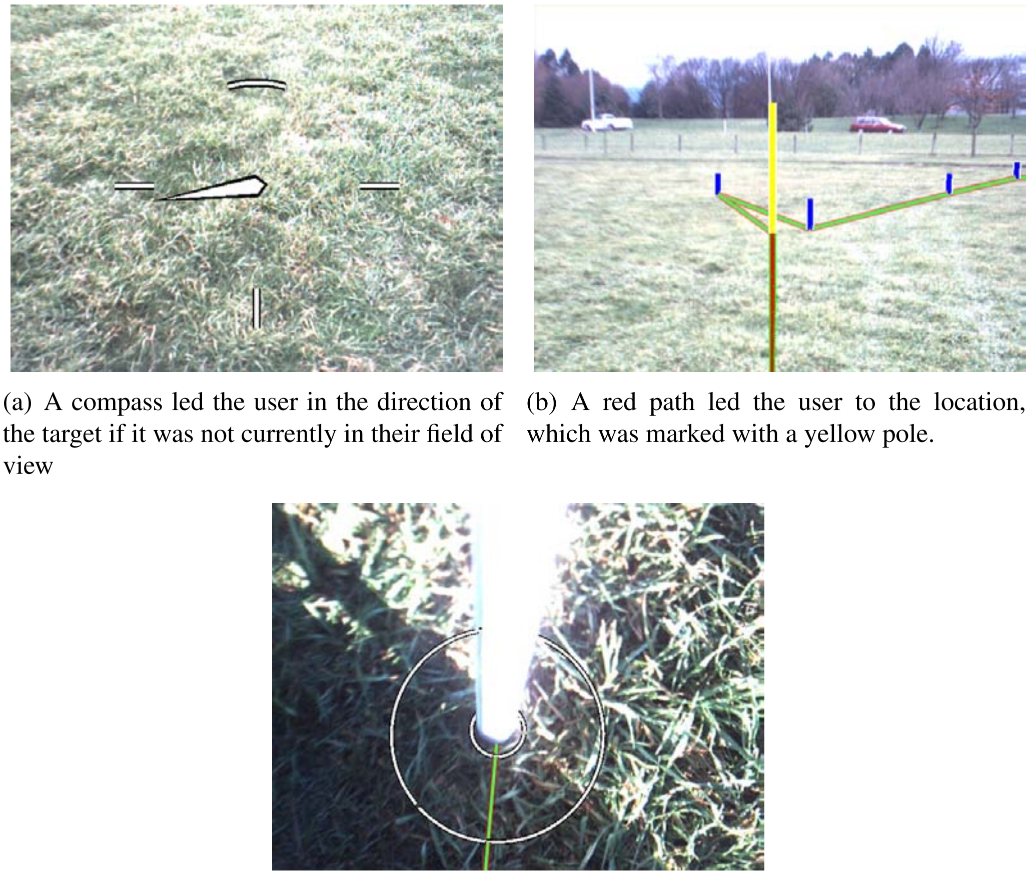
The tracking system managed the tracking sensors, and synchronized and fused sensor data. Sensors were modelled as objects that managed the different input streams coming from the video camera, the GPS receiver, and the orientation sensor.

Real-time sensor data was received and buffered in separate threads as it was being delivered by the physical sensors. The data was then synchronized to the video stream to provide a consistent AR illusion to the user. When the system received a video image, it estimated the time at which the image was taken. This estimate was based on the measured camera lag described in the next section. It then used this time to retrieve the closest matching sensor data from the location and orientation sensor buffers. These buffers were constantly filled by threads that monitored the sensors. The buffers stored the last n sensor measurements along with time stamps. This ensured that the virtual imagery matched the real world imagery. See table 6.1 for a comparison of the sensor sampling rates and lags.

Sensor data, including video images, could be recorded and then used by sensor objects that simulated the real sensors by delivering the recorded data instead of real-time data. This meant that the system could log a video stream together with location and orientation information. This functionality was useful for testing visualisations on desktop computers.

The Visualisation System

The orientation and location of the real camera were used to position and orientate the camera of the OpenGL environment. OpenGL was then used to draw virtual objects and user interface components on the camera image.



(a) A compass led the user in the direction of the target if it was not currently in their field of view

(b) A red path led the user to the location, which was marked with a yellow pole.

(c) A bull's eye helped the user to place the pole accurately on the point. A green line highlighted the path to the next target.

Figure 6.7: The user interface of the WOAR system.

The visualisation for the stakeout application (figure 6.7) was based on the research described in the previous chapters. When the current target location was not in the user's field of view, a circular compass (figure 6.7(a)) indicated the direction of the next target. The compass interface was based on the implementation described in section 3.5. The target locations were connected by a virtual path, with the path to the current target drawn in red (figure 6.7(b)). The current target location was marked with a vertical yellow line when the user was more than one metre away. When the user was closer than one metre from the current target location, the target was displayed as a set of concentric circles (figure 6.7(c)).

The inner circle had a radius of 5cm, and correct occlusion of the stake was provided through optical tracking. The algorithm that provided correct occlusion is described in section 4.10.

The system did not provide stereovision, as the experiment in chapter 4 did not show stereovision to have a significant effect on stakeout performance. The system did also not provide a transparent stakeout pole, as most implementation problems mentioned in chapter 5 have not been solved for this system.

6.1.3 System Lag

The average system lag that a user would perceive was 160ms, most of which was due to the camera lag of 155ms. System lag is the delay between an action in the real world and the display of that action in the HMD. This meant a user would only see their arm move 160ms after the fact.

A setup similar to the one described by Sielhorst et al. [2007] measured system lag. Sielhorst et al. [2007] described how time could be encoded as graphics displayed by an AR system. The system's camera would then capture these graphics. Lag was defined as the time difference between the time encoded in the captured image and the time when the captured image was displayed on screen.

There are two main differences between the approach described in this section and the approach by Sielhorst et al. [2007]: (1) this system does not require the camera to be calibrated, as the location of objects on screen is not relevant and (2) this system measures the minimum lag. The system described by Sielhorst et al. [2007] potentially captured more than one time encoded in the image due to the camera's long exposure time. Both these times were recorded for lag. This system is only concerned with the latest time encoded in the image, as this is the latest information that users are able to see.

With these changes, the presented system was simpler and more robust. With the camera pointed at the screen of the laptop, the following steps were repeated to collect data points:

(a) A black screen was displayed for a second to make sure that the camera did not capture stray signals.

(b) A white screen was displayed, and the time it was displayed with `glSwapBuffers()` was recorded.

(c) The captured image was tested for the white signal. For this, 4 pixels in the centre of the captured image were tested at negligible cost to speed.

(d) When the white signal was found in the captured image, a black screen was displayed again. The time difference between this black screen displayed with `glSwapBuffers()` and the time recorded in step (b) was recorded as system lag. The algorithm then continued with step (a) to collect more data points.

The lag measured using this method was constrained by the system's frame rate of 13.3fps, meaning that a new frame was displayed and measured every 75ms. Figure 6.8 illustrates this in a histogram view of the recorded data points. A high variance in system lag is visible, meaning that the system sometimes skipped two frames before presenting new real imagery to the user. This variance in lag was not noticed by any users of the system. However, it could have had an effect on performance, and more performing hardware is desirable.

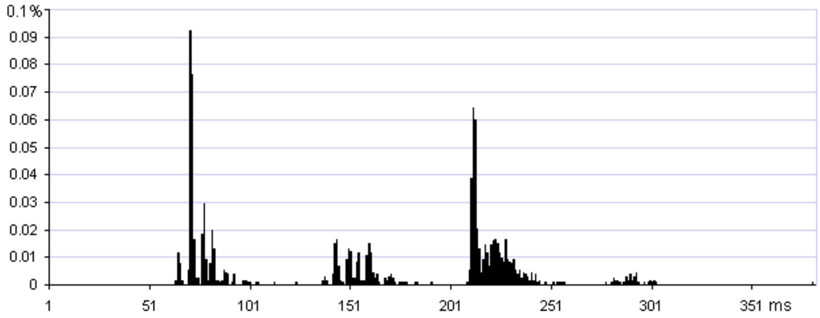


Figure 6.8: A histogram view of the recorded system lag in probability per milliseconds. The average system lag was 160ms

The performance counter of the Windows system was used to time the events. This timer provided a resolution of 3579545 counts per second on the system's laptop.

6.1.4 Conclusion

This WOAR system was built specifically for a stakeout application. As such, it attempted to provide the most accurate outdoor AR tracking to date. One of the main differences between this system and previous WOAR systems is that the GPS antenna was located in the helmet in order to track the location of the camera accurately. However, this introduced new problems of GPS reliability which required further optimisation of the helmet configuration.

The system was divided into sensor management and the visualisation system. The sensor management provided an abstract layer over the sensor devices. It also provided testing functionality through the playback of recorded sensor data.

The experiment described in the next section evaluated the efficiency of the system.

6.2 Experiment

The experiment described in this section evaluated the performance of the WOAR stakeout system with respect to an ideal stakeout system and the state-of-the-art commercial Trimble Survey Controller (TSC). It also compared the performance of the WOAR components and their effect on the user performance with the system.

WOAR systems will only be a commercial success if they can perform at least as well as the current conventional systems that they are trying to replace. WOAR systems have mainly been proofs of concept and have been used to show what would be possible if the appropriate hardware was available today. Previews literature does not describe a formal comparative study for WOAR systems. This experiment formally assessed the objective performance of a WOAR system, and compared the performance to that of a conventional version of the application. This had not been done before.

The experiment had two main goals:

(1) To find the difference in performance between a state-of-the-art WOAR system and a state-of-the-art commercial system for a specific application. This was a benchmark that indicated how well the WOAR system performed.

(2) To identify strengths and weaknesses of the components of the WOAR system to aid in improving its performance.

In the experiment, participants used the different systems to follow a predefined route in the real world, and stake out a set of points. Speed and accuracy of the navigation and stakeout tasks were used as a performance measure.

6.2.1 Design

The experiment followed a within subjects design with the factor ”stakeout system” with the following four conditions:

- *Control*, where participants staked out real markers without help from any electronic system
- *HMD*, where participants staked out real markers while wearing an HMD
- *WOAR*, where participants staked out virtual markers with the WOAR system
- *TSC*, where participants used a state-of-the-art conventional stakeout system

The dependent measures were task completion time and horizontal accuracy of stake placement. Each task consisted of two sub tasks: navigation and pole placement. Section 6.2.4 describes the measurement details of these two phases. Participants also filled out a subjective questionnaire after using each system and at the end of the experiment.

6.2.2 Experiment Conditions

The *control* condition simulated a perfect stakeout system: real target markers and connecting lines were placed on the ground (see figure 6.11(a)), and the participants did not wear HMDs. This simulated perfect tracking and registration as well as an HMD with perfect resolution, field of view, and depth perception. Off-the-shelf hardware is currently not able to provide this quality and probably never will be. This setup is the best possible stakeout aid short of using objects that physically guide the pole into place. Thus, this condition served as a baseline for the performance of both the TSC and the WOAR system.

The *HMD* condition used the same setup as the control with the participants wearing the backpack and HMD of the WOAR system. The HMD showed a video view of the real world and thus introduced delay, removed stereovision, and decreased the refresh rate, field of view, resolution, and colour contrast. The HMD condition was used to quantify the reduction in performance caused by the HMD independently of the tracking system. The tracking system further introduced positional and orientation error, jitter, and delay.

The *WOAR* condition let the participants stake out virtual targets using the WOAR application. It guided the participants to the next point by displaying a virtual compass if the next target was not in the participants' current field of view as seen in figure 6.7(a). It also drew a red path from the participants' location to the target. When the participants were more than one metre from the target, the target was marked by a two metre tall virtual pole. Figure 6.7(b) shows the red path leading up to the yellow pole. Subsequent target locations were marked with short blue poles and connected with green path segments. When the participants were within one metre of the point, the visualisation changed to a bull's eye view that marked the point and provided a scale for estimating stake placement error (figure 6.7(c)). The inner circle had a radius of 5cm and the outer circle had a radius of 10cm. The switch between the virtual pole and the bulls-eye view was made so that the location was both visible from far away, and the real pole would not have to compete with a virtual pole during stakeout. During the iterative development of the interface, we found that the visualisation was less confusing when the virtual pole was omitted during stakeout. The vision based tracking algorithm described in section 4.10 provided correct occlusion of the real pole. The WOAR condition did not show any menu options to the participants and did not require them to interact with the system.

In the *TSC* condition, the Trimble Survey Controller guided the participants to the next point using a compass view as shown in figure 6.10(a). At a distance of 1.5m from the location, the visualisation changed automatically to the bull's eye view as seen in figure 6.10(b). The bull's eye view was a planar view from the top with the location of the pole marked by a cross, and the location of the point marked by a bull's eye. During placement of the pole, the participants had to hold the pole absolutely vertical. A little water level integrated into the pole had to be observed for this. The participants did not use any of the buttons or menus of the TSC. Section 2.5 presents an overview of the TSC user interface and includes a discussion of its usability issues.

6.2.3 Procedure

The participants used each of the four conditions to follow pre-defined routes and stake out five locations. To counter learning effects, the order of the conditions as well as the allocation of the routes to the conditions were randomized. Participants completed several training tasks with each system until they were confident enough to use them for stakeout. On average, they completed none for the control, and three each for the other conditions. Paths between stakeout targets were straight and 8 metres long. Each route had the same type of turns in randomized order: 90° left and right turns, a 180° turn and a 0° turn. See figure 6.9 for an example path.

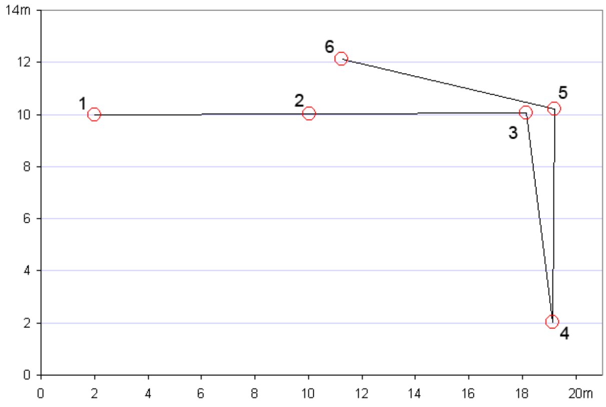


Figure 6.9: A sample path.

A subtask consisted of walking to a target and then placing a stake on that location. The participants were instructed to complete their tasks as quickly and accurately as possible. Participants were asked to stake out targets with an accuracy of at least 5cm. After placing the stake on a location, the participants waited briefly until the experimenter gave them a signal to stake out the next target. This pause was used in the WOAR and TSC conditions to place a marker on the staked out location and to select the next target. In the WOAR condition, a short video of the stake was taken through the camera of the HMD. This video was used after the experiment to measure the “WOAR optical” error as shown in figure 6.13.

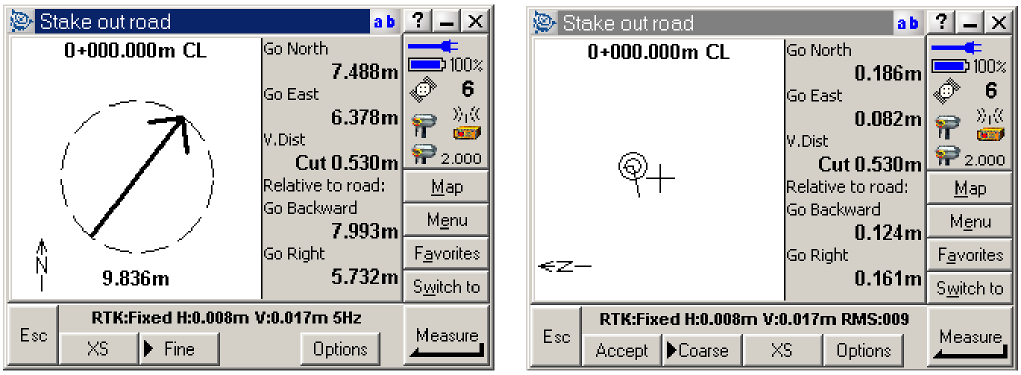
At the end of each condition, as well as at the end of the experiment, participants filled out questionnaires asking them for a subjective evaluation.

6.2.4 *Apparatus*

Participants used a light wooden stake, as seen in figure 6.12 to mark locations in the real world. The stake was 58cm long and had a narrow metal tip. In the TSC condition, participants marked the location with the surveying pole instead.

In the *Control* condition, paper markers were connected by red and white plastic ribbons (figure 6.11).

In the *TSC* condition, participants carried a TSC system as seen in figure 2.15(a). The participants used the directions of the TSC to complete their stakeout tasks. The TSC version used was 11.40 (2007) that ran on the TSCe platform.



Stake out road

0+000.000m CL

Go North 7.488m

Go East 6.378m

V.Dist Cut 0.530m

Relative to road:

Go Backward 7.993m

Go Right 5.732m

9.836m

RTK:Fixed H:0.008m V:0.017m 5Hz

(a) A compass directed the user to a location

Stake out road

0+000.000m CL

Go North 0.186m

Go East 0.082m

V.Dist Cut 0.530m

Relative to road:

Go Backward 0.124m

Go Right 0.161m

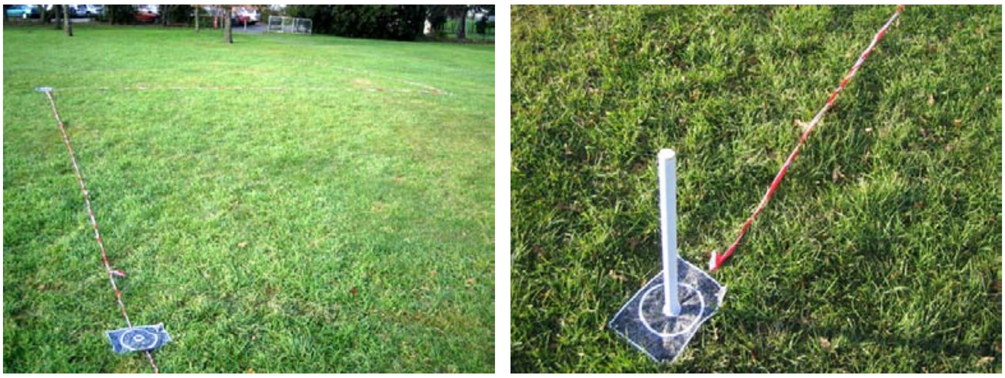
RTK:Fixed H:0.008m V:0.017m RMS:009

(b) A bull’s eye view helped the user to place the pole accurately on the point.

Figure 6.10: The user interface of the TSC.

In the *HMD* and *WOAR* conditions, participants wore the WOAR system shown in figure 6.2.

The GPS accuracy depended on the quality of the current satellite configuration. Experiment sessions were scheduled so that they would coincide with optimal satellite configurations. This meant that some experiment sessions took place



(a) A real path connects a set of real targets (b) The stake placed on a paper target

Figure 6.11: The real path and markers

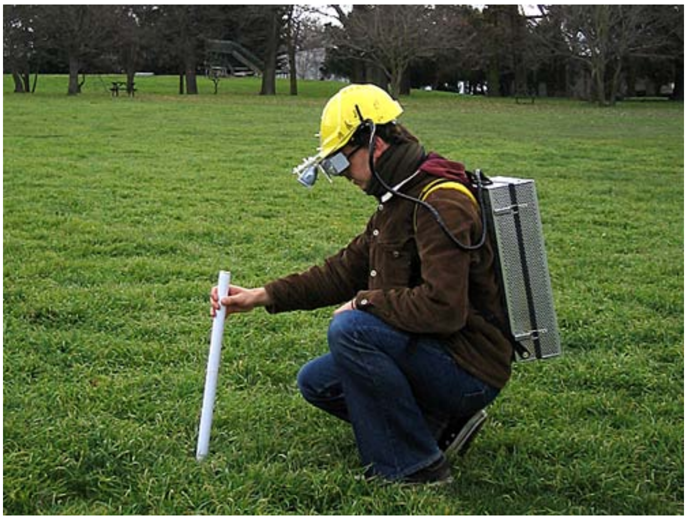


Figure 6.12: Participants had to crouch to stake out a location

in the morning while others took place in the afternoon. In addition, participants were told not to look up too much as this would reduce GPS quality by tilting the GPS antenna. They were told that if they could see the horizon in the HMD, they were looking up too high.

In the HMD and WOAR conditions, participants had to crouch next to the target to accurately place the pole (figure 6.12). This was due to the camera quality and the accuracy of the orientation sensor, and also kept the GPS more level.

6.2.5 Measurements

In all four conditions, time was measured manually by the experimenter. This provided comparability over all conditions, including the control, which did not involve the use of equipment capable of recording any measurements.

Four different types of events were timed manually by the experimenter for each staked out location:

A: The experimenter tells the participant to start the task.

B: The participant stops walking for the first time.

C: The participant places the stake on the ground.

D: The participant says "done", indicating that the stake is placed in the right location.

The duration of marking the staked out location and selecting the next target in the WOAR and TSC conditions was not included in the measured time. This process would rely heavily on the systems' user interfaces such as menus which this study is not investigating.

The following time intervals are defined for the described actions:

- *Task completion time* was D-A.
- *Walking* took place between A and B.
- *Staking* took place between B and D.
- *Verifying* that the stake is in the correct location took place between the last recorded C and D.

Type C events were only logged from participant six on, as it only became clear during the experiment that this would be valuable information.

In all four conditions, stake placement was measured manually by the experimenter. The accuracy of the control and HMD conditions was measured with a ruler using the holes punched into the paper markers at the accuracy of 1mm.

The accuracy of the WOAR and TSC systems was measured with the TSC at an accuracy of 1.1cm. A video with loss-less HuffYUV compression [1] was recorded through the camera of the HMD for each stake placed with the WOAR system, to measure the accuracy of the stake placement as seen by the participant. This measurement was taken at a desk after the completion of the experiment. This showed the accuracy of the WOAR system independent of calibration errors and tracker noise (see figure 6.13 for a sample measurement screen capture).

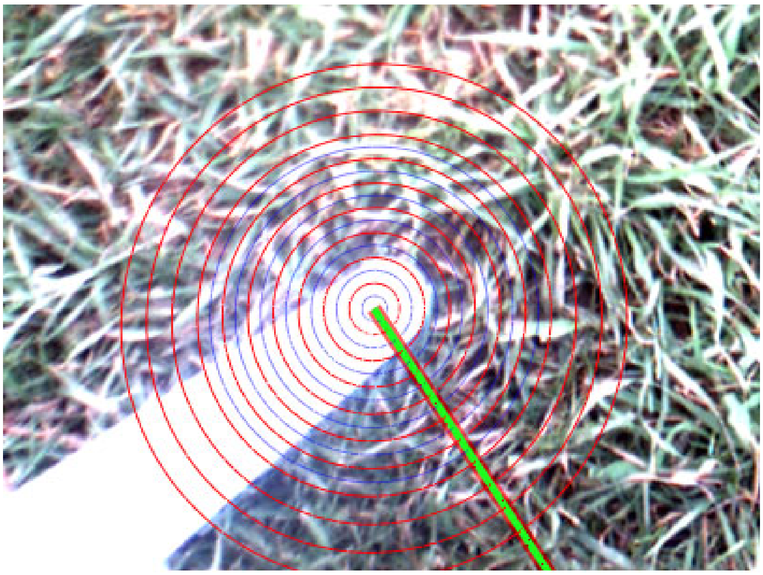


Figure 6.13: A screen capture illustrating the method to measure the optical WOAR placement error. Red circles indicate centimetre steps from the centre, blue circles 0.5 centimetre steps.

In the WOAR condition, GPS position and orientation were continuously logged in order to analyse the participants' movement. Although it would have been desirable, the system did not continuously log the video stream from the WOAR and HMD conditions due to the load it would have placed on the system.

In the TSC condition, the GPS position was not continuously logged, as the TSC did not provide an easily accessible logging function that did not visibly slow down the interface. Instead, three selected participants who were strong enough for the heavy backpack to not have too much of an effect on their performance

[1] http://neuron2.net/www.math.berkeley.edu/benrg/huffyuv.html

staked out an extra five targets with the TSC while wearing the WOAR backpack without the helmet. The backpack's laptop logged the GPS locations from the TSC's GPS receiver. This was done under the same conditions as the other stakeout tasks: participants were asked to stake out targets as quickly and accurately as possible. They were timed, and markers were placed in the ground so that their stakeout accuracy could be measured. However, all data from these trials except for the GPS position trace was discarded.

6.2.6 Participants

15 participants aged from 19 to 29 years with an average age of 22.3 years completed the experiment. Most of them were students at the University of Canterbury. Five participants were female. None of them had any experience with stakeout before, although two of them had briefly used augmented reality HMDs before.

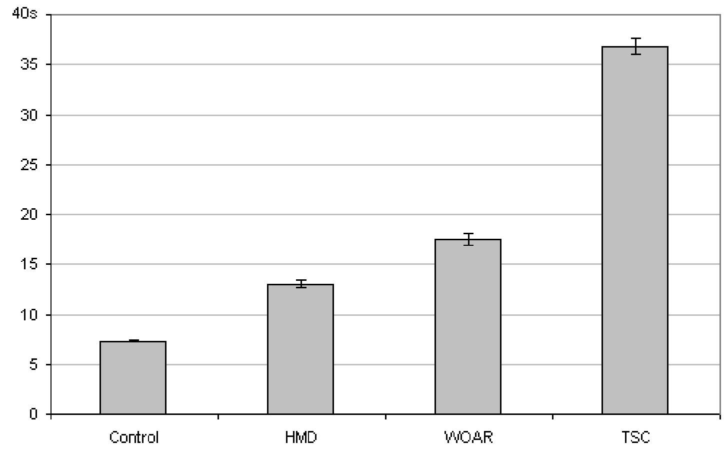
6.3 Results of the Experiment

The participants were able to perform well with all systems. They reached the same accuracy for both the WOAR system and the TSC, with the WOAR system being twice as fast as the TSC system.

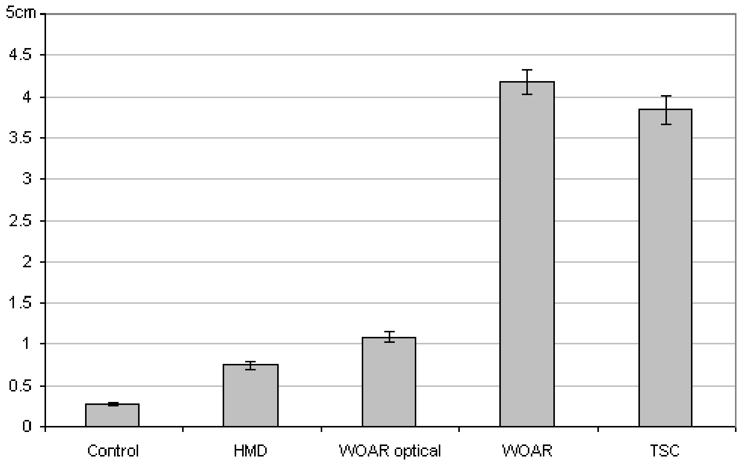
6.3.1 Dependent Measurements

The mean times for task completion (see table 6.2 and figure 6.14(a)) were significantly different (ANOVA, $F(3,56) = 143.00, p < 0.001$). As expected, the control condition was the fastest and the TSC condition was the slowest. There were large differences in task completion times, with the WOAR condition taking more than twice as long as the control, and the TSC taking more than twice as long as the WOAR condition. A post-hoc comparison with a Tukey Test and an HSD of 4.05 seconds revealed that the task completion times of all four conditions were significantly different from each other.

The means for horizontal placement error (see table 6.3 and figure 6.14(b)) were significantly different (ANOVA, $F(4,69) = 73.02, p < 0.001$). As expected, the control condition performed best. Both the WOAR and the TSC condition



(a) Task completion time.



(b) Placement error.

Figure 6.14: The results for the dependent measures.

performed similarly and much less accurately than the other three conditions. A post-hoc comparison with the Tukey Test and an HSD of 8.5mm did not show a significant difference between the means for WOAR and TSC, while both WOAR and TSC were significantly less accurate than the other three conditions. There was also no significant difference between the control, the HMD and the WOAR optical results. The WOAR and the TSC measurements also had a larger variance than the other conditions. The "WOAR optical" condition is the measurement

Table 6.2: Task completion time in seconds

Interface	Task Completion Time	Standard Deviation
Control	7.3	0.7
HMD	13.0	2.9
WOAR	17.5	4.5
TSC	36.8	6.3

Table 6.3: Placement error in centimetres

Interface	Placement Error	Standard Deviation
Control	0.3	0.2
HMD	0.7	0.3
WOAR Optical	1.1	0.4
WOAR	4.2	1.1
TSC	3.8	1.4

taken from screenshots (see figure 6.13). The WOAR optical values were not considered in the ANOVA.

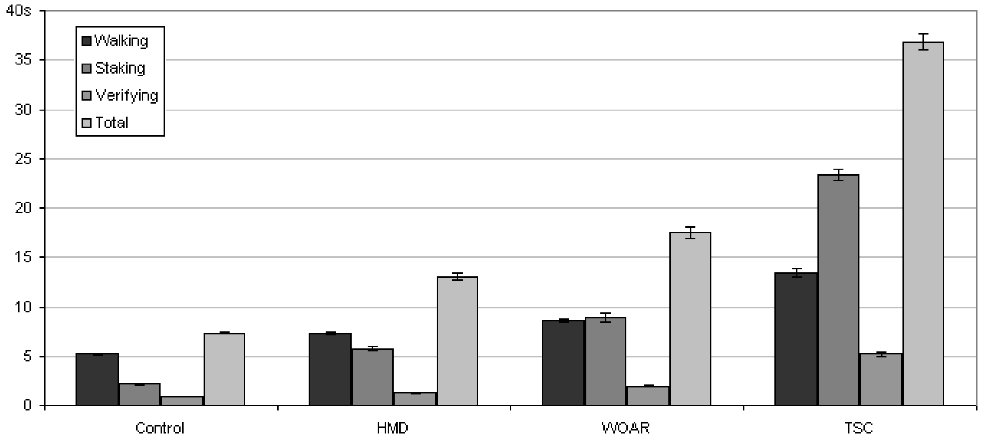


Figure 6.15: A break down of the task completion timeinto walking, staking and verifying the stake location

The break down of the task completion time into walking, staking and verifying is shown in figure 6.15. The longer the task completion time was, the longer all other phases took as well. There were large differences in time for placing the stake, as it took 21 seconds longer in the TSC condition than in the control. Staking in the control condition only took a fraction of the task completion time. For the WOAR condition, it took about the same time as walking, and it took up the majority of the time for the TSC condition. Placing the stake in the TSC condition took even longer than the total task completion time for the WOAR condition.

A similarly dramatic increase can be seen in the time it took to verify a location with the TSC. After placing the stake on the ground with the TSC, participants took on average 5.2 seconds to verify that the location was correct, while it only took 1.9 seconds with the WOAR system and 0.9 seconds in the control condition. It should be noted that the time for placement verification was only taken for the last ten participants.

6.3.2 Movement Patterns

A good illustration of how the participants used the WOAR system and the TSC system to navigate to the next point can be seen in figure 6.16 for the WOAR system and figure 6.17 for the TSC system. The graphs plot continuously logged GPS positions for one of the participants staking out a set of points. The target locations are marked with circles. Both graphs show the same stakeout route. In figure 6.16, the path does not directly touch many of the targets, as the GPS logged the helmet location of the participant standing next to the targets, whereas figure 6.17 shows the location of the pole directly over the targets. It is obvious that the participant with the WOAR system always knew in which direction the next target was, while this was not the case with the TSC system. The sample participant swerved two metres to the side as can be seen in figure 6.17 between targets 1 and 2, overshot as between targets 3 and 4, or initially started walking in the wrong direction as can be seen between targets 5 and 6. The path between targets 4 and 5 shows an example of good navigation between points. During navigation, participants usually continued holding the TSC upright but did not pay attention to the water level. This means that some of the diversion from the ideal path can be attributed to holding the TSC at an angle. Based on observations from the experiment, this would only account for less than 30cm error, much less than the swerving shown in the graph.

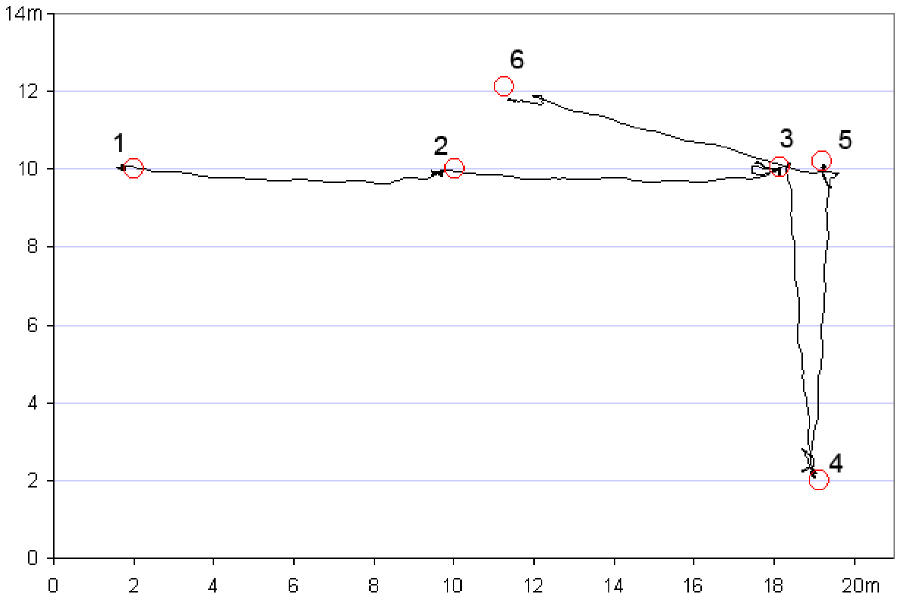


Figure 6.16: A representative GPS path traced with the WOAR system during a stakeout session with the locations of the stakeout targets marked.

6.3.3 *Subjective Measures*

Participants ranked the conditions by preference and accuracy, with 1 as the lowest and 5 as the highest rank (see figure 6.18). As expected, participants preferred the control and thought it was the most efficient interface. The WOAR condition was ranked second for both preference and efficiency. There was a significant difference between the levels for preference (Friedman test, $\chi_r^2 = 11.24, df = 3, N = 15, p = 0.01$) and efficiency (Friedman test, $\chi_r^2 = 27.64, df = 3, N = 15, p < 0.001$).

Participants were also asked to rate the conditions on a Likert scale from 1 (“low”) to 7 (“high”) based on several criteria. See table 6.4 for the results. There was a significant difference between the conditions for all of the questions asked. The WOAR condition and the HMD condition were always rated similarly.

In their comments, the participants said that the *control* condition was “simple, easy and fast”, “easy to use”, and that it was “easy to find [the target] and place” the stake on it. They said it was the “most straightforward” condition of the experiment. One participant also noticed that “it was harder than [he] thought to get

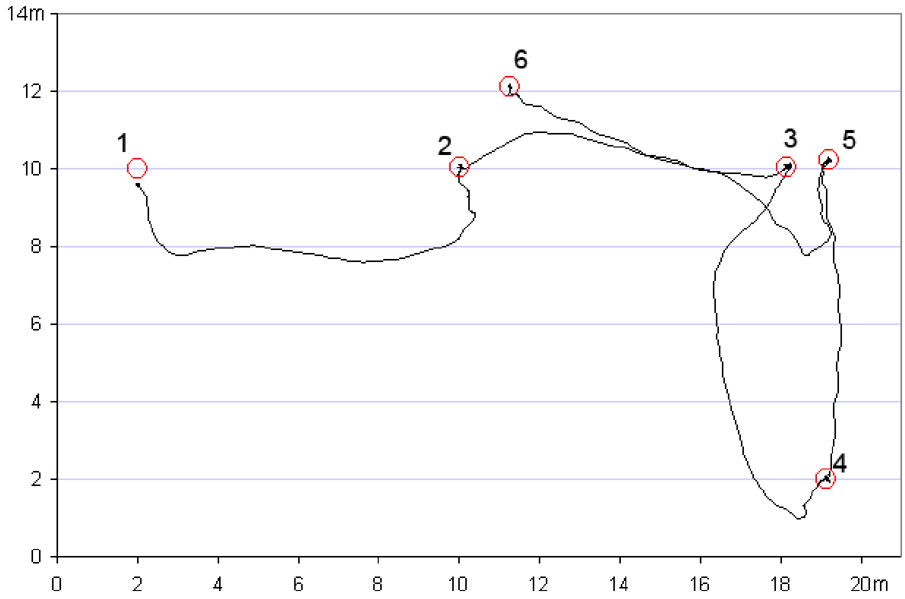
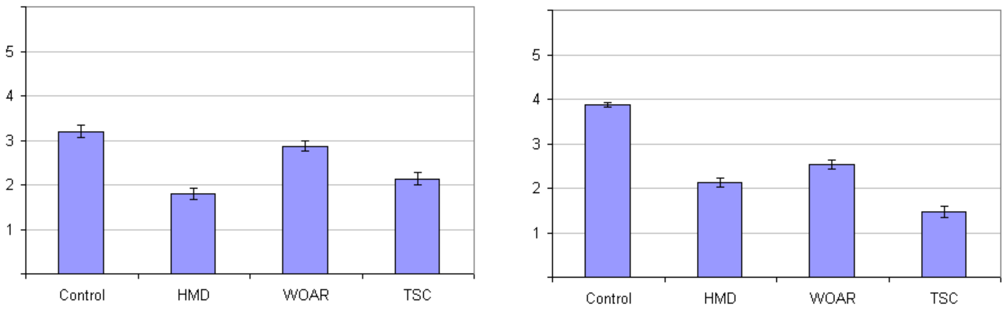


Figure 6.17: A representative GPS path traced with the TSC system after a stakeout session with the locations of the stakeout targets marked.



(a) Subjective ranking by efficiency from 1 ("low") to 7 ("high").

(b) Subjective ranking by preference from 1 ("low") to 7 ("high").

Figure 6.18: Subjective ranking of the interfaces

the centre of the bull's eye", indicating that the participants did use due care when placing the stake on the real target, as opposed to the experiment from chapter 4.

The *HMD* condition also received favourable comments, but participants criticised the weight of the backpack, the uncomfortable helmet and the quality of the

Table 6.4: Subjective ratings of the conditions with standard deviations in parentheses

Measure	Control	HMD	WOAR	TSC
Mental demand	1.3 (0.2)	2.3 (1.1)	2.4 (2.5)	3.7 (2.2)
Friedman	$\chi_r^2 = 24.13, df = 3, N = 15, p < 0.0001$			
Physical demand	1.3 (0.2)	2.7 (1.2)	3.1 (1.4)	3.2 (2.2)
Friedman	$\chi_r^2 = 25.91, df = 3, N = 15, p < 0.0001$			
Performance	6.5 (0.7)	4.5 (2.1)	4.4 (2.1)	4.1 (1.6)
Friedman	$\chi_r^2 = 23.93, df = 3, N = 15, p < 0.0001$			
Effort	1.3 (0.4)	3.5 (1.7)	3.5 (2.1)	4.6 (1.8)
Friedman	$\chi_r^2 = 31.96, df = 3, N = 14, p < 0.0001$			
Frustration	1.3 (0.2)	3.2 (2.0)	2.9 (3.2)	4.6 (1.7)
Friedman	$\chi_r^2 = 30.85, df = 3, N = 15, p < 0.0001$			
Ease of turning	7 (0)	5.8 (1.3)	5.8 (1.7)	2 (2.3)
Friedman	$\chi_r^2 = 33.62, df = 3, N = 15, p < 0.0001$			
Ease of Walking	7 (0)	5.4 (2.0)	5.9 (1.6)	4.1 (2.5)
Friedman	$\chi_r^2 = 28.7, df = 3, N = 14, p < 0.0001$			
Ease of Staking	6.3 (0.9)	3.8 (2.0)	3.4 (3.1)	2.9 (1.4)
Friedman	$\chi_r^2 = 23.16, df = 3, N = 15, p < 0.0001$			
Ease of use	6.8 (0.2)	4.4 (2.0)	4.5 (2.1)	3.1 (1.0)
Friedman	$\chi_r^2 = 30.96, df = 3, N = 15, p < 0.0001$			

video see-through HMD. They said the backpack was “heavy” and the “helmet was uncomfortable”. The heavy HMD caused “compression of the nostrils” and was “uncomfortable”. They mentioned the “limited field of view and visibility”, said that the “limited field of view could be dangerous”, that the “low frame rate made accurate placing of the stake difficult”, and that the “missing depth perception and the camera lag made it difficult to get the stake exactly on target”. Several participants said that “hand eye coordination was awkward because [their] eyes were no longer lined up with reality” and that they “kept walking over the target because the distance felt a bit off”. In general, the quality of the video see-through HMD was identified as the main factor of the problems with this condition as it “could be so much better and less cumbersome with a better goggle with a better field of view.”

However, participants generally said that stakeout with the HMD condition was “easy to use”, “quick to learn and very similar to reality” and that it was “surprisingly similar in the ease of use to the control condition” although it was “not as easy as the control”. They liked “being able to see where to go in a simple way” and that they “could see where all the points were in relation to [them]”.

With the *WOAR* condition, participants identified tracking inaccuracies and the jitter of the virtual objects as the main problems apart from the aforementioned criticism of the HMD in general: “if the WOAR system behaved the way the HMD condition did, I would want one” and “if the WOAR would be as solid as the HMD condition, it would be perfect”. They said it was “intuitive”, “fast and easy”, and that “the interface was very simple and intuitive with no extra clutter” and that they “felt [that they had] very quickly got the hang of it”. Navigation with the WOAR systems was “easy”, and the participants liked that “it showed [them] which way to go” and that they had “very clear directions and lines to follow”. It “seemed easier than the TSC, and the direction to walk to the next target was very obvious”, “it was quite clear where the next target was direction wise” and it was “easy to find the next point”.

However, some reported that they had problems with overshooting and that “it was a bit difficult to tell how far away the yellow pole was”, and that there was a “sudden change from the virtual pole to the bull’s eye. [They] usually had to stop dead suddenly. Some transition or indication that [one is] close would be useful.” The participants identified jitter in the tracking system as making it hard to place the stake accurately. Some said that “the WOAR and the HMD condition [were] very close, [but] the jumping of the marker made [them] rate WOAR worse”. It was “slightly difficult to place the stake while it was very easy to find the general area”, as “the target was slightly shaky all the time” and the virtual objects were “a bit jumpy jerky”. Some reported that the “target seemed very jumpy and moved about a lot” and that “the point to place the stake moved around, so it was a bit harder to get it in the right place,” while others claimed it was not much of a problem for them as they “knew exactly where the stake was supposed to be, even if it was slightly difficult to place the stake” and that the “display [was] very clear with respect to where to put stake”. In addition to this, similar comments on the quality of the video see-through HMD as the ones already reported were made. Interestingly, one participant said that it would be easier to see the target marker if the pole was made transparent, as it obscured the marker.

The opinion on the *TSC* was divided. Some participants liked that it was “very clear and simple", "simple in concept", and "fairly straightforward to use", while others said that it "was extremely annoying and frustrating", that it was "hard to use", that they "cussed a great deal using the system" and that it was "more frustrating" than the other conditions. As compared to the HMD and WOAR conditions, participants liked that it "looked less weird", "did not require an uncomfortable helmet", was "light and easy to carry", "less heavy than the backpack", and it did not require them to crouch. On the other side, they acknowledged that their "hands [were] not free". When navigating towards a point, it was "hard to tell the direction to begin walking" as it "took too long for the right direction to show when turning", "often sent [them] in the wrong direction initially", and that "the direction to the next point was never right after placing the stake, even though [they] didn't rotate the pole". As for staking out a point, participants said that the TSC "seemed to be more accurate", that "it gave a feeling of high accuracy", and that it "might be a bit slower but is more accurate". However, more participants were unhappy with the targeting phase of the TSC as it was "hard to use and very slow to place [the pole] accurately". It "had [them] going in circles", it was "difficult to position [themselves] when [they] were close to the target" and that "the RTK dance was exceedingly frustrating". This was not only due to directional problems, as "the lag in updating was also frustrating", and it was "difficult to position due to low update rate and levelling", that they "had to monitor the water level and screen at the same time" and that the "water level annoyed [them]". Overall, "it took more time and patience to get things right". As for the user interface, "the target screen seemed to arrive more suddenly than expected, so [they] could overshoot it", and "the system gave little indication of when the resolution was higher and required smaller correction movements".

6.3.4 The TSC Expert

In addition to the 15 novice participants, a Trimble employee, who was an expert with the TSC system, performed the same tasks. She had used the TSC thoroughly in product testing and was not affiliated with the WOAR project. Her performance is not included in the general results of the experiment and mainly serves to assess how realistic the participants' performance with the TSC was.

Table 6.5 shows the results of the expert's performance with all four condi-

Table 6.5: The TSC expert's performance

Interface	Placement Error in cm	Task Completion Time in s
Control	0.1	6.4
HMD	0.7	11.3
WOAR Optical	1.5	
WOAR	3.2	15.9
TSC	3.2	32.8

tions. She performed more accurately and faster in all conditions than the participants, except for the HMD condition, in which she reached the same accuracy as the participants. Her performance was similar to that of the participants, with the WOAR and the TSC condition having the same accuracy, and the WOAR system being twice as fast as the TSC. Figure 6.19 shows a GPS path traced by the expert using the TSC system on the same route as in figure 6.17. It is clear that the expert's movement graph shows a more efficient movement than the one in figure 6.17. The expert said that she did not actually use the compass visualisation on the TSC screen. Instead, she used the numeric "Go North" and "Go East" values to the right hand side of the compass as seen in figure 6.10(a). All experiment sessions were run at the Trimble test site, so she knew well where North was. In her experience, she would quickly learn the North direction for a new site as well, and then rely on that knowledge for stakeout. The test courses were aligned with North and East, making it easier for her to quickly navigate to targets. Her TSC walking phase of 9.9 seconds was shorter than for the other participants who took 13.4 seconds. Her TSC staking phase of 22.9 seconds was close to the participants' average of 23.4 seconds, while her verification only took 4.2 seconds as compared to an average of 5.2 seconds from the other participants.

The expert ranked the interfaces as Control, HMD, WOAR, and TSC (from best to worst) for both preference and efficiency. While she believed that the WOAR equipment "would get heavy after a while", she also said that she would have been more accurate with more practice and that "the system would be easier once one got used to it." With the WOAR system, "it was difficult to judge distances" and "it was difficult to walk up to the targets with speed". With the TSC system, "the Northing and Easting values are easy to use and [she could] get

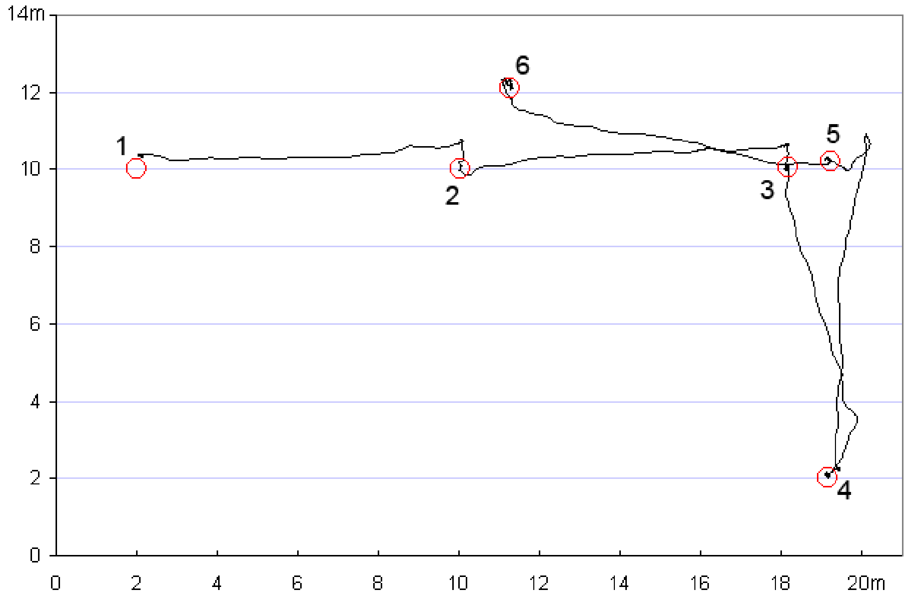


Figure 6.19: A GPS path traced by the Trimble expert using the TSC.

to the approximate location O.K.", while "the arrow/circle displays are disorienting" and, as mentioned before, she did not use them. She noted that "the disjoint between the small screen and what [one] sees in the real world is difficult" and that the water bubble "made the system harder to use", while "the [AR] system removed this complexity".

6.4 Analysis of the Results

The results suggest two main sources of performance loss for the WOAR system:

(1) the video see-through HMD and

(2) the tracking accuracy and calibration.

Section 6.4.1 investigates this in detail.

6.4.1 WOAR Performance Loss By Components

Participants performed slower and less accurately with the HMD than with the control, although there was no significant difference in placement error. In line with the participants' comments, the study identified the following problems with the camera and the HMD:

- system lag
- low frame rate
- loss of depth perception
- smaller field of view
- distortion in the field of view
- low contrast
- low resolution

From the participants' perspective, the main problem with the WOAR system, as compared to the HMD condition, was the jitter of the virtual objects. While jitter is part of the system's tracking error, we will treat it separately in this section, because the participants found that it made stakeout more difficult. Perceived jitter added to the participants' cognitive load, and it forced them to make a decision as to where the stake had to be placed. In addition to this, tracking errors and calibration errors further reduced accuracy. The accuracy measure *WOAR optical* took this into account and measured how accurately the participants would have been able to stakeout a location if there were no tracking or calibration errors.

Using these measurements, figure 6.20 shows the placement error of the WOAR condition split up into separate components. This provides an insight into how much accuracy was lost due to the different components of the system. The "Control" bar shows the accuracy of the control condition. This is the accuracy loss introduced by the task setup itself. The "HMD" value is the difference between the accuracy of the control condition and the HMD condition. It shows accuracy loss due to factors such as lag, frame rate, field of view, and resolution. "Jitter" is the difference between the HMD condition and the WOAR optical measurements. This is the accuracy loss attributed to the jitter of the virtual target. "GPS" is the RTK GPS error and "InertiaCube" is the displacement error at a distance of one

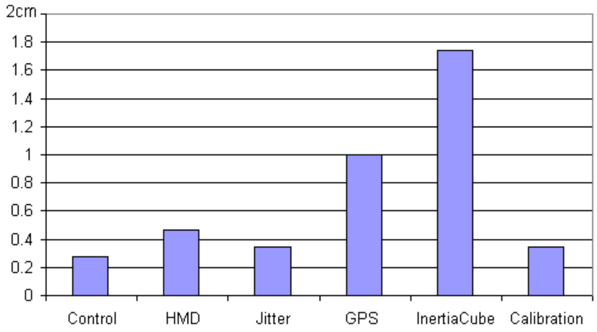


Figure 6.20: Accuracy loss in cm attributed to components of the WOAR system.

metre based on the InertiaCube accuracy of 1°. Both these values are based on the manufacturer's specifications from table 6.1. "Calibration" is the difference of the measured WOAR accuracy and all the other values in the figure. This measures the accuracy of the calibration described in section 6.1.1.

These comparisons have to be made carefully, as there was no statistical difference between the displacement error for the Control, HMD, and WOAR optical conditions. In addition to this, each of the three measures for HMD, WOAR optical and WOAR have been made using different methods with different accuracy levels, and the RTK and InertiaCube errors are based on factory specifications. However, these numbers give valuable insights into which components of the WOAR system caused the most accuracy loss: the tracking sensors.

6.4.2 Navigation

The combination of compass, path, and waypoint was an efficient navigation aid. Showing a virtual path meant that participants already knew in which direction they had to turn next before they completed staking out the current target. There was not much explicit subjective feedback about the compass visualisation, but using the compass worked well for the participants when they had to find the beginning of the path before the experiment session started. When following a path, participants navigated quickly and without error.

However, problems occurred when participants overshot a target and suddenly only saw the compass with no target marker and no path. The application tried to provide for this by drawing additional target circles with a radius of 1 and 0.5 metre on the ground so that the target was easier to find with a small field of view. An example of this can be seen in figure 6.6. However, there were a few cases where participants briefly became confused. They quickly learned that stepping back one or two steps would bring the target back in view. On one occasion, a participant stood a few centimetres in front of the target, looking forward. This meant that they could not see the target, and the compass was still directing them forward. While this did not happen often, and will most likely decrease with more training, a future implementation will have to make directions obvious in these special cases.

6.4.3 Depth Cues

Several participants complained about the lack of depth cues with the HMD and WOAR conditions. There were two situations where participants noticed the lack of depth cues: walking to a target, and placing the stake on a target. While participants where less concerned about the absence of depth cues for actually placing the stake on the ground, several of them had problems when walking up to a target.

When walking up to a target, some participants overshot because they reached the target sooner than they had expected. Some of them said that they would have liked the visualisation turn from the yellow pole into the bull's eye sooner, but this might not help them more than adding depth cues. While the participants did indeed take the change in visualisation as a signal to stop walking, such a signal would have to be given depending on the user's current walking speed to prevent overshooting. However, this might mean that participants stop at inconsistent distances from the target location, making such an interface confusing. A gradual change in visualisation as the user approaches the target might be used as an efficient depth cue. This will have to be explored in the future.

6.4.4 A Disadvantage of the TSC

It has to be noted that the TSC condition was the only condition that did not result in a stake being placed at the target locations. While the other three conditions required the participant to place a stake at the target locations, TSC tasks ended with the surveying pole on the target location. Using the WOAR system, users would be able to place stakes directly on target locations, while the TSC system requires users to first place the pole on a target location and then to place a stake on that same location. This means that the task completion time for the TSC should actually be even longer than the results presented in this chapter. This last step was omitted in the experiment so that participants could concentrate on the task of staking out a target location. However, it should be kept in mind that the TSC would perform even slower in real life due to this.

6.5 Conclusion

This study showed that the WOAR stakeout system was able to compete with a conventional stakeout system, and it was significantly faster than the conventional system. At an accuracy level of 4cm, stakeout with the WOAR system was twice as fast as with the TSC. The participants found the conventional TSC stakeout system frustrating and hard to learn, while they generally found the WOAR system simple and intuitive to use.

The weight of the system, the quality of the video see-through HMD, the video lag, and the jitter of the virtual objects were named the main problems with the system. A production model of the WOAR system would weigh substantially less, with a lightweight mobile computer instead of a laptop, and a lightweight backpack instead of a backpack that is created for research purposes. The laptop was already several years old at the time of this experiment, and modern computers will cause less video lag.

Figure 6.20 indicates that the main part of the placement error was probably due to tracking inaccuracies. Adding vision based tracking may further reduce jitter and increase the position and angular tracking accuracy. Vision based tracking may also allow the WOAR application to accurately measure the location of the pole, a feature that is not implemented for the current WOAR system.

One of the greatest shortcomings of the WOAR system, the loss of satellite fixes, had been avoided for the experiment: since tilting the head resulted in reduced GPS satellite reception and the potential loss of RTK grade GPS, experiment sessions were scheduled so that they coincided with good satellite constellations. During all WOAR condition tasks, there were at least five satellites in the sky with an elevation of at least 35°, meaning the users could tilt the GPS antenna 35° up or down without losing RTK grade GPS. This greatly reduced the available time slots for the experiment sessions to two per day. Such a limitation is prohibitive to a commercial implementation of the system, and a more reliable position tracking method will have to be implemented. For example, the GPS antenna could be mounted on the backpack where it would stay relatively level, and the helmet could be independently tracked by optical means. However, this would introduce another source of tracking error. If an optical tracking system such as the one described in the previous paragraph worked at a sufficient accuracy and reliability, then it could be used for dead reckoning while RTK grade GPS is lost.

Requiring the user to crouch while placing the stake on the ground meant that the stakeout interaction now took place at a distance of 1m, meaning that the research from chapter 4 may no longer be valid for this system. However, a system with better tracking sensors will eventually allow users to stake out while standing.

In conclusion, the WOAR system performed well compared to a commercial stakeout system. Many of the shortcomings identified by the participants can be addressed in the future. Tracking accuracy and reliability can potentially be improved by adding vision based tracking. The WOAR system is not yet ready for commercial release. However, it was robust enough to withstand rigorous testing and outperformed the commercial system for this specific task.

Chapter VII

Conclusion

This thesis describes formal user studies and explorative implementations that improved the usability and the efficiency of a wearable outdoor augmented reality (WOAR) stakeout application. The presented research surveyed previous work, formally compared the performance of interface components, and proposed novel interface components and interaction techniques. It formally compared the performance of the developed WOAR stakeout application to that of a commercial stakeout system and found that it enabled users to perform twice as fast at the same accuracy level. The evaluation also identified strengths and weaknesses of a modern WOAR system.

The presented implementation of a WOAR stakeout application is based on research in the areas of navigation, depth cues and the visualisation of obscured information. The thesis presents significant research contributions in each of these areas.

7.1 Summary of Results

7.1.1 Directional Interfaces for WOAR Navigation

The thesis surveys previous work for WOAR navigation and describes the first comprehensive user study that formally compared directional interfaces for this application. There was a significant difference in the performance of the interfaces. A circular compass with novel modifications was the fastest, most accurate, and preferred solution.

The thesis researches directional interfaces in the context of navigational tasks that require the user to move from waypoint to waypoint on a predetermined route. Waypoints and path elements connecting them are either real or created by the WOAR system. In this framework, directional interfaces help users to orient themselves in the direction of the next target if that target is currently outside their field of view (FOV). The limited FOV of current video see-through head-mounted displays (HMDs) makes it harder for users to search for targets without the help of a directional interface.

The user study was the first to formally compare the performance of a haptic belt to that of other directional interfaces for navigation. It found that participants were able to follow the belt's directions. However, the circular compass visualisation performed better.

The described original research found that the difference between the human FOV and the HMD FOV caused confusion in the users. This thesis proposes to call this difference the phantom field of view. An explicit visualisation of the HMD's FOV and the human FOV in the circular compass avoided this confusion.

The thesis also presents a WOAR implementation of a visualisation of a navigation framework with waypoints, path elements, and the modified circular compass.

7.1.2 Depth Cues for AR Stakeout

The thesis presents a user study that formally compared depth cues for AR stakeout. It found that, for the stakeout task, participants relied on kinesthetic feedback rather than visual depth cues. However, while the participants' performance was not influenced by the different visual depth cues, they strongly preferred some visual cues over others. The thesis proposes a novel depth cue visualisation: the cast circle. This visualisation was preferred by the participants.

A pilot study found that correct occlusion of the real stakeout pole by virtual stakeout markers was fundamental to a good user experience of AR stakeout. By default, AR applications do not provide correct occlusion of real objects by virtual objects, but simply overlay virtual content on top of the real imagery instead. This often results in real objects being wrongly occluded by the virtual objects. The thesis presents an algorithm and its WOAR implementation that provided correct occlusion of a real stakeout pole by virtual markers.

7.1.3 Obscured Information Visualisation for WOAR Road Stakeout

The thesis presents a WOAR visualisation of a road model that allowed users to see both the overall shape as well as the internal structure of the road. Using information filtering techniques, the visualisation automatically adapted to the users' current location and displayed more or less detail. This allowed users to see an overview of the road free of clutter and information overload, while at the same time each of the potentially hundreds of stakeout points of the road model were accessible. No explicit user input was necessary, allowing the users to concentrate on individual stakeout points rather than forcing users to interact with the interface.

The thesis further presents a novel interaction technique with artificially transparent stakeout poles and hands. Stakeout with a transparent stakeout pole ensures that the target location is not obscured by the pole. Explorative implementations showed that users are able to use artificially transparent stakeout poles and hands successfully. However, finding optimal transparency levels that conveyed both the obscured background as well as the shape of the transparent object was difficult. The thesis explores techniques such as selective transparency or wireframe outlines to enhance the most relevant edges of artificially transparent objects. A WOAR implementation of an artificially transparent real stakeout pole is demonstrated.

7.1.4 A WOAR Stakeout Application

The thesis describes the implementation and formal evaluation of a WOAR stakeout application. The system, that the application was built on, allowed for accurate stakeout and is the most accurate WOAR system to date. It demonstrated several techniques to make GPS reception more reliable. The interface of the application applied the results of the original research presented in this thesis.

A formal user study compared the performance the WOAR application to the performance of the Trimble Survey Controller (TSC), a commercial conventional stakeout system. The study found that the WOAR system was significantly faster than the TSC. The WOAR system was twice as fast as the TSC at a significant level at an accuracy of 4cm, and the participants preferred the WOAR system over the conventional one.

The study also identified the loss in accuracy caused by different components of the WOAR system. It found that the tracking errors for the position and orientation sensors outweighed perceptual errors that were caused by factors such as lag, jitter, and the low resolution of the HMD.

7.2 Lessons Learned

There were certain strategies that were useful throughout the development of the WOAR system and the exploration of user interfaces. These practices are recommended when developing new WOAR applications.

- **Indoor testing** allows for far greater flexibility than outdoor testing. Changes to the code can be applied quicker, implementations can be tried out sooner and more often, and the work is not weather or daylight dependent. There are several ways systems and interface components can be tested indoors:

  - Desk-based AR systems allow for rapid prototyping
  - A WOAR application that is able to play back recorded sensor data can convey how well a visualisation might work in the field, and allows for fine-tuning of visualisations.

  However, trialling new interface ideas in an outdoor setting early on is recommended as well. This can reveal problems and can suggest new forms of interaction with the system that would stay unrecognised in an indoor testing system.

- **Informal user studies** allow for a quick and easy assessment of the feasibility of a prototype implementation. They also provide new ideas and directions, and can identify strengths and shortcomings of the developed interface early on. Iterative development is essential for novel interfaces such as WOAR applications.

- **User comfort** may not be the first and foremost concern when developing exciting new visualisation techniques. However, when evaluating a WOAR visualisation in a user study, users might concentrate more on the uncomfortably heavy HMD that presses on their nose than on the visualisation.

- **Abstraction layers** in the software system help with quickly developing or adding interface features. Sensor management and the visual interface should be kept separate, allowing for clean development of visualisations. This also allows the change of tracking sensors without affecting other parts of the system. In a project that runs over several years, extensibility has to be considered.

- **Collecting objective data** throughout the development process, even in informal manner, can make decisions in the design process easier and leaves less room for guesswork. Measurements can include fundamental system properties such as lag and frame rate to ensure the usability of the system. Simple data such as user movement can be logged during test trials of new interfaces to get a better understanding of how the system is used. Little is known about the objective performance of WOAR systems, and every collected set of data may contain relevant new insights.

7.3 Future Work

In general, WOAR systems need better hardware. Tracking solutions need to become more accurate and more affordable. The equipment needs to become smaller and lighter. Head-mounted displays need to become more lightweight and provide higher quality images.

Nevertheless, many research opportunities remain in user interface design for WOAR systems. Examples relating to the research presented in this thesis are given in this section.

7.3.1 Directional Interfaces

While the experiment presented in chapter 3 showed that a horizontal compass did not perform well, a horizontal compass with a reverse orientation might be more successful. A formal comparison of such a horizontal compass with the circular compass is required to answer this.

A study is required to find a less intrusive way of integrating a directional interface into a WOAR display than the one used in this thesis. The presented solution overlaid a large circular compass onto the centre of the user's field of view. A study could explore how small such a compass can be made, and if it is still efficient if presented off centre.

The framework for navigation used in this thesis, consisting of waypoints, path elements, and the circular compass, does not provide an overview of the route or of the user's environment. The integration of map views into the framework needs to be researched.

7.3.2 Depth Cues

The user study described in chapter 4 should be examined to explore if the results can be improved further with reduced lag and higher frame rates. In addition, the influence on kinesthetic depth knowledge for the stakeout task needs to be investigated. For this, the study could examine the effect of the removal of any visual and haptic cues.

Depth cues for navigation need to be researched in order to solve the problem of overshooting stakeout targets. The user study presented in chapter 6 found that participants had difficulty in judging their distance from a target that they were approaching. Appropriate cues that help users decelerate in time before the target is reached are needed for efficient navigation.

7.3.3 Obscured Information Visualisation

With the positive explorative results for artificially transparent stakeout poles and hands presented in this thesis, formal user studies are required to assess the performance of such visualisations.

Optimal transparency levels as well as appropriate edge enhancement techniques need to be investigated first. The observed effect of light objects appearing more transparent than dark objects needs to be investigated in order to present objects with coherently perceived transparency levels.

7.3.4 WOAR Stakeout

The future development of the WOAR stakeout application can be taken in three directions.

(1) The first direction investigates input devices, menu interfaces, and interaction techniques that go towards developing a complete stakeout application.

(2) The second direction investigates accurate six degree of freedom tracking of the pole and of the ground. This would allow the WOAR application to provide the same functionality as the TSC and more.

    With added vision based tracking, a WOAR system would be able to capture a high resolution and high accuracy textured terrain model, which only requires the user to walk the terrain and look at the ground. With the TSC, terrain models are usually captured at a resolution of 20 metres, requiring the user to place the TSC at each of these locations and measure them manually.

    A high-resolution terrain model captured through vision based tracking could be used to ensure that stakeout markers are drawn on the ground. This is an inherent problem in stakeout, as the correct elevation of the stakeout targets is usually not known. In the case of road stakeout, a road design is usually created on a desktop computer based on a rough terrain model surveyed with the TSC as mentioned above. This means that the elevation of the design points is interpolated from the terrain model. For staking out those points, the TSC does not need to know the correct elevation in advance. However, displaying a target marker with the wrong elevation in the WOAR system will cause parallax effects. This means that the marker appears to move horizontally against the ground as the user's head moves. As a result, the marker will not be displayed in the correct location from most viewing angles.

(3) The third direction investigates different forms of outdoor augmented reality. Projector and handheld based AR may be powerful alternatives to HMD based AR. In projector based AR, the virtual content is projected directly onto the real world, avoiding significant problems of HMDs. The wide human field of view and depth cues such as binocular disparities are preserved. Outdoor augmented reality on handheld systems might not be an optimal solution for stakeout, but it might be successful for related applications such

as surveying or for geographic information system (GIS) visualisations.

These research directions explore exciting new features for outdoor AR applications.

7.4 Conclusion

This thesis consolidates WOAR interface research and extends the field with empirical research. The thesis shows for the first time that a WOAR application outperformed a conventional system in a formal user study. These results will hopefully inspire more researchers to investigate outdoor AR applications in order to explore novel interfaces and applications, and to improve inherent problems such as outdoor tracking.

References

Abawi, D. F., Bienwald, J. & Dorner, R. [2004], Accuracy in optical tracking with fiducial markers: An accuracy function for artoolkit, *in* 'ISMAR '04: Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality', IEEE Computer Society, Washington, DC, USA, pp. 260–261.

Avery, B., Piekarski, W. & Thomas, B. H. [2007], Visualizing occluded physical objects in unfamiliar outdoor augmented reality environments, *in* '6th IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR), 2007', pp. 285–286.

Avery, B., Piekarski, W., Warren, J. & Thomas, B. H. [2006], Evaluation of user satisfaction and learnability for outdoor augmented reality gaming, *in* 'AUIC '06: Proceedings of the 7th Australasian User interface conference', Australian Computer Society, Inc., Darlinghurst, Australia, Australia, pp. 17–24.

Avery, B., Thomas, B. H., Velikovsky, J. & Piekarski, W. [2005], Outdoor augmented reality gaming on five dollars a day, *in* 'AUIC '05: Proceedings of the Sixth Australasian conference on User interface', Australian Computer Society, Inc., Darlinghurst, Australia, Australia, pp. 79–88.

Azuma, R. [1997], 'A survey of augmented reality', *Presence: Teleoperators and Virtual Environments* **6**(4), 355–385.

Azuma, R., Baillot, Y., Behringer, R., Feiner, S., Julier, S. & MacIntyre, B. [2001], 'Recent advances in augmented reality', *IEEE Computer Graphics and Applications* **21**(6), 34–47.

Azuma, R., Neely, H., Daily, M. & Leonard, J. [2006], Performance analysis of an outdoor augmented reality tracking system that relies upon a few mobile beacons, *in* 'IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR), 2006', pp. 101–104.

Bajura, M., Fuchs, H. & Ohbuchi, R. [1992], 'Merging virtual objects with the real world: seeing ultrasound imagery within the patient', *SIGGRAPH Comput. Graph.* **26**(2), 203–210.

Bane, R. & Höllerer, T. [2004], Interactive tools for virtual X-ray vision in mobile augmented reality, *in* 'ISMAR '04: Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality', IEEE Computer Society, Washington, DC, USA, pp. 231–239.

Billinghurst, M., Bowskill, J., Dyer, N. & Morphett, J. [1998], An evaluation of wearable information spaces, *in* 'VRAIS '98: Proceedings of the Virtual Reality Annual International Symposium', IEEE Computer Society, Washington, DC, USA, p. 20.

Billinghurst, M. & Kato, H. [2000], 'Out and about — real world teleconferencing', *BT Technology Journal* **18**(1), 80–82.

Bingham, G., Bradley, A., Bailey, M. & Vinner, R. [2001], 'Accommodation, occlusion and disparity matching are used to guide reaching: a comparison of actual versus virtual environments', *Journal of Experimental Psychology: Human Perception and Performance* **27**(6), 1314–1334.

Buchmann, V., Billinghurst, M. & Cockburn, A. [2008], Directional interfaces for wearable augmented reality, *in* 'CHINZ '08: Proceedings of the 8th ACM SIGCHI New Zealand chapter's international conference on Computer-human interaction', ACM, New York, NY, USA.

Buchmann, V., Nilsen, T. & Billinghurst, M. [2005], Interaction with partially transparent hands and objects, *in* 'AUIC '05: Proceedings of the Sixth Australasian conference on User interface', Australian Computer Society, Inc., Darlinghurst, Australia, Australia, pp. 17–20.

Buchmann, V., Violich, S., Billinghurst, M. & Cockburn, A. [2004], Fingartips: gesture based direct manipulation in augmented reality, *in* 'Proceedings of the 2nd international conference on Computer graphics and interactive techniques in Australasia and South East Asia', ACM Press, pp. 212–221.

Cheok, A. D., Fong, S. W., Goh, K. H., Yang, X., Liu, W. & Farzbiz, F. [2003], Human pacman: a sensing-based mobile entertainment system with ubiquitous computing and tangible interaction, *in* 'Proceedings of the 2nd workshop on Network and system support for games', ACM Press, pp. 106–117.

Cheok, A. D., Goh, K. H., Liu, W., Farbiz, F., Fong, S. W., Teo, S. L., Li, Y. & Yang, X. [2004], 'Human pacman: a mobile, wide-area entertainment system based on physical, social, and ubiquitous computing', *Personal Ubiquitous Comput.* **8**(2), 71–81.

Cheok, A. D., Wan, F. S., Goh, K. H., Yang, X., Liu, W., Farzbiz, F. & Li, Y. [2003], Human pacman: A mobile entertainment system with ubiquitous computing and tangible interaction over a wide outdoor area, *in* 'Mobile HCI 2003', pp. 209–223.

Cheok, A. D., Wan, F. S., Yang, X., Weihua, W., Huang, L. M., Billinghurst, M. & Kato, H. [2002], Game-city: A ubiquitous large area multi-interface mixed reality game space for wearable computers, *in* 'ISWC 2002', pp. 156–157.

Cutting, J. E. [1997], 'How the eye measures reality and virtual reality', *Behavior Research Methods, Instruments & Computers* **29**(1), 27–36.

Darken, R. P. & Cevik, H. [1999], Map usage in virtual environments: Orientation issues, *in* 'VR '99: Proceedings of the IEEE Virtual Reality', IEEE Computer Society, Washington, DC, USA, pp. 133–140.

Dizio, P. & Lackner, J. [2002], 'Proprioceptive adaption and aftereffects', *Handbook of Virtual Environments* pp. 751–771.

Drascic, D. & Milgram, P. [1996], Perceptual issues in augmented reality, *in* 'Proceedings of SPIE', Vol. 2653, pp. 123–134.

Ellis, S. R. & Menges, B. M. [1998], 'Localization of virtual objects in the near visual field', *Human Factors* **40**(3), 415–431.

Feiner, S., MacIntyre, B., Höllerer, T. & Webster, A. [1997], A touring machine: Prototyping 3d mobile augmented reality systems for exploring the urban environment, *in* 'Proceedings of ISWC 1997', pp. 74–81.

Feiner, S., Macintyre, B. & Seligmann, D. [1993], 'Knowledge-based augmented reality', *Commun. ACM* **36**(7), 53–62.

Furmanski, C., Azuma, R. & Daily, M. [2002], Augmented-reality visualizations guided by cognition: Perceptual heuristics for combining visible and obscured information, *in* 'ISMAR '02: Proceedings of the 1st International Symposium on Mixed and Augmented Reality', IEEE Computer Society, Washington, DC, USA, p. 215.

Gaines, B. [1991], 'Modeling and forecasting the information sciences', *Information Sciences* **3**(22), 57–58.

Gleue, T. & Dähne, P. [2001], Design and implementation of a mobile device for outdoor augmented reality in the archeoguide project, *in* 'Proceedings of the 2001 conference on Virtual reality, archeology, and cultural heritage', ACM Press, pp. 161–168.

Guven, S. & Feiner, S. [2006], Visualizing and navigating complex situated hypermedia in augmented and virtual reality, *in* 'IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR), 2006', pp. 155–158.

Hart, S. & Staveland, L. [1988], Development of nasa-tlx (task load index): Results of empirical and theoretical research, *in* P. a M Hancock, ed., 'Human Mental Workload', pp. 139–183.

Hendrix, C. & Barfield, W. [1995], 'Relationship between monocular and binocular depth cues for judgements of spatial information and spatial instrument design', *Displays* **16**(3), 103–113.

Henrysson, A. & Billinghurst, M. [2007], Using a mobile phone for 6 dof mesh editing, *in* 'CHINZ '07: Proceedings of the 7th ACM SIGCHI New Zealand chapter's international conference on Computer-human interaction', ACM, New York, NY, USA, pp. 9–16.

Herbst, I., Ghellah, S. & Braun, A.-K. [2007], Timewarp: an explorative outdoor mixed reality game, *in* 'SIGGRAPH '07: ACM SIGGRAPH 2007 posters', ACM, New York, NY, USA, p. 149.

Heuer, H. [2003], Motor control, *in* 'Handbook of Psychology', Vol. 4, Wiley, pp. 317–354.

Höllerer, T. [2004], User Interfaces for Mobile Augmented Reality Systems, PhD thesis, Columbia University.

Höllerer, T., Feiner, S., Terauchi, T., Rashid, G. & Hallaway, D. [1999], 'Exploring mars: Developing indoor and outdoor user interfaces to a mobile augmented reality system', *Computers and Graphics* **23**(6), 779–785.

Höllerer, T., Hallaway, D., Tinna, N. & Feiner, S. [2001], Steps toward accommodating variable position tracking accuracy in a mobile augmented reality system, *in* 'AIMS 01: Seond Int. Workshop on Artificial Intelligence in Mobile Systems', pp. 31–37.

Höllerer, T., Pavlik, J. V. & Feiner, S. [1999], Situated documentaries: Embedding multimedia presentations in the real world, *in* 'Proc. ISWC 1999 (International Symposium on Wearable Computers)', pp. 79–86.

Hubona, G. S., Wheeler, P. N., Shirah, G. W. & Brandt, M. [1999], 'The relative contributions of stereo, lighting, and background scenes in promoting 3d depth visualization', *ACM Trans. Comput.-Hum. Interact.* **6**(3), 214–242.

Julier, S., Baillot, Y., Lanzagorta, M., Brown, D. & Rosenblum, L. [2000], Bars: Battlefield augmented reality system, *in* 'NATO Symposium on Information Processing Techniques for Military Systems'.

Julier, S., Lanzagorta, M., Baillot, Y. & Brown, D. [2002], 'Information filtering for mobile augmented reality', *Projects in VR, IEEE Computer Graphics & Applications* **22**(5), 12–15.

Julier, S., Livingston, M. A., Swan II, J. E., Baillot, Y. & Brown, D. [2004], Adaptive user interfaces in augmented reality, *in* 'Proceedings of the Workshop on Software Technology for Augmented Reality Systems, NRL Technical Memorandum Report'.

Jürgens, V., Cockburn, A. & Billinghurst, M. [2006], Depth cues for augmented reality stakeout, *in* 'CHINZ '06: Proceedings of the 7th ACM SIGCHI New Zealand chapter's international conference on Computer-human interaction', ACM, New York, NY, USA, pp. 117–124.

Kalkofen, D., Mendez, E. & Schmalstieg, D. [2007], Interactive focus and context visualization for augmented reality, *in* '6th IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR), 2007', pp. 191–201.

Kameda, Y., Takemasa, T. & Ohta, Y. [2004], Outdoor mixed reality utilizing surveillance cameras, *in* 'Proceedings of the 31st annual conference on Computer graphics and interactive techniques', ACM Press.

Kato, H. & Billinghurst, M. [1999], Marker tracking and hmd calibration for a video-based augmented reality conferencing system, *in* 'IWAR '99: Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality', IEEE Computer Society, Washington, DC, USA, p. 85.

Kaufmann, H. & Schmalstieg, D. [2002], Mathematics and geometry education with collaborative augmented reality, *in* 'SIGGRAPH '02: ACM SIGGRAPH 2002 conference abstracts and applications', ACM, New York, NY, USA, pp. 37–41.

Kersten, D., Mamassian, P. & Knill, D. C. [1997], 'Moving cast shadows induce apparent motion in depth', *Perception* **26**, 171–192.

Klatzky, R. & Lederman, S. [2003], Touch, *in* 'Handbook of Psychology', Vol. 4, Wiley, pp. 147–176.

Klinker, G., Stricker, D. & Reiners, D. [2001], Augmented reality for exterior construction applications, *in* 'Fundamentals of Wearable Computers and Augmented Reality', pp. 379–427.

Kölsch, M., Bane, R., Höllerer, T. & Turk, M. [2006], 'Multimodal interaction with a wearable augmented reality system', *IEEE Comput. Graph. Appl.* **26**(3), 62–71.

Kretschmer, U., Coors, V., Spierling, U., Grasbon, D., Schneider, K., Rojas, I. & Malaka, R. [2001], Meeting the spirit of history, *in* 'Proceedings of the 2001 conference on Virtual reality, archeology, and cultural heritage', ACM Press, pp. 141–152.

Lepetit, V. & Berger, M.-O. [2001], An intuitive tool for outlining objects in video sequences: Applications to augmented and diminished reality, *in* 'Proceedings of the second International Symposium on Mixed Reality, ISMR 2001', pp. 159–160.

Livingston, M. A., Lederer, A., Ellis, S. R., White, S. & Feiner, S. [2006], Vertical vengence calibration for augmented reality displays, *in* 'Proceedings of IEEE Virtual Reality Conference', pp. 287 – 288.

Livingston, M. A., Swan II, J. E., Gabbard, J. L., Höllerer, T. H., Hix, D., Julier, S. J., Baillot, Y. & Brown, D. [2003], Resolving multiple occluded layers in augmented reality, *in* 'Proceedings of the The 2nd IEEE and ACM International Symposium on Mixed and Augmented Reality', IEEE Computer Society, p. 56.

MacIntyre, B. & Feiner, S. [1996], 'Future multimedia user interfaces', *Multimedia Systems* **4**(5), 250–268.

Malbezin, P., Piekarski, W. & Thomas, B. H. [2002], Measuring artoolkit accuracy in long distance tracking experiments, *in* '1st International Augmented Reality Toolkit Workshop 2002'.

Mann, S. [1994], Mediated reality, *in* 'Technical Report 260, M.I.T. Media Lab Perceptual Computing Section, Cambridge, Ma.'.

Mann, S. [1998], Wearable computing as means for personal empowerment, *in* 'Keynote Address, 1998 International Conference on Wearable Computing ICWC-98'.

Mann, S. & Fung, J. [1996], 'Eyetap devices for augmented, deliberately diminished, or otherwise altered visual perception of rigid planar patches of real-world scenes', **11**(2), 158–175.

Mason, A. H., Walji, M. A., Lee, E. J. & MacKenzie, C. L. [2001], Reaching movements to augmented and graphic objects in virtual environments, *in* 'Proceedings of the SIGCHI conference on Human factors in computing systems', ACM Press, pp. 426–433.

McCandless, J. W., Ellis, S. R. & Adelstein, B. D. [2000], 'Localization of a time-delayed, monocular virtual object superimposed on a real environment', *Presence: Teleoper. Virtual Environ.* **9**(1), 15–24.

Milgram, P. & Drasic, D. [1997], Perceptual effects in aligning virtual and real objects in augmented reality displays, *in* '41st annual meeting of Human Factors and Ergonomics Society, Albuquerque, New Mexico'.

Milgram, P., Takemura, H., Utsumi, A. & Kishino, F. [1994], Augmented reality: A class of displays on the reality-virtuality continuum, *in* 'SPIE Vol. 2351, Telemanipulator and Telepresence Technologies'.

Mourgues, F., Devemay, F. & Coste-Maniere, E. [2001], 3d reconstruction of the operating field for image overlay in 3d-endoscopic surgery, *in* 'IEEE and ACM International Symposium on Augmented Reality, 2001', pp. 191–192.

Newman, J., Ingram, D. & Hopper, A. [2001], Augmented reality in a wide area sentient environment, *in* 'ISAR '01: Proceedings of the IEEE and ACM International Symposium on Augmented Reality (ISAR'01)', IEEE Computer Society, Washington, DC, USA, p. 77.

Nilsen, T. [2005], Tankwar: Ar games at gencon indy 2005, *in* 'ICAT '05: Proceedings of the 2005 international conference on Augmented tele-existence', ACM, New York, NY, USA, pp. 243–244.

Papagiannakis, G., Ponder, M., Molet, T., Kshirsagar, S., Cordier, F., Magnenat-Thalmann, N. & Thalmann, D. [2002], Lifeplus: Revival of life in ancient pompeii, *in* 'Proceeding of the 8th International Conference on Virtual Systems and Multimedia'.

Peuchot, B., Tanguy, A. & Eude, M. [1995], Virtual reality as an operative tool during scoliosis surgery, *in* 'CVRMed '95: Proceedings of the First International Conference on Computer Vision, Virtual Reality and Robotics in Medicine', Springer-Verlag, London, UK, pp. 549–554.

Piekarski, W. [2004], Interactive 3D Modelling in Outdoor Augmented Reality Worlds, PhD thesis, University of South Australia.

Piekarski, W., Hepworth, D., Demczuk, V., Thomas, B. & Gunther, B. [1999], A mobile augmented reality user interface for terrestrial navigation, *in* '22nd Australasian Computer Science Conference', pp. 122–133.

Piekarski, W. & Smith, R. [2006], Robust gloves for 3d interaction in mobile outdoor ar environments, *in* 'IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR), 2006', pp. 251–252.

Reitmayr, G. & Drummond, T. [2006], Going out: robust model-based tracking for outdoor augmented reality, *in* 'IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR), 2006', pp. 109–118.

Reitmayr, G. & Schmalstieg, D. [2003], Location based applications for mobile augmented reality, *in* 'CRPITS '03: Proceedings of the Fourth Australian user interface conference on User interfaces 2003', Australian Computer Society, Inc., Darlinghurst, Australia, Australia, pp. 65–73.

Reitmayr, G. & Schmalstieg, D. [2004], Collaborative augmented reality for outdoor navigation and information browsing, *in* 'Proc. Symposium Location Based Services and TeleCartography'.

Rosenberg, L. [1993], The effect of interocular distance upon operator performance using stereoscopic displays to perform virtual depth tasks, *in* 'Virtual Reality Annual International Symposium, 1993', IEEE, pp. 27–32.

Ross, D. A. & Blasch, B. B. [2002], 'Development of a wearable computer orientation system', *Personal Ubiquitous Comput.* **6**(1), 49–63.

Ross, D. & Blasch, B. [2000], Evaluation of orientation interfaces for wearable computers, *in* 'The Fourth International Symposium on Wearable Computers', pp. 51–58.

Satoh, K., Hara, K., Anabuki, M., Yamamoto, H. & Tamura, H. [2001], Townwear: An outdoor wearable mr system with high-precision registration, *in* 'Proceedings of the Second International Symposium on Mixed Reality', pp. 210–211.

Schmalstieg, D. & Reitmayr, G. [2005], The world as a user interface: Augmented reality for ubiquitous computing, *in* 'Central European Multimedia and Virtual Reality Conference 2005 (CEMVRC 2005)'.

Schmeil, A. & Broll, W. [2007], Mara - a mobile augmented reality-based virtual assistant, *in* 'Proceedings of the Virtual Reality Conference, 2007', IEEE, pp. 267–270.

Sephton, T. [2001], 'Wearable augmented reality as a navigation aid for blind and sighted people (poster)', 5th International Symposium on Wearable Computers.

Sephton, T. [2002], Visualizing maya cities on site. a wearable augmented reality tool for archeological field study.

Sephton, T. [2003], Teaching agents for wearable augmented reality, *in* 'Proceedings of ISWC 2003'.

Sephton, T., Black, J., Naggar, G. E. & Fong, A. [1999], A mobile computing system for testing wearable augmented reality user interface design.

Sielhorst, T., Sa, W., Khamene, A., Sauer, F. & Navab, N. [2007], Measurement of absolute latency for video see through augmented reality, *in* '6th IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR), 2007', pp. 215–220.

Smith, R., Piekarski, W. & Wigley, G. [2005], Hand tracking for low powered mobile ar user interfaces, *in* 'AUIC '05: Proceedings of the Sixth Australasian conference on User interface', Australian Computer Society, Inc., Darlinghurst, Australia, Australia, pp. 7–16.

Stafford, A., Piekarski, W. & Thomas, B. [2006], Implementation of god-like interaction techniques for supporting collaboration between outdoor ar and indoor tabletop users, *in* 'IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR), 2006', pp. 165–172.

Starner, T., Mann, S., Rhodes, B., Healey, J., Russell, K., Levine, J. & Pentland, A. [1995], Wearable computing and augmented reality, Technical Report 355, M.I.T. Media Vision Lab and Modeling Group.

Starner, T., Mann, S., Rhodes, B., Levine, J., Healey, J., Kirsch, D., Picard, R. & Pentland, A. [1997], 'Augmented reality through wearable computing', *Presence* **6**(4), 386–398.

State, A., Chen, D., Tector, C., Brandt, A., Hong, C., Ohbuchi, R., Bajura, M. & Fuchs, H. [1994], Observing a volume rendered fetus within a pregnant patient, *in* 'Proceedings of the IEEE Conference an Visualization 1994', IEEE, pp. 364–368.

State, A., Livingston, M. A., Garrett, W. F., Hirota, G., Whitton, M. C., Pisano, E. D. & Fuchs, H. [1996], Technologies for augmented reality systems: realizing ultrasound-guided needle biopsies, *in* 'SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques', ACM, New York, NY, USA, pp. 439–446.

Stricker, D. [2001], Tracking with reference images: a real-time and markerless tracking solution for out-door augmented reality applications, *in* 'VAST '01: Proceedings of the 2001 conference on Virtual reality, archeology, and cultural heritage', ACM, New York, NY, USA, pp. 77–82.

Suomela, R. & Lehikoinen, J. [2000], Context compass, *in* 'The Fourth International Symposium on Wearable Computers', pp. 147–154.

Suomela, R., Roimela, K. & Lehikoinen, J. [2003], 'The evolution of perspective view in walkmap', *Personal Ubiquitous Comput.* **7**(5), 249–262.

Sutherland, I. E. [1968], A head-mounted three-dimensional display, *in* 'Fall Jopint Computer Conference, American Federation of Information Precessing Society (AFIPS) Conference Proceedings 33', Thompson Books, pp. 757–764.

Swan, J. E. I., Livingston, M. A., Smallman, H. S., Brown, D., Baillot, Y., Gabbard, J. L. & Hix, D. [2006], A perceptual matching technique for depth judgments in optical, see-through augmented reality, *in* 'VR '06: Proceedings of the IEEE conference on Virtual Reality', IEEE Computer Society, Washington, DC, USA, pp. 19–26.

Tan, H. Z. & Pentland, A. [1997], Tactual displays for wearable computing, *in* 'First International Symposium on Wearable Computers, 1997. Digest of Papers', pp. 84–89.

Thakoor, N., Gao, J. & Chen, H. [2004], Autimatic object detection in video sequences with camera in motion, *in* 'Proceedings of the International Conference on Advanced Concepts for Intelligent Vision Systems', pp. 111–117.

Thomas, B., Close, B., Donoghue, J., Squires, J., Bondi, P. D., Morris, M. & Piekarski, W. [2000], Arquake: An outdoor/indoor augmented reality first person application, *in* '4th International Symposium on Wearable Computers', pp. 139–146.

Thomas, B., Close, B., Donoghue, J., Squires, J., Bondi, P. D. & Piekarski, W. [2002], 'First person indoor/outdoor augmented reality application: Arquake', *Personal Ubiquitous Comput.* **6**(1), 75–86.

Thomas, B., Demczuk, V., Piekarski, W., Hepworth, D. & Gunther, B. [1998], A wearable computer system with augmented reality to support terrestrial navigation, *in* '2nd International Symposium on Wearable Computers', pp. 168–171.

Thomas, B. H., Grimmer, K., Zucco, J. & Milanese, S. [2002], 'Where does the mouse go? - an investigation into the placement of a body-attached touchpad mouse for wearable computers', *Personal and Ubiquitous Computing* **6**(1), 97–112.

Thomas, B., Krul, N., Close, B. & Piekarski, W. [2002], Usability and playability issues for arquake, *in* '1st International Workshop on Entertainment Computing'.

Trimble [2005], *Trimble Survey Controller – Staking out a Road Demonstration Guide*, Trimble Navigation Limited.

Tsuda, T., Yamamoto, H., Kameda, Y. & Ohta, Y. [2005], Visualization methods for outdoor see-through vision, *in* 'ICAT '05: Proceedings of the 2005 international conference on Augmented tele-existence', ACM, New York, NY, USA, pp. 62–69.

Van Erp, J., VanVeen, H. & Jansen, C. [2005], 'Waypoint navigation with a vibrotactile waist belt', *ACM Transactions on Applied Perception* **2**(2), 106–117.

Vlahakis, V., Karigiannis, J., Tsotros, M., Gounaris, M., Almeida, L., Stricker, D., Gleue, T., Christou, I. T., Carlucci, R. & Ioannidis, N. [2001], Archeoguide: first results of an augmented reality, mobile computing system in cultural heritage sites, *in* 'Proceedings of the 2001 conference on Virtual reality, archeology, and cultural heritage', ACM Press, pp. 131–140.

Vlahakis, V., Karigiannis, J., Tsotros, M., Ioannidis, N. & Stricker, D. [2002], Personalized augmented reality touring of archaeological sites with wearable and mobile computers, *in* 'Proceedings of ISWC 2002', pp. 15–22.

Wagner, M. [1995], 'The metrics of visual space', *Perception and Psychophysics* **38**(6), 483–495.

Ware, C. [2000], *Information Visualization*, Academic Press.

Ware, C. & Balakrishnan, R. [1994], 'Reaching for objects in vr displays: lag and frame rate', *ACM Trans. Comput.-Hum. Interact.* **1**(4), 331–356.

Webster, A., Feiner, S., MacIntyre, B., Massie, W. & Krueger, T. [1996], Augmented reality in architectural construction, inspection, and renovation, *in* 'Proceedings of the Third ASCE Congress for Computing in Civil Engineering'.

Wightman, F. & Kistler, D. [1999], 'Resolution of front-back ambiguitiy in spatial hearing by listener and source movement', *Journal of the Acoustical Society of America* **105**(5), 2841–2853.

Wither, J. & Höllerer, T. [2004], Evaluating techniques for interaction at a distance, *in* 'ISWC 2004. Eighth International Symposium on Wearable Computers, 2004.', Vol. 1, pp. 124–127.

Wither, J. & Höllerer, T. [2005], Pictorial cues for outdoor augmented reality, *in* 'Proceedings. Ninth IEEE International Symposium on Wearable Computers, 2005.', IEEE, pp. 92–99.

Zelek, J. S. & Holbein, M. [2006], Feeling where you are – wearable haptic directional belt: a device for wayfinding and orientation for people who are blind, *in* '9e dition du Concours Innovation Recherche 2006'.

Zokai, S., Esteve, J., Genc, Y. & Navab, N. [2003], Multiview paraperspective projection model for diminished reality, *in* ‘Proceedings of the Second IEEE and ACM International Symposium on Mixed and Augmented Reality, ISMAR 2003’, pp. 217–226.

Zucco, J. E., Thomas, B. H. & Grimmer, K. [2005], Evaluation of three wearable computer pointing devices for selection tasks, *in* ‘Proceedings of the Ninth IEEE International Symposium on Wearable Computers, 2005.’.

Appendix A

Supplemental Videos

The supplied CD-ROM contains videos that illustrate the implemented WOAR visualisations described in this thesis.

The videos are encoded in the Xvid format[1]. The latest versions of popular media players should be able to play back the videos without additional software. If required, codecs for Windows and Linux can be downloaded: `http://www.xvid.org/Downloads.15.0.html`. Alternatively, the free and open source VLC media player provides native support for the format for all major operating systems: `http://www.videolan.org/`

The videos have a resolution of 600*800 pixels and use low compression settings. It is highly recommended to copy the videos onto a harddrive before playback.

[1] `http://www.xvid.org/`

A.1 Navigation.avi

This video demonstrates the implementation of the navigation framework described in section 3.5.

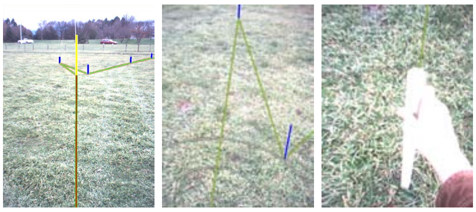


Figure A.1: Navigation with the WOAR system

A.2 Road.avi

This video demonstrates the road visualisation described in section 5.1.

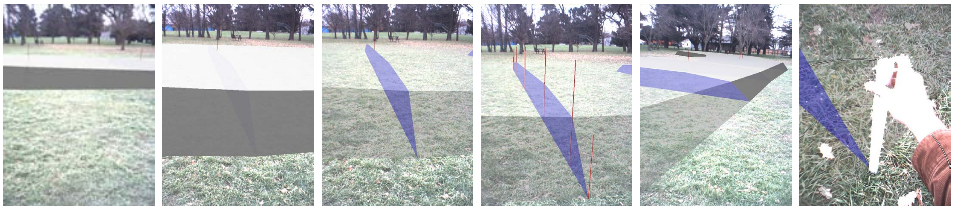


Figure A.2: The WOAR road visualisation

A.3 The Pole Videos

These videos demonstrate different transparency options of the real stakeout pole.

A.3.1 Pole_opaque.avi

This video demonstrates an opaque pole with correct occlusion as described in section 4.10.

Figure A.3: An opaque pole with correct occlusion

A.3.2 Pole_transparent1.avi

This video demonstrates a real stakeout pole that is transparent with respect to the virtual marker but completely opaque with respect to the real environment as described in section 5.4.3.

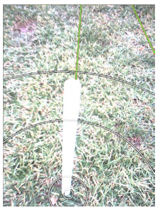


Figure A.4: A real pole that is transparent with respect to the virtual target

A.3.3 Pole_diminished.avi

This video demonstrates the background substitution algorithm described in 5.4.3 that renders a real pole invisible.

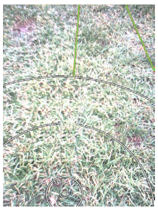


Figure A.5: Background substitution

A.3.4 Pole_transparent2.avi

This video demonstrates a real pole that is artificically made transparent as described in section 5.4.3.

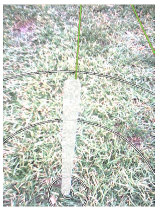


Figure A.6: A real pole that is artificially transparent