2006-02-03

Using Augmented Virtuality to Improve Human-Robot Interactions

Curtis W. Nielsen
Brigham Young University - Provo

USING AUGMENTED VIRTUALITY TO IMPROVE HUMAN-ROBOT INTERACTIONS

by

Curtis W. Nielsen

A dissertation submitted to the faculty of

Brigham Young University

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Department of Computer Science

Brigham Young University

February 2006

BRIGHAM YOUNG UNIVERSITY

GRADUATE COMMITTEE APPROVAL

of a dissertation submitted by

Curtis W. Nielsen

This dissertation has been read by each member of the following graduate committee and by majority vote has been found to be satisfactory.

Date	Michael A. Goodrich, Chair
Date	Dan R. Olsen Jr.
Date	Kevin D. Seppi
Date	Kent E. Seamons
Date	Bryan S. Morse

BRIGHAM YOUNG UNIVERSITY

As chair of the candidate's graduate committee, I have read the dissertation of Curtis W. Nielsen in its final form and have found that (1) its format, citations, and bibliographical style are consistent and acceptable and fulfill university and department style requirements; (2) its illustrative materials including figures, tables, and charts are in place; and (3) the final manuscript is satisfactory to the graduate committee and is ready for submission to the university library.

Date

Michael A. Goodrich
Chair, Graduate Committee

Accepted for the Department

Tony R. Martinez
Department Chair

Accepted for the College

Thomas W. Sederberg
Associate Dean, College of Physical and Mathematical Sciences

ABSTRACT

USING AUGMENTED VIRTUALITY TO IMPROVE

HUMAN-ROBOT INTERACTIONS

Curtis W. Nielsen

Computer Science

Doctor of Philosophy

Mobile robots can be used in situations and environments that are distant from an operator. In order for an operator to control a robot effectively he or she requires an understanding of the environment and situation around the robot. Since the robot is at a remote distant from the operator and cannot be directly observed, the information necessary for an operator to develop an understanding or awareness of the robot's situation comes from the user interface. The usefulness of the interface depends on the manner in which the information from the remote environment is presented.

Conventional interfaces for interacting with mobile robots typically present information in a multi-windowed display where different sets of information are presented in different windows. The disjoint sets of information require significant cognitive processing on the part of the operator to interpret and understand the information. To reduce the cognitive effort to interpret the information from a mobile robot, requirements and technology for a three-dimensional augmented virtuality interface are presented. The 3D interface is designed to combine multiple sets of information

into a single correlated window which can reduce the cognitive processing required to interpret and understand the information in comparison to a conventional (2D) interface.

The usefulness of the 3D interface is validated, in comparison to a prototype of conventional 2D interfaces, through a series of navigation- and exploration-based user-studies. The user studies reveal that operators are able to drive the robot, build maps, find and identify items, and finish tasks faster with the 3D interface than with the 2D interface. Moreover, operators have fewer collisions, avoid walls better, and use a pan-tilt-zoom camera more with the 3D interface than with the 2D interface. Performance with the 3D interface is also more tolerant to network delay and distracting sets of information.

Finally, principles for presenting multiple sets of information to a robot operator are presented. The principles are used to discuss and illustrate possible extensions of the 3D interface to other domains.

ACKNOWLEDGMENTS

I would first like to thank my wife, Anna, for all her support, she has been my biggest fan from the beginning and her encouragement has been priceless. Her devotion to our family has made me want to be the best I can in all aspects of my life. I am also grateful for my parents and siblings as they provided counsel, friendship,encouragement, and support throughout my studies.

I would like to thank my advisor Mike Goodrich for spending the time and energy to help me become a good research student. His review of my work and especially my writing helped me recognize weaknesses and strengthen them. I am especially grateful for the opportunities he provided that helped improve my abilities and confidence and for the enjoyable atmosphere he provided for us to work in.

I am very grateful for the opportunity I had to pursue my education at BYU. I feel that it has been a very unique education that could not have been reproduced anywhere else. I am thankful for the many faculty members I have interacted with who have provided advice or instruction at critical points throughout my studies and who continually pushed me to be a little better. I would also like to thank my fellow students and research assistants for their ideas, input, help, encouragement, and most importantly their friendship. They really made the time at BYU enjoyable.

I am grateful to DARPA and the ARL for financial support and I am especially thankful to the researchers at the INL who provided financial and technical support as well as educational opportunities and employment.

Finally, I want to acknowledge our Father in Heaven who has supported my family and I continually through both good and hard times. His divine guidance and love have been invaluable throughout all aspects of my life.

Contents

Chapter 1

Introduction

Robots have been used in a variety of settings where human access is difficult, impractical, or dangerous. These settings include search and rescue, space exploration, toxic site cleanup, reconnaissance, patrols, and many others. Such settings provide a unique problem in that the robot operator is distant from the actual robot due to safety concerns. In order to operate a robot efficiently at remote distances, it is important for the operator to be aware of the environment around the robot so that the operator can give informed, accurate instructions to the robot. This awareness of the environment is often referred to as *telepresence* [110, 109] or *situation awareness* [35, 93].

1.1 Poor Situation Awareness

Despite the importance of situation awareness in remote-robot operations, experience has shown that interfaces between humans and robots typically do not sufficiently support the operator's awareness of the robot's location and surroundings. As an example, in September 2001, robots were used to search the rubble of the World Trade Center for survivors [23]. The robots were useful because they were able to go into small, dangerous areas that were inaccessible to rescue workers; however, it was quite difficult for the operator to navigate the robot while searching the environment because the robots only provided video information to the operator [23]. The limited angular view of most cameras creates a sense of trying to understand the environment through a 'soda straw' or a 'keyhole' [138, 137]. This limited view of the robot's

environment makes it difficult for an operator to be aware of the robot's proximity to obstacles [3, 6].

In comparison to the robots used at the World Trade Center, many robots used for studying human-robot interactions (HRI) have range sensors and a map-building algorithm in addition to the camera. However, despite better equipment, recent experiments suggest that operators still experience inadequate levels of situation awareness [33, 140]. In one experiment, Yanco and Drury had first responders search a mock environment looking for victims using a robot that had sonar, laser, camera, and map-building capabilities [140]. They found that despite spending up to 30% of their time acquiring situation awareness the participants often expressed and demonstrated confusion concerning the robot's location relative to obstacles, landmarks, and previous locations. A common complaint among the participants was that the map built by the robot was totally useless because it did not help them understand the robot's location [140].

In another field study involving rescue robots in an Urban Search and Rescue (USAR) training exercise, Burke et al. found that operators spent up to 54% of their time acquiring situation awareness as opposed to navigating the robot [18]. Again, despite spending most of their time acquiring information about the robot and the environment, the participants still had difficulty using the robot's information to improve their own understanding of the search and rescue site.

The lack of situation awareness observed in the previous examples is not limited to rescue personnel and others who may be unfamiliar with the robot and the interface. Similar results of poor situation awareness were found in a 2001 AAAI[1] USAR competition [57, 58]. In this competition, it was the engineers that developed the robots and the interface, who competed using their own equipment. Despite the operator's familiarity with the equipment and the abundance of information (laser, sonar, map, and camera), the operators still demonstrated a lack of awareness of the robot's location and surroundings by bumping into obstacles and even leaving the experiment arena [33, 141].

[1]American Association for Artificial Intelligence

One possible reason operators experienced poor situation awareness in the previous studies is that conventional (2D) interfaces were used to display information from the robot to the operator. Conventional interfaces make it difficult for an operator to maintain an awareness of the robot's situation because sets of related information are presented in discrete parts of the display. When related information is presented in different places, an operator must mentally correlate the sets of information, which can result in decreased situation awareness and decreased performance [35, 71, 106]. To improve situation awareness and performance, the interface should correlate and present related information in a single part of the interface, thereby reducing the operator's cognitive workload required to interpret the information.

1.2 Our Solution

In response to the need for an interface that correlates related information for the operator, we have designed a prototype 3D interface. The 3D interface integrates map, video, and robot information into a single *mixed-reality* display which renders a virtual environment based on real data and augmented with real video. The 3D interface significantly increases an operator's situation awareness in comparison to conventional 2D interfaces because a) related sets of information are combined and presented intuitively to the operator and b) the operator can see more of the environment through a larger field of view. Improvements in situation awareness and performance are manifest in a a variety of navigation and exploration tasks.

1.3 Thesis Statement

We show that a 3D prototype interface is better than conventional 2D interfaces for remote robot teleoperation by comparing the 3D interface with a conventional 2D interface in a series of user studies. The user studies focus on navigation and exploration tasks and the use of a pan-tilt-zoom camera. We identify principles that govern the success of the 3D interface over the 2D interface and we show how these principles can be applied to extend the 3D interface to other domains. We

discuss why the 3D interface is better than conventional 2D interfaces from a human factors perspective.

1.4 Overview

This dissertation will proceed as follows. In Chapter 2, we discuss previous work related to our field. This will address a variety of fields of research including robotics, human factors, psychology, philosophy, human-computer interaction, and human-robot interaction. In Chapter 3 we present a list of requirements for a useful 3D interface and we show the technology our interface uses to match the list of requirements. In Chapter 4 we present a series of user studies that compare our 3D interface with a prototype of conventional 2D interfaces in experiments involving navigation. The user studies are performed in both virtual and real world environments. Chapter 5 follows with a series of user studies that compare our 3D interface with a prototypical 2D interface in exploration tasks and discusses the use of a pan-tilt camera. These user studies are also performed in real and virtual environments. Chapter 6 will then discuss principles that helped operators perform better with the 3D interface than the 2D interface for the navigation and exploration experiments. The principles are then used to discuss extensions of the 3D interface to other domains. Chapter 7 summarizes the results of the dissertation and addresses some possible directions for future work.

Chapter 2

Previous Work

In this chapter we will discuss previous work in Human-Robot interaction and we will show how some of the most recent work has lead researchers to the conclusion that interactions between a human and a remote robot are difficult because the operator does not have sufficient situation awareness. We will then review a definition on situation awareness and show how much of the theory surrounding situation awareness is very similar to the ideas behind presence, telepresence, and virtual presence. This chapter will conclude with a discussion on techniques from virtual environments and mixed-reality research that have been used to interact with a remote robot.

2.1 Human-Robot Interaction

The field of Human-Robot Interactions covers many areas including entertainment [11, 21], museum guides [125, 17], health care [61, 99], space exploration [4], protection [60], and rescue robotics [18, 23, 54, 86]. In our research, we are focused on improving remote-robot operations—situations where the robot is distant from the operator, or *Teleoperation*.

One method to improve teleoperation is to use autonomy or intelligence on the robot. Some autonomy-based approaches to teleoperation include *shared control* [110], *safeguarded control* [40, 67], *adjustable autonomy* [10, 14, 47, 104], and *mixed initiatives* [14, 50, 65]. Safeguarded control is the ability of the robot to protect itself despite operator commands, for example, to keep from hitting walls. Adjustable autonomy is the ability to change the intelligence of a robot and adjust the interaction between the human and robot. Mixed-initiatives is the ability for

either the human or the robot to take initiative over the movement of the robot. One limitation of these approaches is that some control of the robot is taken away from the human. This limits the robot to the behaviors and intelligence that have been pre-programmed. There are situations where the operator may know more than the robot's algorithm does and it is unlikely that the robot would be "designed" to handle every possible situation. One solution to overcoming the pre-programmed nature of behaviors and intelligence is through the development of reactive robot architectures or behavior-based robotics where intelligent behavior emerges from a set of low-level primitives [5, 9, 12]. The real robot we use for some of our experiments has a simple safeguarding mechanism to protect itself from walls. However, most of our research focuses on robots that do not have higher autonomy. This is so we can focus on the fundamental aspects of human-robot interaction.

2.1.1 Interaction Methods

Fong observed that there would always be a need for human involvement in vehicle teleoperation despite any intelligence on the remote vehicle [41]. Sheridan holds similar thoughts and introduced the notion of *supervisory control* to explain how the human should be "kept in the loop" of the control of the robot [110] regardless of the level of autonomy of the robot.

There are many approaches for interacting with a robot, including gestures [56, 131], web-based controls [142, 107], and PDAs [63, 112]. Fong and Murphy have also addressed the idea of using dialog to reason between an operator and a robot when the human or robot needs more information about a situation [39, 89]. Skubic's group combined the use of a PDA and a linguistic representation to allow a novice user to draw a sketch of an environment and a path for the robot to follow through the environment on the PDA [24]. The robot uses a qualitative, linguistic representation of the obstacles around the robot to determine where it is in relation to the devised path. Most of these approaches tend to focus on different ways of interacting with a robot rather than determining which ways are more useful than others.

2.1.2 Urban Search and Rescue

Recently, rescue robotics was named as one of the grand-challenges of human-robot interactions [19]. Rescue robotics has really come to the attention of many since robots were used in the World Trade Center disaster. Casper presented a post-hoc analysis of the human-robot interactions during the robot assisted urban search and rescue response at the World Trade Center in September 2001 [23]. This was the first robot response to a real, un-staged urban search and rescue operation. Murphy observed that because only video information was available, the user interfaces were very limited and did not effectively present environment information to the operator [86]. Further, there was no spatial information about previously seen places in the world which made it difficult for operators and rescuers to comprehend and remember where the robot had been and what it had seen. This presented serious challenges for the operator to maintain situation awareness of the robot and its environment. In fact, subsequent studies by Burke et al. revealed that up to 50% of a rescue robot operator's time is spent gathering and maintaining situational information without moving the robot [18]. Part of the reason for the time spent acquiring situation awareness was because it is difficult for the operators to integrate the robot's view of the environment into their own understanding of the rescue site.

Similar results were found by Yanco and Drury in a usability study of their human-robot system with four first-responders [140]. They found that despite spending significant time acquiring situation awareness, the participants did not understand the robot's location and surroundings in the environment and even though the robot's location was presented via a map of the environment, the participants considered the map useless. Yanco et al. also analyzed results from the 2001 AAAI Robot Rescue Competition [1] where robot developers navigated their own robot systems in a mock search and rescue environment [141]. The results of the experiments show that the operators of the competition vehicles did not have sufficient awareness of the robot, its location, and its surroundings. This was in spite of the fact that the operators were the ones who built the robot and its interface, suggesting that

training and experience with the system did not help the operator's situation awareness. The authors recommend a) fusing sensor information to reduce the operator's cognitive load, b) minimizing the use of multiple windows, and c) providing more spatial information about the robot in the environment.

Poor situation awareness has been identified as a reason for operator confusion in robot competitions [33, 141] and urban search and rescue training [18]. In fact Robin Murphy suggests that "More sophisticated mobility and navigation algorithms without an accompanying improvement in situation awareness support can reduce the time spent on a mission by no more than 25 percent" [87].

2.2 Situation Awareness

In her seminal paper, Endsley defines situation awareness as "The perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future" [35]. Additionally, Dourish and Bellotti define awareness as, "...an understanding of the activities of others, which provides a context for your own activity" [27]. When applied to human-robot interactions, these definitions imply that a successful interaction is related to an operator's awareness of the activities and consequences of the robot in a remote environment. Endsley's work has been used throughout many fields of research that involve humans interacting with technology [36, 93, 135] and has been fundamental for exploring the information needs of a human operating a remote robot.

2.2.1 Presence

Similar to the notion of situation awareness are the ideas of *presence*, *telepresence*, and *virtual presence*. According to Sheridan, presence is considered "the sense of actually being at a remote or synthetic workplace which users of telerobot or virtual environment systems developed during operation of the system's human interface" [109]. This statement indicates that for an operator to have presence, they must feel as though they are actually at the remote place. Further, according

to Sheridan, "presence is a subjective sensation or mental manifestation that is not easily amenable to objective physiological definitions and measurements." These perspectives illustrate a rationalistic perspective on presence [51, 113, 116], namely that, a user has presence when they develop a sufficient mental model of the environment and they have adequately processed the perceived information.

Schloerb presents another theory of presence which develops the definition of telepresence. Schloerb begins with *physical presence* as "the existence of an object in some particular region of space and time" [105]. He then discusses how this notion of a physical presence supports *subjective presence* which is the "perception of being located in the same physical space in which a certain event occurs, a certain process takes place, or a certain person stands." This is similar to other perspectives on presence [110, 114, 116]. In addition to subjective presence, Schloerb points to the need for *objective presence* which is the need to interact with objects and is measured, according to Schloerb, by task completion.

Mantovani and Riva claim that the meaning of presence is closely linked to the concept we have of reality and that different views of reality support different criteria for presence [77]. The authors combat Schloerb's theory of presence [105] by claiming that the placement of physical presence as the base of our experience of presence is flawed because telepresence is inherently not physical, even though it does contain interactions with objects. In response to Schloerb's theory, Mantovani and Riva discuss the notion of reality as not somewhere 'outside' people's minds, it is socially constructed [43] based on the relationships between actors and their environments as mediated by artifacts.

Zahorik and Jenison discuss a view of presence based on existential philosophy and ecological psychology [143]. Notions of subjective and objective presence no longer exist. Instead, "presence is tantamount to successfully supported action in the environment", whether virtual or real, local or remote. Further, the concept of a mental representation is discarded, because after all, "how better to represent the environment than with the environment itself?" [44]. The purpose of this approach

as presented by Zahorik and Jenison is to show that the coupling between perception and action is essential for determining how well actions are supported.

Tittle et al. also provide a functional definition of presence to mean the operator receives enough cues to successfully conduct operations without requiring the sense that they are actually situated at the remote location [128].

The definitions of presence as discussed by Mantovani and Riva, Zahorik and Jenison, and Tittle et al. are similar to Endsley's definition of situation awareness in that the operator needs sufficient information to act. We follow this line of thought when working with robots, because the common reason for interacting with a remote robot is to accomplish some task.

2.2.2 Affordances

Gibson has a view on the psychology of perception that differs from traditional theories [52, 62, 127]. He contends that we do not construct our percepts, but that our visual input is rich and we perceive objects and events directly [44]. He claims that the information an agent needs to act appropriately is inherent in the environment. Gibson used the term affordance to describe the relationship between the environment and the agent. In his words "The *affordances* of the environment are what it *offers* animals, what it *provides* or *furnishes*, either for good or ill" (page 127, emphasis in original). Affordances are attractive to the robotics community because they are compatible with the reactive-based robot paradigm and they simplify computational complexity and representational issues [88]. With Gibson's ecological approach, successful human-robot interaction implies that the operator is able to directly perceive the cues from the environment that support the actions of the robot.

Norman disagreed fundamentally with Gibson's approach to how the mind actually processes perceptual information, but he did come to agree with Gibson's theory of affordances [95]. In *The Design of Everyday Things*, Norman discusses perceived affordances, which are what the user perceives they can do with some *thing* whether or not that perception is correct [94]. He claims that the goal of design should be to make affordances and perceived affordances the same. This idea

is directly applicable to mobile robots because it is necessary that information be provided that supports the operator's correct perception of available actions for the robot. Norman also advocates that the culpability of "human error" can often be attributed to "equipment failure coupled with serious design error." Therefore, in cases where an operator is performing poorly, it may be the consequence of a poorly designed system.

In Human-Robot interactions, Endsley's definition fits with Gibsonian affordances, because when information is directly perceived, it should signal to the participant how it can be used and how its use will affect the environment. The challenge is to present the information from the remote environment to the operator in such a way that the perceived affordances of the environment match the actual affordances and the operator can easily perceive, comprehend, and anticipate information from the remote environment.

2.2.3 Field of View

One of the shortcomings when navigating a robot with a conventional interface is that typical cameras have a very narrow field of view. For example, a human's lateral field of view is normally 210 degrees [3], in contrast, the camera on our robot has a field of view of only 37 degrees. The field of view that the operator has of an environment is very important to navigation. A poor field of view has been attributed to negatively affect locomotion, spatial awareness, and perceptions of self-location [3]. Further, Woods described using video to navigate a robot as attempting to drive while looking through a 'soda straw' [138]. One of the main challenges with teleoperating robots is that the operator typically does not have a good sense of what is to the 'sides' or 'shoulders' of the robot [48], and obstacles that need the most attention are typically outside of the field of view of the robot.

One method for overcoming a narrow field of view is to use multiple cameras [130, 8]. For example, Hughes et al. used two cameras and showed that it improved an operator's ability to perform a search task [55]. Another method for improving field of view is to use a panospheric camera [91, 122, 121, 139], which gives

a view of the entire region around the robot. These approaches may help operators better understand what is all around the robot, but they require fast communications to send large or multiple images with minimal delay. We are restricting attention to robots with a single camera.

2.3 Interface Design

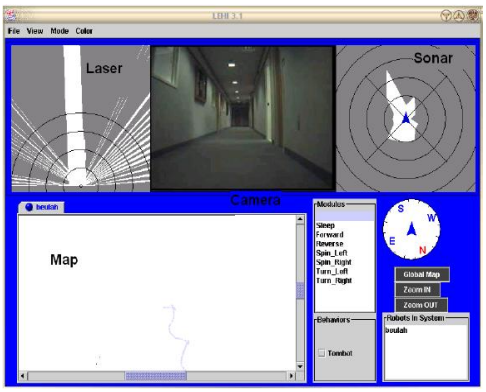
2.3.1 Conventional Approach

Most conventional interfaces used to interact with remote robots focus on the accuracy with which information is presented to the operator instead of focusing on communicating effective environmental cues. This has led to the use of displays such as those shown in Figure 2.1, which show information from the environment, but present it in distinct windows throughout the display. The disparate information leaves to the operator the responsibility of mentally combining the data into a cognitive map of the environment. This approach to interface design follows the constructivist theory of perception which claims that smaller, individual elements are combined to give perceptions [52].

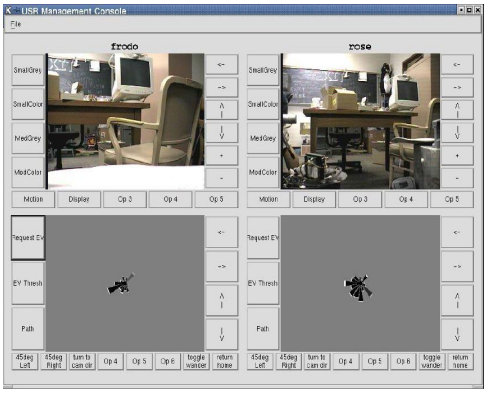
Bruemmer et al. are developing a robot system for remote operations that uses behavior-based algorithms to create a mixed-initiative human-robot team [14, 16]. Information is displayed to the operator via a typical 2D display as shown in Figure 2.1(c). Baker et al. have simplified the interface designed by Bruemmer et al. in an effort to improve the human-robot interactions as shown in Figure 2.1(d). Both interfaces present information using a conventional robot-centric approach with separate windows for different information sources. These interfaces are typically very useful for robot system development and testing as they provide enough information for the engineer to diagnose any problems that exist, but they may not adequately support an operator's situation awareness in many interesting remote environments.

2.3.2 Virtual Environments

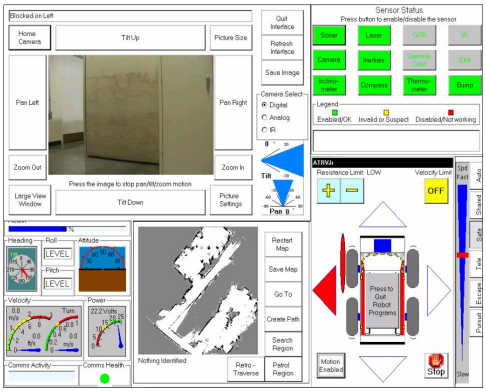
Another way to improve robot teleoperation is to use *virtual environments* to create a virtual scene that represents the real environment. The Virtual



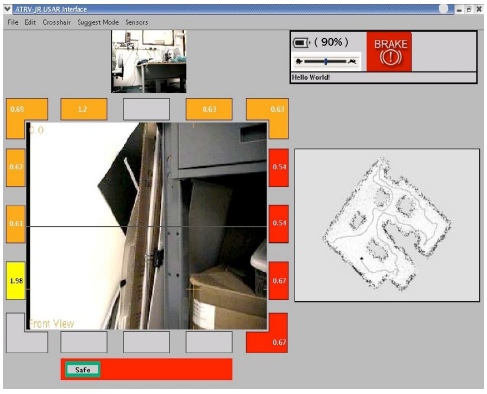
(a) Our 2D interface



(b) Adopted from [141]



(c) Adopted from [16]



(d) Adopted from [8]

Figure 2.1: Conventional interfaces present information in separate windows within the display.

Environment Vehicle Interface (VEVI) was designed by the Intelligent Mechanisms Group (IMG) at NASA Ames Research Center with the goal of supporting control and teleoperation of vehicles on remote planetary surfaces [53, 97]. The system has been used to explore volcano craters in Alaska [38] and Hawaii [117], guide underwater exploration in the Antarctic, service satellites, and direct automated forklifts [92]. The system uses stereoscopic images to create a virtual environment in which an operator can experiment with commands before sending the commands to the robot. Stoker used a similar system to analyze data and interact with the Mars Pathfinder vehicle in [118]. Nguyen et al. report that such systems are less efficient with direct teleoperation because of the high dependence on vehicle sensors. They also observe

that the system works well when the operator can first work out the commands in simulation and then issue the set of commands to the robot [92].

2.3.3 Mixed Reality

Milgram and Drascic discuss Mixed Reality displays that are a "particular subset of Virtual Reality (VR) related technologies that involve the merging of real and virtual worlds somewhere along the 'virtuality continuum' " [81]. On one side of the virtuality continuum are real environments, on the opposite side are virtual environments. Mixed reality, sometimes referred to as augmented reality [7, 83], is the domain between the two extremes. Milgram and Drascic point out that most of the work in mixed reality has been done using head-mounted displays that either provide video feedback of the real world or allow the user some direct visibility of the real world.

Milgram and Kishino present a taxonomy that addresses the real and virtual aspects of mixed reality environments [81]. The dimensions of their taxonomy include: *extent of world knowledge*, *reproduction fidelity*, and *extent of presence metaphor*. This taxonomy fits well with our needs because it splits the notion of presence into "image quality" and "immersion", in contrast to other taxonomies whose primary goal is creating an immersed presence [103, 110]. This split notion of presence provides a useful category for monitor-based mixed-reality displays [30, 37, 133], which are the tools of our research.

Drascic and Milgram found that the use of a stereoscopic display improves the user's interaction because it presents depth information directly to the user [29]. In contrast, monoscopic video images require the operator to interpret shadows and reflections to infer spatial relations [28]. Milgram and Drascic discuss the use of augmented reality as a means to overlay a stereoscopic display with virtual information to facilitate communications between a human and a robotic arm [84]. Their approach is based on the ARGOS (Augmented Reality through Graphic Overlays on Stereovideo) system [30] and combines elements such as a *virtual pointer* [31], a *virtual tape measure*, and *virtual landmarks* to help the user control the robotic arm. By gathering

stereoscopic information from the remote environment, the user is able to view a virtual 3D scene of the environment. Then, the user experiments within the augmented reality environment to determine exactly how they want to manipulate the robotic arm. Once the commands are determined, the user sends them to the remote robot and the commands are actuated. This approach is similar to the interactions with the VEVI control system [92, 97].

Meier et al. explored the possibility of using sensor fusion for making the operator more aware of the environment around a mobile robot [80]. Their display is typical of sensor fusion approaches for mobile robotics in that they overlay real video information with depth and other virtual information [8]. In this particular approach the video is from a stereoscopic camera and it is combined with sonar range information to create a colored depth map. Additionally, the image displays a projected grid which is overlaid on the ground and obstructed by above-ground obstacles. The grid cells are close to the same size of the robot to support the operator's comprehension of distances. The problem with most of these sensor fusion based displays is that even though the video is augmented with virtual information, the field of view of the environment is still limited by the field of view of the camera.

In another example of a mixed-reality display, Johnson et al. created an "EgoSphere" (a term first proposed by Albus [2]) to enhance their robot interface [59]. The EgoSphere consists of a 3D sphere around the robot on which interesting observations are portrayed. They did not find the EgoSphere to be particularly useful with a mobile robot. We believe that an EgoSphere is probably more appropriate for an augmented reality display where the operator is wearing a head-mounted display.

Suomela et al. developed a fully adjustable three-dimensional map that supports traditional two-dimensional map views and a full range of perspective views for a head-mounted display [120]. They found that a single perspective view is useful sometimes, but different participants preferred different perspectives [73]. Further, they identified situations where a "north-up" map is better than a "north-forward" map and vice versa [72]. The purpose of their development was to combine previous

map abilities into a single user-adjustable interface. This research is particularly relevant to mobile robot research because the requirements of successful navigation for robots is similar to that of humans, namely recognition and traversal of possible directions of travel (affordances) and recognition of obstacles. The 3D interface described in this paper is shown to facilitate human navigation in unknown environments, so it stands to reason that a similar interface may improve robot teleoperation in unknown environments.

2.3.4 Augmented Virtuality

Another form of mixed reality is *Augmented Virtuality*. Augmented virtuality refers to virtual environments which have been enhanced or augmented by inclusion of real world images or sensations. Augmented virtuality differs from augmented reality (another form of mixed reality) because the basis of augmented virtuality is that the environment is virtual as opposed to real [32].

In harmony with Gibson's theory of perception, Ricks et al. present ecological displays for teleoperating a mobile robot [102]. The motivation behind the 3D display is Gibson's notion of affordances and direct perception. The displays present a visually pleasing integration of range and camera information that is rendered in three dimensions, but does not require a complex 3D model or registration between real and virtual objects. The displays do not use map-building but render information from the current laser and sonar range scan as green and blue barrels. The video information is scaled and pushed deep into the display such that the range information appears in front of the video, but the video still fills most of the screen [102]. The approach is opposite to typical augmented reality approaches [8, 84, 92, 98, 141] because it builds a virtual environment based on real information and augments the virtual environment with real video—an *augmented virtuality* solution. The advantage of using an augmented virtuality solution is that the information presented to the operator has a much larger field of view. Instead of being constrained by the field of view of the camera, range information can now be used to "see" the sides of the robot and the video information can be located relative to the map information.

Ricks compared an ecological display with a conventional interface in a mobile robot navigation task and found that the ecological display improved performance while decreasing workload and reducing the number of collisions [101]. These results suggest that using a similar augmented virtuality approach could significantly improve an operator's situation awareness in comparison to typical two-dimensional interfaces including augmented reality interfaces. We have extended Ricks' work by adding map-building, a scaled robot model, an adjustable perspective, and the ability to store information in the display.

2.4 Summary

In order to significantly improve performance on a task with a teleoperated robot, an operator's situation awareness of the remote environment must be improved. Since the operator is not collocated in the same environment as the robot, the development of an operator's situation awareness must come through information visible on the user interface. Most of the current research in human-robot interaction focuses on how an operator could interact with a robot (i.e. using a PDA, gestures, the internet, a desktop computer, a head-mounted display) and what information could be useful to the operator (camera, range, map, proximity indicators, sensor status, waypoints, goals). However, the question of how the information should be presented to the operator has not been adequately addressed.

Conventional interfaces for teleoperating a remote robot do not adequately support the development of situation awareness because related information from the robot is usually presented in different parts of the display and the operator is responsible to mentally correlate the information. An interface that better supports the development of situation awareness would display related information in a single part of the display so the operator can immediately observe how different sets of information are related to each other. Additionally, the usefulness of interfaces is typically validated by subjective evaluations or by showing that it fulfills a set of requirements or can be used to accomplish a particular task or set of tasks. A stronger

validation would be to provide empirical evidence that shows which interface yields better results than another interface.

Chapter 3

The 3D Augmented Virtuality Interface

In human-remote robot interactions (HRI), the interface is the tool through which the operator visualizes the robot's environment and communicates instructions to the robot. Information from the robot's environment is gathered from sensors on the robot and transmitted to the interface. The interface renders the information from the robot's environment so the operator can visualize the remote information and make informed decisions about how to use the robot. The operator communicates instructions to the robot through an input device such as a keyboard or joystick that is connected to the interface. The interface then transmits the instructions to the robot where they are actuated.

For this research, a Pentium IV desktop computer with a 19" LCD monitor is used as the interface between the operator and the robot. Operator commands are actuated with a Microsoft Sidewinder joystick or steering wheel which are received and interpreted by the interface computer and sent to the robot over 900 MHZ wireless modems or interprocess communications[1]. Information from the robot's environment is gathered from range and camera sensors on the robot and transmitted to the interface via the wireless modems and an 802.11g wireless network. Information received by the interface computer is rendered on the computer monitor using software including the OpenGL graphics library. Henceforth, when we refer to the *interface* we are referring to the program on the desktop computer that displays information from the robot's environment on the computer monitor. Requirements and technology for

[1]When a simulator was used for driving the robot, the interface and the simulated robot where run on the same computer. When a real robot was used, the interface and robot were separate computers.

designing and building a useful interface for human-robot interactions are presented throughout this chapter.

3.1 Requirements

For a display to be considered useful and effective, we require three features that have been identified by experts in human-robot interaction. First, the interface must allow the user to store information in the display [106, 140]. Second, the interface must integrate sensor information into a single coherent display [33, 42, 78, 110, 140]. Finally, it must allow the user to adjust their perspective of the environment to match the needs of the operator and the task at hand [106, 120, 136]. While each of these aspects is important, it is also necessary that they are easy to use. We will next discuss these three requirements for a useful interface and show how conventional interfaces for teleoperating robots typically do not support these features.

3.1.1 Information Storage

In tasks where an operator is required to remember where objects are, or what was happening at various places in the environment, it is typically left up to the operator to remember the information. As the complexity of such a task increases or the amount of information that must be remembered increases, it quickly becomes likely that the operator will forget some information or their recollection will deteriorate. To reduce the memory requirements on the human operator we require an interface that facilitates information storage.

In conventional remote-robot interfaces, information storage is typically not implemented. However, when it is implemented, the approach is to use small icons such as an 'X' or an 'O' and overlay these on the map. The problem with this approach is that it is difficult for the operator to remember the meaning of the icons and multiple, collocated icons become confusing to the operator [111, 136]. Further, it is usually not practical to store annotations, images, or full video information directly in the display even though it may be advantageous to associate user-customizable entries with the map.

3.1.2 Integrate Information

To accurately store information in the interface that correlates with the remote environment and to make the information easily accessible to the operator, we require that the information from the robot and the environment be integrated into a single display. The information from the robot includes many things including robot status, video, laser readings, sonar data, map information, position data, snapshots, landmarks, and icons. Interfaces that do not integrate information force the user to mentally combine the different sensory information into a cognitive understanding of the robot's situation and its environment. In contrast, an integrated display presents the user with a view of the environment that combines the relevant information into a single display such that the cognitive information processing required to interpret the information is reduced [48, 102].

A typical approach to providing information to the operator is to put the different sets of information in different windows and to provide the operator as much information as is available. This approach is typified by the interface in Figure 3.1(a) from [14, 16]. It is also customary to present a subset of the available information as shown in Figure 3.1(b) from [8, 111, 136]. These displays are typical of what might be found for robot teleoperation interfaces [8, 16, 23, 41, 42, 141] and represent what we refer to as *conventional 2D interfaces*. These interfaces are very useful for testing and debugging a robot system and assuring that the system is behaving appropriately. The main problem with these displays is that information, even related information, is not integrated, but is presented in different places on the screen which makes it difficult for an operator to gain a holistic understanding of the robot within its environment [66, 129, 136].

A more advanced approach to integrating information is the notion of sensor fusion. This work has typically been done with an augmented reality approach wherein video information from the robot is augmented with range information [80] or other information to better understand the environment [8]. There are two main deficiencies with this approach. First, the video image has typically been gathered from stereoscopic cameras, which means there is a high computational workload to

(a) (b)

Figure 3.1: Typical interfaces for robot teleoperation as developed by (a) INL and (b) Baker and Yanco.

determine the 3D scene and communicate it to the operator. Second, the field of view of the environment is limited to that of the camera because it is the video stream itself that is augmented with range information. Others have looked at using panoramic cameras, but these systems also require large bandwidth for sending the panoramic video to the operator.

In systems that use augmented reality interfaces, it has been found that the best method of interacting with the robot is to create a complete virtual environment from the information, then practice the commands the operator would like the robot to perform in the real world. Once the commands are learned, they are sent to the robot to be actuated [82, 118, 119]. Nguyen et al. observed that augmented-reality systems are better for high-level task planning control and less efficient for the direct teleoperation tasks we are interested in [92].

3.1.3 Adjustable Displays

Information storage and integrated displays are important concepts for reducing the mental workload of the user. In addition, research has shown that certain displays are better suited to certain tasks [136]. Scholtz observed that the roles of human operators do not remain constant and, therefore, interfaces should be designed

to adapt accordingly [106]. For example, when performing navigational tasks, an egocentric perspective is typically preferred to an exocentric perspective. However, when performing spatial reasoning tasks such as path planning, an exocentric perspective is preferred [136].

Conventional interfaces, as presented previously (see Figure 3.1), do not support adjustable perspectives. While it could be argued that the video information and the map information suffice for most navigational and spatial needs [136], it is reasonable to expect that some perspective between the two may also prove beneficial for navigating, exploring, or manipulating parts of the robot. Some augmented reality approaches have developed adjustable displays, but these are similar to the ones described earlier [82, 92, 118, 119] which generate a complete 3D model of the remote environment and allow the operator to experiment with possible commands. One successful example of an adjustable display was presented by Suomela et al. where they developed an augmented reality interface that presents a fully adjustable map to an individual wearing a head-mounted display. One observation they had was that users had preferences on the perspective with which they viewed the map. This personal adaptation to fit needs is another advantage of an adjustable display.

3.2 Technology

With the requirements for useful displays set forth, we next present the technologies we developed for useful displays along with the philosophies behind the various technologies.

3.2.1 Information Storage

Transactive Memory

In order to discuss the implementation of information storage within a display we first look at the cognitive science notion of transactive memory. Transactive memory is a term that was first introduced by Wegner as the "operation of the memory systems of the individuals and the process of communication that occur within the group" [134]. In Wegner's definition, he is referring to individuals as the storage

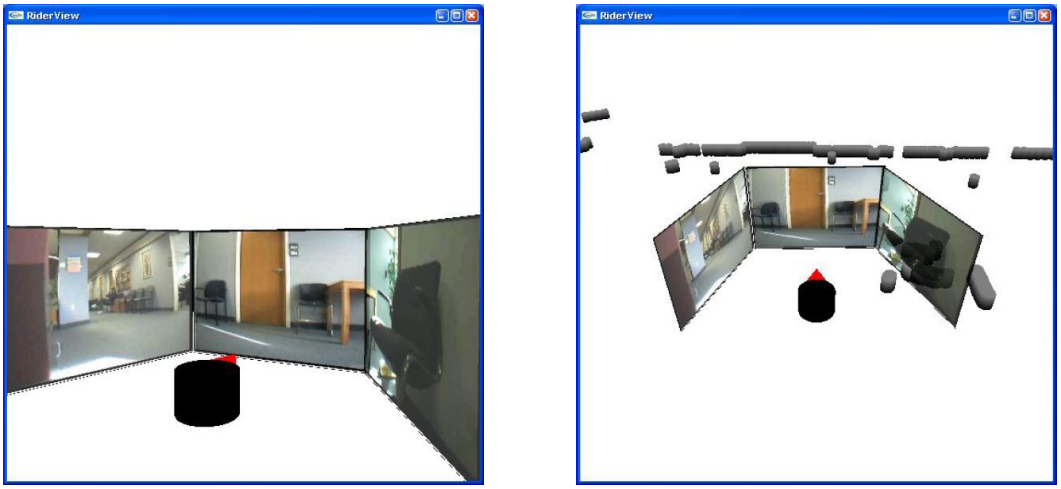


Figure 3.2: Using snapshots to remember information.

container for this transactive memory. When someone has expertise in a field, then a good friend of that individual can have access to the information by asking their friend as opposed to remembering everything on their own. Thus, transactive memory is a form of external memory. It is well known that people use external memory for a variety of common memory tasks from appointments to shopping lists to daily events recorded in a journal [49, 79, 134]. Examples of places where information is stored in external memory include such things as a PDA, a calender, or even a scratch piece of paper. In order to use these forms of external memory, it is important to have a storage device in place that is easy to access. Then, the person desiring to find the information does not have to remember the details of the information itself, just where to find it. This frees the person's mind to focus on other tasks. Similarly, information available to an operator in a human-robot team can be overwhelming unless the user has a means to store the information in an easily accessible manner. One method we have developed for storing information in the interface is via snapshots.

Snapshot Technology

The idea behind snapshots is that visual images contain a lot of information that is understandable by a human, but not necessarily a computer. In human-robot tasks that involve object recognition and recollection, it is important to aid

the user by storing relevant information in the interface, rather than forcing the user to mentally store the information. Consider the case of navigating a robot through an environment looking for objects. Suppose that, at the end of the navigation, the operator is required to tell an administrator, for example, where all of the blue boxes in the environment are located. If the environment is sufficiently large, the operator will likely forget where some of the objects were located. To aid the user in search and identification tasks, we have created snapshot technology.

Snapshots are pictures that are taken by the robot and stored at the corresponding location in a map. In Figure 3.2 we show some snapshots taken from the robot. In the figures, the robot took three pictures from three directions. The pictures are used to show a panoramic view of the visual information around the robot. To take a snapshot, a user indicates the request via a button on the joystick. Upon receiving the user's request, the robot saves the current image along with the position and orientation of the robot when the picture was taken. The snapshot information along with the recorded pose of the robot is then returned to the interface and displayed at the corresponding location and orientation in the operator's view of the map. In search or identification tasks, the snapshots in the display are an implementation of the aforementioned transactive or external memory. By adding the snapshots to the user's perspective of the map, we make the visual information available to the user whenever they need more information about a corresponding place in the environment.

As an example of the usefulness of snapshot technology consider the following. Suppose that part way through a patrolling task a supervisor asks if the operator has seen anything suspicious. If the interface does not support snapshots, the user will have to remember if they observed something, what it was, and where it happened, or they will need to revisit the place of interest. In contrast, by empowering the user with the ability to record information directly into the display, the necessary information is already correlated with the map of the explored environment. This makes the recollection of a previous experience very accessible to the operator. Thus,

snapshots provide one method to store information within a display. The introduction of snapshot technology leads us to a broader external storage medium, namely semantic maps.

Semantic Maps

A relatively new approach to information storage is semantic maps. Semantic maps can be thought of as a map of an environment that is augmented by information that supports the current task of the operator. Semantics simply gives meaning to something; therefore, a semantic map gives meaning to places on the map. The information that is stored in a semantic map might include snapshots, landmarks, icons, laser readings, sonar data, map information, or video. As an example, consider an occupancy grid-based map. The map by itself does a good job of portraying to the operator where the robot can and cannot go. However, with such a map, the user and robot will have difficulty understanding where "Bob's chair" is located, or how to move to "Mike's Door." It is virtually impossible for the robot to learn where Bob's chair is without any user input. By placing semantic information into the map and tying it directly to places in the environment, the human and robot are able to reason about the environment semantically.

Principles of semantic maps have been addressed previously by other researchers. Most notably, Kuipers introduced the notion of a spatial semantic hierarchy as a model of large-scale space with both quantitative and qualitative representations. The model is intended to serve as a method for robot exploration and map building and a model for the way humans reason about the structure of an environment [70, 69]. Additionally, Chronis and Skubic have presented a system that allows a user to sketch a map and a route for the robot to follow on a PDA [24]. This map and path are an example of a semantic map where the map made of obstacles is augmented with route information which gives the user an understanding of what the robot will be doing.

3.2.2 Information Integration

With the ability to store information inside the interface via snapshots, landmarks, and icons, we next look to the requirement of integrating information into a single display. The challenge with creating a single display is to combine the important information that represents the remote environment into a single display that is intuitive and supports efficient interaction with the remote robot [44, 110, 143].

To do this, we use an augmented virtuality display that combines information from the map, robot, and video into a single display as shown in Figure 3.3. The dark rectangles represent walls or objects identified by a mapping algorithm[2] and a model of the robot is rendered at its current location with respect to the discovered map. The mapping algorithm creates an occupancy grid that represents whether each cell in the grid is occupied in the world. This is not a 3D mapping algorithm, but rather 2D information that is portrayed in 3D. The height we give to the 3D map corresponds to the height of the laser range scanner and is used to make the obstacles more obvious to the operator. The robot model is also scaled to match the size of the actual environment, thereby enabling the user to comprehend the relative position of the robot in the real environment.

A texture-mapped plane with the video stream is rendered a small distance in front of the robot, perpendicular to the orientation of the robot [102]. As the robot moves through the environment the visual information displayed by the texture map is updated with the most recent camera image. Furthermore, the video is somewhat transparent so information behind the video can still be seen. To inform the operator of the pan and tilt orientations of the camera, we display the video stream perpendicular to the camera's orientation and at an angle relative to the real robot as shown in Figure 3.4.

As discussed earlier, an important feature of an interface is the ability to store information. When a snapshot is taken, it is saved as a texture and mapped to a plane at the location from which the picture was taken. The snapshot looks similar to the video but does not update as the robot moves. In addition to snapshots, we have

[2]The mapping algorithm was developed by Konolige at the Stanford Research Institute [64].

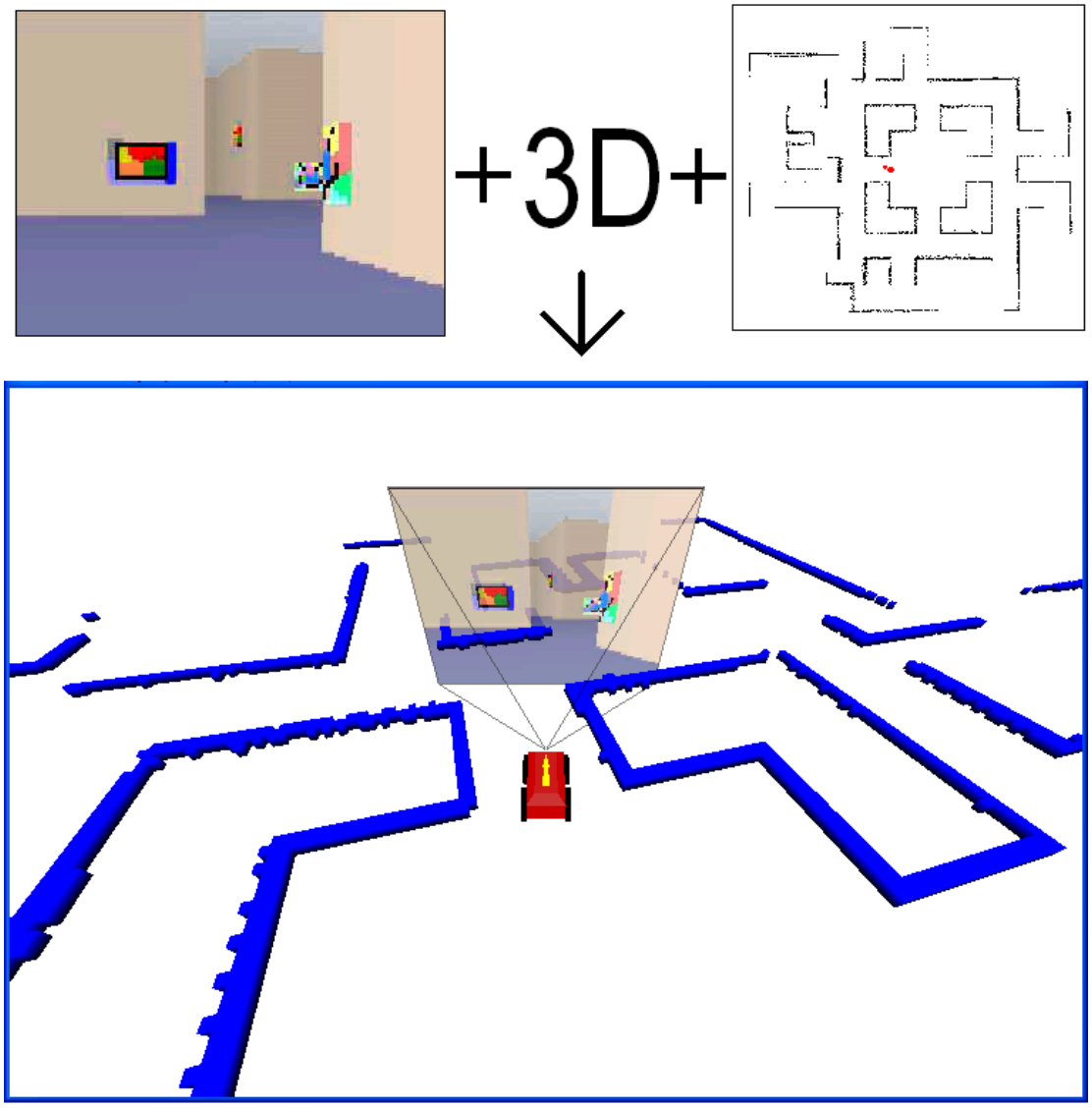
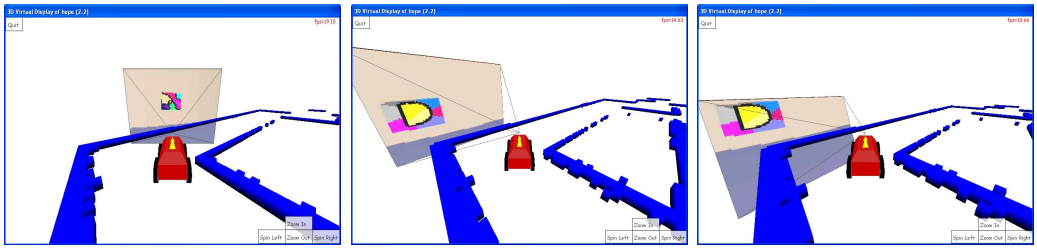


Figure 3.3: Combining two dimensional video and map into a three dimensional, mixed-reality display.



(a) (b) (c)

Figure 3.4: Perspectives of the camera orientation (a) normal, (b) panned left, and (c) tilted down.

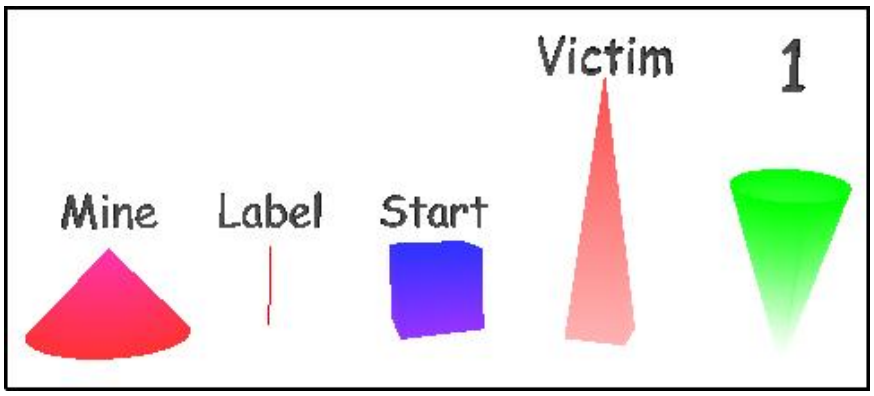


Figure 3.5: Some icons, landmarks, and labels.

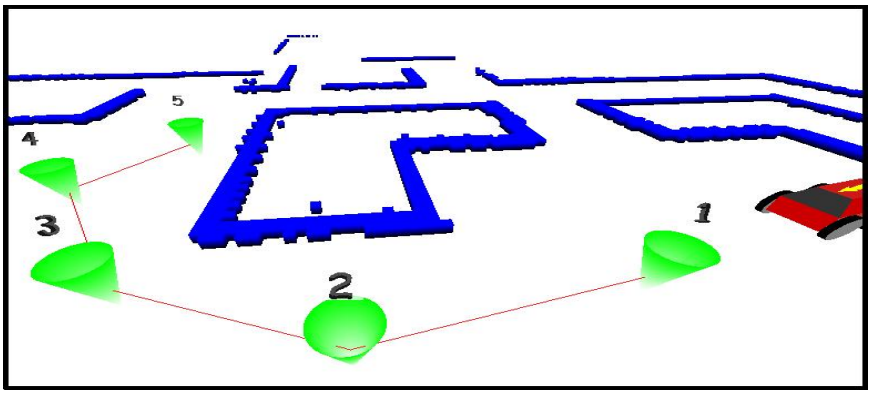


Figure 3.6: Waypoints used to designate a desired path for the robot to follow.

the ability to store landmarks or icons in the virtual environment using the mouse. Some example icons may include "start", "victim", "land mine", "waypoint", and "label" icons as shown in Figure 3.5. The label icons contain user editable text that describes the thing or place visited, such as "Mike's office", "Anna's chair", or "Bob's desk" which ties meaning to a place in the environment. The user can also edit the title on the other icons with the exception of the waypoint icon.[3] As waypoints are placed in the environment with the mouse, they are numbered in the order that they are placed, making it easy for the operator to know the path the robot should traverse (see Figure 3.6).

[3]A decision was made to only allow numbers on waypoint icons because waypoint icons disappear when they have been visited by the robot and are therefore more temporary than the other icons.

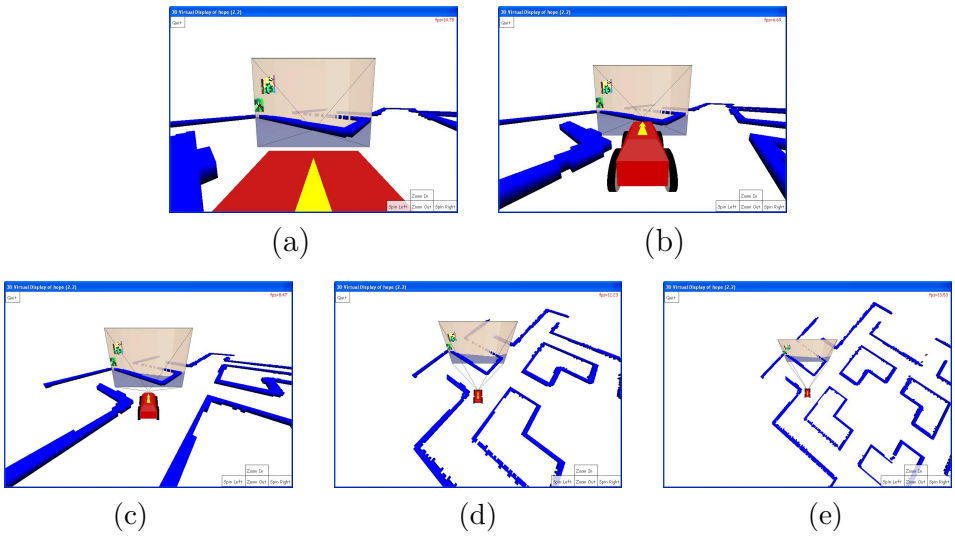


Figure 3.7: Some different perspectives of the virtual environment: egocentric (a, b) to exocentric (c, d, e) perspectives.

This augmented virtuality interface provides an intuitive integration of most of the information necessary for human-robot interactions.

3.2.3 Adjustable Perspective

In addition to displaying the relevant information in a useful way, we also desire that the interface be able to support the user in a variety of tasks. The 3D interface we created supports a dynamic, adjustable interface by allowing the operator to zoom in the virtual perspective towards a robot's egocentric view of the environment and to zoom out to obtain an exocentric perspective or map-view. Additionally, we allow the operator's observation of the world to be disconnected from the robot so that the user can revisit the environment without moving the robot. Some of the possible perspectives of an environment are shown in Figure 3.7.

3.3 Summary

In this chapter we have presented requirements and technology for creating a useful interface for teleoperating a remote robot. The requirements are that the interface must support a) storing information, b) integrating similar information into

a single display, and c) adjusting the perspective through which the operator views the robot's environment. A 3D augmented virtuality interface that fulfills these requirements for a useful display has been described. The next two chapters present user studies that quantify the value of the 3D interface over the 2D interface in certain navigation and exploration tasks.

Chapter 4

Navigation User Studies

One of the fundamental aspects of robot teleoperation is the ability to successfully navigate a robot through an environment. We define successful navigation to mean that the robot avoids obstacles and arrives at a destination in a timely manner. In this chapter we present a series of user studies designed to evaluate an operator's ability to navigate a robot via a conventional 2D interface and the 3D augmented-virtuality interface proposed in Chapter 3. For the remainder of this discussion we will refer to the conventional interface as the *2D interface* and we will refer to the augmented-virtuality interface as the *3D interface*. All of the user studies were performed by novice operators who have had minimal experience with robots.

The user studies begin by reviewing a study by Ricks and continuing with studies performed as part of this research. In Ricks's experiment, the robot is navigated along a path that is pre-determined and conveyed to the operator with instructions over a headset [101]. The next study looks at an operator's ability to discover the physical structure of an environment by building a complete map of the environment. The following study compares the robustness of the interfaces to network delay between the robot and the operator. The penultimate study addresses the usefulness of video and map information on a navigation task. The final experiment in this chapter addresses the navigation of an ATRV-Jr Robot in a real environment. We conclude this chapter with a discussion of the implications of the results of these experiments.

4.1 Path-Following Experiment

In the Human-Centered Machine Intelligence (HCMI) lab at BYU, when we first started using a virtual 3D representation to visualize how information from a robot could be displayed to make teleoperation easier, we looked at two approaches. One approach was to present a local perspective of the immediate environment around the robot based on the raw information from the most recent sensor readings (laser, sonar, video). The other approach was to present a global perspective of the environment based on interpreting and combining the raw information into global information (map, robot pose). Ricks focused on the local perspective of the immediate environment and the work presented in this dissertation focuses on the global perspective of the environment. At the time, both approaches were implemented in simulation, but the global map-based approach could not be implemented in the real-world because we did not have a decent map-building algorithm.

The first navigational experiment comparing a conventional 2D interface with an ecological 3D interface at Brigham Young University was done by Ricks using a local perspective of the immediate environment [101]. The experiment design and results are presented as they are related to the experiments we present. For a more detailed report on the findings see [101, 102].

4.1.1 Framework

In this experiment, participants were asked to navigate a simulated robot or a real Pioneer 2-DXE robot along a pre-planned path through a maze environment as fast as they could. The path was given to the operator as a series of red dots in the camera images plus verbal instructions about what to do when the robot reached the next dot. In addition to completing mazes with the robot, a memory task was devised to try and determine the amount of working memory required to use each interface effectively. The two interfaces compared by Ricks are a conventional 2D interface that had been previously used at the HCMI lab and the ecological (3D) interface developed by Ricks (Figure 4.1).

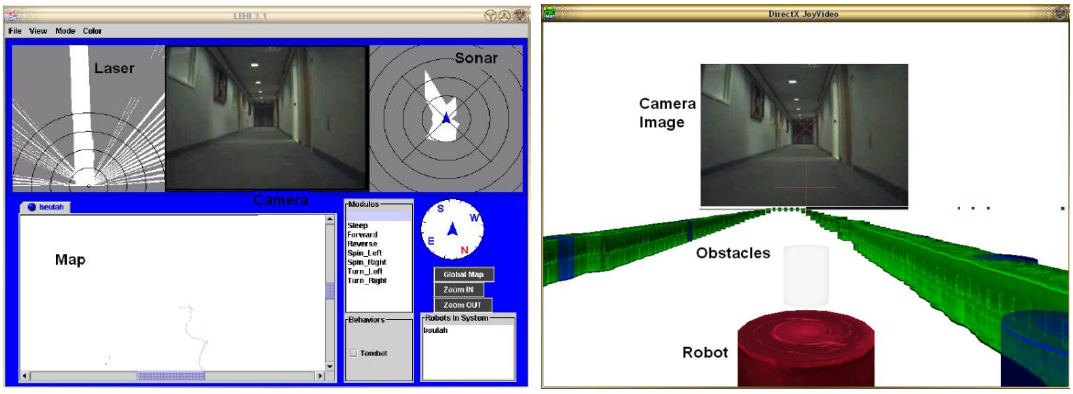


Figure 4.1: The interfaces used by Ricks in the path-following experiment. The left image is the 2D (conventional) prototype and the right image is the 3D (ecological) prototype. Images are from [101].

4.1.2 Results

Thirty-two participants completed the experiment in simulation and 8 completed the experiment with the real robot. The results show that, on average, participants did much better with the 3D interface. Two of the most notable differences are the number of people that crashed using each interface and behavioral entropy.[1]

In simulation, there were over 87% fewer collisions using the 3D interface than the 2D interface. Additionally, more people were able to complete the tasks in different environments without crashing at all using the 3D interface. Behavioral entropy was also 31% lower with the 3D interface. Participants were able to complete the task an average of 15% faster, and their average velocity increased approximately 9% when using the 3D interface in comparison to the 2D interface (see Table 4.1 from [101]).

With the real robot, participants took less than half the time to complete the task with the 3D interface than with the 2D interface. Further, participants drove almost twice as fast and had 93% fewer collisions with obstacles when using the 3D interface than when using the 2D interface (see Table 4.2 from [101]). Additionally, behavioral entropy was 23% lower with the 3D interface than the 2D interface.

[1]Behavioral entropy was first developed to estimate driver workload in an automobile driving context [90]. Later it was used to measure human workload in HRI domains [46]. The metric utilizes operator activity to estimate human workload.

	2D Interface	3D Interface	% Change	p-value
Time to Completion(s)	249	212	-15%	8.5×10^{-6}
Average Collisions	7.4	0.94	-87%	2.2×10^{-4}
Average Velocity (m/s)	0.41	0.45	9.3%	2.3×10^{-5}
Average Memory Task	98.53	98.04	-0.50%	4.9×10^{-1}
Average Entropy	0.519	0.358	-31%	3.8×10^{-15}

Table 4.1: Objective results from the simulation portion of the path-following experiment (from [101]).

	2D Interface	3D Interface	% Change	p-value
Time to Completion (s)	553	270	-51%	4.8×10^{-3}
Average Collisions	10.4	0.75	-93%	5.5×10^{-3}
Average Velocity (m/s)	0.14	0.26	86%	7.8×10^{-4}
Average Memory Task	85.88	95.86	12%	4.2×10^{-2}
Average Entropy	0.509	0.393	-23%	3.6×10^{-2}

Table 4.2: Objective results from the real world portion of the path-following experiment (from [101]).

The subjective evaluations also consistently rate the 3D interface higher than the 2D interface. Ricks observed that participants felt that navigation with the 3D interface was more easily learned, required less effort, and preferred over the 2D interface in both the simulation and real world experiments [101].

4.2 Map Building Experiment

In the path-following experiment, the operator had to rely on current sensor readings and instructions from a headset to navigate the robot successfully towards the goal because there was no global representation of the environment. With a laser range finder and an appropriate simultaneous localization and mapping (SLAM) algorithm, a robot can build a map of an environment as the robot explores the environment [64, 124]. The purpose of this map-building experiment is to compare the usefulness of a 2D and 3D interface in a task where the operator is required to discover the physical structure of an indoor environment. We hypothesized that

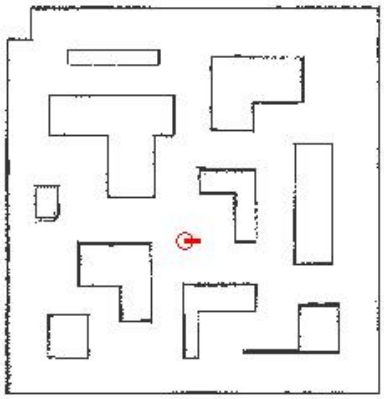


Figure 4.2: The environment used for the map-building experiment.

with the 3D interface, an operator would be able to more quickly build a map of the environment.

4.2.1 Framework

To perform the map-building experiment, we increased the robot's abilities by using Konolige's map-building algorithm to construct a map of the environment from laser range scans and the movement of the robot [64]. The constructed map depicts the location of obstacles using an occupancy grid and shows the location of the robot with respect to the obstacles.

The simulated environment that the operators were asked to explore is a simple box-shaped environment with numerous obstacles of varying shapes and sizes as shown in Figure 4.2. The distance between walls in this environment is at least 2.0 meters and the radius of the robot is about 0.6 meters. In contrast to Ricks's experiment, the task and environment do not dictate a prescribed exploration path, which leaves to the operator the responsibility of a) recognizing where the robot has and has not been, and b) making their own decisions on how to move the robot in order to visit the entire environment.

For this experiment, video, map, and robot pose (position and orientation) information are presented to the operator in order to make the interface as simple as possible. Because of the limitation on the information presented to the operator and

the existence of map-building, we used different prototype interfaces than the previous experiment. The 2D interface that we used is designed to present information similar to conventional interfaces [8, 16, 141]. The difference is that we limit the information presented to only video, map, and robot pose. This is done to minimize information that may or may not affect navigation.

The prototypes of the 2D interface and the 3D interface used for this experiment are shown in Figures 4.3 and 4.4 respectively. The 3D interface differs from Ricks's experiment to better support map-building and recognition of the robot's location within the map. In the 3D interface, readings from the laser and sonar sensors are not explicitly shown, instead we show the map which is an interpretation of the laser readings. In particular, since an occupancy grid-based map-building algorithm is used, we represent occupied grid-cells with a blue box (instead of a barrel). To illustrate the robot with respect to the map, a simplified 3D model of the robot is used instead of a red barrel. Similar to Ricks's experiment, the perspective from which the operator views the virtual world is slightly above and behind the virtual robot. Both the 2D and 3D interfaces use the same simulated robot system, with the same available information; the only difference is the manner in which the information is presented to the operator.

To control the robot, participants used a Microsoft Sidewinder steering wheel (Figure 4.5). Participants were verbally instructed on the use of the steering wheel and pedals prior to operating the robot. Participants were informed that their task was to discover the structure of the simulated environment and that the task would be complete once they had built the entire map.

4.2.2 Results

This experiment took place as a special exhibit in "Cyberville" at the St. Louis Science Center between April 30th and May 5th, 2005. Participants were visitors to the science center who came from local schools and colleges. There were 60 individuals who participated in this experiment between ages 9 and 51 with an average

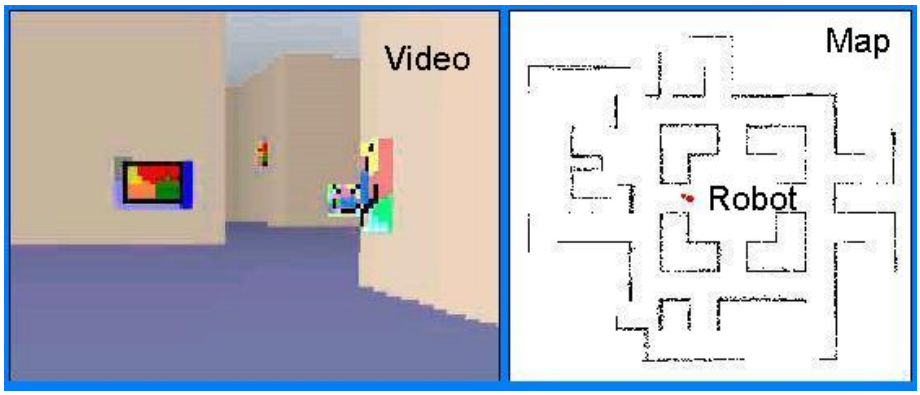


Figure 4.3: The 2D prototype interface used for the map-building experiment.

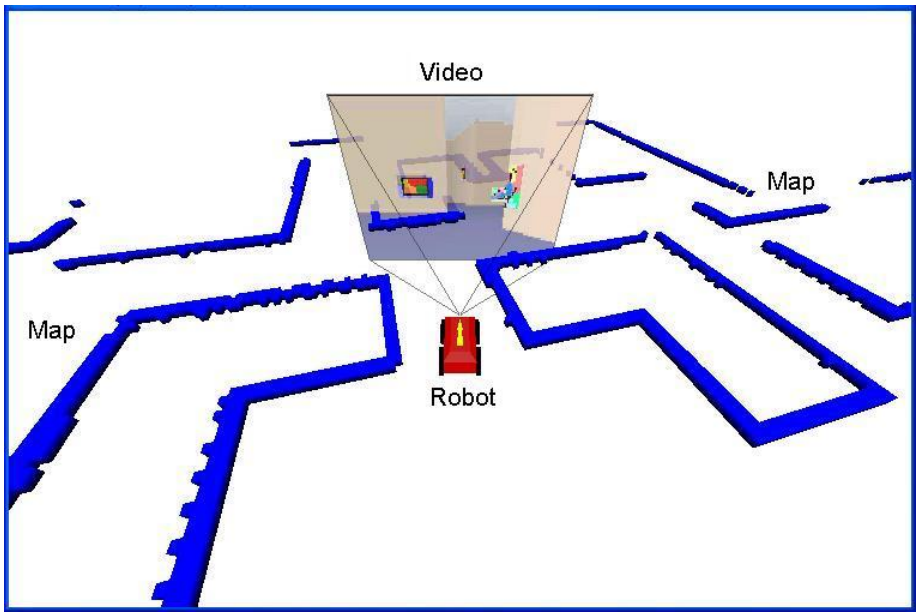


Figure 4.4: The 3D prototype interface used for the map-building experiment.

Figure 4.5: The steering wheel used for the map-building experiment.

and median age of 18 years. Thirty of the participants drove the robot with the 3D interface and the other thirty participants drove the robot with the 2D interface.

Data collection began as soon as the operator started driving the robot and ended when the operator completed the task or the robot became stuck in a wall and could not be extricated.[2] The data has been trimmed so that any delays before moving the robot and any delays upon completion of the experiment are removed. Furthermore, in the instances where the operator was unable to extricate the robot following a collision with a wall, we trimmed the data immediately following the first instance of the final collision. This limits our data to the time between the operator's beginning and final movements of the robot.

Throughout the experiment there were many instances when an operator drove the simulated robot into a wall and was unable to extricate the robot and therefore unable to complete the map-building task. Of the participants, 9 (30%) were unable to complete the task with the 3D interface and 17 (57%) were unable to complete the task with the 2D interface because of collisions with walls. Throughout the remainder of this discussion, significance levels are determined using a two-tailed unequal variance t-test with n = 13 samples with the 2D interface and n = 21 samples with the 3D interface.[3] For those who completed the experiment, the average time for completion was 34% faster with the 3D interface than with the 2D interface ($\bar{x}_{3D} =$ 178s, $\bar{x}_{2D} =$ 272s, $p = 3.4 \times 10^{-4}$). Additionally, on average, operators drove the robot 69% faster with the 3D interface[4] than with the 2D interface. ($\bar{x}_{3D} = 1.16$m/s, $\bar{x}_{2D} = 0.69$m/s, $p = 1.6 \times 10^{-5}$).

We also observed that, on average, there were 66% fewer collisions with the 3D interface than with the 2D interface ($\bar{x}_{3D} = 5.1$, $\bar{x}_{2D} = 14.9$, $p = 2.6 \times 10^{-4}$). Additionally, the robot maintained an average distance 16% further from walls (as

[2]There was a small problem in the simulator where the robot could be driven into a wall but not be extricated from the wall. This happened most often when a collision occurred while the robot was being driven backwards.

[3]The notation $\bar{x}_{condition}$ is used to indicate the average result for a particular condition.

[4]The observation that the operators maintained an average velocity 69% faster but only finished the task 34% faster with the 3D interface may seem confusing. The reason for this is that since there was no designated path that the robot must take, operators most likely drove the robot a further distance through the environment before finishing the task with the 3D interface.

	2D Interface	3D Interface	% Change	p-value
Time to Completion (s)	272	178	-34%	3.4×10^{-4}
Average Velocity (m/s)	0.69	1.16	69%	1.6×10^{-5}
Average Collisions	14.9	5.1	-66%	2.6×10^{-4}
Nearest Obstacle (m)	0.74	0.85	16%	4.2×10^{-3}

Table 4.3: Results from the map-building experiment.

measured by the distance to the closest obstacle) with the 3D interface than with the 2D interface ($\bar{x}_{3D} = 0.85$m, $\bar{x}_{2D} = 0.74$m, $p = 4.2 \times 10^{-3}$). The results from the map-building experiment are summarized in Table 4.3 and Figure 4.6.

4.2.3 Discussion

One useful measure of navigation-relevant situation awareness is the percentage of time the operator navigated the robot in close proximity to walls. To create this measure, we create the cumulative distribution function of the amount of time that the operator spends driving within a given distance of any wall divided by the total time the operator navigated the robot. Therefore, the first data point specifies the percentage of time that the robot is touching a wall, the second data point specifies the percentage of time the robot is within 10cm of a wall, the third data point represents the percentage of time the robot is within 20cm of a wall, and so on. To analyze this data, participants are divided into four groups, based on whether or not they finished the task and which interface they used. The four groups of users are:

- those that used the 3D interface and completed the task,
- those that used the 3D interface and crashed the robot,
- those that used the 2D interface and completed the task, and
- those that used the 2D interface and crashed the robot.

The results of the proximity analysis are shown in Figure 4.7.

This graph shows some interesting trends. First, the results suggest that with the 2D interface, operators tend to spend a larger percentage of their time

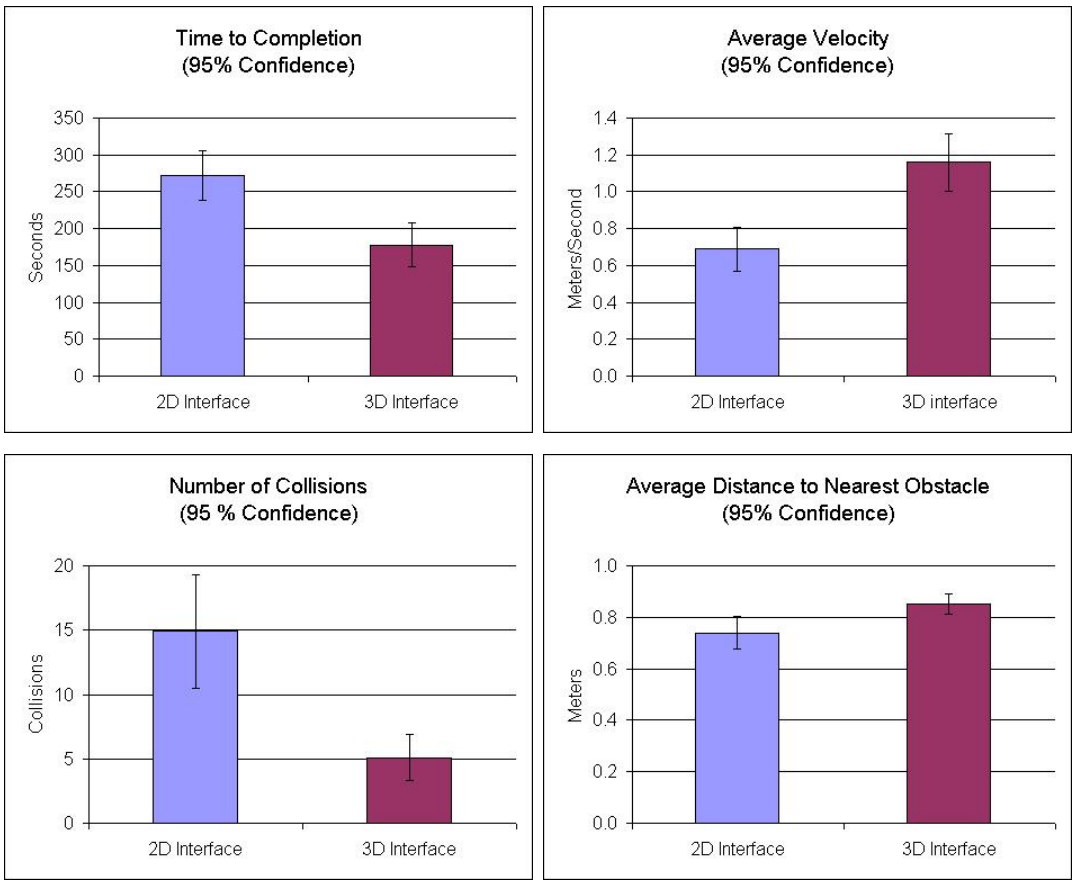


Figure 4.6: Results from the map-building experiment. Clockwise from bottom left: number of collisions, time to completion, average velocity, and average distance to nearest obstacle.

navigating the robot while it is actually touching a wall than with the 3D interface ($\bar{x}_{3D} = 5.1\%$, $\bar{x}_{2D} = 14\%$, $p = 8.6 \times 10^{-4}$).

Secondly, the results show that with the 2D interface, a larger percentage of navigational time is spent with the robot closer to walls than with the 3D interface. For example, the robot is within 40cm of a wall 15% of the time with the 3D interface in comparison to 32% of the time with the 2D interface (n = 30, $p = 3.0 \times 10^{-6}$). In fact, in this experiment we found that with the 3D interface, the percentage of time that operators spent navigating the robot within 40cm of a wall is similar to the percentage of time that operators with the 2D interface spent navigating the robot while it was touching a wall ($\bar{x}_{3D_{40cm}} = 15.0\%$, $\bar{x}_{2D_{0cm}} = 14.5\%$, $p = 0.872$). When coupled with the average number of collisions and the number of participants who

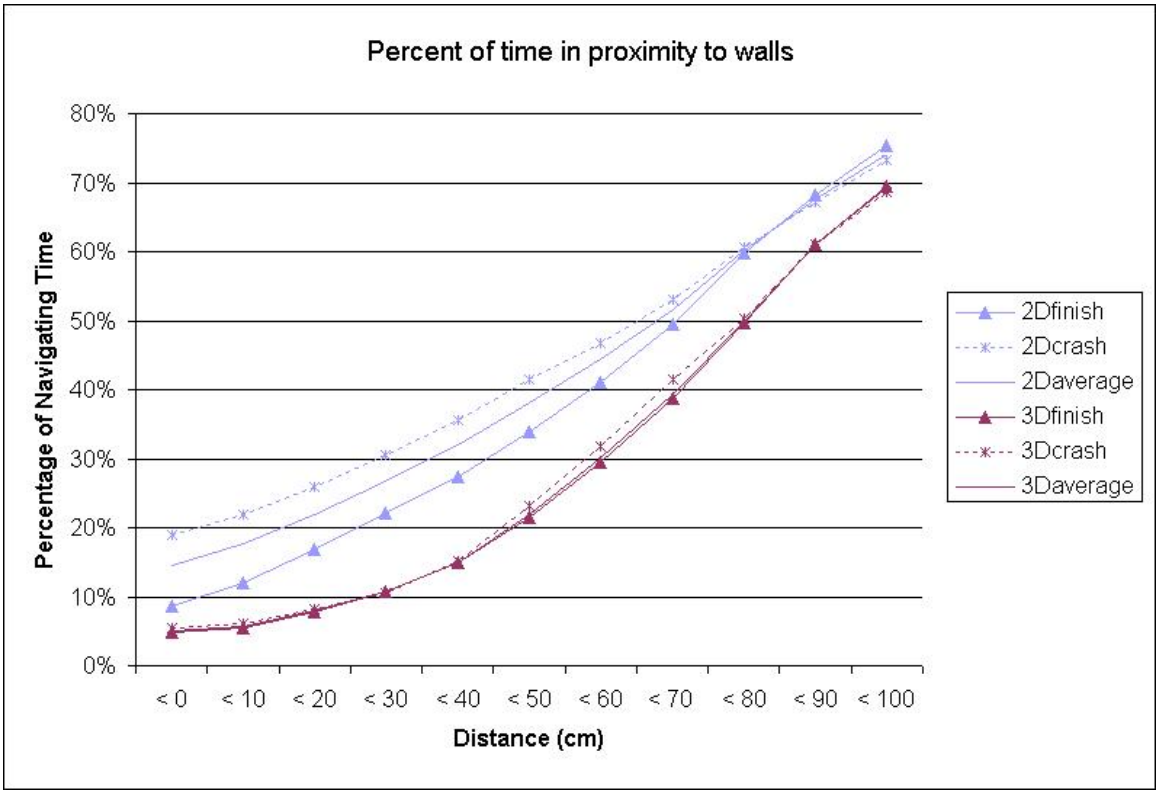


Figure 4.7: Percentage of time the robot is navigated while in proximity to obstacles.

could not complete the experiment, these results suggest that operators are not as aware of the robot's proximity to obstacles with the 2D interface as they are with the 3D interface.

Third, the percentage of time the robot is in contact with a wall is similar for the operators that crashed the robot and those that completed the task with the 3D interface ($\bar{x}_{3Dfinish} = 4.7\%$, $\bar{x}_{3Dcrash} = 5.5\%$, $n_{finish} = 21$, $n_{crash} = 9$, $p = 0.767$), yet there is a significant difference between those that crashed the robot and those that completed the task with the 2D interface($\bar{x}_{2Dfinish} = 8.6\%$, $\bar{x}_{2Dcrash} = 19\%$, $n_{finish} = 13$, $n_{crash} = 17$, $p = 2.1\times10^{-2}$). The difference between those who completed the task and those who crashed in 2D suggests that there is a difference in skill among the participants and the more skilled participants kept the robot further from walls better than less skilled participants. With the 3D interface, however, there seems to be very little difference between the performance of the participants who crashed the

robot and the participants who completed the task, which suggests that less skilled participants were able to perform similarly to more skilled participants.

4.3 Information Usefulness Experiment

In order to support an operator in navigational tasks it is important to present navigation-relevant information to the operator. In remote mobile robot navigation, it is common to use video and/or range information to inform the operator of obstacles and available directions of travel [8, 16, 41, 42, 80, 141].

Both video and range information provide distinct sets of information that have advantages and disadvantages for navigation tasks. For example, a video stream provides a visually rich set of information for interpreting the environment and comprehending obstacles, but it is usually limited by a narrow field of view and it is often difficult to comprehend how the robot's position and orientation relate to an environment. In contrast, range information is typically generated from IR sensors, laser range finders, or sonar sensors, which detect distances and directions to obstacles but do not provide more general knowledge about the environment. Advancements in map-building algorithms [34, 64, 76, 85, 126] allow the integration of multiple range scans into maps that help an operator visualize how the robot's position and orientation relate to the environment.

In the previous experiments described in this chapter, we used both video and range information (current readings or a map) to navigate a robot (see Sections 4.1 and 4.2). During those experiments we observed that operators sometimes focused their attention on the map section of the interface and other times focused their attention on the section that contains the video. These anecdotal observations lead to the question of how video and map information affect an operator's ability to navigate a robot.[5]

In this experiment we look at the usefulness of video and map information as aids for navigation with both a side-by-side (2D) interface and an integrated (3D)

[5]Although the ways to combine maps and visualization tools have been studied in other domains such as aviation (see, for example [22, 123]), this problem has not been well studied in human-robot operation with occupancy grid maps.

interface. We hypothesized that with 2D interfaces video would negatively influence an operator's ability to perform a navigation task because it does not provide sufficient lateral information and it may draw the operator's attention away from more useful areas of the interface such as map or range information [68]. Furthermore, we hypothesized that with a 3D interface, video information would not hinder navigation when other range information is present. To explore the effect of range and video information on navigation, we assess an operator's ability to navigate a maze environment with two interfaces (2D and 3D) and three conditions for each interface (map-only, video-only, and map+video).

4.3.1 Framework

In order to provide a visually rich environment for this experiment, we adopted a simulator based on the popular Unreal Tournament game engine as modified by Michael Lewis and colleagues at the University of Pittsburgh [75, 132]. Their modifications originated with the intent of providing an inexpensive yet realistic simulator for studying urban search and rescue with mobile robots. The Unreal Tournament game engine provides a rich visual environment that, when combined with accurate models of common research robots and the game's physics engine, provides for a very good mobile robot simulator [74].

We used the Unreal Tournament level editor to create maze environments that have the appearance of concrete bunkers filled with pipes, posters, windows, wiring, and electronic devices to provide a visually rich environment for the robot to travel through. Some pictures from the simulated environment are shown in Figure 4.8.

The experiment uses seven separate mazes that are designed to explicitly test low-level navigation skills. There is only one path through each maze and no dead-ends, but it takes considerable teleoperation skill to navigate a maze from start to finish without hitting any walls. One of the mazes is used for training and the other six mazes are used for testing. The training maze contains a continuous path without an exit so that participants can practice driving the robot as long as desired.

Figure 4.8: Pictures of the environment used for the information usefullness experiment.

Each maze is an 8x8 grid where each cell in the grid is 2x2 meters for a total maze area of 256m^2. Each maze is designed to have 42 turns and 22 straight cells to minimize differences in results from different mazes (see Figure 4.9). The simulated robot used for this experiment is a model of the ATRV-Jr robot and has a width and length of 0.6 meters. Pictures of the robot in one of the mazes are shown in Figure 4.10.

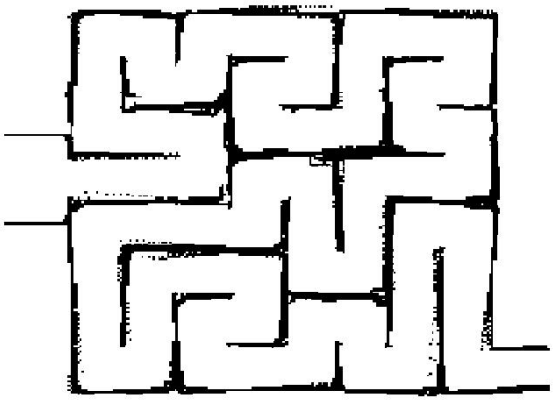


Figure 4.9: A map of one of the mazes used in the information usefulness experiment.

Figure 4.10: Pictures of the model of the ATRV-Jr robot used in the Unreal Tournament simulator.

Instructions

Operators were instructed on how to drive the robot and how to perform the experiment through speakers on a headset, and they were told that their goal was to get the robot out of the maze as quickly as possible without hitting too many walls.

Before testing, operators were given a chance to practice driving the robot with both the 2D and the 3D interfaces. Each interface displayed both map and video information. The operators were asked to drive at least once through the training maze to ensure a minimum amount of training. Once an operator had completed the training maze they were asked to continue practicing until they felt comfortable controlling the robot with the interface (most participants stopped training at this point). Following each training session and each experiment, participants were given a questionnaire to evaluate their performance. The purpose of the questionnaires after the training sessions was to familiarize the operators with the questions we would ask after each experiment.

Secondary task

Shortly after the operator started driving the robot, we introduced a secondary task to the operator. The purpose of the secondary task is to fill the short term memory of the operator and can be used as an indicator of excessive operator workload [136].

For the secondary task, participants were told that they were to count the number of times they heard a particular word spoken through the headset. The word is selected randomly from a list of 20 words (See Table 4.3.1). Throughout training and testing, randomly selected words are spoken once every three seconds. The word of interest is spoken at intervals of 1-12 words with a new interval chosen every time the word of interest is spoken. Upon completion of each test, participants were asked to record how many times they heard the word of interest.

algebra	beluga	binder	caboose	driving
eskimo	falcon	galoshes	gazebo	hallway
lettuce	market	mukluk	pizza	plethora
quickly	racket	title	sunshine	whistle

Table 4.4: List of words used for the secondary task.

Procedure

Once training was complete, each participant was asked if they had any questions and they were told that the experiments would be very similar to the training, except that the trial would begin with the robot in a room and end when the robot exited the maze and that they would have different sets of information visible on the interface for each test; specifically, participants were given conditions of *video-only*, *map-only*, and *map+video* for both the 2D and 3D interfaces.

For testing, we used a within-subjects counter-balanced design where each operator performed a test with each of the six conditions. The conditions were presented in a pseudo-random order with the constraints that the 2D and 3D interfaces were used alternately. The interfaces for each of the six tests are shown in Figure 4.11.[6]

4.3.2 Results

Twenty-four participants were paid to navigate a simulated robot with the six different conditions of information presentation. Participants were recruited from the Brigham Young University community with most subjects enrolled as students. Two participants terminated the experiment prior to completion of the six conditions, but completed portions of the experiment were used for our analysis. We begin by comparing the map-only and video-only conditions for both interfaces. We then discuss how the map+video condition compares to the map-only and video-only conditions for the two interfaces. We then discuss a learning effect that we observed with

[6]We performed the video condition test with both the 2D and the 3D interfaces because we did not want to bias the operators feelings towards one type of interface as people tended to get quite frustrated when driving with only video.

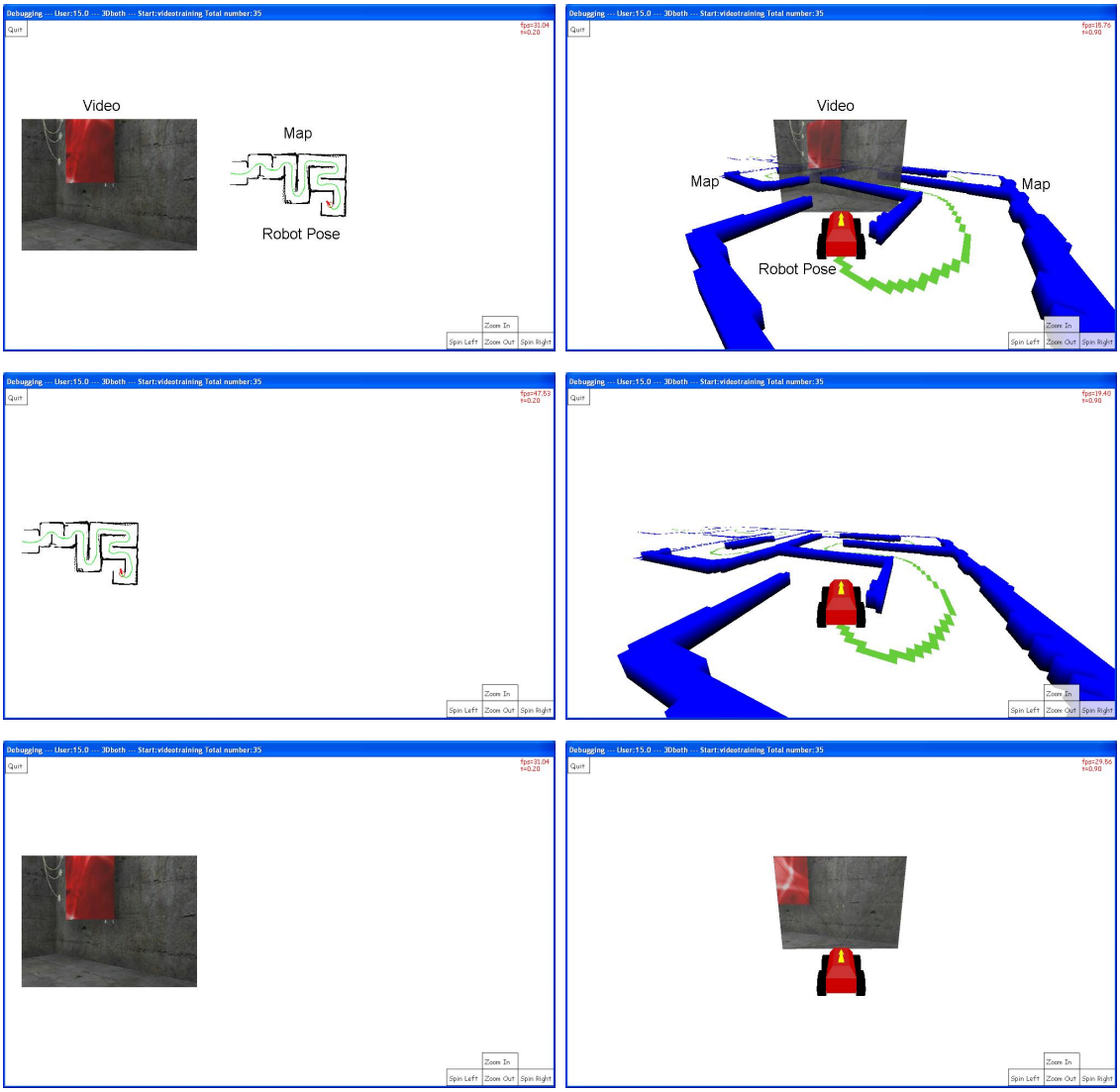


Figure 4.11: The interfaces used for the video effect experiment. Clockwise from bottom left: 2D video-only, 2D map-only, 2D map+video, 3D map+video, 3D map-only, and 3D video-only.

	2D map-only	2D video-only	% Change	p-value
Time to Completion (s)	258	366	42%	7.8×10^{-4}
Average Collisions	9.83	19.10	94%	1.3×10^{-3}
Average Velocity (m/s)	0.54	0.43	-20%	5.4×10^{-5}
Forward Velocity (m/s)	0.44	0.32	-27%	8.3×10^{-5}
Secondary Task Error	2.75	5.67	106%	1.6×10^{-1}

Table 4.5: Objective data comparing the 2D video-only condition with the 2D map-only condition.

some of the conditions. We then present implications of the results of this experiment. Throughout the discussion of the results, statistical significance is obtained with a paired, two-tailed t-test with $n = 24$ samples unless otherwise specified.

Map-only vs. Video-only

In 2D, the video-only condition took significantly longer (42%) than the map-only condition. Similarly, the map-only condition had an average velocity 20% faster than the video-only condition and the robot's forward progress through the maze was 27% faster with the map-only condition than the video-only condition. Additionally, there were nearly twice as many collisions with the video-only condition as compared to the map-only condition. There was only marginal statistical difference in performance on the secondary task. Table 4.5 summarizes the comparison of the map-only and video-only conditions with the 2D interface.

In 3D, the video-only condition took 79% longer to finish the maze than the map-only condition. Similarly, with the map-only condition the robot had an average velocity 38% faster than the video-only condition and the robot's forward progress through the maze was 41% faster with the map-only condition. Additionally, with the video-only condition, the robot collided with the walls, on average, eighteen times more frequently than with the map-only condition. There were also four times the errors in the secondary task with the video-only condition as opposed to the map-only condition. Table 4.6 summarizes the comparison of the map-only and video-only conditions with the 3D interface.

	3D map-only	3D video-only	% Change	p-value
Time to Completion (s)	196	351	79%	1.6×10^{-7}
Average Collisions	1.25	22.71	1717%	1.3×10^{-6}
Average Velocity (m/s)	0.66	0.41	-38%	5.5×10^{-12}
Forward Velocity (m/s)	0.57	0.33	-41%	3.9×10^{-12}
Secondary Task Error	1.04	4.29	313%	2.5×10^{-3}

Table 4.6: Objective data comparing the 3D video-only condition with the 3D map-only condition.

	2D video-only	3D video-only	% Change	p-value
Time to Completion (s)	366	351	-4.1%	0.662
Average Collisions	19.10	22.71	19%	0.175
Average Velocity (m/s)	0.43	0.41	-4.7%	0.198
Forward Velocity (m/s)	0.32	0.33	3.1%	0.613
Secondary Task Error	5.67	4.29	-24%	0.486

Table 4.7: Objective data comparing the video-only conditions for the 2D and 3D interfaces.

One observation from Tables 4.5 and 4.6 is that the differences between the map-only and video-only conditions are more profound with the 3D interface than with the 2D interface. Note that the video-only conditions had very similar results for both interfaces (see Table 4.7). The main difference is in the results from the map-only condition where the 3D interface is better than the 2D interface. We discuss this further in Section 4.3.3.

Map+video

We found that with both the 2D and 3D interfaces, the map+video condition had results that were more similar to the map-only condition in comparison to the video-only condition.

In particular, with the 2D interface, we found that the map+video condition took slightly longer to complete than the map-only condition (5.1%, p = 0.189). Additionally, forward progress through the maze (5.1%, p = 0.144) was slower and average velocity was slower (7.3%, $p = 3.1 \times 10^{-2}$) with the map+video condition than

with the map-only condition. There was also a non-significant difference between the number of collisions with obstacles and the errors in the secondary task when comparing the map-only and map+video conditions (see Table 4.8 and Figure 4.12).

With the 3D conditions, the results are similar to that of the 2D conditions, except that there is stronger statistical evidence. In particular, the map+video condition took 6.2% longer ($p = 4.2 \times 10^{-2}$), average velocity was 4.3% slower ($p = 6.1 \times 10^{-2}$), and forward progress was 5.1% slower ($p = 3.4 \times 10^{-2}$) than the map-only condition. Further, the average number of collisions was identical (1.25, p = 1.0) between the map-only and map+video conditions and there was no statistical difference in the errors of the secondary task (see Table 4.8 and Figure 4.12).

In general there is a slight or insignificant change in time to completion when video information is added to map information for both the 2D and 3D interfaces. However, there is a marginally significant learning effect that took place with the 2D map-only condition and the 3D map+video condition. In particular, the participants who used the 2D map-only condition *after* the 2D map+video condition finished the task 14% faster than the participants who used the 2D map-only condition *before* the 2D map+video condition ($\bar{x}_{2Dmap1} = 278$s, $\bar{x}_{2Dmap2} = 238$s, $p = 9.5 \times 10^{-2}$, $n = 12$, unpaired t-test, see Table 4.9).

Similarly, the participants that used the 3D map+video condition *after* the 3D map-only condition finished the task 15% faster than the participants that used the 3D map+video condition *before* the 3D map-only condition ($\bar{x}_{3Dmap+video1} = 225$s, $\bar{x}_{3Dmap+video2} = 191$s, $p = 1.2 \times 10^{-2}$, $n = 12$, unpaired t-test, see Table 4.10). There was no significant difference between groups with the 2D map+video condition (2%, $\bar{x}_{2Dboth1} = 269$s, $\bar{x}_{2Dboth2} = 273$s, $p = 0.849$, $n = 12$, unpaired t-test) and the 3D map-only condition (.3%, $\bar{x}_{3Dmap1} = 195$s, $\bar{x}_{3Dmap2} = 196$s, $p = 0.962$, $n = 12$, unpaired t-test).

When we compare the set of experiments in 2D where the map-only and map+video conditions were used first (Table 4.9), we find that adding video to the map has an insignificant effect. However in the set of experiments where the map-only and map+video conditions are used second, we found that the map+video condition

	2D map-only	2D map+video	% Change	p-value
Time to Completion (s)	258	272	5.1%	1.9×10^{-1}
Average Collisions	9.83	8.50	-14%	3.5×10^{-1}
Average Velocity (m/s)	0.54	0.50	-7.3%	3.1×10^{-2}
Forward Velocity (m/s)	0.44	0.42	-5.1%	1.4×10^{-1}
Secondary Task Error	2.75	1.79	-35%	2.2×10^{-1}

	2D video-only	2D map+video	% Change	p-value
Time to Completion (s)	366	272	-26%	2.1×10^{-3}
Average Collisions	19.10	8.50	-55%	1.6×10^{-4}
Average Velocity (m/s)	0.43	0.50	16%	1.9×10^{-2}
Forward Velocity (m/s)	0.32	0.42	30%	4.3×10^{-4}
Secondary Task Error	5.67	1.79	-68%	5.1×10^{-2}

	3D map-only	3D map+video	% Change	p-value
Time to Completion (s)	196	208	6.2%	5.1×10^{-2}
Average Collisions	1.25	1.25	0%	1.0×10^{0}
Average Velocity (m/s)	0.66	0.63	-4.3%	6.1×10^{-2}
Forward Velocity (m/s)	0.57	0.54	-5.1%	3.5×10^{-2}
Secondary Task Error	1.04	0.91	-12%	7.4×10^{-1}

	3D video-only	3D map+video	% Change	p-value
Time to Completion (s)	351	208	-41%	1.8×10^{-6}
Average Collisions	22.71	1.25	-95%	1.1×10^{-6}
Average Velocity (m/s)	0.41	0.63	55%	9.8×10^{-10}
Forward Velocity (m/s)	0.33	0.54	62%	1.5×10^{-10}
Secondary Task Error	4.29	0.91	-79%	1.5×10^{-3}

Table 4.8: Comparison of the map+video condition to the map-only and video-only conditions for the 2D and 3D interfaces.

	First	Second	% Change	p-value
2D map-only	278s	238s	-14%	9.5×10^{-2}
2D map+video	269s	273s	1.7%	8.5×10^{-1}
% Change	-3.1%	15%		
p-value	7.4×10^{-1}	9.1×10^{-2}		

Table 4.9: Time to completion in 2D after adjusting for learning.

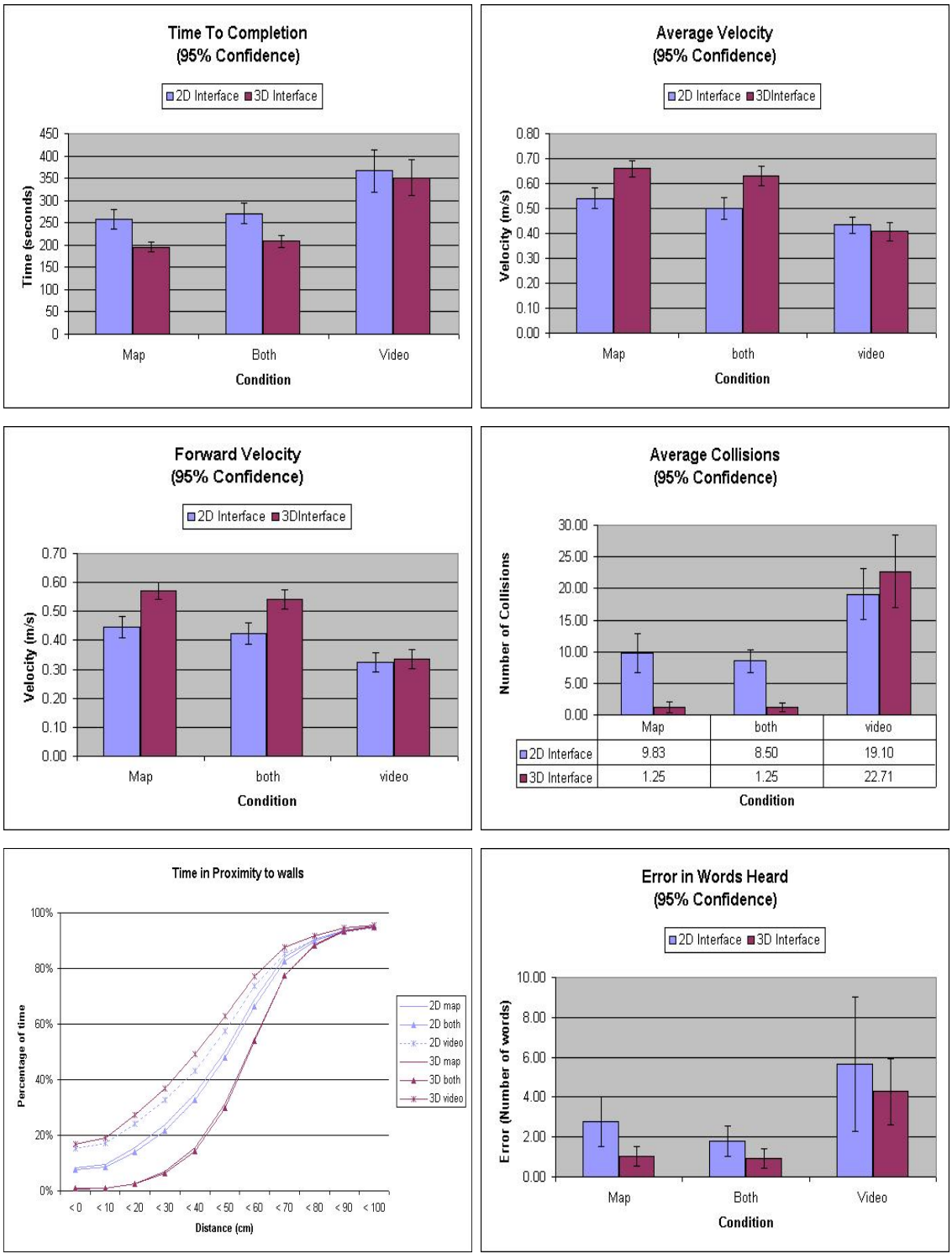


Figure 4.12: Average results for the video experiment.

	First	Second	% Change	p-value
3D map-only	195s	196s	0.32%	9.6×10^{-1}
3D map+video	225s	191s	-15%	1.2×10^{-2}
% Change	-15%	-2.7%		
p-value	3.6×10^{-2}	6.3×10^{-1}		

Table 4.10: Time to completion in 3D after adjusting for learning.

takes 15% longer to complete the task than the map-only condition. This suggests that after accounting for learning, adding video to the map hurts navigation by increasing the time it takes an operator to navigate the robot out of a maze (see Figure 4.13).

When we compare the set of experiments in 3D where the map-only and map+video conditions are used first (Table 4.10), we find that adding video to the map increases the time to completion by 15.2%. However, in the set of experiments where the map-only and map+video conditions are used second, we find the difference in the time to complete the task is insignificant, which suggests that after accounting for learning, adding video to the map in the 3D interface does not affect the time it takes to navigate the robot out of the maze (see Figure 4.13).

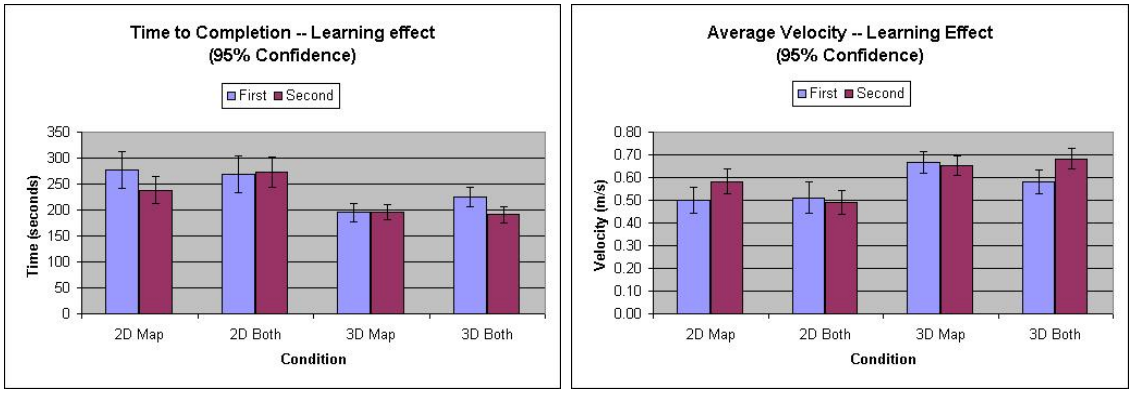


Figure 4.13: Time to completion and average velocity after accounting for learning in the video experiment.

4.3.3 Discussion and Further Observations

These results suggest that video can hurt navigation when the video does not contain sufficient navigational cues and video and map information are placed side-by-side. Even when map information is present and more useful than video for navigating, a novice operator's attention tends to be drawn towards the video, which, in this case, negatively affects their ability to navigate. These results make sense in light of research done by Kubey and Csikszentmihalyi, which has shown that television draws attention because of the constantly changing visual scene [68]. In contrast, video does not negatively affect navigation when added to map information with a 3D interface. This is because even though the video may not contain useful information and may draw attention, the map is still readily visible to the operator because it is presented integrated with the video.

We also observed that with the map-only and map+video conditions, operators did much better using the 3D interface than the 2D interface. With the 3D interface, operators finished the task at least 23% more quickly, with an average velocity at least 22% faster. Additionally, there were at least 85% fewer collisions with the 3D interface than the 2D interface and the operators had almost half the error[7] on the secondary task with the 3D interface. (See Table 4.11 for a detailed description of the comparisons between the 2D and 3D interfaces. The charts are shown previously in Figure 4.12).

Subjectively, 54% of the operators felt that the robot collided with walls more with the video-only condition than either the map-only or map+video conditions when using the 2D interface. With the 3D interface, 88% of the operators felt that the robot collided with walls more with the video-only condition than either the map-only or map+video conditions. Furthermore, 92% of the operators felt that they collided more with the 2D interface than with the 3D interface in conditions other than video-only. These evaluations match objective results discussed earlier. Operators also felt that it was more frustrating to use the video-only condition in comparison to the

[7]Error is defined as the absolute difference between a participant's answer to the number of times the word was spoken and the actual number of times the word was spoken.

	Condition	2D Interface	3D Interface	% Change	p-value
Time to Completion (s)	map-only	258	196	-24%	1.5×10^{-6}
	map+video	271	208	-23%	5.9×10^{-7}
Average Velocity (m/s)	map-only	0.54	0.66	22%	5.5×10^{-12}
	map+video	0.50	0.63	26%	1.4×10^{-7}
Forward Velocity (m/s)	map-only	0.45	0.57	28%	9.9×10^{-8}
	map+video	0.42	0.54	28%	1.9×10^{-8}
Average Collisions	map-only	9.83	1.25	-87%	1.7×10^{-5}
	map+video	8.50	1.25	-85%	1.2×10^{-8}
Secondary Task Error	map-only	2.75	1.04	-62%	1.5×10^{-2}
	map+video	1.79	0.92	-49%	3.0×10^{-2}

Table 4.11: Comparison of the 2D and 3D interfaces with the map-only and map+video conditions. Significance obtained from two-tailed t-test with n = 24 samples.

map-only and map+video conditions, and they felt that using the 2D interface with the map-only and map+video conditions was more frustrating than using the 3D interface with the same conditions.

Participants felt that with the map-only and map+video conditions, the 3D interface made better use of the information than the 2D interface, and that the video-only conditions were lacking important information to navigate the robot, in particular, information about obstacles to the sides of the robot. Furthermore, 23 out of the 24 participants preferred the 3D interface with the map-only or map+video condition over the 2D interface with the same conditions. Twenty-three of the twenty-four participants also felt that they could get out of the maze faster while avoiding walls better with the 3D interface and either the map-only or map+video condition in comparision to the 2D interface with either condition. Figure 4.14 shows the results of the questionnaire and Table 4.12 presents the statistics from the questionnaire.[8]

[8]Operators were asked how much they agreed or disagreed with statements including: *Did you have sufficient information to drive the robot*, *The robot did not have many collisions*, and *It was frustrating to drive the robot*. Numerical results are obtained from: 0 = Agree, 1 = Somewhat Agree, 2 = Neutral, 3 = Somewhat Disagree, 4 = Disagree.

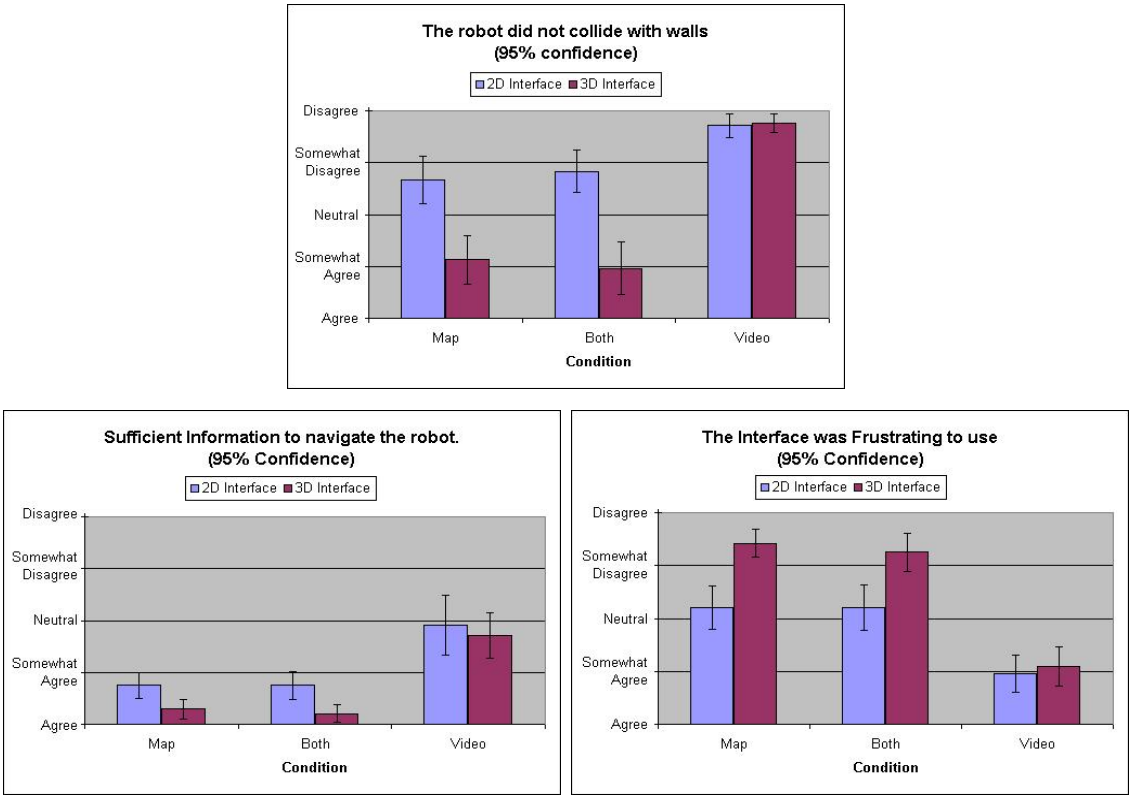


Figure 4.14: Subjective evaluations of the video effect experiment.

	Condition	2D Interface	3D Interface	% Change	p-value
	map-only	0.75	0.29	-61%	2.4×10^{-3}
Sufficient Info	map+video	0.75	0.21	-72%	1.2×10^{-3}
	video-only	1.90	1.71	-10%	4.9×10^{-1}
	map-only	2.67	1.13	-58%	1.0×10^{-4}
No Collisions	map+video	2.83	0.96	-66%	7.4×10^{-6}
	video-only	3.71	3.76	1.3%	7.5×10^{-1}
	map-only	2.21	3.42	55%	1.9×10^{-5}
Frustrating	map+video	2.21	3.25	47%	2.1×10^{-4}
	video-only	0.95	1.10	15%	5.5×10^{-1}

Table 4.12: Subjective results as obtained from the questionnaire.

2D	240 × 320	360 × 480	480 × 640	p-value
Time to Completion (s)	278	273	292	0.524
Average Velocity (m/s)	0.49	0.50	0.48	0.802
Average Collisions	5.30	9.67	5.67	0.190
Secondary Task Error	2.30	2.22	4.01	0.240

3D	240 × 320	360 × 480	480 × 640	p-value
Time to Completion (s)	210	217	218	0.441
Average Velocity (m/s)	0.63	0.62	0.60	0.348
Average Collisions	1.17	0.67	0.50	0.296
Secondary Task Error	1.30	1.67	0.67	0.100

Table 4.13: Objective data for the video size experiment.

4.4 Video Size Experiment

As a follow up to the *Information Usefulness* experiment we also looked at how the size of video affected an operator's ability to navigate the robot. Our hypothesis was that larger video would distract an operator more than smaller video and would therefore lead to a decrease in performance. We performed a pilot study with six participants where each used three different sizes of video for both the 2D and 3D interfaces. The video occupies approximately 240×320, 360×480, and 480×640 pixels of the screen space.[9] The environment for this experiment is the same as the previous experiments using the Unreal Tournament game engine. Operators were presented with the map+video condition for all test cases; the difference between the conditions is that the video information is displayed in different sizes. The interfaces that we used for the video size experiment are shown in Figure 4.15.

Results

We found only minimal and non-significant differences between the different video sizes (see Table 4.4) so we did not pursue this study further. The difference between the 2D and 3D interfaces for the different conditions is consistent with previous results.

[9] Approximate because the perspective of the video in 3D creates a warped trapezoid of the video.

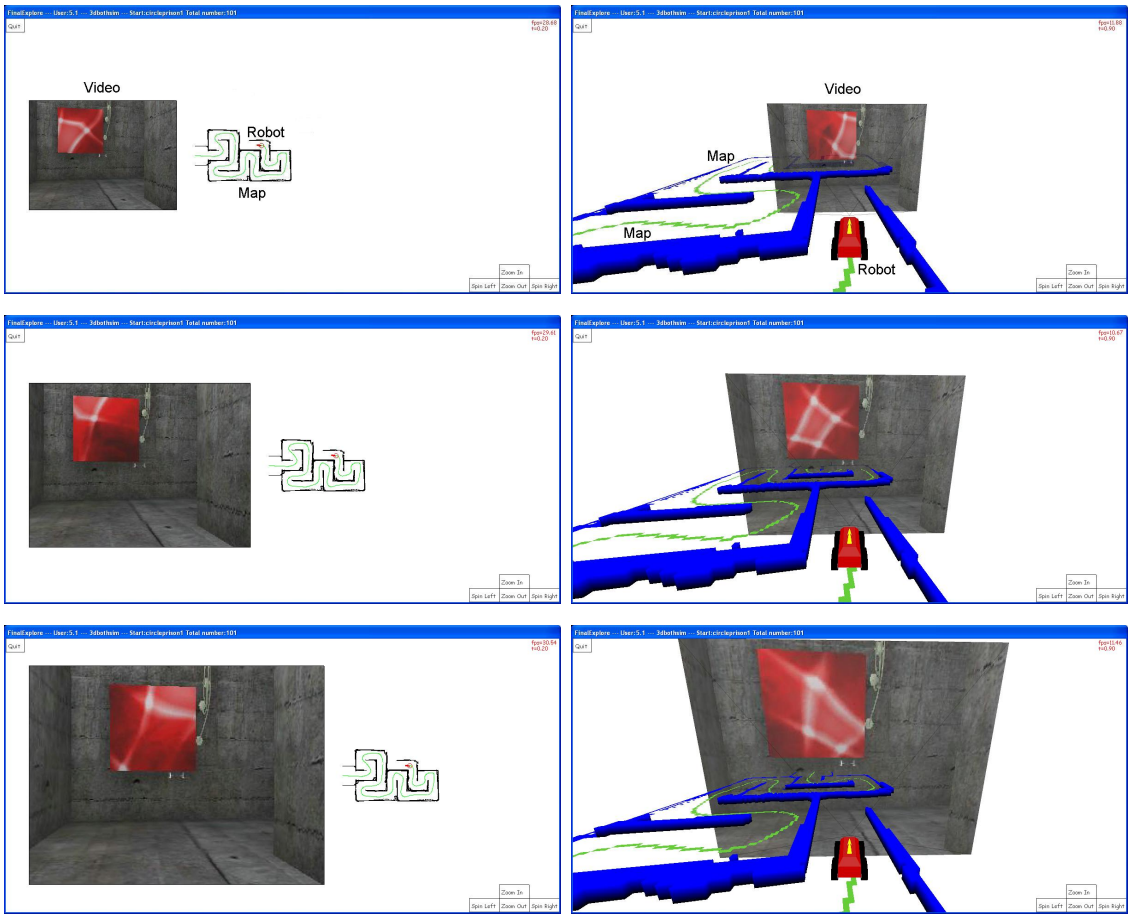


Figure 4.15: Interfaces used for the video size experiment. Clockwise from bottom left: 2D(480 × 640), 2D(360 × 480), 2D(240 × 320), 3D(240 × 320), 3D(360 × 480), 3D(480 × 640).

4.5 Delay Experiment

Throughout our previous experiments, we have observed that people tend to do much better with a 3D interface than a 2D interface. We feel that a large part of this is because the operators are better able to understand the situation around the robot. This situation awareness has three main aspects according to Endsley: perception, comprehension, and projection [35]. In previous experiments we have shown that operators seem to be better able to perceive and comprehend relevant information for robot control from the environment when using the 3D interface as opposed to the 2D interface. In this experiment, we look at an operator's ability to anticipate how the robot will respond to their commands while navigating the robot with network delay. We hypothesized that the 3D interface would allow better performance than the 2D interface for equal amounts of delay up to one second of delay.

4.5.1 Framework

For this experiment we used the Unreal Tournament game engine modifications for our simulator. The experiment is designed and set up the same as the previous *Information Usefulness* experiment, with the exception that instead of changing the information visible on the interface, we changed the network delay from when a command is issued to when the actions of the command are seen by the operator. The three delay conditions used are: *0-seconds*, *0.5-seconds*, and *1-second*.

Instructions

Operators were instructed on how to drive the robot and how to perform the experiment through speakers on a headset. Operators were told that their goal is to get the robot out of the maze as quickly as possible without hitting too many walls. They were also informed that some of the conditions they would use had network delay where it takes a short time for the operator to see the result of a command given to the robot.

Upon completion of the instructions, the operators are given a chance to practice driving the robot with both the 2D and the 3D interfaces with map and video information visible. The operators were asked to drive at least once through the training maze to ensure a minimum amount of training. Once operators had completed the training maze they were asked to continue practicing until they felt comfortable controlling the robot with the interface (again, most participants stopped training at this point). Following each training session and each experiment, participants were given a questionnaire to evaluate their performance.

Secondary task

A secondary task was used to fill the short term memory and increase the workload of the operator [136]. Shortly after the operator started driving the robot, we introduced the secondary task to the operator. We used the same secondary task that was described in Section 4.3.1.

Procedure

Once training was complete, participants were asked if they had any questions and they were told that the experiments would be very similar to the training they did except that the robot would begin in a room and end when the robot exited the maze, and that there would be different amounts of delay during some of the tests. There were six tests performed by each participant, *2D 0-seconds*, *2D 0.5-seconds*, *2D 1-second*, *3D 0-seconds*, *3D 0.5-seconds*, and *3D 1-seconds*. The 2D and 3D interfaces for this experiment present both map and video information to the operator. The two interfaces used for the delay experiment are shown in Figure 4.16. A counter-balanced random schedule was created with the constraint that the interfaces (2D and 3D) were used alternately.

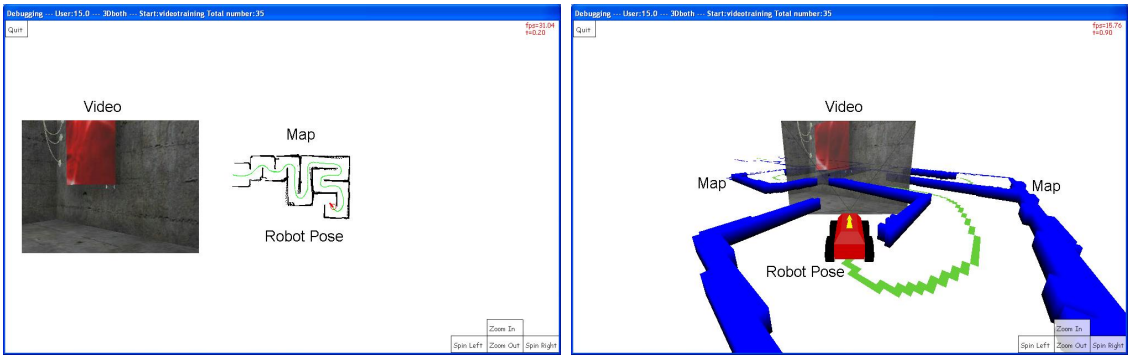


Figure 4.16: The 2D interface (left) and the 3D interface (right) used for the delay experiment.

4.5.2 Results

Eighteen people completed the delay experiment. We begin by discussing results in detail then summarizing the meaning of the results. Throughout the discussion of the results, statistical significance is obtained through a paired t-test with n=18 samples unless otherwise specified.

Time to completion

We found that operators were able to finish the navigation task 27%, 26%, and 19% faster with the 3D interface than with the 2D interface for delays of 0, 0.5, and 1 second respectively (see Table 4.14 and Figure 4.17). Additionally, we found that the results from the 3D conditions were comparable to results from the 2D conditions when the 2D conditions had a half of a second less delay than the 3D conditions (see Table 4.14). For example, the 2D 0-seconds condition had an average time of 302 seconds and the 3D 0.5-seconds condition had an average time of 311 seconds (3% slower, $p = 0.639$). Additionally, the 2D 0.5-seconds condition had an average time of 422 seconds and the 3D 1-second condition had an average time of 466 (10% slower, $p = 0.292$). Furthermore, ten of the participants finished the 3D 0.5-seconds condition faster than the 2D 0-seconds condition and six of the participants finished the 3D 1-second condition faster than the 2D 0.5-seconds condition.

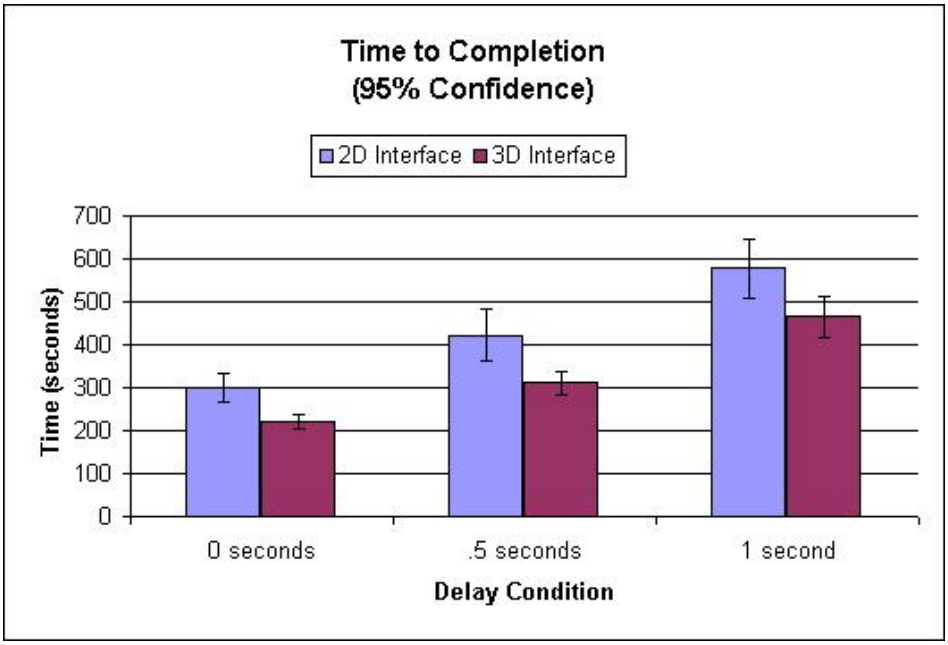


Figure 4.17: Time to completion for the delay experiment.

	2D Interface	3D Interface	% Change	p-value
0-seconds	302s	221s	-27%	5.0×10^{-5}
0.5-seconds	422s	311s	-26%	2.4×10^{-4}
1-seconds	578s	466s	-19%	2.3×10^{-3}

Table 4.14: Time to completion statistics for the delay experiment.

Average Velocity

We also found that the 3D interface produced average velocities that were 30%, 22%, and 18%[10] faster than the 2D interface for the 0-seconds, 0.5-seconds, and 1-second conditions, respectively (see Table 4.15 and Figure 4.18). Again, we found that results from the 3D conditions were comparable to results from the 2D conditions when the 2D conditions had a half of a second less delay than the 3D conditions (see Table 4.15). For example, the 2D 0-seconds condition had an average velocity of 0.46 m/s and the 3D 0.5-seconds condition had an average velocity of 0.47 m/s (1.7% faster, $p = 0.553$). Additionally, the 2D 0.5-seconds condition had an average velocity of 0.38 m/s and the 3D 1-second condition had an average velocity of 0.36 m/s (5.6% slower, $p = 0.493$). Furthermore, thirteen participants drove faster with the 3D 0.5-seconds condition than the 2D 0-seconds condition and eight participants drove faster with the 3D 1-second condition than the 2D 0.5-seconds condition.

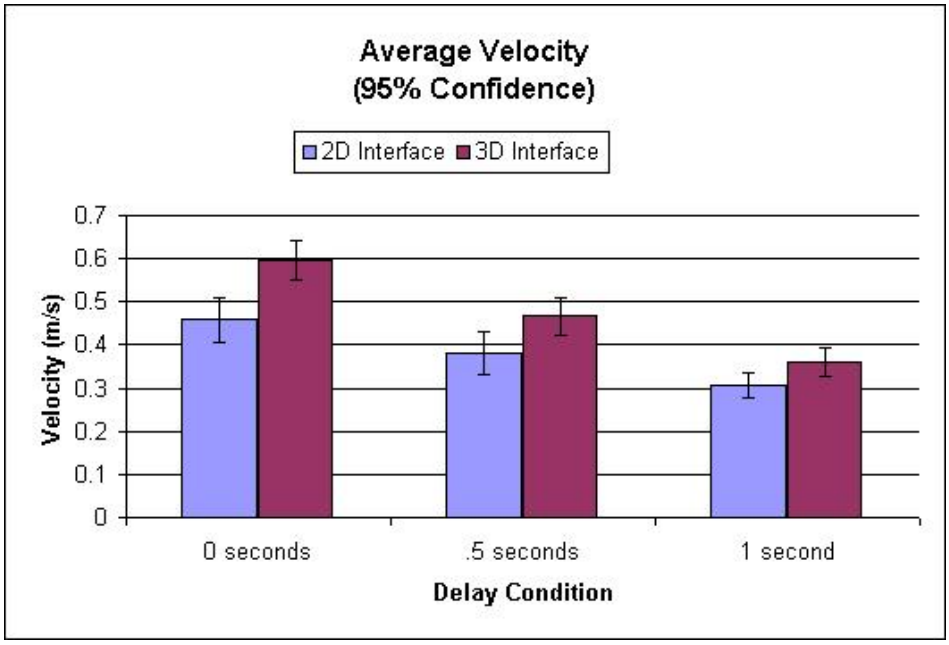


Figure 4.18: Average Velocity for the delay experiment.

[10]Only marginally significant

	2D Interface	3D Interface	% Change	p-value
0-seconds	0.46m/s	0.60m/s	30%	3.7×10^{-4}
0.5-seconds	0.38m/s	0.47m/s	22%	1.4×10^{-3}
1-second	0.31m/s	0.36m/s	18%	1.9×10^{-1}

Table 4.15: Average velocity statistics for the delay experiment.

	2D Interface	3D Interface	% Change	p-value
0-seconds	10.6	1.7	-84%	9.9×10^{-4}
0.5-seconds	22.7	8.1	-65%	6.8×10^{-3}
1-second	38.6	28.4	-27%	1.2×10^{-1}

Table 4.16: Average collision statistics for the delay experiment.

Collisions

There was also an 84%, 65%, and 27% decrease in collisions with the 3D interface in comparison to the 2D interface for the 0-seconds, 0.5-seconds, and 1-second conditions, respectively (see Table 4.16 and Figure 4.19). In fact there were marginally significant fewer collisions with the 3D 0.5-seconds condition than with the 2D 0-seconds condition ($-24\%, x_{3D_{0.5-seconds}} = 8.1, x_{2D_{0-seconds}} = 10.6, p = 0.105$). There was a non-significant difference in the number of collisions with the 2D 0.5-seconds condition and the 3D 1-second condition (p = 0.455). Furthermore, ten participants had fewer collisions with the 3D 0.5-seconds condition than the 2D 0-seconds condition and seven participants had fewer collisions with the 3D 1-second condition than the 2D 0.5-second condition.

Additionally, operators spent more of their navigational time further from obstacles with both the 3D 0-seconds and 3D 0.5-seconds conditions in comparison to the 2D 0-seconds condition (see Figure 4.20).

Secondary task

In the secondary task, participants did marginally better with the 3D interface than the 2D interface for the 0-seconds (43%, $p = 6.8 \times 10^{-2}$) and 0.5-seconds conditions (29%, p = 0.119), see Table 4.17. We found that statistically, the 2D 0s-delay condition and the 3D 0.5s-delay condition had the most similar results with

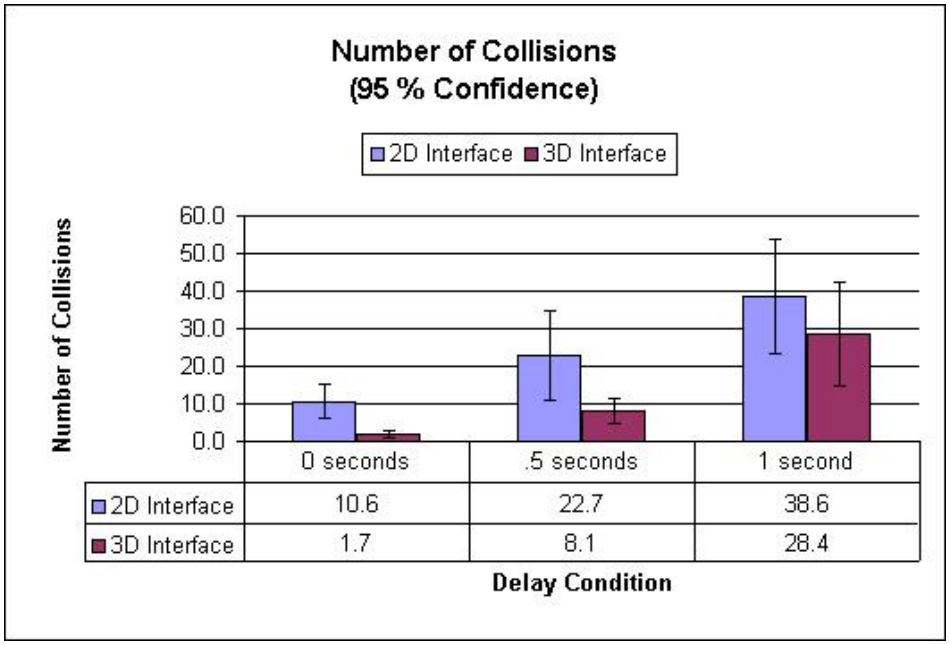


Figure 4.19: Average collisions for the delay experiment.

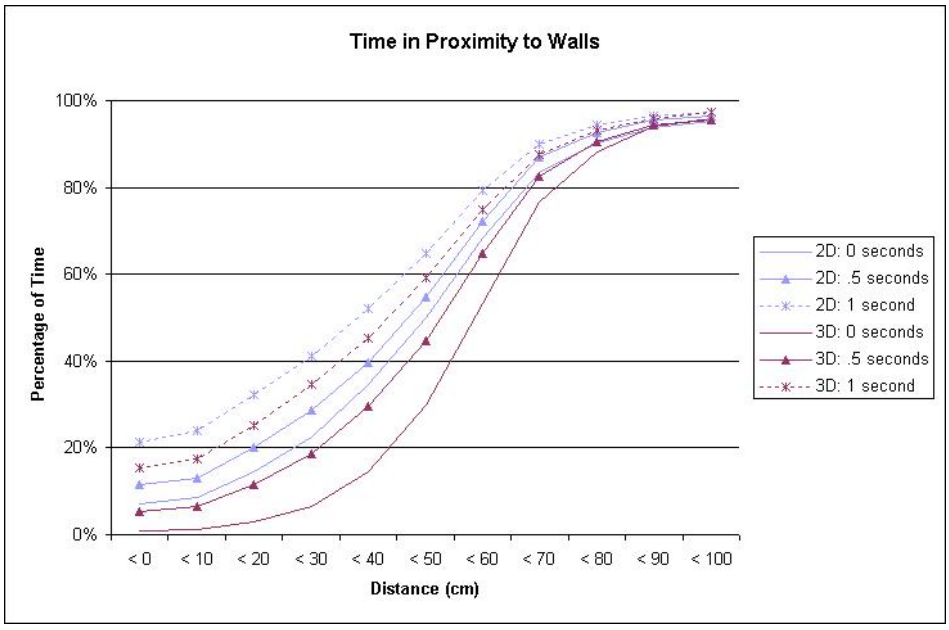


Figure 4.20: Percentage of time spent with the robot in close proximity to an obstacle.

	2D Interface	3D Interface	% Change	p-value
0-seconds	3.29	1.89	-43%	6.8×10^{-2}
0.5-seconds	4.53	3.22	-29%	1.2×10^{-1}
1-second	6.25	5.75	-8.0%	7.9×10^{-1}

Table 4.17: Secondary task error statistics for the delay experiment.

respect to the error in the task ($\bar{x}_{2D_{0-seconds}} = 3.29$, $\bar{x}_{3D_{0.5-seconds}} = 3.22$, $p = 0.958$). The 2D 0.5-seconds condition had marginally fewer errors than the 3D 1-second condition ($\bar{x}_{2D_{0.5-seconds}} = 4.53$, $\bar{x}_{3D_{1-second}} = 5.75$, $p = 0.130$).

Subjective results

Subjectively operators felt that they were better able to anticipate how the robot would respond to their command with the 3D interface (see Figure 4.21), which is also supported by the shorter times to completion. Participants felt that the robot did not collide with as many walls with the 3D interface as it did with the 2D interface (see Figure 4.22, which also correlates with the objective data). Thirteen of the eighteen participants preferred the 3D interface over the 2D interface, and the other five had no preference between the two interfaces. Twelve participants felt that they could move the robot fastest with the 3D interface, with only one claiming the 2D interface was faster and the other five indicating the interfaces were about the same. Twelve of the participants also felt the 2D interface was more affected by delay than the 3D interface. Three felt that the 3D interface was more affected by delay than the 2D interface and three indicated the affect of delay on the interfaces was about the same.

4.5.3 Discussion

The results show that the 3D interface is consistently better than the 2D interface across multiple levels of delay. Additionally, the 2D interface has results similar to the 3D interface when the 3D interface has an additional half second of delay. This suggests that the operator is better able to anticipate how the robot will

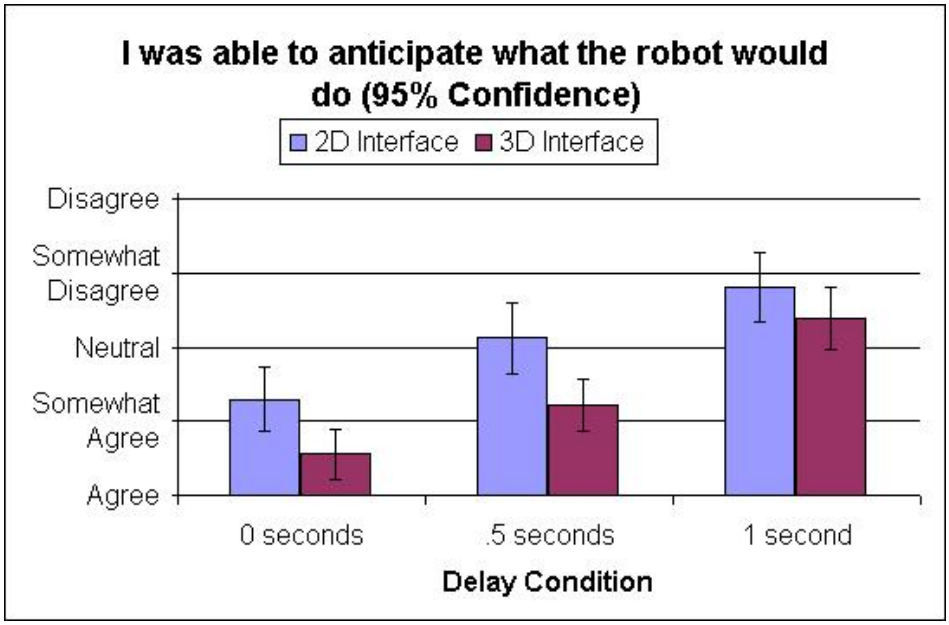


Figure 4.21: Subjective results of the operator's ability to anticipate how the robot would respond to their commands.

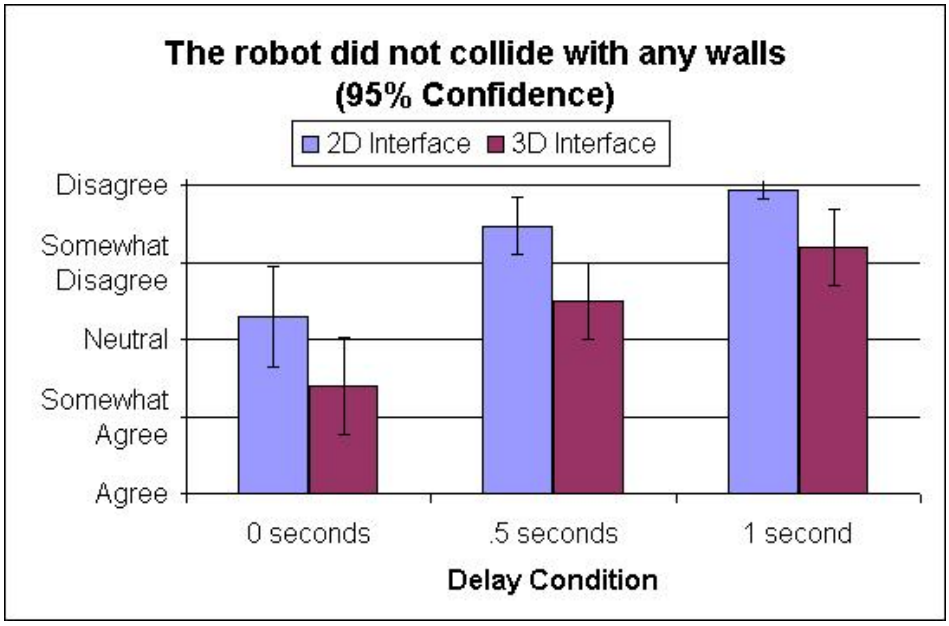


Figure 4.22: Subjective results of the operator's assessment of whether or not the robot collided with the walls.

respond to commands with the 3D interface than the 2D interface in a navigation task. The difference between the 3D and 2D interfaces for similar levels of delay is supported by the results from previous experiments.

From one perspective we can say that the 3D interface is more robust to delay than the 2D interface because it allows similar results even though there is more network delay. However, from another perspective, the rate at which performance decreases as delay increases is similar for both the 2D and 3D interface. So the 3D interface is not robust with respect to slowing the decrease in performance when delay is increased.

4.6 Real-World Experiment

Previously we looked at the usefulness of video and map information and the effect of network delay on navigation tasks with a remote mobile robot in simulation. It is also useful to verify that the results and conclusions in simulation carry over and are applicable to environments and robots in the real world. For this purpose we have designed a follow-up experiment to compare the usefulness of video and map information when navigating a robot in the real world.[11] We hypothesized that the results would be similar to the results from the Information Usefulness experiment in Section 4.3.

4.6.1 Framework

For this experiment we converted part of the second floor of the Computer Science Department at Brigham Young University into an obstacle course for our robot to travel through. The normal hallway width is 2 meters and we used cardboard boxes, Styrofoam packing, and other obstacles to create a 50 meter course which has a minimum width of 1.2 meters. Figure 4.23 shows images of the robot and the two hallways used in the experiment.

[11] Network delay was not explicitly tested because the communications over the wireless network introduced erratic delay.

The Robot

The robot we used for the experiment is an ATRV-Jr developed by IRobot which is approximately 0.6 meters in width and 0.7 meters in length (see Figure 4.23). The robot is equipped with 17 sonar-sensors around its perimeter, a laser range finder located at the front of the robot and near the ground, and a pan-tilt-zoom camera located on top of the robot. The robot uses a map-building algorithm developed by Konolige at the Stanford Research Institute (SRI) to represent the environment and localize the robot within the map of the environment [64]. The map-building algorithm has been integrated with intelligence algorithms by David Bruemmer and Doug Few at the Idaho National Laboratory (INL) to safeguard the robot from colliding with obstacles as it is teleoperated [15, 16].

Specifically, the intelligence on the robot governs the speed at which the robot can move forward based on the robot's sensed proximity to obstacles and the stopping time required to keep the robot safe from colliding with an obstacle. Further, when the operator attempts to drive the robot into an obstacle, the intelligence on the robot stops the robot and warns the operator that the robot is blocked by vibrating the force-feedback joystick.

An operator controls the ATRV-Jr with a Microsoft Sidewinder 2 joystick[12] (see Figure 4.24) and range and video information from the robot are presented to the operator via our software which displays both a 2D and 3D prototype interface. The 3D interface is integrated with the INL base station which handles the communication of movement commands and general information between the operator and the robot via 900 MHz radio modems. Live video from the robot is transmitted to our interface via 802.11b wireless Ethernet. Due to the use of a real robot in a building, the information transmitted from the robot to the interface sometimes came erratically, with delays up to two seconds. Minimal delays (< 0.25 seconds) were the norm and large delays (> 1.0 seconds) were rare.

[12]The INL base station does not support the steering wheel we used in the simulated experiment.

Figure 4.23: Images of the environment and the robot used for the real world experiment.

Figure 4.24: The Microsoft Sidewinder 2 Joystick used to control the real ATRV-Jr robot.

The interfaces used for this experiment have been modified from the previous experiments by including icons that indicate where the robot's intelligence identifies obstacles that might interfere with robot movement. The interfaces used for this experiment are shown in Figure 4.25.

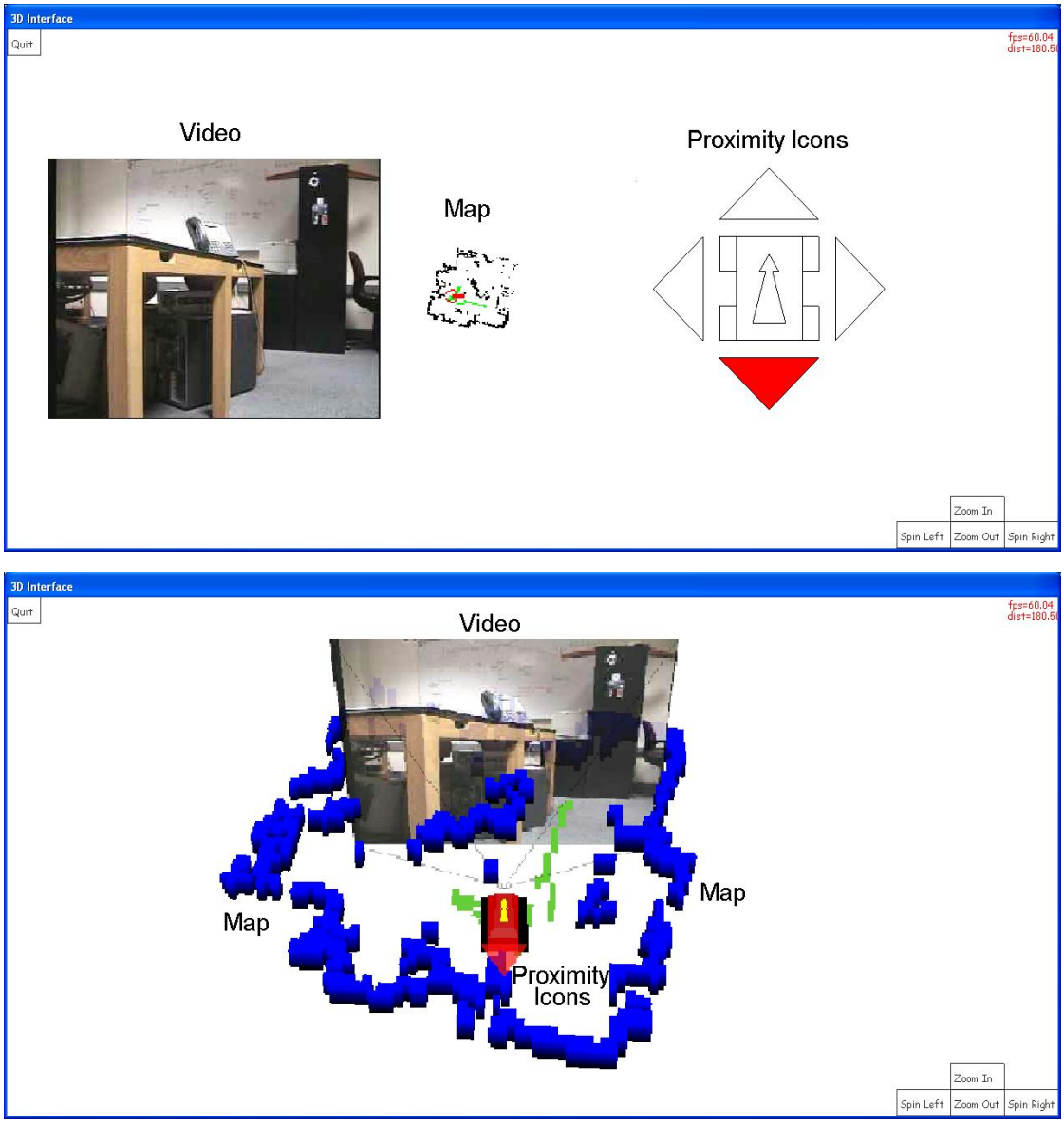


Figure 4.25: The 2D interface (top) and 3D interface (bottom) used for the second (real-world) video experiment.

Procedure

Before using the robot, operators are trained with the Unreal Tournament training maze that we used in previous experiments. While training, operators drove the simulated robot with a joystick for a few minutes with each of the five testing conditions (2D map-only, 2D map+video, video-only[13], 3D map-only, and 3D map+video). Upon completion of the training, the operators were moved to a different base station which communicates with the real robot.

For testing, we used a within-subjects counter-balanced design where each operator used all five conditions in a random order with the constraints that the 2D and 3D interfaces were used alternately and the conditions were counter-balanced on the order in which they were used. The experiment was setup such that an operator would drive the robot through the obstacle course with one condition, then at the end of the course an assistant would change the condition, turn the robot around, reset the map information, and start the next test. After every two conditions the robot was plugged in for three to five minutes to keep the batteries charged.

4.6.2 Results

Twenty-one participants were paid to navigate the ATRV-Jr robot with the five different conditions of information presentation. Participants were recruited from the Brigham Young University community, with most subjects enrolled as students. The first three participants were used as part of a pilot study to determine a sufficient complexity of the obstacle course and to determine how best to use the robot while maintaining a sufficiently high charge on the batteries, therefore, their results were not included as part of the analysis. Additionally, the robot's responsiveness to commands was adversely affected by low batteries in eleven of the testing conditions (out of 90) therefore, this data was also discarded.

One of the differences between this experiment and the simulated experiment is that the real robot has intelligence on board to protect itself from hitting

[13]We did not compare 2D and 3D video-only conditions because we found in the previous experiment that the video-only condition was similar for both the 2D and 3D interfaces.

	Map-only	Video-only	% Change	p-value
2D Interface	319s	243s	-24%	1.6×10^{-3}
3D Interface	227s	243s	7.2%	6.0×10^{-1}

Table 4.18: Average time to completion in the real-world experiment.

	Map-only	Video-only	% Change	p-value
2D Interface	46.9	38.8	-17%	6.6×10^{-1}
3D Interface	28.6	38.8	35%	2.4×10^{-2}

Table 4.19: Average number of times the robot took initiative to protect itself in the real-world experiment.

obstacles. For each test we record the number of times the robot acts to protect itself and discuss these results as *robot initiative*. We begin by comparing the map-only and the video-only conditions. We next discuss how the map+video condition compares to the map-only and video-only conditions. Through the discussion, statistical significance is determined using a paired, two-tailed t-test with $n = 18$ samples except as otherwise noted.

Map-only vs. Video-only

With the 2D interface, there was not a significant difference in the number of times the robot took initiative to protect itself with the map-only and video-only conditions (see Table 4.19), but there was a significant difference in the time taken to complete the task. In particular it was 24% faster to use the video-only condition as opposed to the map-only condition ($\bar{x}_{map} = 319$s, $\bar{x}_{video} = 243$s, $p = 1.6 \times 10^{-3}$, see Table 4.18).

With the 3D interface, there was not a significant difference in the time to completion with the map-only and video-only conditions (see Table 4.18), but the robot took initiative to protect itself 96% more frequently with the video-only condition than with the 3D map-only condition ($\bar{x}_{map} = 18.7$, $\bar{x}_{video} = 38.8$, $p = 2.4 \times 10^{-2}$, see Table 4.19).

The results of comparing the map-only condition to the video-only condition are different and nearly opposite of what we saw in the simulation experiment;

video seems to be more useful than it was in simulation and map information (at least in 2D interfaces) seems to be less useful. Most likely, the reason for the different results between the simulated and real experiments is that the environment in the real-world experiment provides more navigational cues that are visible in the video stream than the environment in the simulation experiment. In the simulation experiment, it was often the case that the video image was filled by a single wall and none of the edges of the wall were visible. Further, the path through the simulation maze doubled back on itself numerous times, so the operator could not see very far in front of the robot. In contrast, in the real-world experiment, the edges of obstacles were nearly always visible through the camera and the operator could see future parts of the maze as most obstacles were not taller than the height of the camera and there was only one 90 degree turn in the environment.

Map+video

When map and video information are combined using the 2D interface, we found the results to be similar to the video-only condition with negligible difference in the time to completion and the number of collisions.

When map and video information were combined using the 3D interface, we found the number of collisions to be similar to the map-only condition but we found that operators finished the obstacle course 9.6% faster with the map+video condition in comparison to the map-only condition ($\bar{x}_{map+video} = 205$s, $\bar{x}_{video} = 227$s, $p = 4.6 \times 10^{-2}$ see Table 4.20 and Figures 4.26 and 4.27).

This result is interesting when combined with the simulation results because it suggests that when useful navigational information is available in both the map and the video sets of information, the 3D interface supports the complementary nature of the information and can lead to an improved performance over the individual sets of information. In contrast, performance with the 2D interface seems to be constrained by the best one can do with an individual set of information (either map-only or video-only).

	2D Map-only	2D Map+video	% Change	p-value
Time to Completion (s)	319	247	-23%	2.7×10^{-2}
Average Robot Initiative	46.9	36.3	-23%	8.4×10^{-2}

	2D Video-only	2D Map+video	% Change	p-value
Time to Completion (s)	243	247	1.6%	5.0×10^{-1}
Average Robot Initiative	38.8	36.3	-6.7%	6.7×10^{-1}

	3D Map-only	3D Map+video	% Change	p-value
Time to Completion (s)	227	205	-9.6%	4.6×10^{-2}
Average Robot Initiative	28.6	24.8	-13%	3.9×10^{-1}

	2D Video-only	3D Map+video	% Change	p-value
Time to Completion (s)	243	205	-16%	1.3×10^{-2}
Average Robot Initiative	38.8	24.8	-36%	1.1×10^{-3}

Table 4.20: Comparison of the map+video condition to the map-only and video-only conditions from the real-world experiment.

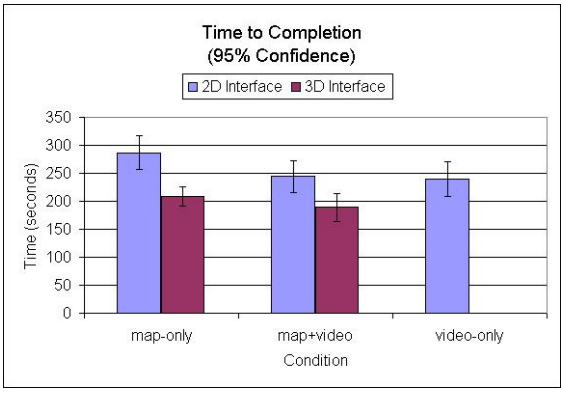


Figure 4.26: Time to completion for the five conditions in the real-world experiment.

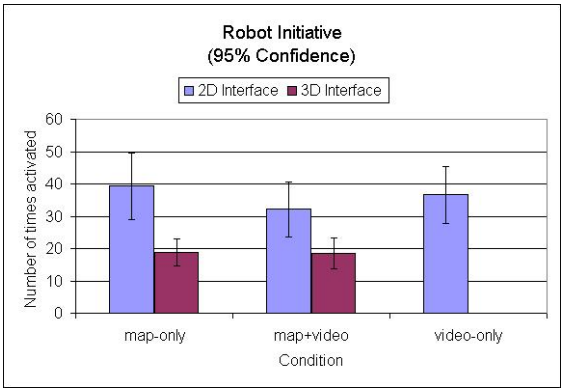


Figure 4.27: Average instances of robot initiative for the five conditions in the real-world experiment.

	Condition	2D Interface	3D Interface	% Change	p-value
Time to Completion (s)	map-only	319	227	-29%	4.5×10^{-3}
	map+video	247	205	-17%	1.6×10^{-2}
Average Instances of Robot Initiative	map-only	46.9	28.6	-39%	2.9×10^{-2}
	map+video	36.3	24.8	-32%	7.5×10^{-3}

Table 4.21: Comparison of the 2D and 3D interfaces with the map-only and map+video conditions in the real-world experiment.

2D vs. 3D

Similar to previous results, we found that operators performed significantly better with the 3D interface than with the 2D interface for similar conditions. In particular, with the map-only condition operators completed the task 29% faster and had 39% fewer instances where the robot took initiative to protect itself. With the map+video condition, operators completed the task 17% faster and had 32% fewer instances of robot initiative (see Table 4.21).

4.6.3 Discussion

To determine an ordering of the usefulness of the conditions, we define one condition to be better than another if all of the categories (time to completion or robot initiative) are at least non-significantly different and one of the categories

is significantly better. Conditions are considered equivalent if there is no statistical difference in either category of analysis.

According to this criteria we found that when using the 3D interface, the map+video condition is better than the map-only condition (because the task took less time), and the map-only condition is better than the video-only condition (because there were fewer instances of robot initiative). These results suggests that when there is useful navigational information in both the map and the video sets of information, integrating the information can yield better results than using either map or video individually.

We also found that when using the 2D interface, the map+video condition is similar to the video-only condition which are both better than the map-only condition (faster time).

Interestingly, these results are different from our simulation studies where we found the video-only condition to be significantly worse than the other conditions. One complaint among participants with the 2D interface was that the map was too small (although it was the same relative size as the previous experiment).

The results from the simulation experiments and the real-world experiments show that map-only conditions can be more useful than video-only conditions if the map resolution is of sufficient quality. Additionally, we have shown that video is helpful in environments where there are navigational cues in the video information, but video can diminish performance when there are no navigational cues and video is placed side-by-side to map information.

4.7 Conclusions

In this chapter we have presented a series of user studies that compare an operator's ability to navigate a robot with a conventional 2D interface and our 3D augmented-virtuality interface. We have compared the two interfaces in four domains, a) path following, b) map building, c) effect of video, and d) effect of delay.

Our results indicate that, in general, the 3D interface yields better performance than the 2D interface. In particular, we found that the 3D interface allows an

operator to finish the task about 25% faster, while moving the robot about 25% faster. Operators are also able to maintain a further average distance from walls, had nearly 80% fewer collisions, and spent a smaller percentage of their time in close proximity to walls. We also found that operators were better able to perform secondary tasks better and more accurately with the 3D interface. Subjectively, participants preferred the 3D interface to the 2D interface twenty to one and felt that they did better, were less frustrated, and better able to anticipate how the robot would respond to their commands.

The ability of the operator to stay farther from obstacles with the 3D interface is a strong indication of the operator's navigational awareness. There is a much lower rate of 'accidentally' bumping into a wall because the operator is more aware of the robot's proximity to obstacles, and the operator does a better job of maintaining a safety cushion between the robot and the walls in the environment.

In 2D, different sets of information seem to compete for the operator's attention, with the caveat that video tends to draw attention towards itself despite its relative usefulness. In other words, if video is more useful than a map, the performance when using both map and video will be similar to using the video alone. If the video is not useful (i.e. does not contain many navigational cues), using both map and video will be less productive than using only the map because the video draws the operator's attention as described by Kubey and Csikszentmihalyi [68].

In 3D, the different sets of information seem to complement each other with respect to operator's attention. It is generally the case that the operator can do better with both map and video information than video alone for navigation tasks. Additionally, the 3D map seems to always be very useful for navigating a robot.

For design purposes, integrating maps with video in a 3D perspective seems much better than presenting map and video side-by-side in a 2D perspective. Most likely this is because the maps are always visible, even if the operator pays too much attention to the video.

Chapter 5

Exploration User Studies

In human-robot interactions (HRI), the interface is the means by which an operator communicates with a remote robot. In order for the operator to issue correct and informed directives to the robot, the operator must understand the robot's situation within the environment. For the operator to have an awareness of the robot's situation, it is important that information from the robot and the remote environment be presented clearly to the operator.

Currently, many mobile robots for research and field applications implement pan-tilt cameras. Pan-tilt cameras allow an operator to overcome the limited field of view of stationary cameras by adjusting the orientation of the camera. An advantage of this is that the camera can be used to search an entire visual area in front of and to the sides of a robot without moving the robot. This is particularly useful in unstable environments such as urban search and rescue where too much movement might cause structural damage. A pan-tilt camera could also be useful in situations where there is significant visual information to the sides of the robot, but it is easier or more efficient to navigate the robot along a forward path instead of rotating the robot to see both sides. Some examples include warehouse inventory, surveillance, patrolling, or reconnaissance tasks.

Despite the theoretical usefulness of a pan-tilt camera, experience has shown that an adjustable camera orientation can contribute to an operator's poor situation awareness. Yanco et al. discussed situation awareness when four first responders teleoperated robots with some of their equipment [141]. In their studies they found that despite spending a significant amount of time acquiring situation

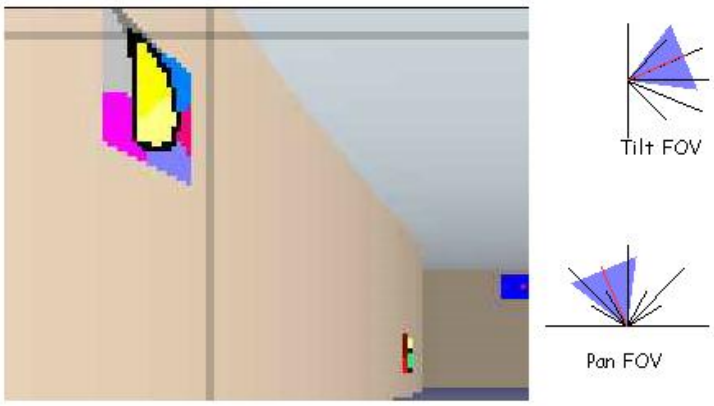


Figure 5.1: Visualizing the orientation of the pan-tilt camera using the 2D interface.

awareness by moving the camera, the first responders felt and demonstrated that they had poor situation awareness.

Furthermore, at a robot competition where participants compete in a mock search and rescue setting, the authors found that when the pan-tilt camera was used it often lead to confusion about the state of the robot. For example, some participants drove the robot forward thinking the robot would move in the direction the video was facing even though the camera was panned to the side [33].

When a robot's camera is panned to the side or tilted up, it is similar metaphorically to a human turning or tilting their head [48]. With the robot distant from the operator, it is important to convey to the operator the orientation of the camera or "head" of the robot with respect to the orientation of the robot body in order to satisfy the cues necessary for navigation while addressing the needs of exploration.

Conventional interfaces provide pan-tilt information to the operator via icons or horizontal and vertical bars overlaid on the video stream as shown in Figure 5.1. In contrast, the 3D interface provides visual support for the operator's comprehension of the camera orientation by rendering the image from the robot at an angle that corresponds to the camera orientation as shown in Figure 5.2.

In this chapter we present a series of user studies designed to evaluate an operator's ability to search an environment with a pan-tilt camera on a mobile robot

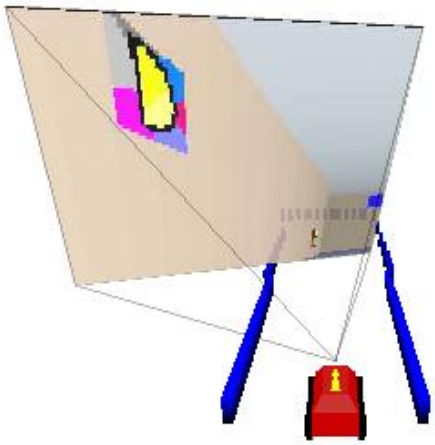


Figure 5.2: Visualizing the orientation of the pan-tilt camera using the 3D interface.

using a prototypical 2D interface and the 3D interface we have developed. First, we present a study that compares the 2D and 3D interfaces in a visual search task where the operators are asked to search all the walls of the environment. Second, we present a study that compares the usefulness of a movable camera to a stationary camera with both a 2D and a 3D interface. The final study compares how quickly an operator can find and identify things in real and simulated environments using the 2D and 3D interfaces.

5.1 Pan-Tilt Camera: 2D vs. 3D

The purpose of this experiment is to compare how well a prototype 2D interface and a 3D interface support the operator in a search task where a pan-tilt-zoom (PTZ) camera is used. In particular we are interested in how quickly operators can complete the task and how aware they are of the robot and its environment. We hypothesized that the PTZ camera is more useful for performing a search task when operators use the 3D interface as compared to a 2D interface.

5.1.1 Framework

To perform this experiment, we designed a task where operators were asked to identify and count pictures on the walls of a simulated environment. The simulated

environment contains light beige walls and the pictures of interest are dark blue so there is a significant contrast between the two. Throughout the environment there are sixteen different distracter pictures, none of which appear similar to the dark blue picture of interest. The idea is that the operators should not have to look closely to identify whether or not the picture is what they are looking for, they just need to observe all the walls in the environment.

The simulated environments used for this experiment have a simple underlying navigational structure, but are complicated by the addition of large rooms and dead-end hallways. Therefore, if the operator were to navigate the robot to look at every wall without moving the camera, the task would take significantly longer than if they navigate along the structure of the environment while moving the camera to look down hallways. The purpose for this design is to encourage the use of the PTZ camera.

The experiment is setup such that the operator first practices driving and using the camera with either the 2D or the 3D interface. Then after the operators are trained and feel comfortable with the robot controls, and any questions are answered, they are given a test using the same interface they practiced with but in a slightly larger environment. The participant is told to traverse the environment as quickly as possible while making sure to count all the blue signs. Upon completion of each test we record the number of pictures that were found and have the operator fill out a survey to subjectively evaluate their performance. The process is then repeated with the other (2D or 3D) interface. An example training world and test world are shown in Figures 5.3 and 5.4 respectively.

In these figures, the dots near the walls indicate the locations of the pictures and the dot and arrow in the middle of the hallway indicate the starting position and orientation of the robot. The path shows the simple underlying structure of the environment. The operator navigates the robot with a force feedback Microsoft Sidewinder Steering Wheel. The pan and tilt of the camera are controlled with buttons on the wheel, and the robot is controlled with the pedal and the steering wheel as shown in Figure 5.5.

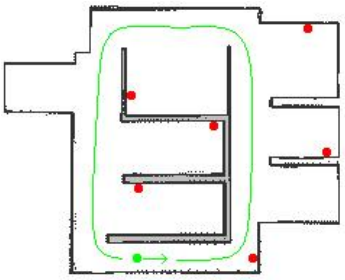


Figure 5.3: One of the training worlds used for the experiment.

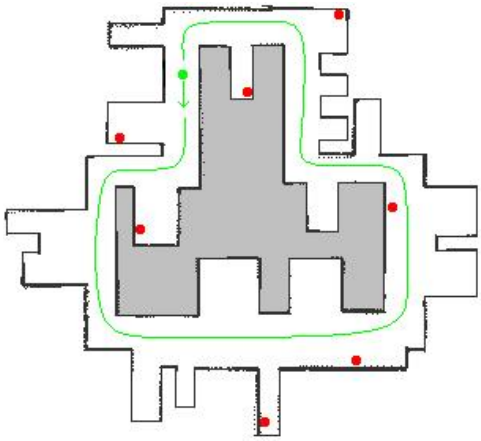


Figure 5.4: One of the testing worlds used for the experiment.

Figure 5.5: The Microsoft Sidewinder Wheel and pedal and the controls used for the experiment.

The two interfaces used for this study present the same information available to the operator, but in different forms. This information is video, map, robot pose, and camera orientation. The design of the software is such that all aspects of the human-robot system are the same except for the manner in which the information is presented to the operator. The 2D interface for this experiment is a prototype interface that displays relevant information similar to conventional human-robot interfaces [141, 16, 8], but it has been simplified to show only video, map, robot pose, and camera orientation. The 3D interface is similar to the ones used in the previous chapter with the addition of the pan-tilt information. The two interfaces used for this experiment are shown in Figure 5.6.

5.1.2 Results

Fifteen participants completed the tests for this experiment. Seven of the participants used the 3D interface first and eight used the 2D interface first. We continue with a discussion of the results as they relate to performance and workload and we present the subjective evaluations as they relate to the objective results. Throughout our discussion, statistical significance is obtained through a paired t-test with n = 15 samples.

Performance

Participants were able to complete the task nearly 20% faster, on average, with the 3D interface in comparison to the 2D interface ($\bar{x}_{3D} = 267$s, $\bar{x}_{2D} = 332$s, $p = 4.3 \times 10^{-2}$). There was no statistical difference in the average number of blue signs reported when using each of the interfaces ($\bar{x}_{3D} = 5.8$, $\bar{x}_{2D} = 5.5$, $p = 0.709$) Additionally, the average time in contact with walls was 89% less with the 3D interface than with the 2D interface ($\bar{x}_{3D} = 4.1$s, $\bar{x}_{2D} = 36.2$s, $p = 8.5 \times 10^{-4}$). Moreover, on average the robot was 20% further from a wall with the 3D interface than with the 2D interface ($\bar{x}_{3D} = 0.80$m, $\bar{x}_{2D} = 0.64$m, 1.0×10^{-3}). Table 5.1 summarizes the objective results from the experiment. The results also indicate that, on average, 10% of the time with the 2D interface was spent with the robot actually touching a

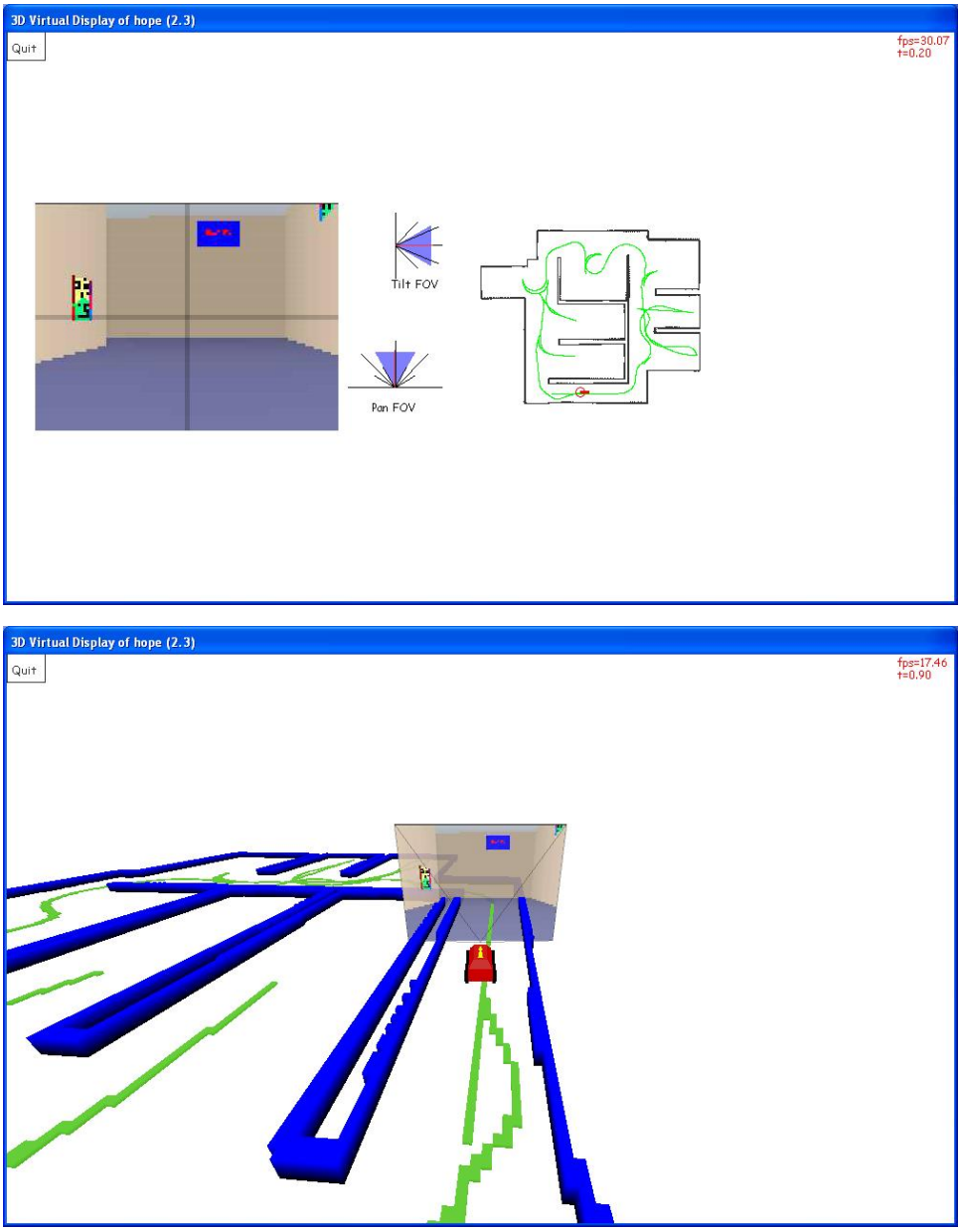


Figure 5.6: The interfaces used for the experiment, the 2D interface (top) and the 3D interface (bottom).

	2D Interface	3D Interface	% Change	p-value
Time to Completion (s)	332	267	-20%	4.3×10^{-2}
Items Found (error)	0.80	0.86	8.3%	7.7×10^{-1}
Time touching a wall (%)	10.4	1.60	-84%	4.5×10^{-5}
Nearest Obstacle (m)	0.64	0.80	25%	1.0×10^{-3}
Average Pan-tilt commands	2.49	3.75	50%	3.9×10^{-4}

Table 5.1: Summary of objective results for the 2D vs. 3D experiment.

.

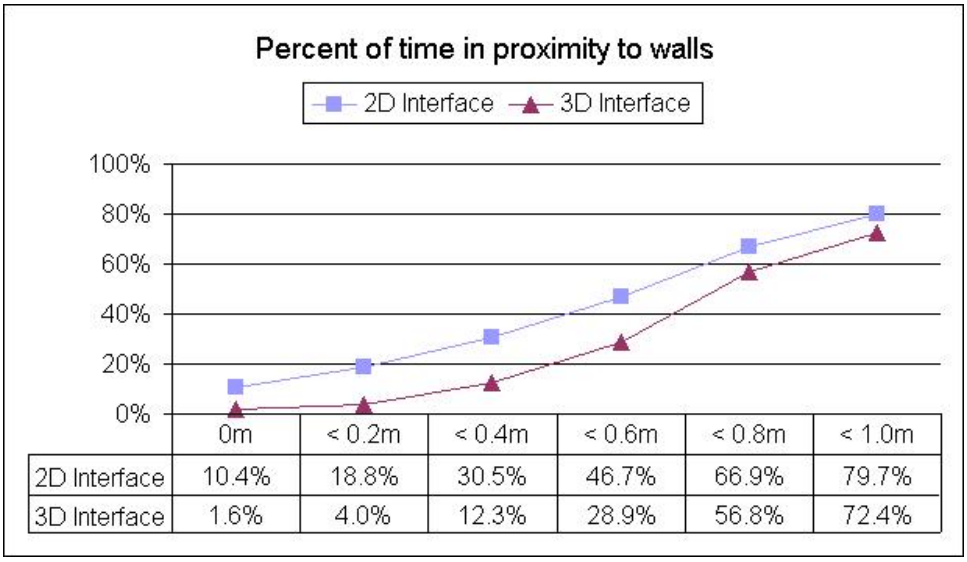


Figure 5.7: Percentage of time the robot is less than one meter from the wall.

wall, this is in contrast to only 1.6% of the time with the 3D interface (84% less, p $=4.5 \times 10^{-4}$). Figure 5.7 shows the percentage of time that an operator is navigating the robot within one meter of a wall. The graph shows that more time is spent closer to walls when using the 2D interface in comparison to the 3D interface.

There was a marginally significant learning effect observed between the group of participants that used the 3D interface first and those that used the 3D interface second. In particular, the group that used the 3D interface second finished 28% faster than the group that used the 3D interface first ($\bar{x}_{3D_{first}} = 314$s, $\bar{x}_{3D_{second}} =$ 226s, p $= 0.157$, $n_{first} = 7, n_{second} = 8$, unequal variance t-test). The standard deviation also decreased by more than half ($s_{3D_{first}} = 139$s, $s_{3D_{second}} = 54$s).

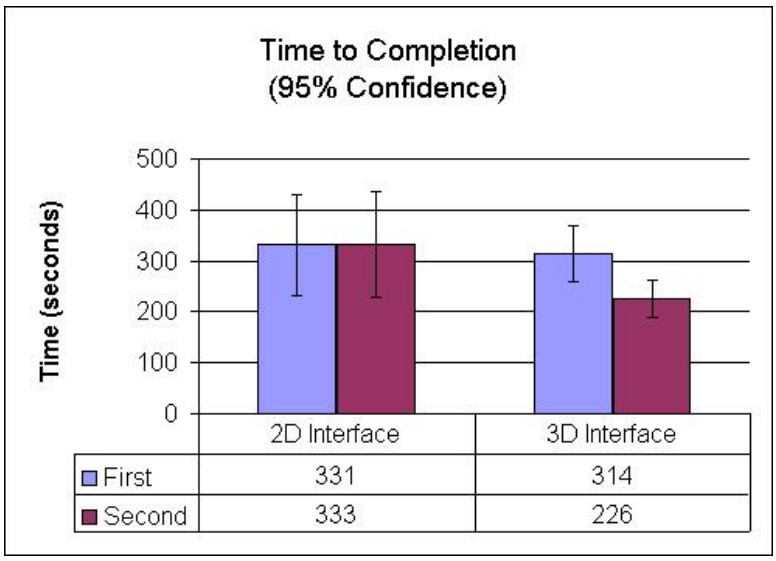


Figure 5.8: Average time to complete the experiment based on which interface was used first.

In contrast, there was no difference in time to completion between participants who used the 2D interface first and those who used it second ($\bar{x}_{2D_{first}} = 331$s, $\bar{x}_{2D_{second}} = 333$s, p = 0.968, $n_{first} = 8, n_{second} = 7$, unequal variance t-test). There was, however, a decrease in the standard deviation of nearly half ($s_{2D_{first}} = 142$s, $s_{2D_{second}} = 74$s); see Figure 5.8.

The decrease in the standard deviations of the second interfaces suggests that the operators have improved their ability to use the robot as would be expected with more experience. The improvement in the 3D interface when it is used second supports this. However, we would also expect to see a more substantial increase in the performance of the 2D interface when it is used second. The reason we do not see this improved performance in 2D is because the interface is simply harder to use.

Workload

Workload was difficult to measure because of the complexity of the task. Each participant had their own unique way of using the robot and the camera to observe the environment. Therefore, measures involving the robot control (such as steering wheel bandwidth, average velocity, average angular velocity, or behavioral

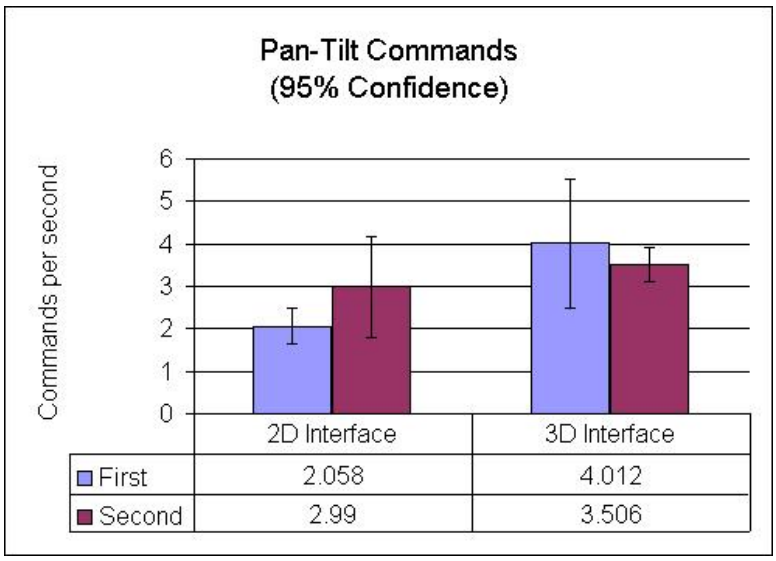


Figure 5.9: The number of pan and tilt commands issued to the robot per second based on which interface was used first.

entropy) did not reveal useful comparisons of the workload of the operator. Instead, we looked at the number of pan and tilt commands that the operator issued to the robot. The results show that, on average, the operator was able to issue 3.7 commands per second with the 3D interface in comparison to only 2.5 commands per second with the 2D interface. This indicates that the user was able to issue 33% ($p = 1.0 \times 10^{-3}$) more camera movement commands with the 3D interface while finishing the task faster. Additionally, when the 3D interface was used first. there were nearly twice as many commands issued as when the 2D interface was used first. This suggests that the PTZ camera was more easily used with the 3D interface and that it required more effort to use the camera with the 2D interface (see Figure 5.9).

The difference in the rate that pan-tilt commands were issued also suggests that participants felt more comfortable with their ability to navigate the robot with the 3D interface than the 2D interface, so more time could be spent manipulating the camera.

	2D	3D	% Change	p-value
Required effort	6.9	4.9	-30%	5.8×10^{-3}
Difficult to learn	5.9	3.6	-39%	3.9×10^{-4}
Effect of camera pose	6.4	3.7	-43%	6.9×10^{-3}
Confidence in robot	6.5	8.1	25%	5.4×10^{-2}
Comprehend camera pose	6.3	8.1	27%	4.0×10^{-2}

Table 5.2: Summary of the subjective results for the 2D vs. 3D experiment.

.

Subjective Evaluations

Subjective evaluations were obtained through surveys following each participant's use of each interface. The surveys indicate that, in general, participants felt that the 3D interface required less effort to use and was easier to learn. The participants also noted that they better understood the orientation of the camera and that panning and tilting the camera affected their navigation abilities less with the 3D interface than with the 2D interface. Table 5.2 and Figure 5.10 summarize the results of the survey. Fourteen of the fifteen participants felt they did better with the 3D interface and preferred it over the 2D interface. Only two of the users thought the 2D interface was more intuitive, but even one of these users commented, "It was confusing to have the map and video integrated. If I spent more time with [the 3D interface] I would probably prefer it".

The subjective evaluations match well with the objective evaluations of the experiment. Our discussion of the different workload between the interfaces is backed up by the surveys where, on average, the operators ranked the required effort as two points lower (on a scale of 1-10) with the 3D interface ($\bar{x}_{2D} = 6.9$, $\bar{x}_{3D} = 4.9$, $p = 5.8 \times 10^{-3}$). Participants also ranked the effect of the camera orientation on driving as almost three points lower with the 3D interface ($\bar{x}_{2D} = 6.4$, $\bar{x}_{3D} = 3.7$, $p = 6.9 \times 10^{-3}$). The surveys also revealed a higher confidence in robot movement and better comprehension of the camera orientation with the 3D interface.

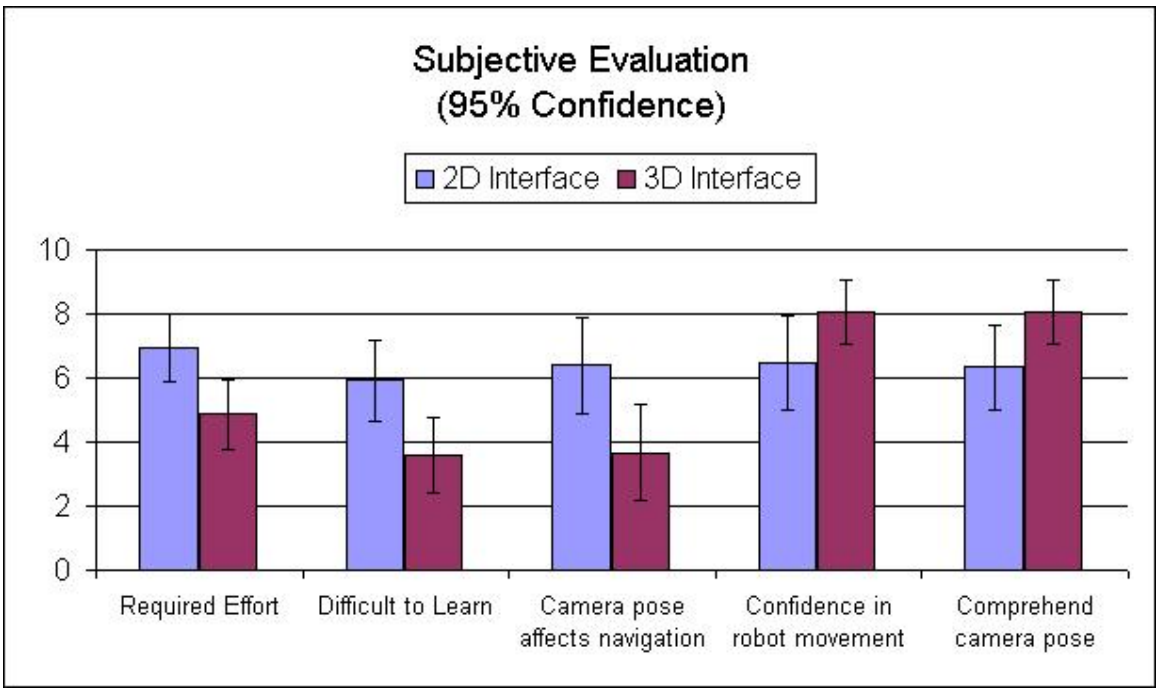


Figure 5.10: Results from the subjective evaluation of the interfaces.

5.1.3 Discussion

The manner in which the operators used the PTZ camera varied significantly even between the first and second tests for each participant. The reason for this is that the training session was not long enough and the operators were still experimenting and learning how to use the pan and tilt on the camera throughout the first experiment and into the second experiment.

These results suggest that the 3D interface makes it easier to use a PTZ camera than the 2D interface. In particular, operators were better able to keep the robot away from walls and finish the task faster—both of which are important for exploration tasks.

5.2 When to Use a Pan-Tilt Camera

One question when using a robot to perform search tasks is whether or not it is more efficient to use a pan-tilt camera or just maneuver the robot to look in various directions. The previous experiment showed that the 3D interface was

more useful than the 2D interface when using a pan-tilt (PTZ) camera, but it did not reveal whether a PTZ camera was useful in comparison to not having one. The purpose of this experiment is to compare the usefulness of the 2D and 3D interfaces while searching for things with and without a pan-tilt camera. We hypothesized that, for both interfaces, it is more efficient to use a pan-tilt camera than to turn the robot in certain environments.

5.2.1 Framework

For this experiment, we created two simple simulation environments for driving the robot. Both environments have a box shape with six dead-end hallways on the outside and six dead-end hallways on the inside of the environment and are designed to exploit the use of a pan-tilt camera. The environments are shown in Figure 5.11. At the end of 8 of the 12 dead-end hallways are flags that operators were asked to find. The operators were told that they only had to look at the flag and that they did not need to drive over the flag or go near the flag.

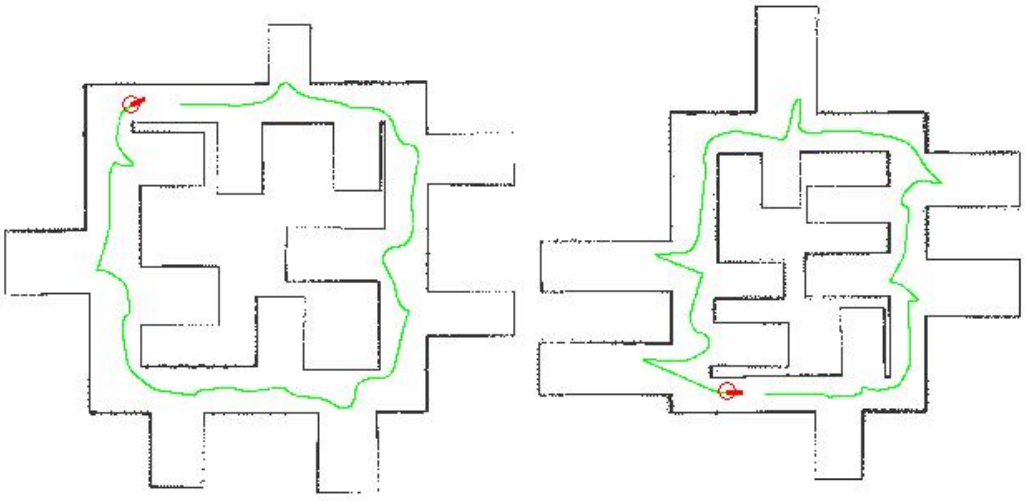


Figure 5.11: Simulation environments used in the St. Louis Science Center exploration tasks.

This experiment took place as part of a week-long special event in 'Cyberville' of the St. Louis Science Center between April 30th and May 5th, 2005. The organization of the experiment and the time spent with each volunteer was largely constrained by the volume of traffic at the Science Center. In particular, if there were a lot of people waiting to participate, we kept each person for only one test instead of two. Some participants used only a robot with a movable (pan-tilt) camera and others used only a robot with a stationary (non pan-tilt) camera. Operators used the Microsoft Sidewinder Steering Wheel to drive the robot and control the camera. Due to limitations on time available with each individual, participants were not allowed to practice driving the robot. The instructions were explained and when they were ready the test began.

Participants who used a robot with a stationary camera were advised that a good strategy for using the robot was to drive forward then, when the robot reaches a hallway, turn the robot to look down the hallway, observe whether or not there is a flag, then turn back and continue around the maze. Participants who used a robot with a movable camera were advised to keep the robot in the middle of the main hall and turn the camera to look down each side hall as opposed to turning the robot to look down the side halls.

5.2.2 Results

There were 88 participants who drove the robot with a stationary camera (half with the 2D interface and half with the 3D interface). Additionally, 44 more participants drove the robot with both the 2D and 3D interfaces with a movable camera. Participants were between 10 and 46 with an average and median age of 18. Of the participants, 28 (64%) finished the 2D stationary camera test, 37 (84%) finished the 3D stationary camera test, 25 (57%) finished the 2D movable camera test, and 39 (89%) finished the 3D movable camera test. The results of the experiments where the operator did not complete the task were not used for the analysis. Throughout the discussion, statistical significance is obtained with an unequal-variance t-test unless otherwise indicated.

2D	Stationary Camera	Movable Camera	% Change	p-value
Time to Completion (s)	249	250	0.30%	9.7×10^{-1}
Average Velocity (m/s)	0.43	0.25	-42%	2.1×10^{-4}
Distance Traveled (m)	107.8	63.1	-41%	3.1×10^{-6}

Table 5.3: Objective results for the 2D interface with a movable and a stationary camera.

Stationary camera vs. moving camera

With the 2D interface, participants averaged similar times to completion with a stationary camera as with a movable camera. This is interesting, because although the robot traveled 41% less distance with the movable camera than with the stationary camera ($\bar{x}_{2Dno_ptz} = 107.8$m, $\bar{x}_{2Dptz} = 63.1$m, $p = 3.1 \times 10^{-6}$, $n_{noptz} = 25, n_{ptz} = 28$), the average velocity was also 42% slower ($\bar{x}_{2Dno_ptz} = 0.43$m/s, $\bar{x}_{2Dptz} = 0.25$m/s, $p = 2.1 \times 10^{-4}$, $n_{noptz} = 25, n_{ptz} = 28$). So, although the robot travels less distance with the movable camera, it travels significantly slower and the task requires the same amount of time as when using the stationary camera (see Table 5.3). Figure 5.12 illustrates actual paths used by the robot with and without a movable camera.

The 3D interface, on the other hand demonstrated an average time to completion 13% faster with the movable camera in comparison to the stationary camera ($\bar{x}_{3Dno_ptz} = 181$s, $\bar{x}_{3Dptz} = 157$s, $p = 4.4 \times 10^{-2}$, $n_{noptz} = 37, n_{ptz} = 39$). The reason for this is that although the robot with the movable camera travels 41% less distance, on average, than the robot with the stationary camera ($\bar{x}_{3Dno_ptz} = 110.4$m, $\bar{x}_{3Dptz} = 65.6$m, $p = 4.8 \times 10^{-8}$, $n_{noptz} = 37, n_{ptz} = 39$), the average velocity only drops by 31% ($\bar{x}_{3Dno_ptz} = 0.61$m/s, $\bar{x}_{3Dptz} = 0.42$m/s, $p = 4.8 \times 10^{-3}$, $n_{noptz} = 37, n_{ptz} = 39$). Therefore, the time to completion is faster because the decrease in distance traveled is more than the decrease in average velocity (see Table 5.4).

Another related observation is the percentage of time that the operator spends moving the robot (not the camera). For the 2D interface, 27% less time is spent driving the robot when the movable camera is used as opposed to the stationary

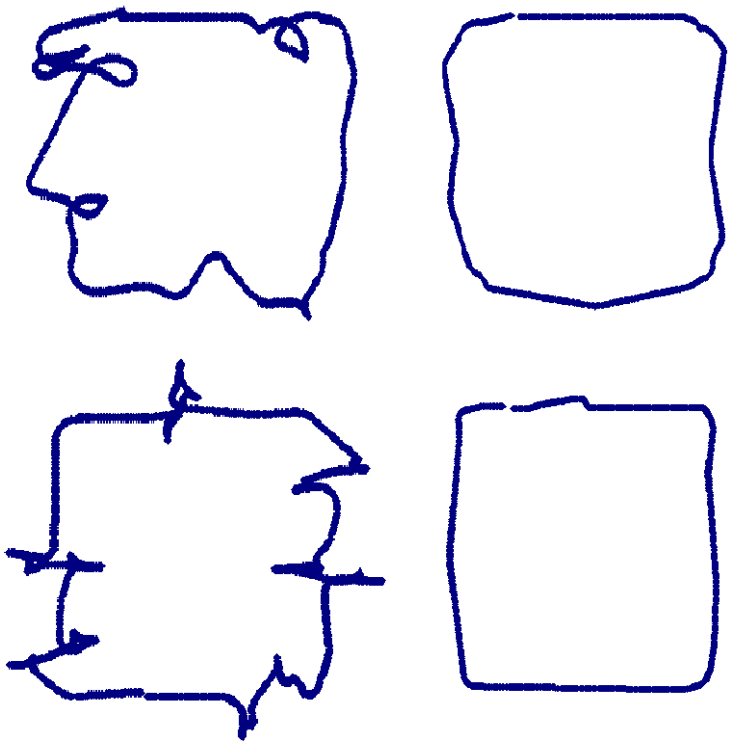


Figure 5.12: Paths taken with different interfaces and camera conditions. Clockwise from bottom left: 3D-noPTZ, 2D-noPTZ, 2D-PTZ, 3D-PTZ.

3D	Stationary Camera	Movable Camera	% Change	p-value
Time to Completion (s)	181	151	-13%	4.4×10^{-2}
Average Velocity (m/s)	0.607	0.416	-31%	4.8×10^{-3}
Distance Traveled (m)	110.4	65.6	-41%	4.8×10^{-3}

Table 5.4: Objective results for the 3D interface with a movable and a stationary camera.

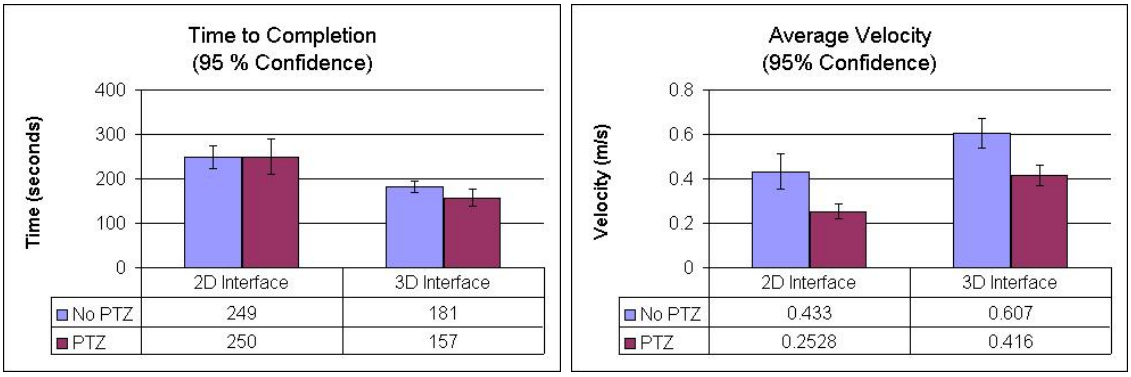


Figure 5.13: Time to completion and average velocity for the different interfaces and cameras.

camera ($\bar{x}_{2Dno\_ptz} = 87\%$, $\bar{x}_{2Dptz} = 63\%$, $p = 6.7 \times 10^{-8}$, $n_{noptz} = 25, n_{ptz} = 28$). For the 3D interface, 23% less time is spent driving the robot when the movable camera is used in comparison to the stationary camera ($\bar{x}_{3Dno\_ptz} = 92\%$, $\bar{x}_{3Dptz} = 71\%$, $p = 3.2 \times 10^{-10}$, $n_{noptz} = 37, n_{ptz} = 39$). Charts representing the time to completion and the average velocity for the different interfaces and cameras are shown in Figure 5.13.

When considering navigational awareness, operators tended to drive more safely with the movable camera than without for both interface conditions; however, the differences are more pronounced with the 3D interface. In particular, with the 2D interface there were 46% fewer collisions under the movable camera condition ($\bar{x}_{2Dno\_ptz} = 11.1$, $\bar{x}_{2Dptz} = 6.0$, $p = 7.9 \times 10^{-2}$, $n_{noptz} = 25, n_{ptz} = 28$) and the percentage of time in contact with a wall was 45% lower ($\bar{x}_{2Dno\_ptz} = 6.4\%$, $\bar{x}_{2Dptz} = 3.5\%$, $p = 7.6 \times 10^{-2}$, $n_{noptz} = 25, n_{ptz} = 28$). There was virtually no difference in the average distance to the nearest wall under the two camera conditions when using the 2D interface (see Table 5.5).

With the 3D interface, there were 86% fewer collisions when using the movable camera as opposed to the stationary camera ($\bar{x}_{3Dno\_ptz} = 4.11$, $\bar{x}_{3Dptz} = 0.56$, $p = 2.4 \times 10^{-2}$, $n_{noptz} = 37, n_{ptz} = 39$) and the percentage of time in contact with a wall was 86% lower as well ($\bar{x}_{3Dno\_ptz} = 2.8\%$, $\bar{x}_{3Dptz} = 0.40\%$, $p = 2.1 \times 10^{-3}$, $n_{noptz} = 37, n_{ptz} = 39$). Additionally, the robot maintained a distance 8% farther from

2D	Stationary Camera	Movable Camera	% Change	p-value
Average Collisions	11.1	6.04	-46%	7.9×10^{-2}
Time Touching a Wall (%)	6.4	3.5	-45%	7.6×10^{-2}
Nearest Obstacle (m)	0.80	0.81	1.4%	7.7×10^{-1}
Navigation time (%)	87	63	-27%	6.7×10^{-8}

Table 5.5: Navigation awareness results for the 2D interface with a movable and a stationary camera.

3D	Stationary Camera	Movable Camera	% Change	p-value
Average Collisions	4.11	0.56	-86%	2.4×10^{-3}
Time Touching a Wall (%)	2.8	0.40	-86%	2.1×10^{-3}
Nearest Obstacle (m)	0.88	0.95	8%	2.1×10^{-3}
Navigation time (%)	92	71	-23%	3.2×10^{-10}

Table 5.6: Navigation awareness results for the 3D interface with a movable and a stationary camera.

walls with the movable camera than with the stationary camera ($\bar{x}_{3Dno_ptz} = 0.88$m, $\bar{x}_{3Dptz} = 0.95$m, $p = 2.1 \times 10^{-3}$, $n_{noptz} = 37, n_{ptz} = 39$); see Table 5.6.

2D vs. 3D

Similar to previous results, participants who used the 3D interface performed considerably better than those who used the 2D interface. Table 5.7 summarizes the comparison between the 2D and the 3D interface.

Another observation with respect to navigational awareness is the percentage of time that the robot is in proximity to walls. Figure 5.14 illustrates the percentage of time that the robot is within a given distance of a wall. The figure shows that the 3D interface with and without a movable camera maintains a safer distance from walls than the 2D interface with either camera condition. This suggests that the 3D interface tends to help the operator keep the robot safer regardless of whether or not a pan-tilt camera is available.

	Camera	2D Interface	3D Interface	% Change	p-value
Time to Completion (s)	Stationary	249	181	-27%	4.3×10^{-5}
	Movable	250	157	-37%	7.7×10^{-7}
Average Velocity (m/s)	Stationary	0.433	0.607	40%	6.3×10^{-2}
	Movable	0.253	0.416	65%	7.7×10^{-7}
Average Collisions	Stationary	11.1	4.11	-63%	9.9×10^{-3}
	Movable	6.04	0.56	-91%	1.9×10^{-3}
Nearest Obstacle (m)	Stationary	0.80	0.88	10%	1.2×10^{-2}
	Movable	0.81	0.95	17%	9.2×10^{-6}
Navigation Time (%)	Stationary	87	92	7%	3.4×10^{-3}
	Movable	63	71	12%	6.1×10^{-2}

Table 5.7: Summary of results comparing the 2D and 3D interfaces when used with movable and stationary cameras.

5.2.3 Discussion

In this experiment, we found that using a movable camera did not improve the time to complete a search task when using the 2D interface. This result is somewhat surprising considering that the environment for the experiment was designed to exploit the use of a pan-tilt camera. The reason for the similar performance with the 2D interface is that operators spend a smaller percentage of their time moving the robot (which results in a lower average velocity) when a movable camera is available. Even though the underlying structure of the environment was very simple, operators tended to stop driving the robot when they were using the camera to look down hallways.

With the 3D interface, participants were able to finish the task somewhat faster when using a movable camera because they spent a larger portion of their time navigating the robot, even while using the camera. This led to a faster average velocity.

We also observed that operators tended to stay further from walls and spend less time in proximity to walls when using a movable camera in comparison to a stationary camera. There were also significantly fewer collisions when using the

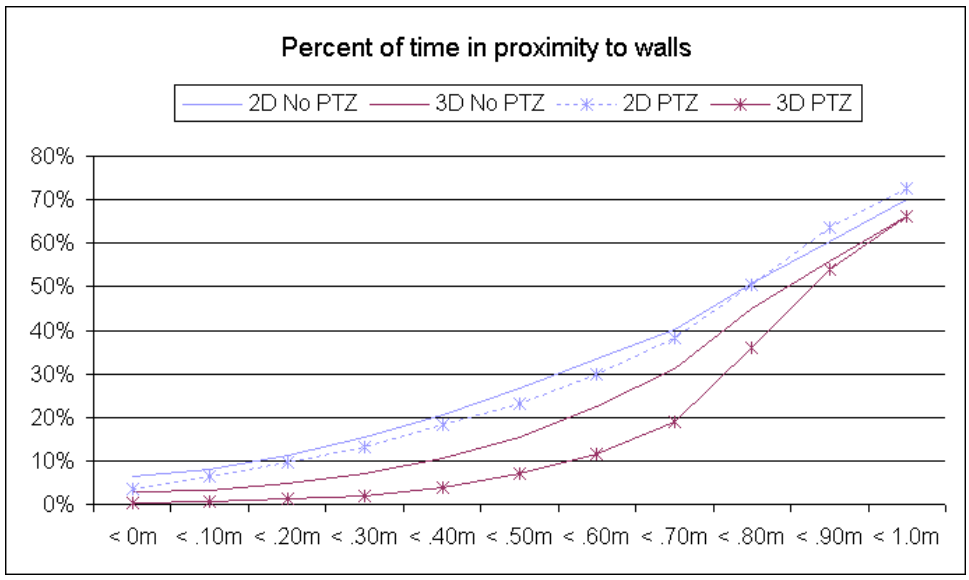


Figure 5.14: The percentage of time the robot is within a given distance of a wall.

movable camera in comparison to the stationary camera. These differences were more pronounced with the 3D interface than with the 2D interface.

5.3 Find the Foo Experiment

Previously, we have shown that an operator can navigate a robot faster and safer and use a movable camera more efficiently with a 3D interface as opposed to a 2D interface. The purpose of this experiment is to test how well operators can use a robot and a pan-tilt camera to find and identify things in an environment. We hypothesized that the 3D interface would provide better performance for exploration than the 2D interface.

5.3.1 Framework

This experiment is designed to employ both a simulation study and a real world study. The purpose for integrating the real and simulated portions of the experiment is that we found the real robot works best when we drive it for up to 30 minutes, then let it recharge for 30 minutes before driving it again. While the

robot is recharging, we have participants perform a similar exploration experiment in simulation.

The simulation experiment

The simulation experiment is setup as a scenario where a robot is used to explore an underground cave before sending in rescuers. We used the Unreal Tournament game engine for our simulator and we used the Unreal Tournament level editor to create a maze that has the appearance of an underground cave that is filled with boxes and prison cells. The task of the operator is to drive the robot through the environment and identify places and victims. If victims are found, we are interested in where they are located and what they are wearing. The places are identified by whether or not jail bars are visible.

The main level of the environment for this experiment is setup as a circular maze with numerous dead-end hallways protruding out of the center of the maze. At the end of each hallway is a jail cell that is partially obscured by many boxes. The normal width of each hallway is 4 meters across, but has been reduced to 2-3 meters because of the numerous boxes in the environment. Figure 5.15 shows a map of the main floor.

In the center of the maze, there is a room with a glass floor and open ceiling to allow the operator to see the levels above and below the main floor through the robot camera. Above and below the main level there are eight more areas that also need to be examined and classified by the operator. Additionally, the center of the main floor is covered with boxes, which make visibility more difficult and requires the operator to maneuver the robot in order to see all the information on the different levels. Figure 5.16 shows some pictures of the simulated environment.

In some of the jailed rooms, we placed "victims" for the robot to find. The victims we used as prisoners are 3D models created by Michael Lewis for the USAR-simulator (see Figure 5.17). There is at most one victim in each cell and they are placed in either a standing, sitting, or reclining position. Each victim is visible from someplace that the robot can reach.

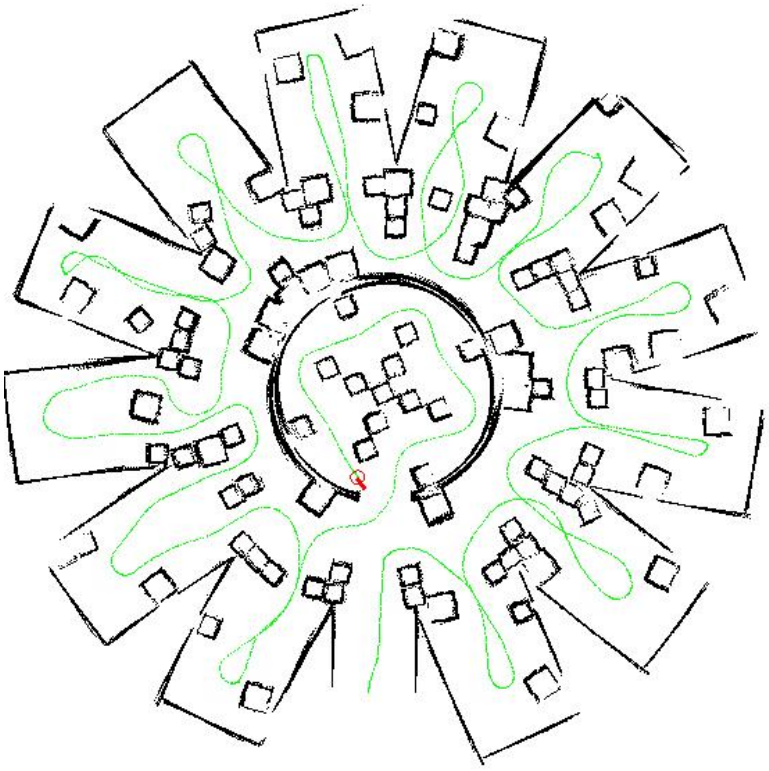


Figure 5.15: Map of the main floor of the simulation environment.

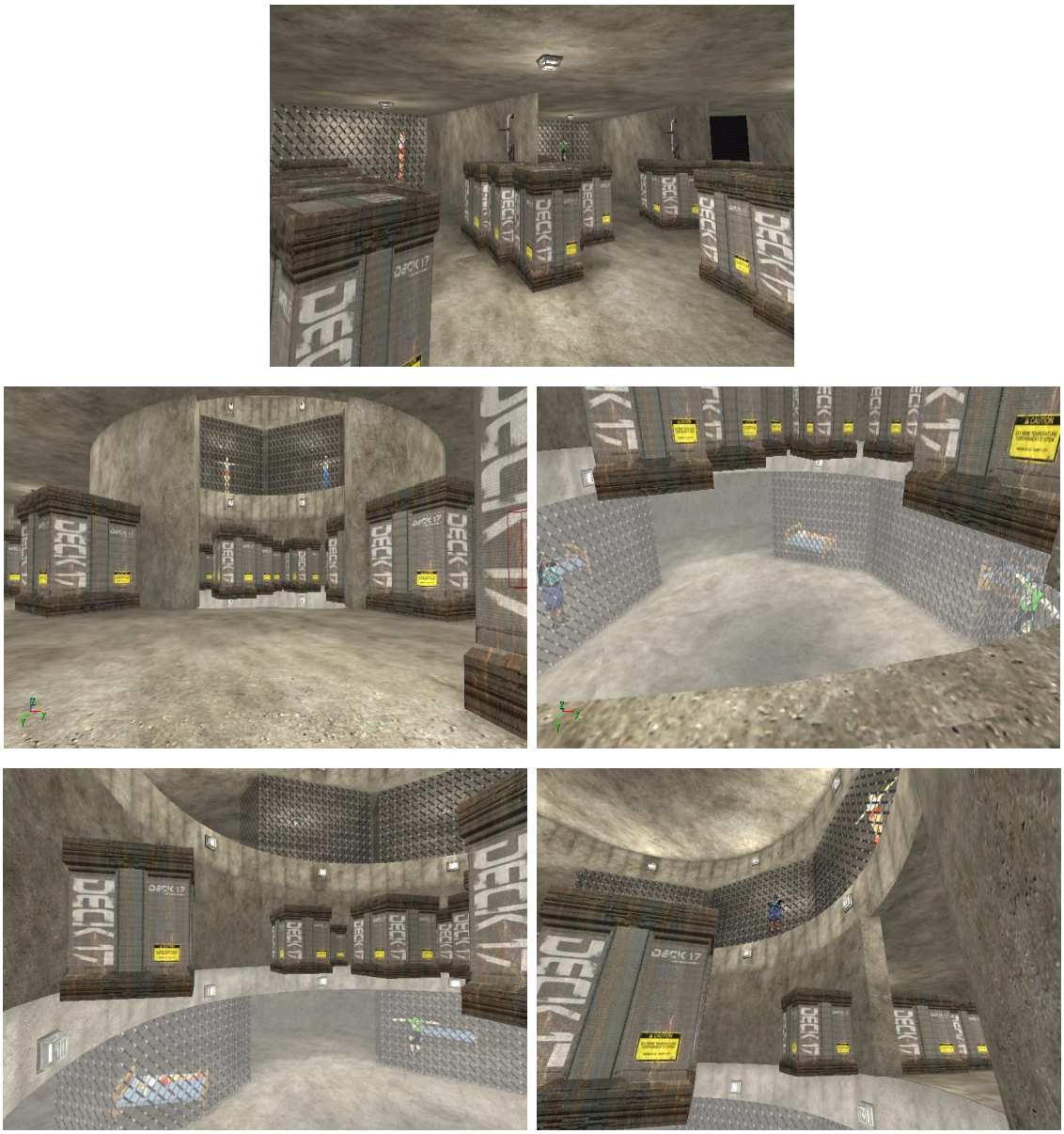


Figure 5.16: Images of the environment used for the simulation experiment.

Figure 5.17: 3D models for victims used in the simulated exploration experiment.

The maze is significantly larger than the previous simulated experiments for two reasons: first, we are not explicitly testing navigation, so we made it somewhat easier to navigate, and second, we wanted the task to be sufficiently challenging that not all participants would be able to search the entire environment in the time allotted for the experiment.

The real world experiment

For the real world exploration task, we again converted part of the second floor of the Computer Science Department at Brigham Young University into an obstacle course for the robot to travel through. The environment is similar to the one used in the previous chapter with the difference that we now use fewer obstacles and the obstacles are in larger groups. The purpose of this design is so that the operator can use the camera to search the group of obstacles to the side of the robot as it is driven forward. After searching one group of obstacles, the robot must be navigated to avoid the next group of obstacles and the camera is again used to search the obstacles on the side of the robot.[1]

The length of the real world course is about 50 meters and there are 5 different "cache" areas where objects were hidden; each cache area is between 3–10

[1]We used the ATRV-Jr robot, algorithms, and software described in Chapter 4.

meters in length. Objects are placed on top of, inside of, or between boxes, or placed on the ground. Most of the objects are placed in such a way that they are only visible when the camera is panned and/or tilted in the right direction. Further, to correctly identify some objects, it was occasionally necessary to use the zoom capability of the camera. (We discuss the technology for preesnting zoom in Chapter 6.) All of the objects are visible when the robot is in the center of the hallway.

The purpose of this experiment is to have the operator find and identify objects in an environment that might be of interest. Participants were told that cardboard boxes and Styrofoam were not interesting, but other things were. In each experiment there were 12 items hidden throughout the environment (2-3 in each group of boxes). Some of the objects are shown in Figure 5.18. The larger objects are the size of a book and the smaller objects are the size of a checker or quarter.

Figure 5.18: Some of the objects used in the real world search experiment.

Procedure

Operators first performed the simulation tests and then performed the real world tests. Before beginning the simulation tests, operators were given a chance to practice driving in an environment similar to the one used for testing. This was done so the operator would be familiar with how to look for victims and where victims are located. Operators are given both the 2D and 3D interfaces during training and are allowed up to 15 minutes to practice driving the robot and familiarizing themselves with the robot controls. For both the simulation and the real world tests, the robot is controlled with a Microsoft Sidewinder II force-feedback joystick. The joystick and controls are shown in Figure 5.19.

Figure 5.19: The joystick and controls used for the exploration experiments.

During training, the task and requirements of the experiment are explained to the operators along with strategies that might help them finish the task faster. In particular, participants were told that the environment contained numerous jail cells and their task was to observe and report the contents of each cell. Participants were asked to report whether or not there was a cell (bars were present), whether a cell

was occupied or unoccupied, and the primary colors of the clothes of any victims. Additionally, participants were asked to report their findings in order (e.g. "Starting from the right, I see a blue shirt victim, an empty cell, no cell, a green shirt victim, a white shirt victim," etc). The findings are reported to an assistant who wrote down what the operator said they had found. Participants were also given the advice that they could accomplish the task faster if they could drive the robot while the camera was not centered in front of the robot. The purpose of sharing this advice is that we wanted the participants to use the camera for the experiment and not rely on moving the robot to search the environment.

Operators were told to move the robot through the environment as quickly as possible because each simulated test lasts only six and a half minutes. Participants were also directed to begin their search with the outer portion of the maze and once that was complete proceed towards the inner area of the maze. This was done so that paths traversed would be similarl and could therefore be compared.

After training, we performed counter-balanced tests for both the 2D interface and the 3D interface. Each interface displayed video, map, robot pose, and camera pose. The interfaces used for this experiment are shown in Figure 5.20. Upon completion of the simulation portion of the experiment, we shut down the simulator and started up the base station for communicating with the real robot.

Before testing with the real robot, we gave each participant a chance to practice driving the robot through a training section of the maze. This was beneficial because although the controls were exactly the same as in simulation, there were some nuanced differences between controlling the real and simulated robots. For example, one participant claimed that they were "more worried about hitting things in the real world." Additionally, the force-feedback activated by the joystick when the robot got too close to walls startled some participants. The training section of the maze allowed participants to experiment with the intelligence on the robot and get an idea of the type of environment they would be exploring and how we hid obstacles in the environment.

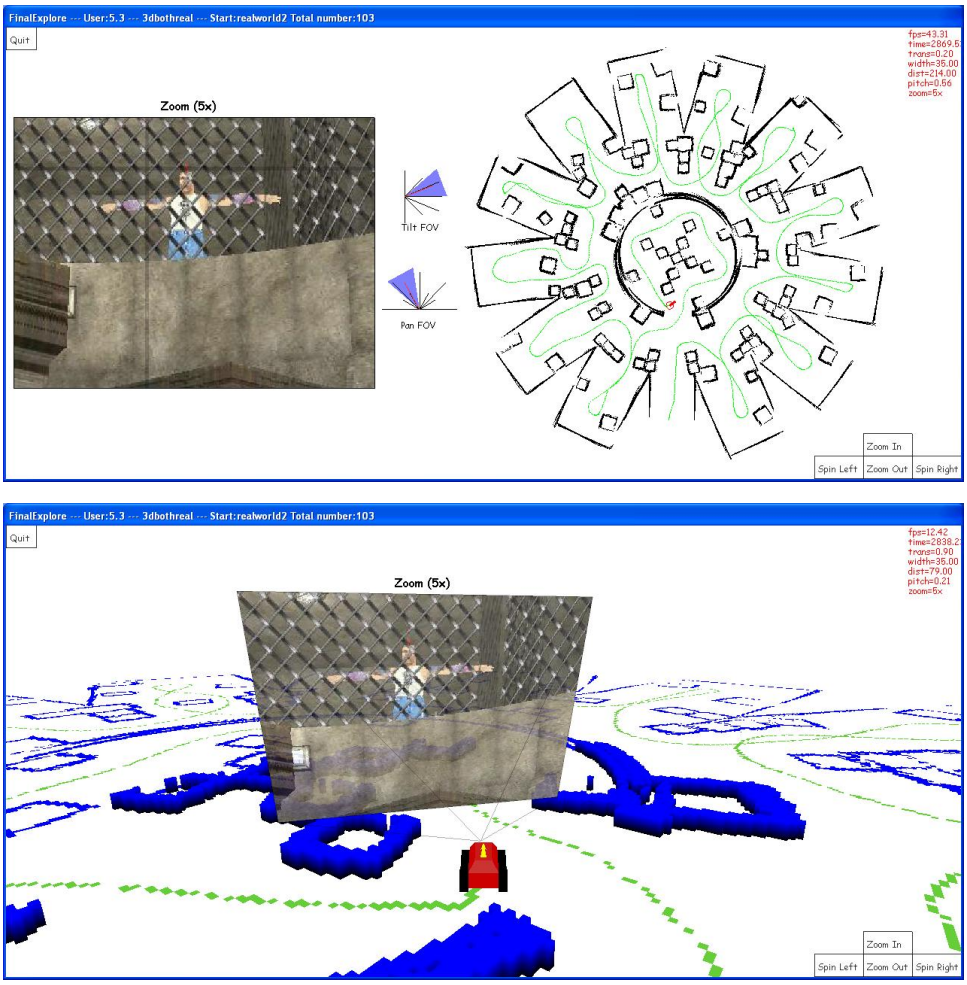


Figure 5.20: The 2D (top) and 3D (bottom) interfaces we used for the exploration experiment.

After training, the operators drove the robot to the start position of the testing environment and began the real world tests. For testing we used a within-subjects counter-balanced design where each operator drove with both the 2D interface and the 3D interface. The experiment was set up such that the operator would drive the robot through the obstacle course with one interface, then at the end of the course an assistant would change the interface, turn the robot around, reset the map building algorithms, change the hidden objects and their locations, and start the next test. Upon completion of the two real world experiments, the robot was plugged in until the next participant's real world tests.

5.3.2 Results

Eighteen participants were paid to perform this exploration experiment with both a simulated and a real robot. Participants were recruited from the Brigham Young University community with most subjects enrolled as students. During the real-robot tests, there was one instance where the robot powered off part way through the second test. The results from this test are not included in the analysis, but the completed portions of the experiment are still used. Results from the simulation portion of the experiment are presented first followed by results from the real world portion of the experiment. Throughout the following discussion, statistical significance is obtained with a paired, two-tailed t-test with n=18 samples in the simulation results and n=17 samples in the real world results unless otherwise specified.

Simulation

On average, participants correctly identified 19% more places in the environment with the 3D interface than with the 2D interface ($\bar{x}_{2D} = 18.1$, $\bar{x}_{3D} = 21.6$, $p = 1.1 \times 10^{-2}$). Since this experiment had a time limit, there was not a significant difference in the average time to completion, however, there were 6 participants (33%) who finished the task before the time expired using the 3D interface, whereas only 3 (17%) participants finished the task before the time expired using the 2D interface. One useful measure of performance is the average time it took to identify a place

	2D Interface	3D Interface	% Change	p-value
Time to Completion (s)	380	373	-1.9%	1.1×10^{-1}
Average Places Identified	18.1	21.6	19%	1.1×10^{-2}
Time per place identified (s)	25.9	18.4	-29%	5.5×10^{-2}

Table 5.8: Identification and time results for the simulation portion of the expeirment.

.

	2D Interface	3D Interface	Change	p-value
Average Velocity (m/s)	0.32	0.37	16%	2.4×10^{-2}
Average Collisions	8.6	4.8	-44%	7.4×10^{-3}
Time Touching a Wall (%)	6.9%	3.0%	-56%	9.1×10^{-4}

Table 5.9: Some of the objective results from the simulation portion of the experiment.

.

in the environment. This was measured for each participant by dividing the total time spent on the task by the number of places identified. With the 3D interface it took an average of 29% less time to identify each place in comparison to the 2D interface ($\bar{x}_{2D} = 25.9$s, $\bar{x}_{3D} = 18.4$s, $p = 5.5 \times 10^{-2}$). The results of the average time to completion and number of places identified are shown in Table 5.8.

Some of the reasons the operators were able to identify victims faster with the 3D interface are that the robot traveled 16% faster, had 44% fewer collisions, and was in contact with walls 56% less frequently than with the 2D interface. Table 5.9 presents a summary of these results for the experiment. Figure 5.21 presents the percentage of time that the robot is within a given distance of the nearest obstacle.

Another interesting observation is that the operator drove the robot with the camera off center a larger portion of the time with the 3D interface in comparison to the 2D interface. In particular, significantly more driving time (59%) was spent with the camera tilted at least 10 degrees up or down ($\bar{x}_{2D} = 28\%$, $\bar{x}_{3D} = 59\%$, $p = 1.8 \times 10^{-3}$) and marginally significantly more driving time (10%) was spent with the camera panned at least 15 degrees off-center ($\bar{x}_{2D} = 78\%$, $\bar{x}_{3D} = 86\%$, $p = 1.3 \times 10^{-1}$). Figures 5.22 and 5.23 show a cumulative distribution function of the

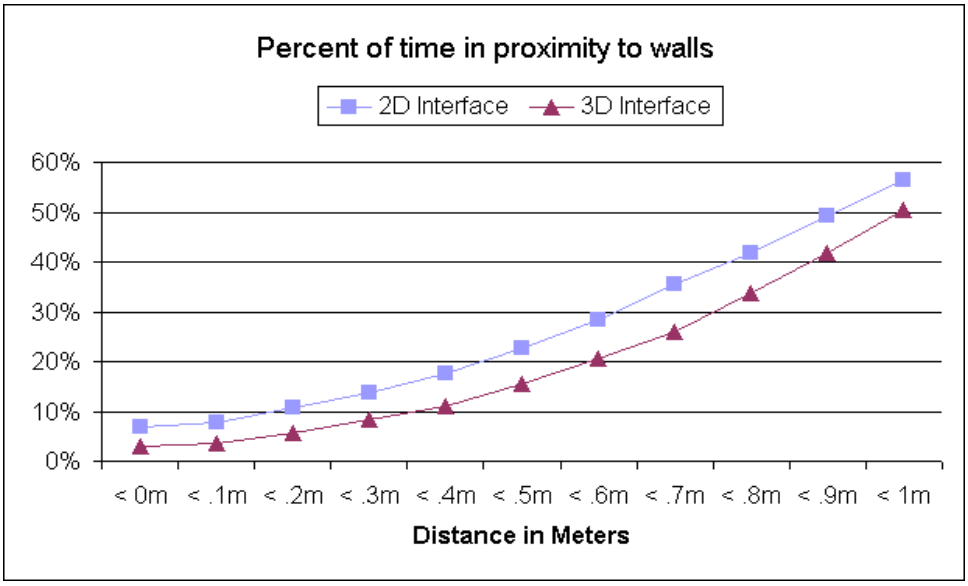


Figure 5.21: The percentage of time that the robot is within a given distance of the nearest obstacle.

time that the operator navigates the robot with the camera greater than a given angle off-center divided by the total time of the experiment. The difference in the camera tilt CDFs are statistically significant and the difference in the camera pan CDFs are marginally significant. Additionally, there was a marginally significant increase in the percentage of time the camera was zoomed in with the 3D interface in comparison to the 2D interface (29%, $\bar{x}_{2D} = 15\%$, $\bar{x}_{3D} = 19\%$, $p = 1.3 \times 10^{-1}$).

Real world

There is no statistical difference in the number of items found with the 2D and 3D interfaces in the real world portion of the experiment. However, there is a slight (8.2%) marginally significant decrease in the time to finish the obstacle course when using the 3D interface over the 2D interface ($\bar{x}_{2D} = 449$s, $\bar{x}_{3D} = 412$s, $p = 2.1 \times 10^{-1}$). There is also a significant decrease in the average time to identify each object, as measured, for each participant, by the time to complete the task divided by the total number of objects identified. The 3D interface required 10% less time per object than the 2D interface ($\bar{x}_{2D} = 44.5$s, $\bar{x}_{3D} = 40.0$s, $p = 2.8 \times 10^{-2}$). We

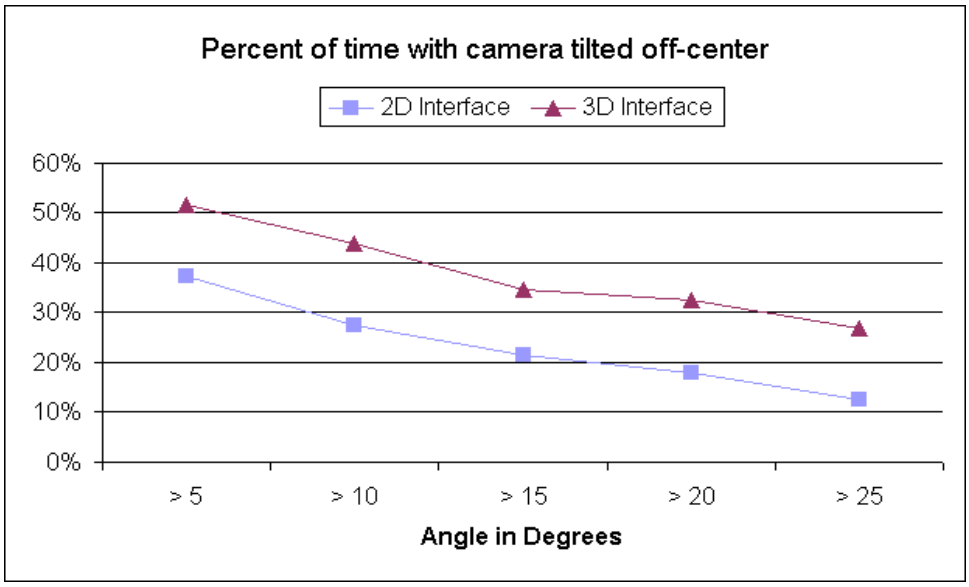


Figure 5.22: The percentage of time the robot was navigated while the camera was tilted up or down at least a given angle from the center position.

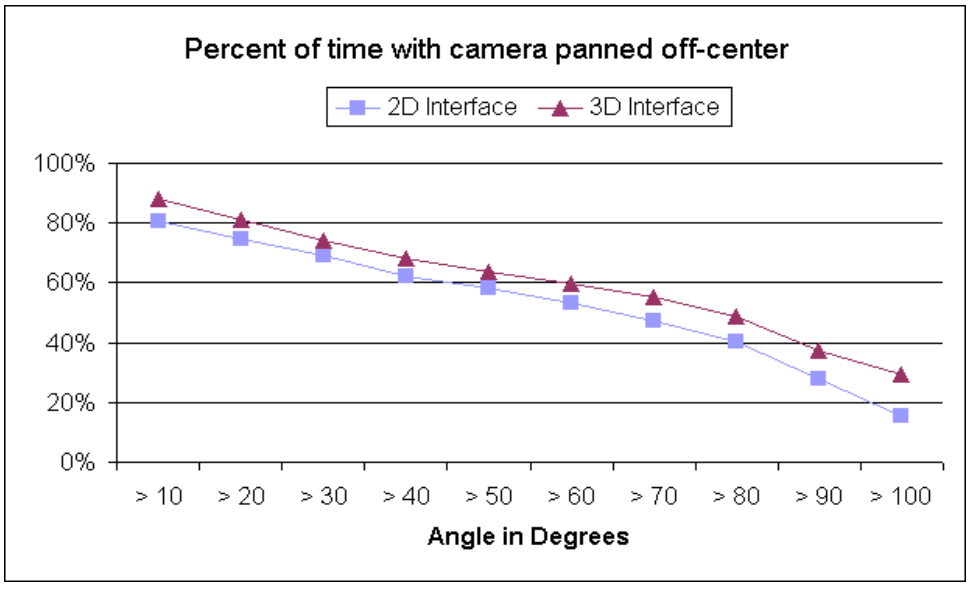


Figure 5.23: The percentage of time the robot was navigated while the camera was panned to a side a given angle from the center position.

	2D Interface	3D Interface	% Change	p-value
Time to completion (s)	449	412	-8.2%	2.1×10^{-1}
Places identified	10.27	10.29	0.2%	7.6×10^{-1}
Time per place identified (s)	44.5	40.0	-10%	2.8×10^{-2}

Table 5.10: Identification and time results for the realworld portion of the expeirment.

.

also observed that the robot, on average, had to protect itself from collisions 44% less with the 3D interface than the 2D interface ($\bar{x}_{2D} = 7.9$, $\bar{x}_{3D} = 4.4$, $p = 3.9 \times 10^{-2}$). The identification and time results are shown in Table 5.10.

One of the reasons the time to completion is not as different between the interfaces as observed in previous experiments is that the participants employed different strategies to search for the hidden objects. In particular, some participants were much more careful searching an area than other participants. This naturally led to more objects found and a correlating longer time for completion. When we compare the participants who used the 3D interface first and those who used the 3D interface second, we find that the group that used the 3D interface second finished the task 24% faster ($\bar{x}_{3D_1} = 472s$, $\bar{x}_{3D_2} = 359s$, $p = 4.1 \times 10^{-2}$, n = 9, unequal variance t-test) than the group that used the 3D interface first, but they also found 16% fewer obstacles on average ($\bar{x}_{3D_1} = 11.25$, $\bar{x}_{3D_2} = 9.44$, $p = 3.3 \times 10^{-2}$, n = 9, unequal variance t-test). There was not a significant difference in time per item found or robot initiative based on which interface was used first.

5.3.3 Discussion

This experiment shows that operators could find things faster with the 3D interface in comparison to the 2D interface despite different strategies for searching. The results also indicate that with the 3D interface operators spend more time with the camera further off-center than with the 2D interface. This result is interesting when combined with the improvment in navigational awareness with the 3D interface

because it suggests that the 3D interface supports the use of a pan-tilt camera better than a 2D interface for search and identify tasks.

5.4 Conclusion

In this chapter we presented three user studies that compare the usefulness of a 2D and 3D interface when performing search tasks with a pan-tilt camera. Our results indicate that in general, the 3D interface supports the use of a movable camera much better than a conventional 2D interface primarily because the orientation of the camera is presented relative to the robot in the integrated 3D display as opposed to a side-by-side iconic representation with the 2D interface.

Specifically, we found that with the 2D interface there was no difference in time to completion when the robot had a movable camera or a stationary camera. This is interesting because the environment was designed to exploit the use of a pan-tilt camera and even though the camera was used, the operator drove so slowly that any benefits of the camera were cancelled out.

In contrast, we found that with the 3D interface there is an improvement in the time to completion of a search task when the robot had a movable camera in comparison to a robot with a stationary camera. Operators did drive the robot slower with the pan-tilt camera than without, but it was not so slow as to overcome the advantages of the movable camera.

We also found that operators were much faster at finding and identifying things of interest with the 3D interface than with the 2D interface. Part of this was due to improved navigability with the 3D and part was due to the fact that the operator used the camera more while navigating the robot. It is interesting that even though the camera was used more with the 3D interface, navigational awareness remained high.

For design purposes, since information is diverse and not usually focused directly at camera level to a robot, it is important in search tasks to have the ability to use a pan-tilt-zoom camera while driving the robot and to make this control intuitive for the operator. Integrating the orientation of the camera in a 3D perspective along

with map, video and robot pose information is a good way to facilitate search because information regarding the orientation of the camera on the robot is readily available and visible to the operator.

Chapter 6

Principles and Extensions

In the previous user studies, we showed that operators were able to perform navigation and exploration tasks much better with the 3D interface than with the 2D interface. The purpose of this chapter is to discuss why the 3D interface supported better performance than the 2D interface.

The chapter begins by discussing the cognitive effort required for remote-robot locomotion given a conventional 2D interface. In the discussion we point out that one goal of the 3D interface is to reduce the cognitive workload required to perform robot locomotion. Next, we present three design principles used by the 3D interface to reduce the cognitive workload of the operator, namely, the use of a common reference frame, the correlation of action and response, and the use of an adjustable perspective. Finally, we discuss how the design principles can be applied to extend the 3D interface to other domains.

6.1 Cognitive Processing

Effectively controlling a robot from a remote location requires the efficient translation of sensor information into purposeful action. If information from the robot is presented poorly, the operator can be overwhelmed with the cognitive workload of just trying to understand what movements can be done by the robot.

In a landmark paper, Gibson and Crooks treat the problem of automobile driving as primarily a perceptual task. They write, "Locomotion is guided chiefly by vision, and this guidance is given in terms of a 'path' within the visual field of the individual such that obstacles are avoided and the destination ultimately reached" [45].

They then present the notion of a "field of safe travel" and discuss how this notion implies that the locomotion portion of driving is primarily a perceptual task with very little input required from higher cognitive levels. In terms of Rasmussen's *Knowledge Base*, *Rule Base*, and *Skill Base* hierarchy of behavior, the locomotion portion of driving is decidedly skill-based [48, 100, 110].

By contrast to Gibson's perception-based treatment of skill-based automobile driving, Woods et al. identify the difficulty of operating a robot remotely given conventional interfaces as seeing the world through a "soda straw." They write, "The limited angular view associated with many remote vision platforms creates a sense of trying to understand the environment through what remote observers often call a 'soda straw' " [138]. This limited perspective requires operators to use higher level cognitive processes to translate sensor readings into a sense of situation awareness. This suggests that the locomotion portion of teleoperation is not a skill-based but rather a rule- or knowledge-based behavior [48].

One of the goals of the 3D interface is to present information to a human in a way that reduces the amount of cognitive information processing required to understand and interpret sensor readings from a robot. In other words, the 3D interface was developed to try and turn the control of robot locomotion from a knowledge-based behavior to a skill-based behavior.[1] In order to reduce cognitive workload, populating the virtual environment with realistic information may seem a reasonable approach because humans see 'realistic' information every day. Nevertheless, Smallman and St. John discuss the notion of Naïve Realism wherein they claim that as displays become more realistic, the perceptual system of humans has a more difficult time interpreting the available information. Processing realistic information requires higher cognitive workload, and although realistic displays are generally preferred by operators, they can hurt performance [115].

The user-studies that we discussed previously show that operators are able to consistently perform navigation and exploration tasks better with the 3D interface.

[1]Note that training can also be used to move the control behavior from a knowledge-based process to a skill-based process. However, the focus of this work is on making robot teleoperation easier for novice operators with little training.

This improvement in performance suggests that robot locomotion can be more of a skill-based behavior when the 3D interface is used as compared to a conventional 2D interface. Additionally, it shows that the design of the 3D interface does not fall into Smallman and St. John's Naïve Realism trap because the virtual representation was not 'too realistic' [115]. We next discuss some of the principles addressed by the 3D interface that led to the improvement in performance.

6.2 Principles

In this section we present three design principles that helped the 3D interface yield better results than the 2D interface. The principles are a) a common reference frame, b) correlation of action and response, and c) an adjustable perspective. The principles describe how information from multiple sources can be presented to the operator in such a way as to reduce the cognitive processing required to interpret and understand the information. We embody the principles by discussing how the 2D and 3D interfaces address each of them.

6.2.1 Common reference frame

When using mobile robots, there are often multiple sources of information that theoretically could be integrated to reduce the cognitive processing requirements of the operator. In particular, a mobile robot typically has a camera, range information, and some way of tracking where it has been. To integrate this information into a single display, a common reference frame is required. The common reference frame provides a place to present the different sets of information such that they are displayed in context of each other. In terms of Endsley's three levels of situation awareness [35] (Section 2.2), the common reference frame aids the perception and comprehension elements of situation awareness. In the user-studies we used both a robot-centric and a map-centric frame of reference to present information to the operator (Chapters 4 and 5).

Robot-based reference frame

The robot itself can be a reference frame because a robot's sensors are physically attached to the robot. This is useful in situations where the robot has no map-building or localization algorithms. The reference frame can be portrayed by displaying an icon of the robot with the different sets of information rendered as they relate to the robot. For example, a laser range-finder typically covers 180 degrees in front of the robot; the information of where the laser detected obstacles could be presented as barrels placed at the correct distance and orientation from the robot. Another example is the use a pan-tilt camera. If the camera is facing towards the front of the robot, then the video information should be rendered in front of the robot. If the camera is off-center and facing towards a side of the robot, the video should be displayed at the same side of the virtual robot. The key is that information from the robot is displayed in a robot-centric reference frame.

Map-based reference frame

In the robot-centric reference frame, it most likely will not be beneficial to represent two or more robots unless they are somehow collocated in a larger reference frame. If the robots have map-building and/or localization capabilities, such a reference frame could be map-based. With a map as the reference frame, the operator has the ability to correlate different sets of information that may not be tied to a robot's current set of information. As an example, consider the process of constructing a map of the environment. As laser scans are made over time, the information is typically combined with probabilistic map-building algorithms into an occupancy grid-based map [64, 124]. Updates to the map depend not only on the current pose of the robot, but on past poses as well. When the range scans of a room are integrated with the map, the robot can leave the room and the obstacles detected are still recorded because they are stored in relation to the map and not the robot.

Another example of where a map can be useful as a common reference frame is with icons or snapshots of the environment. When an operator or robot identifies a place and records information about it, the reference frame of the map provides a

way to store the information as it relates to the map of the environment. Moreover, using a map as the reference frame also supports the use of multiple robots—as long as they are localized in the same coordinate system. This means that places or things identified by one robot can have contextual meaning for another robot or operator that has not previously visited or seen the location.

Reference frame hierarchy

One advantage of reference frames is that they can be hierarchical. At one level, the information related to a single robot can be displayed from a robot-centric reference frame. At another level, the robot-based information from multiple robots can be presented in a map-based reference frame which shows how the robots are spatially related to each other. In the map-based reference frame, each robot still presents its own robot-centric information, but now the group of individual robot-centric reference frames is collocated into a larger reference frame.

We can also imagine another frame of reference wherein multiple maps are collocated with different robots in each of the maps (i.e. a city with maps of different buildings). The common reference frame is simply a way to combine multiple sources of information into a single representation.

2D and 3D reference frames

Both the 2D and 3D interfaces support a common reference frame between the robot pose and obstacles with the use of a map. However, that is the extent of the common reference frame with the 2D interface since video, camera orientation, and operator perspective are not presented in the same reference frame as the map or the robot. In fact, Figure 6.1 illustrates that with the 2D interface there are actually four different frames of reference from which information is presented to the operator. In contrast, the 3D interface presents the video, camera orientation, and user perspective in the same reference frame as the map and the robot pose as illustrated in Figure 6.2.

The multiple reference frames in the 2D interface require more cognitive processing than the single reference frame in the 3D interface because the operator

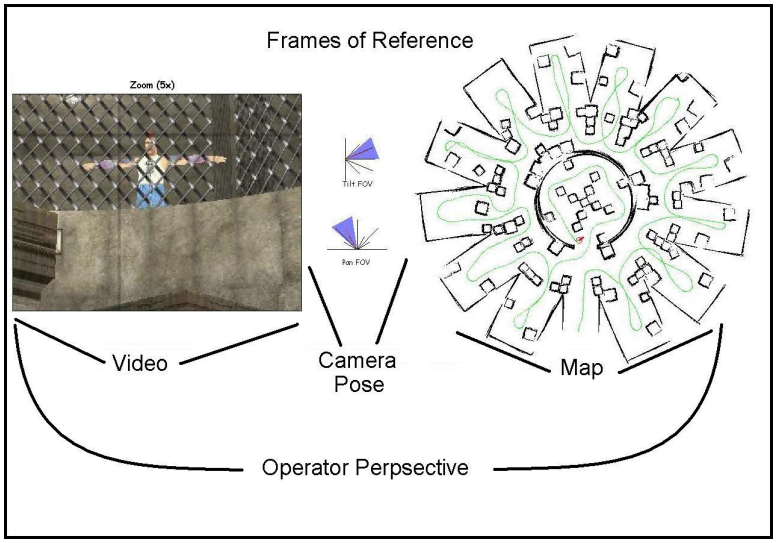


Figure 6.1: The four reference frames of the information displayed in a 2D interface: video, camera pose, map, and operator perspective.

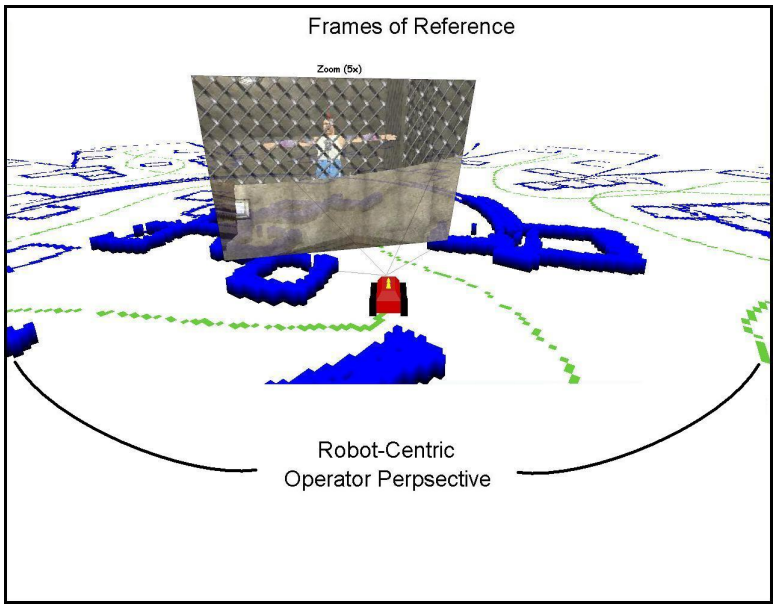


Figure 6.2: The reference frames of the information displayed in a 3D interface: robot-centric and operator perspective (which are both the same).

must mentally translate different reference frames into the same reference frame to understand the information regarding the robot's situation. With the 3D interface, the cognitive work of combining the reference frames is supported by the interface, which reduces the cognitive requirements on the operator.

6.2.2 Correlation of action and response

Another principle to reduce cognitive workload is to maintain a correlation with commands issued by the operator and the expected result of those commands as observed by the movement of the robot and changes in the display. In terms of Endsley's three levels of situation awareness [35] (Section 2.2), the correlation of action and response affects the operator's ability to project or predict how the robot will respond to commands.

An operator's expected response of the robot depends on his or her mental model of how commands translate into robot movement and how robot movement changes the information on the interface. When an operator moves the joystick forward, the general expectation, with both the 2D and the 3D interface, is that the robot will move forward. However, the expectation of how the interface will change to update the robot's new position is different for both interfaces. In particular, operator expectation with respect to the change in the video and the change in the map can lead to confusion with the 2D interface.

Change in video

One expectation of operators is how the video will change as the robot is driven forward. In the 2D interface, the naïve expectation is that the robot will travel "into" the video when moving forward. With the 3D interface, the expectation is that the robot will travel at an angle that correlates to the offset of the camera. Both of these expectations are correct if the camera is in front of the robot. However, when the camera is off-center, an operator with the 2D interface expects the robot to move "into" the video when in reality it moves at an angle to the video which can be confusing [141]. Only when the camera is directly in front of the robot does

the operator's expectation match the observed change in the interface. With the 3D interface, the operator expects the robot with the off-center camera to move at an angle to the video, which is the visual response that happens.

Change in map

Another expectation of the operator is how the robot icon on the map will change as the robot is driven forward. With the 2D interface, the naïve expectation is that the robot will travel up (north) on the map when the joystick is pressed forward. Similarly, with the 3D interface, the expectation is that the robot will travel "into" the display when the joystick is pressed forward. Both of these expectations are correct if the robot is heading "up" with respect to the map. However, when the robot is heading in a direction other than north, an operator with the 2D interface still expects the robot icon to move "up" with respect to the map when in reality the robot icon moves in the direction the robot is heading. This can be particularly confusing when turn commands are issued, since the change in the location of the robot icon on the map depends on the orientation of the robot with respect to the map [108, 136]. With the 2D interface, only when the robot is heading "up" (north) with respect to the map does the operator's expectation match the observed change in the interface.

With the 3D interface, the operator expects the robot to move into the display, which is the visual response that happens because the operator's view of the robot is tethered to the robot as opposed to the map.

Change in camera tilt

One area of operator expectation that is difficult to match is the operator's mental model of what should happen when a camera is tilted up or down. To control the camera tilt in the experiments in Chapter 6, the POV hat on top of the joystick is used, the problem is that some operators prefer to tilt the camera up by pressing 'up' on the POV and others prefer to tilt the camera up by pressing 'down' on the POV.[2]

[2]A conflict in preferences was not observed when a camera is panned left and right.

This observation illustrates the fact that sometimes the mental model of the operator is based on preferences and not the manner in which information is presented.

Cognitive Workload

The advantage of the 3D interface is that the operator has a robot-centric perspective of the robot because the virtual viewpoint is tethered to the robot. This means that the operator issues commands as they relate to the robot, and the expected results match the actual results. Since the operator's perspective of the environment is robot-centric there is minimal cognitive workload to anticipate how the robot will respond to commands.

The problem with the 2D interface is that the operator has a map-centric perspective of the robot that must be translated to a robot-centric perspective in order to issue correct commands to the robot. The need for explicit translation of perspectives results in a higher cognitive workload to anticipate how the robot will respond to commands.

Additionally, the 2D interface can be frustrating because, to novice operators, it seems that the same actions in the same situations lead to different results. The reason for this is that, for the 2D interface used in this dissertation, the most prominent areas of the interface are the video and the map which always appear about the same. The orientation of the robot and the camera, on the other hand, are less prominently displayed even though they significantly affect how the video will change as the robot is moved and how the robot icon will move on the 2D map. If the orientation of the robot or the camera is neglected or misinterpreted, it can lead to errors in robot navigation. Navigational errors increase cognitive workload because the operator must determine why the actual response did not match his or her expected response. For this reason, a novice operator can be frustrated that the robot does different things when it appears that the same information is present and the same action is performed.

6.2.3 Adjustable perspective

Although sets of information may be displayed in a common reference frame, the information may not always be visible or useful because of the perspective through which the operator views the information. Therefore, the final principle that we discuss for reducing cognitive workload is to use an adjustable perspective. An adjustable perspective can aid all three levels of Endsley's situation awareness [35] (Section 2.2) because it can be used to a) visualize required information, b) support the operator in different tasks, and c) maintain awareness when switching perspectives.

Visualization

One advantage of an adjustable perspective is that it can be changed depending on the information the operator needs to "see". For example, if there is too much information on a map, the perspective can be zoomed in closer to eliminate some of the extra information and focus on the area and information of interest. Similarly, if there is some information that is just beyond the visible area of a part of the display the perspective can be zoomed out to allow the visibility of more information.

Visualizing just the right amount of an environment can have a lower cognitive workload than either observing too much or too little of the environment. When there is too little information in the display, the operator is left with the responsibility to remember previously recorded information. When there is too much information in the display the operator is left with the responsibility to find and interpret the necessary information. Determining the best visualization, however, comes at a cost to the operator since he or she must think about choosing the right perspective.

The ability to zoom in and out a perspective is a common feature of most 2D and 3D maps, but, in 2D interfaces the map is usually the only part of the interface with an adjustable perspective.

Different tasks

Another advantage of an adjustable perspective is that the perspective through which an operator views a robot in its environment can influence the performance on a particular task. For example, teleoperation is usually performed better with a more egocentric perspective while spatial reasoning and planning is performed better with a more exocentric perspective [106, 136]. When the perspective of the interface is not adjusted to match the requirements of a task, the cognitive workload on the operator is increased because the operator must mentally adjust their perception of the information to match the requirements of the task. Even though tasks are better performed with the proper perspective, conventional 2D interfaces do not typically support an adjustable perspective, and the user is usually left with one interface for all the robot tasks.

Maintain awareness

Often robots are versatile and can be used to accomplish multiple tasks, so it is reasonable to anticipate that an operator would need to change tasks while a robot is in operation. To facilitate this change, an adjustable perspective can be used to create a smooth transition between one perspective and another. A smooth transition between perspectives has the advantage of allowing the operator to maintain situational context as the perspective changes which reduces the cognitive workload by reducing the need to acquire the new situational information from scratch [25, 96]. Some instances where a smooth transition might be useful include switching between egocentric and exocentric perspectives, information sources (GPS-based, map-based, robot-based), map representations (occupancy-grid, topological), or video sources (cameras in different locations, different types of camera).

In the user-studies of this dissertation, a different perspective was used for many of the 3D interfaces because there were different requirements for the tasks and the information sometimes needed to be viewed differently. In comparison, the 2D interface always had the same perspective because conventional 2D interfaces generally do not provide an adjustable perspective.

6.3 Extensions

In this section we explore extensions that apply the principles discussed in the previous section to the 3D interface. The purpose of this section is to show how the principles from the previous section can be applied to other domains with reasonable expectations for success. To begin with, we discuss the use of GPS (Global Positioning System) for the common reference frame. We then present a representation for indicating camera zoom. We conclude with an example illustrating the use of a movable arm on a robot.

6.3.1 GPS reference frame

Recently, an experiment was performed by the Idaho National Laboratory (INL) using a robot developed by Carnegie Mellon University (CMU) and the 3D interface we developed at BYU. The scenario was designed such that an unmanned air vehicle (UAV) flew a flight pattern over a runway and took pictures of the runway while an unmanned ground vehicle (UGV) was concurrently tasked to find and identify land mines on the runway and designate the path it followed to discover the mines. The requirements for the experiment were that aerial photography, land mines, robot path, and current robot position all be integrated in a GPS-referenced display that could be viewed by operators and/or observers.

Populating the 3D interface

This experiment is possible because the common reference frame used for the 3D interface is compatible with GPS. When the counter-mine robot is turned on, it determines its current GPS position along with the offset of its current heading with GPS-based north. The robot's GPS information is relayed to the 3D interface, which renders a grid on the ground of the 3D interface that is aligned with the GPS axes (North-South, East-West) with grid lines drawn every 10 meters. A 3D model of the counter-mine robot is then rendered at its corresponding GPS location in the 3D interface. A picture of one of the counter-mine robots used for the experiment is shown in Figure 6.3.

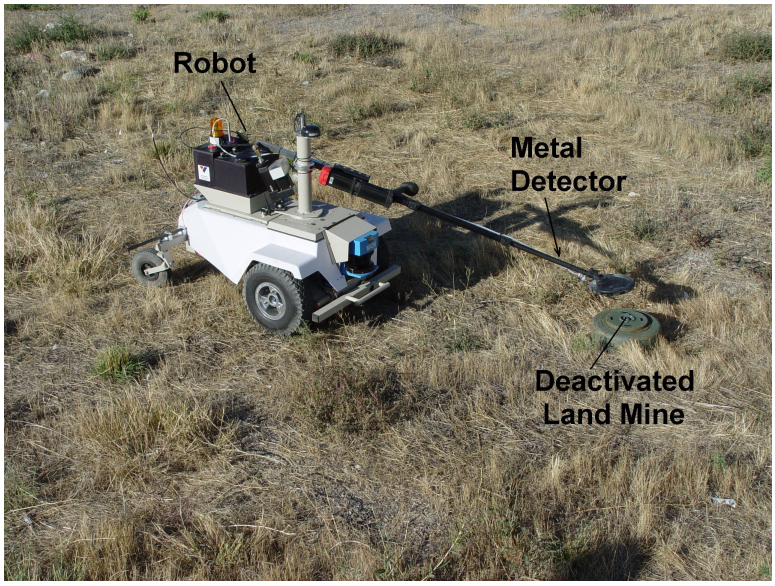


Figure 6.3: The counter-mine robot used for the GPS-based land mine detection experiment [13].

When the robot identifies a land mine, the GPS coordinates of the land mine are determined and the information is used to place a land mine icon in the 3D interface. As the robot positions are updated in the interface, markers indicating the width of the robot are rendered to illustrate the path traversed by the robot. Furthermore, an aerial photograph of the robot's environment is taken from the UAV and transmitted to the 3D interface. The photograph is manually correlated with the obstacles detected by the UGV and rendered under the GPS grid lines to give photographic information of the robot's environment. A screenshot from the experiment is shown in Figure 6.4.

This screenshot is taken from a virtual perspective far above the robot. The reason for this is that the robot was traveling and detecting land mines autonomously and the interface was primarily used to observe the progress and the environment of the robot—not to teleoperate the robot. This exocentric perspective helps observers visualize the robot and its surroundings in the context of the aerial photograph[3]. In

[3]The aerial photograph is not as useful when the viewpoint of the 3D interface is close to the robot because the resolution of the photograph is too low.

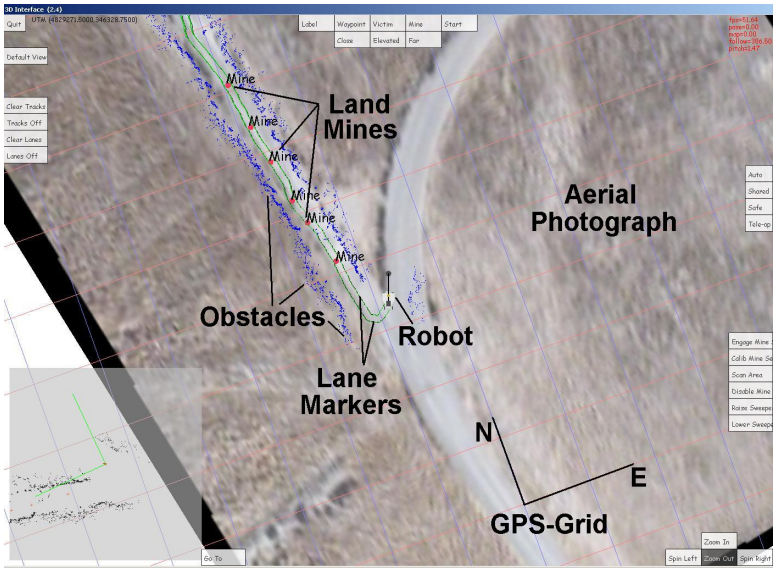


Figure 6.4: Snapshot taken from the INL counter-mine experiment.

the screenshot, icons of the robot and detected mines are shown as they relate to the path traversed by the robot and detected obstacles.

One of the challenges we observed when viewing the 3D interface from an exocentric perspective is that the icons in the display appear so small that they are barely noticeable. Although it is advantageous and often necessary to present the icon of the robot in proper scale when teleoperating the robot, the requirement of proper scale can be relaxed in exocentric perspectives because it is more important to present an imprecise observable representation of the robot's situation than a precise non-observable representation. Therefore, in this experiment, the robot is rendered to appear the same size on the interface, regardless of the perspective from which the virtual environment is displayed.

Coordinating the real and virtual worlds

One of the advantageous of using GPS as the reference frame for rendering information from the robot is that GPS is inherently real-world based which means that information from the real and virtual worlds should, theoretically, be interchangeable. For example, the real counter-mine robot uses spray paint to mark

the ground where land mines are detected and where the robot has traveled so there is a correlation between the information seen in the virtual display and the information in the real world. In this way, an observer can see from the virtual environment where the mines are and what path the robot has taken. Then, upon entering the real environment, the paint left by the robot helps the observer correlate the virtual information with the real world. Additionally, if something of interest is observed from the aerial photograph in the virtual environment, the GPS location can be determined from the virtual environment and used to go to the place of interest in the real world even though the robot itself has never been there.

Validation

In order to validate the usefulness of the GPS reference frame, an experiment could be designed where an observer tracks a robot's progress in the 3D interface, then uses information from the virtual interface to perform a task in the real world. From a search and rescue scenario, the robot could be used to help an operator find and identify victims. The information from the search portion of the task could be relayed to another team member that is collocated in the environment with the robot to complete the rescue portion of the task. The question to answer is how well the information in the interface can be conveyed to someone who has not seen the interface.

Performance metrics for the search and rescue task could include time to completion and time to fulfill individual instructions. Measurements could also be made regarding the accuracy with which instructions are given, how frequently instructions are repeated, and how much communication is required to complete the task. Since the experiment might require team management, significant training would be required of novice operators and, therefore, the experiment may be better suited to those who have had team-training such as military or search and rescue professionals.

6.3.2 Visualizing camera zoom

Pan-tilt cameras are useful for searching an environment because they can be used to gain a better understanding of the visual scene without moving the robot. Often, pan-tilt cameras also have a zoom feature which allows the operator to focus the camera on a small area of the environment. Although the zoom feature can be useful for observing, in detail, parts of the environment, it is difficult to convey to the operator the level of zoom of the camera.

The difficulty of making the level of zoom perceivable to the operator is a result of the tension between the goal to show an increase in image detail and the goal to show a decrease in the field of view as the camera is zoomed in. In previous work the tension is described as follows: "shrinking the dimensions of the image can convey the feeling of a decrease in the field of view, but the greater detail of the image is masked by the fact that the presentation is smaller; a similarly displeasing visualization results when the image is 'pulled' closer to the robot to emphasize the amount of detail" [48].

A previous approach

In previous work, a concept for visualizing zoom is described based on an approach sometimes used in the gaming community which presents an area of magnification within the virtual environment. The theory is that the ratio between the image height and the height of the magnified obstacles gives the perception of a changing field of view and the ratio between the normal and magnified obstacle heights gives the perception of increased detail [48] (see Figure 6.5).

When this approach was tested in a 3D environment, we found that it worked fairly well to convey the ideas of increased detail in the image and decreased field of view. However, the boundary between the magnification area of the display and the regular portion of the display had quite a visual disconnect because of the difference in the relative size of obstacles. Furthermore, the increased size of the obstacles in the magnified portion of the display did not seem indicative of the environment. To improve the magnification window, we tried a larger, opaque boundary

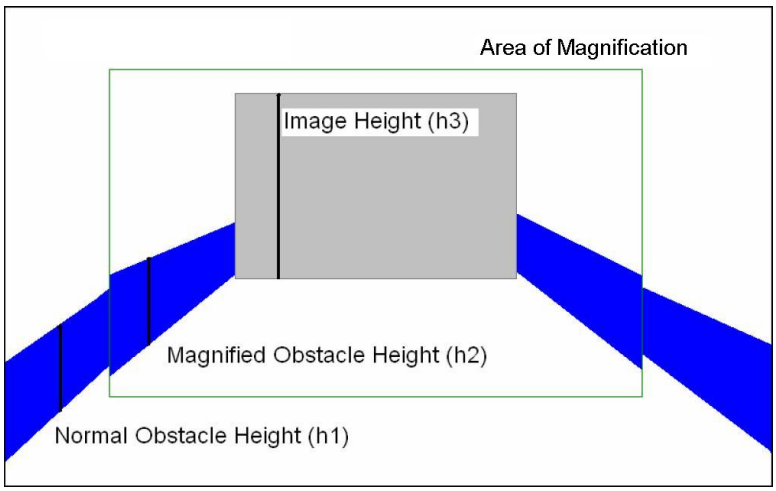


Figure 6.5: A display concept for zoom that helps resolve the tension between increased detail and decreased field of view by allowing the operator to perceive the differences between a magnified camera image and the perspective of the real world. Adopted from [48].

to offset the magnification area with a "tunnel", but the disconnect between the magnified and normal obstacles did not change and it seemed to make the visual scene too crowded.

The current approach

To find another solution that can support operator awareness of zoom, we first discuss some observations. First, additional icons or windows in the display are undesirable because they tend to make the display more cluttered and extra information requires cognitive interpretation by the operator. Second, since the primary purpose of zooming a camera is to gain more visual information about a place in the environment, this should be supported by increasing the percentage of the display that is used to display the video. Third, illustrating the field of view of the camera may be helpful, but should not be naïvely enforced—especially if doing so conflicts with the first two observations.

The solution we settled on makes use of an adjustable perspective to give the sensation of changing the zoom of the camera. In particular, we move the virtual perspective closer to the robot as the camera is zoomed in. This naturally increases

the size of the video relative to the size of the display and it loosely conveys the sense of decreasing the field of view because less of the virtual environment is visible. To further improve the perception of the decreased field of view, we also shrink the relative size of the video as the camera is zoomed in. To minimize the tension between the decreased field of view and the increased detail of the image, the decrease in the relative size of the video with respect to the virtual scene is a slower process than the increase in the perceived size of the video with respect to the size of the display. The end result is that as the virtual perspective is moved towards the robot, the virtual environment becomes larger, but the camera image gets larger more slowly. Figure 6.6 illustrates what the 3D interface looks like at different levels of camera zoom.

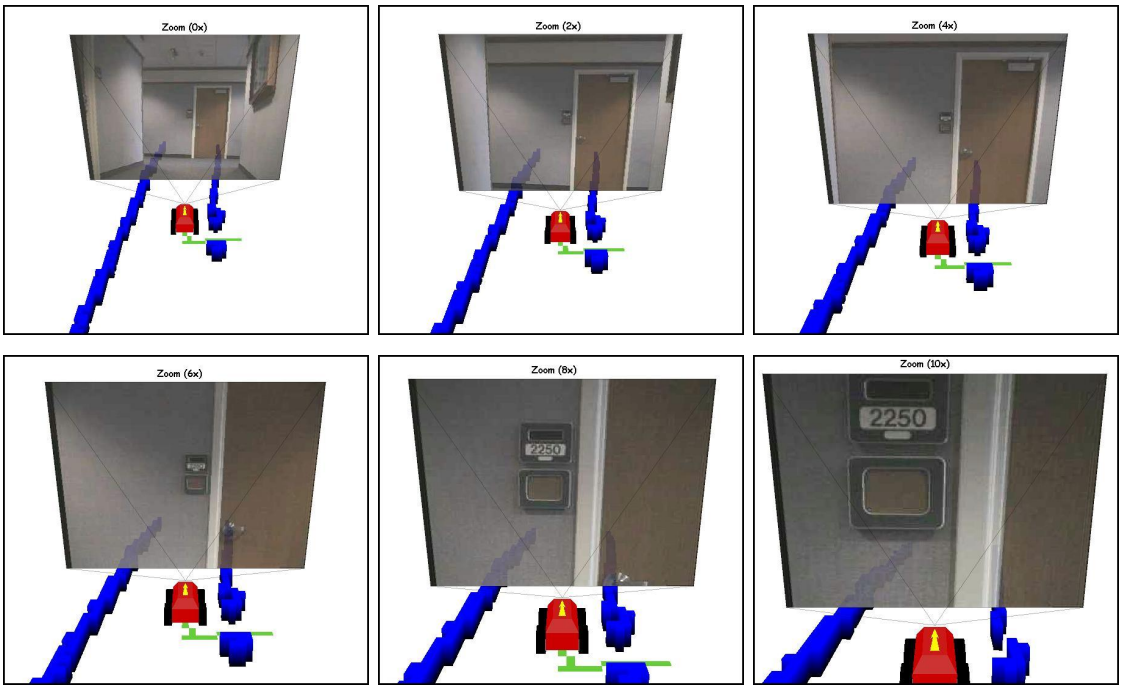


Figure 6.6: 3D representation of the level of zoom with a PTZ camera. The appearance of zoom is effected by adjusting the operator's perspective of the environment. On the top row from left to right the zoom levels are 1x, 2x, and 4x. On the bottom row from left to right the zoom levels are 6x, 8x, and 10x.

Validation

To test different interfaces representing camera zoom, search tasks could be performed where zoom is required to succeed. The task could be to find and identify items of interest, wherein the camera must be zoomed in to correctly identify the item. This would necessarily require the use of the pan-tilt controls and perhaps some robot navigation, but a significant aspect of the task would be controlling the zoom of the camera. In order to focus the task on the use of the camera, the need to navigate the robot could be eliminated with the implementation of a shared control algorithm where the operator only controls the speed at which the robot moved forward; the robot is required to avoid obstacles. Before performing experiments for this task, operators would need to have sufficient training with the camera controls since they are not as intuitive for novice operators as navigation controls.

6.3.3 Robot arm manipulation

Some robots have movable arms to help an operator with dexterous tasks such as manipulating door handles (see Figure 6.7) or utilizing human tools [4]. Other, robots with "arms" have been designed for dangerous situations such as with SWAT teams, military, or space missions [4, 20, 60, 86].

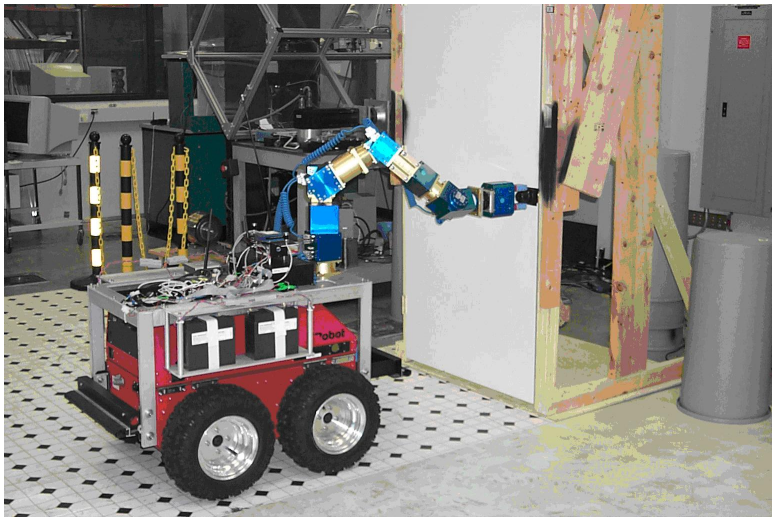


Figure 6.7: An ATRV-jr robot with an arm attachment used to open a door [13].

Robots in these situations may have a variety of sensors to provide information regarding the arm, the robot, and the environment and how they are related to each other. In this section we discuss how the principles discussed earlier could be used to extend the 3D interface to facilitate the use of a movable arm. Since we do not actually have a robot with a movable arm, this discussion is approached from a theoretical perspective and illustrations of what the display could look like are shown. For this discussion, we consider a robot with a movable arm and two video cameras, one on the base of the robot and one on the arm of the robot.

Frame of reference

To visualize the information from a robot with an arm and two cameras, we must first identify the reference frame from which the information will be displayed. We chose a robot-centric reference frame because the arm and the two video cameras are physically connected to the robot. Additionally, a map-centric reference frame is not so useful with a robot arm because although the 3D interface can provide navigational information, it has not been extended to provide a complete 3D representation of the environment (we leave this as a possible direction of future work). The map that is provided from the laser range finder is obtained from obstacles that are found in the same plane as the laser range finder. This means that when the arm is moved to positions that differ from the plane scanned by the laser range finder, the map information becomes less useful to aid the movement of the robot arm. The relative uselessness of the map implies that the information required for controlling the robot arm will most likely come from the video cameras and the pose of the arm with respect to the environment.

Since there are two cameras on the robot, the question naturally arises of which camera will provide the best perspective for the task. The choice of which camera to use is further complicated by the fact that both cameras may be useful during different parts of the task or there may be a situation when both cameras are needed at the same time. To help the operator make this choice, it is advantageous to present both video images at the same time and as they relate to the arm and the

robot so the operator can directly observe which will be most helpful. The problem with this is that rendering two video streams in a single, robot-centric reference frame can lead to operator confusion because there will frequently be overlap between the information from the two cameras and they will both attract the operator's attention [68]. Furthermore, rendering a three-dimensional model of an arm in the same reference frame as the cameras may only make the representation more confusing.

To simplify the interface, we do not want to eliminate any of the available information because all the information may be required to support the operation of a robotic arm. Even toggling some of the information on and off may not be beneficial because there is cognitive workload required when an operator's situation awareness is disrupted and must be recreated [25, 96].

Transparency

To handle the issue of presenting all the available information without overwhelming the operator, we utilize the ability to make objects semi-transparent in the 3D interface. Transparency is helpful because it serves to reduce the prominence of an object in the virtual environment without removing the object entirely. The increase in transparency of one object also serves to increase the relative prominence of other objects in the scene.

As an operator switches between video streams, the old video information can be made more transparent and the new video information can be made more opaque. The advantage of this is that both video streams are always presented in context of each other. The theory is that switching focus between video streams when both are somewhat visible does not disrupt the situation awareness of the operator, but smoothly shifts the situation awareness to the new source of information.

Transparency can also be used when rendering the pose of the robot arm in the virtual interface so that the model of the arm does not obscure the video. In fact, the pose of the entire arm may not be of much interest in comparison to the pose of the hand at the end of the arm; therefore, we can make the arm mostly transparent and leave the hand more opaque. The advantage of using transparency is that the

prominence of the different sets of information can be adjusted to support the needs of the operator.

The 3D representation

One challenge with representing the arm of the robot in three-dimensions is that it is difficult to perceive the actual location of the arm with respect to the body of the robot because of the perspective of the 3D interface. To address this issue and illustrate the horizontal distance of the arm relative to the robot, a "shadow" of the arm is projected onto the floor of the virtual environment. Another shadow is also rendered at the height of the robot (since the floor of the virtual environment is often obscured by the robot). To illustrate the vertical height of the robot arm, we considered a vertical bar from the ground to the height of the robot hand (similar to height-above-ground displays used for aviation [136]). However, it was somewhat distracting to have the vertical line always following the hand of the robot.

An illustration of how a robot arm might be rendered in the 3D interface is shown in Figure 6.8. In the figure, the arm of the robot is about 60 degrees to the left of the front of the robot. The arm is drawn mostly transparent so the video from the base of the robot and behind the arm can easily be seen. The hand on the arm, however, is drawn mostly opaque so the pose and the grippers can be observed. In the top image, the video from the base camera is more opaque to indicate that the camera on the base of the robot is in use and in the bottom image the video from the arm is more opaque to indicate that the camera on the arm is in use.

Other Considerations

Although the 3D interface addresses many of the issues we considered important, it is difficult to know just how well the interface will help when using a robotic arm. The main problem is that we do not have an arm to test the interface on which limits our understanding of how the physical arm of the robot will appear in the video streams as it is moved. Because of this we do not know the best way to represent the virtual arm to support the observation of the real arm within the

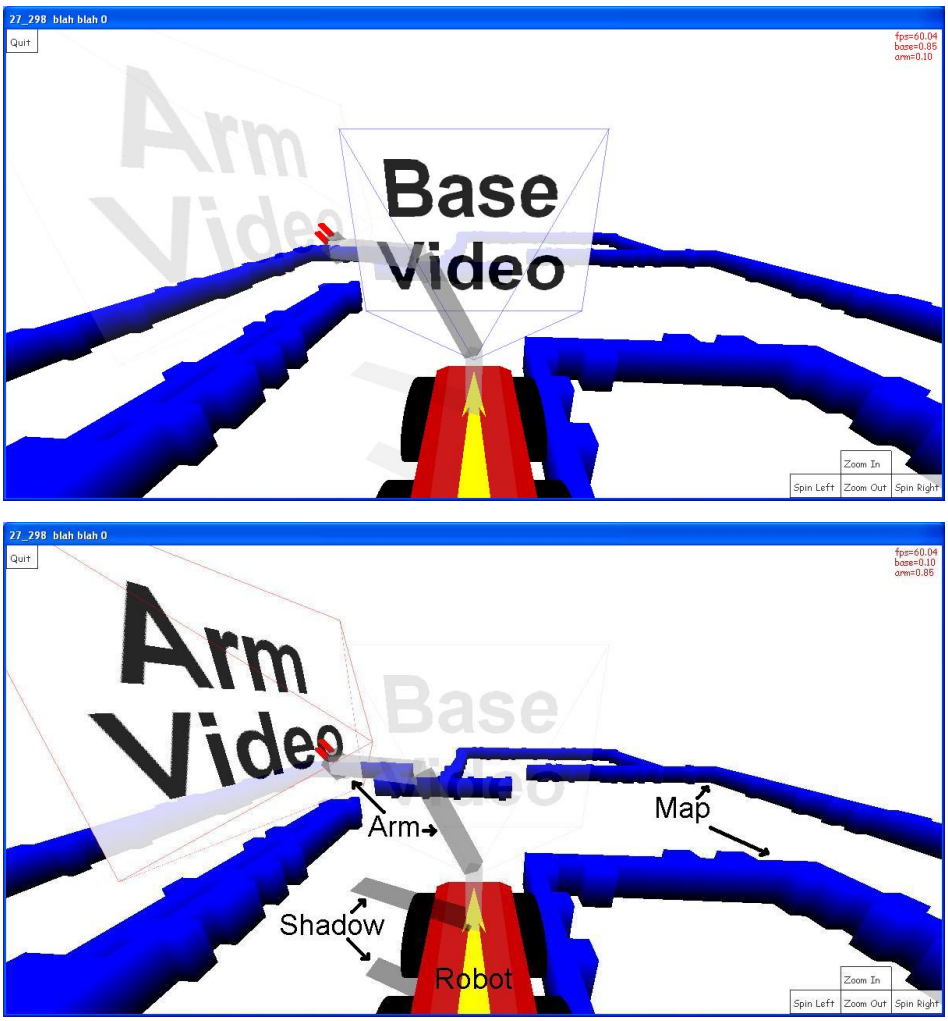


Figure 6.8: An illustration of how the 3D interface and transparency could be used to support the use of a mobile robot with an arm and two cameras.

video. If the hand of the arm is easily visible in the video, it may be the case that a representation of the pose of the arm is not as useful as simply representing the different sets of camera information as they relate to the pose of the arm and the robot.

One thing to consider that may improve the usefulness of any interface used to control a robot arm is the correlation of an operator's commands to control the arm with his or her expected response of how the arm will move and how the display will change. Once a designer has a good understanding of how an operator expects the arm to move and the display to change, the expectations can be used to guide the development of the controls and interface.

Another consideration that may be beneficial is to focus on changing the perspective from which the operator views the environment. For example, it may be beneficial to tether the perspective of the virtual environment to the movement of the arm so actions taken by the operator always affect the interface the same way.

Validation

To determine whether or not the 3D virtual representation of a robot with a movable arm is more useful than another interface, an experiment could be designed where an operator is required to find and move objects of interest. Since the arm is on a mobile robot, the experiment would be more interesting if the task not only involved the use of the arm, but also the movement of the robot. To minimize confounding factors, only the manner in which the information is presented should change between the interfaces. This means that there should be no difference between the way commands are issued by the operator (e.g. by joystick or keyboard) and the way the commands are handled by the interface and the robot.

Participants should perform the task with each interface so that fewer participants are required and comparisons between the interfaces can be made for individual participants. Training should depend on the types of users (novice vs. expert) and the goal of the experiment (usability vs. learnability), but should be sufficient that the arm can be reasonably controlled with each interface before actual testing

begins. Performance metrics should measure how long it takes to complete the entire task, how long it takes to grasp individual objects, and how many commands an operator issues to the robot. Additionally, awareness of the arm's pose with respect to the environment could be measured by the number of accidental collisions with the robot arm and the change in distance between the robot hand and the object of interest.

6.4 Summary

In this chapter, the goal of improving robot teleoperation by reducing the cognitive workload of an operator was discussed. In the context of reducing cognitive workload, interfaces that are too realistic were addressed because they can unexpectedly lead to an increased cognitive workload. Three principles were presented for interface design that can be used to reduce the cognitive workload on the operator. The three principles are a) a common reference frame, b) correlation of action and response, and c) an adjustable perspective. These principles were used to extend the 3D interface to other domains including a GPS-based reference frame, visualization of camera zoom, and the use of a movable arm on a robot.

Chapter 7

Summary and Future Work

This dissertation presents a 3D augmented-virtuality interface for teleoperating a remote mobile robot in navigation and exploration tasks. The 3D interface is validated through a series of user-studies comparing performance and navigational awareness against a prototype 2D interface. In this chapter we summarize the dissertation and present future work.

7.1 Summary

Requirements and technology for creating a useful interface for teleoperating a remote robot were first set forth. The requirements are that the interface must support a) storing information, b) integrating similar information into a single display, and c) adjusting the perspective through which the operator views the robots environment. A 3D augmented virtuality interface is described which fulfills the requirements for a useful display.

The 3D interface was validated through user studies and was observed to help an operator perform significantly better than a conventional 2D interface in the tasks performed. In particular, operators were able to finish navigation tasks about 20% faster because they a) drove the robot faster, b) had fewer collisions, and c) maintained a further distance from walls. Additionally, operators were able to complete exploration tasks better because the 3D interface supports the use of a pan-tilt camera better than the 2D interface. Specifically, operators used the pan, tilt, and zoom of the camera more with the 3D interface than with the 2D interface.

Additionally, operators drove the robot with the camera off-center more with the 3D interface than the 2D interface.

In addition to finishing tasks faster, we objectively showed that operators tended to have better navigational awareness with the 3D interface in comparison to the 2D interface as measured by the number of collisions, the average minimum distance to obstacles, and the percentage of time in proximity to walls. The 3D interface also supports the operator better in conditions of network delay and is preferred more than ten to one by participants in comparison to the 2D interface.

Finally, three design principles for presenting multiple sets of information to the operator were discussed that ultimately led to the success of the 3D interface. The principles are a) a common reference frame, b) the correlation of action and response, and c) an adjustable perspective. The principles were then used to discuss extensions of the 3D interface to other domains, including a) a GPS referenced experiment, b) visualizing camera zoom, and c) the use of a robotic arm.

7.2 Future Work

In the current implementation of the 3D interface, the map is obtained from a laser range finder that scans a plane of the environment a few inches off the ground. This approach works particularly well for planar worlds, which generally limits the work to indoor environments. In order to apply the research to an outdoor environment, we will look at approaches for measuring and representing terrain (e.g. an outdoor trail). One of the main challenges with presenting a visualization of terrain is that it necessarily will increase the cognitive workload on the operator because there will be more information displayed in the interface since terrain information is available at every place in the environment. A solution will be determined by answering the question of how much information is required to give the operator sufficient awareness with a minimal effect on the operator's cognitive workload.

Another area that we are interested in pursuing is the use of multiple, heterogeneous robots. In Chapter 6 we described a situation where a UAV provided some information for visualizing the environment around a robot in a counter mine

task. We are interested in extending the interface such that multiple UAVs and UGVs can be used to populate the virtual environment with photographs and other sensed information that can help an operator understand the environment around the robot. On a larger scale, consider a mission with many different participants and resources: some human, some robotic, some air-based, and some ground-based, all of which have information that needs to be combined to give a director or commander sufficient understanding to make good decisions. The question to address is how to present the information to an operator or team of operators.

Related to the use of heterogeneous robots is the ability to make the interface adjustable or adaptable based on the role of the operator using the interface. For example, in a search and rescue operation there may be one operator who is in charge of moving the robot while another is in charge of searching the environment. Further, consider the director of the search operation who may not be in charge of operating a robot but may require information about what has been explored, what has been found, and how resources are being used. Each individual may require different sets of information to adequately perform their task. If too much information is provided then the cognitive workload to understand the required information for a particular task will lead to decreased performance. Similarly, too little information will also lead to decreased performance. Moreover, it is reasonable to consider that multiple operators may use the same interface at the same or different times based on the situation of the response and available resources. Therefore, it would be interesting to try and find a satisfying balance between the information needs of multiple operators performing different tasks.

Lastly, most of our research has been under the assumption that an operator will be in charge of the navigational responsibilities of the robot. Recent advancements in intelligent navigation algorithms demonstrate that intelligent vehicles can now traverse very difficult environments successfully without human supervision [26]. When considering human-robot interactions with an intelligent robot, communication is best when the human understands the decision process of the robot. This could be facilitated by a 3D interface wherein cues are presented to the operator concerning

the intent of the robot. Moreover, it would be interesting to study how and when robot intelligence might help an operator accomplish a task with a robot in comparison to not having intelligence on the robot. Following such a path could enable the comparison of how the interface and the intelligence on the robot can be combined to improve robot usability.

Bibliography

[1] AAAI rescue robot competition. www.cs.uml.edu/aaairobot/.

[2] J. A. Albus. Outline for a theory of intelligence. *IEEE Transactions on Systems, Man, and Cybernetics*, 21(3):473–509, 1991.

[3] P. L. Alfano and G. F. Michel. Restricting the field of view: perceptual and performance effects. *Perceptual and Motor skills*, 70(1):35–45, 1990.

[4] R. Ambrose, H. Aldridge, R. Askew, R. Burridge, W. Bluethmann, M. Diftler, C. Lovchik, D. Magruder, and F. Rehnmark. Robonaut: NASA's space humanoid. *IEEE Intelligent Systems*, 15(4):57–63, 2000.

[5] R. C. Arkin. Behavior-based robot navigation for extended domains. *Adaptive Behavior*, 1(2):201–225, 1992.

[6] K. Arthur. *Effects of field of view on performance with head-mounted displays*. Ph.D. Dissertation, University of North Carolina at Chapel Hill, 2000.

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

[8] M. Baker, R. Casey, B. Keyes, and H. A. Yanco. Improved interfaces for human-robot interaction in urban search and rescue. In *Proceedings of the IEEE Conference on Systems, Man and Cybernetics (SMC)*, The Hague, The Netherlands, October 2004.

[9] R. P. Bonasso, R. J. Firby, E. Gat, D. Kortenkamp, D. P. Miller, and M. G. Slack. Experiences with an architecture for intelligent, reactive agents. *Journal of experimental and theoretical artificial intelligence*, 9(2):237–256, 1997.

[10] J. Bradshaw, M. Sierhuis, A. Acquisti, P. Fetovich, R. Hoffman, R. Jeffers, D. Prescott, N. Suri, A. Uszok, and R. Hoff. Adjustable autonomy and human-agent teamwork in practice: An interim report on space applications. In *Agent Autonomy*, pages 9–38. Kluwer, 2002.

[11] C. Breazeal. *Designing sociable robots*. The MIT Press, Cambridge, MA, 2002.

[12] R. A. Brooks. A robust layered control system for a mobile robot. *IEEE Transactions on Robotics and Automation*, 2(1):14–23, 1986.

[13] D. Bruemmer, 2005. Personal communication.

[14] D. J. Bruemmer, D. D. Dudenhoeffer, and J. L. Marble. Dynamic autonomy for urban search and rescue. In *Proceedings of the 2002 AAAI Mobile Robot Workshop*, Edmonton, Canada, August 2002.

[15] D. J. Bruemmer, J. L. Marble, D. D. Dudenhoeffer, M. O. Anderson, and M. D. McKay. Mixed-initiative control for remote characterization of hazardous environments. In *Proceedings of the Hawaii International Conference on System Sciences*, Waikoloa, Hawaii, January 2003.

[16] D. J. Bruemmer, J. L. Marble, D. A. Few, R. L. Boring, M. C. Walton, and C. W. Nielsen. Shared understanding for collaborative control. *IEEE Transactions on Systems, Man, and Cybernetics—Part A*, 35(4):494–504, 2004.

[17] W. Burgard, A. B. Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schultz, W. Steiner, and S. Thrun. Experiences with an interactive museum tour-guide robot. *Artificial Intelligence*, 114(1-2):3–55, 1999.

[18] J. L. Burke, R. R. Murphy, M. D. Coovert, and D. L. Riddle. Moonlight in miami: A field study of human-robot interaction in the context of an urban search and rescue disaster response training exercise. *Human-Computer Interaction*, 19(1&2):85–116, 2004.

[19] J. L. Burke, R. R. Murphy, E. Rogers, V. J. Lumelsky, and J. Scholtz. Final report for the DARPA/NSF interdisciplinary study on human-robot interaction. *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, 34(2):103–112, 2004.

[20] R. Burridge and J. Graham. Providing robotics assistance during extravehicular activity. In *Proceedings of the SPIE - The International Society for Optical Engineering*, volume 4573, pages 22–33, 2002.

[21] Z. Byers, M. Dixon, K. Goodier, C. M. Grimm, and W. D. Smart. An autonomous robot photographer. In *Proceedings of the International Conference on Robots and Systems (IROS 2003)*, pages 2636–2641, Las Vegas, NV, October 2003.

[22] G. L. Calhoun, M. H. Draper, M. F. Abernathy, F. Delgado, and M. Patzek. Synthetic vision system for improving unmanned aerial vehicle operator situation awareness. *Enhanced and Synthetic Vision 2005*, 5802(1):219–230, 2005.

[23] J. Casper and R. R. Murphy. Human-robot interactions during the robot-assisted urban search and rescue response at the world trade center. *IEEE Transactions on Systems, Man, and Cybernetics Part B*, 33(3):367–385, 2003.

[24] G. Chronis and M. Skubic. Robot navigation using qualitative landmark states from sketched route maps. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1530–1535, New Orleans, LA, 2004.

[25] J. W. Crandall, M. A. Goodrich, D. R. Olsen Jr., and C. W. Nielsen. Validating human-robot interaction schemes in multi-tasking environments. *IEEE Transactions on Systems, Man, and Cybernetics, Special issue on Human-Robot Interaction*, 35(4):438–449, 2004.

[26] DARPA Grand Challenge. www.grandchallenge.org.

[27] P. Dourish and V. Bellotti. Awareness and coordination in shared workspaces. In *Proceedings of the ACM conference on Computer-supported cooperative work (CSCW)*, pages 107–114, Toronto, Ontario, 1992.

[28] D. Drascic. Skill acquisition and task performance in teleoperation using monoscopic and stereoscopic video remote viewing. In *Proceedings of Human Factors Society 35th Annual Meeting*, San Francisco, CA, 1991.

[29] D. Drascic and J. J. Grodski. Using stereoscipic video for defence teleoperation: A preliminary study. In *Proceedings of SPIE Vol. 1915: Stereoscopic Displays and Applications IV*, San Jose, CA, 1993.

[30] D. Drascic, J. J. Grodski, P. Milgram, K. Ruffo, P. Wong, and S. Zhai. ARGOS: A display system for augmenting reality. In *Proceedings of InterCHI Conference on Human Factors in Computing Systems*, page 521, Amsterdam, The Netherlands, 1993.

[31] D. Drascic and P. Milgram. Positioning accuracy of a virtual stereoscopic pointer in a real stereoscipic video world. In *Proceedings of SPIE vol. 1457: Stereoscopic Displays and Applications II*, San Jose, CA, 1991.

[32] D. Drascic and P. Milgram. Perceptual issues in augmented reality. In *Proceedings of SPIE vol. 2653: Stereoscopic Displays and Virtual Reality Systems III*, San Jose, CA, 1996.

[33] J. L. Drury, J. Scholtz, and H. A. Yanco. Awareness in human-robot interactions. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Washington D.C., October 2003.

[34] A. Elfes. Sonar-based real-world mapping and navigation. *IEEE Journal of robotics and automation*, 3(3):249–265, June 1987.

[35] M. R. Endsley. Design and evaluation for situation awareness enhancement. In *Proceedings of the Human Factors Society 32nd Annual Meeting*, pages 97–101, Santa Monica, CA, 1988.

[36] M. R. Endsley. Automation and situation awareness. In R. Parasuraman and M. Mouloua, editors, *Automation and human performance: Theory and applications*. Lawrence Erlbaum, Mahwah, NJ, 1996.

[37] S. Feiner, B. MacIntyre, M. Haupt, and E. Solomon. Windows on the world: 2D windows for 3D augmented reality. In *Proceedings of ACM Symposium on User Interface Software and Technology*, pages 145–155, Atlanta, GA, 1993.

[38] T. Fong, H. Pangels, and D. Wettergreen. Operator interfaces and network-based participation for Dante II. In *SAE 25th International Conference on Environmental Systems (ICES)*, San Diego, CA, July 1995.

[39] T. Fong, C. Thorpe, and C. Baur. Robot as partner: Vehicle teleoperation with collaborative control. In A. Schultz and L. Parker, editors, *Proceedings of the NRL Workshop on Multi-Robot Systems*. Kluwer, 2002.

[40] T. Fong, C. Thorpe, and C. Baur. A safeguarded teleoperation controller. In *IEEE International Conference on Advanced Robotics (ICRA)*, Budapest, Hungary, August 2001.

[41] T. W. Fong and C. Thorpe. Vehicle teleoperation interfaces. *Autonomous Robots*, 11(1):9–18, 2001.

[42] T. W. Fong, C. Thorpe, and C. Baur. Advanced interfaces for vehicle teleoperation: Collaborative control, sensor fusion displays, and remote driving tools. *Autonomous Robots*, 11(1):77–85, 2001.

[43] K. J. Gergen. *Realities and Relationships: Soundings in social construction*. Harvard University Press, Cambridge, MA, 1994.

[44] J. J. Gibson. *The ecological approach to visual perception*. Houghton Mifflin, Boston, MA, 1979.

[45] J. J. Gibson and L. E. Crooks. A theoretical field-analysis of automobile-driving. *The American Journal of Psychology*, 51(3):453–471, 1938.

[46] M. A. Goodrich, E. R. Boer, J. W. Crandall, R. W. Ricks, and M. L. Quigley. Behavioral entropy in human-robot interaction. In *Proceedings of Performance Metrics for Intelligent Systems*, Gaithersburg, MD, August 24-26, 2004.

[47] M. A. Goodrich, D. R. Olsen Jr., J. W. Crandall, and T. J. Palmer. Experiments in adjustable autonomy. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI) Workshop on Autonomy, Delegation, and Control: Interacting with Autonomous Agents*, pages 1624–1629, Seattle, WA, 2001.

[48] M. A. Goodrich, R. J. Rupper, and C. W. Nielsen. Perceiving head, shoulders, eyes, and toes in augmented virtuality interfaces for mobile robots. In *Proceedings of the 14th IEEE International Workshop on Robot and Human Interactive Communication (RO-MAN)*, Nashville, TN, 2005.

[49] J. Harris. External memory aids. In M. Gruneberg, P. Morris, and R. Sykes, editors, *Practical Aspects of Memory*. Academic Press, London, England, 1978.

[50] M. Hearst. Mixed-initiative interaction—trends and controversies. *IEEE Intelligent Systems*, 14(5):14–23, 1999.

[51] C. Heeter. Being there: The subjective experience of presence. *Presence, teleoperators, and virtual environments*, 1(2):262–271, 1992.

[52] H. v. Helmholtz. Treatise on physiological optics vol 3. translated and edited by P.C. Southall, New York, Dover, 1962.

[53] B. Hine, P. Hontalas, T. Fong, L. Piguet, E. Nygren, and A. Kline. VEVI: A virtual environment teleoperation interface for planetary exploration. In *SAE 25th international conference on environmental systems (ICES)*, San Diego, CA, July 1995.

[54] S. Hirose and E. Fukushima. Development of mobile robots for rescue operations. *Advanced Robotics*, 16(6):509–512, 2002.

[55] S. Hughes, J. Manojlovich, M. Lewis, and J. Gennari. Camera control and decoupled motion for teleoperation. In *proceedings of the 2003 IEEE International Conference on Systems, Man, and Cybernetics*, Washington, D.C., 2003.

[56] S. Iba, J. Weghe, C. Paredis, and P. Khosla. An architecture for gesture based control of mombile robots. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 851–857, Kyongju, Korea, October 1999.

[57] A. Jacoff, E. Messina, and J. Evans. A reference test course for autonomous mobile robots. In *Proceedings of SPIE vol. 4364: AeroSense Conference*, pages 341–348, Orlando, FL, April 2001.

[58] A. Jacoff, E. Messina, and J. Evans. A standard test course for urban search and rescue robots. In *Proceedings of the Performance Metrics for Intelligent Systems (PERmis) Workshop*, pages 253–259, Gaithersburg, MD, August 2000.

[59] C. A. Johnson, A. Koku, K. Kawamura, and R. Peters II. Enhancing a human-robot interface using a sensory egosphere. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Washington D.C., May 11-15, 2002.

[60] H. Jones, S. Rock, D. Burns, and S. Morris. Autonomous robots in swat applications: Research, design, and operations challenges. In *Proceedings of the Symposium for the Association of Unmanned Vehicle Systems International (AUVSI)*, Orlando, FL, 2002.

[61] K. I. Kang, S. Freedman, and M. J. Matarić. A hands-off physical therapy assistance robot for cardiac patients. In *Proceedings of the IEEE International Conference on Rehabilitation Robotics (ICORR)*, Chicago, IL, June, 2005.

[62] I. Kant. *Critique of Pure Reason*. Cambridge University Press, Cambridge, UK, 1998. Original from 1787, Edited and Translated by Paul Guyer and Allen W. Wood.

[63] H. Keskinpala, J. Adams, and K. Kawamura. A PDA-based human robotic interface. In *Proceedings of the international conference on Systems, Man, and Cybernetics (SMC)*, Washington D.C., 2003.

[64] K. Konolige. Large-scale map-making. In *Proceedings of the National Conference on AI (AAAI)*, San Jose, CA, 2004.

[65] D. Kortenkamp, R. Bonasso, D. Ryan, and D. Schreckenghost. Traded control with autonomous robots as mixed initiative interaction. In *AAAI Spring Symposium on Mixed Initiative Interaction*, Stanford, CA, 1997.

[66] P. Kroft and C. Wickens. Displaying multi-domain graphical database information: An evaluation of scanning, clutter, display size, and user activity. *Information Design Journal*, 11(1):44–52, 2002.

[67] E. Krotkov, R. Simmons, F. Cozman, and S. Koenig. Safeguarded teleoperation for lunar rovers: From human factors to field trials. In *IEEE Planetary Rover Technology and Systems Workshop*, Minneapolis, MN, April 1996.

[68] R. Kubey and M. Csikszentmihalyi. Television addiction is no mere metaphor. *Scientific American*, 286(2):62–68, 2002.

[69] B. Kuipers. The spatial semantic hierarchy. *Artificial Intelligence*, 119(1-2):191–233, 2000.

[70] B. Kuipers and Y.-T. Byun. A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. *Journal of robotics and autonomous systems*, 8:47–63, 1991. Reprinted in Walter Van de Velde (ed.), *Towards Learning Robots*, Bradford/MIT Press, 1993, pages 47-63.

[71] J. D. Lee, B. Caven, S. Haake, and T. L. Brown. Speech-based interaction with in-vehicle computers: The effect of speech-based email on driver's attention to the roadway. *Human Factors*, 43:631–640, 2001.

[72] J. Lehikoinen. An evaluation of augmented reality navigational maps in head-worn displays. In *Proceedings of the Human-Computer Interaction Conference (INTERACT)*, Tokyo, Japan, July, 2001.

[73] J. Lehikoinen and R. Suomela. Perspective map. In *Proceeding of the Sixth International Symposium on Wearable Computers (ISWC)*, Seattle, WA, October 2002.

[74] M. Lewis and J. Jacobson. Game engines in research. *Communications of the ACM*, 45(1):27–48, 2002.

[75] M. Lewis, K. Sycara, and I. Nourbakhsh. Developing a testbed for studying human-robot interaction in urban search and rescue. In *10th International Conference on Human-Computer Interaction*, Crete, Greece, 2003.

[76] F. Lu and E. Milios. Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, 4(4):333–349, 1997.

[77] G. Mantovani and G. Riva. Real presence: How different ontologies generate different criteria for presence, telepresence, and virtual presence. *Presence: Teleoperators and Virtual Environments*, 8(5):538–548, 1999.

[78] J. L. Marble, D. J. Bruemmer, D. A. Few, and D. D. Dudenhoeffer. Evaluation of supervisory vs. peer-peer interaction with human-robot teams. In *Proceedings of the 37th Annual Hawaii International Conference on System Sciences (HICSS)*, Waikoloa, Hawaii, January, 2004.

[79] J. Meacham and B. Leiman. Remembering to perform future actions. In U. Neisser, editor, *Memory Observed*, pages 326–337. Freeman, San Francisco, CA, 1982.

[80] R. Meier, T. Fong, C. Thorpe, and C. Baur. A sensor fusion based user interface for vehicle teleoperation. In *International conference on field and service robotics (FSR)*, Pittsburgh, PA, 1999.

[81] P. Milgram and F. Kishino. A taxonomy of mixed reality visual displays. *IEICE Transactions on Information Systems*, E77-D(12):1321–1329, 1994.

[82] P. Milgram, A. Rastogi, and J. J. Grodski. Telerobotic control using augmented reality. In *Proceedings of the 4th IEEE International Workshop on Robot and Human Communication (RO-MAN)*, Tokyo, Japan, 1995.

[83] P. Milgram, H. Takemura, A. Utsumi, and F. Kishino. Augmented reality: A class of displays on the reality-virutality continuum. In H. Das, editor, *Proceedings of SPIE vol. 2351: Telemanipulator and Telepresence Technologies*, pages 282–292, 1994.

[84] P. Milgram, S. Zhai, D. Drascic, and J. Grodski. Applications of augmented reality for human-robot communication. In *Proceedings of the international conference on Intelligent Robots and Systems (IROS)*, pages 1467–1472, Yokohama, Japan, July 1993.

[85] H. P. Moravec. Sensor fusion in certainty grids for mobile robots. *AI Magazine*, 9(2):61–74, 1988.

[86] R. R. Murphy. Human-robot interaction in rescue robotics. *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, 34(2):138–153, 2004.

[87] R. R. Murphy. Humans, robots, rubble, and research. *Interactions*, 12(2):37–39, 2005.

[88] R. R. Murphy. Case studies of applying Gibson's ecological approach to mobile robots. *IEEE Transactions on Systems, Man, and Cybernetics–Part A*, 29(1):105–111, January 1999.

[89] R. R. Murphy and E. Rogers. Cooperative assistance for remote robot supervision. *Presence*, 5(2):224–240, 1996.

[90] O. Nakayama, T. Futami, T. Nakamura, and E. Boer. Development of a steering entropy method for evaluating driver workload. In *SAI Technical Paper Series:*

#1999-01-0892: Presented at the International Congress and Exposition, Detroit, MI, March 1999.

[91] S. Nayar. Catadioptric omnidirectional camera. Technical report, Bell Laboratories, Holmdel, NJ, 1997.

[92] L. A. Nguyen, M. Bualat, L. J. Edwards, L. Flueckiger, C. Neveu, K. Schwehr, M. D. Wagner, and E. Zbinden. Virtual reality interfaces for visualization and control of remote vehicles. *Autonomous Robots*, 11(1):59–68, 2001.

[93] A. A. Nofi. Defining and measuring shared situation awareness. Center for Naval Analyses, November 2000.

[94] D. A. Norman. *The design of everyday things*. Doubleday, New York, NY, 1988. Previously published as *The psychology of everyday things*.

[95] D. A. Norman. Affordance, conventions, and design. *Interactions*, 6(3):38–43, 1999.

[96] D. R. Olsen Jr. and S. B. Wood. Fan-out: Measuring human control of multiple robots. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 231–238, Vienna, Austria, 2004.

[97] L. Piguet, T. W. Fong, B. Hine, P. Hontalas, and E. Nygren. VEVI: A virtual reality tool for robotic planetary exploration. In *Proceedings of Virtual Reality World*, Munich, Germany, February, 1995.

[98] L. Piguet, B. Hine, P. Hontalas, T. W. Fong, and E. Nygren. The virtual environment vehicle interface: a dynamic, distributed, and flexible virtual environment. In *IMAGINA '96: New Frontiers of CyberExistence*, Monaco, France, February, 1996.

[99] J. Pineau, M. Montemerlo, M. Pollack, N. Roy, and S. Thrun. Towards robotic assistants in nursing homes: Challenges and results. *Robotics and Autonomous Systems*, 42(3-4):271–281, 2003.

[100] J. Rasmussen. Outlines of a hybrid model of the process plant operator. In T. Sheridan and G. Johannsen, editors, *Monitoring Behavior and Supervisory Control*, pages 371–383. Plenum, 1976.

[101] B. W. Ricks. An ecological display for robot teleoperation. Master's thesis, Brigham Young University, August 2004.

[102] B. W. Ricks, C. W. Nielsen, and M. A. Goodrich. Ecological displays for robot interaction: A new perspective. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sendai, Japan, 2004.

[103] W. Robinett. Synthetic experience: A proposed taxonomy. *Presence, Teleoperators and virtual environments*, 1(2):229–247, 1992.

[104] P. Scerri, D. Pynadath, and M. Tambe. Adjustable autonomy in real-world multi-agent environments. In *Proceedings of the fifth international conference on autonomous agents (AGENTS)*, Montreal, Canada, 2001.

[105] D. W. Schloerb. A quantitative measure of telepresence. *Presence: Teleoperators and Virtual environments*, 4(1):64–80, 1995.

[106] J. Scholtz. Human-robot interactions: Creating synergistic cyber forces. In *Proceedings from the 2002 NRL Workshop on Multi-Robot Systems*, Washington, D.C., March 2002.

[107] D. Schulz, W. Burgard, D. Fox, S. Thrun, and A. Creemers. Web interfaces for mobile robots in public places. *IEEE Robotics and Automation Magazine*, 7(1):48–56, 2000.

[108] R. Shepard and J. Metzler. Mental rotation of three-dimensional objects. *Science*, 171:701–703, 1971.

[109] T. Sheridan. Musings on telepresence and virtual presence. *Presence, teleoperators, and virtual environments*, 1(1):120–125, 1992.

[110] T. B. Sheridan. *Telerobotics, automation, and human supervisory control*. the MIT Press, Cambridge, MA, 1992.

[111] B. Shneiderman and C. Plaisant. *Designing the User Interface: Strategies for Effective Human-Computer Interaction, 4th Edition*. Addison-Wesley, 2004.

[112] M. Skubic, D. Perznowski, S. Blisard, A. Schultz, W. Adams, M. Bugajska, and D. Brock. Spatial language for human-robot dialogs. *IEEE Transactions on Systems, Man, and Cybernetics-Part C*, 34(2):154–167, 2004.

[113] M. Slater, M. Usoh, and A. Steed. Depth of presence in virtual environments. *Presence, teleoperators, and virtual environments*, 3(2):130–144, 1994.

[114] M. Slater and S. Wilbur. A framework for immersive virtual environments (FIVE): speculations on the role of presence in virtual environments. *Presence, teleoperators, and virtual environments*, 6(1):603–616, 1997.

[115] H. S. Smallman and M. S. John. Naïve realism: Misplaced faith in the utility of realistic displays. *Ergonomics in Design*, 13(3), 2005.

[116] J. Steuer. Defining virtual reality: Dimensions determining telepresence. *Journal of Communication*, 42(2):73–93, 1992.

[117] C. Stoker and B. Hine. Telepresence control of mobile robots: Kilauea marsokhod experiment. In *Proceedings of the American Institute of Aeronautics and Astronautics*, Reno, NV, 1996.

[118] C. Stoker, E. Zbinden, T. Blackmon, B. Kanefsky, and et al. Analyzing pathfinder data using virtual reality and superresolved imaging. *Journal of Geophysical Research–Planets*, 104(E4):8889–8906, 1999.

[119] C. Stoker, E. Zbinden, T. Blackmon, and L. Nguyen. Visualizing mars using virtual reality: A state of the art mapping tool used on mars pathfinder. In *Extraterrestrial Mapping Symposium: Mapping of Mars (ISPRS)*, Caltech, Pasadena, CA, 1999.

[120] R. Suomela, K. Roimela, and J. Lehikoinen. The evolution of perspective view in WalkMap. *Personal and Ubiquitous Computing*, 7(5):249–262, 2003.

[121] G. Thomas, T. Blackmon, M. Sims, and D. Rasmussen. Video engraving for virtual environments. In *Proceedings of Electronic Imaging: Science and Technology*, San Jose, CA, 1997.

[122] G. Thomas, W. D. Robinson, and S. Dow. Improving the visual experience for mobile robotics. In *Seventh annual Iowa Space Grant Proceedings*, pages 10–20, Des Moines, IA, November 1997.

[123] L. C. Thomas and C. D. Wickens. Eye-tracking and individual differences in off-normal event detection when flying with a synthetic vision system display. Proceedings of the Human Factors and Ergonomics Society 48th Annual Meeting., 2004.

[124] S. Thrun. Robotic mapping: A survey. In G. Lakemeyer and B. Nebel, editors, *Exploring Artificial Intelligence in the New Millennium*. Morgan Kaufmann, 2002.

[125] S. Thrun, M. Beetz, M. Bennewitz, W. Burgard, A. Cremers, F. Dellaert, D. Fox, D. Hähnel, C. Rosenberg, N. Roy, J. Schulte, and D. Schulz. Probabilistic algorithms and the interactive museum tour-guide robot MINERVA. *Journal of Robotics Research*, 19(11):972–999, 2000.

[126] S. Thrun, W. Burgard, and D. Fox. A probabilistic approach to concurrent mapping and localization for mobile robots. *Machine Learning*, 31:29–53, 1998. also appeared in Autonomous Robots 5:253-271 (joint issue).

[127] E. Titchener. *A textbook of psychology*. Macmillan, New York, NY, 1924.

[128] J. S. Tittle, A. Roesler, and D. D. Woods. The remote perception problem. In *Human Factors and Ergonomics Society 46th annual meeting*, Baltimore, MD, 2002.

[129] I. Vekiri. What is the value of graphical displays in learning? *Educational Psychology Review*, 14(3):261–312, 2002.

[130] M. G. Voshell and D. D. Woods. Breaking the keyhole in human-robot coordination: Method and evaluation. Technical report, Ohio State University, 2005. Submitted to HFES 2005.

[131] S. Waldherr, R. Romero, and S. Thrun. A gesture based interface for human-robot interaction. *Autonomous Robots*, 9(2):151–173, 2000.

[132] J. Wang, M. Lewis, and J. Gennari. A game engine based simulation of the NIST urban search and rescue arenas. In *Proceedings of the 2003 Winter Simulation Conference*, New Orleans, LA, 2003.

[133] C. Ware, K. Arthur, and K. Booth. Fish tank virtual reality. In *Proceedings of the InterCHI Conference on Human Factors in Computing Systems*, pages 37–42, Amsterdam, The Netherlands, 1993.

[134] D. Wegner. Transactive meory: A contemporary analysis of the group mind. In B. Mullen and G. Goethals, editors, *Theories of Group Behavior*, pages 185–208. Springer-Verlag, New York, NY, 1986.

[135] C. Wickens. Situation awareness and workload in aviation. *Current Directions in Psychological Science*, 11(4):128–133, 2002.

[136] C. D. Wickens and J. G. Hollands. *Engineering psychology and human performance 3rd edition*. Prentice Hall, 1999.

[137] D. Woods and J. Watts. How not to have to navigate through too many displays. In M. H. 2nd edition, T. Landauer, and P. Prabhu, editors, *Handbook of Human-Computer Interaction*, pages 1177–1201. Elsevier Science, Amsterdam, The Netherlands, 1997.

[138] D. D. Woods, J. Tittle, M. Feil, and A. Roesler. Envisioning human-robot coordination in future operations. *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, 34(2):210–218, 2004.

[139] K. Yamazawa, Y. Yagi, and M. Yachida. Obstacle avoidance with omnidirectional image sensor hyperomni vision. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1062–1067, Nagoya, Japan, 1995.

[140] H. A. Yanco and J. L. Drury. "Where am I?" Acquiring situation awareness using a remote robot platform. In *Proceedings of the IEEE Conference on Systems, Man, and Cybernetics*, The Hague, The Netherlands, October 2004.

[141] H. A. Yanco, J. L. Drury, and J. Scholtz. Beyond usability evaluation: Analysis of human-robot interaction at a major robotics competition. *Journal of Human-Computer Interaction*, 19(1&2):117–149, 2004.

[142] L. Yu, P. Tsui, Q. Zhou, and H. Hu. A web-based telerobotic system for research and education at Essex. In *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Como, Italy, July 2001.

[143] P. Zahorik and R. L. Jenison. Presence as being-in-the-world. *Presence*, 7(1):78–89, 1998.