Optimizing Agent Behavior and Minimizing User Cognitive Load for Mixed Human-Robot Teams

O'Hara, Christopher

Award date:
2020

Awarding institution:
Technische Universität Berlin

Technische Universität Berlin

Optimizing Agent Behavior and Minimizing User Cognitive Load for Mixed Human-Robot Teams

Master Thesis
am Fachgebiet Agententechnologien in betrieblichen Anwendungen und der Telekommunikation (AOT)
Prof. Dr.-Ing. habil. Sahin Albayrak
Fakultät IV Elektrotechnik und Informatik
Technische Universität Berlin

vorgelegt von
Christopher O'Hara

Betreuer: Dr.-Ing. Christopher-Eyk Hrabia,
Gutachter: Prof. Dr.-Ing. habil. Sahin Albayrak
Prof. Dr. Odej Kao

Christopher O'Hara
Matrikelnummer: 406037

Erklärung der Urheberschaft

Ich erkläre hiermit an Eides statt, dass ich die vorliegende Arbeit ohne Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe; die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht. Die Arbeit wurde bisher in gleicher oder ähnlicher Form in keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Ort, Datum Unterschrift

Abstract

Autonomous systems continue to improve at specialized tasks allowing for a higher quality of life for many individuals. Intelligent robots have historically assumed tasks that would otherwise be arduous, vacuous, or even dangerous for humans. Specifically designed for distinct tasks, these machines do not grow tired over time and have efficiency and accuracy rates not typically achievable by humans. Personal safety appears to be a field that could benefit from the inclusion of robotic systems, especially in police and military operations. Adding *Unmanned Aerial Vehicles* (UAVs) to the special forces teams, or other similar operations, allows for a unique opportunity to minimize the danger of field operators during their missions. Special Forces operations are often conducted in confined, indoor environments where mobility and visibility are limited. Concurrently with the stressful nature of their tasks, safety-focused tactical gear negatively impacts vision, mobility, and cognitive load. The goal of this thesis is to reduce the cognitive load of users through proper system design and integration.

Since Special Forces groups heavily rely on voice communication, a drone with speech recognition has been designed and tested in a virtual environment. The goal was to discover meaningful insights on user interaction that would lead to the improvement of the user interface (UI) for a drone communicating back with the operative about its tasks. Users tested three types of UIs and qualitative and quantitative information has been analyzed to evaluate which UI is the best candidate for pursuing future research and implementation. These results are expected to be useful in the creation of an optimal UI that mitigates user cognitive load, improves both operative and agent behavior (accuracy of tasks), and ultimately mission success. This combination will advance the field of human-robot interaction and allow for mixed human-robot teams. Three reputable user testing frameworks were utilized including the NASA-TLX, PVT, and SAGAT. This thesis was conducted with the *Distributed Artificial Intelligence Laboratory* (DAI-Labor) at the Technische Universität Berlin as a component of the InLaSeD (Indoor Situation Survey for Special Units with Drones) for the German Special Forces (SEK).

The results include the development of a voice recognizer with a custom dictionary for special operations. A high-fidelity environment was created in Unity with communications sent to ROS to replicate agent communication behaviors and delays. Users

evaluated three types of models including voice-only, visual-only, and multimodal user interfaces. The outcome is that, compared to single modalities, the multimodal implementation is the most robust with respect to performance and cognitive load, with analyzable trends in situational awareness and response time. Future work will be to further improve the multimodal user interface to continue to decrease cognitive load for SEK operatives and improve artificial agent behavior with regard to interpretation and decision-making behaviors.

Zusammenfassung

Autonome Systeme verbessern sich ständig bei speziellen Aufgaben, die eine höhere Lebensqualität für viele Menschen ermöglichen. Intelligente Roboter haben in der Vergangenheit Aufgaben übernommen, die sonst mühsam, leer oder sogar gefährlich für den Menschen wären. Diese Maschinen, die speziell für unterschiedliche Aufgaben entwickelt wurden, werden mit der Zeit nicht müde und haben eine Effizienz und Genauigkeit, die normalerweise vom Menschen nicht erreicht werden kann. Die persönliche Sicherheit scheint ein Bereich zu sein, der von der Einbeziehung von Robotersystemen profitieren könnte, insbesondere bei Polizei- und Militäreinsätzen. Die Aufnahme von unbemannten Luftfahrzeugen (UAVs) in die Spezialeinheiten oder ähnliche Einsätze ermöglicht eine einzigartige Möglichkeit, die Gefahr für die Feldbediener während ihrer Einsätze zu minimieren. Spezialeinsätze werden oft in engen, geschlossenen Räumen durchgeführt, in denen Mobilität und Sicht eingeschränkt sind. Gleichzeitig mit der stressigen Natur ihrer Aufgaben wirkt sich die sicherheitsorientierte taktische Ausrüstung negativ auf das Sehen, die Mobilität und die kognitive Belastung aus. Das Ziel dieser Arbeit ist es, die kognitive Belastung der Benutzer durch richtiges Systemdesign und Integration zu reduzieren.

Da die Gruppen der Spezialeinheiten stark auf Sprachkommunikation angewiesen sind, wurde eine Drohne mit Spracherkennung in einer virtuellen Umgebung entwickelt und getestet. Ziel war es, aussagekräftige Erkenntnisse über die Benutzerinteraktion zu gewinnen, die zur Verbesserung der Benutzeroberfläche (UI) für eine Drohne führen würden, die mit dem Operator über ihre Aufgaben kommuniziert. Benutzer testeten drei Arten von Benutzeroberflächen und qualitative und quantitative Informationen wurden analysiert, um zu bewerten, welche Benutzeroberfläche der beste Kandidat für die zukünftige Forschung und Implementierung ist. Diese Ergebnisse sollen bei der Erstellung einer optimalen Benutzeroberfläche nützlich sein, die die kognitive Belastung des Benutzers mildert, sowohl das operative als auch das Agentenverhalten (Genauigkeit der Aufgaben) verbessert und letztendlich den Missionserfolg steigert. Diese Kombination wird das Feld der Mensch-Roboter-Interaktion vorantreiben und gemischte Mensch-Roboter-Teams ermöglichen. Es wurden drei renommierte Anwendertestsysteme verwendet, darunter NASA-TLX, PVT und SAGAT. Diese Arbeit wurde mit dem Distributed Artificial Intelligence Laboratory (DAI-Labor) an der Technischen Univer-

sität Berlin als Bestandteil der InLaSeD (Indoor Situation Survey for Special Units with Drones) für die Deutschen Spezialeinheiten (SEK) durchgeführt.

Zu den Ergebnissen gehört die Entwicklung eines Spracherkenners mit einem benutzerdefinierten Dictionary für spezielle Operationen. In Unity wurde eine High-Fidelity-Umgebung mit an ROS gesendeten Kommunikationen geschaffen, um das Kommunikationsverhalten und die Verzögerungen der Agenten zu replizieren. Die Benutzer bewerteten drei Arten von Modellen, darunter sprachbasierte, visuelle und multimodale Benutzeroberflächen. Das Ergebnis ist, dass im Vergleich zu einzelnen Modalitäten, die multimodale Implementierung ist die robusteste in Bezug auf Leistung und kognitive Belastung, mit analysierbaren Trends in der situativen Wahrnehmung und Reaktionszeit. Zukünftige Arbeiten werden darin bestehen, die multimodale Benutzeroberfläche weiter zu verbessern, um die kognitive Belastung für SEK-Mitarbeiter weiter zu verringern und das Verhalten künstlicher Agenten in Bezug auf Interpretations- und Entscheidungsverhalten zu verbessern.

Acknowledgements

I would like to thank Christopher-Eyk Hrabia, the DAI-Labor, and the SEK for providing me with the opportunity to work on this topic. I have been highly interested in mixed human-robot teams for special forces missions for as long as I can remember. I would also like to thank EIT Digital for an amazing opportunity to pursue multiple degrees in multiple countries simultaneously. Additionally, I would like to thank TU/e and TU Berlin for having amazing research facilities that allowed me to develop and explore topics related to robotics, artificial intelligence, cybernetics, and cognitive science. Especially, at TU Berlin: the Agent Core Technologies (CC-ACT) at DAI-Labor, the Robotics and Biology Laboratory (RBO), Raumfahrttechnik (RFT), and Mensch-Maschine-Systeme (MMS) departments. At TU/e, the Human-Technology Interaction (HTI) department help define a lot of the approaches and tasks conducted in this thesis. Special thanks to Dr. Dajsuren for all of her assistance and support which allowed the overall quality to be much better. Approximately 500 liters of Monster Energy were consumed in the making of this thesis.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Autonomous systems continue to improve at specialized tasks allowing for a higher quality of life for many individuals. Intelligent robots have historically assumed tasks that would otherwise be arduous, vacuous, or even dangerous for humans. Specifically designed for distinct tasks, these machines do not grow tired over time and have efficiency and accuracy rates not typically achievable by humans (i.e., the manufacturing of vehicles or throughput of products on an assembly line). However, these events are usually "behind the scenes," as little-to-no human interaction is required. Recently, autonomous systems are proposed to coalesce and proliferate everyday activities via autonomous driving, product distribution, and tasks that require repetitive actions that do not require critical thinking. Intuitively, personal safety appears to be a field that could benefit from the inclusion of robotic systems as safety concerns nearly everyone.

Throughout the world, safety is a major concern for people and is a major factor of *Quality of Life* (QoL). In Germany alone, 23% of urban denizens report feeling unsafe due to crime in 2016 [EU, 2016]. There is a major need for improving the QoL for citizens, especially in urban areas. In general, police forces are responsible for the reduction of crime. Special forces units are called in whenever there are high-profile missions containing dangerous criminals (homicides, hostage situations, counterterrorism, etc.). In these situations, there is a high level of risk for the operatives that can result in permanent physical and mental damage or death. Another complication is that these operatives must wear additional protective gear. This gear blocks their ability to function ideally, as the total weight of a loadout can be upwards of 50kg and helmets/flak jackets restrict mobility. Fig. 1.1 shows an example of the heavy gear that SEK operatives wear [Moll, 2017]. To combat these limitations, mixed human-robot teams can be assembled, sending in robots in to assess certain dangers (reconnaissance), structural damage, and even first contact with the perpetrator. Ideally, the robot would be able to communicate with the police officer the location and status of the person of interest. This could increase the likelihood that the perpetrator is apprehended successfully and without harm, as operatives may injury or kill the suspect if startled. In short, au-

tonomous robots could improve the personal safety of civilians, officers, and suspects simultaneously.

Figure 1.1: Image of the Brandenburg SEK Force.

While certain aspects of autonomous systems currently prevent their widespread use in urban environments (privacy concerns, resources, technological feasibility, etc.), there are additional niche areas that can immediately benefit from mixed human-robot teams for safety outside of police work. These topics include firefighters, disaster relief, item retrieval, contraband and explosives disarmament, as well as other environments in which the presence of a robot would reduce the risk of danger for human operators. In terms of *interoperability* and *scalability*, human-robot system design can deliberately create commonalities that function across multiple systems with a standardized framework and approach. Robotic systems require the fulfillment of-functional requirements while also being *distributed systems*, concurrently sharing resources and information in real-time to optimize performance, reliability, and coordination [Brugali and Fayad, 2002].

Within the fields of autonomous systems and human-robot interaction, there is are seemingly endless opportunities. In the case of mixed human-robot teams, proper perception and cognition (for the robot) and safety-distance away from the user (*Personal-Space Model* [Torta et al., 2013]) is being analyzed with parametric models based on commonalities in user experience and comfort. Decision-making operations are currently in development for autonomous agents for reactive-adaptive hybrid behavior-based planning (*ROS Hybrid Behavior Planner* (RHBP) [Hrabia et al., 2017]). In fact,

simple navigation and interaction with robots is not a trivial task that cannot be solved solely with intuition. Each of these use cases can benefit from a *Simulator* created in *Virtual Reality*. Mass physical implementations are costly and prone to changes, making virtual environments ideal for simulation and validation [Mizuchi and Inamura, 2017]. However, most current simulated environments lack fidelity and realism which can lead to mismatches in the predicted behavior as compared to the actual behavior for robots, humans, and systems [Novikova et al., 2015].

Chapter 2

Motivation

This section describes the motivation, approach, and structure of the thesis.

2.1 Motivation

Drones can provide flexibility in various domains include package delivery (medicine, supplies, etc.), aerial reconnaissance, and communication arrays. Drones have an advantage over ground-based mobile and fixed robots, as they can take off vertically, are not susceptible to obstructions or slippage in the environment, and have more degrees of freedom in movement. Furthermore, they can move quite quickly at the cost of additional noise and energy consumption. Drones would make excellent companions for police field operations as they can scan environments for civilians, criminals, structural damage, or hazards. Drones are also replaceable whereas human life is not. Drones could also lead to improving mission success as a ”first respondent” on a scene, preventing human errors that occur when an officer or operative is caught by surprise (unintentionally eliminating a suspect).

To pursue this topic in-depth with respect to being able to operate such a drone, this Master's Thesis was conducted with the *Distributed Artificial Intelligence Laboratory* (DAI-Labor) at the Technische Universität Berlin. This thesis is a component of the Indoor-Lageerkundung für Spezialeinheiten mit Drohnen (InLaSeD) (Indoor Situation Survey for Special Units with Drones) project for the needs of the Brandenburg *Spezialeinsatzkommandos* (SEK) (*Special Deployment Commandos*), a German Special Forces group [Bundesministerium für Bildung und Forschung, 2019]. Special Forces operations are often conducted in confined, indoor environments where mobility and visibility are limited. The SEK is a special-operations, counter-terrorism, and high-risk unit handling critical missions including hostage sieges, building raids, and surveillance ([Traumberuf, 2019]).

Concurrently with the stressful nature of their tasks, safety-focused tactical gear

negatively impacts vision, mobility, and *cognitive load*. Cognitive load here is defined as the distribution and utilization of cognitive resources that impact personal performance, response time, situational awareness, working memory, and personal stress. The focus of this thesis is to evaluate how adding using a drone in a mixed human-robot team might impact the operator cognitively. Following this analysis, artificial agent performance can be optimized based on user behavior.

Adding *Unmanned Aerial Vehicles* (UAVs) to the SEK teams, or other similar operations, allows for a unique opportunity to minimize the danger of field operators during their missions. While certain military groups, like the *United States Marine Corps* (USMC), have been experimenting with drones (*Drohne*) during operations, these have largely been outdoors and a safe distance from the target with the drone acting as a spotter ([Rogoway, 2018]). The InLaSeD project will have additional challenges in which autonomous drones act as support members in mixed human-robot teams, providing information on target identification/classification, environmental hazards (smoke, fire), mapping and structural layout.

2.2 Approach and Goals

Since Special Forces groups heavily rely on voice communication, a drone with speech recognition has been designed and tested in a virtual environment. The goal was to discover meaningful insights on user interaction that would lead to the improvement of the UI for the drone communicating back with the operative about its tasks. Users tested three types of UIs and qualitative and quantitative information has been analyzed to evaluate which UI is the best candidate for pursuing future research and eventually implementation. These results are expected to be useful in the creation of an optimal UI that mitigates user cognitive load, improves both operative and agent behavior (accuracy of tasks), and ultimately mission success. This combination will advance the field of human-robot interaction and allow for mixed human-robot teams.

2.3 Structure of the Thesis

This thesis is structured as follows. In Chapter 3, some related background aspects that are considered fundamental to the research are discussed. Chapter 4 discusses the research questions and approach. In Chapter 5, the details of the experimental designs are covered with Chapter 6 detailing some additional design and testing aspects used to guide design decisions for the final version of the implementation. Chapter 7 covers the user testing aspects of the project. Evaluation results are discussed in Chapter 8. Chapter 9 contains the conclusion. Chapter 9.3 gives additional related information for data analysis, modeling, and the administered user test.

Chapter 3

Background

This Chapter contains the three main aspects of the thesis including the speech recognition, *User Interface* (UI) design, and cognitive load. Speech recognition is split into two primary domains. The UI design is approached based on multiple sensory channels and focuses on how to approach a feasible implementation. Cognitive load aspects are considered on how to evaluate if a UI design and ultimately, mission evaluations, are impacted by the designed UI and artificial agent teammate.

3.1 Speech Recognition

Controlling a drone via voice commands required several steps. First, *Automatic Speech Recognition* (ASR) must be implemented. Since the drone will only be required to recognize and respond to a small set of words or phrases, creating a custom dictionary is ideal and directly impacts the memory size of the data set. A small voculary results in more accurate interpretations when proper heuristics are implemented. Next, the interpretations needed to be processed and the drone's behavior needs to reflect the spoken dialog. Since the SEK is operating in dynamic and dangerous environments, this process needs to happen with high accuracy, precision, and little delay as possible.

ASR has two main approaches in implementation: either using *Hidden Markov Models* (HMM) or *Deep-Learning* (DL). Each implementation has a number of trade-offs and the manner of the implementation determines which approach is, in some sense, optimal. HMMs are typically easier to understand and implement (fewer parameters) at the cost of accuracy [Receveur and Fingscheidt, 2014]. Creating HMMs for ASR requires five steps: Feature Extraction, an Acoustic Model, a Lexicon Model, a Language Model and then a Decoder (usually the Forward-Backward or Viterbi algorithms) [Maas, 2017]. HMMs are good for problems that have a small number of states (grammar, words) and could allow for the drone to perform entirely offline [Ward, 2017].

There is already a large library of existing solutions and implementations utilizing the CMUSphinx API [CMU-Sphinx, 2019].

DL methods, usually with *Convolutional Neural Networks* (CNN) or *Recurrent Neural Networks* (RNN) (or combination CNN-RNN encoder/decoder, [Wang et al., 2016]), allow for higher accuracy in results given proper parameter tuning and a large enough dataset [Song, 2015]. DL implementations, on the other hand, require that a network is trained with large amounts of data and the model be upload directly to the drone (or a microcontroller interface). In general, even using only an RNN will outperform an HMM [Graves et al., 2006]. However, DL implementations require dedicated and stable network communication and real-time processing cannot handle latency when making important decisions.. Network communication can be expensive with respect to bandwidth and energy (increased power consumption). As the focus was to create, implement, and simulate a speech recognition model useful for evaluating user cognitive load, an offline method was employed (via PocketSphinx).

Text-to-Speech and *Speech-to-Text* are two methods of machine translation. Intuitively, these technologies are mappings to shift from one domain to the other. These are common in modern ASR technologies (like Alexa and Siri) for providing the user with a confirmation of audio commands.

3.2 User Interface Design

Once the drone can properly be given instructions via ASR, it is important that the drone is able to communicate back to the user in a meaningful manner. The main user-experience related criteria that are being pursued in this project is a "seamless, natural integration of a drone into a mixed human-robot team." This means that the drone needs to be able to communicate effectively (quickly, clearly, and without ambiguity). As such, there are very few approaches that would be sufficient in an environment that places a great deal of stress on the user. Haptic feedback (tactile vibrations) is likely not a good approach, as they can be easily misunderstood (and rely on the user being trained to interpret the message [Cho and Proctor, 2003]). A speech response from the drone may be problematic, as it requires that the user is primed for listening to the confirmation of the planned tasked [Dyson, 2008]. During stress, the cognitive ability of the user will degrade [Broadbent, 1958]. This means that long response might be forgotten, misunderstood, or not properly heard [Engle, 2002]. Any of these events would require that the user ask the drone to repeat the flight plan or resend instructions. These are issues that exist even in human-human interactions, so it would be highly advantageous if a robotic system could mitigate or alleviate these issues through design.

An alternative to tactile and audio feedback is visual feedback. This will improve the response time of the user and mitigate errors due to multimodal bottlenecks in information processing [Sommer et al., 2001]. One approach would be to put LEDs on

the drone that signal certain sequences of commands. Again, in this case, the operator needs to be trained to interpret these responses from the drone. Furthermore, once the drone is out of the line-of-sight, the drone will not be able to effectively communicate with visual commands. Therefore, a UI within a Mixed Reality device (i.e., HoloLens) can also be utilized. The UI can be designed in a mixed-reality format that displays, with written words or intuitive icons, the flight/task plan of the drone. It is hypothesized that this will mitigate the forgetfulness of the user due to the high cognitive load. The UI should be designed in a manner that will not obstruct the vision of the user. This allows the user to maintain situational awareness and analyze UI feedback without introducing additional risks. Regardless, both audio and visual solutions need to be approached as often times theory and reality do not match in real-world implementations. User behavior is not trivial and many environmental factors, previous experiences, and unknown variables lead to trends and tendencies in demonstrated behavior.

In general, humans are able to ”fuse” signals from multiple sensory sources with an overall decrease in cognitive load. Simultaneously fusing multiple channels for sensory inputs is known as *multimodal* integration [Kipp et al., 2005]. For this thesis, multimodal integration is a combination of audial and visual information in the form of confirmations.

3.3 Cognitive Load and Testing

Many different tests have been created to evaluate the cognitive performance of individuals during various conditions. *Situational Awareness* (SA) is limited by available resources in memory capacity and computation during decision-making [Endsley, 1995]. *Cognitive Load* is the concept that users have a finite amount of cognitive resources that are distributed between multiple channels and memory locations. When cognitive resources are limited or have been depleted, users begin to make errors in accuracy or decision making as a result. Users may fail to notice important details in their environment, have delays in reaction time, or experience diminishes in memory. Situational awareness can be approximated using the *Situation Awareness Global Assessment Technique* (SAGAT). Cognitive load (and SA) can be evaluated with two types of tests: qualitative and quantitative. A common qualitative test is the *NASA Task Load Index* (NASA-TXL) which allows users to access their *perceived* stress and workload levels while completing other tasks. Subjects report their perceived performance after tasks are given for problem-solving or split-attention tasks with a *Self-Reporting Test* (SRT) [Chandler and Sweller, 1991].

For qualitative experiments, attention and working memory are common metrics. The *Operation Span Task* (OSPAN) is used to measure the *Working Memory* (WM) capacity of users [Turner and Engle, 1989]. This is completed by giving the user an item to remember and then interjecting with random simple mathematics problems, after

which they are asked to recall the item. In the case of drone operations, a set of directions could be provided as the "item." The *Attentional Blink* test is used to measure the response time and accuracy of a user when two items are displayed for brief periods of time with a short (variable) delay between objects [Nieuwenstein et al., 2009]. NASA has employed the *Psychomotor Vigilance Task* to measure the cognitive performance of astronauts during space missions (NASA Extreme Environment Mission Operation NEEMO [Dinges, 2019a]) and aboard the *International Space Station* (ISS). This test measures the response time for the user to click a button after an image has been displayed on a screen (faster is better).

Chapter 4

Objectives

This Chapter discusses the objectives and research questions intended to be derived from evaluating a voice-controlled drone and user cognitive load in a virtual simulator. Furthermore, the general approach and process engineering are described.

4.1 Objectives

Each *Research Question* (RQ) was approached and completed with the mindset that this is not a final, shippable product, i.e., this is a prototype/mock-up. In reality, the foundation and framework need to be available in the form of a robust *proof-of-concept* that can be adaptable for specific needs of the stakeholders that follow (e.g., the SEK will likely want to change specific details prior to mass implementation, as well as future researchers within the DAI-Labor).

Each research question was broken down into work packages. Each implementation will be given clear objectives and criteria for evaluation. Some work packages had a slight overlap related to tuning, implementation, and iterative design. The end results are intended for the interoperability between distributed systems, applicable to many projects.

4.1.1 RQ1: Human-Robot Communication

RQ1: How should the drone send confirmations to the operator to minimize ambiguity, human errors, and temporal descrepancies?

After viewing the field demonstration and discussion with the SEK, it is evident controlling the drone via voice commands is preferred. The SEK operatives primarily utilize speech (through headsets) for communication (local and global). While gestures, positioning, and other non-verbal communication methods are employed (i.e., patting

and tapping) this is based on proximity (locality) and the situation. Furthermore, as local communication is based on quick confirmations, it does not appear to impact the overall goal. Thus, the drone will still have its core mission criteria that are independent of these non-verbal interactions.

Once the drone has received its orders, it should begin to complete its tasks. However, it is currently not clear how the drone will communicate with the operator that it has correctly interpreted the task. Consider the following task given to a drone: "go forward to an open doorway, enter the room, scan the area, proceed into any additional room on the right side." It is possible that the drone correctly begins its task and enters the first room but did not properly interpret the sequence afterward. If the drone is able to confirm the sequence with the operator, proper navigation can be ensured (at least on the command level). The operator should be able to properly confirm the sequence, make any necessary changes, and then send a "confirm" command. The drone should also be able to send confirmations once each individual command has been completed if desired (i.e., passing through a waypoint).

In designing *Natural User Interfaces* (NUIs), research has shown that gesture-based interfaces are less natural for communication than voice interfaces [Israel et al., 2009], [Norman, 2010]. *Tangilble User Interfaces* (TUIs) are meant to translate intuitive gestures in to appropriate physical output based on the behavior of the user, similar to "pinching" a screen on a smart device to make an object smaller [Ishii, 2008]. However, these are typically "learned" behaviors that become "intuitive" after a prolonged experience with similar UIs [Fitzmaurice et al., 2002]. Due to the high criticality of missions and adherence to safety, any additional ambiguity related to NUI/TUI would impact mission performance due to the technology level of these UIs. Audial and visual interfaces can decrease ambiguity if properly designed which is why they will be the emphasis. The first logical step of RQ1 was to determine which of these two channels, audial, visual, or a combination, were preferred prior to users prior to a finalized, real-world implementation. A user study was conducted evaluating preference, subjective load, and qualitative factors for different sensory channels and UIs.

4.1.2 RQ2: Operator Cognitive Load

RQ2: What are the negative effects (response time delay, accuracy, situational awareness) on the operator's cognitive load during operation?

Automatic Speech Recognition (ASR) models are not perfect. The middleware might not properly understand the commands of the user which greatly impacts the mission's objectives due to a loss of time for correction and operator frustration/distraction. A user study was required in which the user completes some simple tasks simultaneously as validating the drone's confirmation. The concept here is that a drone could mistakenly report incorrect information to the operator which could lead to mission

failure while compromising the operator. For quantitative evaluation, several tests from the cognitive science field have been employed to measure *Situational Awareness* (SA) capacity (while under pressure, is the operator able to properly acknowledge objects, hazards, enemies, and allies in the field?) and *Response Time* during events requiring a high cognitive load and split-attention. The results and analysis will infer the user's situational awareness and confirmation accuracy. The user will be asked to give commands to the drone and listen for the confirmation while completing other tasks. Quantitative results were calculated and compared while receiving audial confirmations (Voice-Only), visual confirmation (Visual-Only), and a multimodal implementation (both Voice and Visual confirmations).

4.2 Approach

Based on the literature and considerations stated in the previous sections, the goal was to enable a drone to accept user commands, accurately follow the commands, and send a confirmation to the user (for review). Prior to selecting the deliverable form for the drone UI, qualitative (user preferences, perceived cognitive load) and quantitative (reaction times, accuracy of memory) were needed to be derived from user study results. A custom library/dictionary was created for ASR and validated in ROS. ROS then communicates these commands to the drone (SST) and the drone's behavior follows correctly. A UI for visual representations of the confirmation (text-based) was created in Unity for the HTC Vive Pro but was also designed to work on any Virtual-Reality or Mixed-Reality platform.

4.3 Process Engineering

RQ1 and RQ2 were approached simultaneously. The first aspect is to implement a library/dictionary for *Automated Speech Recognition* (ASR) that can allow for commands to be sent to a drone (either simulated or actually sent to a drone via ROS). Next, a Speech-to-Text (STT) library was designed for machine translation into ROS for the real-time validation and verification of commands. In the physical implementation, the STT commands would be sent directly to the drone for demonstrating the simulation results on a live drone.

For cognitive evaluation alone, it is might have been better to focus on a Text-to-Speech (TTS) model. Utilizing a terminal, TTS commands could be manually typed and sent to the user's device to evaluate their hit/miss rate for confirming the drone's response. This would mitigate any errors that would be caused by utilizing ASR (control variable). TTS commands would be sent to the user's headset during a cognitive test. For an MR solution, STT would allow for commands to be translated and sent to the

device's UI in the form of words or icons that represent words (i.e., a right arrow for "right"). However, this approach would not have provided insights into the response time of the speech recognizer since all communication would have taken place within the VR simulator.

The NASA-TXL was used for measuring perceived cognitive load while asking users to make a preference for receiving audial or visual confirmations. The SAGAT was implemented to analyze the situational awareness of users. For quantitative tasks, it was intended to use the Inquisit Laboratory program with the PVT test to get immediate (and accurate) results [Millisecond, 2019]. However, Unity was unable to interface directly with Inquisit Laboratory so a simulated test was created within the virtual environment (see Sec. 7.3.3). Both tests are regularly used by NASA to evaluate astronauts during space missions and after *Extravehicular Activities* (EVAs). The original plan was to evaluate three use cases for the experiment: one without any drone commands (baseline), one with audial drone confirmations, and one with visual drone confirmations. The goal is to have the users complete the tasks of the PVT (button click) while measuring their response times and accuracy and documenting their accuracy for confirming correct drone command sequences (hit). However, the final implementation consisted of receiving confirmations from the drone based on visual-only (UI), voice-only, and a multimodal implementation (visual and voice confirmations combined). The decision to add a multimodal implementation came from considering that humans can receive information on multiple sensory channels simultaneously and multimodal communications are more robust than single channel signals [Krakauer et al., 2010]. Furthermore, using multimodal implementations has shown to reduce user cognitive load and improve working memory for split-attention tasks [Mousavi et al., 1995].

Chapter 5

Experiment Design

This Chapter describes that design of the ASR and simulation environment. The primary theme of designing the experiments and environments has been *high-fidelity* model creation. Discrepancies between real-world implementation and virtual simulation arise whenever user testing in virtual environments is not properly extrapolated to real-world situations. Therefore, a high level of realism and *immersion* (a sense of "being there", [Witmer and Slater, 1999]) was pursued. Another reason for pursuing highly realistic (high-quality) environments is that users tend to make decisions in simulation that more closely represents actual responses in the real-world [Fox et al., 2009]. Similarly, highly-detailed environments are not required for immersion but they more accurately reflect the reactive behavior of users [Haans and IJsselsteijn, 2012].

Unity was used to create high-fidelity user scenarios. Unity was chosen due to its ability to create highly realistic environments and is commonly used in the creation of high-end virtual reality simulators and video games. The *Scenes* (scenarios) for conducting user testing a voice controlled drone could have been created entirely in Unity with integration to the *Google Cloud Platform* (GCP). However, to create a realistic prototype, ROS was used along with Unity to reflect communication delays and system behavior for sending commands to an actual robot. It was posited that the results of a Unity-only implementation would been vastly different than an implementation utilzing communication to external devices and would provide merely negligible insights for real applications.

The following sections include information regarding the speech recognizer design including the corpus, language model, keyword spotting, and grammar. Furthermore, ROS integration and virtual environment design are discussed.

5.1 Automatic Speech Recognition

For a drone that can respond to voice commands, ASR is the most crucial application in the project. ASR can be approached with "offline" HMMs or "online" with Cloud-based, DL solutions. SEK operatives usually work in an offline mode in which connectivity to the internet/Cloud is highly limited or unavailable. This decreases security risks (frequency scanning) and potential failures due to network instability. Offline methods are commonly employed by high-security and high-risk industries (military, space, etc.). This limits the methods in which an ASR can be constructed as most ASR technologies utilize the internet/Cloud (e.g., Alexa). There are primarily two options that can be pursued including using an offline, HMM-based speech recognizer (like Sphinx) or communication with a mobile embedded platform to act as a mobile GCS. If the vocabulary/grammar is small enough then continuous-listening Sphinx-based approaches are optimal. For large vocabularies that require a large overhead of resources (processing), embedded platforms are able to remotely run trained models capable of properly classifying the vocabulary. However, embedded platforms require additional energy, add weight and responsibilty to the operative (they must carry and protect the device), and add a communication channel between the operative and the drone.

5.1.1 PocketSphinx

The dictionary and key phrases for commands can be relatively small. Furthermore, utilizing an embedded platform as a mobile *Ground Control Station* (GCS) requires additional load on the operator (recall up to 50kg of base tactical gear) and the embedded platform needs to be protected. These factors lead to PocketSphinx being selected for implementation. Furthermore, PocketSphinx is capable of real-time classification and prediction in "continuous mode". This means that the recognizer is constantly running. A notable benefit that emerges from using continuous mode to overtrain a model is that the accuracy of a single user is optimized. For the highest possible accuracy, many hours of recordings can be saved from a single user. However, this effectively leads to overfitting for a single individual and leads to poor results for other operators. This is not necessarily an issue if one individual is responsible for all commands sent to the drone, similar to how a *Capsule Communication Officer* (CAPCOM) is the only authorized communicator for pilots and astronauts. However, classical scenarios consider the case in which the CAPCOM is out of harm's way (not in a combat scenario). In the event the drone operator is unable to send commands to a drone, it would be necessary for another operator to be able to be in control. Similarly, operatives might be in difficult locations and be able to directly observe the drone while having a desire to send commands based on the current scenario. Continuous mode provides robustness when considering multiple operators.

5.1.1.1 Corpus & Language Model

The corpus (also known as a vocabulary) was designed based on potential commands that could be given to the drone relative to desired movements with an emphasis on spatial information. Temporal commands were also created though they appear to be less useful. For simplicity, the traverse axis (z-axis) was not controllable by the operator via voice commands for users (though it was still encoded for future use by operators). Similarly, rotational movements can be controlled via adjusting the yaw but the roll and pitch are not adjustable (or appropriate for most UAVs). Movements consist of two primary types: linear and angular. Linear movements include the drone moving *forward*, *backward*, *left*, and *right* (relative to the operator). Similarly, angular movements are initialized via the keyword *rotate* followed by *left* or *right*. Next, available integer values are dependent on the type of movement, i.e., *five* is relative to *forward* whereas *ninety* is related to *rotate right*. Finally, measurement values were added to reduce ambiguity for the drone and the user including *meters*, *degrees*, and *seconds*.

Commands were also created for system initialization (*takeoff*), *shutdown*, *landing*, *pivot* (rotate 180 degrees), and *activate* (activation word). Discussion regarding action activation is in Sec. 6.1. The sentence structure is discussed more in Sec. 5.1.1.2 . Example sentences include *Activate Forward Five Meters* and *Activate Rotate Right Ninety Degrees*.

Only acceptable sentences were implemented in the *Language Model* (LM). An LM is a probability distribution over sequences of words. Sphinx Knowledge Base Tool (Version 3) was used to build a set of lexical and language modeling files for the PocketSphinx decoder. The ARPA format was utilized with a fixed discount mass of 0.5. The backoffs were computed using the ratio method and the model based on a corpus of 79 sentences and 31 words. For optimization, all models were converted to .bin to improve decoder initialization time and memory allocation, both of which are critical for real-time embedded systems and critical mission performance.

5.1.1.2 Keyword Spotting & Grammar

The first solution was to replace words that had a similar phonetic structure with a synonymical (and familiar) replacement word (i.e., instead of “turn” the word ”rotate” was utilized). The second improvement was to increase the number of syllables for words which decreases the overall ambiguity (i.e., instead of “no” the word “negative” was utilized). Third, some words from the NATO phonetic alphabet were implemented (i.e., instead of “five” the phonetically similar “fife” was used which also lead to improved results from non-primary English speakers). The fourth improvement was to use *Keyword Spotting* (KWS). KWS allows for the model to emphasize discrete keywords (or in this case, keyphrases) as acceptable inputs. Essentially, the model is seeking out appropriate keyphrases during continuous operation. The fifth improvement was to utilize

grammar within the model. Typically, ASR grammars are similar to spoken language, i.e., ⟨ subject ⟩ - ⟨ particle ⟩ - ⟨ verb ⟩ - ⟨ particle ⟩... etc. However, grammar was utilized to explicitly enforce only a particular pattern of words that could be accepted.

Effectively, this pattern is in the form of:
⟨ activation ⟩ - ⟨ direction ⟩ - ⟨ integer ⟩ - ⟨ measurement ⟩

An example of utilizing grammar is:
⟨ rotate ⟩ = rotate (right | left) ⟨ number ⟩ + (degrees);

Data was collected from two users over the course of two hours each. Initially, the speech commands were poorly recognized, as was the "confirmation" word. Combining these different methods greatly increase the accuracy of the speech recognizer from approximately 50% to 98% and from 30% to 95% for command recognition and confirmation, respectively, for subsequent testing sessions. This data was calculated by recording the total number of correct responses from the PocketSphinx terminal (i.e., 19/20 correct confirmations within a session yields 95% accuracy). With proper training for pronunciation and timing, the results are 100% accurate (for the native speaking designer).

5.2 ROS Integration

This section discusses the different phases of integration in the *Robot Operating System* (ROS).

5.2.1 System Architecture

This section contains the general system architecture for the VR implementation. The architecture can be seen in Fig. 5.1. Currently, both Unity SteamVR (for rendering the HTC Vive) works poorly in Linux-based systems. C# is a native Windows language that is used for a majority of the scripting in Unity and is directly integrated ("Unity-integrated"). Similarly, ROS development has been primarily in Linux. While a "ROS for Windows" development kit has been developed as a part of the Azure/Windows IoT Platform, many of the ROS distributions (i.e., Kinetic) have limited functionality or package support.

ROSBridge is a communication package for accessing and communicating with ROS on machines that do not have access to ROS innately (i.e., Windows OS) [Crick et al., 2017]. ROSBridge utilizes the *WebSocket* communication protocol and was utilized to send/receive commands (small data) related to the drone. *SocioIntelliGenesis (uni-)Verse* (SIGVerse) was developed by the *SocioIntelliGenesis* group at

National Institute of Informatics to communicate large data generated by VR applications in Unity to ROS environments [Inamura et al., 2011]. SIGVerse is based on the previously widely-used *Open Architecture Humanoid Robotics Platform* (OpenHRP) from the Advanced Institute of Science and Technology (AIST) [Hirukawa et al., 2003]. Both SIGVerse and OpenHRP are integrated software platforms for robot simulations and software developments that utilize dynamic simulations via control programs and with respect to the original robot models. These platforms incorporate the user directly.

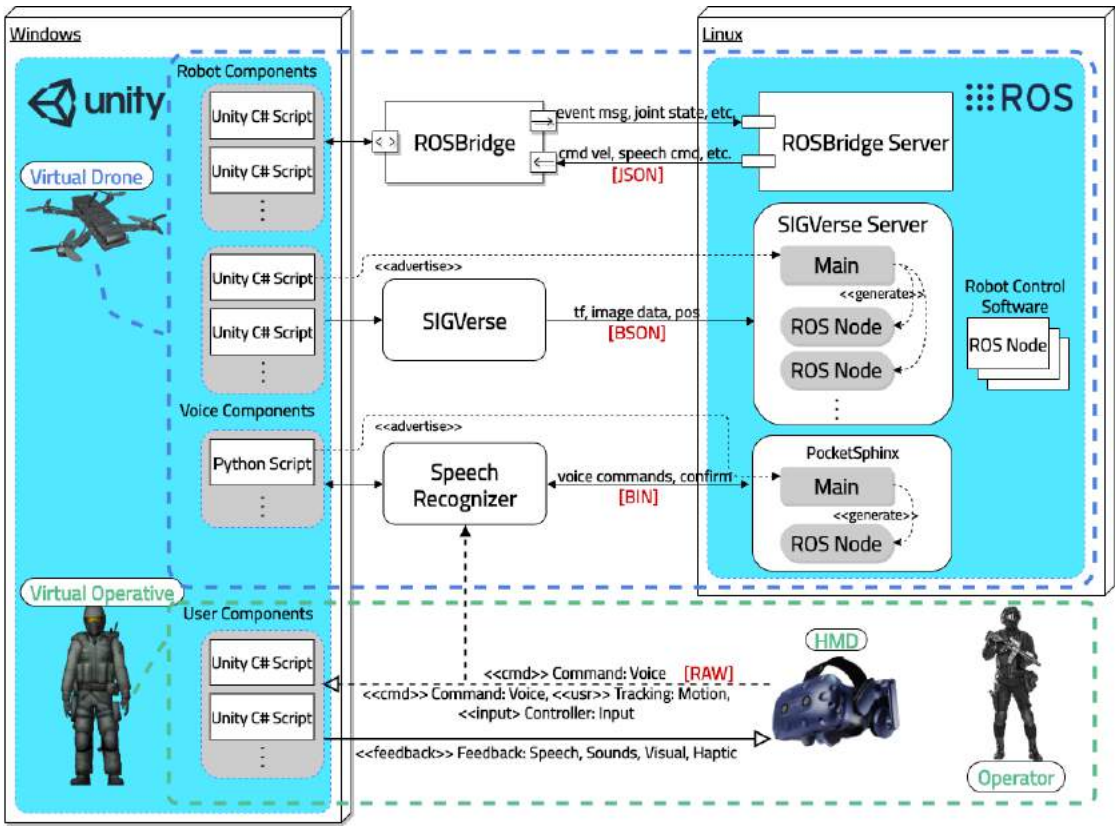


Figure 5.1: System Architecture.

SIGVerse can use the BSON format to send and receive binary data. Effectively, small data related to the robot's joint states, commanded velocity, and speech commands were sent over ROSBridge directly, while larger data (like image data containing the depth or RGB features) are sent over SIGVerse. SIGVerse was used for the simulated RGB and depth cameras (Intel RealSense) via the ROS RealSense package. PocketSphinx was configured to work with BIN files. The approach was to make an efficient communication structure that has a small memory footprint. Drones have limited processing power and memory storage, and thus, optimizations are crucial to mission performance. Without a proper system architecture, either the drone will not meet real-time

deadlines or it will allocate too many of its resources to tasks that could be completed by the host computer (an emulated GCS).

ROSBridge and SIGVerse have dedicated servers initialized on ROS with their own launch files. The HMD sends raw voice data directly to the speech recognizer and to the Unity environment. Unity sends the motion tracking, controller inputs, and voice confirmation back to the HMD. Unity C# scripts are used to control the laser pointer, teleportation, and other user components, as well as the robot components for movement. The Python script is responsible for translating the voice commands into a format that ROS and Unity can use with PocketSphinx.

5.3 Virtual Reality Environment

This section describes the design of the virtual reality environment and integration with hardware for real-world implementation.

5.3.1 Environment Design

SEK operatives often conduct missions within narrow, multi-level indoor areas. An open-source environment Asset package was utilized [Strong, 2019]. The lobby level of a building was constructed to match real-world scenarios. There is a hallway along with a stairwell and an additional room (not currently accessible by the user). Dynamic lighting and shadow effects were added for increased realism and immersion. The environment (elevators, doors) can be interacted with as well. Furthermore, the "physical" environment (walls) were constructed to include physical properties (box colliders) such that they are not simply silhouettes (meaning the drone can collide with the environment). Each item was scaled to match the height of the user within the virtual environment. Environment images can be seen in the Appendix (Sec. 9.3). Fig. 9.3, 9.4, and 9.5 display examples of the created environment. The initial user scene can be seen in Fig. 9.6 and the lobby interior can be seen in Fig. 9.7.

5.3.2 Operative Design

In an attempt to create a highly immersive environment meeting the criteria for a high-fidelity model, operatives were constructed with realistic textures, movements (animations), weaponry, and attire. The operatives have been scaled to be approximately 1.9m in height to match the height of the average SEK Operative. Tactical poses (crouching, idle, etc.) were modeled including gestures that imply the operative is actively observing the environment. Weapons like the HK MP5A3 (*Maschinenpistole 5*) were also modeled based on standard-issue SEK hardware. Materials for operatives were acquired via SketchFab were converted to the AutoDesk .fbx format and edited in Blender

to match the attire, pose, and joint behavior of SEK operatives. Open-source combat animations were acquired from Adobe Mixamo and rigged to the character models in Blender. Examples of the characters can be seen in Fig. 9.15 in Sec. 9.3.

5.3.3 Drone Design

A high-fidelity custom drone was also constructed as the SEK (and other stakeholders) will be using custom drones in implementation (Fig. 9.16 in Sec. 9.3 shows various views of the custom drone). Custom sound effects were added based on the location of the drone from the user (motor loudness changes proportionally) and based on the applied acceleration (motor speed changes proportionally). For additional realism, the drone makes "sparks" whenever it interacts with the environment as well as a "destruction" parameter which makes it no longer operable (though this feature is was not used in user testing as it adds uncontrolled variables and instantaneous reset is unrealistic). An example of collision detection and response can be seen in Fig. 9.8 in Sec. 9.3.

5.3.4 Head-Mounted Display Design

Designing *Heads-Up Displays* (HUD) is not a trivial task. HUDs need to readily provide important information without being distracting. *Head-Mounted Displays* (HMD), like virtual reality headsets, are a subset of HUDs that have additional criteria for practical usage. These criteria include the *Interpupillary distance* (IPD), binocular overlap, distance focusing, and resolution. The IPD varies between users but is generally adjustable in high-end VR headsets. Distance focusing and resolution depend on *Depth Cues* and the *Field of View* (FOV) of objects. Poorly designed UIs increase the rate at which users acquire a nauseogenic ailment known as *virtual reality sickness*. VR sickness is due to mismatches between the vestibular system and the visual data [Shah, 2018]. UI elements that have a movement delay (lag) or "choppy" movements rapidly cause discomfort. All UI elements were designed with the specifications and behavior of the HTC Vive Pro headset and Unity game engine.

5.3.4.1 UI Elements Color Selection

UI elements must be designed utilizing colors that are easily noticeable and distinguishable. Colors that are difficult to perceive go unnoticed. Schrödinger's "ideal colors" are colors that have a spectral reflectance of values at exactly 0 or 1 [Schrödinger, 1920]. Colors change based on a gradient making very few color candidates for the criteria. Furthermore, the perception of these depends on the environmental conditions in which they are viewed, with a wavelength of $\lambda = 555$ (often called "safety green", the color of cyclist jackets) as the most visible during the daytime.

Unfortunately, simply making colors that meet the luminosity criteria is not sufficient. Users have gained an "intuition" for the meaning of various colors based on previous exposure (e.g., "green" means "go", "red" means "stop"). Another aspect in considering the user experience for viewing UIs considers "aesthetics," as some colors are considered unappealing or even repulsive. This point is more critical than intuition alone can suggest, as every major company uses colors to influence purchases; from colors that envoke desires to eat at a fast-food restaurant to the mandatory usage of repulsive hues for tobacco products in some countries like Australia [Jalil et al., 2012].

With these trade-offs in mind, the colors for the UI elements (clickable buttons and the waypoint goals) were designed to be *green* for the immediate task, *yellow* for the next task, and *red* for the task has been completed. Additionally, the UI button was also changed to *black* to prevent the user from interacting with a disabled button (see Sec. 7.3.3).

5.3.4.2 UI Elements Positioning

Three main factors were considered when positioning the UI elements. First, having a smaller FOV can decrease VR sickness [Fernandes and Feiner, 2016]. Second, UI elements need to track the HMD's movement in real-time to prevent VR sickness [Buker et al., 2012]. Third, elements should not occlude much of the user's view, as the user must be able to see threats, allies, and objects in the environment with the minimum amount of visual information transmitted to the user [van der Horst, 2004].

The FOV was limited to 110° to prevent VR sickness (up to 134.48° possible with the HTC Vive Pro) [Mizuchi and Inamura, 2018]. In user beta testing, two users were able to wear the HMD for upwards of an hour, totally more than six hours each, without experiencing VR sickness.

5.3.5 Hardware Integration

The Unity environment was build as a VR game on an HTC Vive Pro systems via SteamVR and the Windows Mixed Reality plugin. Two Vive controllers are available for interacting with the environment (firing the controller). Two Base Station 2.0 modules were used to ensure proper tracking and rendering of the environment for the user. For interactable objects, a "laser pointer" was created with the ability to "fire" when hitting a trigger button. The bottom trigger of the Vive Controller was chosen for this task as it is the most intuitive for "shooting." A teleportation environment was given to the user to allow free movement throughout each scene for an added sense of realism. Initially, only teleportation beckons were implemented as the general Unity community (forum board) does not recommend allowing users too much freedom. Teleportation beckons allow for users to travel only to pre-selected spaces with confined mobility. The teleportation environment spans the entire map.

5.4 Physical Environment

The physical environment for user testing was the RoboCup Laboratory at TU Berlin. The room dimensions are approximately 4x6m^2 and the userspace configured for the HTC Vive is 2x2m^2.

Chapter 6

Proto-Typing

This Chapter includes additional design, testing, simulation, and integration aspects taken over the course of the thesis. These components were useful and used to influence the final design of the system and have be maintained for documentation purposes. This includes speech command testing, Gazebo and RViz simulation, AR.Drone integration, and some augmented/mixed reality aspect.

6.1 User Experience

On the user experience side, a few improvements needed to be made as well. For example, the default voice from the pyttsx (Python Text-To-Speech) library was not optimally intelligible for all users. A small survey and test were conducted using the keyphrases designed in Sec. 5.1.1.1. A less robotic, female voice with British pronunciation (commonly learned throughout the EU and East Asia) was considered to be the most pleasant and intelligible.

An activation phrase was also added to prevent the speech recognizer (and drone) from responding to unintended commands. Operators might discuss (via headset) general tactics and the drone might prematurely engage in operations (this happened empirically during testing). This can lead to mission failure and endanger the user. The current activation word is ”activate” and must be used prior to any string of commands. This ensures proper usage. Furthermore, utilizing a small dictionary (less than 200 words) in continuous mode results in an issue in the calculated confidence values for the predictor. Essentially, “recognized word”, even if incorrect, always return a confidence of one (100%). By enforcing that confidence only applies to complete sentences paired with the activation function, the actual confidence value is higher (less ambiguity). Sphinx-based models can only accurately predict the confidence values for trained models and for systems the do not use continuous mode (i.e., pre-recorded .wav files).

6.2 Speech Command Testing

Upon completion of the PocketSphinx implementation, a ROS Node was created to publish the results of the decoder (Recognizer). Additionally, a subscriber for the Recognizer node was created to recognize and confirm specific keyphrases and publishes standard ROS messages and odometry/trajectory (Command). A launch file with the default TurtleBot was used to test and evaluate speech commands in a low-fidelity environment (low-poly Gazebo [Koenig and Howard, 2004]). This testing allowed for confirmation of the coordinate system, timing profiles (delay/jitter), and analysis of the coordinate transformations.

6.3 Gazebo & RViz Simulation

Beginning with a low-resource environment for Gazebo, testing was conducted to ensure the proper behavior of the drones. Initially, acquiring proper operation with the TurtleBot allowed for an extension for drones. TurtleBot is a common mobile simulated robot used in ROS, Gazebo, and Morse for designing navigation and movement profiles for simulated robots. Afterward, two drones were created: an AR.Drone 2.0 and a "Custom" drone. Parrot has an official SDK for AR. Drones in ROS (extending upon the core Hector drone architecture). Effectively, both use the AR.Drone Autonomy library for command translation (ROStopics). Prior to integrating the Speech Recognizer nodes, a Teleoperation (teleop) node was created for usage with the keyboard and a PlayStation 3 controller via the ROS Joystick package and a simple Qt GUI [Patil et al., 2017]. Furthermore, voice commands that were implemented for the simulated AR.Drone launch file was also used on a physical AR.Drone 2.0 via an extension of the AR.Drone Autonomy package, known as the TUM Simulator [Lugo and Zeil, 2013]. This setup was later needed when integrating the Speech Recognizer in Unity via ROSBridge and SIGVerse for debugging (as it was not clear if publishers/subscribers are properly sending commands to Unity). The validation environment's system architecture varies from the system architecture shown above and can be seen in Appendix D.

Upon integration with Unity via ROSBridge and SIGVerse, RViz was used to map the environment correctly. Mapping was conducted via a Simultaneous Mapping and Localization (SLAM) approach by adding a virtual LiDAR scanner to the AR.Drone model and simultaneously launching a teleop node while the drone was scanning in Unity. The TUM Simulator also supports a *Parallel Tracking and Mapping* algorithm for multi-agent drone systems, making it suitable for the InLaSeD project [Klein and Murray, 2007]. The results of mapping were used in designing the environmental space and test for collisions (i.e., walls and objects) for usage in the HTC Vive, as users would be able to both teleport to a location (typically implying static positions) and allow them to move within a set of boundaries without being able to warp through

walls.

6.4 AR.Drone 2.0 Integration

Commands were sent to the drone based on the transform messages from the TUM Simulator that are compatible with the physical operation of the drone as well as a simulated AR.Drone 2.0. This allowed for additional command testing that is impossible with the TurtleBot (traversal commands like */takeoff*). It was discovered that linear movements do not translate 1:1 as 1m of movement sent by the Twist messages was approximately .5m on the simulated drone and even less on the physical drone. Unfortunately, it was discovered that this discrepancy was non-linear and thus, an offset did not suffice for adjusting the traversed distance withing the virtual environment. For the physical drone, some characteristics of the environment are responsible as the lift/drag created by the drone influence the distance that it moves. Since there is no control system or filtering process (i.e., a PID or Kalman filter), the linear movements sent the drone are static. Effectively, the drone's movement is highly susceptible to drift depending on the distance from the floor, operator, or object, as well as conditions like open windows or doors. In future InLaSeD projects, special handling of the drone's precise position must be taken as some sensors (like GPS) will not be effective in updating the pose of the drone as (i.e., GPS is ineffective in the indoor environments). An example of a physical AR.Drone 2.0 being used with the HTC Vive can be seen in Fig. 6.1. Using a live drone during testing can be dangerous and lead to personal injury, especially when the view of the user is obstructed by an HMD. Therefore, a live drone was not used during the user testing phase. The system architecture for the prototyping/physical design can be found in Fig. 9.2 in Sec. 9.3.

6.5 Augmented & Mixed Reality

The front camera of the HTC Vive was also fed into the Unity environment to protect the user, as well as provide an augmented reality environment. The boundary was necessary for one user as the immersive environment would captivate them and they "believed they were really in the environment", causing them to walk at a faster-than-average speed. This is dangerous for both the operator and the hardware within the real environment. However, allowing the user to be able to walk around as freely as possible greatly improves the immersion and realism of the scenario and results in more authentic user behavior [Jacob et al., 2008]. An example of the mixed-reality layout can be seen in Fig. 6.2. Note that the boundaries only appear if the user is leaving the designed userspace (Sec. 5.4).

Figure 6.1: Image of Drone Integration with VR.

Figure 6.2: Image of the Virtual Environment with Boundaries for the Real Environment.

Chapter 7

User Testing

This section details the aspects of user testing including an initial survey, goals, methods, scenarios (*Scenes*), and *missions*. The initial survey was conducted in order to find the best approach to designing an optimal UI for users. Scenes are the VR simulation scenarios containing different UIs for testing modalities. The missions validate the criteria for evaluating the UI with quantitative and qualitative data. The testing methods are meant to measure quantitative user performance metrics including response time, accuracy, and completion speed. Qualitative results focus on the preference of the user with respect to the type/number of modalities. The testing methods for evaluating user performance and cognitive load include the NASA-TLX, an augmented PVT (A-PVT), and the SAGAT. The goal of user testing is to find a model the balances between user preference (qualitative) and user performance (quantitative).

7.1 Initial Survey

Prior to preparing the tests described in subsequent sections, nine users from the Mensch-Maschine-Systeme (MMS, Human-Machine Systems) Group at the Technische Universität Berlin were surveyed regarding their initial thoughts on the sufficient and necessary sensory channels for interactive interfaces. They were first asked "which sensory channels are sufficient/necessary for effective interactive interfaces?" including visual, auditory, haptic, and multimodal channels in the general context of human-machine systems (i.e., including autonomous vehicles). The first survey yielded the following:

After a video demonstration of voice recognition systems for human-robot interactions (chatbots, HUDs, and GUIs for autonomous vehicles), the question was asked again. The second results show a tendency to favor multimodal integrations over single sensory channels.

However, during testing, users demonstrated a preference for receiving voice con-

Table 7.1: Initial Survey on Sufficient and Necessary Sensory Channels taken in the MMS Department at TU Berlin.

Channel/Participant	P01	P02	P03	P04	P05	P06	P07	P08	P09
Sound	*	*		*			*		*
Visual									
Haptic					*				
Multimodal			*			*		*	

Table 7.2: Follow-up Survey on Sufficient and Necessary Sensory Channels taken in the MMS Department at TU Berlin.

Channel/Participant	P01	P02	P03	P04	P05	P06	P07	P08	P09
Sound	*								*
Visual									
Haptic									
Multimodal		*	*	*	*	*	*	*	

firmations while considering the visual UI as supplemental. In all cases, users reported a dislike for receiving only visual confirmations. Multimodal tests demonstrated users would view the voice confirmation with more authority than the visual confirmation (See Sec. 8.2). Based on the results of the survey and a discussion with researchers in the MMS department, it was decided to add a multimodal implementation for testing and scenario creation.

7.2 Missions

This section details the requirements for missions. Missions were created to give users tasks within a high-fidelity environment. Defining the requirements and goals of the missions allowed for properly designing Scenes. All missions are conducted in parallel with the two primary missions having equal priority and importance (reaching the goal and deactivating the target button). Users were instructed to focus on achieving these goals. The sub-mission was an additional component for testing the situational awareness and working memory of the users.

7.2.1 Mission: Reach the Goal via Voice Commands

Controlling the drone via voice commands by inexperienced users is challenging. In order to evaluate the effectiveness of a voice-operated drone combined with user cognitive

load, a simple navigation game was created. The navigation game provides guidance and direction for deciding appropriate voice commands for meeting objectives. Three *waypoints* were created in which the drone must navigate through in sequential order for completion. The direction that the drone enters the waypoint does not matter though there is an optimal path to pass through all waypoints efficiently.

The first/next immediate waypoint is green in color. The waypoint following the "current waypoint" is initially yellow but changes to green when it is the current goal for the drone (i.e. the previous waypoint has been completed). After passing through a waypoint, the waypoint will turn red to demonstrate that it is disabled. The user must utilize voice commands to send the drone through all checkpoints to complete the game and end the scene.

7.2.2 Mission: Quickly Deactivate the "Target"

"Targets" are buttons built into the HMD that are initially green and become red (and then black) upon activation. The button is a UI element with text that states "Shoot on Green." Activation is caused by a laser pointer highlighting the button and the user "shooting" the interactable button with the Vive controller's trigger. There is a random timer between five and twenty seconds before the button can be reactivated. A random timer was created such that users cannot predict when the button will trigger (based on the PVT, see Sec. 7.3.3). The response time of deactivating the button was recorded and used for the augmented PVT. Users can subjectively report how well they think they performed when attempting the mission for qualitative analysis of both the NASA-TLX and PVT.

7.2.3 Sub-Mission: Observe the Environment

In addition to the two primary missions, users were asked to observe the environment during the test and attempt to remember various objects that they saw with as much detail as possible. These details include the type and location of a particular object. These objects include the number and positions of other operatives, the locations of the waypoints, type and location of environmental effects, and any "objects of interest" that were notable. The objects of interest were *robots* that were added to the scene simply to see if the user noticed out-of-place objects while having other priorities. Users were specifically told that observing the environment was secondary to the two previously mentioned missions. The motivation behind this sub-mission was to analyze the situational awareness and working memory of the user under a high cognitive load in complex scenes.

7.3 Testing Methods

The section contains various user testing methods. The NASA-TLX was used to gain insight into the subjective, self-reported evaluation of their personal performances. The SAGAT was employed to evaluate the user's situational awareness under high cognitive load while also hoping to gain insight into the impacts on working memory. The PVT and SRT were used to evaluate the response time of users for a task to acquire additional information regarding the impacts cognitive load throughout the missions.

7.3.1 NASA - Task Load Index

The NASA-TLX was used to evaluate user load for each Scene. The NASA-TLX consists of six questions:

- How mentally demanding was the task?
- How physically demanding was the task?
- How hurried or rushed was the pace of the task?
- How successful were you in accomplishing what you were asked to do?
- How hard did you have to work in order to accomplish your level of performance?
- How insecure, discouraged, irritated, stressed, and annoyed were you?

Users respond subjectively on a scale from 1 to 10 with 1 being *very low* and 10 being *very high*. Other than the self-reported "successfulness," it is desired to have users respond with lower values. Lower values for these metrics imply users consciously experienced a lower cognitive load during the task.

7.3.2 Situation Awareness Global Assessment Technique

The Situation Awareness Global Assessment Technique (SAGAT) was utilized to assess user Situational Awareness (SA) throughout the testing. The concept is to analyze whether users had varying SA depending on the sensory channel receiving the notification for confirmation. The following questions are some examples that were asked during the initial testing:

- How many operatives were in the scene?
- How many targets did you shoot?
- How many shots did you fire (with a range)?

However, these questions did not yield meaningful information and were improved to be more specific. For example, users were unaware of how many "shots" they made with the laser pointer or even how many button activations they attempted. Therefore, the questions were improved based on environmental effects. "Objects of Interest" were added to the virtual environment including environmental effects (fire, water, and electricity) that replicate structural damage that can occur during missions (e.g., explosions, broken water mains, and live wires. Images of the environmental effects can be found in Sec. 9.3 "Out of Place" objects were added to each scene as well. These objects were robots of a different type, size, location, and color. Each Object of Interest varied between different versions of the test but maintained a fixed location to prevent variation that would lead to different results.

- How many operatives were in the scene?
- Where were the waypoints (coordinates)?
- What environmental effects (fire, water, etc.) were present?
- Where were the environmental effects (coordinates)?
- What and where was the "Out of Place" object? (object, coordinates)

7.3.3 Augmented Psychomotor Vigilance Task

The Psychomotor Vigilance Task (PVT) was simulated by having the user shoot targets in between sending/confirming voice commands. Since the PVT is usually as a click test after a mission, the "augmented" aspect here is that the PVT is conducted during the mission to analyze the *Response Time* of the users while currently being impacted by a high cognitive load. "The PVT Self Test has wide application to any group that must operate remotely at high levels of alertness, such as first responders, Homeland Security personnel, flight crews, special military operations, police, and firefighters [Dinges, 2019b]." This makes the test ideal in evaluating subjects for high cognitive load tasks in safety-critical environments. The response time of the user is measured and compared between various sensory channels based on the scenes. Furthermore, when cognitive load is at the highest due to the drone misinterpreting the user's commands or changes in the immediate environment, users would occasionally neglect to activate the button.

7.4 Scenes

This section details the layout of the scenarios (*Scenes*) based on the needs of acheiving proper data for the previously mentioned tests. Three types of scenes were created

for simulation including *voice-only confirmations*, *visual-only confirmations*, and *multimodal confirmations*. A demonstration scene was also created so that users could learn how to navigate the environment. Users spent between five and ten minutes within the demonstration environment to learn the basic controls of the drone, movements and actions for the user, and general layout. The demonstration environment did not contain any waypoints, operatives, environmental elements, or objects of interest. The HMD UI elements were designed to be highly visible, simplistic (non-distracting), and with spatially small elements.

Throughout user testing, users were not reminded of tasks such as clicking the button or observing their environment. "The most important problem associated with this technique is that halting the simulation and prompting the [pilot] for information concerning particular aspects of the Situation is likely to disturb the very phenomena the investigator wishes to observe" [Sarter and Woods, 1995]. While this led to two results in which the users did not complete the PVT tasks, it was inherently better for observing user behavior within the simulator.

7.4.1 Simulator: Voice-Only Confirmation

The Voice-Only simulator does not have any visual UI confirmations. Fig. 7.1 shows an image of the user's view with a marked environmental effect (electricity), operative, and waypoint example. Fig. 7.2 shows the placement of the various objects for the SAGAT. The placement was designed to require a minimum of five rotations and to have a goal completion time of 210 seconds.

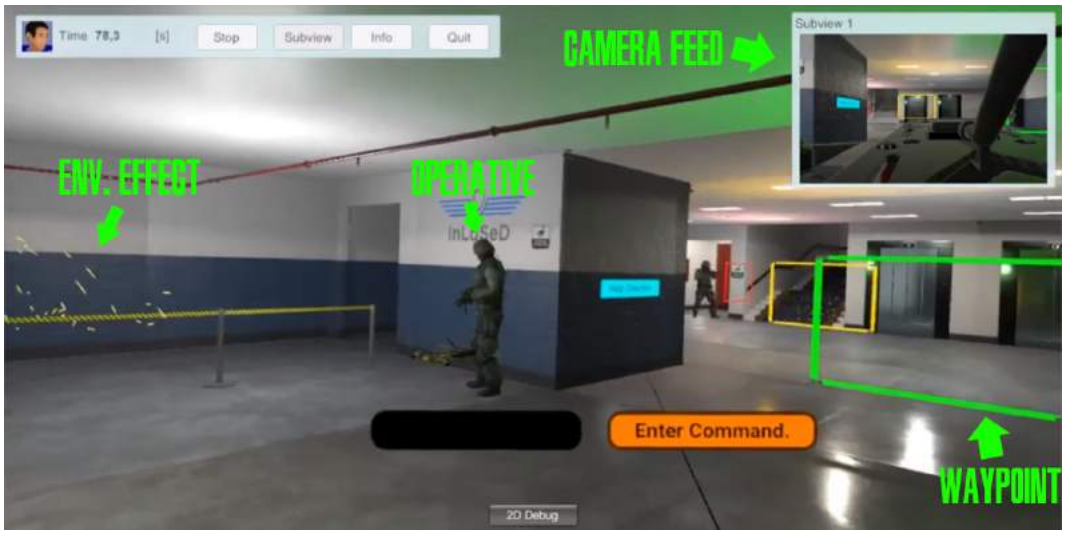


Figure 7.1: Example of User View with various objects.

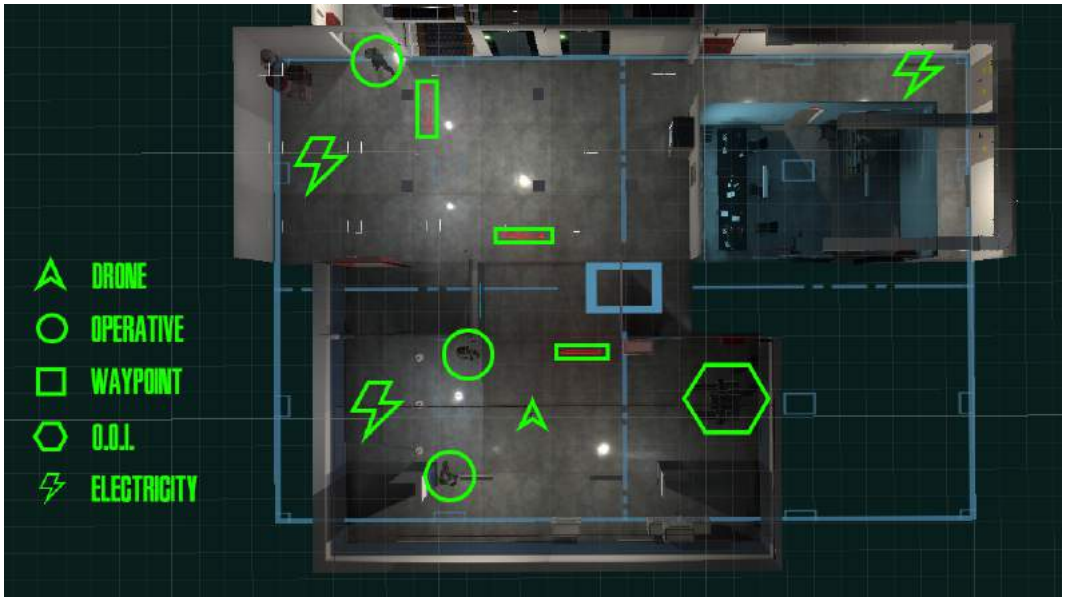


Figure 7.2: Top-down view of the Voice-Only Simulator with markers.

7.4.2 Simulator: Visual-Only Confirmation

The Visual-Only simulator does not have any audial UI confirmations. Fig. 7.3 and Fig. 7.4 show images of the user's view with a marked Object of Interest (O.O.I.) (robot), visual UI example, and button activation/deactive/busy/ready states. Fig. 7.5 shows the placement of the various objects for the SAGAT. The placement was designed to require minimum of three rotations (using the "backward" command) and to have a goal completion time of 180 seconds.

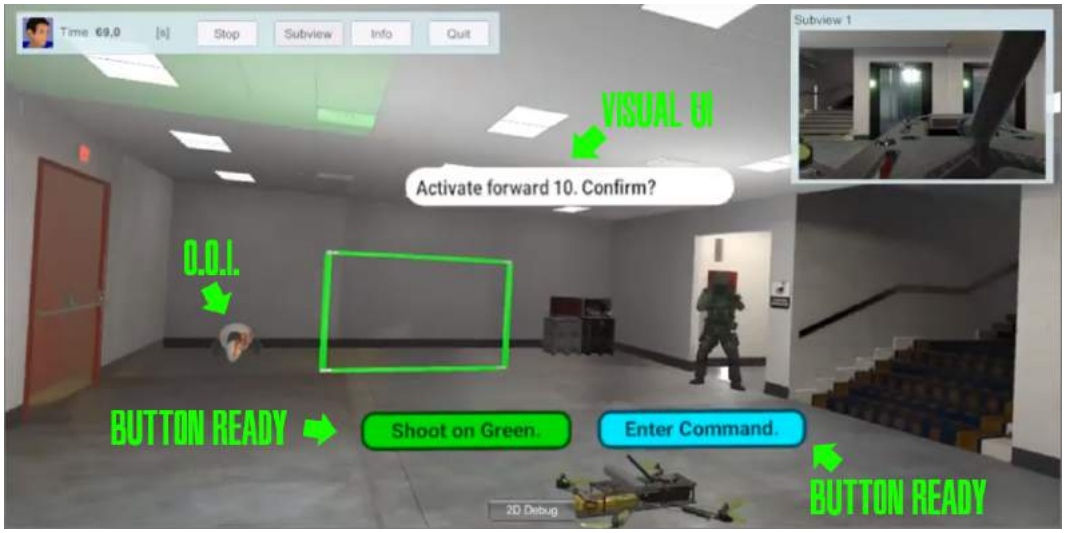


Figure 7.3: Example of User View with ”Ready” Buttons.

Figure 7.4: Example of User View with ”Busy” Buttons.

7.4.3 Simulator: Multimodal Confirmation

The Multimodal simulator has both visual and audial UI confirmations. Fig. 7.1 shows an image of the user's view with a marked environmental effect (electricity), operative, and waypoint example. Fig. 7.6 shows the placement of the various objects for the SAGAT. The placement was designed to require minimum of five rotations and to have a goal completion time of 240 seconds.

Users were given an simplified version of the map with coordinates to place marker

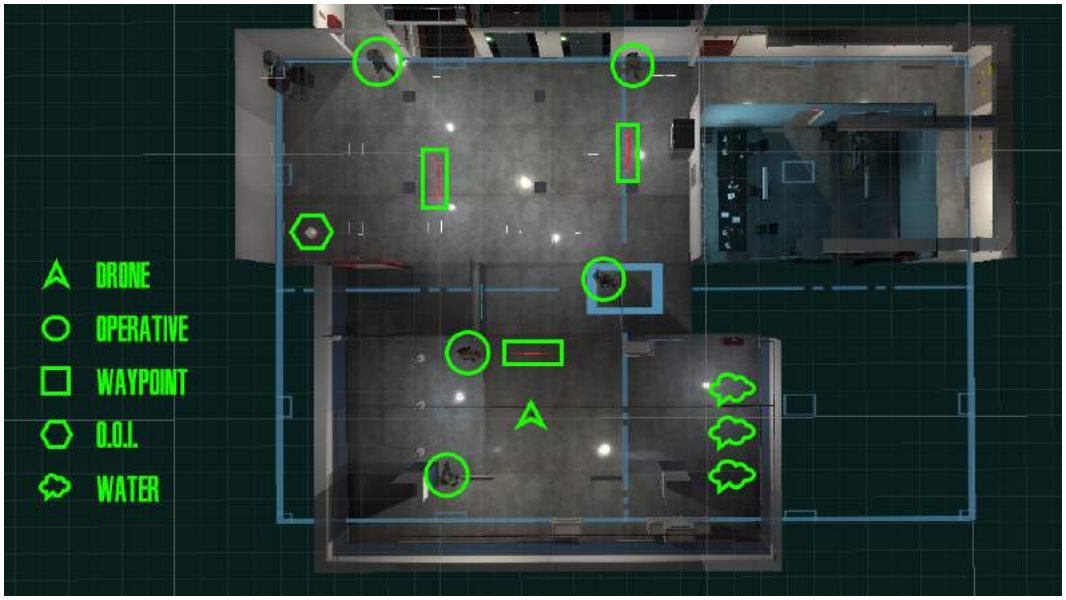


Figure 7.5: Top-down view of the Visual-Only Simulator with markers.

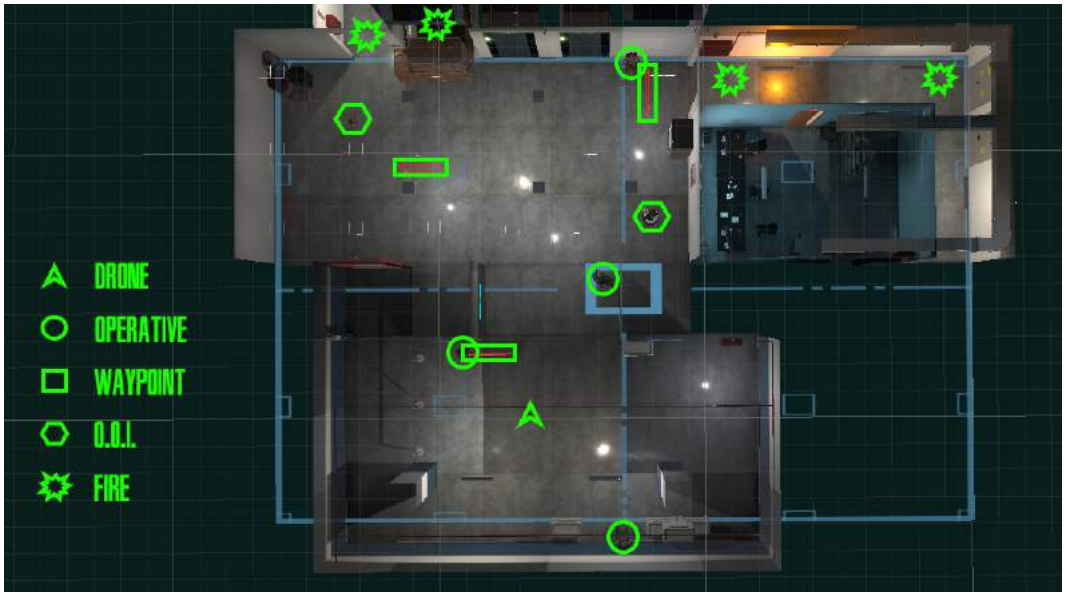


Figure 7.6: Top-down view of the Multimodal Simulator with markers.

positions during their User Survey (Fig. 7.7). The full user survey can be found in Sec. 9.3 in the Appendix.

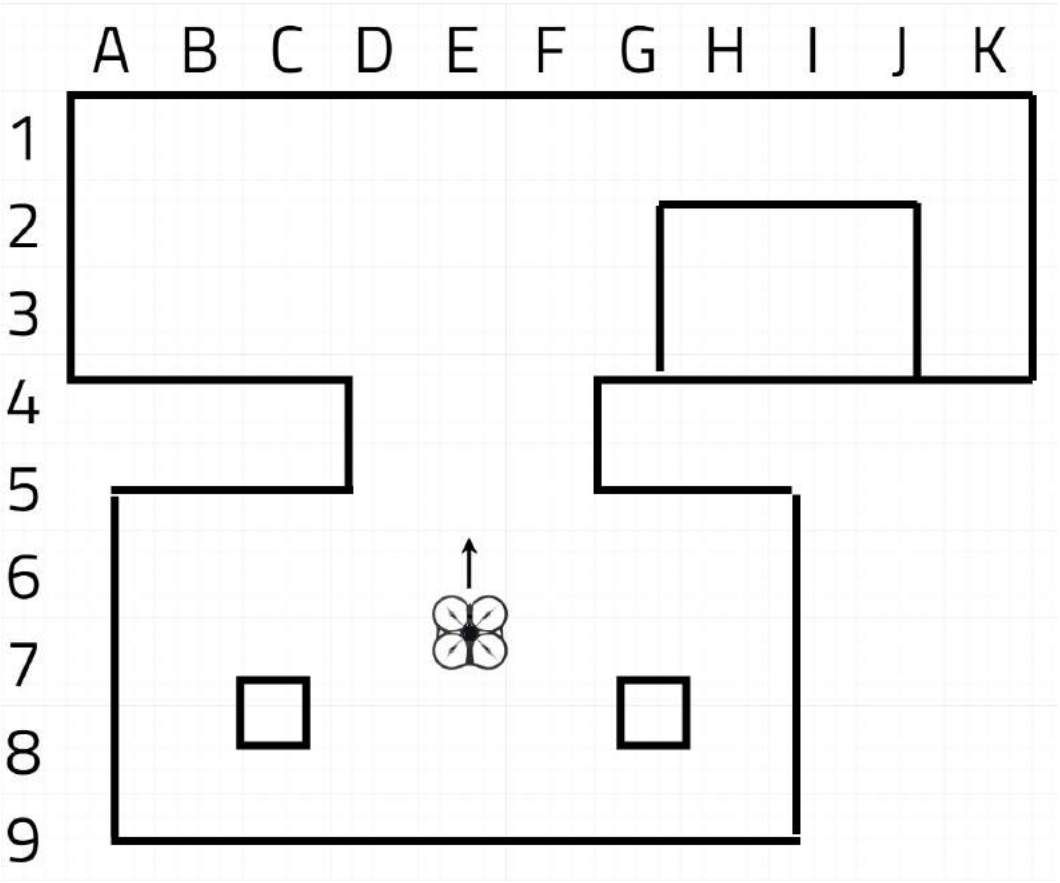


Figure 7.7: Example of Empty Coordinate Map for User Survey.

Chapter 8

Evaluation

This Chapter describes the results of the three primary tests (NASA-TLX, A-PVT, and SAGAT) along with a comparison of the quantitative and qualitative results of the users. This is followed by a discussion regarding additional insights that arrived during testing that will be useful for future designs and implementation.

8.1 Results

There are six subsections for results. The first three are the direct results of the proposed tests. The fourth subsection attempts to determine how accurate users were at self-reporting to evaluate the importance of subjective qualities in the final implementation. With respect to meeting functional requirements, user preferences are often considered *quality requirements* or *non-functional requirements* as they are more challenging to validate scientifically and can vary between users. However, in the case of UIs for mixed human-robot systems, the results indicate that proper system behavior and ultimately, mission success, will be influenced by user preferences. The fifth subsection investigates whether or not trends can be derived based on the scoring results over time. Finally, the sixth results section describes the overall scoring results and is used for recommending the modality approach for future implementations.

8.1.1 NASA-TLX Results

This section lists the results of the *Self-Reporting Test* (SRT) of the NASA-TLX. Fig. 8.1 details the Voice-Only Simulation, Fig. 8.2 details the Visual-Only Simulation, and Fig. 8.3 details the results of the Multimodal Simulation. The median values are also displayed for each user to provide insights in the general feelings toward the metric. A total of six questions were asked (Sec. 7.3.1).

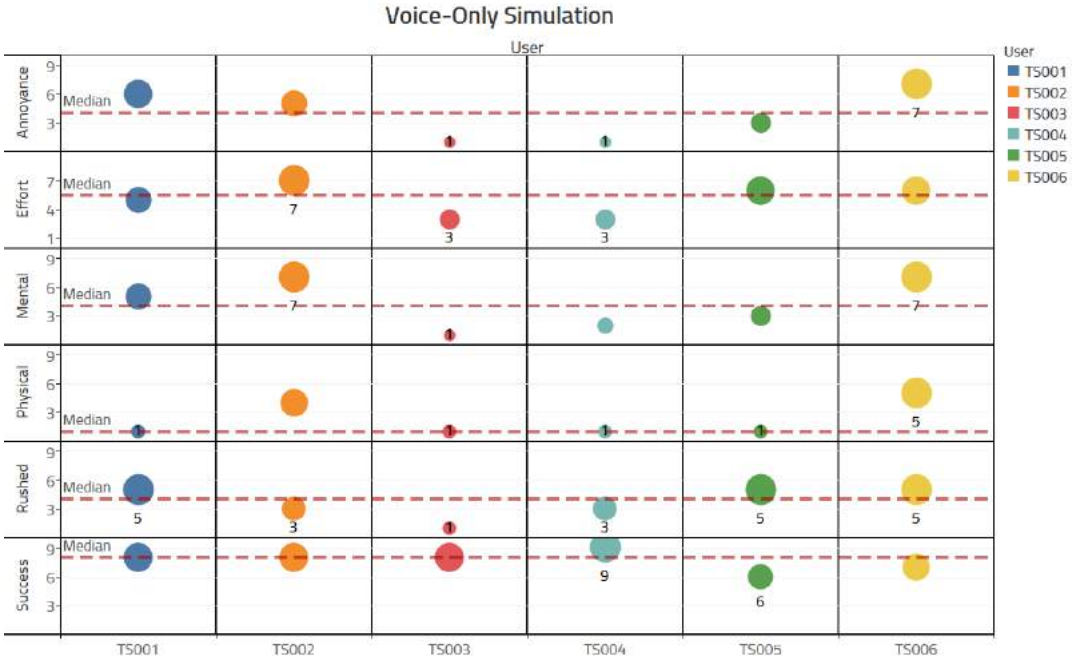


Figure 8.1: Results of Voice-Only Simulation.

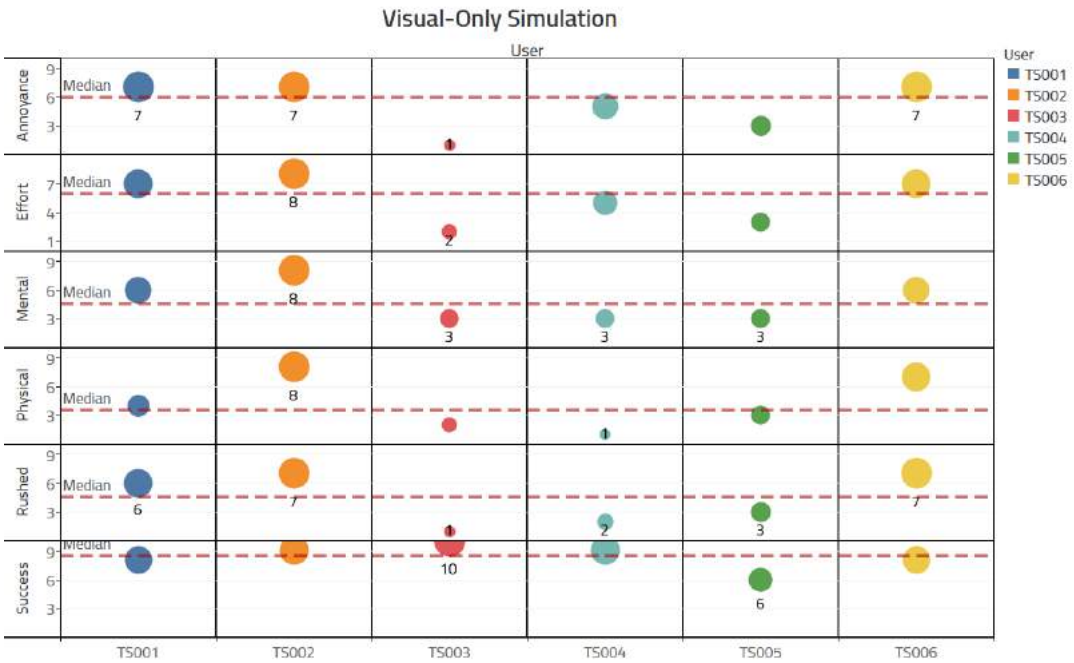


Figure 8.2: Results of Visual-Only Simulation.

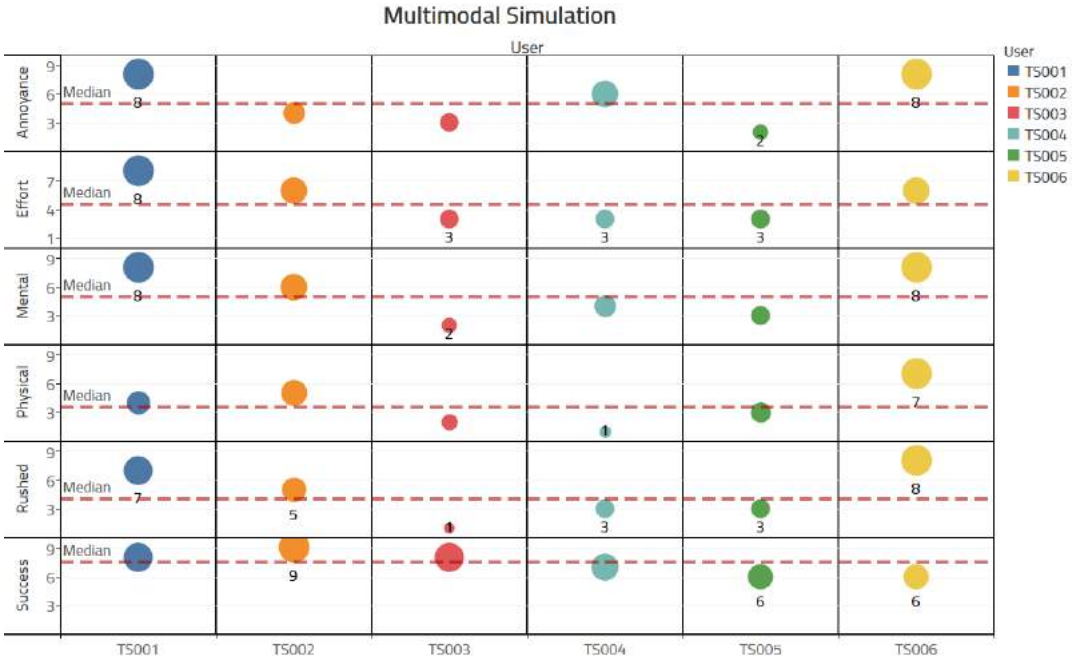


Figure 8.3: Results of Multimodal Simulation.

8.1.2 A-PVT Results

This section contains the results for the A-PVT. Sec. 9.3 contains the recorded values for the response time and total success button deactivations (*hit*). Fig. 8.4 shows the final scoring with the defined metrics. Fig. 9.17 shows the results of the pre-PVT test though they were not used in the final analysis (please see Sec. 9.3 for details).

Since the order of the scene player impacts user performance, an *Order Factor* was created. The Order Factor is the penalty for preventing bias related to increased experience and exposure within simulation environments. The deduction is 5% per Scene played. Table 8.1 shows the number of scenes that occurred in a particular order (chosen randomly) along with the factor for being in that order.

Table 8.1: Order of Occurence and Order Factor for Scenes.

Scene	**First**	**Second**	**Third**
Voice	3	2	1
Visual	1	2	3
Multimodal	2	2	2
Order Factor	1.00	0.95	0.90

In order to prevent other potential biasing in the results, several additional metrics were created including the *Miss Factor*, *Time Factor*, and *Penalty Rate*. The *Time Factor* is a linear decrease in potential score based on the *Response Time* (RT) of the user. Essentially, for every second that the A-PVT button is active, the maximum score decreases until 10 seconds pass and the activation is considered a *miss*. As a miss is highly detrimental to the "Mission: Quickly Deactivate the Target" (Sec. 7.2.2), it is weighted more heavily. Table 8.2 shows the scaling chart used to calculate the miss and time factors (though the exact proportion was used in the final analysis, e.g., a weight median response time has a time factor of 0.77).

Deviations from the *Target Goal Time* also occur a penalty based on the timings created in Sec. 7.4. The *Penalty Rate* (Pen Rate) is based on the difference between the *Goal Completion Time* of a scene and the user's *Actual Completion Time*. The *Penalty Rates* have been derived based on the proportion of the time that passes the *Target Goal Time*, i.e., since the Visual-Only Scene was created to take less time than the Multimodal Scene, the time to occur a penalty should be less (set at 5% per 30 seconds and scaled to 1% per 5 seconds). The scaling is based on a 10% decrease for each fixed 33% of time exceeded (i.e., the goal time of completing the Visual-Only Scene is three minutes and if the user requires four minutes to complete the scene their score drops 10%). The *Penalty Rates* for the scenes can be seen in Table 8.3.

Fig. 8.4 shows the final scoring with the metrics and penalties for each user and scene. A *hit* is only counted if the button is deactivated within ten seconds of becoming

Table 8.2: A-PVT: Miss and Hit Factors based on Response Time and Weighted Hit Percentage.

Wgt Hit %	**Miss Factor**	**Response Time**	**Time Factor**
1.00	1.00	0.00	1.00
0.90	0.85	1.00	0.90
0.80	0.70	2.00	0.80
0.70	0.55	3.00	0.70
0.60	0.40	4.00	0.60
0.50	0.25	5.00	0.50
0.40	0.10	6.00	0.40
0.30	fail	7.00	0.30
-	-	8.00	0.20
-	-	9.00	0.10
-	-	10.00	0.01

Table 8.3: A-PVT: Penalty Rates based on Time Differences between Average Completion Times and Target Goal Times.

Scene	**Goal Time [s]**	**Average Time [s]**	**Time Difference [s]**	**Penalty Rate**
Voice	210	375	-165	**1% / 7sec**
Visual	180	408	-228	**1% / 6sec**
Multimodal	240	401	-161	**1% / 8sec**

active. The count value is the number of times the button was considered active during a scene. The *Weighted Median Response Time* (Wgt Med RT) is an adjusted value that omits the response time from misses to prevent skewing. The *Time Factor* is based on a linear deduction of points related to the optimal response time (0 seconds). The *Weighted Hit %* (Wgt Hit %) is the value in which misses greater than 20 seconds ($RT > 20$) add an additional miss value to the count. This value was chosen as the maximum random timer value is 20 seconds. The *Penalty Count* (Pen Count) is the number of Pen Rate occurrences based on the total time exceeded. The *Penalty Factor* (Pen Factor) is the normalized value ($(100 - Pen_{count})/100$) to keep the factor value bounded between 0 and 1. The *Score* is the average of the four Factors to normalize the final value between 0 and 1. Scores can have a maximum value of 1 (good) with values approaching zero indicating a poor score.

		Order		Response Time			Hit/Miss		Time based on Difficulty					
User	Test	Order	Order Factor	Wgt Ave RT	Wgt Med RT	Time Factor	Wgt Hit %	Miss Factor	Total Time	T Difference	Pen Rate	Pen Count	Pen Factor	Score
TS001	VOX	2	0.90	3.68	2.28	0.77	0.71	0.57	406	196	1% / 7sec	28.00	0.72	0.74
	VIZ	3	0.95	2.42	1.89	0.81	0.75	0.63	333	153	1% / 6sec	25.50	0.75	0.78
	MM	1	1.00	3.50	3.01	0.70	0.76	0.65	414	174	1% / 8sec	21.75	0.78	0.78
TS002	VOX	1	1.00	2.96	2.26	0.78	0.93	0.90	335	125	1% / 7sec	17.86	0.82	0.87
	VIZ	3	0.90	2.77	1.68	0.83	0.94	0.96	349	169	1% / 6sec	28.17	0.72	0.85
	MM	2	0.95	3.22	2.23	0.78	1.00	1.00	329	89	1% / 8sec	11.13	0.89	0.90
TS003	VOX	3	0.90	1.48	1.22	0.88	0.95	0.93	306	96	1% / 7sec	13.71	0.86	0.89
	VIZ	2	0.95	1.88	1.39	0.86	0.83	0.75	183	3	1% / 6sec	0.50	1.00	0.89
	MM	1	1.00	1.66	1.31	0.87	1.00	1.00	258	18	1% / 8sec	2.25	0.98	0.96
TS004	VOX	2	0.95	2.71	1.52	0.85	0.82	0.73	598	388	1% / 7sec	55.43	0.45	0.74
	VIZ	1	-	-	-	-	-	-	805	625	1% / 6sec	-	-	-
	MM	3	0.90	3.11	2.48	0.75	0.65	0.48	480	240	1% / 8sec	30.00	0.70	0.71
TS005	VOX	1	1.00	5.71	5.09	0.50	0.56	0.35	243	33	1% / 7sec	4.71	0.95	0.70
	VIZ	3	0.90	2.80	1.95	0.80	0.82	0.73	453	273	1% / 6sec	45.50	0.55	0.74
	MM	2	0.95	3.62	3.30	0.67	0.96	0.95	413	173	1% / 8sec	21.63	0.78	0.84
TS006	VOX	1	-	-	-	-	-	-	360	150	1% / 7sec	-	-	-
	VIZ	2	0.95	4.48	3.44	0.66	0.67	0.43	327	147	1% / 6sec	24.50	0.76	0.70
	MM	3	0.90	4.32	4.50	0.55	0.47	0.21	510	270	1% / 8sec	33.75	0.66	0.58

Figure 8.4: A-PVT Results: Metrics and Scoring.

8.1.3 SAGAT Results

This section contains the data values and analysis for the SAGAT. Fig. 9.21, Fig. 9.23, and Fig. 9.24 show user results for reporting the *Number* and *Position* of *Operatives*, *Environmental Elements*, and *Objects of Interest* (O.O.I.), in each scene, respectively. Fig. 9.22 shows the results of users recalling the locations of the *Waypoints*. Fig. 8.5 shows the final *Score* based on a normalized *Base Score* multiplied with the *Order Factor* (same as the previous Order Factor). The *Base Score* is calculated based on the following formula:

$$Base = \frac{2 \times \Sigma(Corr_{find}) + \Sigma(Corr_{pos})}{k}$$

where $Corr_{find}$ (Corr Find %)is the proportion of objects correctly found, $Corr_{pos}$ (Corr Pos %) is the proportion of locations correctly recalled, and k is the total number of values and is used to normalize value of the final results between 0 and 1. Correctly identifying the number of operatives, environmental effects, and O.O.I. were given double the weight as they are critically more important to discover (and report). Knowing the precise location is also beneficial for reporting but is usually held with lower regard than what was seen. Furthermore, this scoring mitigates ”double punishment” as users that were unable to detect certain field objects are not going to be able to report their locations either.

			Operatives		Environmental		Object of Interest (OOI)		Waypoint	Scoring		
User	Test	Order	Corr Find %	Corr Pos %	Corr Find %	Corr Pos %	Corr Find %	Corr Pos %	Corr Pos %	Base Score	Order Factor	Score
TS001	VOX	2	1	0.66	1	0.33	1	1	0	0.73	0.90	0.65
	VIZ	3	1	0.6	1	0	1	1	1	0.87	0.95	0.83
	MM	1	0.666	0.4	1	1	0.5	0.5	1	0.75	1.00	0.75
TS002	VOX	2	1	1	1	0.66	1	1	0.8325	0.94	1.00	0.94
	VIZ	3	1	0.9	1	1	1	1	1	0.99	0.90	0.89
	MM	1	1	1	1	0.5	0.5	0.5	1	0.82	0.95	0.78
TS003	VOX	3	1	1	1	1	1	0	0.666	0.85	0.90	0.76
	VIZ	2	1	1	1	1	1	0	1	0.91	0.95	0.86
	MM	1	1	1	1	1	1	1	1	1.00	1.00	1.00
TS004	VOX	2	1	1	1	0	0	0	1	0.64	0.95	0.60
	VIZ	1	0.333	0.4	0	0	0	0	1	0.28	1.00	0.28
	MM	3	0.666	0.5	1	1	0	0	1	0.62	0.90	0.56
TS005	VOX	1	1	1	1	1	0	0	1	0.73	1.00	0.73
	VIZ	3	0.666	0.8	1	1	1	1	1	0.92	0.90	0.83
	MM	2	1	0.375	1	0.5	0	1	1	0.72	0.95	0.68
TS006	VOX	1	1	0.66	0	0.66	1	1	1	0.76	1.00	0.76
	VIZ	2	0.666	0.7	1	1	1	1	0.8325	0.88	0.95	0.84
	MM	3	0.666	0.375	1	0	0	0	0.8325	0.49	0.90	0.44

Figure 8.5: SAGAT Results: Metrics and Scoring.

8.1.4 Ranking Results and Comparison

The motivation behind this section is to evaluate whether or not the NASA-TXL was an effective measure for subjective responses in self-report. This is crucial when attempting to determine what weight or priority user preference should be regarding when designing the final system. Previously, all of the *Scores* were normalized to provide a means of comparison between the results of tests, modalities, and users. Fig. fig:SA06c shows the three scores for the A-PVT, SAGAT, and NASA-TXL SRT. The values of the SRT (between 1 and 10) have been normalized to be between 0 and 1. The average scores (*Ave Score*) between the *Score A-PVT* and *Score SAGAT* have be calculated. The result is subtracted from the *TXL-SRT* score that users self-reported to acquire the difference (*Diff SRT*). Values close to zero imply that the user was relatively good at self-reporting, as can be seen with 4/6 of the users being within, on average, 10% of their estimated score. Negative values indicate that the user over-estimated their performance whereas positive values indicate users under-estimated their performance.

Fig. 8.7) demonstrates another metric calculated for estimating the accuracy of the NASA-TLX with respect to *Rank*. Based on the users' TXLs (*Rank TXL*), the users were ranked based on the values reported (with 1 meaning best performance). These values were then compared to their individual average rankings (*Ave Rank*) for the *Score A-PVT* and *Score SAGAT*. The *Rank TXL* was then subjected from the *Ave Rank* (Ave - TXL). This value is used as a *Step* (from mathematics notation) to demonstrate within how many steps the rank of the user was from their reported NASA-TLX values. Overall, users were able to correctly assess their own performance quite well (except for one user that over-estimated their performance. Fig. 8.8 shows that 22% of self-reported ranks were exactly correct, with 56% of results being within one place of comparable ranks. 11% of results are within two steps and the final 11% are within four steps (both from

			Score				
User	Test	Order	Score A-PVT	Score SAGAT	TXL-SRT	Ave Score	Diff SRT
TS001	VOX	3	0.74	0.65	0.8	0.70	0.10
	VIZ	2	0.78	0.83	0.8	0.81	-0.01
	MM	1	0.78	0.75	0.8	0.77	0.03
TS002	VOX	2	0.87	0.94	0.8	0.91	-0.11
	VIZ	3	0.85	0.89	0.9	0.87	0.03
	MM	1	0.90	0.78	0.9	0.84	0.06
TS003	VOX	1	0.89	0.76	0.8	0.83	-0.03
	VIZ	3	0.89	0.86	1.0	0.88	0.12
	MM	2	0.96	1.00	0.8	0.98	-0.18
TS004	VOX	2	0.74	0.60	0.9	0.67	0.23
	VIZ	1	-	0.28	0.9	0.28	0.62
	MM	3	0.71	0.56	0.7	0.63	0.07
TS005	VOX	1	0.70	0.73	0.6	0.71	-0.11
	VIZ	3	0.74	0.83	0.6	0.79	-0.19
	MM	2	0.84	0.68	0.6	0.76	-0.16
TS006	VOX	1	-	0.76	0.7	0.76	-0.06
	VIZ	2	0.70	0.84	0.8	0.77	0.03
	MM	3	0.58	0.44	0.6	0.51	0.09

Figure 8.6: Cumulative Scoring and Difference.

the same user).

As a result, it appears that users are quite adept at reporting their own performance based on subjective cognitive load. Therefore, user preference should be taken into account and accommodated when designing UIs for mixed human-robot teams.

			Score			Rank			Step		
User	Test	Order	Score PVT	Score SAGAT	TXL-SRT	Rank PVT	Rank SAGAT	Rank TXL	Ave Rank	Ave - TXL	Step
TS001	VOX	3	0.28	0.65	0.8	3	5	2	4.0	2.0	2
	VIZ	2	0.36	0.83	0.8	3	3	3	3.0	0.0	0
	MM	1	0.35	0.75	0.8	4	3	2	3.5	1.5	2
TS002	VOX	2	0.57	0.94	0.8	2	1	2	1.5	-0.5	1
	VIZ	3	0.52	0.89	0.9	2	1	2	1.5	-0.5	1
	MM	1	0.66	0.78	0.9	2	2	1	2.0	1.0	1
TS003	VOX	1	0.63	0.76	0.8	1	2	2	1.5	-0.5	1
	VIZ	3	0.61	0.86	1.0	1	2	1	1.5	0.5	1
	MM	2	0.85	1.00	0.8	1	1	2	1.0	-1.0	1
TS004	VOX	2	0.26	0.60	0.9	4	6	1	5.0	4.0	4
	VIZ	1	-	0.28	0.9	6	6	2	6.0	4.0	4
	MM	3	0.23	0.56	0.7	5	5	5	5.0	0.0	0
TS005	VOX	1	0.16	0.73	0.6	5	5	6	5.0	-1.0	1
	VIZ	3	0.29	0.83	0.6	4	5	4	4.5	0.5	1
	MM	2	0.47	0.68	0.6	3	4	4	3.5	-0.5	1
TS006	VOX	1	-	0.76	0.7	6	3	5	4.5	-0.5	0
	VIZ	2	0.20	0.84	0.8	5	3	3	4.0	1.0	1
	MM	3	0.07	0.44	0.6	6	6	6	6.0	0.0	0

Figure 8.7: Comparing Average Scores with Self-Reported Scores.

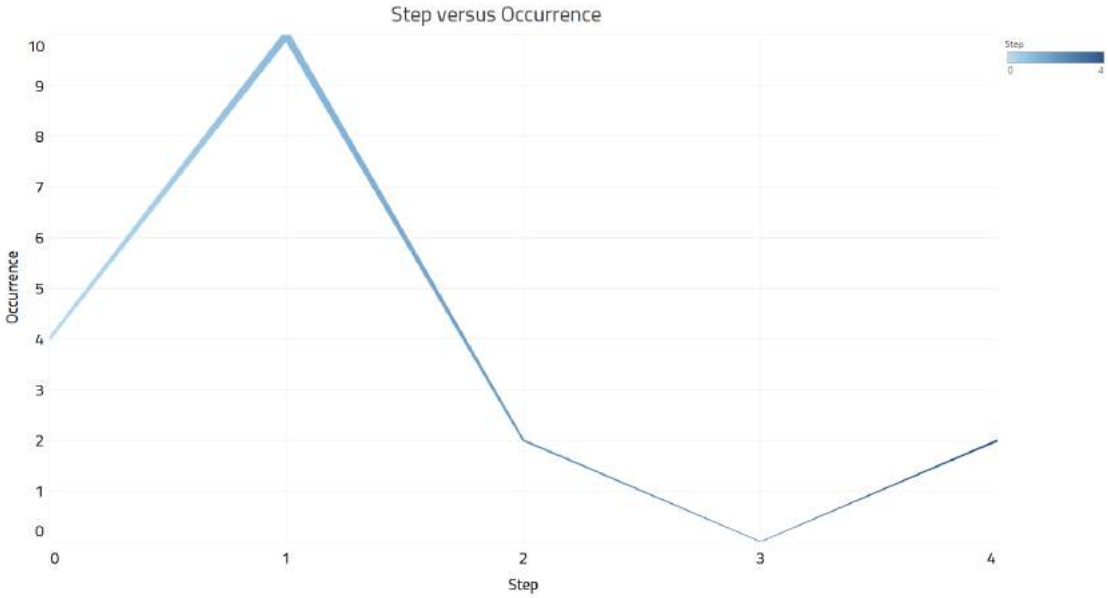


Figure 8.8: Relative Accuracy of User Self-Reporting.

8.1.5 Trend Analysis

This section seeks to investigate trends between different modalities with respect to the A-PVT and SAGAT scores overtime. The scores used are the same as previously derived in Sec. 8.1.2 and Sec. 8.1.3. Fig. 8.9

For the A-PVT results over time (Fig. 8.9), an *exponential* trendline shows *Multi-*

modal R-Squared score is approximately is approximately 0.85. *Linear* trendlines show that *Visual-Only R-Squared score* is approximately 0.39 and the *Voice-Only R-Squared score* is approximately 0.05. These are weak values and do not confidently predict trends (though the general can be seen graphically).

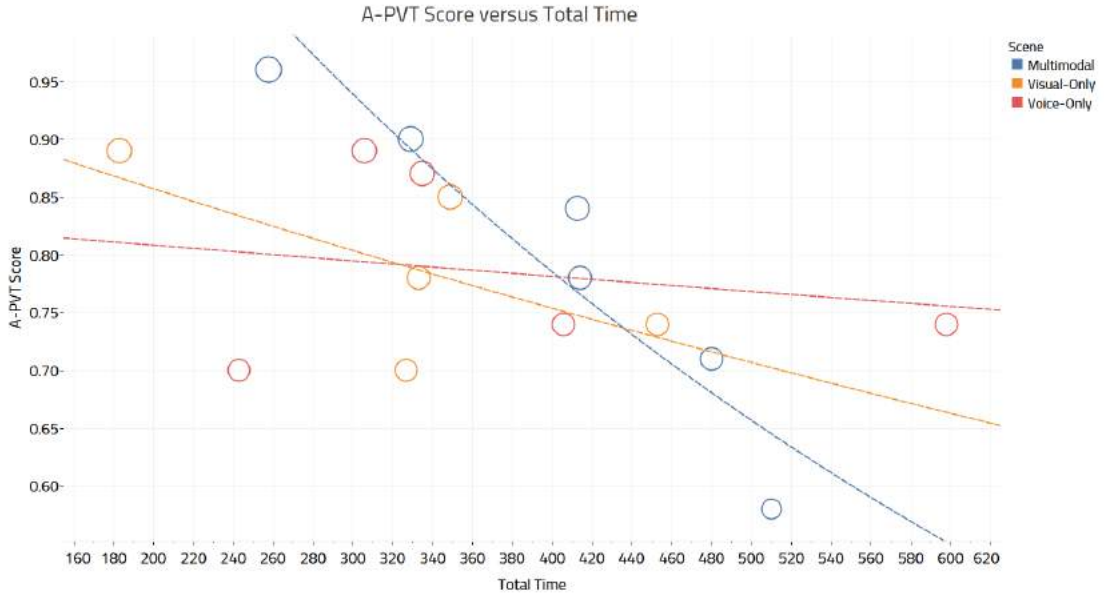


Figure 8.9: A-PVT Score versus Total Time.

For the SAGAT results over time (Fig. 8.10), an *exponential* trendline shows *Multimodal R-Squared score* is approximately 0.89 and the *Visual-Only R-Squared score* is 0.85. A *linear* trendline was used for the *Voice-Only R-Squared score* resulting in a value of 0.41 (not very reliable).

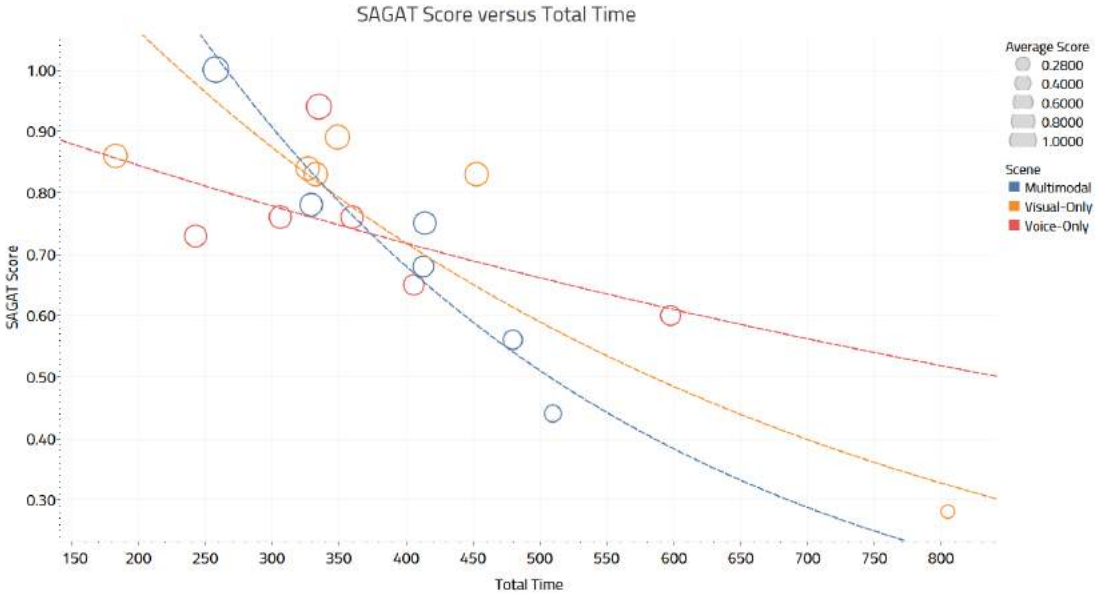


Figure 8.10: SAGAT Score versus Total Time.

Finally, both the A-PVT (quantitative) and SAGAT (qualitative) results are compared (Fig. 8.9). Using an *exponential* trendline, the *Multimodal R-Squared score* is approximately 0.90 and the *Visual-Only R-Squared score* is approximately 0.88. Both of these values are considered accurate and indicate clear trends in performance over time when both quantitative and qualitative results are compared. A *linear* trendline shows that the *Voice-Only R-Squared score* is again only approximately 0.05, making the results unreliable.

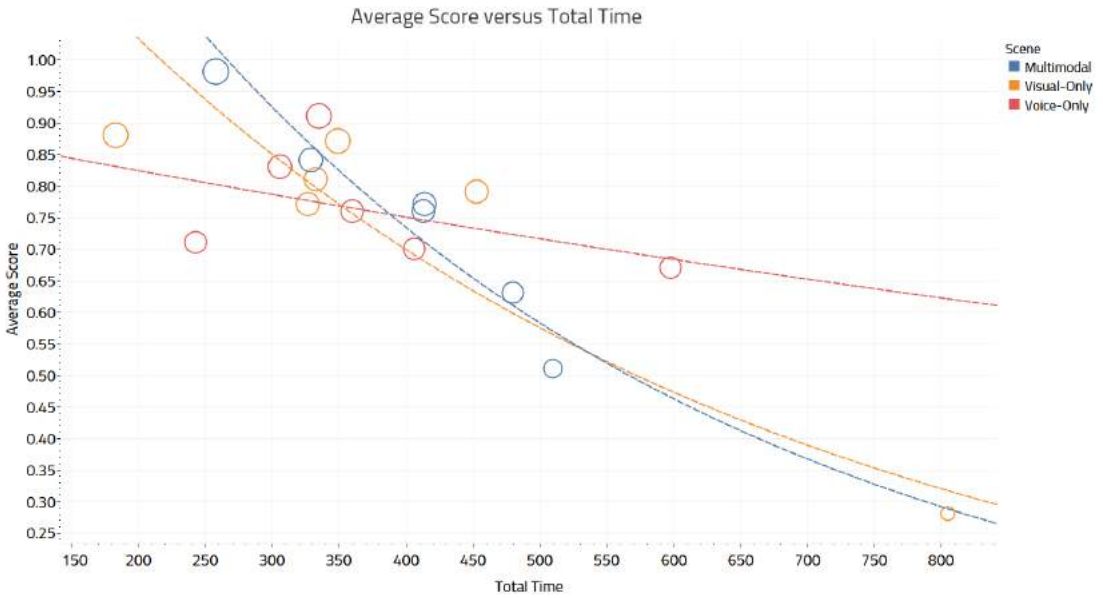


Figure 8.11: Average Score versus Total Time.

8.1.6 Scoring Results

With the previous scores/results considered, enough information is available to make future design decisions. Table 8.4 lists the overall scores for each model compared with the users' reported preference after the experiment. The recommendation is to use a *multimodal* approach to UI design in future implementations. Furthermore, it was found that there are distinguishable trends in the multimodal implementation that were not as well defined in the voice-only and visual-only models.

Table 8.4: Scoring results for NASA-TLX, A-PVT, and SAGAT.

Test / Scene	**Voice**	**Visual**	**Multimodal**
USER-SRT	2/6	0/6	**4/6**
NASA-TLX	0.77	**0.83**	0.73
A-PVT	0.32	0.30	**0.43**
SAGAT	0.74	**0.75**	0.70
Average	0.61	0.62	0.62

8.2 Discussions

While six users were enough to derive some meaningful insights, results, and trends, having more users to test the scenes would have likely been better. Therefore, having six users is considered a limitation. Furthermore, it was hoped to be able to find more insights on the *working memory* of users but these aspects are not explicit and can only be assumed (i.e., from Fig.8.10 we might make an assumption that the working memory of the user declines over time leading to lower SAGAT scores). However, making such assumptions can be detrimental and therefore, working memory results have not been further explored at this time [Lavie et al., 2004].

For the multimodal implementation, all six users gave higher precedence to the auditory confirmation than the visual confirmation. Whenever a command was not properly understood by the speech recognizer, all users would repeat the command even if the visual UI correctly displayed the command. To investigate this, modality and contiguity aspects of CTL were revisited. In multimodal applications, a modality effect occurs in which information communicated auditorily leads to a reduced cognitive load and better learning rate over visually (text) based information sources [Moreno and Mayer, 1999].

Unintentionally, the visual UI would display the text of the visual confirmation slightly after the voice confirmation due to unforeseen lag in Unity rendering text for UI elements. During scene creation, this was not noticeable and it was not anticipated to have any impact on user testing. Apparently, crossmodal information that falls outside of a reaction time window will lead to the information not properly being fused in a statistically optimal (Bayesian) manner [Colonius and Diederich, 2010]. Especially in complex scenes, the modality that is registered first will be prioritized, in this case, auditory confirmation [van Wanrooij et al., 2009], [Murai and Yotsumoto, 2018]. This insight demonstrates that future multimodal implementations must report real-time information to the user's UIs simultaneously or the signals will not be integrated resulting in the leading signal taking precedence for user-level decision-making processes.

Robotic systems can also benefit from multimodal input signals. Analyzing several cues simultaneously, like speech and gestures, would allow for the system to respond with a higher reliability in interpretation [Karpov and Yusupov, 2018]. If the drone receives multiple valid command hypotheses or if sensors have noise, operations with higher safety levels and the channel with the higher quality signal should be prioritized [Rossi et al., 2013]. This allows the drone to behave optimally in various situations like the operative engaging a target or the environment has environmental effects. Essentially, multimodal inputs are necessary for both optimizing the behavior of both human agents and adaptive, intelligent agents.

Outside of the research questions proposed in this Master's Thesis, a wealth of additional insights arrived from user testing. One of the most notable is the impact of parallax on users at a distance from the drone. If the user was more than approximately three meters away and off-angled (between 90° and 180°), the users could not reliably

align the drone to fly through the waypoints. To obtain reliable alignment, users needed to be orthogonal and parallel to one side of the drone. A virtual RGB camera and a depth camera were added to the drone for the observer but not for the user. Future HMDs should include a camera feed from the drone to reduce parallax. A camera feed would also be beneficial in events in which the operator cannot see the drone (visual obstruction) or if their attention is elsewhere and they are not looking at the drone.

While the environment was reported as realistic and immersive, all users demonstrated the same behavior with regards to their personal safety distance from the drone. Users held no minimum safety distance from the virtual drone, often colliding directly with the drone, subconsciously knowing they were not in any danger of injury. Therefore, even highly immersive virtual simulations cannot (currently) provide all of the proper insights into safety-critical missions [Haans, 2014]. Had the users viewed the drone as a dangerous device, they may have behaved differently which would have impacted their testing scores. Users need immersive virtual environments in which they can move around freely in the real-world and view objects in the virtual environment as they would in real-life for accurate emulation [Caporusso et al., 2020].

Chapter 9

Conclusion and Future Work

9.1 Summary

In Sec. 1 and Sec. 2, the topic and motivation for this Master's Thesis were described. Sec. 3 defined some important concepts and related work. Sec. 4 detailed the objectives and research questions. Sec. 5 covered the design and set up for the experiment followed by Sec. 6 which contains information used during the design and prototyping phase. In Sec. 7, the missions, scenes, and user tests were discussed. Sec. 8 provided results based on the data collected during the user testing phase.

9.2 Conclusion

The original goal was to design a UI for voice-controlled drones that would improve user cognitive load and optimize agent behavior for mixed human-robot teams. Through user testing, insights regarding the cognitive load including situational awareness and performance (response time) were acquired in complex virtual environments. A speech recognizer was created that was able to correctly control a drone and only upon explicit activation to prevent unintended behaviors. Furthermore, a specialized library was created to improve voice recognition and data communication was optimized to prevent the drone from using additional resources (memory, power consumption). In this sense, the robotic agent (drone) was optimized working with users based on speech commands. However, the ASR was not perfectly accurate which lead to increase mission completion time and cognitive load. Furthermore, it was discovered that implementations should have low-latency and should consider that wireless networks are not available or desirable (robustness, safety). Multimodal UI implementations should have signals and responses that are synchronized for optimal integration which leads to a reduction in user cognitive load.

From this Master's Thesis, an improved UI for indoor reconnaissance drones can

be created. It turns out that designing drones (or other agents) for mixed human-robot teams is not a trivial task and requires careful design at each layer. Aspects as minor as selecting the wrong color for the HMD UI elements or selecting a particular word within a vocabulary (i.e, “turn” instead of “rotate”) can lead to vastly different results. Constraints and functional requirements such as offline communications completely change the final implementation and performance of the system. This demonstrates that system engineering and requirements engineering are absolutely crucial and may require a large time investment and many iterations in order to get an effective system that meets the goals of the use case.

9.3 Future Work

At this time, general guidelines can be created for designing UIs in HMDs for voice-controlled robots. In the future, the multimodal UI will be improved and optimized within the created Unity environments. Ideally, additional user testing would be taken in the same form as followed in this Master’s Thesis for additional iterations to gain more insights and improvements. A proposed UI is shown in Fig. 9.1 to include indications on which goals have been completed as well as using icons instead of text for visual confirmations. It is posited that icons would reduce cognitive load further by quickly showing the user the directionality of the voice command instead of requiring the user to read longer texts. Users could also use the Vive controller to disable targets in the field and the response time could be calculated to imply some form of personal safety whenever enemies are present (where delays in response time can be fatal).

Figure 9.1: An example of a proposed multimodal UI for future implementations.

Furthermore, it is intended to apply this framework to multi-agent systems consist-

ing of multiple drones or mixed robotic teams (drones and ground robots) where the agents have different tasks. This can easily be done by giving each agent its own activation word. It would be interesting to find how adding additional robots to the team impacts the cognitive load and performance of the field operator. Agents will need to be responsible for some of their own decision-making processes, especially in environments in which the user is not able to give valid/safe commands confidently. It is intended to use the RHBP from DAI-Labor for the decision making processes [Hrabia et al., 2017].

Bibliography

[Broadbent, 1958] Broadbent, D. (1958). *Perception and Communication*. Pergamon Press.

[Brugali and Fayad, 2002] Brugali, D. and Fayad, M. (2002). Distributed computing in robotics and automation. *Robotics and Automation, IEEE Transactions on*, 18:409 – 420.

[Buker et al., 2012] Buker, T., Vincenzi, D., and Deaton, J. (2012). The effect of apparent latency on simulator sickness while using a see-through helmet-mounted display: Reducing apparent latency with predictive compensation. *Human factors*, 54:235–49.

[Bundesministerium für Bildung und Forschung, 2019] Bundesministerium für Bildung und Forschung (2019). Projektumriss_inlased. `http://mobile.sifo.de/files/Projektumriss_InLaSeD.pdf`.

[Caporusso et al., 2020] Caporusso, N., Carlson, G., Ding, M., and Zhang, P. (2020). *Immersive Virtual Reality Beyond Available Physical Space*, pages 315–324.

[Chandler and Sweller, 1991] Chandler, P. and Sweller, J. (1991). Cognitive Load Theory and the Format of Instruction. *Faculty of Education - Papers*, 8.

[Cho and Proctor, 2003] Cho, Y. S. and Proctor, R. W. (2003). Stimulus and Response Representations Underlying Orthogonal Stimulus-Response Compatibility Effects. *Psychonomic Bulletin & Review*, 10(1):45–73.

[CMU-Sphinx, 2019] CMU-Sphinx (2019). ROS-Pocketsphinx. `https://github.com/cmusphinx/ros-pocketsphinx`.

[Colonius and Diederich, 2010] Colonius, H. and Diederich, A. (2010). The optimal time window of visual–auditory integration: A reaction time analysis. *Frontiers in integrative neuroscience*, 4:11.

[Crick et al., 2017] Crick, C., Jay, G., Osentoski, S., Pitzer, B., and Jenkins, O. (2017). *Rosbridge: ROS for Non-ROS Users*, volume 100, pages 493–504.

[Dinges, 2019a] Dinges, D. F. (2019a). Experiment Summary Reaction Self Test.

[Dinges, 2019b] Dinges, D. F. (2019b). Psychomotor Vigilance Self Test on ISS.

[Dyson, 2008] Dyson, P. (2008). *Cognitive Psychology*. British medical bulletin. Academic Press.

[Endsley, 1995] Endsley, M. (1995). Toward a Theory of Situation Awareness in Dynamic Systems. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 37:32–64.

[Engle, 2002] Engle, R. W. (2002). Working Memory Capacity as Executive Attention. *Current Directions in Psychological Science*, 11(1):19–23.

[EU, 2016] EU (2016). Quality of Life Indicators - Economic and Physical Safety.

[Fernandes and Feiner, 2016] Fernandes, A. and Feiner, S. (2016). Combating vr sickness through subtle dynamic field-of-view modification. pages 201–210.

[Fitzmaurice et al., 2002] Fitzmaurice, G., Ishii, H., and Buxton, W. (2002). Bricks: Laying the foundations for graspable user interfaces.

[Fox et al., 2009] Fox, J., Arena, D., and Bailenson, J. (2009). Virtual reality: A survival guide for the social scientist. *Journal of Media Psychology: Theories, Methods, and Applications*, 21:95–113.

[Graves et al., 2006] Graves, A., Fernández, S., Gomez, F., and Schmidhuber, J. (2006). Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural 'networks. volume 2006, pages 369–376.

[Haans, 2014] Haans, A. (2014). In search of the fixed points on the presence scale. In Felnhofer, A. and Kothgassner, O., editors, *Proceedings of the International Society for Presence Research (ISPR 2014)*, pages 17–21. ISPR.

[Haans and IJsselsteijn, 2012] Haans, A. and IJsselsteijn, W. (2012). Embodiment and telepresence: Toward a comprehensive theoretical framework. *Interacting with Computers*, 24:211–218.

[Hirukawa et al., 2003] Hirukawa, H., Kanehiro, F., and Kajita, S. (2003). *OpenHRP: Open Architecture Humanoid Robotics Platform*, volume 23, pages 99–112.

[Hrabia et al., 2017] Hrabia, C., Wypler, S., and Albayrak, S. (2017). Towards Goal-Driven Behaviour Control of Multi-Robot Systems. In *2017 3rd International Conference on Control, Automation and Robotics (ICCAR)*, pages 166–173.

[Inamura et al., 2011] Inamura, T., Shibata, T., Sena, H., Hashimoto, T., Kawai, N., Miyashita, T., Sakurai, Y., Shimizu, M., Otake, M., Hosoda, K., Umeda, S., Inui, K., and Yoshikawa, Y. (2011). Simulator platform that enables social interaction simulation — sigverse: Sociointelligenesis simulator. pages 212 – 217.

[Ishii, 2008] Ishii, H. (2008). Tangible bits: Beyond pixels.

[Israel et al., 2009] Israel, J., Hurtienne, J., Pohlmeyer, A., Mohs, C., Kindsmüller, M., and Naumann, A. (2009). On intuitive use, physicality and tangible user interfaces. *International Journal of Arts and Technology*, 2:348–366.

[Jacob et al., 2008] Jacob, R., Girouard, A., Hirshfield, L., Horn, M., Shaer, O., Solovey, E., and Zigelbaum, J. (2008). Reality-based interaction: A framework for post-wimp interfaces.

[Jalil et al., 2012] Jalil, N., Yunus, R., and Said, N. (2012). Environmental colour impact upon human behaviour: A review. *Procedia - Social and Behavioral Sciences*, 35:54–62.

[Karpov and Yusupov, 2018] Karpov, A. and Yusupov, R. (2018). Multimodal interfaces of human–computer interaction. *Herald of the Russian Academy of Sciences*, 88:67–74.

[Kipp et al., 2005] Kipp, M., Wahlster, W., Maybury, M., and Bunt, H. (2005). *Fusion and Coordination for Multimodal Interactive Information Presentation*, pages 325–339.

[Klein and Murray, 2007] Klein, G. and Murray, D. (2007). Parallel tracking and mapping for small ar workspaces. pages 225–234.

[Koenig and Howard, 2004] Koenig, N. and Howard, A. (2004). Design and use paradigms for gazebo, an open-source multi-robot simulator. pages 2149 – 2154 vol.3.

[Krakauer et al., 2010] Krakauer, D., Flack, J., and ay, N. (2010). Probabilistic design principles for robust multi-modal communication networks. *Modelling Perception with Artificial Neural Networks*, pages 255–268.

[Lavie et al., 2004] Lavie, N., Hirst, A., Fockert, J., and Viding, E. (2004). Load theory of selective attention and cognitive control. *Journal of experimental psychology. General*, 133:339–54.

[Lugo and Zeil, 2013] Lugo, J. and Zeil, A. (2013). Framework for autonomous onboard navigation with the ar.drone. *2013 International Conference on Unmanned Aircraft Systems, ICUAS 2013 - Conference Proceedings*, pages 575–583.

[Maas, 2017] Maas, A. (2017). Spoken Language Processing-ASR: HMMs, Forward, Viterbi.

[Millisecond, 2019] Millisecond (2019). Inquisit Lab. https://www.millisecond.com/download/, publisher = Millisecond.

[Mizuchi and Inamura, 2017] Mizuchi, Y. and Inamura, T. (2017). Cloud-based multimodal human-robot interaction simulator utilizing ros and unity frameworks. pages 948–955.

[Mizuchi and Inamura, 2018] Mizuchi, Y. and Inamura, T. (2018). Evaluation of human behavior difference with restricted field of view in real and vr environments. pages 196–201.

[Moll, 2017] Moll, T. (2017). SEK Brandenburg Erhielt Achleitner HMV-Survivor. https://sek-einsatz.de/spezialeinheiten-intern/sek-brandenburg-erhielt-achleitner-hmv-survivor-i/20441, publisher = SENews,.

[Moreno and Mayer, 1999] Moreno, R. and Mayer, R. (1999). Cognitive principles of multimedia learning: The role of modality and contiguity. *Journal of Educational Psychology*, 91:358–368.

[Mousavi et al., 1995] Mousavi, S. Y., Low, R., and Sweller, J. (1995). Reducing cognitive load by mixing auditory and visual presentation modes.

[Murai and Yotsumoto, 2018] Murai, Y. and Yotsumoto, Y. (2018). Optimal multisensory integration leads to optimal time estimation. *Scientific Reports*, 8.

[Nieuwenstein et al., 2009] Nieuwenstein, M. R., Potter, M. C., and Theeuwes, J. (2009). Unmasking the Attentional Blink. *Journal of Experimental Psychology, Human Perception and Performance*, 35(1):159–169.

[Norman, 2010] Norman, D. (2010). The way i see it: Natural user interfaces are not natural. *interactions*, 17:6–10.

[Novikova et al., 2015] Novikova, J., Watts, L., and Inamura, T. (2015). Modeling human-robot collaboration in a simulated environment. pages 181–182.

[Patil et al., 2017] Patil, D., Xie, M., and Velamala, S. (2017). Development of ros-based gui for control of an autonomous surface vehicle.

[Receveur and Fingscheidt, 2014] Receveur, S. and Fingscheidt, T. (2014). On likelihood histogram equalization for multimodal automatic speech recognition.

[Rogoway, 2018] Rogoway, T. (2018). Marine Corps Considers Deploying Small Drones at Squad Level.

[Rossi et al., 2013] Rossi, S., Leone, E., Fiore, M., Finzi, A., and Cutugno, F. (2013). An extensible architecture for robust multimodal human-robot communication. *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2208–2213.

[Sarter and Woods, 1995] Sarter, N. and Woods, D. (1995). How in the world did we ever get into that mode? mode error and awareness in supervisory control. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 37:5–19.

[Schrödinger, 1920] Schrödinger, E. (1920). *Theorie der Pigmente von grösster Leuchtkraft*. Annalen der Physik 4. Wiley-VCH.

[Shah, 2018] Shah, M. (2018). Survey on causes of motion sickness in virtual reality. pages 1–5.

[Sommer et al., 2001] Sommer, W., Leuthold, H., and Schubert, T. (2001). Multiple Bottlenecks in Information Processing? An Electrophysiological Examination. *Psychonomic bulletin & review*, 8 1:81–8.

[Song, 2015] Song, W. (2015). End-to-end deep neural network for automatic speech recognition. Stanford University.

[Strong, 2019] Strong, A. (2019). Unity building assets. `https://assetstore.unity.com/publishers/28542`, publisher = Unity.

[Torta et al., 2013] Torta, E., Cuijpers, R., and Juola, J. (2013). Design of a Parametric Model of Personal Space for Robotic Social Navigation. *International Journal of Social Robotics*, 5(3):357–365.

[Traumberuf, 2019] Traumberuf (2019). Berufsbild-Polizist Spezialeinsatzkommando. `http://www.traumberuf-polizei.de/berufsbild-polizist/spezialeinsatzkommando/`, publisher = Traumberuf,.

[Turner and Engle, 1989] Turner, M. L. and Engle, R. W. (1989). Is Working Memory Capacity Task Dependent? *Journal of Memory and Language*, 28.

[van der Horst, 2004] van der Horst, R. (2004). Occlusion as a measure for visual workload: An overview of tno occlusion research in car driving. *Applied ergonomics*, 35:189–96.

[van Wanrooij et al., 2009] van Wanrooij, M., Bell, A., Munoz, D., and Opstal, J. (2009). The effect of spatial–temporal audiovisual disparities on saccades in a complex scene. *Experimental Brain Research*, 198:425–437.

[Wang et al., 2016] Wang, J., Yang, Y., Mao, J., Huang, Z., Huang, C., and Xu., W. (2016). CNN-RNN: A Unified Framework for Multi-label Image Classification. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2285–2294.

[Ward, 2017] Ward, W. (2017). Hidden Markov Models in Speech Recognition.

[Witmer and Slater, 1999] Witmer, A. and Slater, M. (1999). Measuring presence: A response to the witmer and singer presence questionnaire. *Presence (Camb.)*, 8.

Appendices

Appendix A: Abbreviations

AIST	Advanced Institute of Science and Technology
ASR	Automatic Speech Recognition
CAPCOM	Capsule Communication Officer
CLT	Cognitive Load Theory
CNN	Convolutional Neural Network
DL	Deep-Learning
EVA	Extravehicular activity
FOV	Field of View
GCS	Ground Control Station
HMD	Head-Mounted Display
HMM	Hidden Markov Model
HUD	Heads-Up Display
InLaSeD	Indoor-Lageerkundung für Spezialeinheiten mit Drohnen (Indoor Situation Survey for Special Units with Drones)
IPD	Interpupillary distance
ISS	International Space Station
NASA-TLX	NASA-Task Load Index
NEEMO	NASA Extreme Environment Mission Operation
NATO	North Atlantic Treaty Organization
OpenHRP	Open Architecture Humanoid Robotics Platform
OSPAN	Operation Span Task
PVT	Psychomotor Vigilance Task
QoL	Quality of Life
ROS	Robot Operating System
RNN	Recurrent Neural Network
SA	Situational Awareness
SAGAT	Situation Awareness Global Assessment Technique
SEK	Spezialeinsatzkommandos
SIGVerse	SocioIntelliGenesis (Uni-)Verse

UAV Unmanned Aerial Vehicles
UI User Interface
USMC United States Marine Corp

Appendix B: Specifications

This section contains the versions and models for the primary specifications.

Table 9.1: Host Machine Parameters.

Component	Version
Device	Acer Nitro AN515-52
OS	Windows 10 Home 10.0.18362
CPU	i5-8300H @ 2.39 GHz (8x)
Instruction Set	x86-64 - INT64
Memory	8192MB RAM
Graphics	NVIDIA GeForce GTX 1050
Graphics2	Intel UHD Graphix 630 Adapter
DirectX	DirectX12

Table 9.2: Host Machine Software.

Program	Version
Blender	2.80
Unity	2019.2.0f
Mixamo	Browser (2019)
SteamVR	1.7.15
DirectX	DirectX12
Visual Studio	VS2015 Pro

Table 9.3: Virtual Machine Parameters.

Component	Version
VM Client	VMware Workstation 15 Player
OS	Ubuntu 16.04 LTS
CPU	vCPU - ESXi (2x)
Instruction Set	x86-64 - AMD64
Memory	6276MB RAM (Memory Swapping)
Graphics	NVIDIA (Passthrough/Accelerate 3D graphics)
Graphics Memory	2000MB

Table 9.4: Virtual Machine Software.

Program	Version
ROS	Kinetic Kame
SigVerse	3.0
PocketSphinx Stack	5 pre-alpha
Gazebo	7.15.0 (Prototyping Only)

Table 9.5: Additional Hardware.

Device	Model/Version
HTC Vive Pro	99HANW00100
Headset	Sony MDR-ZX220BT (Prototyping Only)
PS3 Controller	Sony CECHZC2U (Prototyping Only)
Drone	Parrot AR.Drone 2.0 (v2.4.8) (Prototyping Only)

Appendix C: Validation/Physical System Architecture

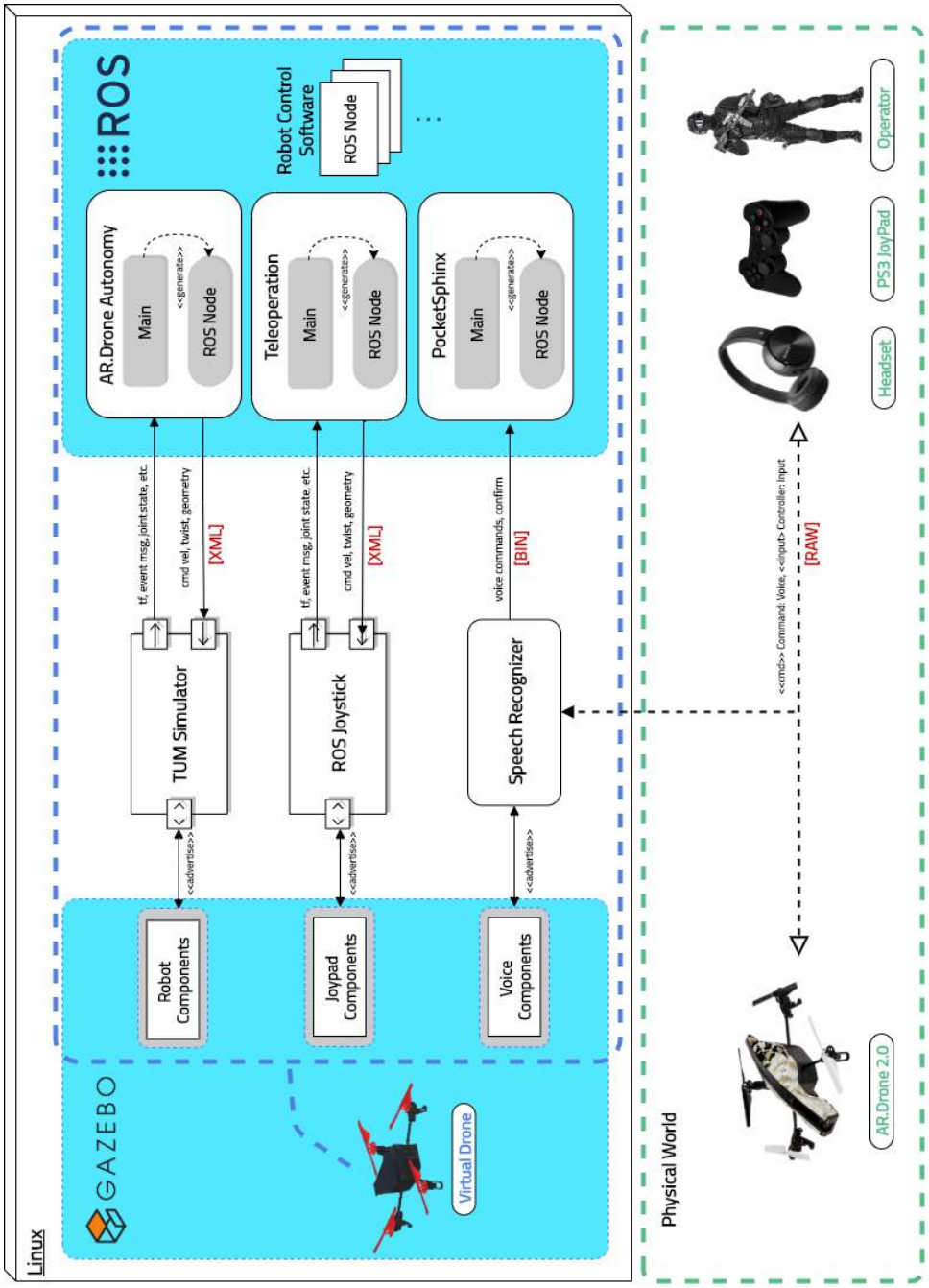


Figure 9.2: Validation/Physical System Architecture (Prototyping Only).

Appendix D: Unity Models

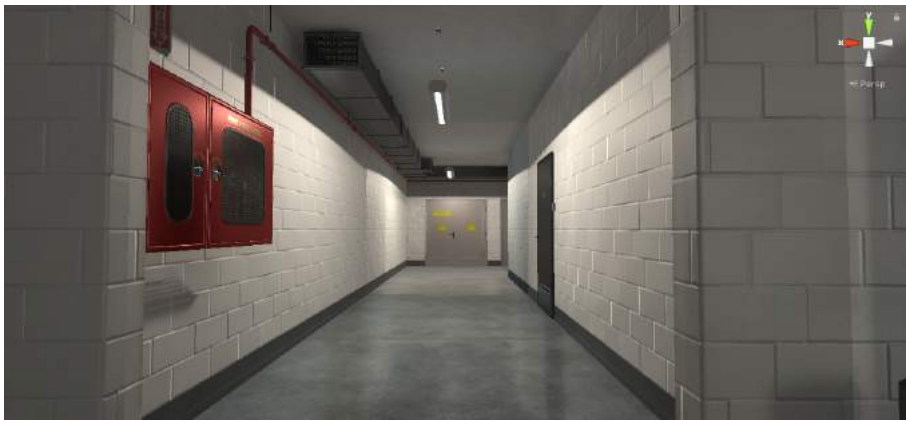


Figure 9.3: Hallway (Unity)

Figure 9.4: Stairwell (Unity)

Figure 9.5: Control Room (Unity)

Figure 9.6: Initial User Scene (Unity)

Figure 9.7: Lobby Interior (Unity)

Figure 9.8: Drone with Spark Effects (Unity)

Appendix D-2: Environmental Effects

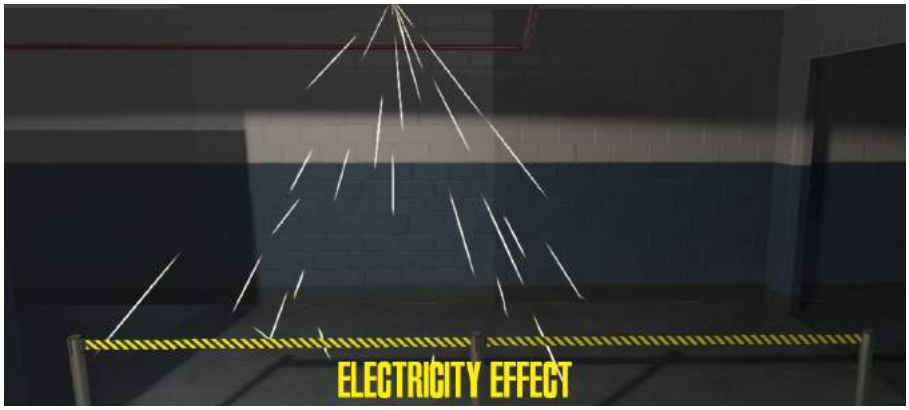


Figure 9.9: Environmental Effect: Electricity

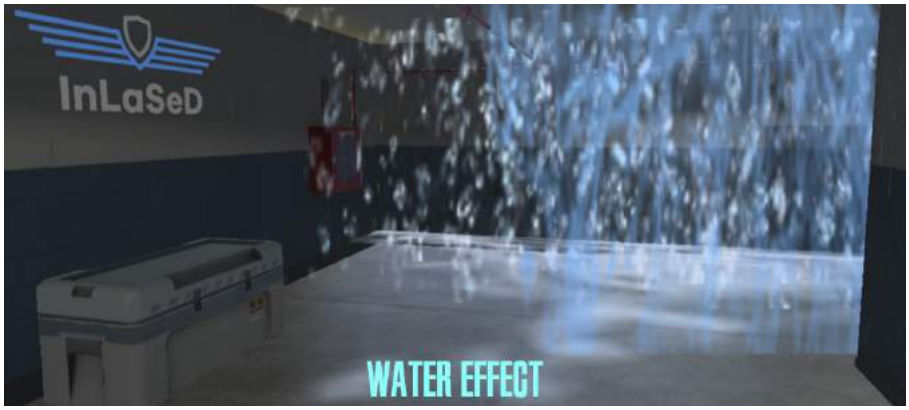


Figure 9.10: Environmental Effect: Water

Figure 9.11: Environmental Effect: Fire

Appendix D-3: Objects of Interest

Figure 9.12: Objects of Interest: robotSphere

Figure 9.13: Objects of Interest: HSK

Figure 9.14: Objects of Interest: Mech

Appendix D-4: Operator Models

Figure 9.15: Images of an Operative (model) with increasing functionality (left to right, top to bottom).

Appendix D-5: Drone Models

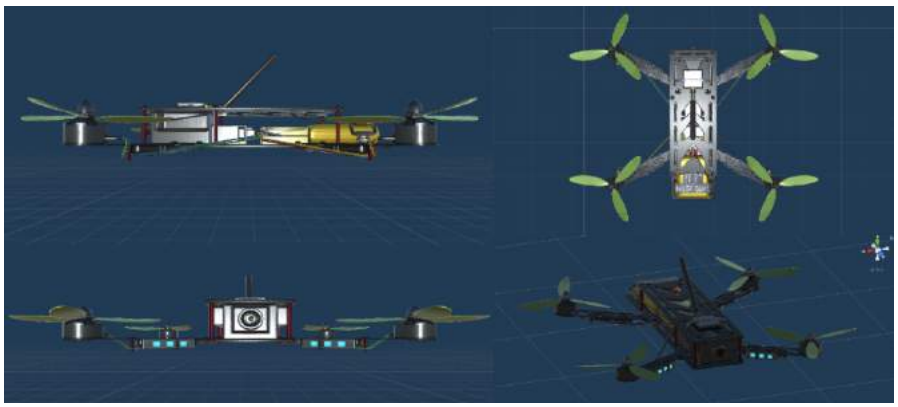


Figure 9.16: Images of Custom Drone from various perspectives (from left to right, top to bottom: side view, top view, front view, off-angled view.

Appendix F: A-PVT Data Analysis

This section contains the data values and analysis for the A-PVT. Fig. 9.18, Fig 9.19, and Fig 9.20 show the response time and total success button deactivations (*hit*). Fig. 9.17 shows the results of the pre-PVT test.

The PVT test has a SRT component in which users report on four subjective metrics including their current *Mental Awareness*, *Physical Energy*, *Stress*, and *Tiredness*. While the A-PVT did not require analysis of these values, they were collected and have been provided for completeness. Evaluating subjective reports from the SRT requires the test be taken several times with the same user to evaluate how changes in their reported values correlates with their performance on the *button click test*. As user's for this study only have one session, it is not possible to derive meaningful results at this time.

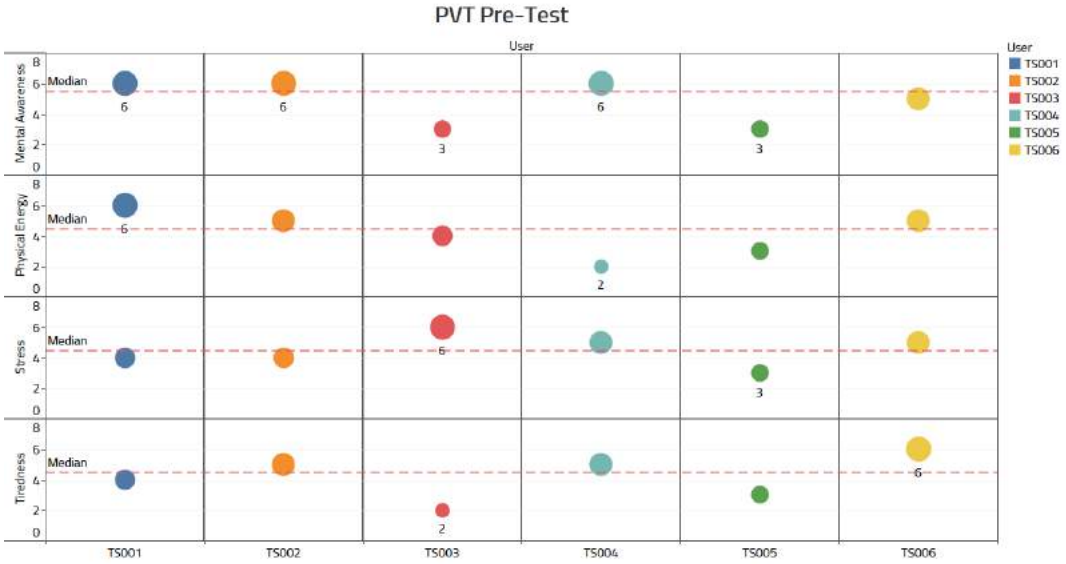


Figure 9.17: Results of PVT pre-test.

User	Test					
TS001	VOX		VIZ		MM	
Count	Time	Hit	Time	Hit	Time	Hit
1	0.52	1	1.11	1	1.56	1
2	19.84	0	10.96	0	1.40	1
3	2.11	1	1.78	1	1.17	1
4	1.11	1	5.71	1	1.23	1
5	1.60	1	0.85	1	3.00	1
6	1.16	1	2.18	1	0.92	1
7	1.20	1	2.70	1	4.90	1
8	2.95	1	11.42	0	1.00	1
9	1.97	1	1.28	1	1.78	1
10	1.08	1	1.50	1	1.43	1
11	1.27	1	0.80	1	1.31	1
12	1.04	1	0.86	1	1.15	1
13	3.23	1	-	-	3.40	1
14	1.22	1	-	-	1.27	1
15	1.27	1	-	-	1.09	1
16	1.12	1	-	-	1.00	1
17	1.39	1	-	-	1.36	1
18	1.17	1	-	-	0.99	1
19	1.74	1	-	-	1.53	1
20	0.94	1	-	-	-	-

User	Test					
TS002	VOX		VIZ		MM	
Count	Time	Hit	Time	Hit	Time	Hit
1	15.54	0	1.70	1	1.79	1
2	0.71	1	9.56	1	0.78	1
3	2.28	1	3.15	1	1.69	1
4	16.24	0	21.65	0	1.54	1
5	9.62	1	2.62	1	1.00	1
6	8.41	1	10.52	0	3.84	1
7	1.06	1	0.82	1	12.10	0
8	1.70	1	21.02	0	1.09	1
9	24.71	0	0.85	1	12.44	0
10	1.11	1	1.95	1	2.17	1
11	6.17	1	2.63	1	4.36	1
12	6.03	1	0.81	1	5.07	1
13	11.34	0	1.07	1	1.85	1
14	1.92	1	0.93	1	5.09	1
15	6.46	1	0.66	1	4.22	1
16	2.94	1	2.36	1	11.29	0
17	1.78	1	5.25	1	3.85	1
18	17.46	0	1.89	1	11.19	0
19	4.00	1	-	-	27.50	0
20	1.04	1	-	-	9.68	1
21	-	-	-	-	7.93	1

Figure 9.18: TS001 and TS002 Response Times and Accuracy.

User	Test					
TS003	VOX		VIZ		MM	
Count	Time	Hit	Time	Hit	Time	Hit
1	0.57	1	2.59	1	1.32	1
2	4.20	1	6.06	1	0.84	1
3	3.70	1	3.63	1	1.47	1
4	3.64	1	8.41	1	3.48	1
5	2.37	1	1.14	1	1.75	1
6	1.54	1	1.01	1	7.04	1
7	1.73	1	0.65	1	1.50	1
8	5.30	1	0.80	1	4.19	1
9	3.76	1	13.91	0	0.84	1
10	1.00	1	0.89	1	0.94	1
11	1.78	1	0.65	1	1.66	1
12	1.79	1	1.68	1	6.64	1
13	1.95	1	2.85	1	7.43	1
14	5.00	1	6.84	1	3.29	1
15	0.90	1	0.62	1	5.01	1
16	1.23	1	1.75	1	2.08	1
17	4.51	1	6.65	1	6.03	1
18	3.54	1	0.94	1	2.38	1
19	4.49	1	-	-	-	-
20	9.75	1	-	-	-	-
21	1.03	1	-	-	-	-
22	1.01	1	-	-	-	-
23	2.15	1	-	-	-	-
24	30.71	0	-	-	-	-
25	1.68	1	-	-	-	-
26	2.96	1	-	-	-	-
27	5.27	1	-	-	-	-

User	Test					
TS004	VOX		VIZ		MM	
Count	Time	Hit	Time	Hit	Time	Hit
1	0.82	1	-	-	1.76	1
2	17.01	0	-	-	2.48	1
3	1.50	1	-	-	25.39	0
4	3.90	1	-	-	6.01	1
5	5.66	1	-	-	49.72	0
6	1.39	1	-	-	8.63	1
7	1.12	1	-	-	18.92	0
8	17.22	0	-	-	1.39	1
9	1.53	1	-	-	4.73	1
10	1.69	1	-	-	2.32	1
11	1.16	1	-	-	1.64	1
12	2.82	1	-	-	1.46	1
13	1.52	1	-	-	2.45	1
14	9.42	1	-	-	2.64	1
15	10.26	0	-	-	1.57	1
16	1.20	1	-	-	14.05	0
17	0.92	1	-	-	2.72	1
18	1.41	1	-	-	3.23	1
19	0.88	1	-	-	3.61	1
20	5.16	1	-	-	-	-
21	4.71	1	-	-	-	-
22	27.53	0	-	-	-	-
23	1.32	1	-	-	-	-
24	1.56	1	-	-	-	-
25	6.39	1	-	-	-	-
26	4.91	1	-	-	-	-
27	1.31	1	-	-	-	-

Figure 9.19: TS003 and TS004 Response Times and Accuracy.

User	Test					
TS005	VOX		VIZ		MM	
Count	Time	Hit	Time	Hit	Time	Hit
1	9.43	1	27.98	0	5.46	1
2	10.54	0	9.44	1	1.27	1
3	5.09	1	2.54	1	3.29	1
4	1.84	1	3.38	1	12.90	0
5	23.41	0	1.80	1	2.91	1
6	3.95	1	22.72	0	2.32	1
7	8.22	1	2.71	1	1.46	1
8	19.38	0	1.21	1	1.25	1
9	-	-	1.81	1	3.81	1
10	-	-	1.88	1	7.46	1
11	-	-	1.83	1	3.98	1
12	-	-	1.19	1	3.94	1
13	-	-	2.92	1	0.50	1
14	-	-	2.56	1	0.74	1
15	-	-	1.71	1	3.81	1
16	-	-	7.95	1	8.61	1
17	-	-	1.16	1	8.96	1
18	-	-	2.02	1	3.34	1
19	-	-	1.65	1	1.58	1
20	-	-	2.60	1	1.26	1
21	-	-	-	-	4.79	1
22	-	-	-	-	3.30	1
23	-	-	-	-	1.58	1
24	-	-	-	-	9.07	1
25	-	-	-	-	2.21	1

User	Test					
TS006	VOX		VIZ		MM	
Count	Time	Hit	Time	Hit	Time	Hit
1	-	-	1.95	1	0.72	1
2	-	-	3.74	1	4.47	1
3	-	-	34.72	0	10.84	0
4	-	-	2.13	1	1.46	1
5	-	-	7.28	1	4.50	1
6	-	-	1.23	1	10.02	0
7	-	-	3.13	1	28.21	0
8	-	-	10.29	0	6.32	1
9	-	-	6.49	1	10.13	0
10	-	-	13.11	0	1.64	1
11	-	-	11.52	0	6.29	1
12	-	-	9.50	1	29.73	0
13	-	-	3.10	1	10.11	0
14	-	-	6.23	1	9.19	1
15	-	-	-	-	12.37	0

Figure 9.20: TS005 and TS006 Response Times and Accuracy.

Appendix G: SAGAT Data Analysis

This section contains the data values and analysis for the SAGAT. Fig. 9.21, Fig. 9.23, and Fig. 9.24 show user results for reporting the *Number* and *Position* of *Operatives*, *Environmental Elements*, and *Objects of Interest* (O.O.I.), in each scene, respectively. Fig. 9.22 shows the results of users recalling the locations of the *Waypoints*.

			Number of Operatives			Position of Operatives		
User	Test	Order	# Oper	Actual #	% Corr	Pos Oper	Actual Pos	% Corr
TS001	VOX	3	3	3	1	C6,C7	B1,C6,D7	0.66
	VIZ	2	5	5	1	B1,C6,D8	B1,G1,D7,C6,F4	0.6
	MM	1	3	4	0.666	B1,C6,C9	G1,F4,F9,C6	0.4
TS002	VOX	2	3	3	1	C7,C5,A2	B1,C6,D7	1
	VIZ	3	5	5	1	C5,C7,F4,I1,C1	B1,G1,D7,C6,F4	0.9
	MM	1	4	4	1	G9,C5,G4,G1	G1,F4,F9,C6	1
TS003	VOX	1	3	3	1	C5,C1,C8	B1,C6,D7	1
	VIZ	3	5	5	1	C8,C5,F4,C1,F1	B1,G1,D7,C6,F4	1
	MM	2	4	4	1	C5,F4,G1,G9	G1,F4,F9,C6	1
TS004	VOX	2	3	3	1	C6,C8,C1	B1,C6,D7	1
	VIZ	1	3	5	0.333	C1,G2,C3	B1,G1,D7,C6,F4	0.4
	MM	3	3	4	0.666	C5,G1,C1	G1,F4,F9,C6	0.5
TS005	VOX	1	3	3	1	C5,D8,C1	B1,C6,D7	1
	VIZ	3	4	5	0.666	F4,C6,C1,F2	B1,G1,D7,C6,F4	0.8
	MM	2	4	4	1	C5,C9,C1,F3	G1,F4,F9,C6	0.375
TS006	VOX	1	3	3	1	B8,C6	B1,C6,D7	0.66
	VIZ	2	4	5	0.666	B8,C6,A1,I1	B1,G1,D7,C6,F4	0.7
	MM	3	5	4	0.666	B8,C6,A1,I1	G1,F4,F9,C6	0.375

Figure 9.21: SAGAT Results: Number and Position of Operatives.

			Position of Waypoint		
User	Test	Order	Pos Oper	Actual Pos	% Corr
TS001	VOX	3	C6,C9	F5,D3,C2	0
	VIZ	2	E5,C2,F2	E5,C2,F2	1
	MM	1	C2,F2,E5	D5,C2,G1	1
TS002	VOX	2	F5,F3,D2	F5,D3,C2	0.8325
	VIZ	3	E5,G3,D2	E5,C2,F2	1
	MM	1	C6,C2,G1	D5,C2,G1	1
TS003	VOX	1	F5,C3,F2	F5,D3,C2	0.666
	VIZ	3	E4,B3,F2	E5,C2,F2	1
	MM	2	D5,C2,G1	D5,C2,G1	1
TS004	VOX	2	F4,E3,C2	F5,D3,C2	1
	VIZ	1	E3,B2,F2	E5,C2,F2	1
	MM	3	D5,C2,G1	D5,C2,G1	1
TS005	VOX	1	F5,E4,C1	F5,D3,C2	1
	VIZ	3	E5,B2,F2	E5,C2,F2	1
	MM	2	D6,B2,F1	D5,C2,G1	1
TS006	VOX	1	G6,E2,B2	F5,D3,C2	1
	VIZ	2	E6,C2,H1	E5,C2,F2	0.8325
	MM	3	A3,E6,F2	D5,C2,G1	0.8325

Figure 9.22: SAGAT Results: Location of Waypoints.

			Environmental Type			Environmental Position		
User	Test	Order	Env Type	Actual Env	% Corr	Pos Env	Actual Pos	% Corr
TS001	VOX	3	Electricity	Electricity	1	I2	A2,B7,J1	0.33
	VIZ	2	-	Water	1	-	H6-H9	0
	MM	1	Fire	Fire	1	B1,H1	B1-D1, H1-K1	1
TS002	VOX	2	Electricity	Electricity	1	K2,B2	A2,B7,J1	0.66
	VIZ	3	Water	Water	1	H6	H6-H9	1
	MM	1	Fire	Fire	1	I1	B1-D1, H1-K1	0.5
TS003	VOX	1	Electricity	Electricity	1	A7,A2,I1	A2,B7,J1	1
	VIZ	3	Water	Water	1	H5-H9	H6-H9	1
	MM	2	Fire	Fire	1	C1-E1,H1-K3	B1-D1, H1-K1	1
TS004	VOX	2	Electricity	Electricity	1	on the roof	A2,B7,J1	0
	VIZ	1	Fire	Water	0	-	H6-H9	0
	MM	3	Fire	Fire	1	K1,E0	B1-D1, H1-K1	1
TS005	VOX	1	Electricity	Electricity	1	B5	A2,B7,J1	1
	VIZ	3	Water	Water	1	H7	H6-H9	1
	MM	2	Fire	Fire	1	H1	B1-D1, H1-K1	0.5
TS006	VOX	1	-	Electricity	0	8B,6C	A2,B7,J1	0.66
	VIZ	2	Water	Water	1	H7	H6-H9	1
	MM	3	Fire	Fire	1	G3	B1-D1, H1-K1	0

Figure 9.23: SAGAT Results: Number and Position of Environmental Elements.

			Object of Interest Type			Object of Interest Position		
User	Test	Order	OOI	Actual OOI	% Corr	Pos OOI	Actual Pos	% Corr
TS001	VOX	3	Robot	Robot	1	H7	H6	1
	VIZ	2	Robot	Robot	1	A3	A3	1
	MM	1	Robot	Robot,Drone	0.5	G3	G3,B2	0.5
TS002	VOX	2	Robot	Robot	1	6H	H6	1
	VIZ	3	Robot	Robot	1	B3	A3	1
	MM	1	Robot	Robot,Drone	0.5	G3	G3,B2	0.5
TS003	VOX	1	Robot	Robot	1	H6	A3	0
	VIZ	3	Robot	Robot	1	A3	H6	0
	MM	2	Robot,Drone	Robot,Drone	1	G3,B2	G3,B2	1
TS004	VOX	2	-	Robot	0	C8	H6	0
	VIZ	1	-	Robot	0	G2	A3	0
	MM	3	-	Robot,Drone	0	E1	G3,B2	0
TS005	VOX	1	-	Robot	0	-	H6	0
	VIZ	3	Robot	Robot	1	3A	A3	1
	MM	2	Robot,Drone	Robot,Drone	0	3G,2A	G3,B2	1
TS006	VOX	1	Robot	Robot	1	H6	H6	1
	VIZ	2	Robot	Robot	1	A3	A3	1
	MM	3	-	Robot,Drone	0	-	G3,B2	0

Figure 9.24: SAGAT Results: Number and Position of Objects of Interest.

Appendix H: User Survey

InLaSeD - VR Drone User Testing

This test will be used to identify a brief objective measure of cognitive function and subjective self-evaluations of performance during VR testing. The user will complete three different VR Scenes. There are two primary goals for the user:

1. Complete the "game" by commanding the drone through all three waypoints.
2. Click the "button" every time it turns green.

These criteria are equally important and should both be satisfied with the best of the user's ability.

The user also has a sub-goal of being aware of the environment, reporting the locations of "objects of interests." These may include other operatives, environmental effects (like fire), or other "out of place" items. Please try to remember the approximate solutions. A 2-D map will be provided for you to list the grid coordinates.

There are eight sections (which occur depending on the random order of testing). All users will begin with a pre-test. The pre-test will be administered prior to the NASA-TLX and SAGAT tests for the InLaSeD Drone Testing Environment and is used for personal data and subjective self-assessment collection prior to entering the VR environment.

Sections 2 through 7 include a NASA-TLX and SAGAT test for each scene. Please stop between each section and wait for the researcher. The final section will ask you to rank your preference for the method of receiving confirmations from the drone.

If you have any questions, please notify the researcher. If you feel sick or dizzy at any time, please stop immediately and notify the researcher. Good luck and have fun!

* Required

1. **Do you consent to allow your information to be analyzed by only the researcher and his direct supervisor?** *
   Your information will never be made publicly available. Only screen recordings on the environment (including your voice) will be kept. Your information will be anonymized. Your personal details including age, gender, and ethnicity, will not be used in evaluating any of your results from the experiment. You are allowed access to your screen captures per request and they will be provided physically on a flash drive. You may request that your information is deleted at any time.
   Mark only one oval.
   - ◯ Yes
   - ◯ No
2. **Name (can be a number)** *
3. **Email**
4. **Age**
5. **Ethnicity**

Figure 9.25: User Test Form - Page 1.

6. **Gender**
Mark only one oval.

- Female
- Male
- Prefer not to say
- Other:

7. **Field of Study?**

Self-Evaluation (Pre-Test)

This test allows the user to evaluate their current performance metrics based on their feeling.

Based on the NASA-PVT:

https://www.nasa.gov/mission_pages/station/research/experiments/explorer/Investigation.html?#id=955

https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20090029851.pdf

8. **How tired are you?** *
Mark only one oval.

	1	2	3	4	5	6	7	
Tired	◯	◯	◯	◯	◯	◯	◯	Fresh, Ready to go

9. **How mentally aware are you?** *
Mark only one oval.

	1	2	3	4	5	6	7	
Mentally fatigued	◯	◯	◯	◯	◯	◯	◯	Mentally sharp

10. **How much physical energy do you have?** *
Mark only one oval.

	1	2	3	4	5	6	7	
Physically exhausted	◯	◯	◯	◯	◯	◯	◯	Energetic

11. **How much stressed do you feel?** *
Mark only one oval.

	1	2	3	4	5	6	7	
Totally stressed	◯	◯	◯	◯	◯	◯	◯	Not stressed at all

12. **Please complete the Scene before proceeding.**
Please do not hit "Ready." until instructed by the researcher.
Mark only one oval.

- Ready.

Skip to question 13.

NASA-TLX - Audio Only

This is a qualitative test that evaluates the users personal feelings with regard to the experiment. Please answer directly as there is no "right" or "wrong" answer. These results will be compared between users and with respect to the PVT and SAGAT tests to find correlations between the subjective self-reporting of

Figure 9.26: User Test Form - Page 2.

performance and quantitative results.

https://humansystems.arc.nasa.gov/groups/TLX/downloads/TLXScale.pdf

13. **How mentally demanding was the task?** *
Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
Very Low	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Very High

14. **How physically demanding was the task?** *
Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
Very Low	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Very High

15. **How hurried or rushed was the pace of the task?** *
Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
Very Low	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Very High

16. **How successful were you in accomplishing what you were asked to do?** *
Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
Very Low	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Very High

17. **How hard did you have to work to accomplish your level of performance?** *
Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
Very Low	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Very High

18. **How insecure, discouraged, irritated, stressed, and annoyed were you?** *
Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
Very Low	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Very High

Figure 9.27: User Test Form - Page 3.

19. **What are the reasons the test might be challenging?**

Check all that apply.

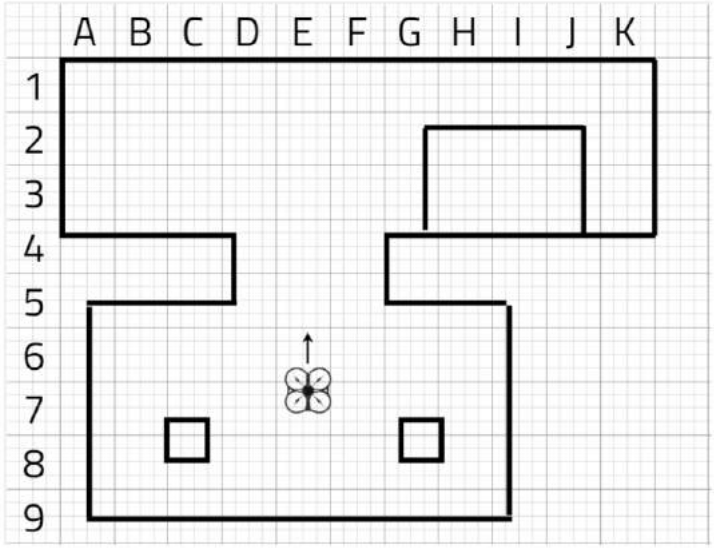
- [ ] Too many tasks
- [ ] User unfocused
- [ ] Environmental issues (noise, objects)
- [ ] Fatigue
- [ ] Issues with equipment/testing
- [ ] Unclear instructions
- [ ] Physical pain/load
- [ ] Sickness caused by VR environment
- [ ] Other: _____

SAGAT - Audio

This test will evaluate your situational awareness and potentially, working memory. Please try to recall details from the experiment to the best of your ability. If you do not remember the precise locations, please approximate.

https://ntrs.nasa.gov/search.jsp?R=20000109865

Environment Map (Top-View)



20. **How many operatives were there?** *

Mark only one oval.

1	2	3	4	5	6	7
◯	◯	◯	◯	◯	◯	◯

21. **Where were the operatives?**

Please enter the approximate coordinates.

_____

Figure 9.28: User Test Form - Page 4.

22. **Where were the waypoints (goals)?**
Please enter the approximate coordinates.

23. **What environmental effects were present?** *
Mark only one oval.

- ◯ Fire
- ◯ Water
- ◯ Electricity
- ◯ Wait... what?
- ◯ Other: _____

24. **Where were the environmental effects?**
Please enter the approximate coordinates.

25. **What and where was the "Out of Place" item? (thing, coordinates)**

26. **Please complete the Scene before proceeding.**
Please do not hit "Ready." until instructed by the researcher.
Mark only one oval.

- ◯ Ready.

NASA-TLX - Visual Only

This is a qualitative test that evaluates the users personal feelings with regard to the experiment. Please answer directly as there is no "right" or "wrong" answer. These results will be compared between users and with respect to the PVT and SAGAT tests to find correlations between the subjective self-reporting of performance and quantitative results.

https://humansystems.arc.nasa.gov/groups/TLX/downloads/TLXScale.pdf

27. **How mentally demanding was the task?** *
Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
Very Low	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Very High

28. **How physically demanding was the task?** *
Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
Very Low	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Very High

29. **How hurried or rushed was the pace of the task?** *
Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
Very Low	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Very High

Figure 9.29: User Test Form - Page 5.

30. **How successful were you in accomplishing what you were asked to do?** *

Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
Very Low	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Very High

31. **How hard did you have to work to accomplish your level of performance?** *

Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
Very Low	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Very High

32. **How insecure, discouraged, irritated, stressed, and annoyed were you?** *

Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
Very Low	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Very High

33. **What are the reasons the test might be challenging?**

Check all that apply.

- ☐ Too many tasks
- ☐ User unfocused
- ☐ Environmental issues (noise, objects)
- ☐ Fatigue
- ☐ Issues with equipment/testing
- ☐ Unclear instructions
- ☐ Physical pain/load
- ☐ Sickness caused by VR environment
- ☐ Other: _____

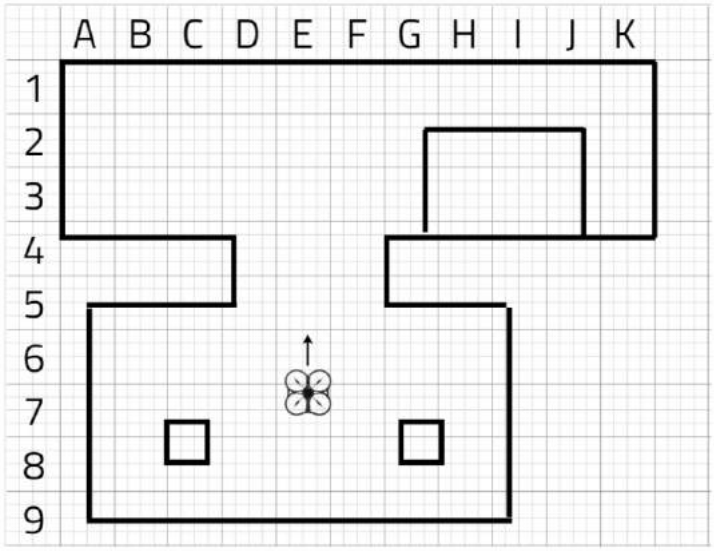
SAGAT - Visual

This test will evaluate your situational awareness and potentially, working memory. Please try to recall details from the experiment to the best of your ability. If you do not remember the precise locations, please approximate.

https://ntrs.nasa.gov/search.jsp?R=20000109865

Environment Map (Top-View)

Figure 9.30: User Test Form - Page 6.



34. **How many operatives were there?** *
Mark only one oval.

1 2 3 4 5 6 7

35. **Where were the operatives?**
Please enter the approximate coordinates.

36. **Where were the waypoints (goals)?**
Please enter the approximate coordinates.

37. **What environmental effects were present?** *
Mark only one oval.

- Fire
- Water
- Electricity
- Wait... what?
- Other:

38. **Where were the environmental effects?**
Please enter the approximate coordinates.

39. **What and where was the "Out of Place" item? (thing, coordinates)**

Figure 9.31: User Test Form - Page 7.

40. **Please complete the Scene before proceeding.**
Please do not hit "Ready." until instructed by the researcher.
Mark only one oval.

◯ Ready.

NASA-TLX - Multimodal

This is a qualitative test that evaluates the users personal feelings with regard to the experiment. Please answer directly as there is no "right" or "wrong" answer. These results will be compared between users and with respect to the PVT and SAGAT tests to find correlations between the subjective self-reporting of performance and quantitative results.

https://humansystems.arc.nasa.gov/groups/TLX/downloads/TLXScale.pdf

41. **How mentally demanding was the task?** *
Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
Very Low	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Very High

42. **How physically demanding was the task?** *
Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
Very Low	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Very High

43. **How hurried or rushed was the pace of the task?** *
Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
Very Low	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Very High

44. **How successful were you in accomplishing what you were asked to do?** *
Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
Very Low	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Very High

45. **How hard did you have to work to accomplish your level of performance?** *
Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
Very Low	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Very High

46. **How insecure, discouraged, irritated, stressed, and annoyed were you?** *
Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
Very Low	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Very High

Figure 9.32: User Test Form - Page 8.

47. **What are the reasons the test might be challenging?**

Check all that apply.

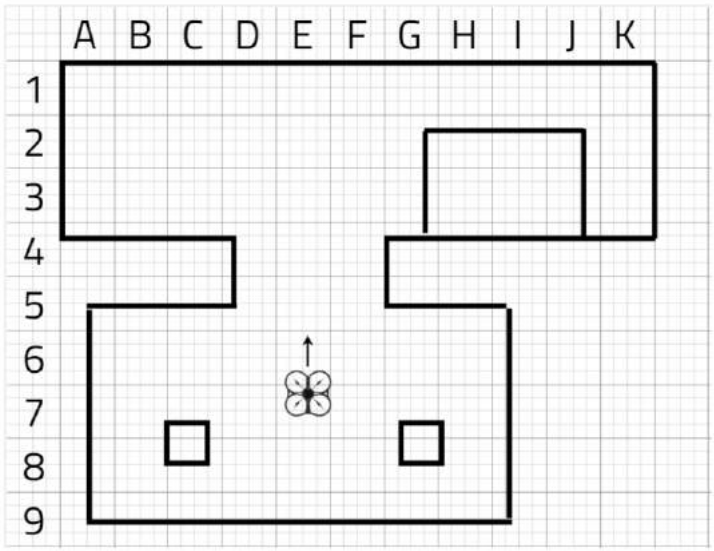
- [ ] Too many tasks
- [ ] User unfocused
- [ ] Environmental issues (noise, objects)
- [ ] Fatigue
- [ ] Issues with equipment/testing
- [ ] Unclear instructions
- [ ] Physical pain/load
- [ ] Sickness caused by VR environment
- [ ] Other: _____

SAGAT - Multimodal

This test will evaluate your situational awareness and potentially, working memory. Please try to recall details from the experiment to the best of your ability. If you do not remember the precise locations, please approximate.

https://ntrs.nasa.gov/search.jsp?R=20000109865

Environment Map (Top-View)



48. **How many operatives were there?** *

Mark only one oval.

1	2	3	4	5	6	7
◯	◯	◯	◯	◯	◯	◯

49. **Where were the operatives?**

Please enter the approximate coordinates.

_____

Figure 9.33: User Test Form - Page 9.

50. **Where were the waypoints (goals)?**
Please enter the approximate coordinates.

51. **What environmental effects were present?** *
Mark only one oval.

- ◯ Fire
- ◯ Water
- ◯ Electricity
- ◯ Wait... what?
- ◯ Other: _____

52. **Where were the environmental effects?**
Please enter the approximate coordinates.

53. **What and where was the "Out of Place" item? (thing, coordinates)**

54. **Please complete the Scene before proceeding.**
Please do not hit "Ready." until instructed by the researcher.
Mark only one oval.

- ◯ Ready.

Final Considerations

55. **Please rank the different models based on your preference:** *
Audio-only, Visual-only, and Multimodal

56. **Did you experience VR sickness?**

57. **Is there any other feedback you would like to provide?**
We would like to improve the environments and implementation as much as possible. Please provide any feedback that you are willing to share. Thank you for your participation.

Powered by
Google Forms

Figure 9.34: User Test Form - Page 10.