

EVALUATION OF FLIGHT CONTROL TECHNIQUES USING VIRTUAL
REALITY IN AN ARTIFICIAL GRAVITY ROTATING ENVIRONMENT

A Thesis

by

ROBERT DALLAS HOGAN

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Chair of Committee,	Gregory Chamitoff
Committee Members,	Nancy Currie-Gregg
	Kristi Shryock
Head of Department,	Rodney Bowersox

August 2019

Major Subject: Aerospace Engineering

Copyright 2019 Robert Dallas Hogan

ABSTRACT

The purpose of this thesis is to develop a collaborative virtual reality (VR) engineering platform for space system and mission design and to demonstrate its utility in the context of evaluating human interfaces for future control systems. Much of the work for this project was the development of the platform itself (called SpaceCRAFT). Additionally, a user study was done with 33 subjects to examine the potential advantages of designing and testing systems in virtual environments that otherwise may be difficult to replicate on Earth. The task evaluated was flying a drone in a rotating artificial gravity environment, which involves numerous unfamiliar forces. Different control strategies were tested using VR in comparison to flat screen interfaces. This particular challenge was chosen to emphasize the difference between immersive and non-immersive environments, and the results demonstrate that VR is a promising tool for human-interface system design and evaluation.

The 50-meter radius space station simulated an open-air, long-term habitable environment and was designed with considerations of human factors for rotating reference frames. A quadrotor control model was developed and simulated a variety of stabilization and sensitivity modes. Subjects piloted the drone through an obstacle course to evaluate flight characteristics for each configuration. The Modified Cooper-Harper Rating Scale for Handling Qualities was used to measure participants' subjective rating of these characteristics. The average rating on the 10-point scale was 0.24 points better using VR over traditional monitor viewing. Objective performance based on displacements from the

path was measured to be up to 11% more favorable for VR with 95% confidence. Both of these results indicate advantages for user preference and user performance while utilizing immersive simulations. No notable correlations were found between experience levels.

ACKNOWLEDGEMENTS

First, I would like to express my gratitude towards Dr. Greg Chamitoff; without him, my continued education would not have been possible and my passion for space system design may have never developed. He has been both inspiring and patient with me throughout my time as a graduate student at Texas A&M University.

Second, I would like to thank my committee members, Dr. Nancy Currie-Gregg and Dr. Kristi Shryock, for their advice and support.

Finally, I would like to thank all of my friends and family members, old and new; each of you have impacted my life and molded me into who I am today.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

Part 1, faculty committee recognition

This work was supervised by a thesis committee consisting of Professor Chamitoff (advisor) and Professor Shryock of the Department of Aerospace Engineering and Professor Currie-Gregg of the Department of Industrial Engineering.

Part 2, student/collaborator contributions

The 3D models for the space station and quadrotor were created in part by Alex Sein in the Department of Aerospace Engineering. The graduate students of the ASTRO Lab in the Department of Aerospace Engineering, the Aggie_Challenge students working with Dr. Chamitoff, and I are responsible for the development of the SpaceCRAFT platform which this research was conducted for.

All other work conducted for the thesis was completed by the student independently.

Funding Sources

This work was made possible, in part, by NASA under the Space Act Agreement, which funded the initial development of SpaceCRAFT. Its contents are solely the responsibility of the author and do not necessarily represent the official views of the NASA.

NOMENCLATURE

3D	3-Dimensional
CAVE	CAVE Augmented Virtual Environments
DOUG	Dynamic Onboard Ubiquitous Graphics
DSES	Distributed Space Exploration Simulation
ECLSS	Environmental Control and Life Support System
EVA	Extra-Vehicular Activity
HITL	Human-In-The-Loop
HMD	Head-Mounted Display
HRPP	Human Research Protection Program
IRB	Institutional Review Board
IQR	InterQuartile Range
ISRU	In-Situ Resource Utilization
JSC	Johnson Space Center
MCH	Modified Cooper-Harper
MMSEV	Multi-Mission Space Exploration Vehicle
NASA	National Aeronautics and Space Administration
PDF	Probability Density Function
ROI	Return-On-Investment
ROS	Robotic Operating System
SES	Systems Engineering Simulator

SRMS	Shuttle Remote Manipulator System
TAMU	Texas A&M University
UE4	Unreal Engine 4
VIVE	HTC VIVE Virtual Reality Head Mounted Display
VR	Virtual Reality
VRL	Virtual Reality Laboratory

TABLE OF CONTENTS

ABSTRACT	ii
ACKNOWLEDGEMENTS	iv
CONTRIBUTORS AND FUNDING SOURCES.....	v
NOMENCLATURE.....	vi
TABLE OF CONTENTS	viii
LIST OF FIGURES.....	xi
LIST OF TABLES	xiii
CHAPTER I INTRODUCTION	1
1.1 Introduction	1
1.2 Motivation – SpaceCRAFT.....	3
1.2.1 Development of SpaceCRAFT	5
1.2.2 Artificial Gravity Simulation	7
CHAPTER II LITERATURE REVIEW.....	8
2.1 Space Mission and Engineering Simulation.....	8
2.2 Space Related VR Studies.....	10
2.3 Quadrotor Related VR Studies.....	12
2.4 Analysis of VR Studies used in Serious Games.....	13
2.5 Institutional Benefits of VR	14
2.6 Rating Scales for Flight Characteristics	16
CHAPTER III SIMULATION DESIGN	18
3.1 Choosing a Simulation	18
3.2 Artificial Gravity Space Station	19
3.3 Simulation Objective.....	23
3.3.1 Quadrotor Flight Mode Selection.....	23
3.3.2 Flight Modes	23
3.3.2.1 Stabilization Modes.....	24
3.3.2.1.1 Attitude-Control	24
3.3.2.1.2 Rate-Control	25
3.3.2.2 Sensitivity Modes	26
3.3.2.2.1 High Sensitivity.....	26

3.3.2.2.2	Low Sensitivity	27
3.3.2.3	Viewing Medium Modes	27
3.3.2.3.1	Traditional Desktop Mode	27
3.3.2.3.2	VR Mode	27
3.4	Physics.....	28
3.4.1	Artificial Gravity	28
3.4.1.1	Implementation.....	29
3.4.2	Quadrotor	30
3.5	Simulation Development Tools.....	31
3.5.1	Blender Modelling.....	31
3.5.2	Unreal Engine.....	32
3.5.2.1	Blueprints	33
3.5.3	C++ and Visual Studios	33
3.6	Hardware	34
3.6.1	HTC VIVE	34
3.6.2	Desktop Computer.....	34
3.6.3	Monitors	35
3.6.4	Xbox One Controller.....	35
CHAPTER IV STUDY DESIGN		36
4.1	Development of the Study.....	36
4.1.1	IRB	36
4.1.2	Structure of the Study Trials	36
4.1.3	Challenges with the Experimental Setup	38
4.2	Simulation Task.....	40
4.3	Subjective Measures.....	42
4.3.1	Modified Cooper-Harper Rating Scale	42
4.3.2	Objective Measures	42
4.3.3	User Study Procedures	43
4.3.4	Questionnaire	46
CHAPTER V STUDY RESULTS		47
5.1	Analysis Tools.....	47
5.1.1	Python.....	47
5.1.2	Pandas.....	47
5.1.3	SciPy.....	48
5.1.4	Matplotlib and Seaborn	48
5.2	Importing Data	49
5.3	MCH Data Analysis	51
5.3.1	Analysis Steps	51
5.3.2	Aggregating MHC Data	52
5.3.3	Normalization of the MCH Results.....	53

5.3.4	MHC Statistics	55
5.3.4.1	Averages	55
5.3.4.2	Standard Deviations	56
5.3.4.3	P-values and the T-Test	56
5.4	Trial Data Analysis.....	70
5.4.1	Analysis Steps	70
5.4.2	Data Reduction	71
5.4.3	Normalization of Average Errors Between Users.....	72
5.4.4	Average Error Statistics	72
5.4.5	Post-Study Questionnaire Results	81
CHAPTER VI CONCLUSION.....		84
6.1	Study Observations	84
6.1.1	Modified Cooper-Harper Rating Discussion	84
6.1.2	Post-Study Questionnaire Discussion	84
6.1.3	Average Error Analysis Discussion	85
6.1.4	Experience Observation	85
6.1.5	Anomaly	85
6.2	Study Influences.....	86
6.3	Study Outcome.....	87
6.4	Future Work	88
REFERENCES.....		90

LIST OF FIGURES

	Page
Figure 1: External View of Hogan Station.....	4
Figure 2: Mars Base Simulation, Built in SpaceCRAFT	6
Figure 3: NASA’s Modified Cooper-Harper Scale for Rating Aircraft Characteristics. .	17
Figure 4: Artificial Gravity Comfort Zone.....	20
Figure 5: Internal View of Hogan Station.....	21
Figure 6: Internal View of Hogan Station with Spline.....	22
Figure 7: 3D Model of the Quadrotor used in the study	32
Figure 8: Drone with Spline	41
Figure 9: Briefing Checklist for User Study Procedures.....	44
Figure 10: Briefing Checklist for User Study Procedures Continued.....	45
Figure 11: Briefing Checklist for User Study Procedures Continued.....	46
Figure 12: Distribution and Histogram of Aggregate MCH Results	60
Figure 13: Pairplot for Hovering Flight Characteristic	63
Figure 14: Pairplot for Prograde Flight Characteristic	64
Figure 15: Pairplot for Retrograde Flight Characteristic	65
Figure 16: Pairplot for Axial Flight Characteristic	66
Figure 17: Pairplot for Cornering/Obstacles Flight Characteristic	67
Figure 18: Boxplot for Attitude-Control Flight Modes and Handling Qualities.....	69
Figure 19: Boxplot for Rate-Control Flight Modes and Handling Qualities	69
Figure 20: Distribution and Histogram of Rate-Control High Sensitivity Mode.....	75
Figure 21: Distribution and Histogram of Rate-Control Low Sensitivity Mode	76
Figure 22: Distribution and Histogram of Attitude-Control High Sensitivity Mode.....	77

Figure 23: Distribution and Histogram of Attitude-Control Low Sensitivity Mode.....	78
Figure 24: Distribution and Histogram for All Trials	79
Figure 26: Box Plots for Attitude-Control Mode	80
Figure 25: Box Plots for Rate-Control Mode.....	80

LIST OF TABLES

	Page
Table 1: Fight Mode Matrix	24
Table 2: First 10 Rows of Data Output from the Simulation for Trial.....	50
Table 3: MCH Rating Results from User BY261	52
Table 4: Normalized MCH Rating Results from User BY261	54
Table 5: Normalized MCH Rating Results for All Users	55
Table 6: Statistic Matrix for Compared Means of MCH Handling Quality Ratings	58
Table 7: Statistic Matrix for Compared Means of Trial Data	74
Table 8: Post-Study Questionnaire.....	81
Table 9: Question 3 Results	82

CHAPTER I

INTRODUCTION

1.1 Introduction

Virtual Reality (VR) and Head-Mounted Displays (HMDs) have been around since the 1960's, starting with the Sword of Damocles, created by Ivan Sutherland, the “father of computer graphics.” [24] Though 50 years has passed since then, computer graphics and processing power necessary to develop and run simulations in VR has only recently become commonly accessible. Off the shelf HMDs, such as the Oculus Rift and HTC VIVE, have helped bring the power of VR to people around the world for the last three years. This has opened the door for VR to be used in research and development for many different fields. Virtual reality for space mission engineering brings the potential for substantial improvement not only in higher fidelity simulations, but also in cost effectiveness. Additionally, certain environments cannot be physically simulated on Earth; building simulations for these environments in VR can be a significant advantage for rapid development and testing without ever leaving the ground.

Three vital components for space systems engineering – system design, human interfaces, and operator performance – benefit greatly from coupled testing in VR. System design, including the physical and analytical models for the environment, physics, and control systems, benefit from one another when investigated in an integrated simulation. The interaction between these models, such as how a vehicle model requires physical measurements from an analytical atmospheric model to determine aerodynamic forces,

requires a cohesive simulation. Human interfaces for systems used in extraterrestrial environments need to be thoroughly scrutinized since mistakes in these environments have higher consequences. Applying human interfaces in virtual environments allows human-in-the-loop evaluations. The final component requires every relevant design and interface to be integrated so end-to-end testing can be done. Using a virtual reality system in an artificial environment is an ideal approach to test operator performance as it allows the user to experience the final design as close as possible to the actual product.

The purpose of this thesis is to develop a collaborative virtual reality (VR) engineering platform for space system and mission design and to demonstrate its utility in the context of evaluating human interfaces for future control systems. The demonstration explores the advantages of using VR simulation for human-in-the-loop evaluations of space system designs. This is studied through the performance of a complex task in an unusual environment. A user study was conducted in which subjects were instructed to fly an unmanned quadrotor drone within a rotating space station, simulating artificial gravity. This particular task was selected for evaluation in a VR environment because the environment cannot be exactly created on Earth due to the pull of gravity. The dynamics of airborne objects moving in this reference frame behave unintuitively due to the Coriolis effect and centrifugal forces, which influence the difficulty of controlling a drone. From the perspective of the pilot, the aircraft has a tendency to rotate in the direction of spin of the space station, as well as translate in a similar manner. Without proper pilot input, the drone will begin drifting and tumbling with altitude variations. The effects of artificial gravity are altered when controlling the drone with or against the rotation of the space

station, furthering pilot mental workload. The subjects were tasked to fly the drone using a range of display and hand control interfaces, sensitivities, and stabilization modes, and then rate the flight characteristics. This simulation incorporated system design, human interfaces, and operator performance design elements in a fully-integrated environment.

The hypothesis of this study asks whether or not utilizing VR for space system design has objective and subjective benefits. The immersion that VR provides is expected to allow the users to perform the task better than with traditional desktop viewing. With the ability to look around the environment, relative to the quadrotor, the users are expected to have a better understanding of their position and velocity. It is also expected that the users will feel more in control of the drone and will rate the flight characteristics better while using VR.

1.2 Motivation – SpaceCRAFT

At present, it is essentially impossible to completely test an entire space mission architecture in any integrated fashion on Earth, especially one with components designed and developed at different institutions worldwide. Future missions of exploration and settlement in space will involve many technologies, systems, and capabilities working together on a solar-system scale. [11] The concept for this thesis comes from the developmental work performed for SpaceCRAFT, a collaborative virtual reality systems engineering platform. This section outlines the development of the architecture of SpaceCRAFT and the potential use of this program as a global space design tool.



Figure 1: External View of Hogan Station

A virtual environment provides a unique ability to visualize engineering systems and simulations. The ability to integrate all of the components of entire mission architectures will provide what rapid prototyping and modelling lack – a means for inexpensive and fast paced development. Spaceflight system development has always been expensive; however, with the right tools and technologies, costs can be minimized while creating higher quality products and mission designs. The platform offers an alternative to traditional mission design and simulation tools, and this thesis was chosen to help determine that value with respect to virtual reality, control systems, and unfamiliar environments.

1.2.1 Development of SpaceCRAFT

My initial role for SpaceCRAFT was to determine how to develop the core architecture of the program and to build a team of researchers to realize the concept. Texas A&M University offers a unique research course entitled AggieE_Challenge. This program connects multi-disciplinary undergraduate students with graduate student mentors with the goal of providing a team to work on relevant topics. I set up and mentored an AggieE_Challenge group and researched latest technologies that could be used to create the vision of SpaceCRAFT. Open-source simulation tools, such as Robotic Operating System (ROS), Trick (NASA's simulation software package), and visualization tools such as the Unreal Engine and Unity programs, among others, were considered. I contributed to each aspect of the program including backend development, simulation structures, model creation, and more. This provided knowledge necessary to continue guiding the teams towards the vision of SpaceCRAFT.

SpaceCRAFT is intended to be a collaboration resource for the entire space community. Creating integrated simulations that include numerous technical disciplines is a difficult task. In order to create a product that will not only be exciting to use, but also be capable of providing intuitive and meaningful results, SpaceCRAFT needs to exceed the limitations of existing simulation software. One distinctive characteristic of SpaceCRAFT is its modular nature. Different aspects of the program are compartmentalized to reduce complexity and allow users to reuse and adapt work to their unique purposes. I developed a set of standards for various mathematical models to allow modules to be replaced seamlessly. Instead of having to create every model required to

test one specific system within an environment, a user would be able to focus on their system while using environmental models provided by the platform or other users. This is what SpaceCRAFT aims to provide: a robust platform with the integration capabilities necessary to run complex interactive simulations.

During the early development stages of SpaceCRAFT, I directed senior level design courses, which contributed to the program's growing library of modules and mathematical simulations. I developed guidelines for each of these projects to connect to one-another within the SpaceCRAFT architecture and guided the student teams throughout the courses. The largest project was developing an entire Mars mission architecture – from telecommunications to robotic operations and habitat design, inter-system communication was vital to the success of the mission.



Figure 2: Mars Base Simulation, Built in SpaceCRAFT

1.2.2 Artificial Gravity Simulation

The simulation for this thesis was chosen as a proof-of-concept scenario to showcase virtual reality's utility in the evaluation of human interfaces for control systems in extra-terrestrial environments. I designed a space station with physical attributes acceptable for human habitation, developed control modes and interface designs for a quadrotor drone, and implemented a task-based scenario to test. I developed a user study to test the simulation, attaining both subjective and objective data towards demonstrating the value of mission and system design in VR. The results of the study reinforce the assumption that virtual reality offers benefits that less-immersive viewing mediums lack.

CHAPTER II

LITERATURE REVIEW

This section presents current and past technologies related to space system design and simulation, space-based virtual reality applications, and developmental advantages of virtual reality simulation and testing. It also briefly covers rating scales for pilot workload of flight characteristics.

2.1 Space Mission and Engineering Simulation

The paper, “The Distributed Space Exploration Simulation (DSES),” states that the “traditional approach to the simulation of space vehicles has been through disjoint collections of simulations. Each individual simulation usually focuses on a specific domain aspect of a space vehicle and is developed, maintained and executed within the confines of the facility having the particular domain expertise.” [25] DSES was an internal initiative by NASA to coordinate large scale simulations within the agency’s separate locations across the United States. This system, similar to SpaceCRAFT, suggests that there is a better way to collaborate on simulations as to not re-create architectural models and tools unnecessarily, resulting in more cohesive and cost-effective testing and simulations. Although DSES as a whole does not exist anymore, architectural pieces built previously are still being used today.

At the core of DSES is the Trick Simulation Environment. Developed at the NASA Johnson Space Center (JSC), Trick has been used for many space simulations including

the Shuttle Remote Manipulator System (SRMS), the Systems Engineering Simulator (SES), the Virtual Reality Laboratory (VRL) and the Multi-Mission Space Exploration Vehicle (MMSEV). [19] Trick has been used and refined since the early 1990's and has grown to allow other entities to use the open source software. Trick handles the architecture of the simulations including networking between analytical models, job ordering, inputs and data recording rather than high-level visuals. This program links together mathematical models of both the environment and the test subjects and performs time-based physics calculations. NASA has developed a 3D visualizer for Trick simulations, among other applications, called EDGE 3D. This program was built-in house and is used to display the simulations' 3D models during run-time for real-time testing and allows human-in-the-loop (HITL) capabilities.

Throughout 25 years of development, Trick has improved its capabilities continuously and broadened its scope to allow for more diverse simulations. During my internship at Odyssey Space Research, I was introduced to Trick and its capabilities. It offered a refined testbed for the simulations we performed while reducing the workload of developing our own simulation architecture. Although the program has many benefits, I found it to require a large learning curve and a lot of code-based configuration to run the sims. These issues emphasize the need for a program like SpaceCRAFT that the world can use together without explicit knowledge about the technical side of the platform.

2.2 Space Related VR Studies

Many organizations in the aerospace sector are adopting the use of VR and virtual environments to help develop and test new technologies. The authors of “Analysis of a Moon outpost for Mars enabling technologies through a Virtual Reality Environment” claim that “the wide variety of devices currently present on the market, ranging from Oculus Rift to HTC Vive to Sony PlayStation VR, each characterized by different strengths and weaknesses, has pushed both VR and AR to gain momentum in several scientific domains including the space sector.” [7]. This paper details an analysis of systems required for a Moon outpost, which would test technologies for future Mars missions. The authors ran simulations in VR to determine an appropriate base location based on solar availability and assessed environmental control and life support system (ECLSS) and in-situ resource utilization (ISRU) systems. Similar to the research presented in this paper, the simulation within this thesis is a virtual environment and habitat in which future space systems could be tested. Though the idea and utilization of VR has been implemented for decades, it has only recently become powerful enough to use as a universal tool that can be widely distributed. The Moon Base for Mars study concluded that VR and space system design will have a coupled future as the technology progresses and collaboration continues to propel the forefront of engineering space technologies forward. The authors state “VR could offer a powerful multidisciplinary platform where the modular hardware design process, the operations procedures optimization, low gravity environment familiarization, and EVA training could be tested in an innovative and more effective way,” an idea which the SpaceCRAFT team shares.

Aside from VR proof-of-concept and theoretical simulations, real world applications for training purposes have been in use for some time. The Virtual Reality Lab at NASA's Johnson Space Center in Houston, Texas, has been innovating virtual reality devices and scenarios for over 20 years. Eddie Paddock, the VR Technical Discipline Lead of the VRL, has been an advisor for the SpaceCRAFT team from the beginning of the program. He and his team have created Extra-Vehicular Activity (EVA) exercises in VR for astronauts to help facilitate faster and more robust training methods. In an article from the online Tech Republic group, astronaut Doug Wheelock stated, "I know exactly where I am, it looks just like the VR Lab" in reference to an EVA he was performing outside of the ISS. [12] His training in VR gave him invaluable knowledge about his surroundings and the spacecraft environment he was working in.

Although the VRL uses some in-house applications, such as Trick, EDGE and DOUG (Dynamic Onboard Ubiquitous Graphics), they are transitioning to off-the-shelf hardware and game engines as consumer technology progresses. The switch to openly accessible devices and software, such as the HTC VIVE and Unity game engine, allows the lab to focus on the application of the tools rather than development of the tools. Having access to the same technology as NASA allows the public to work in conjunction with their efforts and permits researchers to collaborate more effectively on a global scale.

2.3 Quadrotor Related VR Studies

Research performed at the School of Computer Science of University College Dublin in Ireland used an Oculus Rift HMD and a commercial quadrotor to create a VR control scheme incorporating head movements as a controller. The paper entitled “Oculus Rift Application for Training Drone Pilots” describes the use of a real-time video feed from the on-board camera to create a 3D view inside the HMD. Their primary objective was to test the combination of the Oculus Rift and Robotic Operating System (ROS), which could be used to fly a quadrotor using the HMDs gyroscope and accelerometer. While the use of the HMD, which is a VR product, was used as a pseudo-AR device and their test was physical and not simulated, there were several similarities with my research. Both studies used a variety of experienced and inexperienced users and required participants to follow a predetermined path with a quadrotor. The researchers created a similar view to what my study provided and tested users controllability of a drone.

Although the paper does not provide an abundance of details related to the data of the findings, they appear to have had similar results. Some users in both studies initially had difficulty flying the drone which was mitigated by allowing multiple test flights and allowing the user’s eyes to adjust to estimating distances. Their research closely resembles mine on several fronts; however, the objectives of the two differ substantially. My research focusses on the use and evaluation of VR relating to space system design and space-based environments whereas their primary aim was to develop a system which could be used to train drone pilots.

2.4 Analysis of VR Studies used in Serious Games

“Serious games” is a term used to describe simulations which utilizes various aspects of game technologies and methodologies to augment their research or designs. One article, “An Analysis of VR Technology Used in Immersive Simulations with a Serious Game Perspective,” analyzed 46 simulations with varying levels of immersion in order to better understand the benefits and limits of using VR for serious applications. These simulations ranged from experimental designs to utilizing user studies to determine VR effectiveness and used products ranging from CAVEs (CAVE Augmented Virtual Environments) to HMDs. Some of the findings from this study which are related to the work presented in this thesis are as follows:

- 1) “Most [user] studies assessed effectiveness through comparisons between the simulator and a traditional approach used for the same purpose.”
- 2) “Ninety percent of the studies measures user performance to assess effectiveness, having the expected outcome attached to the simulation purpose.”
- 3) “Although the VR technology still might be limited by motion sickness, using it in ‘gamified’ simulations presented great benefits for knowledge gain and retention, firefighting training...”

While there are countless studies related to VR and serious games or simulations, the research in this paper took a variety of those studies and analyzed them with positive outcomes. The first related finding reinforces the choice to compare VR testing with traditional desktop testing. Although a real-world control would have been ideal, the physics of rotating environments makes that infeasible. The second finding validates the

decision to run a user study to obtain data on the engineering application as well as determine the consequences of applying VR to the study. The final point sums up their findings in a positive manner as well as acknowledges the downside of VR, motion sickness, which was also observed in my study.

2.5 Institutional Benefits of VR

Along with the benefits of training and knowledge retention by augmenting VR to current practices, there are many other advantages VR offers. The first of which is monetary savings. Designing engineering applications, especially those which will go to orbit and beyond, requires exhaustive testing and iterative design. With the advent of virtual reality technologies, the design cycle can cut costs in physical rapid prototyping and destructive testing. Confining intermediate designs to virtual prototypes reduces personnel and material costs associated with building physical prototypes. While there are still benefits to having physical models, there is much to learn before the product gets to that phase, and many times, products will be released with drawbacks that could have been caught before prototyping.

Industries, such as automobile manufacturing, have been implementing VR in the design process for a multitude of points. A web article for Automation World entitled “Ford Uses Virtual Reality to Reduce Costs” brings to light some of these reasons. The author states “virtual reality technology is being tested at the Ford Valencia facility to increase overall quality and reduce the physical stress of assembly line jobs, as personnel have found it useful in determining the best installation process of a component on the

assembly line.” [8] Not only has VR helped Ford iterate on designs, it has helped the company find better solutions to the manufacturing process which further reduces cost and can mitigate risks to the factory employees.

Another article titled “Measuring the Benefits and ROI Of VR” reaffirms the time and monetary savings of using VR, stating “one of the main ROI of VR is definitely the reduction of projects duration and budget.” [17] The mentioned article gathers statistics for a variety of industries with positive correlations between the adoption of VR and cost and time savings, reduced injuries during the manufacturing process, a reduction of prototyping, employee training effectiveness, and optimizing the design phase and processes.

Many engineering products have hard deadlines that must be met. Space sector manufacturers must adhere to these timelines as close as possible while still providing a high-quality product or they risk setting back launch dates, which have narrow windows, at significant cost. VR allows companies a new path to test and design products that can be done within the design loop. Lockheed Martin advocates the use of VR, stating that they use “3D imaging in VR to catch engineering missteps before the asset hits the production floor, saving time and money by correcting those errors sooner rather than later.”

Many industries have seen positive results of using virtual reality spanning a wide range of applications. VR has economic and engineering benefits with positive return on investment.

2.6 Rating Scales for Flight Characteristics

The user study performed for this research asked participants to rate flight characteristics of an unmanned drone which was used to follow a path within a rotating artificial gravity environment. Rating flight characteristics for aircraft has been an important engineering step towards proving flight-worthy designs for decades. Two rating methods were considered, the NASA-TLX scale and the Modified Cooper-Harper scale. Each of these scales focusses on pilot mental workload for various maneuvers and flight characteristics and have been shown to “have a positive correlation” to one another. [15] Another study which related the TLX and MHC scales concluded that both rating systems are “moderately to highly acceptable tools;” however, the TLX scale was considered superior. [13]

The MCH scale being unidimensional, whereas the TLX scale is multidimensional, takes less time to administer and perform data analysis on. Both scales were considered; however, the MCH was selected for this study due to requiring less time to administer and the time-proven benefits. Figure 3 shows the scale in the form of a logic tree, beginning in the bottom left corner. Each participant rated five separate maneuvers of the quadrotor adhering to the scale and descriptions of each value. This scale was the primary subjective measure employed by the user study.

Handling Qualities Rating Scale

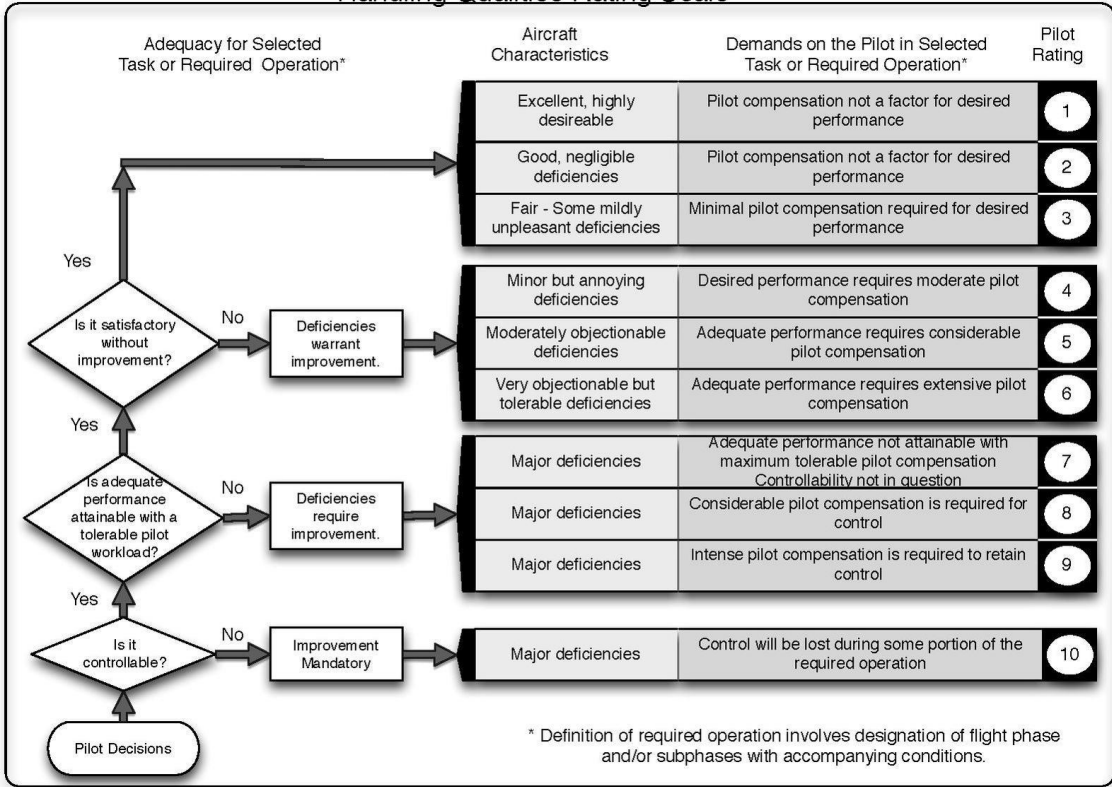


Figure 3: NASA’s Modified Cooper-Harper Scale for Rating Aircraft Characteristics. [12]

CHAPTER III

SIMULATION DESIGN

3.1 Choosing a Simulation

Previous chapters discussed the SpaceCRAFT platform and the development of a proof-of-concept simulation to obtain objective and subjective data to test the value of utilizing virtual reality for space system design. Selecting an appropriate simulation required consideration of multiple areas of interest including an environment not suited for physical simulation on Earth, interactive HITL, real-time decision-making procedures, and a feasible and familiar task that is altered due to environmental differences.

Several options were considered, and two potential scenarios emerged: navigating a spacecraft through an asteroid field to dock with a space station and flying a quadrotor drone in an artificial gravity, rotating space station. Both scenarios employ the areas of interest described above and have similar components to one another, a rotating space station and a user-controlled vehicle. Ideally, both simulations would have been pursued; however, it was decided that only one would be necessary for the scope of the project. Either selection would have been adequate, though, the latter simulation was chosen for a few reasons.

The first reason was the consideration of potential familiarity to the participants regarding their task. Neither environment, microgravity nor artificial gravity, can be easily reproduced terrestrially; however, artificial gravity is closer to life on earth than microgravity and flying (or crashing) a quadrotor is a familiar activity for many.

Controlling a vehicle that is already familiar to the user allows the comparison of the dynamic differences between the space station and normal gravity environments to produce a good medium to test virtual reality applications.

The second reason was that developing an artificial gravity environment allows for further tests and demonstrations of the unique dynamics that are created by the strong Coriolis and centrifugal forces. If a similar space vehicle was considered as a long-term habitable environment for astronauts, this VR environment can be used to show how different daily activities would be. Throwing a ball or pouring a glass of water results in unexpected motion in this environment.

3.2 Artificial Gravity Space Station

The artificial gravity space station, aptly named Hogan Station, had a few constraints that determined the general architecture, namely the rotation rate and radius. Much research has been done on the topic of artificial gravity for human safety and comfort, and Figure 4 shows a “comfort zone” that describes the limits and ideal values of rotation rate and station size for human habitation. Minimizing the radius while staying within the bounds of the “comfort zone” idealizes the cost and feasibility of such a space station. Although Hogan Station is a large structure that would require an enormous effort to create in reality, it follows the scale of the “comfort zone” with the intentions to showcase the gravity gradients in any rotating space habitat with a similar radius and rotation rate. The open area cylindrical design allows for continuous airspace to perform the task with inclusive habitable structures which act as obstacles to the user.

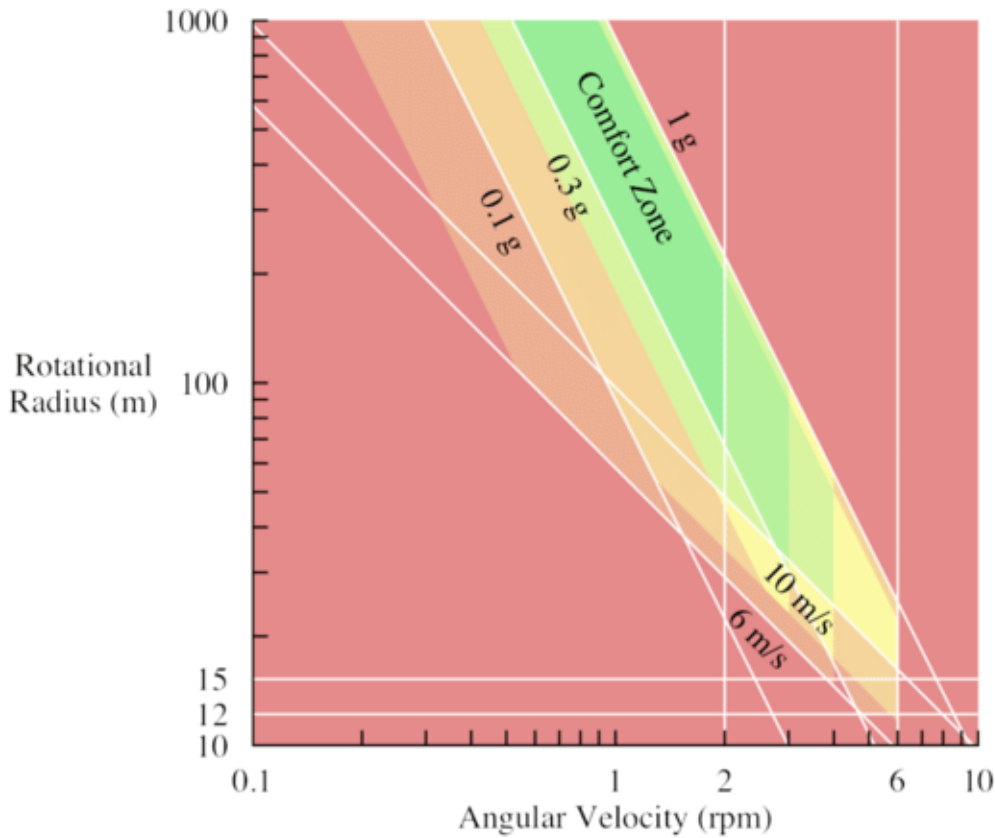


Figure 4: Artificial Gravity Comfort Zone [2]

The task was to fly a quadrotor along a three-dimensional path through the space station to arrive within a target zone. The challenge of designing the path was to give the users get a sense of the five flight characteristics of the drone, described later, while being able to complete all of the trials within a two-hour period. Numerous paths were designed and tested and ultimately one was selected. This gave me an idea of how long each trial should take on average.

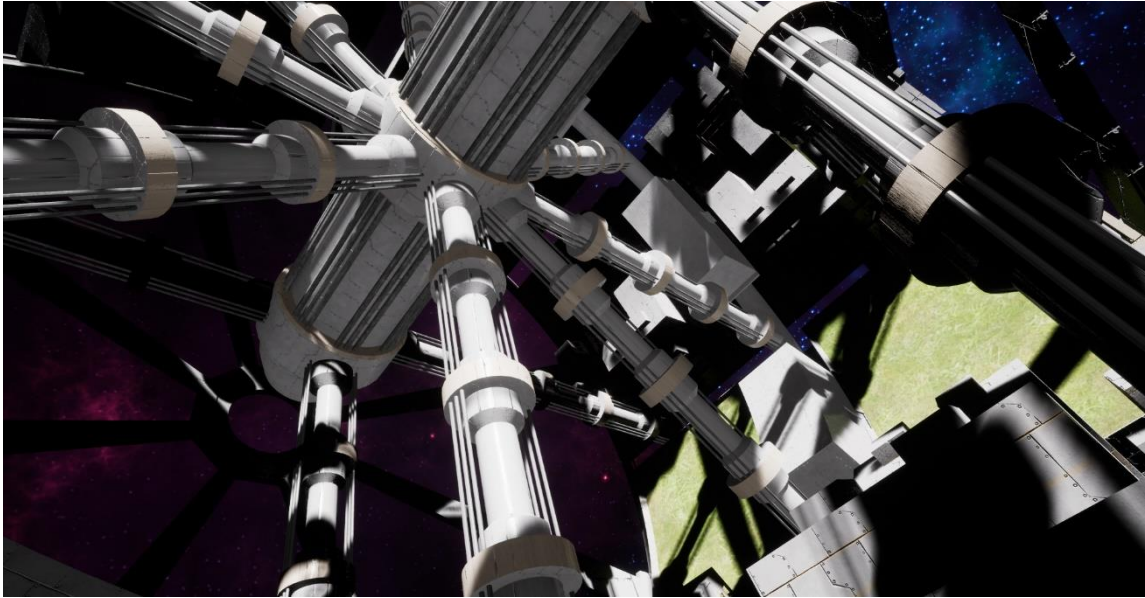


Figure 5: Internal View of Hogan Station

The final path did not include sections travelling retrograde to the station's rotation and did not have any specific instructions to hover at any instance. It was determined that these tasks would have unnecessarily complicated the overall task of flying between two points. Nevertheless, the participants would often overshoot and end up hovering or flying retrograde in order to continue the trial. As such, the course inherited retrograde motion, though not at any specified location. Retrograde motion and hovering were explained in the description and demo of the study to each participant, which is described in the following chapter.

The simulation architecture included data recording to a comma-separated-value, or .csv, document. This allowed the data to be analyzed efficiently with Python and reduced

the workload of having to perform data analysis on 600 sets of trial data. Data recorded included the tick number, simulation elapsed time, state data from the quadrotor relative to the space station, state data of the station, location data for the point along the path that the quadrotor is closest to at any time, the distance along the spline, the error of the quadrotor to the spline and a Boolean value which recorded whether or not the quadrotor was colliding with anything. The analysis of this data provides the objective data from the simulation, which tells us how the user performed with each configuration.

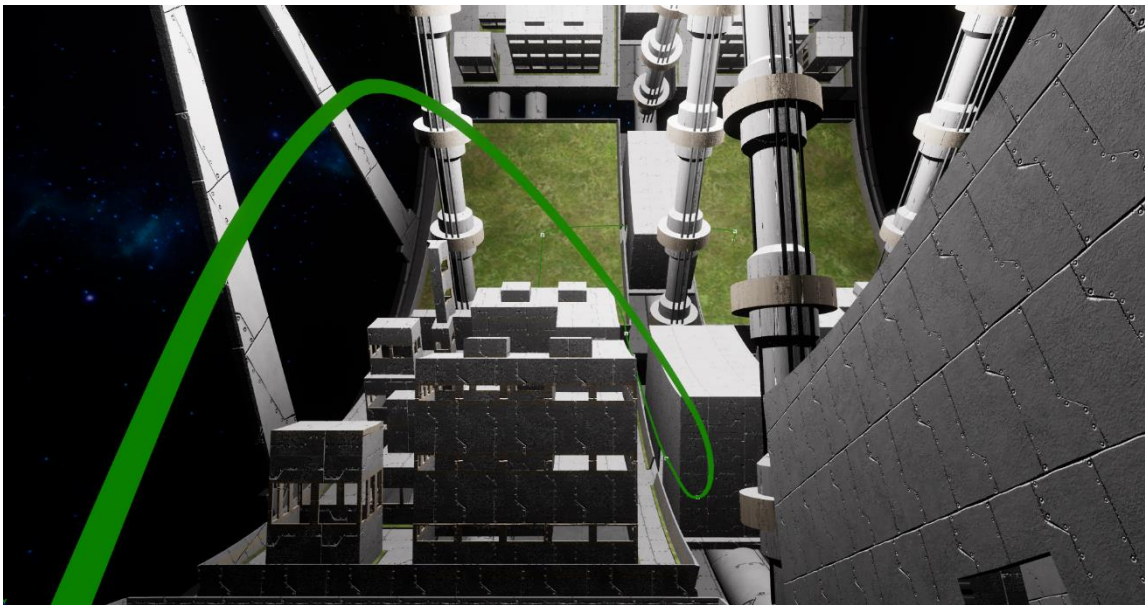


Figure 6: Internal View of Hogan Station with Spline

3.3 Simulation Objective

3.3.1 Quadrotor Flight Mode Selection

The study was designed to evaluate operator performance for different interface designs, and the purpose was to test whether VR is a useful tool to evaluate design concepts. In this case, control mode and sensitivity were the variables. A matrix of flight modes was conceived which is described in the following section. Each participant completed three trials of the simulation for each flight mode and rated the flight characteristics of each mode as they progressed. At the end of the study, they were asked to choose the best flight configuration for the task. The difference in their ratings of the flight modes relative to the viewing medium and the data of those trials provided the results for the study's main objective.

3.3.2 Flight Modes

Today's commercial quadrotors have several different flight modes depending on their intended application. These flight modes do not directly carry over to a rotating environment, and cannot be implemented in the same way. In order to develop new control schemes for a rotating environment, the control modes needed to be defined. Three aspects for flight modes used for the simulation of a rotating environment are stabilization, sensitivity and viewing medium, each with two options. Each combination was tested to create eight distinct flight modes which the users tested and are shown Table 1 below.

Table 1: Flight Mode Matrix

Medium	VR				Desktop			
Mode	Rate-Control		Attitude-Control		Rate-Control		Attitude-Control	
Sensitivity	High	Low	High	Low	High	Low	High	Low

The bottom row in the Flight Mode Matrix, pictured above, shows each of the eight flight modes selected. For example, the first flight mode would be VR, Rate-Control, High Sensitivity.

3.3.2.1 Stabilization Modes

The stabilization aspect of the flight mode is split into rate-control and attitude-control modes. This aspect was the primary difference in the final product design and had the largest impact on the users' performance.

3.3.2.1.1 Attitude-Control

This flight mode aspect added a stabilization effect to the quadrotor which reoriented the aircraft to be level with the point on the ground closest to the quadrotor. The stabilization was only in effect when there was no pitch, yaw or roll command;

however, it was applied when only the throttle was active. Though this stabilization did not control the throttle, it aided the user in hovering after the appropriate velocity and altitude were attained. Because this flight mode is only active when no attitude rates are applied, and the rate-control mode was implemented in a similar fashion, continuously adjusting the attitude manually resulted in the same flight characteristics between the two stabilization modes.

The control layout of the Xbox One controller used in the study included the use of both axes of each joystick. This causes a coupling between throttle and yaw, or roll if the user preferred a different control scheme. This coupling made only having the throttle applied somewhat difficult as any unintentional deflection laterally on the joystick would disable the stabilization and subsequently yaw or roll the drone instead.

3.3.2.1.2 Rate-Control

This flight mode maintained the orientation of the drone when attitude control ceased. Similar to the previous stabilization mode, this aspect was only applied when there was no yaw, pitch, or roll command. This feature allowed the user to apply a given orientation and then feather the throttle without needing to continuously apply attitude control to approximately maintain the drone's orientation.

The sacrifice of automatic reorientation gave the user more control over the drone and allowed for more complex flight maneuvers. This added to the learning curve and increased the skill ceiling which made getting used to this flight mode more difficult.

3.3.2.2 Sensitivity Modes

The sensitivity aspect was also binary and consisted of a high and low sensitivity which was applied to the rotational rates of the quadrotor. Each of the three attitude control inputs used a parabolic control curve. The input, from 0 to 1, of the deflection of the joystick was squared to create this effect in order to have a wider range of fine control while still having the ability to have full control rates at max deflection. The throttle control of the quadrotor was not affected by the sensitivity mode and used a linear control curve.

The sensitivities of the attitude controls were initially tested in development until a natural and realistic set of gains was found. The high and low sensitivities were implemented with a single multiplier that affected each of the three attitude control inputs.

3.3.2.2.1 High Sensitivity

The higher sensitivity mode increased the attitude input from the values found from developmental testing by a factor of 1.25, or 25%. The user was able to rotate the drone at a faster rate which allowed for quicker maneuvers at the expense of a smaller range for fine adjustments. The parabolic control curve helped minimize the downside of this control mode. Similar to the rate-control stabilization mode, this sensitivity was expected to have a higher skill ceiling and to take longer to learn.

3.3.2.2.2 Low Sensitivity

The low sensitivity mode decreased the attitude input from the testing values to .75, or 75%, of the values found from developmental testing. The lower sensitivity was useful for slower maneuvers and gave the user finer control at the expense of faster rotations.

3.3.2.3 Viewing Medium Modes

The viewing medium was either traditional desktop mode or utilizing an HMD for VR simulation. Though the viewing medium category was not a product implementation feature, it is still considered a flight mode aspect for the purposes of the study.

3.3.2.3.1 Traditional Desktop Mode

The desktop mode was used as a baseline to compare results with the VR mode and acts as a control group. The desktop mode used a 21” Dell monitor for the trials. This mode lacked any head tracking, and the scale of the station and drone were obscured by the smaller viewing area relative to the participants’ bodies.

3.3.2.3.2 VR Mode

The VR mode used an HTC VIVE head-mounted display and immersed the users in the simulation. While performing the task, the users were seated and able to move their heads to add a local rotation to the drones gimbaled camera. This feature was not implemented using the desktop setup as it would have further complicated the drone’s controls.

One challenge that the VR mode had was that the users would occasionally forget that looking forward would align with the forward vector of the drone. They would turn their head to look in a different direction and leave it there, usually resulting in a coupling of yaw and roll, or swapping them altogether. They were told to find a neutral position before they began the trial to get a good reference to what is ‘forward’.

3.4 Physics

3.4.1 Artificial Gravity

The primary component of artificial gravity stems from the centrifugal acceleration term when determining the acceleration of an object moving within a rotating reference frame. An inertial, or non-accelerating, reference frame has a very basic force equation given from Newton: $f = m * a$. When switching to a rotating frame, the equation becomes more complicated and several terms are added which comes from the cross-product when taking derivatives of an objects position vector. The final equation looks like this:

$$\mathbf{a}_r = \mathbf{a}_i - 2 * \boldsymbol{\omega} \times \mathbf{v}_r - \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{r}) - \frac{d\boldsymbol{\omega}}{dt} \times \mathbf{r}$$

where

- $\boldsymbol{\omega}$ is the angular velocity vector of the rotating reference frame
- \mathbf{a}_i is the apparent acceleration within the rotating frame
- \mathbf{v}_r is the velocity in the rotating frame
- $-\boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{r})$ is the term representing the centrifugal acceleration

- $-2 * \boldsymbol{\omega} \times \mathbf{v}_r$ is the Coriolis acceleration
- and
- $-\frac{d\boldsymbol{\omega}}{dt} \times \mathbf{r}$ is the Euler acceleration, which is 0 for a constant angular acceleration.

The centrifugal acceleration is caused by the normal force of the surface of the space station acting on the quadrotor. While the quadrotor is airborne, the tendency for the drone to fall back to the surface is caused by the centripetal acceleration, which is opposite of the centrifugal acceleration. This is the force wanted by advocates of artificial gravity; however, the Coriolis term prevents this from being as ideal. When an object accelerates in the direction pointing to the axis of rotation, the Coriolis acceleration causes the object to drift prograde, along with the rotation of the space station, or retrograde, opposite the rotation of the space station. This additional force magnifies the difficulty in controlling airborne objects within this environment.

3.4.1.1 Implementation

Some programs which utilize game engines, such as Kerbal Space Program using Unity, opt to keep the user at the center of space and have the environment translate and rotate around that point. Implementing this approach would have complicated the physics and could have had undesirable side effects. I chose to try a straightforward approach and keep the inertial space constant and apply a rotation to the space station to get the artificial gravity effect using PhysX, the built-in physics engine offered by UE4. This approach worked well and only had minor issues which needed to be worked out. One of them was camera shake and stuttering while on the surface. This was alleviated by adding a physics

sub-stepping in the project, which continually calculated the physics in between frames. Physics sub-stepping also helped the drone to avoid falling through the floor and into space while landing.

Adding the rotation to the space station and allowing PhysX to do the rest meant that I did not have to implement the acceleration equation myself. These extra forces found while in a rotating reference frame did not need any additional coding as they are products of the rotation and linear physics.

3.4.2 Quadrotor

The quadrotor control modes were described earlier, this section is to describe how the controls are applied to the drone. The quadrotor needed to behave similar to a quadrotor on earth and have similar flight characteristics. Initially, the drone was to have integrated physics in the controls which would affect the angular rates and accelerations directly; however, that would have required development of all of the control aspects of the drone, which was not the objective of the study. Instead, the physics of the controls were simulated with local rotations applied to the 3D model of the drone. This had several advantages with additional drawbacks. Directly implementing local rotations based on control stick deflection was quick to develop and was reasonably realistic in control; however, because the controls did not affect any angular velocities, any unintended interaction with the space station could not be counteracted with the controls. This caused an additional level of difficulty if the user was not careful.

Originally, the users were to traverse the course as accurately as possible until they either got to the end and completed the task, or until they hit an object. Any collision would result in a failed trial and the task would restart. Completing the task without colliding with any walls or obstacles proved to be too difficult and time intensive for the study and the requirements were reduced. Instead, in the event that a collision occurred, the users were told to land the craft on flat ground in order for the space station to reset the drone's angular rates to that of the station. This disadvantage, while not ideal, was outweighed by the advantage of realistic flight simulation and faster development time.

3.5 Simulation Development Tools

3.5.1 Blender Modelling

The 3D model for Hogan Station was developed in blender by Alex Sein, an undergraduate researcher working on model development for the SpaceCRAFT team. His design was created with the radius constraints and moment of inertia effects in mind, as well as input from me for habitat layout. The radius constraint determined the outer radius of the station while the moment of inertia effects required the station to be shorter along the axis of rotation. This decision was made for feasibility reasons, as objects in space stably spin along their axis of highest moment of inertia.

The quadrotor model was also created in blender by Alex with later revisions done by Neil McHenry. The model was loosely based on off-the-shelf quadrotor designs with an on-board gimballed camera.



Figure 7: 3D Model of the Quadrotor used in the study

3.5.2 Unreal Engine

Epic Games' game engine, Unreal Engine 4, was used to create the simulation. Engine version 4.16.3 was used as it was the current version of Unreal that the SpaceCRAFT team had been working with. The only plugin used in conjunction with UE4 was the Victory plugin from Rama for additional blueprint functions.

Unreal Engine is primarily a game development engine used in countless applications. Many AAA games have been made using Unreal Engine, including Rocket League and Fortnite. Although the software was developed for, and is typically used in, video games, it contains all of the necessary tools to create simulations of varying fidelity. It is an open-source platform that uses C++ at its core, which has many benefits. Some of these benefits include royalty free open-source libraries, a time-tested language that is constantly evolving, and robust object-oriented programming capabilities.

3.5.2.1 Blueprints

Unreal Engine 4 has a built-in visual programming tool called Blueprints. This visual scripting allows anyone without extensive programming experience to create and develop programs using UE4. The application converts any Blueprints in the simulation to C++ during runtime and can be modified by other C++ files. It is known that the applications tend to run slower or less efficient because of the generalized approach to the conversion. Initially, Blueprints were solely used to prototype functionality; however, it was not necessary to convert the Blueprint functions that I made into native C++ code as the simulation ran smoothly and the conversion process would have not yielded any additional value. Though using C++ and an object-oriented approach to programming is preferred, Blueprints made developing the simulation quicker and just as robust.

3.5.3 C++ and Visual Studios

At the core of UE4 is C++. Using Blueprints for the project made sense and was able to achieve the desired output; however, it lacks the immediate versatility of C++ and the large number of libraries available. Some functions had to be implemented in C++ to avoid developing custom Blueprints functions. This was done using Visual Studios and was installed through the UE4 launcher which ensured that all required libraries and dependencies were installed at the same time.

3.6 Hardware

3.6.1 HTC VIVE

The head-mounted display used for the study was a first-generation HTC VIVE. This HMD was one of the first affordable and commercially available virtual reality devices with adequate specifications and full 360-degree head and controller tracking. The specs for this device are listed below.

- Refresh Rate: 90 Hz
- Field of View: 110 degrees
- Display Resolution (Per Eye): 1080x1200

The VIVE uses an infrared tracking system with the included Lighthouses. These Lighthouses emit an infrared pulse at 60 Hz which is measured by the headset and hand controllers and converted to virtual rotations and translations.

3.6.2 Desktop Computer

The computer used was my personal desktop computer. The specs are listed below.

- CPU: Intel i7-8700k @ 3.70 GHz
- GPU: NVIDIA GeForce GTX 1080 8GB GDDR5X
- RAM: 16 GB 2400 MHz
- OS: Windows 10 64-bit

3.6.3 Monitors

Two Dell P2210H monitors were used to conduct the study, with one being used to display the simulation and the other for data recording. The specs are listed below.

- Size: 21.5 inches diagonally
- Resolution: 1920x1080
- Refresh Rate: 60 Hz

3.6.4 Xbox One Controller

The controller used to fly the quadrotor was an off the shelf Xbox One controller. Using this controller rather than a traditional quadrotor controller was due to familiarity with a wide range of users.

CHAPTER IV

STUDY DESIGN

4.1 Development of the Study

This chapter explains the process of setting up and performing the user study, addressing nausea concerns, participant goals, the simulation task, and what measures would be taken to determine the success of the application.

4.1.1 IRB

The user study for this research was conducted under the instruction of the Texas A&M University (TAMU) Human Research Protection Program (HRPP). The IRB number associated with the study is IRB2018-1010, and the expiration date is December 11, 2019.

4.1.2 Structure of the Study Trials

In order to attain measured data from the simulation, a user study was conducted to collect objective data and subjective ratings on the flight characteristics of the quadrotor and the varying control modes. The target audience was students attending Texas A&M University in engineering, science, and architecture. An email was sent through the university's bulk email service to each student enrolled in these colleges. The goal was to get a set of users with varying experience levels with quadrotors, real aircraft flight, simulations, virtual reality, and using joysticks/controllers, so the data could be cross-

examined in multiple ways. There were not enough female participants to perform a gender-balanced study while still having an even variety of experience levels.

A target of 32 users was selected on the basis of achieving significant results. Incidentally, this was higher than the average number found for similar studies. [17] The users would be split into four groups of eight that could perform the trials in different orders to adjust to the learning curve associated with gradually gaining more experience in the environment throughout the tests. The four groups would perform the trials in the following order:

- Group 1: VR followed by desktop for the first flight configuration, repeating throughout the four modes.
- Group 2: desktop followed by VR for the first flight configuration, repeating throughout the four modes.
- Group 3: All flight modes in VR, followed by all flight modes in desktop.
- Group 4: All flight modes in desktop, followed by all flight modes in VR.

Conducting the experiment in this way proved to be difficult as some users would feel too nauseous while testing in VR before getting used to the controls using the desktop mode. In total, 33 users were able to be tested within the timeframe, and eight of them did not finish the trials either due to consistent nausea or a lack of time, usually caused by frequent breaks due to the induced nausea.

4.1.3 Challenges with the Experimental Setup

The simulation was built to represent a theoretical test for a theoretical product. Keeping the details of the simulation as close to the theoretical test was a requirement. This meant preventing the user's view from being adjusted when the head-mounted display (HMD) was translated due to their head movements. The theoretical product would only have gimbaling capabilities for the onboard camera and as such, the simulated product only allowed head rotations to affect the user's view while testing in VR. Locking the view to the virtual camera's location was an intended strategy; however, it goes against best practices for preventing nausea in virtual reality.

A major factor that causes nausea is a motion/view mismatch, which is when a user expects their view to change in one way, and it changes in another. This is related to sea or motion sickness and utilizes the balancing abilities of the inner-ear. As described above, constraints within the simulation prevented translational head-tracking which is against VR best practices. Another factor that causes nausea is the Vergence-Accommodation Conflict. This issue arises when looking at a two-dimensional plane while viewing stereoscopic three-dimensional content. The vergence distance is how far away from a person's eyes an object appears, and can be correlated to the angle between the two eyes' focus point. The accommodation distance is the individual focus of each eye. If the two align properly, there is no conflict, such as normal visual observation. When they do not match up, the conflict arises and visual fatigue and discomfort may occur. [4] Several other factors of the nature of this simulation furthered this issue including the rotating reference frame and the difficulty of flying the drone. The artificial gravity was

caused by a rotation of the space station in which the drone was flying. Seeing the environment move without feeling the accelerations you expect to feel induces the nausea response.

The difficulty in flying the drone was magnified by the counter-intuitive effects of airborne objects in this environment. For example, when a user inputs only a throttle command, yet the drone drifts prograde and radially, a perceived motion/view mismatch occurs. This is accurate for the rotating artificial gravity environment. Correcting for these effects was difficult for the test subjects, especially at first. Until the users became familiar with the environment and motion effects, this mismatch had a higher probability of inducing nausea.

A final factor that worked against a well-designed VR application was the variable frame rate. A minimum of 90 frames per second, or fps, is ideal for virtual and augmented reality applications and this simulation was unable to stay consistent at that level. The HTC VIVE utilized in the study used interleaved reprojection techniques to account for the variability in frame rate. This prevents the VIVE from running on any frame rate that is not a divisor of 90 – meaning if the frame rate drops to 89 fps, the VIVE would render at 45 fps. Similarly, it would continue to reduce the number to 30 fps, 15 fps, etc. as needed. Asynchronous reprojection would have been smoother, and would have delivered the highest frame rate possible up to the maximum of 90 fps. For most views in the simulation, the frame rate would not hit the 90-fps threshold and it would automatically be reduced to 45 fps. Occasionally it would run at 90 frames; however, the majority of the time the frame rate would be half the ideal number.

The frame rate issues could have originated from several factors including the hardware used for the study and/or the software/coding limitations. I believe the hardware could handle the simulation had the software configuration and the simulation been implemented more efficiently. Unreal Engine is a fantastic tool; however, it takes a high level of experience to get ideal performance.

Each of these issues, some necessary, others unfortunate, contributed to nausea symptoms resulting in close to a non-ideal scenario for VR testing. While there is no way to assess the impact that testing this simulation in VR with these drawbacks had, it can be a reference to other studies and these effects should be factored into the results of this study.

4.2 Simulation Task

Once the simulation environment was constructed and the control modes determined, and user study defined, the task for the study was developed. The primary objective for the user was to test and rate the different flight modes of the quadrotor to determine the most ideal configuration. In order to provide a consistent experience for each participant, a virtual path was created for the quadrotor to follow. Each participant was asked to follow the line as closely as possible to get within a 5-meter radius of the end. Although each user put varying weights on precision and timing, they were given the same task and instructions. Because the data was normalized between users in the end, the noise due to how they weighed the time and error was reduced.

Users were allowed to reset the simulation as many times as they found necessary. There was a wide split between users ranging from less than an average of one reset per trial to requiring multiple each time. The trials continued for each flight configuration until the task had been completed 3 times for each of the 8 configurations. Figure 8 below shows the path that the users were instructed to follow, along with the model of the drone.

The participants were told that landing and colliding with objects was acceptable; however, they should attempt to finish the course without any collision. Once more, if a user did collide with an object, or at times take off from the ground awkwardly, they should land flat and take off again. Not only was this to ensure they did not ignore the colliding criteria, but to also ensure the physics of the drone behaved appropriately.

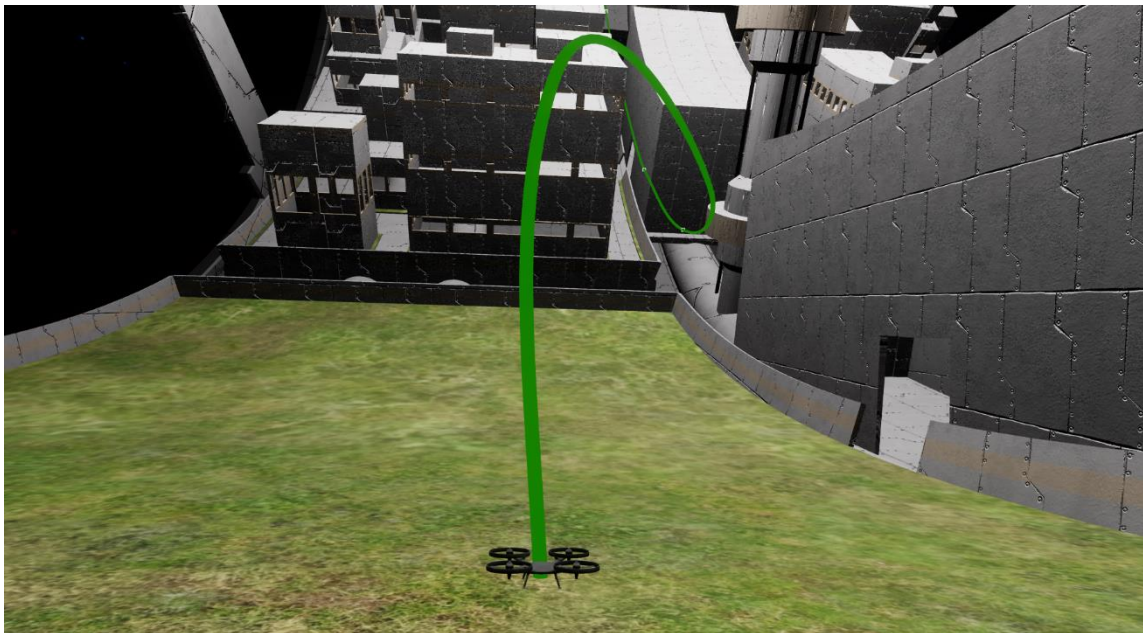


Figure 8: Drone with Spline

4.3 Subjective Measures

Subjective measures are opinions of the individual performing the study. Although opinions differ and can be influenced by external sources, there are methods which are used and have been proven to be beneficial and consistent in design analysis. One of these methods, the Modified Cooper-Harper scale, was used and is described below.

4.3.1 Modified Cooper-Harper Rating Scale

The primary subjective measure for the rating the flight characteristics of the quadrotor was the Modified Cooper-Harper, or MCH, rating scale. This scale is a pilot workload rating system used extensively for testing aircraft and has been an industry standard since early after its inception in 1969. It is best suited for determining handling qualities of flight maneuvers for a variety of vehicles. This scale was used for five different flight characteristics of the quadrotor: hovering, prograde motion, retrograde motion, axial motion, and cornering and obstacle avoidance. Each user rated these maneuvers following the three trials they performed with each configuration. Completing the task three times provided the opportunity to familiarize themselves with the difference in flight modes and viewing mediums necessary to have an adequate understanding of the maneuvers. The MCH scale is elaborated on in the Literature Review chapter of this document.

4.3.2 Objective Measures

It has been stated that the objective measures were results from the simulation, saved to a file and analyzed with Python. Although a considerable amount of data was

collected, the primary measure used was the users' average errors from the path throughout the trials. The simulation saved the state data to .csv file and each measure was taken from Unreal Engine using built in functions. The data included precision up to the sixth decimal place, though the significant digits varied by the digits to the left of the decimal. Most of the data collected was used to determine this average error, though some was used to recreate the trials and play them back in the simulation. The playback option was initially included to ensure that the data recorded was accurate.

The average errors, being the primary measure, were calculated in the Python program and categorized based on their control mode, sensitivity, and viewing medium for each user. The analysis of the data is explained in detail in the following chapter.

4.3.3 User Study Procedures

In order to ensure users had consistent experiences performing the study, a set of procedures was developed. These procedures covered each step in the process while the participants were present. Each step was described by the author, no variation occurred because only one researcher conducted the trials. Although the list is not a comprehensive script, it was followed as closely as possible. A script would have been difficult to follow since examples of the study were performed live to showcase the varying flight modes and control schemes.

- 1) Review Consent and Provide Details
 - a) Read through the user consent form with the study participant.
 - i) Ensure they understand each section.
 - ii) Have them sign it and initial each page.
- 2) Introduce User to Study Equipment and Controls
 - a) Briefly show the participant the virtual environment within the editor.
 - i) Explain terminology of prograde, retrograde, and axial motion.
 - b) Describe the view the user will have within the simulation.
 - c) Describe and show the user the different control modes they will be testing.
 - i) Describe the effects of artificial gravity in the three “planar” directions of relative motion (prograde, retrograde, and axial), as well as the effects of changing altitude.
 - d) Give tips on how to move in each direction. Show them simultaneously.
 - i) Prograde: increasing altitude (decreasing the radius) results in increased prograde velocity. Prograde velocity effectively increase the gravity, making it difficult to slow down. Keeping forward velocity at a reasonable pace is the most difficult part of this motion.
 - ii) Retrograde: decreasing altitude (increasing the radius) results in increased retrograde velocity. Retrograde velocity effectively decreases the gravity, causing vertical movement to be harder to avoid. Initial movements in the retrograde direction are difficult due to initial prograde motion from lifting off the ground.
 - (1) One potential maneuver is to gain altitude and cancel out the coupled prograde velocity. Once a hover state is obtained, decreasing altitude causes a retrograde drift that can be controlled with little input.
 - (2) Another maneuver is to continually pitch forward towards the desired velocity while increasing altitude, which is applicable in any direction, but more so here.
 - iii) Axial: moving axially from a landed configuration has a coupled effect from altitude and prograde/retrograde motion. Lifting off causes prograde drift, which must be cancelled out with pitch/roll and forward movement along the axial direction must be attained via the other input. It is possible to decouple pitch and roll to just one or the other, depending on yaw angle and desired path.
 - (1) Once a desired altitude is attained, the primary challenge with moving axially is the coupled altitude changes and prograde/retrograde velocity changes. This can be used to your advantage if you understand the physics

Figure 9: Briefing Checklist for User Study Procedures

and allow yourself to drift prograde/retrograde by altering your altitude above and below the hovering height.

- 3) Describe the objective of the simulation.
 - a) Follow the glowing path to the final destination.
 - i) Attempt to stay as close to the line as possible without colliding with any structures. Collisions are allowed, as well as controlled landings in order to keep from getting out of control.
 - ii) Get within 5 meters of the endpoint of the path.
 - b) Inform the user that they may restart each trial as they wish, for any reason. The number of resets will be documented with no penalty to the user.
- 4) Describe the Modified Cooper-Harper rating scale and the 5 modes the user will be rating per configuration. Inform the users that they will complete the task three times per flight configuration, including flight mode and viewing medium. This results in 24 completions of the task.
 - a) Hovering
 - b) Controlled prograde motion
 - c) Controlled retrograde motion
 - d) Controlled axial motion
 - e) Navigating corners and obstacles.
- 5) Introduce Xbox One controller configuration.
 - a) Allow the user to choose a desired control layout, with a picture showing two potential configurations. They may use either axis of either stick for each of the four degrees-of-freedom. Inverted controls are also up to the user.
 - b) Button B is to reset the level
 - c) Button X is to stop the simulation and save the data for that trial. No data will be recorded if B is pressed.
- 6) Begin Simulations
 - a) Review the controls and set the user up for whichever viewing medium they use first.
 - b) Open the simulation in the desktop configuration for the first flight mode and allow the user to practice controlling the drone for up to 15 minutes, or until they can complete the course and are comfortable starting.
 - c) Restart the sim and begin recording the trial.
 - d) Simulation data begins recording.
 - e) Observations by study personnel begin.
 - f) Restart the trial until the objective is met.
 - g) Users will fully complete the trial three times per mode.
 - h) Instruct the users to fill out the MCH rating scale and comment on the controllability of the drone.

Figure 10: Briefing Checklist for User Study Procedures Continued

- 7) Setup the next trial configuration.
 - a) This could mean switching viewing mediums, the flight mode, or both.
 - b) Allow the participants to familiarize themselves with the VIVE on, if it is their first time using it during the simulation.
 - c) Perform the three timed trials until the objective is met.
 - d) Instruct the users to fill out the MCH rating scale and comment on the controllability of the drone.
- 8) Allow the users to take a break.
- 9) Repeat until all trials for each control mode and viewing medium are complete.
- 10) Finishing up
 - a) Guide the user through the post-study questionnaire. Provide more details as necessary.
- 11) Each participant had a different experience to some degree. Measures were taken to minimize this effect, such as providing the same information.

Figure 11: Briefing Checklist for User Study Procedures Continued

4.3.4 Questionnaire

After all tasks and trials were performed by the users, an end of study questionnaire was performed verbally and the answers recorded. These questions included rating the difficulty of piloting the drone in both VR and desktop modes, selecting the best flight mode for the task, determining the benefits, if any, that the use of VR provided and what could have improved the study. The answers to these questions are discussed further in the results section.

CHAPTER V

STUDY RESULTS

The user study generated a significant amount of data that required post-processing before statistical results could be analyzed. Six-hundred trials were conducted and required an automated program to process. This section explains the programs used to perform the data analysis, how the analysis was done, and the results of the analysis.

5.1 Analysis Tools

5.1.1 Python

Python is a programming language that offers high-level, object-oriented tools and is growing in popularity. There are many data analysis packages that can be downloaded and used for free including Pandas, SciPy, Matplotlib, and Seaborn. Python, while using these packages, provides a workflow similar to Matlab (which was also considered as an option). All of the data processing and analysis was done in Python using Visual Studio Code. VS Code is a source code editor that incorporates a compiler and debugging features.

5.1.2 Pandas

Pandas is a Python library that provided efficient and easy to use data structures, called Data Frames, and analysis tools. Data frames are objects that are similar to arrays but with more functionality. Each column within a data frame can be uniquely typed and has a

corresponding label, which makes accessing the data more intuitive and streamlined. When accessing columns, rows, or individual values, the index name and column name can be called explicitly, rather than counting which column the data is and specifying that in the call. This allows for quicker coding that provides an ease of readability that traditional indexing does not.

Aside from storing and manipulating data in data frames, Pandas was useful in importing the data from the simulation and user study. The first step was to import the data from Excel workbooks (xlsx) or comma separated values (csv) files. Once read in to the program, storing the data into a data frame was as easy as one line of code.

5.1.3 SciPy

SciPy is a library in Python that provides a wide range of modules for scientific and technical computing. The Stats package of SciPy was used for its statistical analysis functionality. The built in t-test function was used to determine the statistical significance of the results of the user study.

5.1.4 Matplotlib and Seaborn

These two libraries were used to provide various figures for the study data. Seaborn acts as a wrapper to matplotlib, and their functionality is coupled. Box plots from Seaborn were used to show how different groups of data, primarily the difference between VR and desktop, are distributed. A more detailed description is given for the first box plot. Distribution plots from Seaborn were also used and include a histogram of the data, the

columns, and a kernel density estimation function, the curve. This plot shows the probability of a datapoint lying within a range relative to the average. This will become apparent in the results presented below.

5.2 Importing Data

Each trial of the simulation performed by the participants created a csv file that contained the trial data. Distances are measured in centimeters (cm) and time is measured in seconds (s). This data consisted of:

- Tick number, a tick, or frame, being a small slice of time
- Tick length, how long each tick took to compute
- The quadrotor's position relative to the space station, in each axis
- The quadrotor's orientation relative to the space station, in each axis
- The space station's orientation relative to the inertial frame, in each axis
- The location on the spline relative to the space station that was closest to the quadrotor, in each axis
- A Boolean used to determine if the quadrotor was colliding with anything. This was mostly used to tell if the user had landed and for how long.

	Time	Quad Rel X pos	Quad Rel Y pos	Quad Rel Z pos	Quad Rel Roll rot	Quad Rel Pitch rot	Quad Rel Yaw rot	Station X rot	Station Y rot	Station Z rot	Spline Rel X pos	Spline Rel Y pos	Spline Rel Z pos	Distance Along Spline	Percent Along Spline	Error	Is Colliding ?
1	0.010449	-1139.92	-911.464	-4576.85	0.000024	11.30081	-89.9999	0.250787	0	0	-1138.99	-1036.19	-4539.92	498.3942	0.022777	260.1621	0
2	0.063256	-1139.92	-830.902	-4596.45	0.000021	10.03265	-89.9999	1.518152	0	0	-1139.35	-1035.17	-4535.96	512.5864	0.023425	426.0674	0
3	0.07159	-1140.14	-919.257	-4598.41	-1.36181	12.78002	-90.0257	1.718165	0	0	-1139.35	-1035.17	-4535.96	512.5864	0.023425	426.0674	0
4	0.079924	-1140.36	-918.656	-4597.63	-1.78717	13.4995	-90.0536	1.918171	0	0	-1137.16	-1041.65	-4561.22	421.8178	0.019277	256.6283	1
5	0.088257	-1140.58	-918.072	-4596.9	-2.20751	14.20663	-90.0888	2.118178	0	0	-1137.16	-1041.65	-4561.22	421.8178	0.019277	256.6283	0
6	0.096591	-1140.8	-917.503	-4596.22	-2.62315	14.90158	-90.1315	2.31819	0	0	-1137.16	-1041.65	-4561.22	421.8178	0.019277	256.6283	0
7	0.105875	-1141.01	-915.121	-4595.96	-3.03423	15.56174	-90.1815	2.541003	0	0	-1137.16	-1041.65	-4561.22	421.8178	0.019277	256.6283	0
8	0.116174	-1141.26	-912.569	-4595.71	-3.48754	16.28492	-90.2456	2.78818	0	0	-1137.45	-1040.76	-4557.73	434.3767	0.019851	267.5109	0
9	0.126865	-1141.52	-911.173	-4595.23	-3.98514	17.08858	-90.3274	3.044774	0	0	-1137.45	-1040.76	-4557.73	434.3767	0.019851	267.5109	0
10	0.137673	-1141.8	-910.309	-4594.71	-4.49591	17.91143	-90.424	3.304146	0	0	-1137.45	-1040.76	-4557.73	434.3767	0.019851	267.5109	0

Table 2: First 10 Rows of Data Output from the Simulation for Trial

Each file was loaded into the Python program and assigned to the appropriate user. The file names consisted of the date and time, user code, and the mode, sensitivity, medium and trial number. A sample file name is:

“2019-2-2-18-48-BY261-AC-high-2-Desktop.csv”

“2019-2-2-18-48” describes the year, month, day, hour, and minute the trial was saved. “BY261” is the user code. “AC-high” describes the mode and sensitivity, in this case, Attitude-Control High sensitivity mode. “Desktop” describes the viewing medium, with the other option being “VR”.

This data was used to organize each trial into the appropriate group so that cross-analysis could be done on the varying flight modes and between-users. The Modified Cooper-Harper results were stored in a separate Excel file with a separate sheet for each user. This file was read into the program. Each sheet was matched to the user and the MCH scores organized.

A final spreadsheet containing the participants’ questionnaire data was imported as well. This file contained the information necessary to sort the users into different skill groups.

5.3 MCH Data Analysis

This section details the steps I took to analyze the Modified Cooper-Harper rating scale and presents the results. These results are the subjective measures of how difficult controlling the drone was in both the VR and desktop modes. The discussion section will elaborate on the significance of these results and how it relates to the primary objective of determining how VR can augment space system research and design.

5.3.1 Analysis Steps

Presented here is an overview of the steps taken to analyze and present the data.

- Aggregate the data
- Normalize the data
- Determine the averages and standard deviations of each handling quality for each control mode with all of the users' data combined
- Compare flight mode ratings for VR and desktop and perform a T-test on each of the 20 pairs of data and describe the statistical significance
- Present data in colorized table format and describe
- Present VR/desktop comparisons in distribution/scatter plots and describe

5.3.2 Aggregating MHC Data

The users completed the trial three times for each flight configuration and subsequently used the MCH scale to rate the five handling qualities for both VR use and desktop use. They were unable to modify their previous scores once another trial was run; however, they did have visual access to their previous scores and could use them as a guideline to help them rate their current flight mode. The ratings, on a scale of 1 to 10, provide a numerical subjective dataset, which could be compared between users as described above.

MCH-Scale Ratings											
VR Mode						Desktop Mode					
Rate Control - High						Rate Control - High					
Average	Hovering	Prograde	Retrograde	Axial	Cornering/Obstacles	Average	Hovering	Prograde	Retrograde	Axial	Cornering/Obstacles
3.4	2	5	4	3	3	3.6	3	4	4	3	4
Rate Control - Low						Rate Control - Low					
Average	Hovering	Prograde	Retrograde	Axial	Cornering/Obstacles	Average	Hovering	Prograde	Retrograde	Axial	Cornering/Obstacles
5	4	5	5	6	5	2.8	2	3	4	2	3
Attitude Control - High						Attitude Control - High					
Average	Hovering	Prograde	Retrograde	Axial	Cornering/Obstacles	Average	Hovering	Prograde	Retrograde	Axial	Cornering/Obstacles
1.8	1	1	3	2	2	2.6	2	1	4	3	3
Attitude Control - Low						Attitude Control - Low					
Average	Hovering	Prograde	Retrograde	Axial	Cornering/Obstacles	Average	Hovering	Prograde	Retrograde	Axial	Cornering/Obstacles
3.6	4	2	5	3	4	5	5	2	6	5	7

Table 3: MCH Rating Results from User BY261

5.3.3 Normalization of the MCH Results

While conducting the user study, it was easy to see that users rated the MHC scale for the handling qualities of the various flight modes differently. Some users would rate all of the values higher or lower than others, which increases the noise when making comparisons between users. My objective was to determine how users scored each flight characteristic per flight mode differently when using the VIVE over the traditional desktop monitor. Inferential statistics are not normally used when analyzing MCH results; however, this type of analysis is included. Two datasets of MCH results were aggregated, one for the traditional desktop mode and one for the VR mode. Comparing the two was the primary goal within the analysis and normalization between users was used as an attempt to combat the non-linearity of the scale.

The average MCH value of each user was found and divided by the total average rating for all users and all flight characteristics. Each of the users' scores was divided by that value. This normalized the data between users in such a way that a T-test could be conducted to determine the statistical significance. Another advantage of this approach was the usefulness of visualizing the probability plots with both VR and desktop scores together. It was easy to see any visible trends in the data and what relationships should be further explored.

Table 3 above shows an example of user ratings for the MHC scale while Table 4 below shows the same example with values normalized to the rest of the users.

MCH-Scale Ratings											
VR Mode						Desktop Mode					
Rate Control - High						Rate Control - High					
Average	Hovering	Prograde	Retrograde	Axial	Cornering/ Obstacles	Average	Hovering	Prograde	Retrograde	Axial	Cornering/ Obstacles
4.02	2.36	5.92	4.72	3.55	3.55	4.258	3.55	4.73	4.73	3.55	4.73
Rate Control - Low						Rate Control - Low					
Average	Hovering	Prograde	Retrograde	Axial	Cornering/ Obstacles	Average	Hovering	Prograde	Retrograde	Axial	Cornering/ Obstacles
5.91	4.73	5.91	5.91	7.09	5.91	3.31	2.36	3.55	4.73	2.36	3.55
Attitude Control - High						Attitude Control - High					
Average	Hovering	Prograde	Retrograde	Axial	Cornering/ Obstacles	Average	Hovering	Prograde	Retrograde	Axial	Cornering/ Obstacles
2.126	1.18	1.18	3.55	2.36	2.36	3.074	2.36	1.18	4.73	3.55	3.55
Attitude Control - Low						Attitude Control - Low					
Average	Hovering	Prograde	Retrograde	Axial	Cornering/ Obstacles	Average	Hovering	Prograde	Retrograde	Axial	Cornering/ Obstacles
4.256	4.73	2.36	5.91	3.55	4.73	5.9	5.91	2.36	7.09	5.91	8.23

Table 4: Normalized MCH Rating Results from User BY261

Each value in the above table was multiplied by a scaler ratio that was equal to the aggregate average for all users over the current user's average. That forced each user to have the same average and their VR and desktop results could be compared between users. Although the MCH scale is a 1-10 rating system, each value has a corresponding description, and the 1-10 scale may not have equal weights between numbers. Had the users rated the flight characteristics on a normal 1-10 scale and not had the MCH descriptions available to them, their ratings may have differed. This also causes some potential complications when normalizing and averaging the scores, though, it was not known how the data would be affected and was not considered for the analysis.

5.3.4 MHC Statistics

The statistics used to determine the variances between the ratings were the averages between users, standard deviation from the averages, and a p-value that determines whether or not the data is statistically significant or if it has a high probability of only being different due to randomness in the sampling.

5.3.4.1 Averages

Determining the averages is straightforward. Each flight mode rating for each handling quality for every user was aggregated into an array. Each value was summed and the result was divided by the number of users to determine the average rating. Table 5 below shows the aggregate average for each user’s MCH results, once again, using the normalized values. From this data, we can see how the different handling qualities compare to one another for the entire study population.

MCH-Scale Ratings											
VR Mode						Desktop Mode					
Rate Control - High						Rate Controlled - High					
Average	Hovering	Prograde	Retrograde	Axial	Cornering/Obstacles	Average	Hovering	Prograde	Retrograde	Axial	Cornering/Obstacles
4.10	3.55	4.05	4.51	4.10	4.28	3.95	3.51	3.66	4.48	3.99	4.11
Rate Control - Low						Rate Controlled - Low					
Average	Hovering	Prograde	Retrograde	Axial	Cornering/Obstacles	Average	Hovering	Prograde	Retrograde	Axial	Cornering/Obstacles
4.41	3.84	3.99	4.99	4.41	4.80	4.70	3.89	4.18	5.23	4.97	5.24
Attitude Control - High						Attitude Control - High					
Average	Hovering	Prograde	Retrograde	Axial	Cornering/Obstacles	Average	Hovering	Prograde	Retrograde	Axial	Cornering/Obstacles
3.44	2.88	2.80	4.09	3.75	3.70	3.80	3.40	3.03	4.42	4.10	4.05
Attitude Control - Low						Attitude Control - Low					
Average	Hovering	Prograde	Retrograde	Axial	Cornering/Obstacles	Average	Hovering	Prograde	Retrograde	Axial	Cornering/Obstacles
3.89	3.42	3.07	4.68	4.02	4.26	4.58	3.81	3.35	5.20	5.12	5.40
Average	3.42	3.48	4.57	4.07	4.26	Average	3.65	3.56	4.83	4.55	4.70

Table 5: Normalized MCH Rating Results for All Users

The color variation emphasizes the better and worse ratings with green cells being rated better, or closer to 1, and red cells worse, or closer to 10. Based on the values shown above, the best rated control mode was Attitude-Control with High sensitivity for both VR and desktop, with the VR mode being even more favorable. The best rated maneuver, on average, was hovering while in VR and prograde while using a desktop monitor.

5.3.4.2 Standard Deviations

Many programs and libraries have built in functions to determine the standard deviation. The Numpy library in Python was used to determine both the averages and the standard deviations.

5.3.4.3 P-values and the T-Test

The p-values are more complex and have a specific use. With any sampled or collected set of data, there is noise and randomness that effect the values. When comparing two or more sets of data, determining the variation of mean averages is useful, but may be susceptible to the noise of sampling. To determine whether or not the results hold any statistical significance, several tests can be used. For this set of data, a paired T-test was used as each dataset was compared from the same user on the same flight configuration. Because we normalized the data, comparing between users was possible with this test. Other significance tests include the Student's T-test, ANOVA, and CHI Squared tests.

The paired T-test runs is a hypothesis test and the result determines whether or not to accept or reject the hypothesis. As is standard, the hypothesis, called the null hypothesis,

assumes that the true mean difference, the true difference in the two averages of the datasets, is zero. The T-test calculates a t-statistic that is compared to t values in a table. Matching the t-statistic on the table correlates to a p-value. This value is the probability that the hypothesis is correct. The t-statistic can be calculated with the following equation:

$$t = \frac{\bar{X}}{\left(\frac{std}{\sqrt{n}}\right)} = \frac{\bar{X}}{SE(\bar{X})}$$

Where \bar{X} = *mean difference*, n = *number of number of samples* and std = *standard deviation of the mean difference*. Large t-statistic values indicate a low p-value which in turn indicates a low probability of the null hypothesis being true.

The table used to find the p-value is dependent on the standard deviation, the mean, and the sample size. Higher samples provide a more complete picture of the data and can reduce the sampling noise. An acceptable probability of statistical significance is 95% and is represented as $\alpha = 0.95$. If $1 - \sigma > \alpha$, the data is considered to have statistical significance and the null hypothesis is rejected, accepting the claim that the two datasets have different means.

Normalized MCH Results Comparison for each Flight Characteristic of each Flight Mode						
Mode	p-value	VR avg	Desktop Avg	Mean Difference	VR std	Desktop std
RCHh	0.899	3.55	3.51	-0.04	1.10	1.17
RCHp	0.105	4.05	3.66	-0.39	1.04	0.75
RCHr	0.864	4.52	4.48	-0.04	0.99	1.03
RCHa	0.683	4.10	3.99	-0.11	1.13	0.82
RCHc	0.560	4.28	4.11	-0.17	1.11	1.29
RCLh	0.891	3.84	3.89	0.04	1.09	1.57
RCLp	0.602	3.99	4.18	0.19	1.08	1.47
RCLr	0.501	4.99	5.23	0.24	1.33	2.25
RCLa	0.220	4.41	4.97	0.56	1.21	1.89
RCLc	0.235	4.80	5.24	0.44	1.40	1.45
ACHh	0.075	2.88	3.40	0.52	1.43	1.55
ACHp	0.273	2.80	3.03	0.23	0.97	0.97
ACHr	0.058	4.09	4.42	0.33	1.13	0.94
ACHa	0.209	3.75	4.10	0.35	1.49	1.23
ACHc	0.110	3.70	4.05	0.35	1.12	0.87
ACLh	0.156	3.42	3.81	0.39	1.20	1.18
ACLp	0.245	3.07	3.35	0.28	0.92	0.99
ACLr	0.048	4.68	5.20	0.51	1.32	1.65
ACLa	0.003	4.02	5.16	1.14	1.78	2.14
ACLc	0.001	4.26	5.40	1.14	1.42	1.96
Total:	0.0004	4.02	4.25	0.24	1.37	1.57

Table 6: Statistic Matrix for Compared Means of MCH Handling Quality Ratings

The table above contains a lot of useful data. The first column, labeled “Mode”, describes which flight mode the rest of the row contains data for. The first two characters discern the control mode with:

RC = Rate-Control and AC = Attitude-Control.

The third character represents the sensitivity level with:

L = low and H = high.

The final character represents the handling quality for that flight configuration with:

h = hovering, p = prograde, r = retrograde, a = axial, c = cornering/obstacles. Using this key, the first row, “RCHh” is decoded as “Rate-Control, High sensitivity, hovering characteristic”.

The second column, labeled “p-value”, presents the probability that the mean difference, in the fifth column, of the averages, in the second and third columns, to be statistically insignificant. The final two columns output the standard deviation for both the VR data and desktop data that are being tested. A similar color scheme was used for this data as before, with greener values indicating more favorable ratings. The exception to this rule is the mean difference column, where red indicates better desktop scores and blue represents better VR scores.

It is clear from the bottom row that for all of the data combined, users had a high probability of rating the VR scores 0.24 points better (with lower values) than the desktop scores, on average. Although the p-scores for most of the individual comparisons do not suggest statistical probability that the mean difference is significant, the aggregate data has a very high probability that the mean difference is significant. This could be due to the lower number of samples used for each subset of data, which has less margin for noise.

Table 6 above also shows that the Rate-Control High sensitivity mode was the only mode which users rated better in desktop. Based on observations from subjects performing the study, this mode was more disorienting and determining the drone’s attitude was more difficult. VR accentuated the disorientation and participants were left asking which way they were facing more than in other modes. The following figure shows the probability density function with an associated histogram of the aggregate data.

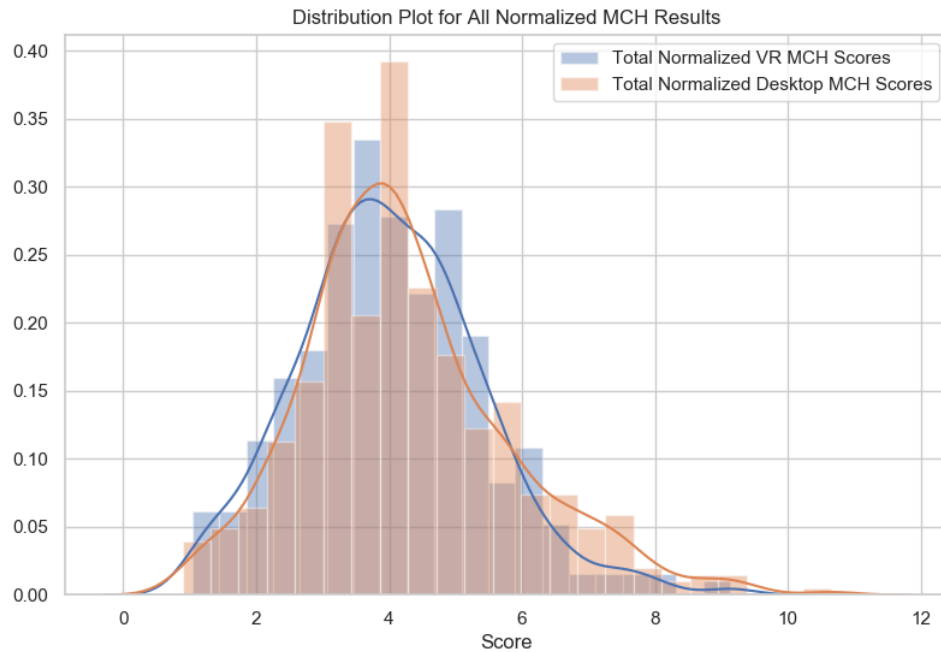


Figure 12: Distribution and Histogram of Aggregate MCH Results

A probability density function (PDF) provides a visual representation of the variation in the data. The x-axis are the scores given by the users. When the function is integrated over a range, the result is a probability, out of 1, that a datapoint will lie within that range. The y-axis represents the integrated variable.

A histogram, the bars in Figure 12 above, visually represents the amount of datapoints that lie within the range the bar covers. As you can see, the two plots are related and coupled. The PDF smooths some of the variation to show the trend in nearby data. Both PDFs and histograms are useful; however, the same inferences can be made and as such, only one is necessary to describe the data.

Although the plots appear to overlap almost completely, the histograms reveal that there are more VR scores closer to a rating of one than desktop scores, and the reverse is true on the other tail of the plot. The higher blue bars represent that more VR datapoints fell in that range, while higher orange scores represent that more desktop scores lie there.

Figures 13-18 are pairplots obtained from the seaborn library of Python. The diagonal plots are PDFs, described above, for the handling quality listed on the axis. The off-diagonal plots are scatter plots, which show the relationship between different flight characteristics. The patterns in the scatterplots can reveal potential correlations in the data and may induce further examination. As the legend suggests, each sub-plot contains a comparison between the VR and desktop viewing medium modes. The lines intersecting in the scatterplots show the trendline for the associated data and the shaded regions surrounding the lines indicate the error.

Once more, the key used previously is used to decode the labels and is presented once more for reference.

RC = Rate-Control and AC = Attitude-Control.

The third character represents the sensitivity level with:

L = low and H = high.

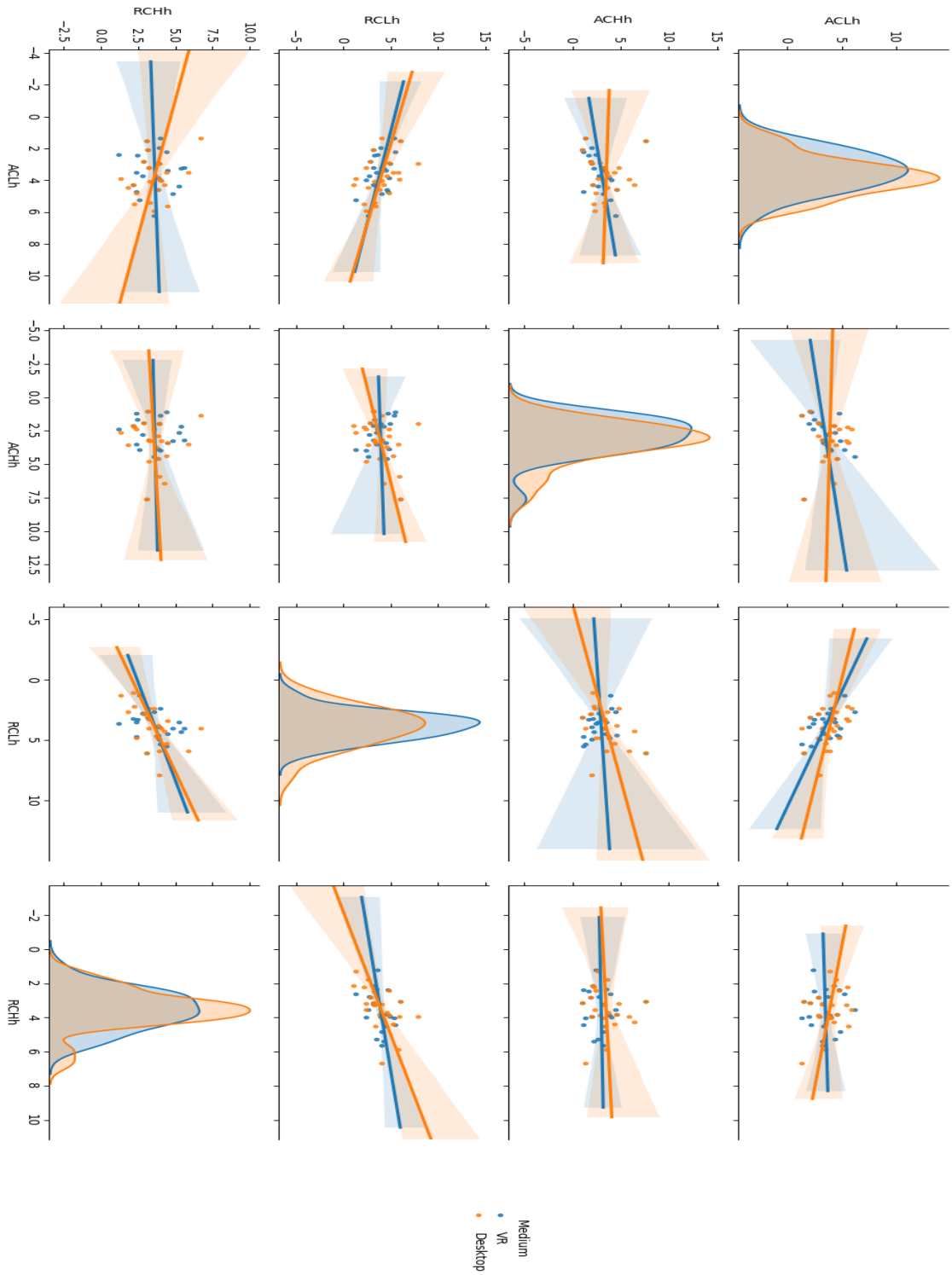
The final character represents the handling quality for that flight configuration with:

h = hovering, p = prograde, r = retrograde, a = axial, c = cornering/obstacles.

Using this key, the top left plot, labeled “ACLh,” represents data for “Attitude-Control, Low sensitivity, hovering characteristic.”

The pairplots are presented in the following order: hovering, prograde, retrograde, axial, and cornering/obstacles. The x-axis of the sub-plots ascends in the order of attitude-control low, attitude-control high, rate-control low, and rate-control high. The y-axis descends in the same order.

The pairplots below are useful for inspecting individual flight characteristics and comparisons between flight modes. The scatterplots do not reveal any significant trends that are easy to discern; however, the probability plots allow the viewer to compare VR and desktop ratings for each flight mode independently of one another. For example, inspection of the probability plot for axial motion using the attitude control, low sensitivity mode, the top left plot in Figure 16, shows a near normal distribution for VR ratings. The desktop ratings for the same mode shows characteristics of a bimodal distribution, which has two peaks. This inspection allows the viewer to deduce that the VR ratings were more consistent around the average, which was lower (better) than the desktop counterpart.



Compared Data for Hovering Ratings

Figure 13: Pairplot for Hovering Flight Characteristic

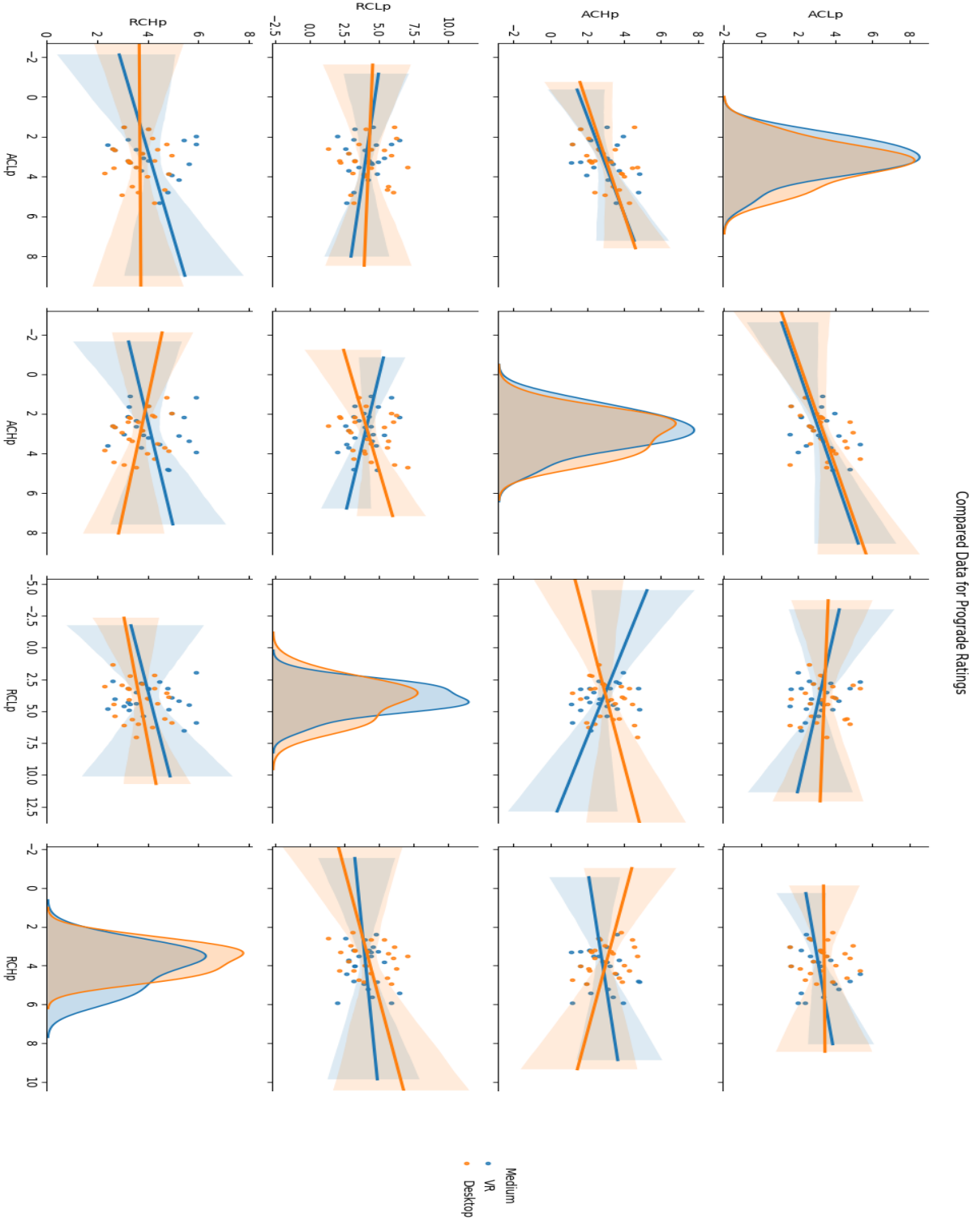


Figure 14: Pairplot for Prograde Flight Characteristic

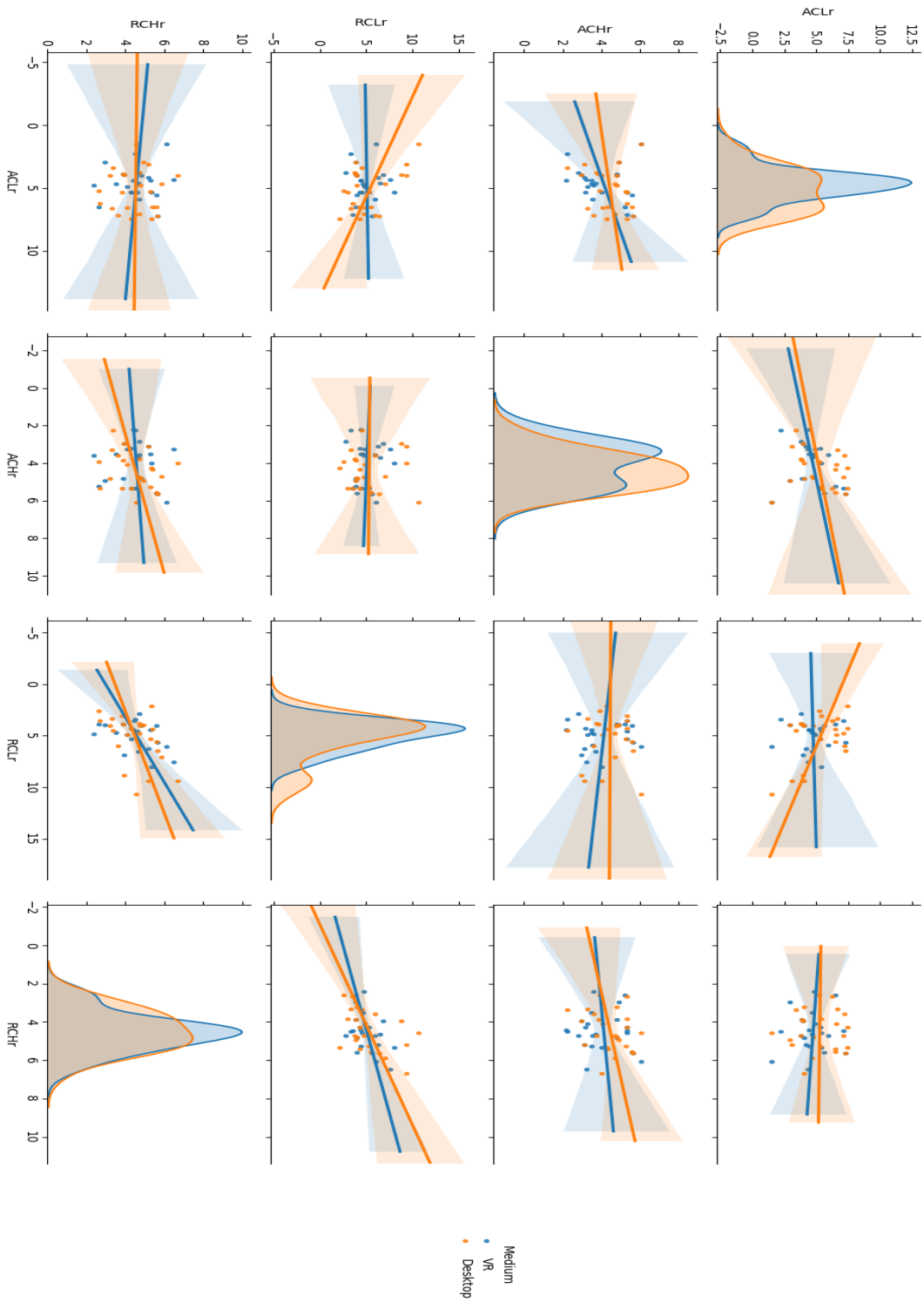
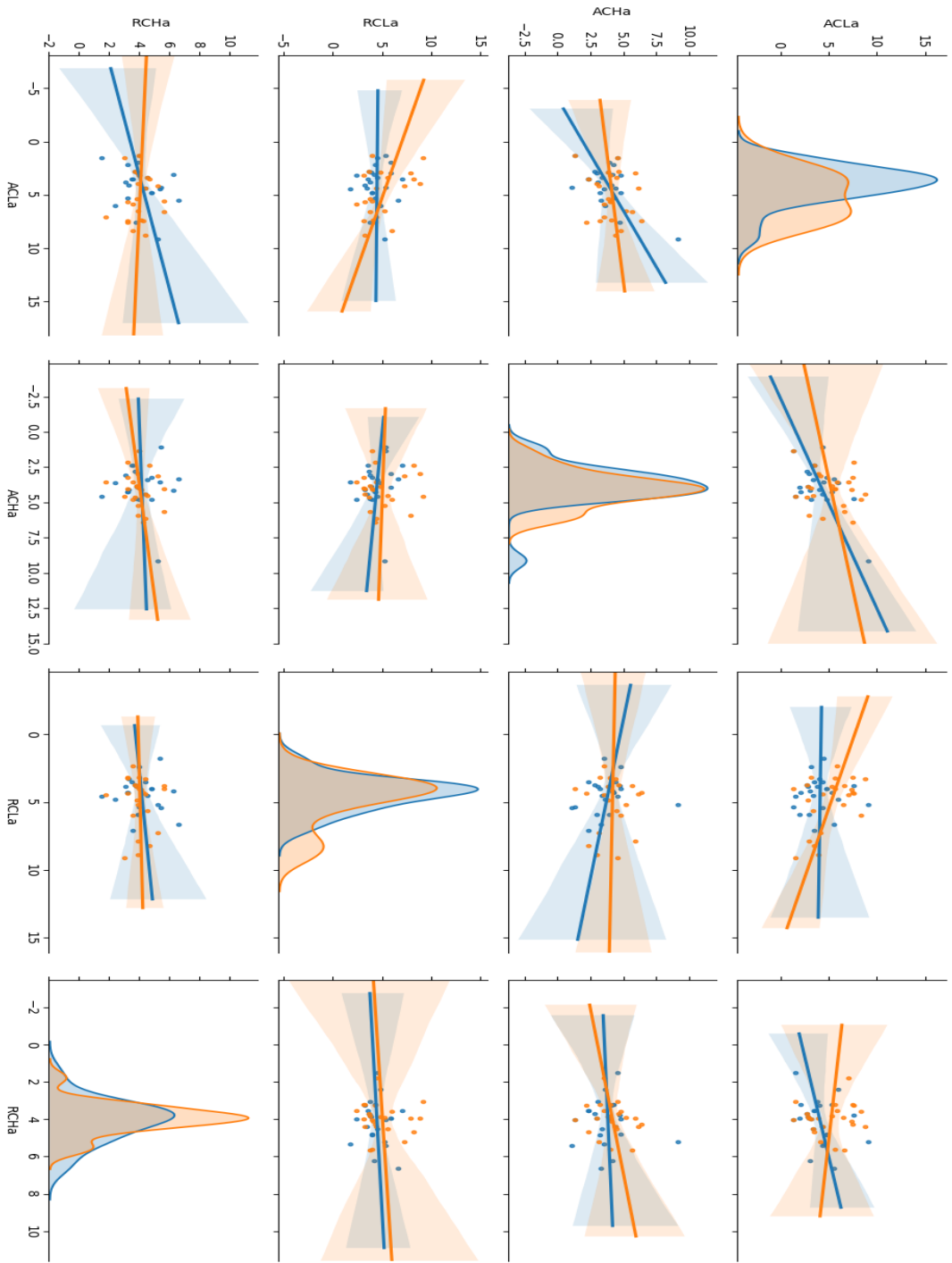


Figure 15: Pairplot for Retrograde Flight Characteristic



Compared Data for Axial Ratings

Mediu
VR
Deu

Figure 16: Pairplot for Axial Flight Characteristic

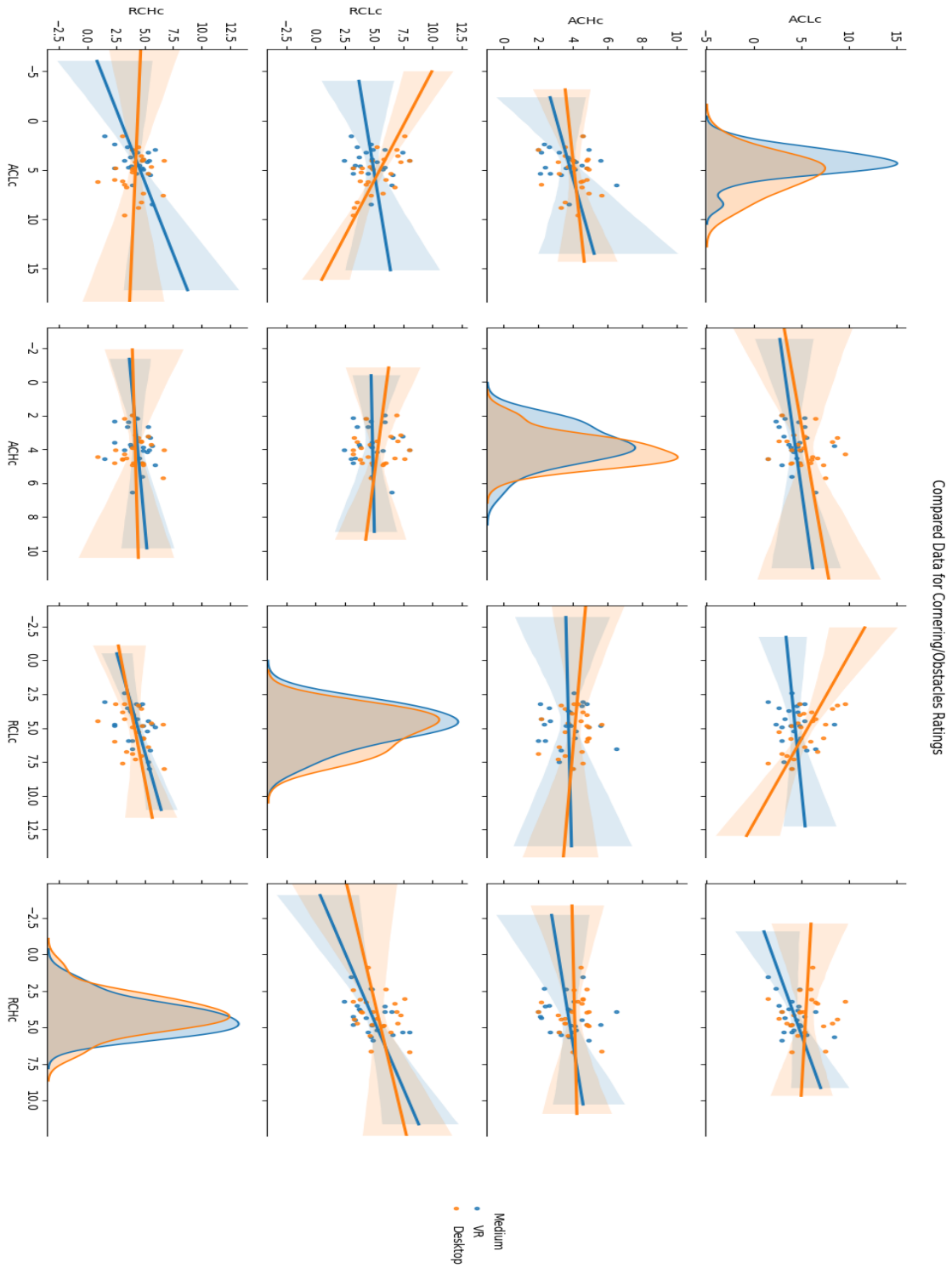


Figure 17: Pairplot for Cornering/Obstacles Flight Characteristic

The following two figures show boxplots detailing handling qualities for each combination of control mode and sensitivity, with blue points representing VR ratings and orange points representing desktop ratings. A boxplot is another visual tool to inspect data and allows the viewer to see the distribution of the data based on the minimum, maximum, first and third quartiles, and median. The box spans the first quartile to the third quartile, or the IQR, and the whiskers, or lines, show the minimum and maximum locations. The line inside the box shows the location of the median datapoint. Outliers are datapoints that are assumed to be abnormal values and are shown to be outside of the minimum and maximum values.

The outliers are not considered for the calculations of the quartiles, minimum and maximum locations. Inspecting figures 18 and 19 below reveals a trend between Attitude-Control and Rate-Control modes. Inspecting the median line, separating the boxes into two tones of grey, of each set of paired data shows that the Attitude-Control ratings, Figure 19, have higher variability between VR and desktop ratings than the Rate-Control mode, Figure 18. These plots are also useful for comparing how different flight characteristics' ratings were spread. It is clear that Attitude-Control High sensitivity ratings for retrograde motion were much more spread out than Attitude-Control Low sensitivity ratings for axial motion. This can be helpful in determining problem areas of the control modes during development.

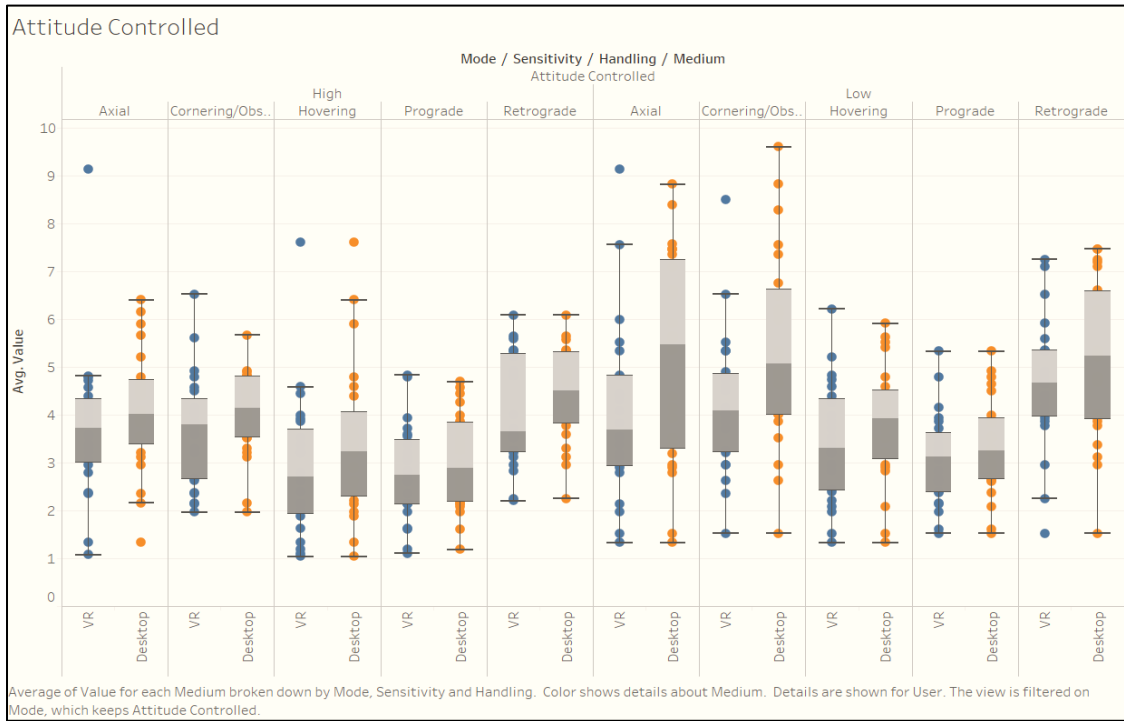


Figure 19: Boxplot for Attitude-Control Flight Modes and Handling Qualities

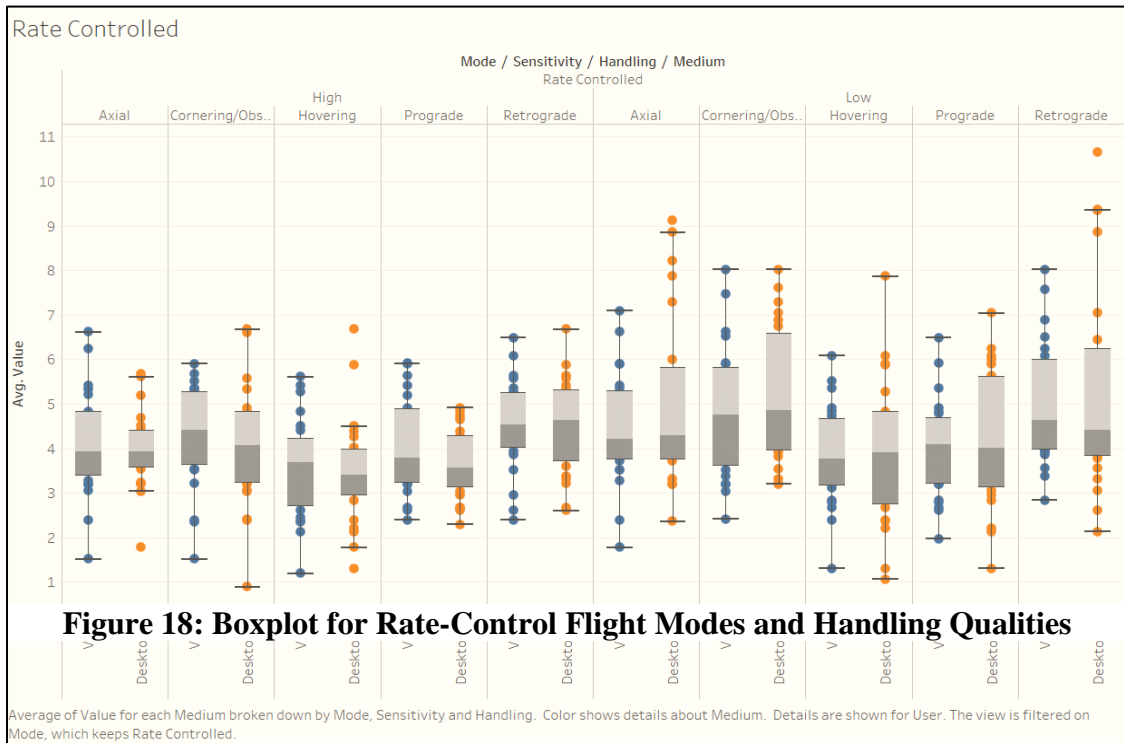


Figure 18: Boxplot for Rate-Control Flight Modes and Handling Qualities

5.4 Trial Data Analysis

5.4.1 Analysis Steps

Presented here is an overview of the steps taken to analyze and present the data.

Similar methods were used for both objective and subjective data.

- Aggregate and reduce the data
- Normalize the data
- Determine the averages and standard deviations of all trials for each control mode with all of the users' data combined
- Compare average errors for VR and desktop modes and perform a T-test on each of the sets of data
- Describe the statistical significance
- Present data in colorized table format and describe
- Present VR/desktop comparisons in distribution and box plots and describe

5.4.2 Data Reduction

The raw data recorded by the simulation required some refinement to account for various factors. The first process used was truncation, or removing data after a certain point. As stated previously, the users were told to follow the path and get within a certain radius of the endpoint. Once there, they ended the simulation with a button on the controller and the data was saved to a file. Many times, a user would fly past the endpoint, land, and reorient themselves in order to get closer. The objective was to reach the radius, regardless of how close within that range they were. When doing so, some of them would deviate quite far from the final location and cause a much longer trial than if they had ended it as soon as they reached the goal. Because of this, it made sense to truncate the data after the goal had been achieved.

The second method of data reduction was to remove each datapoint whose colliding Boolean was true. While it was acceptable to land the vehicle along the way to the goal, some users would spend a significant amount of time reorienting the drone or getting comfortable. It was possible that users could land directly underneath the path while regaining their bearings, which could lower their average errors. In order to reduce these effects, those datapoints were removed. In reference to datapoints which had a true colliding value but were not landed, i.e. when users would hit a wall; the frame rate of the simulation was between 30 and 90 frames per second and hitting a wall would only have a colliding attribute for a frame or two, which did not have a significant effect on the average error. These datapoints were removed as well since there was no way to determine what the user was colliding with.

The third, and final, data reduction technique was to match the tick-rate for the various functions in the simulation. While the HMD or monitor was updating the users' view between 45 and 90 frames per second, the functions within the simulation had predetermined rates at which calculations would be performed. Specifically, the drone's location data updated more frequently than the splines location data by up to a factor of 3. The data was reduced to match that of the splines update rate and was consistent throughout the trials. This ensured that the variability of the frame rate did not influence the data.

5.4.3 Normalization of Average Errors Between Users

Similar to the method described for the MCH data, the average errors for each user were normalized to the average of all users' data combined. Once again, this forced the mean average error for each user to be identical. Doing so prevented users who were better or worse than others to influence the results disproportionately.

5.4.4 Average Error Statistics

The averages and standard deviations of the average errors were calculated similarly to the MCH results and can be referred to the above section. The T-test results were also calculated in the same way, using the Paired T-test, aggregated and normalized.

Initially, I had calculated the distributions and T-test results for the entire set of data, with the VR trials in one array and the desktop trials in another. After determining the average errors and the standard deviations for each set, I ran a Paired sample T-test

and found the results to be mostly statistical insignificant. I then performed data analysis on each set of trials that had the same configuration, testing the difference between VR and desktop. Regardless of the results, this method was inclined to provide more accurate results as the VR trials were compared to their desktop counterpart. This method reduced the noise stemming from the different flight configurations grouped together before the viewing medium comparisons were made.

A pre-study questionnaire asked the participants questions regarding their experience levels in the following categories: quadrotor flight, aircraft flight, simulations, using joysticks, and VR. The answers were aggregated and the users separated into two groups, more experienced and less experienced. Bifurcating the users allowed a determination to be made about their performances and how they compared.

The following figure displays the statistical variables for the average errors. The key, color scheme, and relations are described in the MCH Statistics section and can be used as a reference. This figure includes a decomposition of the data in relation to the two experience levels.

Normalized Trial Results Comparison for each Flight Mode						
Mode	p-value	VR avg	Desktop Avg	Mean Difference	VR std	Desktop std
All Users						
RCH	0.258	593.82	575.59	-18.23	95.91	107.97
RCL	0.457	597.38	612.71	15.34	130.27	135.76
ACH	0.035	482.07	536.16	54.09	96.12	197.71
ACL	0.088	516.19	547.77	31.58	116.73	111.02
Total:	0.050	547.36	568.06	20.69	120.91	145.06
Experienced Users						
RCH	0.063	621.20	578.36	-42.83	101.03	105.02
RCL	0.340	595.87	619.30	23.43	112.41	130.89
ACH	0.039	467.54	560.14	92.60	87.17	260.34
ACL	0.475	501.58	517.69	16.10	86.68	104.84
Total:	0.142	546.55	568.87	22.32	115.75	166.00
Inexperienced Users						
RCH	0.708	564.16	572.58	8.42	81.41	112.50
RCL	0.848	599.01	605.57	6.57	148.86	142.36
ACH	0.587	497.81	510.18	12.37	103.90	87.86
ACL	0.109	532.01	580.37	48.36	141.98	109.66
Total:	0.197	548.25	567.17	18.93	126.65	118.87

Table 7: Statistic Matrix for Compared Means of Trial Data

We can see that for the aggregate data for all users, the mean difference is 20.69cm and the p-value is 0.05 which satisfies the statistical significance test. The two experience groups' total p-values were above the 0.05 threshold to assume statistical significance. The discrepancy in this data can be attributed to insufficient data in the two categories to overcome noise.

The following distribution plots show the correlation between VR and desktop trials for each flight mode.

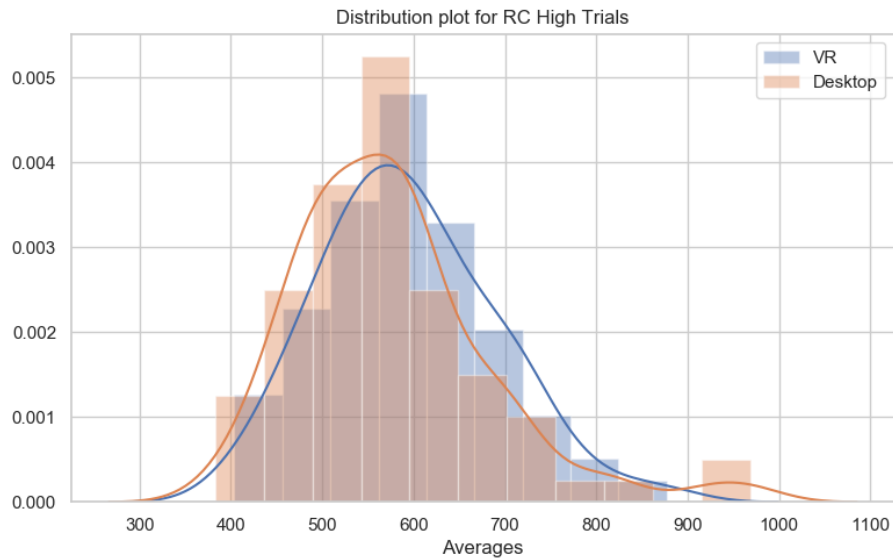


Figure 20: Distribution and Histogram of Rate-Control High Sensitivity Mode

The desktop peak in the Figure 20 is higher and closer to the y-axis, suggesting lower averages and a smaller standard deviation. Although the data is not perfectly normally distributed, so the assumption that these suggestions are true cannot be made, visually inspecting these plots give the reader a better understanding of the data.

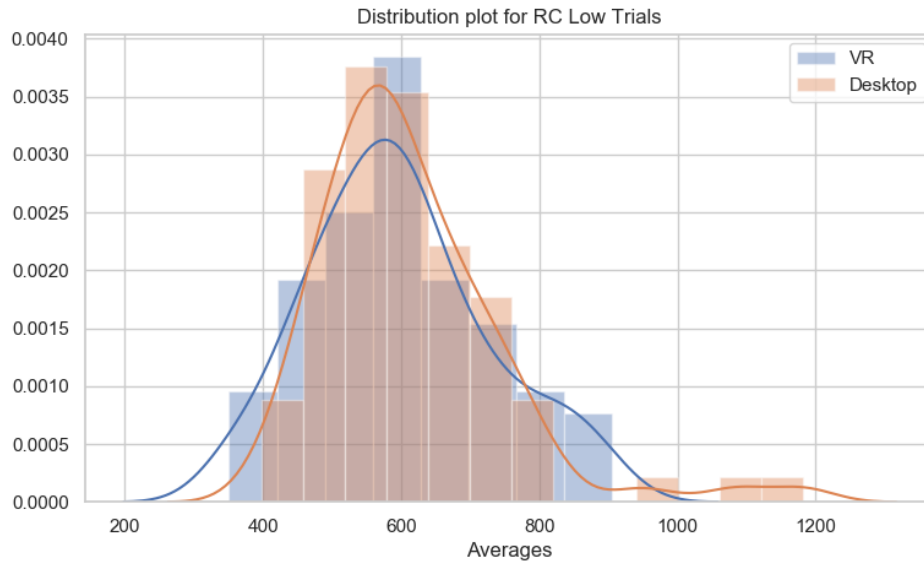


Figure 21: Distribution and Histogram of Rate-Control Low Sensitivity Mode

The distributions for the flight mode for the above figure overlap one another closely. With the exception of the highest desktop averages, these two datasets would have similar averages. Table 7 confirms these assumptions as this mode has the smallest difference between mean averages.

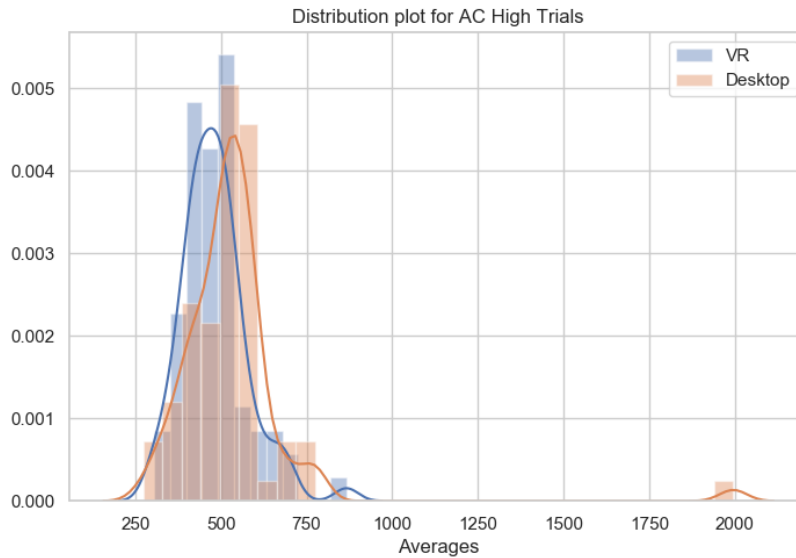


Figure 22: Distribution and Histogram of Attitude-Control High Sensitivity Mode

This figure shows two distinct peaks for both VR and desktop trials. While they do overlap, the curve allows the user to infer that VR trials had better performance compared to desktop trials. There is one value for desktop trials that is severely removed from the rest of the trials. I looked at the data both ways, truncating that datapoint and not. The single value did not meaningfully change the plot or the averages, and was left in to include all the data recorded.

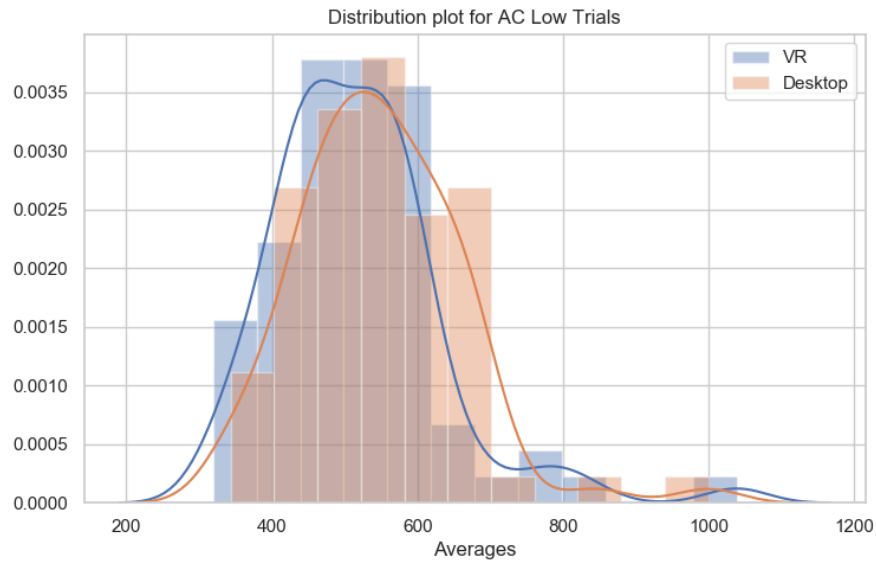


Figure 23: Distribution and Histogram of Attitude-Control Low Sensitivity Mode

The two distributions shown in this figure are similar and shifted from one another, with minor differences. Due to their similarity in shape, assuming the VR trials had lower averages is reasonable, and Table 7 confirms the assumption.

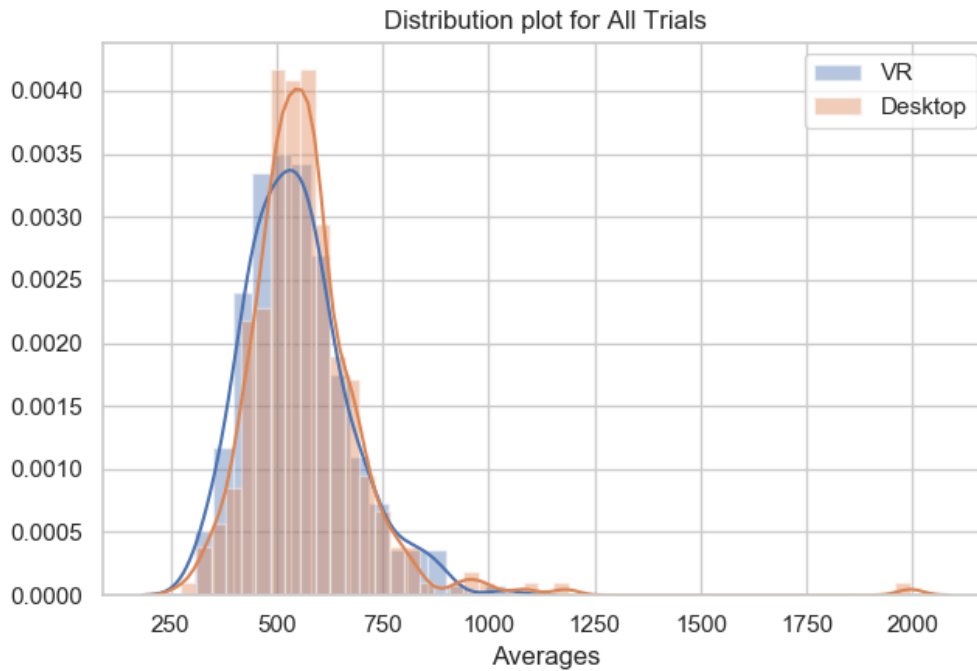
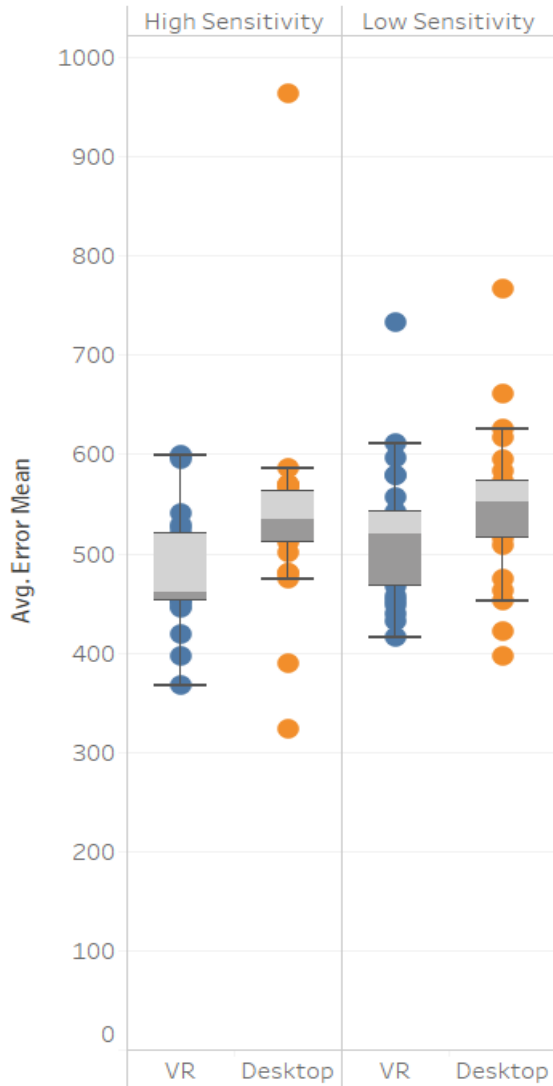


Figure 24: Distribution and Histogram for All Trials

With the exception of one errantly high desktop value, this plot shows that the VR averages were lower, though the distribution is more spread apart. Displayed below are boxplots for the average errors of each flight mode. Similar to the plots shown for the MCH results, these plots enhance visual inspection of the data and show clear differences between control modes, sensitivities, and viewing mediums. Similar to the MHC results boxplots, Figures 25 and 26 show how the data is spread out in a different manner from distribution plots. The trend seen in the MCH ratings that VR and desktop median values have more variance using the Attitude-Control mode continues in these plots. This distinction reveals that users felt that the difficulty of flying the drone in the different control modes matched their performance.

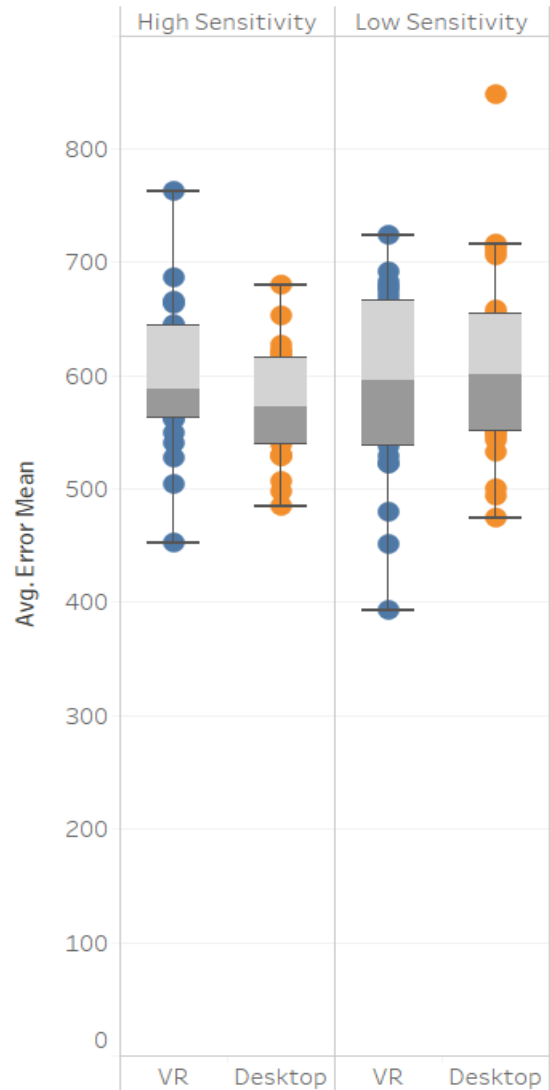
Attitude Controlled



Average of Averages for each Medium broken down by Mode and Sensitivity. Color shows details about Medium. Details are shown for User. The view is filtered on Mode, which keeps A&RC.

Figure 26: Box Plots for Attitude-Control Mode

Rate Controlled



Average of Averages for each Medium broken down by Mode and Sensitivity. Color shows details about Medium. Details are shown for User. The view is filtered on Mode, which keeps RC.

Figure 25: Box Plots for Rate-Control Mode

5.4.5 Post-Study Questionnaire Results

The following questionnaire was given to each participant following the end of their trials and their answers recorded:

Post-Study Interview	
Question 1:	
On a scale of 1 to 10, how difficult was it to fly the drone using the Desktop setup?	
Question 2:	
On a scale of 1 to 10, how difficult was it to fly the drone using the VR setup?	
Question 3:	
Which flight mode do you think was the best for the task?	
Question 4:	
Did you feel as though using VR made choosing the best flight mode easier?	
Question 5:	
If you were to design and test your own control mode, would you want to incorporate VR into your study?	
Question 6:	
If yes, what made using VR beneficial to this type of testing?	
Question 7:	
What could have made this study better? (i.e. better graphics, different controller, better control algorithms, etc.)	
Question 8:	
Do you have any final comments about the study?	

Table 8: Post-Study Questionnaire

The results to these questions were aggregated and are shown below.

Question 1: On a scale of 1 to 10, how difficult was it to fly the drone using the desktop setup?

The average answer was 5, with scores ranging from 2 to 8.

Question 2: On a scale of 1 to 10, how difficult was it to fly the drone using the VR setup?

The average answer was 4.625, with scores ranging from 2 to 7.

The comparison of these two questions reveal that users aggregately thought flying the drone in the desktop mode was 8% more difficult than while using VR.

Question 3: Which flight mode do you think was the best for the task?

Best Flight Mode for Task			
Rate Controlled		Attitude Controlled	
Low	High	Low	High
4	3	5	12

Table 9: Question 3 Results

The numbers in the cells above refer to the number of participants who selected that flight mode. Each flight mode was selected; however, the Attitude-Control, High sensitivity mode was the popular vote with half of the users agreeing.

Question 4: Did you feel as though using VR made choosing the best flight mode easier?

20 users agreed that yes, VR helped, 4 users agreed that it did not.

Question 5: If you were to design and test your own control mode, would you want to incorporate VR into your study?

22 users answered yes, 2 users answered no.

The remaining questions were not quantifiable and could not be aggregated; however, some notable responses are shown below.

Question 6: If yes, what made using VR beneficial to this type of testing?

“Controls and physics clicked once in VR”

“3D viewing and depth perception is a lot better. Relative speeds were easier to understand”

“Perspective and scaling. Inputs look more to scale.”

“You can feel the controls better in VR.”

Question 7: What could have made this study better?

“Decoupling control inputs. Individual input sensitivity”

“Equal frame rates, decoupled yaw control from throttle, Throttle to triggers would be good, better control.”

Question 8: Do you have any final comments about the study?

“Liked the environment and had to learn the physics.”

“Big learning curve.”

CHAPTER VI

CONCLUSION

6.1 Study Observations

6.1.1 Modified Cooper-Harper Rating Discussion

The MCH scale revealed that users preferred the handling qualities while using VR. With an average rating of 0.24 points better than desktop scores, and a p-score of 0.0004, these results were statistically significant. The individual ratings had less statistical significance levels, though some of them did meet the 0.05 criteria. Since individual ratings had only 5% of the datapoints as the aggregate total, the noise was more dominant. A higher sample size would have reduced this noise and would have had a better chance at determining the difference between viewing medium scores.

6.1.2 Post-Study Questionnaire Discussion

The results from the post-study questionnaire followed the trend of preferring VR. The users rated the difficulty of controlling the drone 8% harder for desktop trials and the majority of users, 83%, believed VR to have helped them choose the best flight mode for the task.

6.1.3 Average Error Analysis Discussion

The analysis of the aggregate average errors for each trial revealed better performance in VR by an average of 3.6%.

6.1.4 Experience Observation

The results between experience levels for the users' average errors did not reveal a correlation. Intuitively, the more experienced group was expected to perform better in VR than in desktop as a result of having more control over the drone and being less disoriented while using VR. This did not appear to be the case. Users who had experience with drone flight prior to this study had a more difficult time getting used to the controls, as they had to ignore their normal instincts and learn how to fly in this environment. Some users without previous quadrotor or flight experience picked up the controls very quickly, having only experience with joysticks and gaming. Those users performed the best.

6.1.5 Anomaly

An anomaly occurred with one user that I found to be interesting. When practicing the controls at the beginning of the study, the user could not complete the course and did not appear to get better over the span of 30 minutes. I suggested we stop and I could find another participant; however, he opted to try it out in VR. After about one minute, he could finish the course with relative ease. He proceeded to finish the study, performing all of the VR trials before moving to the desktop trials. He remarked that once he was immersed in

the simulation in VR, the controls clicked, and he had an intuitive understanding of the mechanics.

While several other users mentioned that VR helped them understand the controls, no other user experienced such a large impact from VR.

6.2 Study Influences

Several factors aside from nausea had potential to influence the results and were minimized as much as possible. As mentioned in the Study Design Chapter, users performed the trials in different orders to combat one of these factors, the learning curve. While it was more difficult to have the participants perform the VR trials first (i.e. potentially due to nausea that might be a factor for following trials), I found that they could handle the VR trials before the desktop trials for the latter set of control modes. This allowed for a more random mode sequence.

Another influence was specific limitations on how the hand controller was set up for the trials, which had to be consistent for all participants. Several users suggested allowing the bumper triggers on the Xbox One controller to control the yaw axis of the quadrotor; they experienced difficulty staying oriented the way they intended to fly. This would have worked well for advanced RC pilots, but made the task more challenging for others. Another solution would be to use a traditional quadrotor controller, which has more degrees of freedom (more inputs), and again is more complicated for a beginner.

The study was not gender-balanced, and making claims for male versus female users is not possible. This could have been included with more emphasis on making participant

selections based on gender; however, there would not have been a spectrum of experience levels which was preferred.

When requesting users to provide information regarding their experience levels for various skills, the options (none, novice, intermediate, experienced, expert, and professional) did not have definitions. Their responses were subjective based on how they interpreted the options and should be considered a limitation for the data on experienced versus inexperienced groups.

6.3 Study Outcome

The motivation for performing this research and analysis originated from the development of SpaceCRAFT. I sought to demonstrate virtual reality's utility in the context of evaluating human interfaces for future control systems – potential applications for SpaceCRAFT.

The study successfully provided results that compare the advantages of utilizing VR to traditional two-dimensional viewing mediums. Although, for practical reasons, the simulation did not follow VR best practices for avoiding nausea, the results still point towards positive benefits of using VR. This study (the simulation test case used) had a higher probability of finding null or poor results due to nausea than other potential applications; however, the negative effects were still outweighed by the advantages.

The results of the study prove VR to be an advantageous engineering analysis tool for this specific application. The Literature Review Chapter of this thesis and the results

presented here suggest that utilizing VR for other space systems design challenges is viable, valuable, and should be explored when the opportunity is available.

6.4 Lessons Learned and Future Work

Lessons learned throughout the design, development, and implementation of this simulation and study can be used to develop higher fidelity simulations as well as nullify some of the issues experienced herein. The implementation of the control modes was an approximation to real-world counterparts. As described in the Simulation Design Chapter, the quadrotor dynamics were assumed to be sufficient; however, choosing to not implement more realistic controls affected the performance of the users. Developing physics-based controls is something to consider as future work. Additionally, more frequent VR testing may have identified potential inefficiencies that could be addressed and, in the future, should be practiced. Something to consider when running user studies in VR is whether or not you should sacrifice visual fidelity in favor of better performance, and where to draw the line.

An important lesson learned was to ensure your data collection works as intended, and test your analysis methods prior to conducting the user study. I realized after three days of running trials that I was not collecting enough data and those trials could not be used. Setting up data analysis software early also allows you to refine your processes as you go, and get an idea if the data you collect behaves as you expect it to before it is too late.

The field of virtual reality is constantly expanding. Much research has yet to be done, and more studies like this one help find specialized uses for future technologies. I think it

is possible to have multiple objectives for a single user study that may not be apparent at first glance. For example, about halfway through this study, I had the idea to give each user a balance test to determine how well their inner-ear handles physical balance and compare it to their performance in VR. This could have been related to the level of nausea, if any, they felt, or directly to the scores for my primary objective. While determining if the inner-ear has any correlation to VR performance does not directly relate to my study, the simulation I designed seemed to be a good testbed for additional research.

Studies related to this thesis can further explore the benefits of VR related to space mission designs in a collaborative environment. Simulating docking maneuvers with varying controls for different flight phases would further this research on the front of determining control schemes for vehicles in extreme environments. Full space mission mock-ups can provide engineers feedback for unforeseen design flaws, such as whether or not a spacesuit allows the user to properly control a rover based on movement and vision restrictions. Teleoperations can be simulated within a real-time VR environment to test communication arrays and robotic control modes.

While many studies are yet to come, the research presented within this document and many similar to it point toward positive response from utilizing virtual reality for space system design.

REFERENCES

- [1] *Artificial Gravity by Centrifuge* Available: <https://www.permanent.com/zero-gravity-effects-on-humans.html>.
- [2] “(PDF) Artificial Gravity in Theory and Practice - ResearchGate” Available: https://www.researchgate.net/publication/305768041_Artificial_Gravity_in_Theory_and_Practice.
- [3] Air Force Test Pilot School Edwards Afb Ca, “Volume II. Flying Qualities Phase, Chapter 16. Chapter 16: Reprint of NASA TN D-5153, The Use of Pilot Rating in the Evaluation of Aircraft Handling Qualities.” 1969.
- [4] Banks, M. S., Kim, J., and Shibata, T., “Insight into vergence/accommodation mismatch,” *Head- and Helmet-Mounted Displays XVIII: Design and Applications*, 2013.
- [5] Carreira, P., Castelo, T., Gomes, C. C., Ferreira, A., Ribeiro, C., and Costa, A. A., “Virtual reality as integration environments for facilities management,” *Engineering, Construction and Architectural Management*, vol. 25, 2018, pp. 90–112.
- [6] Carson, E., “NASA shows the world its 20-year virtual reality experiment to train astronauts: The inside story,” *TechRepublic* Available: <https://www.techrepublic.com/article/nasa-shows-the-world-its-20-year-vr-experiment-to-train-astronauts/>.
- [7] Casini, A. E., Maggiore, P., Viola, N., Basso, V., Ferrino, M., Hoffman, J. A., and Cowley, A., “Analysis of a Moon outpost for Mars enabling technologies through a Virtual Reality environment,” *Acta Astronautica*, vol. 143, 2018, pp. 353–361.

- [8] “Ford Uses Virtual Reality to Reduce Costs,” *Ford Uses Virtual Reality to Reduce Costs / Automation World* Available:
<https://www.automationworld.com/article/technologies/simulation-modeling/ford-uses-virtual-reality-reduce-costs>.
- [9] Georgiev, G., “Confidence Intervals & P-values for Percent Change / Relative Difference,” *Analytics* Available: <http://blog.analytics-toolkit.com/2018/confidence-intervals-p-values-percent-change-relative-difference/>.
- [10] Hale, J., “Applied virtual reality in aerospace,” *Wescon/96*.
- [11] Hogan, Robert, Coen, Mauricio, McHenry, Neil, Morrell, Benjamin, and Chamitoff, Gregory, “SpaceCRAFT: A Virtual Reality Sandbox Environment,” 2018 International Astronautical Congress, Adelaide, Australia, 25-29 September 2017.
- [12] Harper, P., and Cooper, G. E., “Handling qualities and pilot evaluation,” *Journal of Guidance, Control, and Dynamics*, vol. 9, 1986, pp. 515–529.
- [13] Hill, S. G., Iavecchia, H. P., Byers, J. C., Bittner, A. C., Zaklade, A. L., and Christ, R. E., “Comparison of Four Subjective Workload Rating Scales,” *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 34, Aug. 1992, pp. 429–439.
- [14] Lhotka, C., Souchay, J., and Shamsavari, A., “Obliquity, precession rate, and nutation coefficients for a set of 100 asteroids,” *Astronomy & Astrophysics*, vol. 556, 2013.
- [15] Mansikka, H., Virtanen, K., and Harris, D., “Comparison of NASA-TLX scale, modified Cooper–Harper scale and mean inter-beat interval as measures of pilot mental workload during simulated flight tasks,” *Ergonomics*, vol. 62, 2019, pp. 246–254.

- [16] McHenry, Neil, Hogan, Robert, Abdou, Edmond, Coen, Mauricio, Morrell, Benjamin, Zochowski, Yan, and Chamitoff, Gregory, “Virtual Reality Multi-User Space System Mission Design and Simulation: Engaging the Public Through Open-Source Collaboration,” 2018 International Astronautical Congress, Adelaide, Australia, 25-29 September 2017.
- [17] “Measuring the Benefits and ROI Of VR - Improov, VR Teleconferencing For CAD,” *Improov* Available: <https://www.improovr.com/blog/measuring-the-benefits-and-roi-of-vr/>.
- [18] Menin, A., Torchelsen, R., and Nedel, L., “An Analysis of VR Technology Used in Immersive Simulations with a Serious Game Perspective,” *IEEE Computer Graphics and Applications*, vol. 38, 2018, pp. 57–73.
- [19] Nasa, “nasa/trick,” (2019), GitHub Available: <https://github.com/nasa/trick/wiki/Related-Projects>.
- [20] “NASA is Using Virtual Reality to Train Astronauts,” *Unimersiv* Available: <https://unimersiv.com/how-nasa-is-using-virtual-and-augmented-reality-to-train-astronauts-37/>.
- [21] Okeeffe, E., Campbell, A., Swords, D., F.laefer, D., and Mangina, E., “Oculus Rift Application for Training Drone Pilots,” *Proceedings of the 10th EAI International Conference on Simulation Tools and Techniques - SIMUTOOLS 17*, 2017.
- [22] Penn, J., and Lin, A., “The Trick Simulation Toolkit: A NASA/Opensource Framework for Running Time Based Physics Models,” AIAA Modeling and Simulation Technologies Conference, 2016.

[23] “Spotlight | National Inventors Hall of Fame,” *National Inventors Hall of Fame*

SutherlandIvan Comments Available:

<https://web.archive.org/web/20160219231939/http://invent.org/inductees/sutherland-ivan/>.

[24] Stone, R. J., Panfilov, P. B., and Shukshunov, V. E., “Evolution of aerospace simulation: From immersive Virtual Reality to serious games,” *Proceedings of 5th International Conference on Recent Advances in Space Technologies - RAST2011*, 2011.

[25] “The Distributed Space Exploration Simulation (DSES),” *NASA Available:*

<https://ntrs.nasa.gov/search.jsp?R=20070006475>.