VMCC: A VIRTUAL REALITY FRAMEWORK FOR AUGMENTING MISSION CONTROL OPERATIONS

ESWAR ANANDAPADMANABAN
S.B., EECS, Massachusetts Institute of Technology (2019)

Submitted to the Department of Electrical Engineering and Computer Science in Partial Fulfillment of the Requirements for the Degree of Master of Engineering in Electrical Engineering and Computer Science at the Massachusetts Institute of Technology.

SEPTEMBER 2020

Author:

Eswar Anandapadmanaban
Department of Electrical Engineering and Computer Science
August 14th, 2020

Certified by:

Dava J. Newman
Apollo Professor of Aeronautics and Astronautics
Thesis Supervisor

Accepted By:

Katrina LaCurts
Chair, Master of Engineering Thesis Committee

VMCC: A VIRTUAL REALITY FRAMEWORK FOR AUGMENTING MISSION CONTROL OPERATIONS

ESWAR ANANDAPADMANABAN

Submitted to the Department of Electrical Engineering and Computer Science on August 14th, 2020 in Partial Fulfillment of the Requirements for the Degree of Master of Engineering in Electrical Engineering and Computer Science at the Massachusetts Institute of Technology

ABSTRACT

Mission control has played an integral part of NASA missions since the early space exploration days. As NASA aims to return back to the Moon and prepare for sending astronauts to Mars, the mission architectures are increasing in complexity. Our ground based mission control operators will be working with more data and tighter constraints, creating an evident need for new and improved tools. With the advent of Virtual Reality (VR), we can leverage this immersive medium to create better tools and software to do exactly that. In this thesis I present vMCC - Virtual Mission Control, which is a multi-user Virtual Reality mission control tool for data visualization. I present the system design, tools, and user interface built for vMCC. I discuss how vMCC can be a foundational platform for prototype concepts to be built and tested. Finally, I present what a vision for future operations can look like and provide directions for future work.

Thesis Supervisor:
Dava J. Newman
Title: Apollo Professor of Aeronautics and Astronautics

ACKNOWLEDGEMENTS

I want to sincerely thank my research supervisor - Professor Dava Newman - for the mentorship, guidance and support over the last 4 years since I started working on the Holo-SEXTANT project. Thank you for being supportive of my ideas and teaching me how to do proper research, publish papers, and connecting me with support whenever needed. I've been very fortunate to have an advisor who allowed me to follow my interests, and guided me to finding real world value from research.

Thank you to my Undergraduate Researcher Trent Piercy for working at my side on this project and being a sounding board for ideas. vMCC would not have been possible without our late nights debugging, testing, and writing code.

Thanks to the rest of the HSL grad students and community. Many of you helped test vMCC and provided feedback on ideas and concepts. Thanks to the NASA RESOURCE colleagues for providing data and context to the Virtual Mission Control Center.

Thanks to the VR/AR@MIT for being a community of curious creative technologists. I enjoyed all of the conversations, exchanges, and experiments with VR and AR at the building 9 Cave. Special shout out to Austin Edelman for always being around for a conversation, stirring up new ideas, and keeping me sane. Many thanks to all of the people who never hesitated to jump into a headset to test a new feature or interface. Thanks to my Reality Hack team that helped put on the world's largest VR/AR hackathon. Those 5 days were worth the 6 months of planning stress.

Big thanks to friends at MIT keeping life exciting and preventing work from taking over my life. Lastly, a big thank you to my family for lifelong support, especially in the final months of my work when I returned home due to Covid-19 and set up a lab-from-home to continue research.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

AR Augmented Reality

BASALT Biologic Analog Science Associated with Lava Terrains

CAD Computer Aided Design

COVID-19 CoronaVirus 2019

ConOps Concept of Operations

CSV Comma Separated Values

DoF Degrees of Freedom

EVA Extra-Vehicular Activity

FOV Field of View

HUD Heads Up Display

JPL Jet Propulsion Laboratory

JSC Johnson Space Center

MCC Mission Control Center

MSC Mission Support Center

MVP Mojaves Volatiles Prospector

UI User Interface

VoIP Voice over Internet Protocol

VR Virtual Reality

xGDS Exploration Ground Data System

1 INTRODUCTION

Human beings face ever more complex and urgent problems, and their effectiveness in dealing with these problems is a matter that is critical to the stability and continued progress of society. A human is effective not just because he applies to a problem, a high degree of native intelligence or physical strength (with a full measure of motivation and purposefulness), but also because he makes use of efficient tools, methods, and strategies.

— Douglas Engelbart, from Program on Human Effectiveness

The earliest pioneers of the computing industry envisioned the computer as the tool to augment human intellect and capabilities. Douglas Engelbart, one of these pioneers envisioned a vast information space, where intellectual workers would access "working stations" and work collectively to solve the most daunting problems. He envisioned computers guiding and assisting humans as they put forth the best usage of the tools for the highest degree of problem solving. Augmenting human intellect meant "more-rapid comprehension, better comprehension, the possibility of gaining a useful degree of comprehension in a situation that previously was too complex, speedier solutions, better solutions, and the possibility of finding solutions to problems that seemed insoluble" (Engelbart, 1962).

In the near 5 decades since Engelbart's concepts, the software and computing industry has transformed human productivity with tools of automation, visualization, and communication. Yet, the Augmented Human concept is still not at reach. The growth of data is exponential and we are looking for better tools to process this data and make decisions. Our data collection tools have gotten stronger, as have automated processing frameworks. Numbers no longer live just in spreadsheets as we move towards more dynamic visualizations and representations. Data-driven decisions are critical to solve challenges and problems that exist today.

Space exploration is one of these grand challenges. Human space exploration addresses the fundamental questions about our place in the Universe and satisfies an eternal curiosity. It is vital to the human spirit and perhaps our greatest endeavor as a species. The road to Mars and deep space exploration is daunting with a long list of problems and constraints. To accomplish these missions, augmenting human capabilities and information processing is paramount to success. NASA Mission Control is one of the most complex data environments already, but it will only become exponentially more complex for such missions. Being able to sift through data feeds and make real-time mission critical decisions can mean life or death. Our deep space explorers will be a small group of highly trained generalists in unknown environments faced with unprecedented challenges. Communication latencies make them isolated from depending on earth-based support from specialists. For example, Mars communication to Earth involves up to 40 minute round-trip speed-of-light latency between 2 messages. In such constraints, the explorers can use any and all augmentation available to help them navigate their environment. Teams of scientists, engineers, and specialists will be even more strained to help the explorers. A variety of questions come to mind when thinking about what this will look like:

1. What does the evolution of mission control look like to enable assisting explorers in such adverse conditions?
2. What technologies can be utilized to assist these operators?
3. How can we automate portions of the data analysis to aid in faster insights?
4. How can remote science-tactical experts collaborate with the co-located mission control team providing even more expertise?

In this thesis, I argue that virtual reality can help answer the second question. A tool built around the strengths of VR can increase situational awareness and decrease cognitive load for operators. Such a tool, if designed for collaboration and rapid analysis could help answer the remaining questions. Using modern virtual reality technology as a medium, simulations of scenarios, spatial data visualization, and virtual face-to-face communication are all possible.

1.1 VIRTUAL REALITY

Virtual reality technology has existed for decades since Ivan Sutherland introduced the world to the first 3D Head-Mounted Display (Sutherland, 1968). In the 1990s Head Mounted Display(HMD) VR devices were prohibitively expensive, difficult to use, bulky, and very nascent. Global internet was in its early days, PCs were primitive, and computing as we know it was fundamentally different. However, the past few years have represented a rebirth in the virtual reality industry. In 2016, the first set of mass-market virtual reality devices became available.

This was led by the Oculus Rift and HTC Vive consumer headsets

This was possible due to a culmination of advances in computer processing power, computer graphics pioneered by the gaming industry, 3D authoring workflows such as CAD (Computer Aided Design), display technologies, sensing/input technologies and more. Virtual reality was not just a concept technology anymore but an immersive, usable, product. Since 2016 a second and third wave of VR technology has arrived providing even more usability and features. Along with it have come improvements in firmware, software, applications, and design paradigms.

This combination of benefits allows developers to build immersive worlds that bring a new dimension to computing beyond what can be experienced through the window of a PC monitor. Ultimately, virtual reality brings our interactions with computers to resemble our interactions with the world, removing a layer of abstraction. Virtual Reality utilizes, as artist and engineer Myron Krueger holds, "the ultimate interface, the human body and human senses." (Kreuger, 1993) Removing this layer of abstraction, VR increases bandwidth of information transfer both in consumption and user input.

The obvious benefit to consider virtual reality for mission control is to provide the ability to visualize 3D data. Operators can visualize what the rover sees on Mars while analyzing the instrument samples at each sampling site. They can view multiple layers of data superimposed on top of each other. Effective use of the three dimensionality can dramatically increase the data intake bandwidth of a single operator. More-so, annotation and other authoring tools can create powerful communication workflows between users.

My work addresses how virtual reality could be utilized for hybrid systems that leverage the benefits of immersive computing for data visualization in the context of exploration. The Virtual Mission Control Center (vMCC) is a prototype exploration into augmenting human capabilities for a mission control setting. The vMCC system enables

Figure 1: Astronaut Buzz Aldrin using an HTC Vive VR headset. Image Credit:anthillonline.com

teams of users to remotely meet in VR to visualize and analyze data in a novel manner.

1.2 THESIS CONTRIBUTIONS

The goal of my work is to create a system that uses spatial capabilities of humans and our inherent three-dimensionality to enhance data visualization in a mission control setting. Such a system uses motion and parallax to bring data to life as if it were a real life sculpture. It allows the scientist to manipulate and work with data with their hands. It allows for them to communicate to others with body language and voice while drawing and annotating their data space. This system is a framework called vMCC - a multi-user virtual reality data visualization software. It can be used for training mission operators who do not have access to the equipment and architecture of a mission control. It can be used for rapid deployment in Earth analog missions for data analysis post-exploration or bringing in remote science-tactical specialists to share insights. Lastly, it can be used as a preliminary proof of concept prototype for virtual reality systems that may one day be used for flight missions and planetary EVAs.

I summarize my contributions in both the design and engineering realms as follows:

- A VR spatial framework for recreating mission control virtually
- Novel methods for interfacing with data
- Improved design of scientific communication
- Pipelines for integrated data analysis
- Unique representations of higher dimensional data
- Techniques to leverage existing tools to improve presence and understanding of users

1.3 THESIS OVERVIEW

This thesis is organized around the design and development decisions that were made in building vMCC. It begins with Chapter 2, where I describe related lines of work. First I analyze current mission control tools and analog missions to create a Concept of Operations (ConOps). I also present cognitive research motivating the benefits of virtual reality. With these benefits in mind, I present implementations of immersive technology in other projects and insights from these.

In Chapter 3, I walk through the vMCC system design. First, I provide an overview of the underlying architecture, followed by implementation of the vMCC software system.This involves libraries used, the data viewer modalities, and the tools available for the user.
In Chapter 4, I present the design principles and guiding heuristics behind the user interface as well as implementation of the interface.
In Chapter 5, I provide perspective on the final working prototype. This section also shares preliminary user feedback and thoughts on the scalability of the system in enhancing operations.
In Chapter 6, I conclude the major findings and summarize my contributions.

2 LITERATURE REVIEW

2.1 CONCEPT OF OPERATIONS

During the Apollo missions, at Houston Mission Control, operators were busy analyzing data and communicating with one other to provide ground-support for the astronauts. Over 800 sensors were transmitting data down through the NASCOM system on "high speed" 2.4 KBPS connections (Meigs and Stinet, 1970). Despite the constraints, the Apollo Mission Control was an icon of technological achievement. It is even labeled as a "cathedral of engineering" in the National Register of Historic Places (von Ehrenfield, 2018).

Figure 2: Recently restored mission control center at Johnson Space Center in Houston Image Credit: nytimes.com

NASA has conducted over 400 EVAs to date, however only nine have been conducted on planetary bodies with scientific objectives - the very same Apollo missions(J-class missions 15, 16, 17). Space exploration is very much in its infancy and our past experiences are far from enough to be prepared for the needs of future planetary missions to the Moon and Mars. These missions will be longer than any past human missions, with a proportional increase in data analysis and processing needs (NASA, 2005). One of NASA's approaches to designing new mission architectures is through "Analog Missions" - Earth bound missions that examine scientific, operational, and technological elements analogous to conditions on other planetary and deep space environments. (Lim *et al.*, 2019). Concept of Operations, or ConOps is the instantiation of design elements that guide the organization and flow of personnel, communications, hardware, software, and data products involved in a mission concept.

NASA currently has multiple analog missions underway to explore new operational concepts and hardware/software systems for science and exploration. With an emphasis on science, looking towards current analog missions illuminates many of the data requirements and needs of future exploration.

I will analyze two particular analog missions for their emphasis on performing real scientific investigation. The BASALT analog mission and MVP analog both stand out for their intensive field activities, tools developed, and insights in science, operations, and engineering. The BASALT (Biologic Analog Science Associated with Lava Terrains) Missions conducted scientific investigation into lava terrains as an analog to scientific operations astronauts might conduct on Mars. A focus of BASALT was to employ Mars-realistic communication bandwidths and latencies. The Mojave Volatiles Prospector (MVP) project was also a science-driven field program but focused on real-time robotic exploration of the moon in search of lunar volatiles (primarily water). This simulated rover mission to investigate composition and distribution of volatiles incorporated limited duration constraints as well as near real-time operations creating a rapid pace to maximize productivity. Both missions have roots in conducting real science, and generated a wealth of data. MVP, had temporal data processing constraints to make near real-time decisions making innovative uses of tools. BASALT had limited bandwidth in communication and time latencies which meant the science team was constrained to make decisions based on available data.

2.1.1 *Mojaves Volatiles Prospector (MVP)*

When looking into MVP, the planning, data analysis, and situational awareness decision support tools can be analyzed for insights. (Heldmann *et al.*, 2016a) The structure of MVP involved sending a lunar rover to navigate 3-5 km of terrain to examine numerous sites. Measurements were collected to (1) confirm the presence of volatiles, (2) quantify spatial distributional, amount, and accessibility of volatiles, (3) characterize the influence of topography, surface mineralogy and other factors on volatiles retention (Heldmann *et al.*, 2016a). MVP used two primary payload elements - Near Infrared Volatiles Spectrometer System (NIRVSS) and the Neutron Spectrometer System (NSS). Both of these provided the data for prospecting the terrain.

Figure 3: K-Rex Rover used in the Mojave field tests. Image Credit: Akash Arora, MVP

MVP utilized two major software systems for working with data - xGDS and Playbook. xGDS (Exploration Ground Data System) is a set of web based services for science operations that encompasses tools for planning, monitoring, visualization, documentation, analysis, and search (Deans et al, 2015). The usage of xGDS can be broken down into four mission phases - (1) planning, (2), monitoring, (3), archiving, and (4) exploring. In the planning phase, the science team would use xGDS and a priori map information as well as remote sensing data to create mission plans and traverse paths. Monitoring involved map and traverse based tools to visualize telemetry as well as rover position. In archiving mode xGDS is used to convert data into meaningful representations and searchable databases allowing for easy future uses. Lastly, in exploration mode, xGDS allows users after missions to quickly understand the collected data, where it was collected, and when it was collected.

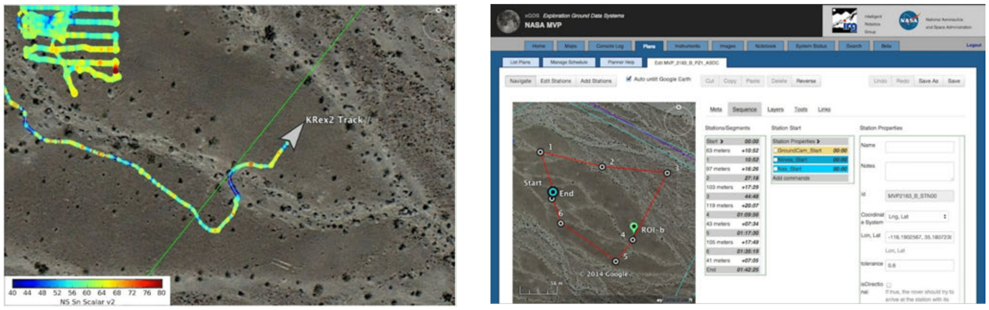

Figure 4: xGDS software used in MVP. Left shows rover paths and instrument data overlaid on top of a satelite image.

The Playbook software was used in conjunction with xGDS primarily for timelining and scheduling of mission activities. Similar to xGDS, Playbook is a mobile and web-based tool, providing timeline management. Playbook shows a full timeline with bands for each user, task, and schedule. It contains activities, which can be opened as streams to show all activities associated with a person, place, or instrumentation equipment. Playbook also has the capability for users to send rich media (pictures, video, text note) through an off-band communication channel to all users. This would be sent to everyone without interrupting the on-going voice loops (Marquez *et al.*, 2019)

Voice loops have long been the standard technique since the early Apollo mission controls for synchronous communication. Multiple operators both on the field and in back rooms can be on a single voice loop. Some operators are designated listen-only, while others have talking roles. Many people will also be on multiple voice loops contributing to different simultaneous conversations at once. Patterson. et al (1999) propose that voice loops were an extremely important function in mission control as it "is a powerful groupware tool because it allows practitioners to ‘listen in’ on others’ activities while also pursuing their own goals and activities. Perhaps more importantly, listening in on the voice loops does not interfere with ongoing activities of other personnel." Patterson et al. also stress that voice loops are distinct from other media because they can allow communication across spatially distributed people and do not overload the visual channels. MVP utilized four voice loops, while BASALT had two.

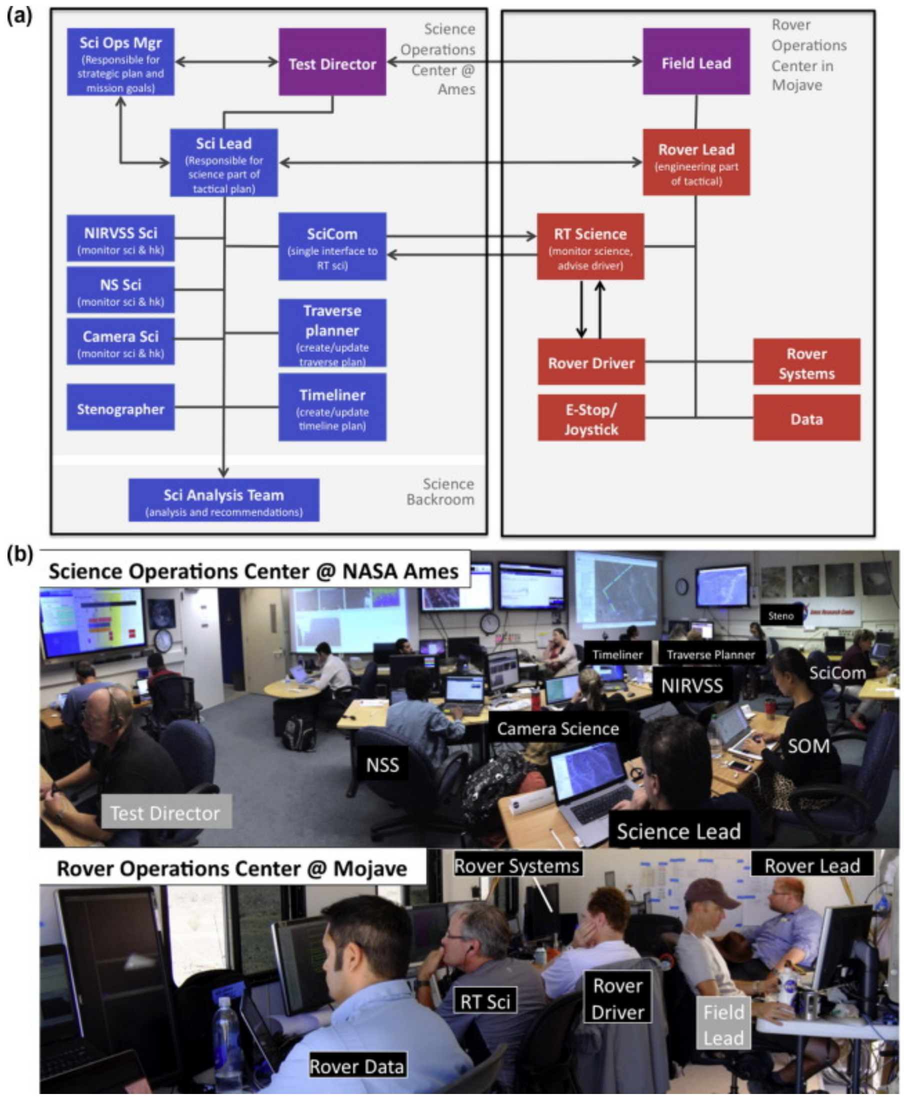

Figure 5: Operations center for MVP. (a) shows the communication hierarchies and roles involved. (b) shows how the physical Science and Rover Operations centers were laid out.

2.1.2 *BASALT Analog Mission*

BASALT deployments typically had 10 missions days, with one nominally 4 hour human led Extravehicular Activity (EVA) per day (Lim *et al.*, 2019). The latest deployment, BASALT-3 took place in the Kilauea Volcanic region of Hawaii in 2018. The BASALT concept of operations was derived from current proposed Mars mission architectures as well as past best practices from other analog research programs. (Paylor *et al.*, 2019) Communication delays anywhere from four to 22 minutes would be created artificially between the intravehicular (IV) team and the Mission Support Center (MSC). The IV team would be on real time communications with the extravehicular (EV) team out on the field. IV and EV would communicate through sending images, voice, and other data which would get sent to EV crew members' wrist displays. The EV crew would traverse to designated sample locations, perform surveys, and collect instrument data as well as geological samples to be analyzed later.

The science team in the backroom would have access to images and notes through xGDS. They would also have the delayed audio and video feeds through a multi-channel, multi-intercom system (VCOM). BASALT utilized two primary voice communication loops. BASALT did not have any remote science-tactical specialists as all the members were co-located at the Mission Support Center. Integration of remote specialists would require reconfiguration in the communication and data architectures used in BASALT, however might be a very important capability to explore in the future. The BASALT MSC had multiple individuals dedicated to data categorization and management due to the volume generated. A "Leaderboard" was used to keep track of sampling priorities, status, and decisions. The leaderboard would assist all of the operators to manage priorities and view at a glance. Multiple leaderboards would be used based on need. This categorization od data was one of the major tasks performed by MSC. Similar to MVP, Playbook was used for the real time scheduling and planning (Beaton *et al.*, 2019b).

Aside from these software tools, during the BASALT-3 deployment to Hawaii, a few experimental capabilities were also tested. Surface based 360 imagery was generated using Gigapan hardware to generate panoramas that could be annotated. Surface based LiDAR data was generated for high-resolution data elevation maps producing

Figure 6: BASALT Field Images from BASALT-3 Hawaii deployment. (a) and (b) shows crew members with communications backpack traversing through the environment.(c) shows a look into the Mission Support Center, projection screens, and team working

scans of 5cm resolution. Augmented Reality tools Holo-SEXTANT, HoloSkype, and BASALT-OnSight were used as immersive capabilities (Beaton *et al.*, 2019a). Holo-SEXTANT allowed EVA Crew members to visualize traverses and environmental information while still having situational awareness of their immediate environment (Anandapadmanaban *et al.* 2018). HoloSkype enabled better communication and guidance from MSC to crew members. MSC would be able to mark points of interest directly in the field of view of a crewmember. They would be able to draw on a tablet which would lock directly into the environment in Augmented Reality allows the crew to easily understand MSC's guidance. OnSight utilized the surface based LiDAR data to create an immersive virtual walk-through of the traverse pre-mission (Beaton *et al.*, 2020).

Paylor et. al (2019), posit that a flexible built environment is important for the team to reconfigure as needed including components such as chairs, people, tables, A/V equipment, and monitors. BASALT's

Figure 7: Holo-SEXTANT in use during the BASALT-3 field deployment. (a) shows crew member calibrating their path and traverse. (b) shows a through-the-headset view of the augmented reality path and navigational information

deployment used a large room with operators sitting in rows as seen in figure [BLANK]. Two large projector displays acted as shared screens for viewing the most important content such as the Playbook mission log, Science Leaderboard, time displays, EV positional tracking, and EV camera video. While this layout allowed for the shared view at all times, it did not facilitate face to face discussions as easily. One scientist stated that "there is no point in having a team of experts in different disciplines if they can't all talk to each other and arrive at an information consensus." (Paylor *et al.*, 2019) Some chose to reconfigure their layout to face their collaborators at the expense of having to turn to view the projected screens.

Reviewing the BASALT and MVP analog mission show the tools used and the purpose they serve. Future directions for analogs and problems that were experienced in these are useful to note when designing new tools for a mission control environment. The reconfigurability of the environment, need for more screen space, difficulties in communication, and desire to incorporate remote experts are all areas to explore.

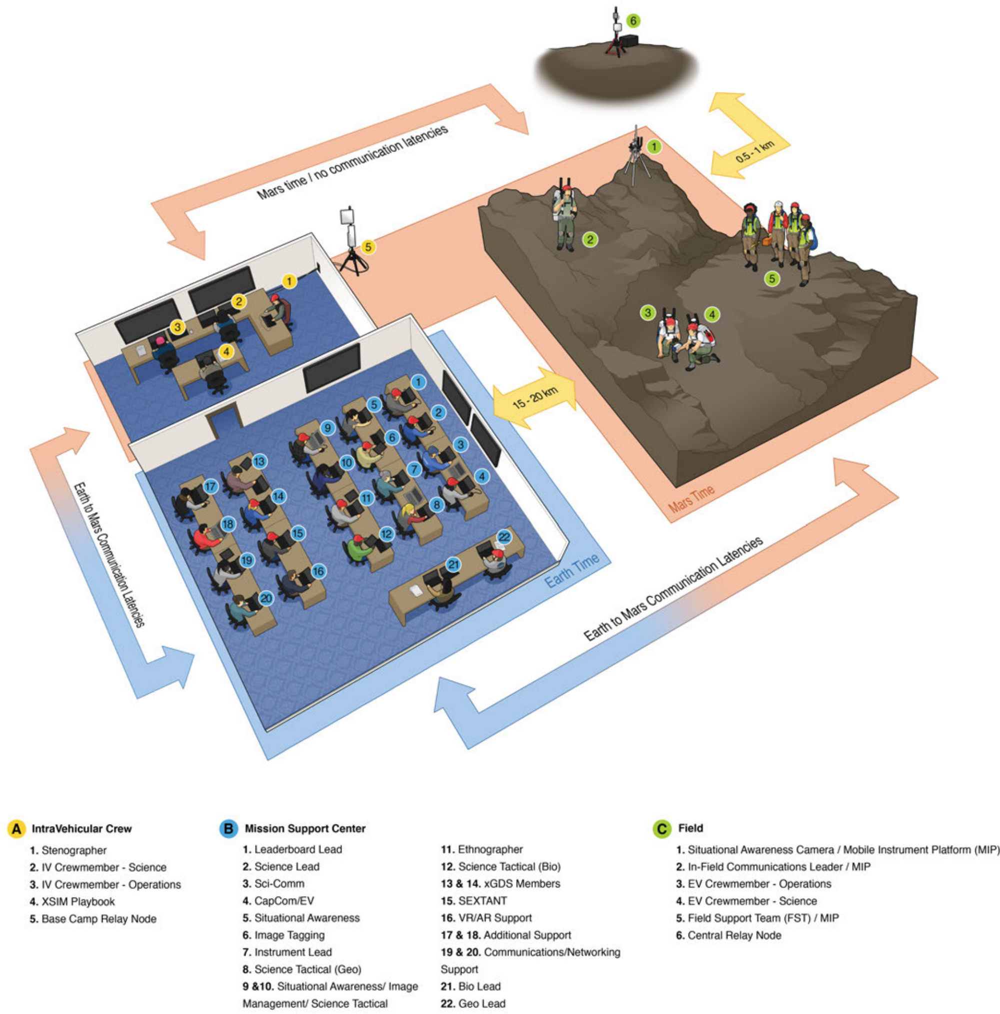

Figure 8: Outline of the BASALT Research Program. The MSC, Intravehicular team, and extravehicular team can be seen. Individual roles within the Mission Support Center can also be seen. The IV crewmembers operated on "Mars time" along with the EV crew. MSC was on "Earth time" working with the given simulated Mars latency conditions for that mission day.

2.2 THE COMPLEXITY OF MISSION CONTROL

As seen from this analysis of BASALT and MVP, mission control architectures can be quite complex. The tools currently designed address certain challenges but present areas of improvement just as well. Here I would like to discuss a model for human information processing to frame the difficulties a complex mission control environment can present. Second, I will discuss how flight operations mission control systems for past missions looked. Flight operations missions have a much larger group of individuals to coordinate, stricter constraints, and generally more complexity than the simulated analog environment.

In 1968 Atkinson and Shiffron presented a structural model to understand how the brain processes information. They presented the multistore model of memory that is commonly agreed upon today. Memory consists of three stores, a sensory register, short term memory (STM), and long-term memory (LTM) (Mcleod, 2017). The Cognitive Load Theory (CLT) builds upon this human information processing model. CLT relates to the amount of information that can be stored in the working memory at one time. Since working memory has a limited capacity, preventing it from overloading will improve information processing. This has been commonly used from improving educational tools and visuals but more generally applies to all data visualization as well. Cognitive load theory identifies properties that can make information easier or harder to digest based on its representation. Cognitive load can be split into intrinsic and extrinsic load. Intrinsic load refers to inherent challenges in understanding content while extrinsic focuses on the representation. Reducing intrinsic load can be challenging, while designing better representations, visuals, and software, extrinsic load can be decreased. Much of the data from analogs and flight scenarios is three-dimensional. Representing this in two dimensions for traditional software, typically involves showing multiple representations. This requires a mental transformation to generate the full 3D picture in our heads. Removing this step, by representing three-dimensional data in true 3D, can reduce extrinsic load.

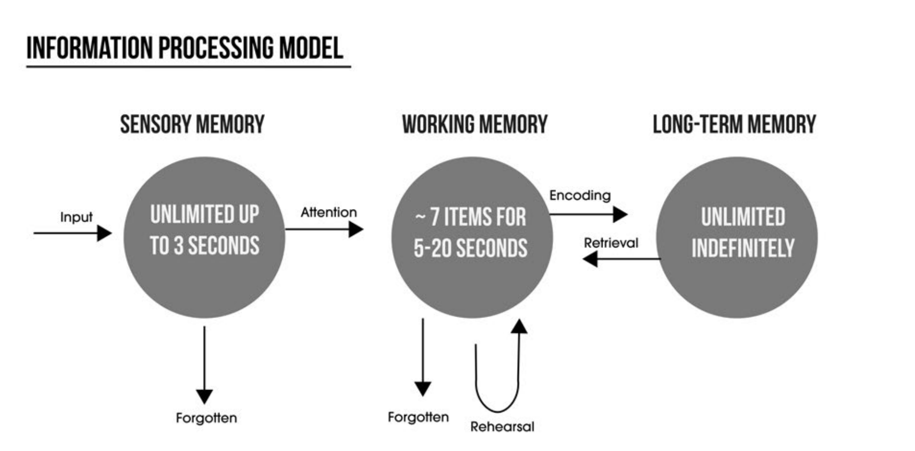

Figure 9: Information Processing Model created by Atkinson and Shiffron showing the 3 memory stores and their functions. It can be easy to overwhelm the sensory and working memory.

The second area of extrinsic load to analyze can be explained by a phenomenon called split-attention affect. This effect occurs when visual information is presenting in split parts that need to be integrated to be fully understood. This requires users to look back and forth until a combined mental representation can be created creating additional mental cognitive load. Michael Granaas talks of the psychology of display systems in Mission Controls. He posits that "STM is quite easy to overload in the real-time environment of a mission control center." Working on multiple displays, across multiple programs or other inconveniences like this can quickly add up the limited slots in the STM and overload the user (Granaas, 1988).

Here is an illustrative example to understand how both of these can manifest themselves for analog missions: Rover instrument data currently is shown in separate representations for rover traverse, instrument data, terrain characteristics, and data analysis. The terrain data is inherently 3 dimensional, but represented in 2D. Users pan around in software to understand the geological features. Combining these 3 views is critical to understand the entire mission objectives and results, but is presented in split views. The split view model requires the viewer to create a mental model within their head combining the different representations of data. There is potential here for significantly reducing the amount of cognitive load and providing a faster, more intuitive visualization.

In the Apollo missions days, mission control was far more complex. In his book, Digital Apollo, David Mindell presents the story of the faulty Apollo 14 abort button. "For all of the Apollo missions, ground controllers spent the long hours of the flights staring at lists of numbers on computer screens as they downloaded the telemetry in real-time. The controllers became exquisitely sensitive to the binary bits that made up the numbers" (Mindell, 2008, p. 270). Right before the lunar module was about to begin its descent, operators were meticulously watching the data stream when they noticed an anomaly on the ABORT button. A faulty solder joint was to blame as the computer was showing a faulty indication to abort landing even though the button was not enabled. The team of controllers and remote experts at MIT who wrote the software got to work immediately. They had two hours to come up with a fix to prevent the Apollo Lunar Module computer from performing a premature mission abort procedure. Engineer Don Eyles was part of the team working frantically. They came up with a programmatic procedure that could be entered by the crew members to instruct the computer to ignore the abort button entirely. If they needed to abort during the lunar descent, they would still be able to do so through a backdoor manual procedure. This quick fix saved the mission as Apollo 14 was able to land successfully.

The team of operators and engineers were able to problem solve a spacecraft malfunction thousands of miles away to generate a new piece of code that could be programmed and fix the malfunction. Undoubtedly the stress and cognitive workloads would have been very high during those moments. The early Apollo mission control was build on state-of- the-art mainframe computers with monochrome displays. The displays would show a constant stream of telemetry and space craft data as raw text and numbers. Flight controllers would spend over 60 percent of their time making sense of the data and turning it into meaningful representations, typically with pen and paper. Each operator had trained for multiple years to retain the focus and knowledge needed to do this. In 1987, the Real Time Data System (RTDS) project was undertaken to introduce advanced automation in MCC (Heindel *et al.*, 1991). RTDS was designed to improve response times, simplify data representations, and provide graphical user interfaces that replaced the text displays in preparation for the Space

Shuttle Program.

Since then, the Houston Mission Control Center has gone through further updates to bring in electronic documentation, better hardware, and more ergonomics. As complexity of mission increased from Mercury to Gemini to Apollo to Skylab to Space shuttle, the Mission Control architecture and design was updated to better serve the new needs and utilize state-of-the-art technology (von Ehrenfield, 2018). Today, NASA is committed to landing astronauts back on the Moon by 2024 through the Artemis program. After establishing sustainable missions by 2028, the next leap will be to send astronauts to Mars (NASA, 2020). The complexity of mission control will once again increase and mission architecture is becoming more complex. The transition from using pure monochrome text displays, to graphical user interfaces enabled operators and flight controllers much more capability. I propose that transitioning from two-dimensional interfaces towards incorporating virtual reality can improve performance in a similar manner. Virtual reality can help reduce cognitive load by reducing context switching between screens of information. Spatial interfaces can be highly configurable and facilitate improved levels of collaboration. And of course, VR adds another dimension to the user interface for mission control creating more bandwidth for information transfer.

2.3 IMMERSIVE TECHNOLOGY IN SPACE APPLICATIONS

Numerous projects at NASA have utilized immersive technologies like Virtual and Augmented Reality for various space applications. These are presented here as concepts of experimental usage of immersive technology. Insights from their design and features are very useful for understanding what is possible with virtual reality.

2.3.1 *Virtual Interface Environment Workspace (VIEW)*

The VIEW Project influenced much of the later work on vMCC presented in Chapter 3

NASA has a wide history of experimenting with virtual reality dating back to the 1980s. At NASA Ames Research Center, Scott S. Fisher was an early pioneer of VR systems. His team at Ames developed a head-mounted VR system controlled by operator position, voice, and gesture for use as a multipurpose interface environment for teler-

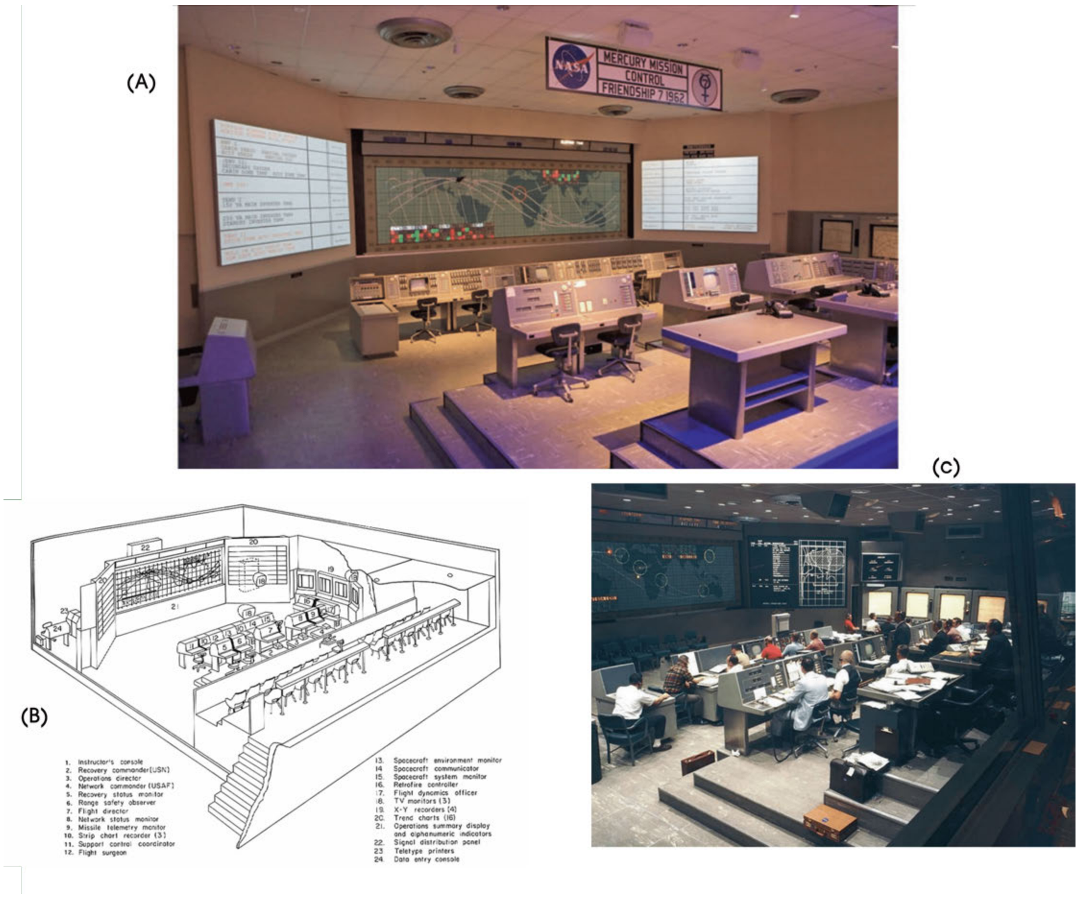

Figure 10: Space flight Mission control from the Mercury, Apollo, and Gemini Eras. (c) shows the operators meticulously converting the telemetry data into better representations of data with pen and paper. Image Credit Manfred von Ehrenfield - Apollo Mission Control

obotics and data-management. Fisher's system was a novel hardware system that did not require room-sized equipment and provided display, tracking, input, and feedback for users.. It could be used to interact with a simulated telerobotic task with immersive spatial cues as an operator takes precise viewpoint control with tactile input while controlling the robot. Users could also use it to manipulate data and monitor systems rapidly. Figure 11 shows the two modes of using the VIEW system as well as the hardware worn by a user. A major long term goal of the VIEW project was extending these capabilities for collaboration with an objective of providing a collaborative workspace for users to interact with nuances of face-to-face meetings (Fisher, 1988; Fisher 1989). The VIEW Project was far ahead of its time with a vision for the potential of VR technology.[1]

Figure 11: VIEW Project shown. (a) shows a schematic of different components of VIEW, including virtual control panel, 360 degree data space, 6DoF gesture tracking, tactile input, voice interface, and 3D sound cues. (B) shows a user wearing the VIEW hardware. (C) shows a colored rendering of the manipulable data panels. Image Credit: NASA/W.Sisler, S. Fisher, 1988

2.3.2 *OnSight*

A modified variant of OnSight was used during the BASALT-3 Hawaii Analog mission called BASALT-OnSight.

NASA JPL (Jet Propulsion Laboratory) partnered with Microsoft to showcase Mixed Reality technology using Mars rover data. The OnSight project allows remote experts to meet together in a simulated Martian environment to view and manipulate rover data. Using the HoloLens, 3D terrain reconstruction, and specially-designed motion capture systems, OnSight delivers an immersive collaboration software for scientists and engineers. Abigail Fraeman, a member of the

1 Video of VIEW at: http://itofisher.com/sfisher/portfolio/files/viewlab.html

Curiosity's science team states that "Being able to visualize Curiosity's drives and virtually walk them before we actually do it with the rover is really helpful to give me a sense of how safe or challenging the terrain will be." OnSight provides tools to understand the environment as well as control the rover in the simulated environment.

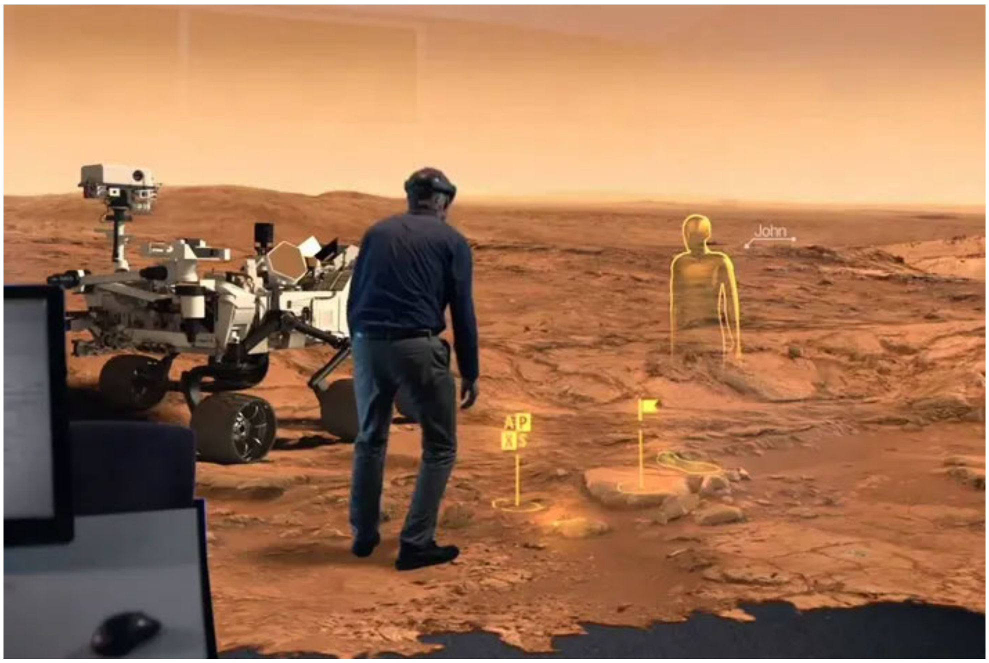

Figure 12: Onsight virtual environment and two users collaborating. Rover waypoints can be seen as well as rover path marked out.

2.3.3 *Hybrid Reality and Advanced Operations Concept Lab*

At NASA Johnson Space Center, a low cost, scalable Hybrid Reality (HR) systems that integrates head+body tracking, visual feedback, weightlessness, tactile feedback, and photo-realistic graphics are being explored. Utilizing a detailed 3D virtual replica of the International Space Station (ISS), the Hybrid Reality Lab enabled astronauts in training to be in the actual environment before heading up to space. The NASA installation also utilizes real physical spaces to create the Hybrid Reality. This allows astronauts to feel walls and furniture to complement the visual environment. Using structured light 3D scanning technology, high quality virtual replicas of tools and props are also utilized. Tactile feedback for these props is provided by using tracked 3D physical props. Using object tracking, the realistic visuals combine with tactile feedback of using props such as drills, instruments, and more (Delgado et al., 2017).

To provide a sense of weightlessness, the Hybrid Reality Lab have combined their virtual reality simulation tool with the existing Active Response Gravity Offload System (ARGOS) at JSC. Argos is "essentially a smart tether which attaches to your back, offloads your body weight and accounts for your momentum in the vertical and horizontal directions to make you feel like you are in lunar gravity, Martian gravity, micro-gravity, or anywhere in between," says Matthew Noyes, Software Lead at the Hybrid Reality Lab.

The Hybrid Reality Lab training system creates realistic VR environments, with simulated weightlessness, physical environmental feedback from the stage as well as tracked props, an a multi-user networked experience. This system is actively helping train astronauts preparing for trips to the ISS.

2.4 SUMMARY

As NASA approaches more complex mission architectures, analog missions such as BASALT and MVP provide insights into technological solutions that need to be built to accommodate. Looking at these two analog missions and their software architectures shows the key requirements as well as constraints. Understanding the multi-store model of human memory and cognitive load theory allows for seeing the difficulties a flight controller or operator in complex environments such as mission control can face. Looking at specific examples from flight operations mission controls and the evolution of the mission control design elucidates these difficulties and the design changes made to improve flight controller performance. Lastly, looking at the history of immersive technology within NASA shows some of the concepts and designs that have been explored representing truly pioneering work. These highlight the promise of using virtual reality for space applications. Design lessons and insights from these projects are also useful to keep in mind when building immersive VR tools.

(A)

(B)

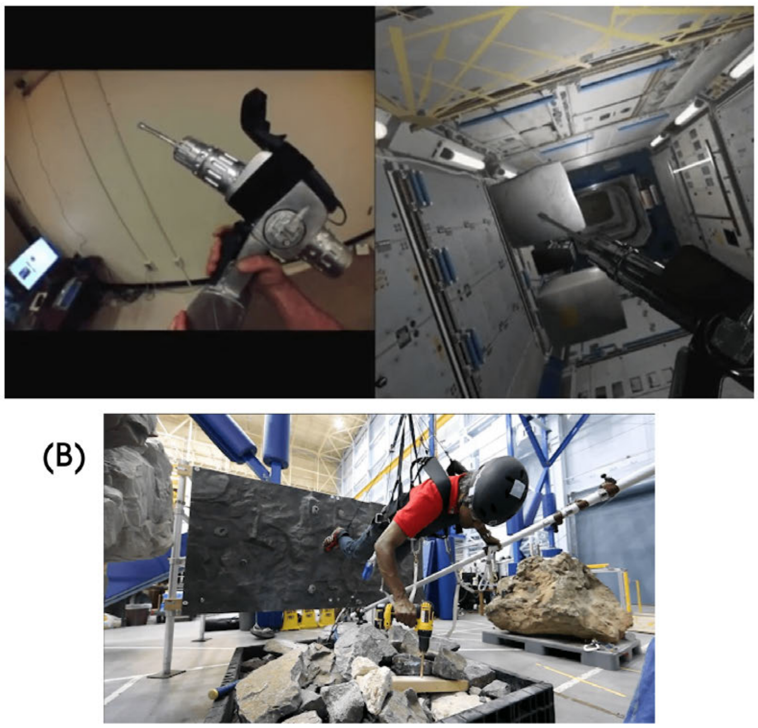

Figure 13: Hybrid Reality lab at Johnson Space Center. (A) shows a physical drill prop on the left and a virtual model mapped one-to-one in VR. (B) shows the ARGOS system as a trainee performs a drilling task while suspended in simulated micro-gravity.

3 VIRTUAL MISSION CONTROL CENTER(VMCC) - SYSTEM DESIGN

3.1 OVERVIEW

The Virtual Mission Control Center (vMCC) is an experimental framework to integrate multiple components of mission control for better data analysis, decision making, and communication. vMCC can enable remote collaboration with science-tactical experts during an analog mission, assist with pre and post-mission analysis of data, and lastly, help train operators by simulating mission control environments without the need for large scale equipment or access. The focus is primarily on usage for scientific analog missions although future extendability to flight scenarios is important. In this chapter, I describe the design principles behind vMCC, its architecture, and implementation. This project was designed and implemented during the Fall 2019 and Spring 2020 semesters implemented by myself and Trent Piercy under the supervision of Professor Dava Newman. Our work led to the first version of vMCC, and I will describe future directions and potential that our platform enables.

The following principles guided the design of vMCC:

FOSTER A COLLABORATIVE ENVIRONMENT: Collaborative discussion is a key component of mission control architectures as seen with the voice loops and ConOps analysis from Chapter 2. vMCC will integrate shareability and group communication in each component and feature.

DYNAMIC SPATIALIZED VISUALIZATIONS OF DATA: To utilize virtual reality as a medium to its fullest extent, visualization of data should be inherently three-dimensional. With an emphasis on 3D and multi-dimensional data, vMCC can work to complement existing tools such as xGDS.

GEO-SPATIAL DATA CATEGORIZATION: Multiple representations are required for all data, however, as a first step, geo spatializing all data is the best categorization for analog missions. Almost all data can be tied to a specific geographical location. Utilizing a geo-spatial data organization system makes it easy for creating comprehensive mental models of all the data.

INTUITIVE NATURAL USER INTERFACE: As a new tool and medium to most of the potential users, Virtual reality design needs to emphasize building simple and natural user interfaces that take minimal to no training. This ease of usability is also critical to helping users make fast manipulations and decisions

The current tools that analog missions have been using are limited by the constraints of 2D software. Operators often have multiple monitors in mission control settings. Teams huddle around a single screen to collaborate. Screen real-estate is divided into different visual representations of data in separate applications in many cases. Only after analyzing for hours and creating complex mental models combining the different representations of data, are conclusions gained and decisions made. This workflow is not optimized for collaboration or rapid visualization of data. Looking at scientific analogs, there are 4 mission phases as split up by xGDS: (1) planning, (2) monitoring, (3) archiving, and (4) exploring (Deans et al., 2019). For this project, the focus was on (1) and (4). Pre and post-mission analysis of data does not require real time data inputs but rather utilizes data from previous missions or data collected during a mission. There is often collaboration and extensive discussion in these phases. Additionally, the lack of real time data input reduces the complexity for an initial exploration and is easier to test by design.

Virtual reality, as a medium, inherently provides several affordances as the application is not be bound to a 2D monitor window. Several screens, and data models can be visualized side by side with ease. Multi-user VR applications can create a sense of presence for remote collaboration with tracked avatars. Building on top of these fundamental features provided by the medium, tools and modalities were designed to foster collaboration and data visualization. With mission phases (1) and (4) as the focus of vMCC, explorative data visualization and communicative data visualization are the user tasks that need to be fulfilled. An overall system architecture was first created for the collaborative multi-user VR environment. Within this environment, three input Data-viewer modalities were developed: the Virtual Desktop, Joint Dashboard, and Spatial Console. In each of these modes, users can perform a variety of tasks and visualize different forms of data. A suite of tools have also been created to facilitate user collaboration and data visualization.

3.2 architecture

The vMCC platform is built on top of the Unity 3d game engine[1] (Version 2019.3.7f1). Today, commercial game engines are the most robust frameworks to build virtual reality systems because of their ability to simulate real world properties, graphical output, and inherent three dimensionality. The interaction code and backend is written in C#, the language compatible with Unity. I chose to build vMCC using Unity for its rapid prototyping and iterative testing capabilities. A strong asset store and community provides a plethora of plugins and libraries for faster development.

This project was designed primarily for the HTC Vive Pro SteamVR headset[2] although it natively works with all SteamVR Head Mounted Devices (HMD). The HTC Vive Pro is an off-the-shelf 6DoF VR system consisting of a headset, a pair of hand controllers, and tracking base stations. Each system requires a dedicated graphics workstation desktop, although a single set of tracking base stations can track multiple HMDs. Each headset tracks itself in 3 dimensional space by receiving optical signals from the mounted base stations (laser and IR emissions) to orient its position and rotation. The Vive Pro system has robust tracking compared to other VR systems such as the Oculus Rift or Quest[3]. This system, while more expensive and difficult to set up, provides high visual fidelity, and quality of experience. I choose to go with this higher end virtual reality system after considering the ease for development and possible features for users when compared with lighter-weight mobile hardware.

3.2.1 *Networking*

Networking is a key component of vMCC to communicate between the users. Unity does not have built-in networking components, so vMCC was built with Normcore, a peer-to-peer networking tool[4].

1 Unity 3D: https://unity.com/
2 HTC Vive Pro: https://www.vive.com/eu/product/vive-pro/
3 Oculus Rift, Rift S, and Quest: https://www.oculus.com/
4 Normcore: https://normcore.io/

Normcore allows for all clients to be synced for avatars, voice, and persistent objects. It enables message passing, custom events, and data models. Built on top of this framework, vMCC can guarantee persistent synced experiences for multiple users without latency or lag. Normcore has rooms to separate groups of users. For vMCC, only a single room is created, meaning multiple instances of vMCC are currently not supported. All users in vMCC will be in a single shared persistent virtual space.

This networking framework uses a system called `DataStores` to sync data. The `DataStores` act as a ground truth copy of any data that is synced. They automatically detect changes and informs all of the software clients connected to the room. Each client then updates their world to match the `DataStores`. Many other networking frameworks are built on top of message passing systems where application state is communicated between clients through events and messages. This however, does not handle new clients who join late and requires complicated solutions to keep all users in sync. Normcore also utilizes an `MVC` (Model, View, Controller) based architecture to create separation of networking code from local code. The `DataStore` holds a series of `RealtimeModel` objects. Each `RealtimeModel` is a piece of data that needs to be synchronized across clients. Each client maintains the view of objects in Unity. This is the local copy of the data. Lastly, the controllers are a series of `RealtimeComponents` which act as controllers. The role of the `RealtimeComponents` is to manage the state of the local view. When the view is changed by the `RealtimeComponent`, it will also update the `RealtimeModel`. The `DataStore` then automatically updates all clients with the changes.

Normcore also provides a concept of "ownership" of data. Only one client is able to make changes to a given `RealtimeModel` at any given time. Ownership transfers occur when one client is done manipulating data and another starts to modify data. Currently, vMCC is designed for 8-10 synchronous participants, although Normcore networking can support over 50 users in a single room. Creating multiple rooms can also enable expansion of this limit.

Co-located rooms are not supported and was not in the scope of this project. If multiple users are co-located in the same tracking region, a calibrated single coordinate system would be needed to align their physical and virtual locations to prevent them from colliding with each other. Adding this layer of spatial awareness to vMCC, would enable the networking system to work seamlessly whether remote or co-located.

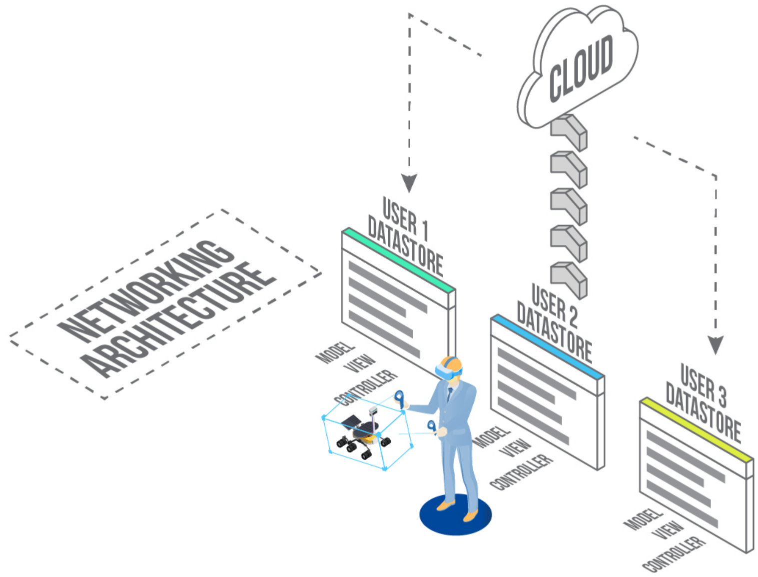

Figure 14: Networking architecture showing the flow of data during a concurrent session of vMCC with multiple users. User 2 is making manipulation changes to a model, and the Datastore gets updated. In the cloud this information gets registered and sent to all other clients for updating.

3.2.2 *Input Management*

From a development perspective, it's common to use frameworks such as React, Vue, SwiftUI, and others to speed up development in web and 2D development. Virtual reality is a novel medium in many ways and design patterns have yet to be set. There is a great deal of freedom with designing interfaces and features. Every object needs to be created in three dimensions and have properties such as physics, colliders and renderers to exist in the world. Input management, feedback, and visual design are all important to consider. Mixed Reality ToolKit (MRTK), is a spatial toolkit built by Microsoft originally for the HoloLens but since extended for a variety of Virtual and Augmented Reality HMDs.[5] MRTK provides a series of utilities, scripts, and a framework for accelerating cross-platform Mixed Reality application development in Unity. It provides an input management system that abstracts the actual device an application is built for. This enables extendability to future devices easily. MRTK provides UI building blocks such as panels, buttons, keyboards, and tooltips to standardize the user experience. vMCC's input system, VR device management, and user interface are built on top of MRTK.

3.2.3 *Avatars*

An avatar in VR is a material representation of the user (Lessig, 1999). For the scope of this project, it was imperative that we have avatars that represented our users and provided a sense of presence when interacting and talking with other other avatars. We also wanted it to fit the visual language of the rest of the environment and application. The first iteration of the avatars were abstract spheres for heads. Preliminary user feedback showed that these were too primitive and abstract. Without better representations, it was tough for users to communicate with others. Creating avatars with lots of detail and realism would create a need for customizability. Each user would need their avatar personalized to best represent them. Additionally, too much detail can lead to the Uncanny Valley effect. Roboticist Masahiro Mori coined the term Uncanny Valley as:

> “*The discomfort experienced when human replicas which appear almost, but not exactly, like real human beings elicit feelings of eeriness and revulsion (or uncanniness) among observers.*”
>
> Masahiro Mori, *1970*

The Uncanny Valley effect can destroy presence and co-presence in VR. What was needed was an avatar system that would be representative of people and the basic sense of presence without introducing too much detail that would enter the Uncanny Valley. With this in mind, we designed our avatars to be simple, androgynous and provide just

5 MRTK:https://docs.microsoft.com/en-us/windows/mixed-reality/mrtk-getting-started

the right amount of detail. After analyzing avatar designs from several of the current virtual reality applications and systems, we designed the vMCC avatar to have a translucent androgynous human head, VR headset, torso, and two hands. This proved to be the bare minimum needed for convincing expressive avatars. The translucent head was not distracting and users could even look through someone standing in the way. The androgynous head provided enough context to associate the avatar with a person behind the headset without the need for customization. Placing a headset on the avatar also mitigated the need for eye contact and emotional eye animations. Eye contact can be a crucial key for human presence, and replicating it in VR is difficult. Placing the headset, allows for suspension of disbelief and is a realistic representation of the real user as well.

The torso is a polyhedron shape to allude to the concept of a torso. The hands are represented as controllers for each user locally, while represented as mittens for all other users. Mittens provide a very basic representation of hands without needing to animate fingers and joints. For the context of vMCC, this basic representation of the avatar satisfies the requirements for usability.

By creating simple, usable avatars, we focused on the presence that was needed to convey that these were real life colleagues and humans behind the avatar. This enables users to gesture with their, see head movement, and basic characteristics like posture and gaze. Body language is an important part of human communication and with the available data, vMCC creates a simple representation of the body to communicate this.

Figure 15: vMCC user on left wearing a SteamVR HMD. On the right are the vMCC avatars with nametags above each user.

Aside from body language, users can also communicate verbally with one other in vMCC. A Voice Over Internet Protocol (VoiP) is used for transmitting mic audio from each user to all other users. This audio is spatialized, providing spatial awareness for where each user is. The human brain interprets audio to understand and make decisions about the surrounding environment. In particular, we use our two ears in conjunction with the ability to move in 6 degrees of freedom to precisely locate the direction, distance, and position of audio signals. For vMCC, since VR headsets have stereo headphones as well as 6DoF head tracking, this real world spatialization of audio can be simulated. When one user is closer to the others, they can hear each other louder compared to being farther away. With these spatial audio cues, it is easy to locate specific people and create a sense of co-presence. It is also possible to have multiple audio zones within the single shared space. Because of the volume falloff from distance, two groups of users could stand at far ends of the vMCC hangar and have two separate conversations without interfering with one another.

This creates new possibilities for breakout conversations where users can start off together, analyze data, and separate into the different parts of the room. All the while they're in the same virtual space and have all the same assets, tools, and data to use.

3.2.4 *The Environment*

The final component of the system architecture to discuss before diving into the capabilities is the virtual environment. When designing this virtual environment, I wanted to draw on much of the inspiration that brought about this project. The development of vMCC was inspired by concepts such as the holodeck, as well as futuristic command decks that portray natural user interfaces, and intuitive data manipulation as a user swipes through data to make a decision. Fitting in with both the inspiration, and types of data visualized within vMCC, a space based geographical setting was chosen. A space hangar is where vMCC is set in. Outside the windows are stars in the horizon, and inside is a futuristic industrial environment. This visual style also provided affordances that leverage the virtual reality medium. With sharp lines and simple textures, the space hangar design looks crisp in today's VR headsets. It provided context to users and set the theme of entering vMCC, preparing them for interacting with data in an entirely novel manner. Figure 16 shows multiple views of the environment. Early tests showed that the built environment can focus the users and provide a framework for them to navigate in and mentally prepare for task completion.

From an architectural standpoint, the vMCC environment is about 50x75 meters in size. While designed for 8-10 users, vMCC can accommodate many more from the architecture. The environment is split into 2 sections. One is a private lobby that all users enter on opening the applications. Here, they can configure settings, and get ready to enter the shared hangar. The hangar is a shared virtual space where all users can see and talk to each other. Three distinct zones separate the hangar representative of the three data-viewing modalities. Decals are located on the floor and walls to label these regions with their function. Simple props add ambiance to the atmosphere such as server racks and tables with chairs. The hangar also contains a balcony observatory area. This can be useful for observers watching a team operate, ethnographers studying the overall human dynamics, or even visitors curious about mission operations. Lighting of the environment was fine-tuned to provide enough brightness for working conditions, while still maintaining realism. Shadows and spotlights are used for emphasizing areas of importance such as the spatial console. Regardless of the real world environment for users, they can enter a comfortable and pleasant virtual space for collaborating with colleagues in vMCC.

Both the avatar and environment contribute to the core concept of presence and immersion. With traditional mediums, suspension of disbelief leads to the audience accepting a presented reality. In virtual reality, this effect is heightened as more of the senses are stimulated. The use is surrounded by virtual objects, other people. and their environment. It's important to distinguish immersion from presence. Mike Alger presents a model for doing this: "Immersion is being presented only with data from a false environment, but presence is having the body feel it on a fundamental level" (Alger, 2015). Where immersion could simply be the process of seeing and experiencing a virtual reality, presence is the feeling of believing it and losing track of one's real world surroundings. Presence is one of the ultimate goals of any virtual reality experience. For vMCC, creating co-presence is perhaps more important. I define co-presence as the fundamental sense of being with another person and co-experiencing

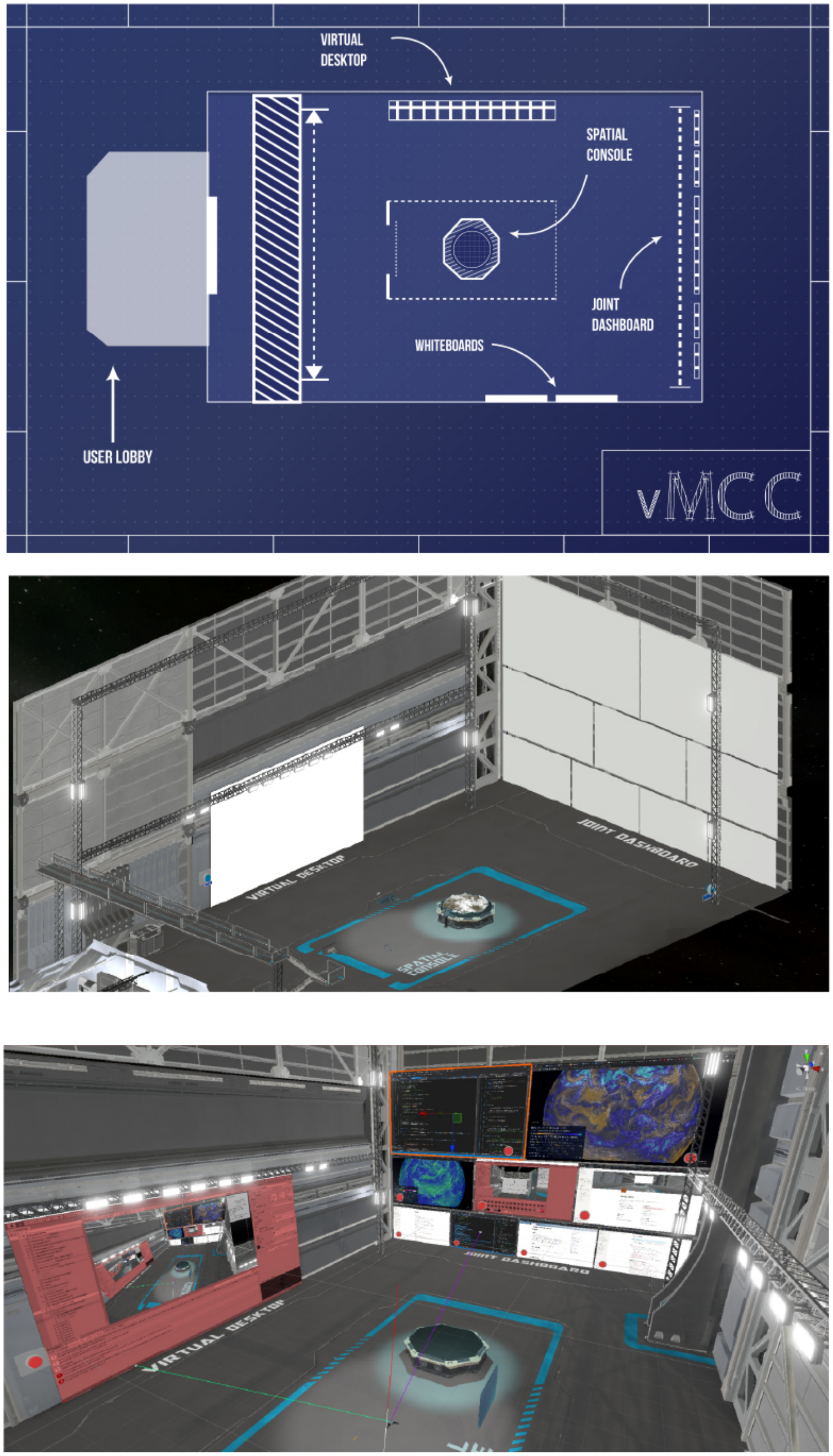

Figure 16: Top shows a blueprint sketch of the vMCC environment. Bottom two images show the environment in Unity from different views. The bottom-most image shows data and images populated in the environment

something. In the context of collaboration, co-presence can be an intangible driver of productivity. When the human perceptual system believes that an avatar is a person, communication and collaboration can be stronger. Creating a high quality environment for individual presence, and an adaquete avatar system for co-presence was important to create co-presence.

3.3 data viewer modalities

3.3.1 *Virtual Desktop*

The vMCC tool is designed to work as a companion to existing desktop and web tools such as xGDS. A virtual desktop within the VR environment provides a private monitor for each user to pull up their tools and use them. This large screen monitor provides enough screen real-estate for running multiple applications and controlling them from within vMCC. The virtual desktop acts as a bridge between the strong data organization and archival features already built into these desktop tools while allowing for the affordances of VR to bring those into a shared spatial environment. The virtual desktop comes into focus when a user clicks on the large display. Once in focus, the right hand controller acts as a mouse. When users move their controller in 3d space, a projection of the controller ray is used to estimate where the mouse should be and a trigger press translates to a mouse click. A virtual keyboard was also implemented for keyboard input. While slower than traditional keyboard input, this version is useful for simple queries and lookup tasks.

3.3.2 *Joint Dashboard*

The joint dashboard is a wall of critical information for the entire team of operators and decision makers to visualize. Users can bring in any images to place onto the 9 available virtual panels. Screenshots from the virtual desktop can be automatically sent to the dashboard. Figure 17 shows this process of sending screenshots to the dashboard. Where the virtual desktop enables users to work independently with their desktop tools, the joint dashboard allows them to share a visual with the rest of the team. Panels of varying size enable different forms of content to be seen whether it's a spreadsheet, graph, map, or document. The joint dashboard is a progression of the concept of huddling around a single screen to look at data. Rather than multiple people looking at a single screen, the joint dashboard creates an entire wall of displays that users can look at from afar or up close. It adds an extra level of shareability and persistence.

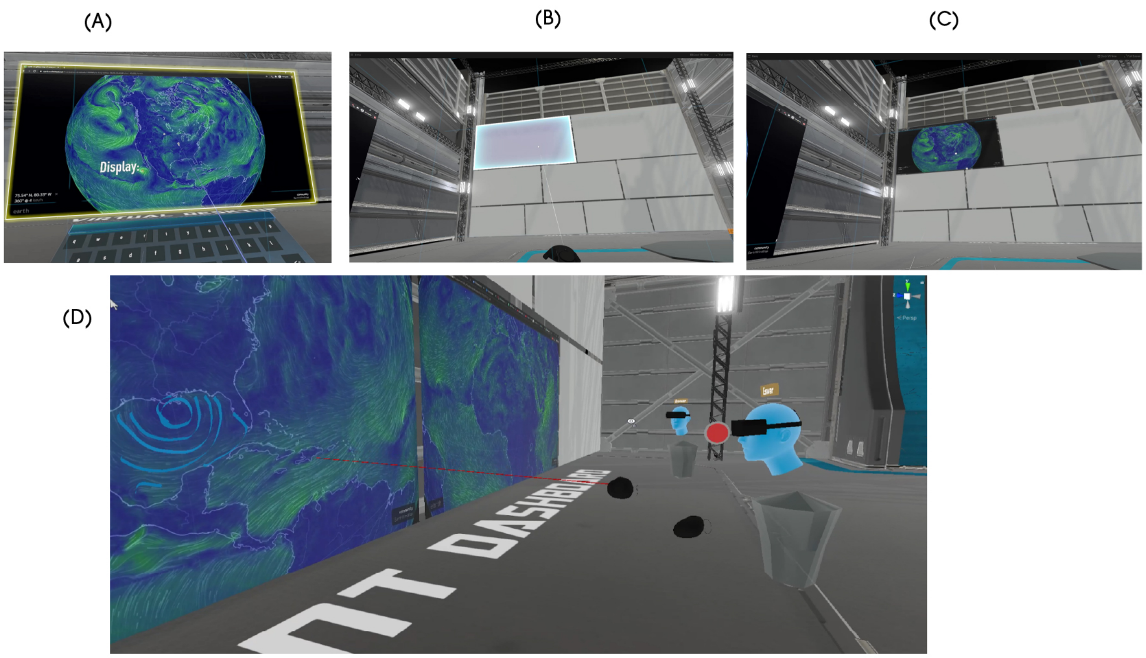

Figure 17: (A) shows a user creating a screenshot on the virtual desktop. Yellow highlight around the monitor indicates capture. (B) shows the user selecting a joint dashboard panel to push the screenshot to. (C) shows the screenshot now on the dashboard for all users to see and discuss. (D) shows two users having a discussion around the Joint Dashboard.

3.3.3 *Spatial Console*

The spatial console is the critical novel modality that was designed. The spatial console is a 3D data visualization table. Architecturally, it's designed for users to stand around a low-rising table to look into and analyze the data. On the spatial console, users can pull up different forms of three dimensional data to annotate, discuss, and analyze.

To begin, a user first selects the Data-viewer module to use: Model-Loader, Geo-Loader, or Data-Loader.

3.3.3.1 *Model-Loader*

The Model-Loader is a 3d model visualization system for static pre-generated models. These can be exports of terrain models, CAD models of tools, vehicles, etc, or 3D visualizations of data. On opening the Model-Loader, the user sees a file explorer pointing to a persistent folder. The file explorer will show all compatible models to be selected from as well as a small preview pin image and file name to guide the use. The Model-Loader can import 40 different types of 3D file formats and any of these types that are located in the asset loading folder will populate into the file explorer. Upon selecting one, the model will load for all users in synchrony, and vMCC will also appropriately scale and place the model over the spatial console table for all to view. This data viewer module is an optimized method to quickly load in exports from other software tools and collaborate on working with it. If users do not all have the same shared models locally, the Model-Loader also has capability to download and load in models from a given web URL link. This method ensures that all users have the model. Multiple models can be loaded in at once, and users can dynamically scale, resize, and manipulate the models as they would like to.

3.3.3.2 *Geo-Loader*

The Geo-Loader is the second data viewer module for the spatial console. With the Geo-Loader, users can pull up an overhead bird's eye view of any location in the world. Utilizing global Digital Elevation Maps (DEM) data combined with satellite imagery, vMCC creates 3d models of terrains and cities automatically. The Geo-Loader Menu requires a latitude and longitude entered in Decimal Degrees format. Once entered, the Geo-Loader can load in this data dynamically. Under the hood, this data is managed in a split tile format and only the necessary tiles needed are loaded in to optimize performance. Custom high resolution DEM data can also be used for specific regions. For example, drone based LiDAR might be used to capture DEM to 1cm accuracy at an analog field site. This can be imported and used with vMCC. The Geo-Loader will use the highest resolution elevation

LiDAR is a remote sensing technique using pulsed lasers to measure distance. It has been used in previous analog field tests to capture high resolution terrain models.

map available from either the default global elevation map, or custom data. The Geo-Loader can still load the lower accuracy DEM data for all other parts of the global map.

Once coordinates are entered, the Geo-Loader will load the map for all users. The map can be manipulated and moved around dynamically, zoomed into areas of interest, or pan to neighboring regions. Since this is built on a global elevation map, users can continuously pan to any location they would like to see. The elevation values can also be scaled to view contrast and features more evidently. Scaling the elevation can distort the map, but provide useful insights into geographical features.

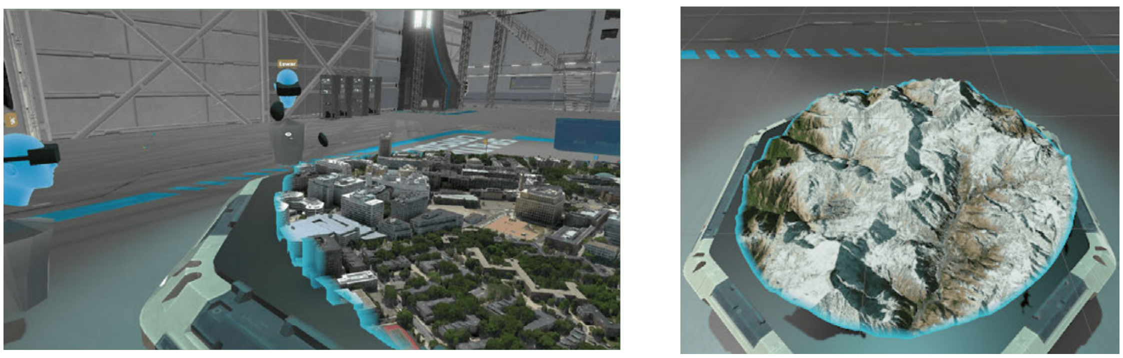

Figure 18: Geo-Loader with 2 different geographical locations loaded up. On the left is the MIT Campus. On the right is a segment of the Himalayan mountain range.

A series of marker pins are also provided for users. During discussions, when talking about specific points of interest such as a sampling site, or interesting geologic formation, a user can pick and place one or more of these marker pins. The pins are provided in different colors and shapes for easy differentiation. The set contains four generic colored pins, 2 flags, 2 orbs, and 1 tent. After placing a pin on the map, it will lock onto those coordinates unless moved by hand. This enables users to pan and zoom the map, keeping the pin locked onto the position it was placed in.

The Geo-Loader module can also load in custom raw data to view on top of the map. For example, if a series of sampled sites is provided in a `.csv` file as coordinates, the Geo-Loader will automatically load in waypoint map markers at each of these locations. This automates the map marker process and provides an accurate visualization.

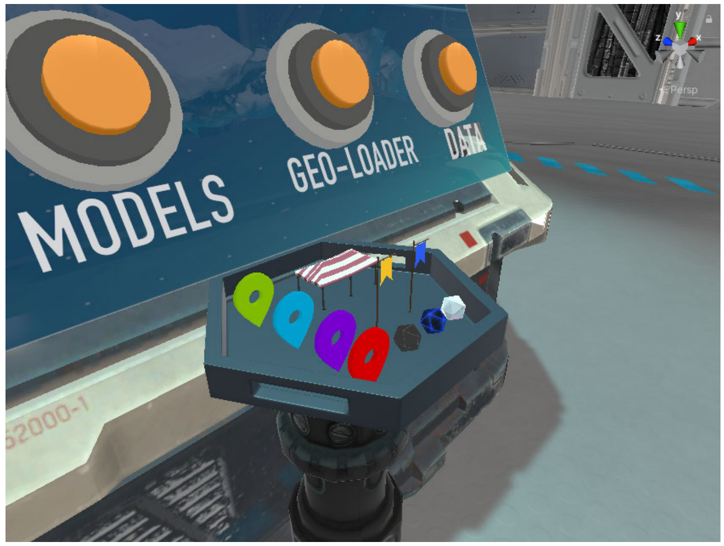

Figure 19: Map Markers including location icons, 1 tent, 2 flags, and 3 orbs. These markers can be placed and locked onto a set of coordinates on the map

This can be extended to visualizing a traverse path on top of the terrain as well. Both of these features can be used for pre/post mission navigation and point of interest analysis. Observing and analyzing waypoints and traverses in conjunction with a three dimensional terrain map from a bird's eye perspective can provide an overhead view of a mission that augments traditional 2D views of the same data.

3.3.3.3 *Data-Loader*

The Data-Loader is a Data-viewer mode to create panels of 2D charts from raw data. Users can provide a `.csv` file with data that they would like to see visualized. The Data-Loader will generate 2D graphs and charts from this data and create these as panels in the environment. Users can move these panels around to look at multiple charts and graphs for comparison.

3.4 TOOLS

To complement the three data viewing modalities previously presented, vMCC also has a suite of tools for users to communicate, annotate, and present. These tools are organized into the hand toolkit

- a menu system that is always with the user. With a press of a button, users can pull up the hand toolkit. Any of the tools can be brought up from the toolkit. Selecting a tool transforms the user's right hand into that tool.

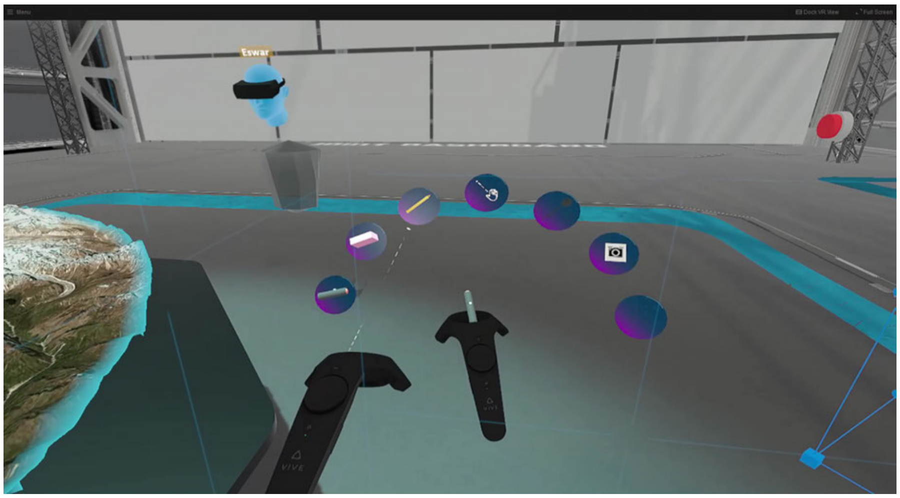

Figure 20: Hand toolkit shown on user's right hand. Using the left hand, any of the tools can be selected. Skeumorphic icons indicate the function of each button

3.4.1 *Selector Tool*

The selector tool is the default tool for both controllers. When selecting tools, the right hand transforms into the tool, while the left always remains in the selector tool mode. This is a cursor, pointer, and interaction tool. It is only visible to the user locally and not to others. With it, they can point and select intractable objects, menus, buttons, and handles. Many of these interactions can be done in the "near-field" where users are touching their controller with the interactable object. From afar, the selector tool acts as an extension of the user's hand allowing them to press a button from several feet away. The selector can only be used to interact with one object at a time. With a selector tool in each hand, users can interact with one object per controller. Typically, interacting with one object at a time avoids confusion for users. The selector tool is graphically represented as a 3D dotted line with a cursor. The line and cursor dynamically raycast into the environment to land on top of objects when hovered. This prevents them from going through objects and provides visual feedback on the currently selected object.

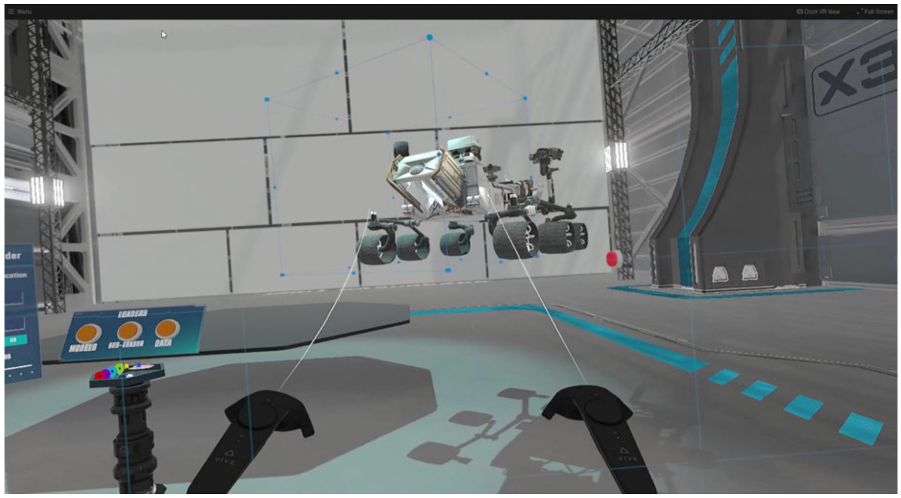

Figure 21: Using the selector tools on both hands, a user is scaling this rover model.

3.4.2 *Laser Tool*

The laser tool is a communication tool that works similarly to the selector tool. The laser tool was built in parallel to real world laser pointers, often used in presentations. When enabled, pressing the trigger, creates a laser shooting out of the right hand. It can be used to point at any virtual objects or parts of the vMCC environment. For example, a user might point at specific parts of a rover model to highlight damage, or point towards data points on a graph that is visible on the joint dashboard. This laser is visible to all of the other users raycasted onto the appropriate object that's being gestured towards.

3.4.3 *3D Draw Tool*

The 3D Draw tool is a 3D marker that allows for drawing and annotating the virtual environment. Users can start painting into the real world at any location. This can be useful for drawing arrows, a path onto the terrain, or even relationships between different visual elements. The 3D maker also has customizable colors such that each user can differentiate their annotations. An eraser functionality

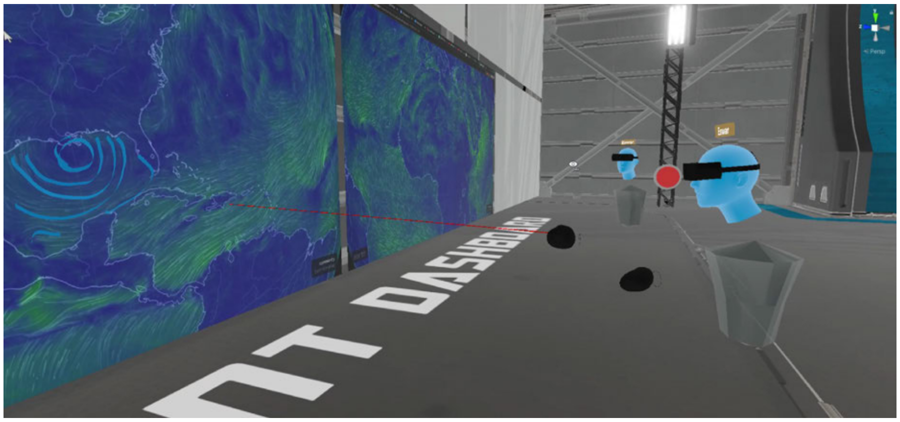

Figure 22: Two users discussing at the joint dashboard while using the laser tool to point out features.

enables erasing the last stroke or erasing all drawings in the environment. Compared to the other annotation tools in vMCC, the 3D draw tool is primarily for large scale, less precise annotation.

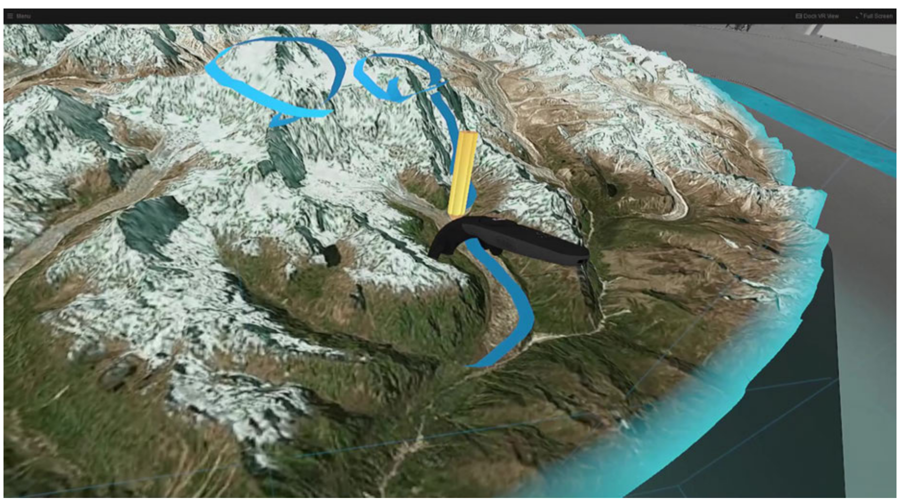

Figure 23: Using the draw tool over the map to mark out regions of interest and a traverse path

3.4.4 *Paint Tool*

Inspired by texture painting tools in other 3D visualization software, the paint tool is another annotation tool. When utilized near or above an object, it will create textured decals onto the object itself. The paint

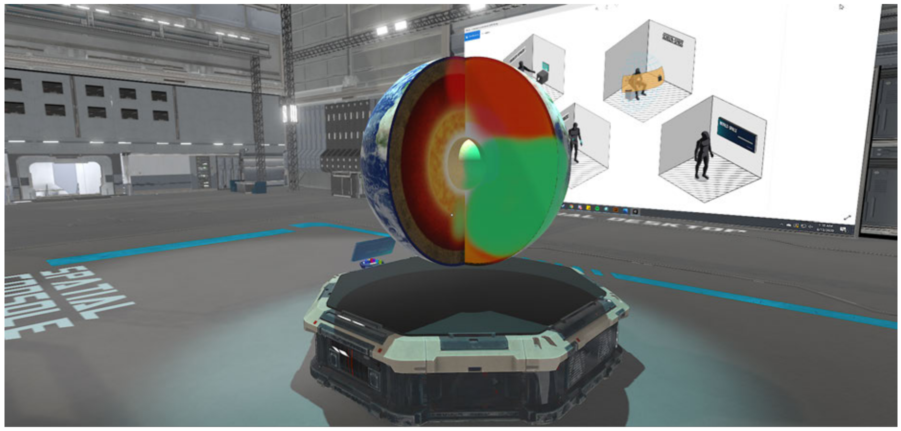

Figure 24: The paint tool is being used here to indicate regions of interest. After loading a model of the earth with a cutout of the core, a user has highlighted regions of interest in green and red.

tool operates like a real world spray paint canister. While the tool is used in 3D space, the output is actually in 2D, on the nearby surface of an object. It projects the motion and textures onto that object to draw on it.

As an example, one potential use case is for marking out regions of interest on a terrain. One area on the ridge of the mountain has high calcium deposits while the lower valley has large amounts of sub-surface hydrogen. With the paint tool, the user can paint these regions in different colors to differentiate them. In contrast to the 3D Drawing tool, this is a much better tool for this application. With the 3D draw tool, the precision would be low and the drawings would get in the way of observing the geographical features of the marked out regions. Figure 24 shows the paint tool being used highlight that areas of interest for all users to view.

3.4.5 *Whiteboard*

The whiteboard is a virtual replica of the real world whiteboard commonly used in meeting rooms and collaborative spaces. A whiteboard and a set of markers are available for users to use on one side of the vMCC hangar. The markers and drawings are all synced and networked across clients for collaborative editing. Similar to the paint tool, the whiteboard is a more precise alternative to the general purpose 3D draw tool. New concepts and diagrams can be quickly drawn on the whiteboard. Flowcharts, mind maps, bullet points, and more are great examples of whiteboard usage. This is one of the most familiar tools to provide users.

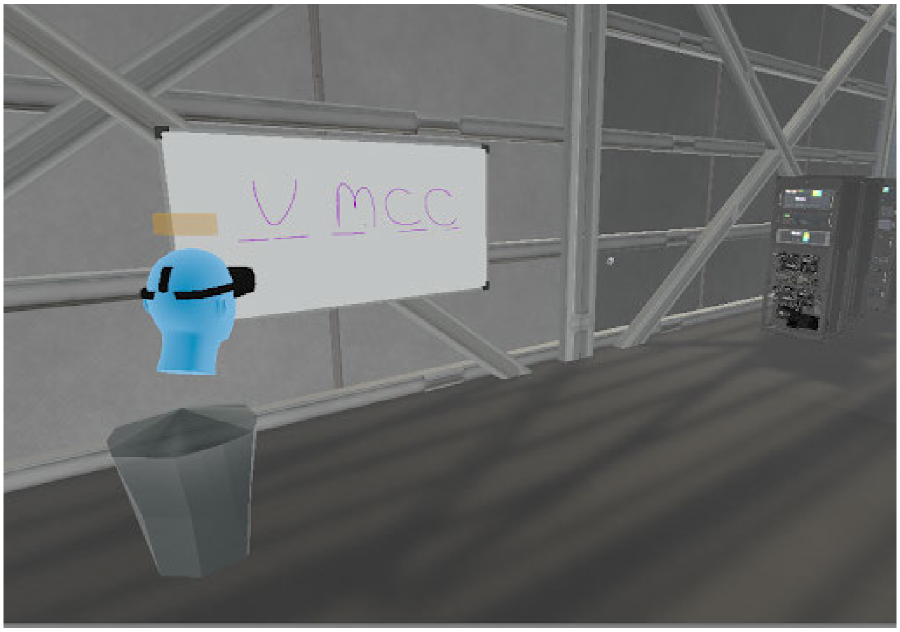

Figure 25: Using the whiteboard

Let's take an example with all three tools in usage together. On completion of a rover sampling mission, users are analyzing mission data. Utilizing the paint tool scientists mark out regions of interest that they can categorize based on collected data. The 3D paint tool is used to draw an arrow towards a specific sampling site and the rover path taken to get there. The whiteboard is used for sketching out the entire mission timeline from start to end. Each tool provides unique affordances and comes with its own limitations. Combined, these three tools enable users to tackle almost any annotation task.

3.4.6 *Keyboard and Mouse*

As previously mentioned, for interacting with the virtual desktop, a virtual keyboard and mouse are available. The virtual keyboard can also be used for other interactions such as text fields. A regular Bluetooth or wired keyboard can also be used as input for all of the text needs. However, this can be difficult while wearing a headset. This led to the creation of the virtual keyboard. It's modeled after real world drum sets rather than traditional keyboards. The vMCC implementation currently does not utilize hand tracking, and needs to use the hand controllers. Turning the hand controllers into a set of virtual mallets to press on a large keyboard was a much more natural usable interface. It was convenient, faster, and easier to use a mallet-based keyboard rather than one based on pointers. A pointer based keyboard would require users to use their selector tool to press the keys as if they were buttons. The mallet based approach using physics colliders proved to be a better interface. The keyboard is designed for short typing use cases making speed a non-factor.

The keyboard can be pulled up at any point from the hand toolkit. It spawns into a convenient location in front of the user. They can grab the keyboard by its handle and move it around as well as resize for their comfort and convenience. The mouse for the virtual monitor is similar to the selector tool. It raycasts the selector ray onto the screen and projects the mouse position accordingly. Sending this info to the system allows for utilizing the mouse from vMCC.

Early experiments were conducted in virtual keyboards incorporating hand tracking as well as tracked physical keyboards. This is an under-explored space in virtual interactions. Ultimately, because vMCC does not have any typing heavy tasks, and virtual monitor tasks will likely be short, this virtual keyboard implementation was satisfactory without needing additional hardware or complexity. Other interesting concepts such as swipe keyboards, voice input, radial typing interfaces and more were considered.

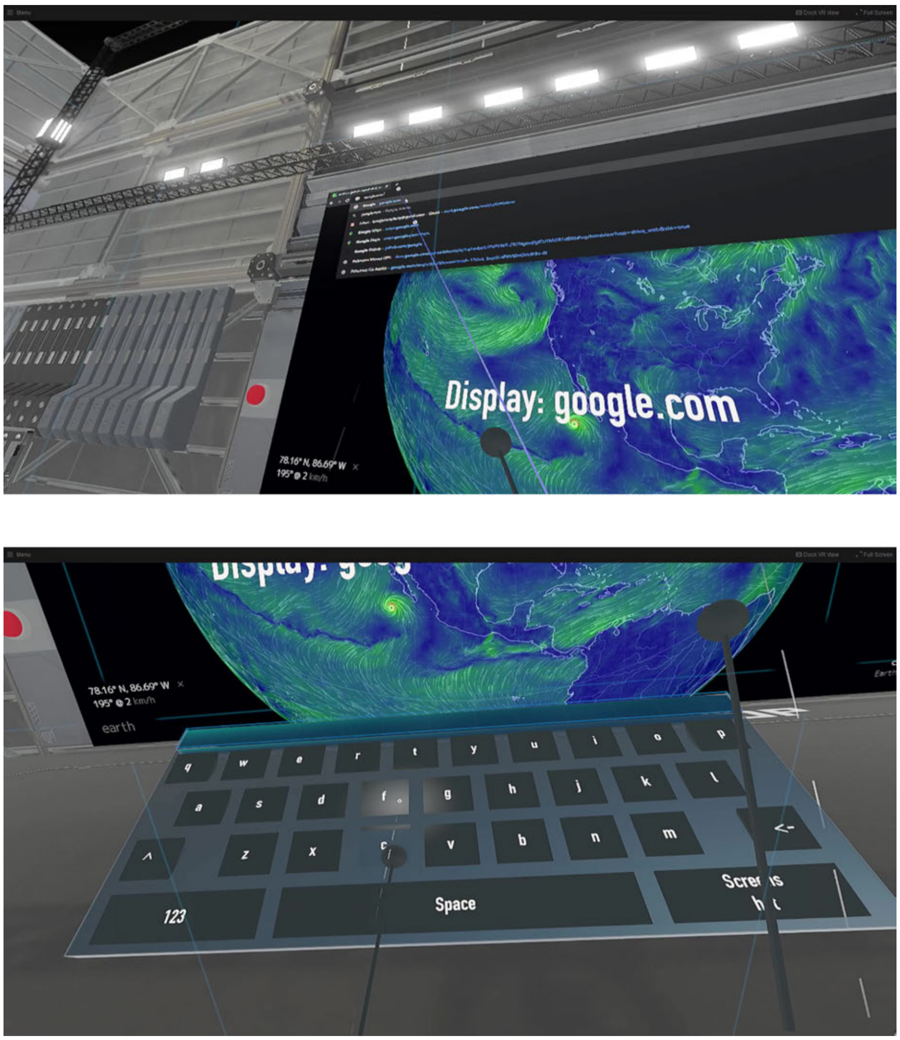

Figure 26: Virtual Keyboard and mouse. Top image shows the mouse ray being used to select the url input field. Bottom image shows virtual keyboard being used to type in URL.

3.4.7 *Presentation Camera*

Presenting data and analysis is a core goal of the vMCC user. After understanding the data from a mission, presenting it to a remote science-tactical expert, or other stakeholders is a critical function. For this purpose, vMCC has a presentation mode camera. Using presentation mode, users inside of vMCC can present to an audience using traditional 2D video conferencing software. The presentation camera acts exactly like a real world camera would. The virtual camera, once enabled, is a 3D camera object that can be moved, angled, and placed. Once it's in the intended position, users can see what the view looks like and start streaming. This stream is directly outputted to the computer as a webcamera. This viewport into vMCC acts just like a regular webcam would. Users can select the virtual camera as their webcam on videoconferencing software or streaming software rather than their traditional webcam. An audience can take a step into vMCC and watch as other users in VR use the tools and data viewer modalities to present. A team of scientists can setup the environment with their data, models, and findings. As they present, they move the camera through the environment to show the details, annotating the data as they talk.

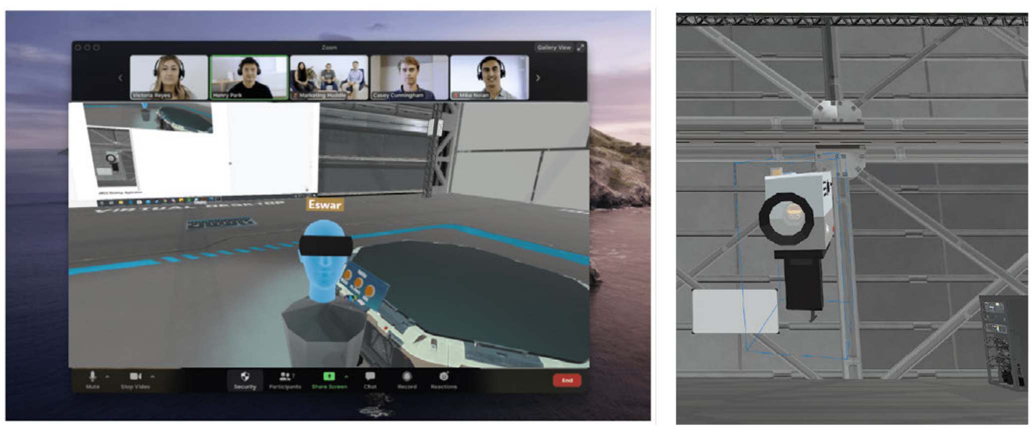

Figure 27: Presentation camera being used for a video call. On the right is the virtual camera in vMCC. On the left is the same camera streaming to Zoom as a user presents to the camera and audience.

The presentation camera can also be programmed with more complex features. It can be set on auto-follow mode where it will follow the user's avatar at a fixed distance creating seamless presentations where the user can move across the hangar while presenting. Preplanned paths can also be created for dynamic streamed visuals.

With the presentation camera a group of vMCC users can bring in an even larger audience of teleconferenced virtual viewers. Leveraging VR spatial visualization capabilities and tools, combined with modern video conferencing software, users can present models, data, and talk in an expressive manner going beyond what's possible in presentation software like Powerpoint. The presentation camera creates a bridge between VR and traditional desktop/laptop users.

3.5 VMCC DESKTOP APPLICATION

Having access to multiple VR headsets can be a challenge today with VR yet to reach ubiquity. We developed a 2D desktop application as a lightweight version of the vMCC VR application for this reason. While the presentation camera enables a team to present to an audience of passive viewers, the desktop mode would allow basic functionality and interactivity to its users. Arrow keys and mouse movement control movement in the environment and camera angle. It also includes full controller simulation capabilities, to point and select objects as well as use tools. Given that vMCC was developed for VR HMDs, there are definite limitations in the 2D mode. We intend this to be minimally used for users who don't have access to a VR headset and need to partake in a meeting.

ATTRIBUTE	VR	2D DESKTOP
Immersion	Yes	No
Stereo 3D Display	Yes	No
Head Movement	Yes	No
Interaction	Yes(controllers)	Yes(mouse and keyboard)
Input Actions	Trigger, buttons, motion	Mouse, keyboard
Body Language	Yes	No

Table 1: A comparison of VR and 2D Desktop Application

The desktop mode can be installed on any Windows 10 machine opening up vMCC to a wide range of users without needing to invest in equipment and training.

4 USER INTERFACE

Human factor design principles, and User Experience standards have been well established for 2D traditional interfaces. With modern immersive technologies, such as VR, there is a lot more uncertainty. In three dimensional space, there are additional constraints to think about in the design process.

In 1995, user advocate and computer scientist Jakob Nielsen created 10 general principles for interaction design known as the 10 usability heuristics (Nielsen, 1993). They are as follows:

1. Visibility of system status
2. Match between system and the real world
3. User control and freedom
4. Consistency and standards
5. Error prevention
6. Recognition rather than recall
7. Flexibility and efficiency of use
8. Aesthetic and minimalist design
9. Help users recognize, diagnose, and recover from errors
10. Help and documentation

Nielsen's heuristics can be applied to virtual reality quite easily. All of these principles hold true for all interfaces regardless of medium. Keeping these in mind, I designed a natural user interface for vMCC. The interface would mimic real world interactions as close as possible to create learnable interactions, and an easy user experience. Discoverability and ergonomics come to even more of a consideration in VR when compared to two dimensional software. Interfaces must be easily discoverable and accessible to users when they can be anywhere in the environment. Proper feedback is also necessary for any well-designed interaction. The lack of feedback, especially in motion based interactions can leave the user confused because the application's state is not clear. Visual feedback prevents accidental user interactions where they are unaware of the state and what the application is doing.

The design behind the interface of vMCC creates a satisfying response and is simple to use because of its discoverability. Short tutorials and trials can teach users how to use all of the features and tools. The user experience of vMCC has its roots in the concept of skeuomorphism, fulfilling Nielsen's second heuristic. In his book, *The Evolution of Technology*, George Basalla defines skeuomorphic design as:

> *A skeuomorph is a derivative object that retains ornamental design cues [attributes] from structures that are inherent to the original.*
>
> George Basalla,

Skeuomorphism represents affordances by creating interface objects that mimic their real world counterparts. Early digital design consisted of heavy skeuomorphic elements because it enabled users to easily transition to digital tools. For example, virtual calculators looked exactly like real world calculators to provide the recognition and familiarity to users. In the case of VR, interfaces must mimic familiar 2D interfaces or real world objects. The virtual keyboard mimics a real keyboard, 3D draw tool looks like a pencil, and the presentation camera looks exactly like a real world camera.

The vMCC interface can be split into three different types of coordinate spaces. World-space interfaces are located in the virtual environment and users can move freely while these interfaces are locked in space. They act like persistent virtual objects. Screen-space interfaces are locked to the display, and in this case to the field of view of the user since the display is locked to the users' head. This is traditionally known as a Heads-up-display (HUD). In vMCC, screen-space is avoided because of usability concerns. These visuals can clutter the users' field of view, and often distract from specific activities that they are trying to accomplish. On-demand information through the other coordinate space interface mechanisms is a better approach without causing discomfort for users. Object-space interfaces are locked to specific objects providing relevant information and intractability relevant to that object. Lastly, avatar-space interfaces are locked to the

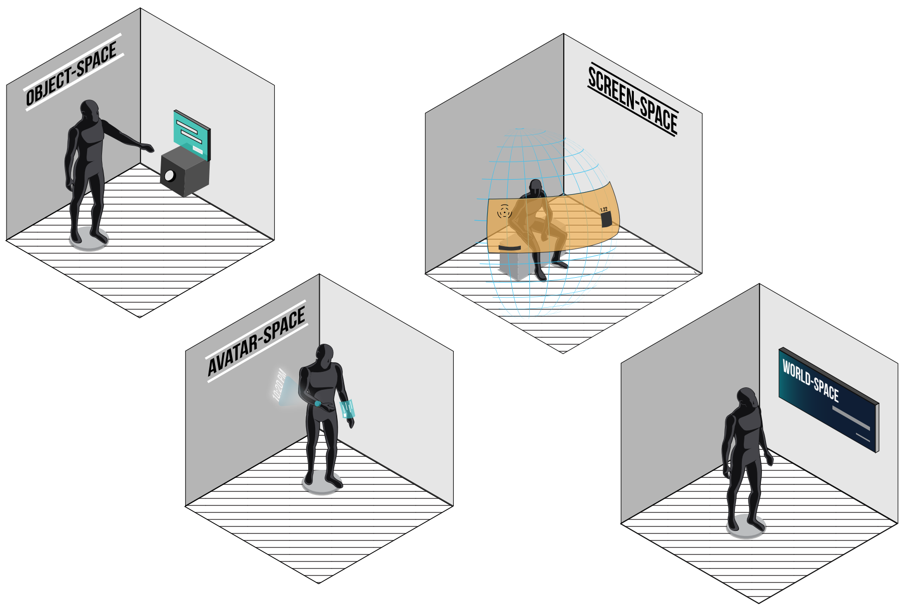

Figure 28: An illustration of the 4 coordinate spaces: World-space, Avatar-Space, Object-Space, and Screen-Space.

user providing the most important functions. These are always available to the user, and can be pulled up with ease.

Many of the interfaces in vMCC are designed as world-space panels. This can be seen with Model-Loader, Geo-Loader and other menus. These menus are locked into a fixed location. In the case of the three loaders, they are conveniently next to the spatial console where users will likely be. These panels also billboard to face the user. Billboarding is a technique where the panels rotate automatically to look at the user. This makes them convenient and easy to use regardless of where the users is located. Interfaces such as the Model-Loader menu bring familiarity with traditional 2D menu systems that users are familiar with. While they have some depth associated with buttons and other elements, they look and feel like traditional web, desktop, or mobile interfaces.

With these menus, the data loaders can be accessed and used. The various annotation and communication tools in vMCC are all clustered into a hand toolkit. The hand toolkit is an avatar-space interface that users can pull up at any point. It is locked to the user's right control. Pressing a button on that controller brings up the hand toolkit, a semi-circle of buttons that each correspond to enabling specific tools. Using their left hand, users can select the appropriate tool they would like to enable. Tools for annotation were constantly needed regardless of data that users were currently viewing. Additionally switching between tools such as drawing and laser pointing was often. A hand based avatar-space interface would be the fastest and most accessible option. The hand-toolkit, similar to the world-space menus, has a billboarding effect. It will always face the user regardless of hand position of the right hand. This allows for easily pressing the buttons with the other hand.

All the models created from the Model-Loader have object-space interfaces. Up close to these models, users can grab them by using a near field interaction. Near field interactions are collision based interactions, where a user can grab an object by simply colliding their controller with the object and grabbing. Far interactions contrast to this direct manipulation as users are standing several feet away. By using the selector tool raycast, they can grab an object they are pointing at. Both of these manipulation methods are imprecise, albeit natural. For precise motion and manipulation, the object-space locked manipulation interface is useful. In 2D visual tools it is common to find handles around an image to scale and rotate it. Extending this paradigm into 3d, each object has a series of handles on and around it. Grabbing a specific handle allows for rotation in that specific axis alone. Likewise, handles for scale can also be used to scale in small intervals. Combined, the near field, far, and object-space handle interactions provide into a variety of options for users to manipulate objects.

Regarding feedback, visual and audio feedback were integrated into all of the interfaces. All interactable elements have hover indicators which show that they can be interacted with, creating discoverability. Buttons can also pull up contextual labels on hover if more information needs to be presented to the user. Pressing a button has distinct feedback to represent the state change and user activity. Color changes, sizes, glow, and audio response all create this feedback. Sliders, buttons, and toggle buttons are all created as volumetric interactable objects. Feedback plays a crucial part in fulfilling Nielsen's principles 1, 4, 5, 7, and 9.

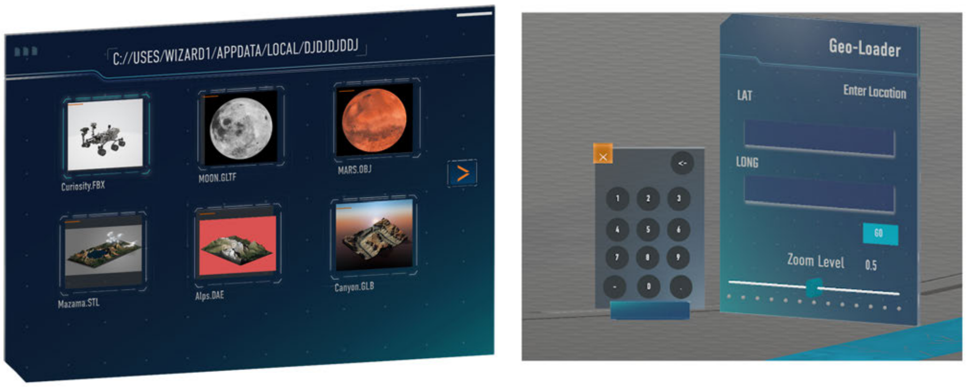

Figure 29: On the left is the Model-Loader UI menu. Each button shows a pin of the model and the name. Selecting it provides visual feedback and loads the model. Right shows the Geo-Loader menu with a number pad next to it.

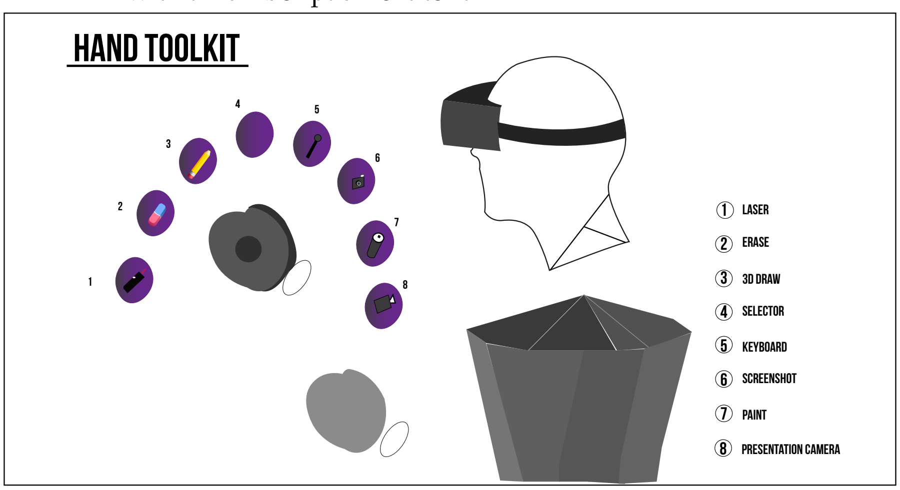

Figure 30: Hand toolkit being used. On the user's right hand are 7 buttons arranged in a semi-circle. This is the hand-toolkit an avatar-space interface.

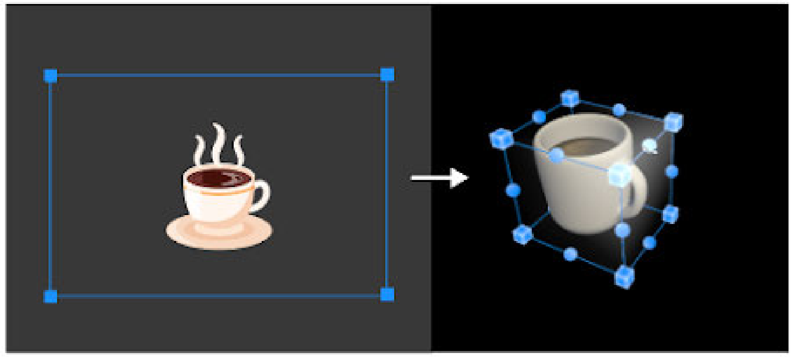

Figure 31: Manipulation handles for objects. On the left is a traditional image with manipulation handles. On the right is a 3D manipulation handle interfaces for finite and preces control.

4.1 CONCLUSION

Interfaces create the bridge between the real world user, their input actions, and the virtual environment. In this chapter, I demonstrated how we integrated Skeuomorphic design based on a series of design principles to create a simple usable interface for vMCC. I also presented the world-space, object-space, and avatar-space interfaces that we built for vMCC and the feedback systems in place for all of these. With VR rapidly becoming more accessible, new forms of natural input methods and being explored. I believe that in the early phases of VR skeuomorphic design is best suited for user experiences. As users become more familiar with VR and the design paradigms get established, it will be possible to design more minimalist, as well as more complex interfaces.

5 THE VMCC WORKING PROTOTYPE

The design and system described in the previous two chapters comes together to form the vMCC first working prototype. This work is foundational for future development of the virtual mission control concept. The architecture as described in Chapter 3 is robust and designed for future expansion. The networking framework can handle almost 5 times the number of concurrent participants vMCC is built for today. All of the tools are cross-platform allowing for porting to more VR hardware and even potentially extending to Augmented Reality. The interface system provides a generalized visual language as well as building blocks for new interactions and functionality. This project was the development of a platform as well as a set of features built on top of this platform. Through future work, it is possible to add several features and capabilities on top of the ones I developed.

It can be difficult to visualize many of the features and capabilities of vMCC explained in this thesis without moving picture visual aids. For the reader to visualize the final prototype, a video demonstration is available. In the demonstration, vMCC as a whole is shown with all features utilized. Both in first person and third person view, the video shows the interfaces, interactions, data viewer systems, and multi-user capabilities.

To view the video demonstration of the vMCC working prototype, visit this link: `https://eswar.io/vmcc-video.html`

5.1 PRELIMINARY USER FEEDBACK

Through the design and development process of vMCC frequent user testing guided the decisions. Several milestones were followed with users trying out features and providing feedback. While performing controlled experiments with a group of users was not possible due to Covid-19 restrictions, informal feedback and user testing guided the iterative design of vMCC. Full scale user studies and experiments are highly recommended for future work and further development.

In investigating the usability of the interface, I found that many users intuitively understood and discovered the buttons, panels, and other input systems. The affordances and design was discoverable to them. In some cases, minor tweaks were necessary where the application did not perform as expected. One example is with far and near field interactions. At first, the distance threshold for a far field interaction to turn into a near field interaction was far too high. Users expected to directly manipulate an object because they were close enough, although the system would not allow for the near field interaction. It would still use the ray based interaction. Based on iterative feedback such as this, we redesigned vMCC to better fit users' needs.

Figure 32: Images from early user testing: Users in Valve Index SteamVR headsets are testing vMCC.

5.2 TOWARDS THE MISSION CONTROL OF THE FUTURE

I have presented vMCC as a data analysis, visualization, and collaboration tool. However, vMCC is a platform for any multi-user mission control applications. The mission control of the future will need far more capability, but this can be built on top of the foundation of vMCC. Real-time data can be piped to be visualized, as well as control directly from the virtual environment. Here is a potential scenario - a team of co-located operators enter vMCC, while a few remote science experts are joining them virtually. The team is partway through a mission and is meeting to make appropriate courses of action. There, they analyze real-time rover or human EVA data to make decisions about the next mission objective. After deciding, they can directly communicate or send this decision to the rover or EV team through a real-time data streaming pipeline. This could mean direct control of the rover by placing waypoints on the map and watching a live feed of the rover. Users jump into a first person view from the rover's vantage point in full 3D 360. Effectively, they can be on Mars virtually to watch the mission while controlling it from Earth. An operations example like this shows the potential for a virtualized mission control fully built with end-to-end features. My development of vMCC was focused on the data analysis and visualization use cases, but the platform can be extended to play a part in all parts of a mission.

The vMCC system can also be very powerful for training applications as well as pre and post mission analysis. From a training angle, the environment within vMCC can be reconfigured to fit any physical mission control space. Even without access to the space, potential operators and trainees could train together virtually. Simulations of past missions can be conducted with data from that period and tests could be conducted to compare trainees between sessions.

For the data analysis use case, vMCC can provide a holistic view of a mission's data. Let's take an example of analyzing the post-mission data of an earth analog EVA operation such as BASALT. Users can start with an Earth model, looking at the various EVA sites from a very zoomed out angle. Following this, they can "zoom into" the terrain of the specific mission location and see the various days of operation and where those traverses occurred. They zoom further into one specific day, looking into the traverse of the users and the sampling sites. In parallel they load up the instrument data charts and images of samples collected. Without context switching, the users can easily look at all of this in different parts of the room and collaborate. Together, with all this data, they can see the holistic picture of the mission understanding how each sample, traverse, and mission fits into the overall objectives. This simply would not be possible with traditional 2D interfaces alone. I hope that further exploration of the virtual mission control concept, and development of vMCC can continue to explore how immersive technologies can be an integral part of these collaborative decision making environments. When mission critical decisions are being made in environments of tight constraints, empowering the scientists, engineers, operators, and tactical experts with better tools and technology can yield better mission goals.

6 CONCLUSION

In this thesis I claim that virtual reality and immersive technology can be used to augment data visualization in a mission control setting. First I outlined background work stemming from multiple fields starting with establishing a concept of operations. Future planetary missions will have numerous constraints and difficulties requiring an improved toolset for astronauts and operators on Earth. Motivated by this, I reviewed two NASA Analog missions, their mission architectures, and software they used. Taking a bigger picture view, I reviewed the complexity of mission control and some of the underlying cognitive needs and theories. Finally, I outlined work in parallel sectors utilizing virtual reality and benefits that were identified from these projects.

After this background review, I presented the vMCC system, an exploration towards the vision of augmented mission controls. The vMCC system unifies various representations of data creating a centralized virtual environment for all data visualization. It's a multi-user environment that fosters collaboration and exploratory data analysis. I described the various data-viewing modalities and tools that exist for users to interact. Lastly, I described the design considerations, interface implementation, and user experience.

These contributions are a small but necessary step towards the vision of an augmented virtual mission control. My working prototype can act as a foundational piece of software to build features and tools on top-of to further explore experimental mission control architectures. This prototype has several areas for improvement and expansion.

6.1 FUTURE WORK

6.1.1 *Short Term*

In the short term, improvements on vMCC can be made in a few key areas: avatars, ingestible data formats, experimental design and interface improvements.

Increasing the data format compatibility is crucial for being an integrated solution. Importing more 2D formats and creating visuals for these can help replace existing tools that users already use. This includes geo-tagged instrument data, rover path traverses, and other forms of mission data. As described before, completing a thorough experimental design and user study can help guide future design and interface improvements. Lastly, a customizable avatar system could bring more expressive emotion for communication. Lip motion and other facial expressive capabilities can increase sense of presence for users. In the future, these improvements would help make vMCC more user-friendly and capable.

6.1.2 *Long Term*

In the long term, the potential for vMCC is limitless and there are many steps that can be taken. I propose a few key design and development directions that can improve vMCC. In Chapter 3, I describe my decision to develop vMCC for the Vive Pro platform. While a great tool for high fidelity VR experiences, porting vMCC towards portable mobile VR headsets such as the Oculus Quest can open up far more possibilities. The Oculus quest system, as an example, is fully standalone and wireless, requiring no PC or external sensors. A fleet of 20 Oculus quests can be deployed in an analog mission with relative easy and minimal cost when compared to PC based VR systems such as the Vive Pro. On the other hand, porting vMCC to the Oculus Quest will require significant optimization and performance upgrades to work properly. Creating an optimized visualization tool that can run on mobile hardware would enable easier user studies and improve usability for scientists who aren't currently familiar with virtual reality as a tool.

The second design direction I would recommend is pursuing more complex data analysis tools and capabilities. Traditional 2D software such as Matlab, R, Tableau, Excel, and more can provide a variety of data analysis features. For example, taking raw data and creating regressions, and running statistical tests, and predictive extrapolation are common tasks. The current vMCC implementation of importing and visualizing raw data, can be extended to adding in these mathematical and analytic tools. Adding in predictive modeling capabilities and advanced data analysis techniques can provide even more insight for users.

The last design direction I would recommend is exploring novel input interfaces. Virtual reality bridges the physical world with the virtual world. Most VR applications use hand controllers as the sole input device. Hand controllers can be cumbersome and difficult to use, especially for precise control. Similarly, tasks like drawing and writing depend on the physical affordances of objects like pens and markers. It is much more difficult to write precisely with a controller as it is much larger and differently shaped than a pen. Experiments, prototypes, and investigation into novel input devices can lead to more productive VR tools. Hand tracking, VR styluses, and tracked drawing tablets are examples of potential directions. Input interfaces, as a whole, is an under-explored research area in virtual reality but can offer increased functionality in vMCC.

6.2 CLOSING THOUGHTS

As with any novel technological innovation, it is important to continue experimenting in VR while testing iteratively with users. The vMCC system is a prototype and proof-of-concept for future mission control architectures. There are still many open unanswered questions:

- Where is VR complementary to existing tools and what is the quantitative benefit that VR can offer?
- How does a remote team of operators using virtual tools compare to a co-located mission control?
- What does the cognitive load impact of mission control look like when using spatial 3d interfaces?

Answering questions like these will guide future development of vMCC. This project hopefully shows that it is worth the time, effort, and money to invest in virtual tools such as this. Based on this research, my optimism remains strong that endeavouring into experimental tools can empower our scientists and operators to make better decisions and perform better science. Future work can build on the foundation of vMCC to bring us closer to the human-computer symbiosis envisioned by the early computer pioneers over 50 years ago.

[1] Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The LaTeX Companion*. Addison-Wesley, Reading, Massachusetts, 1993.

[2] Albert Einstein. *Zur Elektrodynamik bewegter Körper*. (German) [*On the electrodynamics of moving bodies*]. Annalen der Physik, 322(10):891–921, 1905.

[3] 1991NASCP3103..236. Page 236. http://articles.adsabs.harvard.edu//full/1991NASCP3103..236./0000236.000.html

[4] Alger, Mike. *Visual Design Methods for Virtual Reality*. Sept. 2015 https://pdfs.semanticscholar.org/94d9/115ffc204c4a904312d6cecaf3032101009c.pdf.

[5] Anandapadmanaban, Eswar, et al. "Holo-SEXTANT: An Augmented Reality Planetary EVA Navigation Interface." Proceedings of the 48th International Conference on Environmental Systems, 2018. ttu-ir.tdl.org, https://ttu-ir.tdl.org/handle/2346/74212.

[6] Atkinson and Shiffrin | Multi Store Model of Memory | Simply Psychology. https://www.simplypsychology.org/multi-store.html.

[7] Augmenting Human Intellect: A Conceptual Framework - 1962 (AUGMENT,3906,) - Doug Engelbart Institute.https://www.dougengelbart.org/content/view/138.

[8] Azuma, Ronald T. "A Survey of Augmented Reality." Presence: Teleoperators and Virtual Environments, vol. 6, no. 4, Aug. 1997, pp. 355–85. DOI.org (Crossref), doi:10.1162/pres.1997.6.4.355.

[9] "The Challenge of Making Augmented Reality Work Outdoors." Mixed Reality, edited by Yuichi Ohta and Hideyuki Tamura, Springer Berlin Heidelberg, 1999, pp. 379–90. DOI.org (Crossref), doi:10.1007/978-3-642-87512-0_21.

[10] Billinghurst, Mark, et al. "A Survey of Augmented Reality." Foundations and Trends® in Human–Computer Interaction, vol. 8, no. 2–3, 2015, pp. 73–272. DOI.org (Crossref), doi:10.1561/1100000049.

[11] Carberry, Chris, et al. Mixed Reality Technologies Supporting Space Exploration. White Paper, Explore Mars Inc., 2018, https://www.exploremars.org/wp-content/uploads/2019/12/H2MMR-Whitepaper-FINAL.pdf.

[12] Deans, Matthew, David Lees, et al. Field Testing Next-Generation Ground Data Systems for Future Missions. Mar. 2011.

[13] Deans, Matthew, Tamar Cohen, et al. Tools for Enabling Real Time Volatile Prospecting with Surface Rovers. 2015.

[14] Engelbart: Program on Human Effectiveness (1961). http://web.stanford.edu/dept/SUL/library/extra4/sloan/mousesite/Archive/Post68/PrHumanEffectiveness.html.

[15] Feng Zhou, et al. "Trends in Augmented Reality Tracking, Interaction and Display: A Review of Ten Years of ISMAR." Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality, IEEE Computer Society, 2008, pp. 193–202. ACM Digital Library, doi:10.1109/ISMAR.2008.4637362.

[16] Fisher, S. S., et al. "Virtual Environment Display System." Proceedings of the 1986 Workshop on Interactive 3D Graphics - SI3D '86, ACM Press, 1987, pp. 77–87. DOI.org (Crossref), doi:10.1145/319120.319127.

[17] Fisher, S.S., et al. "Virtual Interface Environment Workstations." Proceedings of the Human Factors Society Annual Meeting, vol. 32, no. 2, Oct. 1988, pp. 91–95. DOI.org (Crossref), doi:10.1177/154193128803200219.

[18] Fisher, Scott S. "VIRTUAL ENVIRONMENTS, PERSONAL SIMULATION TELEPRESENCE." Virtual Reality: Theory, Practice and Promise, Meckler Publishing, 1991, https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.467.7055&rep=rep1&type=pdf.

[19] Granaas, Michael. "The Psychology of Computer Displays in the Modern Mission Control Center." 15th Aerodynamic Testing Conference, American Institute of Aeronautics and Astronautics, 1988. DOI.org (Crossref), doi:10.2514/6.1988-2065.

[20] Greenwald, Scott Wilkins. The Equipped Explorer: Virtual Reality as a Medium for Learning. Massachusetts Institute of Technology, Feb. 2018.

[21] Greicius, Tony. "Mars Virtual Reality Software Wins NASA Award." NASA, 2 Oct. 2018,http://www.nasa.gov/feature/jpl/mars-virtual-reality-software-wins-nasa-award.

[22] Heindel, Troy A., et al. "Advanced Automation in Space Shuttle Mission Control." Fourth Annual Workshop on Space Operations Applications and Research, vol. (SOAR '90)., 1991, http://articles.adsabs.harvard.edu//full/1991NASCP3103..236./0000236.000.html.

[23] Heldmann, Jennifer L., et al. "Lunar Polar Rover Science Operations: Lessons Learned and Mission Architecture Implications Derived from the Mojave Volatiles Prospector (MVP) Terrestrial Field Campaign." Advances in Space Research, vol. 58, no. 4, Aug. 2016, pp. 545–59. DOI.org (Crossref), doi:10.1016/j.asr.2016.05.011.

[24] NASA-TM-2005-214062, NASA, Nov. 2005, https://www.nasa.gov/exploration/news/ESAS_report.html.

[25] Kobs Nawotniak, Shannon E., et al. "Opportunities and Challenges of Promoting Scientific Dialog throughout Execution of Future Science-Driven Extravehicular Activity." Astrobiology, vol. 19, no. 3, Mar. 2019, pp. 426–39. DOI.org (Crossref), doi:10.1089/ast.2018.1901.

[26] Krueger, Myron W. "Interfacing Realities to the Human Body (Abstract)." Proceedings of the 1993 ACM Conference on Computer Science - CSC '93, ACM Press, 1993, p. 508. DOI.org (Crossref), doi:10.1145/170791.171136.

[27] LeapMotion. "6 Principles of Leap Motion Interaction Design." Medium, 6 Sept. 2016, https://medium.com/@LeapMotion/6-principles-of-leap-motion-interaction-design-98bddff77a53.

[28] Lessig, Lawrence. Code and Other Laws of Cyberspace. Basic Books, 1999.

[29] Lim, Darlene S. S., et al. "The BASALT Research Program: Designing and Developing Mission Elements in Support of Human

Scientific Exploration of Mars." Astrobiology, vol. 19, no. 3, Mar. 2019, pp. 245–59. DOI.org (Crossref), doi:10.1089/ast.2018.1869.

[30] Marquez, Jessica J., Steven Hillenius, et al. "Enabling Communication Between Astronauts and Ground Teams for Space Exploration Missions." 2019 IEEE Aerospace Conference, IEEE, 2019, pp. 1–10. DOI.org (Crossref), doi:10.1109/AERO.2019.8741593.

[31] Marquez, Jessica J., Matthew J. Miller, et al. "Future Needs for Science-Driven Geospatial and Temporal Extravehicular Activity Planning and Execution." Astrobiology, vol. 19, no. 3, Mar. 2019, pp. 440–61. DOI.org (Crossref), doi:10.1089/ast.2018.1838.

[32] Mattern, Shannon. "Mission Control: A History of the Urban Dashboard." Places Journal, no. 2015, Mar. 2015. DOI.org (Crossref), doi:10.22269/150309. —. "Mission Control: A History of the Urban Dashboard." Places Journal, Mar. 2015. placesjournal.org, doi:10.22269/150309.

[33] Meigs, Bruce E., and Larry L. Stine. "Real-Time Compression and Transmission of Apollo Telemetry Data." Journal of Spacecraft and Rockets, vol. 7, no. 5, May 1970, pp. 607–09. DOI.org (Crossref), doi:10.2514/3.29999.

[34] Mori, Masahiro, et al. "The Uncanny Valley [From the Field]." IEEE Robotics Automation Magazine,vol. 19, no. 2, June 2012, pp. 98–100. IEEE Xplore, doi:10.1109/MRA.2012.2192811.

[35] "NASA Chooses Artec Space Spider to Create Ultra-Precise Models (up to 50 Microns) for Its Hybrid Reality Astronaut Training." Professional 3D Scanning Solutions | Artec3D,`https://www.artec3d.com/cases/nasa-hybrid-reality-lab.`

[36] NASA Technical Reports Server (NTRS).`https://ntrs.nasa.gov/citations/19880014769.`

[37] NASA's Plan for Sustained Lunar Exploration and Development.NASA, 2020.

[38] NASA's Virtual Reality Lab For Astronaut Training - YouTube. `https://www.youtube.com/watch?v=cwK3MatOQFc.`

[39] Niehorster, Diederick C., et al. "The Accuracy and Precision of Position and Orientation Tracking in the HTC Vive Virtual Reality System for Scientific Research." I-Perception, vol.

8, no. 3, June 2017, p. 204166951770820. DOI.org (Crossref), doi:10.1177/2041669517708205.

[40] Nielsen, J. "Iterative User-Interface Design." Computer, vol. 26, no. 11, Nov. 1993, pp. 32–41. DOI.org (Crossref), doi:10.1109/2.241424.

[41] Norheim, Johannes. Path Planning for Human Planetary Surface Exploration in Rough Terrain. Massachusetts Institute of Technology.

[42] Patterson, Emily S., et al. "Voice Loops as Coordination Aids in Space Shuttle Mission Control." Computer Supported Cooperative Work (CSCW), vol. 8, no. 4, Dec. 1999, pp. 353–71. DOI.org (Crossref), doi:10.1023/A:1008722214282.

[43] Payler, S. J., et al. "Developing Intra-EVA Science Support Team Practices for a Human Mission to Mars." Astrobiology, vol. 19, no. 3, Mar. 2019, pp. 387–400. DOI.org (Crossref), doi:10.1089/ast.2018.1846.

[44] "Restore Apollo Mission Control Center." Space Center Houston, `https://spacecenter.org/support/restore-apollo-mission-control/`.

[45] "Restoring Apollo Mission Control Center." Space Center Houston, `https://spacecenter.org/restoring-apollo-mission-control-center/`.

[46] Rometsch, Flavie. Design of an AR-IoT Tool for Future Human Space Exploration. TU Delft, Aug. 2020.

[47] Scott S Fisher - Portfolio of Work. `http://itofisher.com/sfisher/portfolio/files/biography.html`.

[48] Silva, Jennifer N. A., et al. "Emerging Applications of Virtual Reality in Cardiovascular Medicine." JACC: Basic to Translational Science, vol. 3, no. 3, June 2018, pp. 420–30. ScienceDirect, doi:10.1016/j.jacbts.2017.11.009.

[49] Sutherland, Ivan E. "A Head-Mounted Three Dimensional Display." Proceedings of the December 9-11, 1968, Fall Joint Computer Conference, Part I on - AFIPS '68 (Fall, Part I), ACM Press, 1968, p. 757. DOI.org (Crossref), doi:10.1145/1476589.1476686.

[50] Tamar Cohen, et al. Conducting Efficient Remote Science and Planning Operations for Ocean Exploration Using Exploration Ground Data Systems (XGDS). 2020, https://ntrs.nasa.gov/citations/20200001567.

[51] "VR Interface Design Manifesto." Vimeo,https://vimeo.com/116101132.