Mixed-Reality for Unmanned Aerial Vehicle Operations in Near Earth Environments

A Thesis

Submitted to the Faculty

of

Drexel University

by

James T. Hing

in partial fulfillment of the

requirements for the degree

of

Doctor of Philosophy in Mechanical Engineering

March 2010

Dedications

To my mother who has always supported my dreams. In her eyes, I can never fail.

Acknowledgments

I would like to thank everyone who has played even the smallest role in helping me to complete my Doctoral studies. In this section I have noted a few key individuals out of that group. If I have left you out, please note that you are greatly appreciated.

First and foremost, I would like to express my most sincere gratitude and admiration to my advisor Dr. Paul Y. Oh, without whose competence, generosity, and vision, this dissertation would not have been possible. Dr. Oh was always ready to offer support and guidance in my Ph.D. research, but more than that, he offered his friendship. I would also like to thank the rest of my thesis advisory committee. Dr. Lau, Dr. Kwatny, Dr. Hsieh, and Dr. Izzetoglu provided me with invaluable advice and support to help focus my research since my proposal.

In addition to my committee members, the faculty and staff of the Department of Mechanical Engineering and Mechanics have been tremendously helpful throughout my academic career at Drexel University. In particular, Dr. Mun Choi, MEM's former Department Head, played a key role in keeping me at Drexel and in graduate school during a time when unexpected events made my academic future uncertain. His support and guidance during that period will always be remembered. Of course I can not forget Ms. Kathie Donahue who always lovingly gave me a hard time whenever I visited the main office needing something. Regardless of how busy she was, my requests would always get taken care of immediately.

Thank you to Hasan Ayaz and Justin Menda from the Drexel Optical Brain Imaging Lab for their knowledge and help with the studies involving fNIR. Their contributions have given great strength to my thesis.

I will always be grateful for the friendship and support from my colleagues in both the PRISM laboratory and the Drexel Autonomous Systems Laboratory (DASL). Every one of you made my many years in grad school worth the time just to have

been able to meet you. In particular Greg Tholey and Keith Sevcik I would like to thank for also playing the role of a mini advisor at the PRISM lab and DASL respectively when my actual advisor was unavailable. It was incredibly helpful to have someone of their intellectual caliber to bounce research ideas off of.

To my golfing friends, I owe you a thank you for pulling me out of the lab and forcing me to get sunlight and fresh air.

I have already thanked Dr. Lau as part of my committee but he will forever be thanked as the matchmaker who introduced me to my fiance Lauren Ciccarelli in his Finite Element Course. I would like to thank Lauren for her unwaivering love and support. She was more than just supportive while pursuing my doctorate. She was the extra set of hands when I needed help building my experiment setups, she was the extra pair of eyes when reviewing my paper submissions, she was the extra brain power when I hit research roadblocks, and she was the extra strength I needed to push through moments of self doubt. Without her, I would still be in the lab trying to graduate.

Last but not least I would like to thank my family (which includes Lauren's family) for their love and for always reminding me of how proud they are. My potential is limitless with them in my corner.

Table of Contents

List of Tables

List of Figures

Abstract

Mixed-Reality for Unmanned Aerial Vehicle Operations in Near Earth Environments

James T. Hing

Advisor: Paul Y. Oh, Ph.D.

Future applications will bring unmanned aerial vehicles (UAVs) to near Earth environments such as urban areas, causing a change in the way UAVs are currently operated. Of concern is that UAV accidents still occur at a much higher rate than the accident rate for commercial airliners. A number of these accidents can be attributed to a UAV pilot's low situation awareness (SA) due to the limitations of UAV operating interfaces. The main limitation is the physical separation between the vehicle and the pilot. This eliminates any motion and exteroceptive sensory feedback to the pilot. These limitation on top of a small field of view from the onboard camera results in low SA, making near Earth operations difficult and dangerous. Autonomy has been proposed as a solution for near Earth tasks but state of the art artificial intelligence still requires very structured and well defined goals to allow safe autonomous operations. Therefore, there is a need to better train pilots to operate UAVs in near Earth environments and to augment their performance for increased safety and minimization of accidents.

In this work, simulation software, motion platform technology, and UAV sensor suites were integrated to produce mixed-reality systems that address current limitations of UAV piloting interfaces. The mixed reality definition is extended in this work to encompass not only the visual aspects but to also include a motion aspect. A training and evaluation system for UAV operations in near Earth environments was developed. Modifications were made to flight simulator software to recreate current UAV operating modalities (internal and external). The training and evaluation sys-

tem has been combined with Drexel's Sensor Integrated Systems Test Rig (SISTR) to allow simulated missions while incorporating real world environmental effects and UAV sensor hardware.

To address the lack of motion feedback to a UAV pilot, a system was developed that integrates a motion simulator into UAV operations. The system is designed such that during flight, the angular rate of a UAV is captured by an onboard inertial measurement unit (IMU) and is relayed to a pilot controlling the vehicle from inside the motion simulator.

Efforts to further increase pilot SA led to the development of a mixed reality chase view piloting interface. Chase view is similar to a view of being towed behind the aircraft. It combines real world onboard camera images with a virtual representation of the vehicle and the surrounding operating environment. A series of UAV piloting experiments were performed using the training and evaluation systems described earlier. Subjects' behavioral performance while using the onboard camera view and the mixed reality chase view interface during missions was analyzed. Subjects' cognitive workload during missions was also assessed using subjective measures such as NASA task load index and non-subjective brain activity measurements using a functional Infrared Spectroscopy (fNIR) system. Behavioral analysis showed that the chase view interface improved pilot performance in near Earth flights and increased their situational awareness. fNIR analysis showed that a subjects cognitive workload was significantly less while using the chase view interface. Real world flight tests were conducted in a near Earth environment with buildings and obstacles to evaluate the chase view interface with real world data. The interface performed very well with real world, real time data in close range scenarios.

The mixed reality approaches presented follow studies on human factors performance and cognitive loading. The resulting designs serve as test beds for studying

UAV pilot performance, creating training programs, and developing tools to augment UAV operations and minimize UAV accidents during operations in near Earth environments.

1. Introduction

1.1 Motivation

Teleoperation in its most basic sense is the operation of a system while separated by some distance. The idea of teleoperation has been around ever since humans have had the desire to extend direct control of objects beyond the physical bounds of their own bodies. The physical separation can be a necessity due to operations within a hazardous environment such as a nuclear facility when handling toxic materials. The separation can also be necessary for scaling reasons such as a surgeon who uses robotic arms to scale down their hand motions for dexterous laproscopic surgical operations. In recent years, the teleoperation of unmanned aerial vehicles (UAVs) has become increasingly common as they are consistently proving themselves to be a tremendous force multiplier for the military [16]. These vehicles are well suited for military missions because the pilots controlling the vehicles are safely secured in mobile ground stations, well away from potential enemy fire.

1.1.1 New Paradigm

UAVs have been around for a very long time; nearly as long as the history of manned aircraft itself. The first successful powered unmanned flight was conducted by Samuel P. Langley's Number 5 in 1896 [17]. Mission capable UAVs began appearing during World War II. In the 1940's, the Germans developed an unmanned aircraft called the V-1 "Buzzbomb" that was capable of flying far distances to desired targets [16]. Since then there have been dramatic improvements in the capabilities and reliability of unmanned aircraft. Systems like the Predator (see Figure 1.1 left) and Reaper have a incredible success rate conducting medium to high altitude

Figure 1.1: Left: The MQ-1 Predator. Right: The RQ-11 Pathfinder Raven. Reprinted from [1]

long endurance missions that include surveillance, targeting, and strike missions [1]. However, UAVs are evolving and quickly expanding their role beyond the traditional higher altitude surveillance. Due to advances in technology, small, lightweight UAVs, such as the Raven (Figure 1.1 right) and Wasp, are now capable of carrying complete avionics packages and camera systems, giving them the capability to fly in environments much too cluttered for the proven large scale systems such as the Predator [18].

The successful record of the UAVs in the military has fueled a strong desire to adapt these vehicles for civilian applications. There are a myriad of potential applications that could benefit from UAV technology [19]. Most of these applications fall into the following categories: search and rescue, surveillance, transportation, communications, payload delivery and remote sensing [20]. These applications will extend UAVs beyond high altitude and passive interaction(surveillance) with the environment to lower altitudes and active interaction with objects in the environment (autonomous air cargo transport and medical evacuation (med-evac) missions). This new shift in the role of UAVs will require a change in the way that they are currently operated.

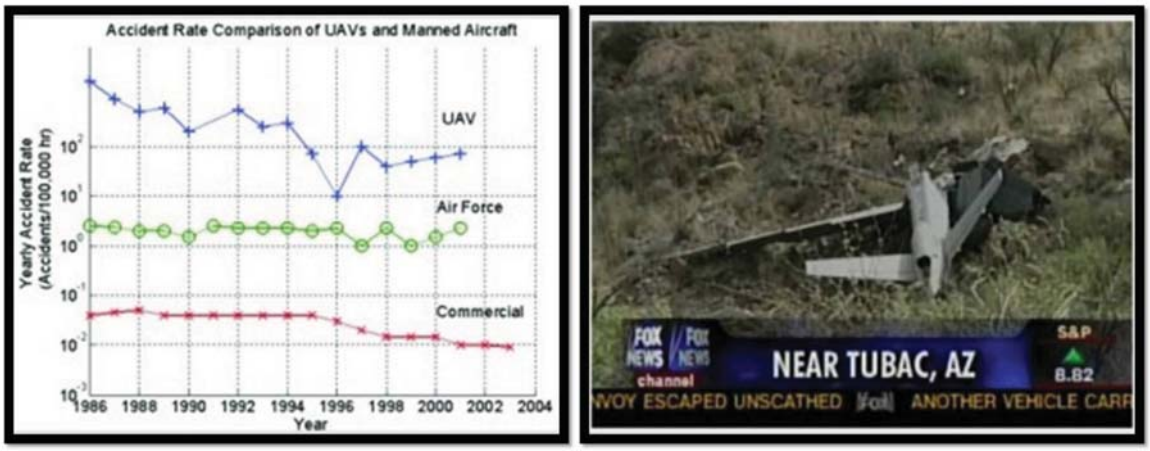

Figure 1.2: Left: Accident rate of UAVs compared with manned aircraft accident rates [2]. Right: News media capture of a Predator accident in Arizona.

1.1.2 Issues in UAV Operations

As the appeal and proliferation of UAVs increase, urgent and important issues arise. First, there are pressures to open the national airspace (NAS) to UAVs. The Federal Aviation Association (FAA), who regulates every aspect of air travel in the United States, is being pressured by the U.S. Department of Commerce to quickly establish standards so that UAVs and commercial airliners can share the national airspace (NAS). Second, a 2004 report states that the commercial market for UAVs will exceed the defense market by 2015 [21]. Civilian applications for UAVs will introduce these vehicles into cluttered near Earth environments [19]. These are low flying areas typically cluttered with obstacles such as buildings, trees and power lines. More importantly, these areas are also populated with civilians. Third, as UAV demand grows, so will the need for well-trained operators. Currently, there are only two major UAV schools in the United States, both of which are restricted to military personnel. Fourth, while no fatal accidents have occurred, the number of mishaps has been steadily rising and is still much more common than that of manned aircraft, as

represented by the chart seen in Figure 1.2 left [2]. A media capture of a published accident is shown in Figure 1.2 right. As such, the urgent and important issue is to design systems and protocols that can prevent UAV accidents, better train UAV operators, and augment pilot performance.

1.1.3 Requirements for a UAV Pilot Interface

For this work, the focus is on teleoperation interfaces specifically for aerial vehicles. Regardless of the application, all teleoperation systems have the following general components [22]:

- A **local site** where the human operator has some type of interface and input device used to monitor and control the remote system. The monitoring interface could be a display showing sensor data such as a camera view or an area cleared for direct line of site of the remote system. The input could be a joystick, mouse, keyboard, touch screen, manipulator arms, or any other input type devices.

- A **remote site** containing the teleoperated system that interacts with the environment. The teleoperated system contains sensors and other control elements to facilitate operators commands.

- A system for **transmitting information** between the local and the remote sites.

The goal of the interface is to provide tools to the human operator for decision making, generating commands and perception of the operating environment. This perception is known as situational awareness (SA). The accepted definition of SA comes from Endsley et al. [23] and it is broken down into three levels. Level 1 SA is the perception of the elements in the operating environment within a volume of time and space. Level 2 SA is the comprehension of their meaning and Level 3 SA is the

projection of their status in the near future. Certainly, most interfaces are designed to try and maximize operator situational awareness (SA) while minimizing the cognitive workload. A number of studies have evaluated the situational awareness and cognitive workload requirements for teleoperation operators [24, 25, 26]. Also of importance is the ultimate goal of achieving telepresence. Telepresence is the perception of being present at the remote site with no notice of the physical separation between the operator's self and the remote vehicle.

1.1.4 Challenges

There are many challenges to face when trying to incorporate high situational awareness and telepresence for a UAV pilot. For one, the pilot is not present in the remote vehicle and therefore has no direct sensory contact (kinesthetic/vestibular, auditory, smell, etc.) with the remote environment. The operator's physical separation from the vehicle eliminates all motion feedback whereas manned aircraft pilots utilize this motion to assist with vehicle control. Manned aircraft pilots often fly by "feel", reacting to acceleration forces while maneuvering the aircraft. When pilots perceive these forces as being too high, they often ease off the controls to fly more smoothly. Losing this sense of "feel", the pilot may unknowingly make excessive maneuvers or fly into hazardous environment conditions [27]. Therefore, sensory information that is lacking for a UAV pilot, must somehow be compensated for by the interface.

The visual information relayed to the UAV pilot is usually of a degraded quality when compared to direct visualization of the environment. This has been shown to directly affect a pilot's performance [28]. The UAV pilot's field of view is restricted due to the limitations of the onboard camera. The limited field of view also causes difficulty in scanning the visual environment surrounding the vehicle and can lead to disorientation [28]. Color quality in the image can also be degraded which can hinder

tasks such as search and targeting. Different focal lengths of the cameras can cause distortion in the periphery of images and lower image resolution affecting the pilot's telepresence [29]. Data lag in the video images as well as control commands leads to increased task completion times and in some cases, uncontrolled operation [30].

Near Earth flight also produces many challenges. Obstacles are much more commonplace in these environments compared to the frequency of obstacles in the higher altitudes where Predator systems operate. While high altitude operations are mostly focused on stable flight and waypoint navigation, near Earth flight requires high agility to account for obstacle avoidance in three dimensions. Near Earth environments are very dynamic which lead to a high potential of rapidly changing mission plans. Facing these challenges, researchers have developed a wide variety of vehicle teleoperation interfaces that are described in detail in Chapter 2.

1.2 Thesis Contributions

The work conducted for this thesis is motivated by the desire to improve UAV operations in near Earth environments. It contains hardware and software integration and design in addition to human performance analysis. The contributions can be broken down into the following:

- Development of an indoor virtual UAV test facility that integrates a large robotic 6DOF gantry and flight simulation software for UAV pilot training (Chapter 3). The system allows for safe training and evaluation of UAV pilots in near Earth environments while using actual UAV sensor hardware. Inside the gantry workspace, a scaled mock real world environment was built representative of a near Earth environment. Subjects sit at a console and input commands to a simulated UAV. The dynamics of the UAV are calculated by a flight simulation package and used to drive the end effector of the gantry

through the environment with the dynamics of the simulated UAV. The end effector holds a servo unit that houses the UAV sensors. The resulting information is relayed back to a graphical interface. Subjects flew simulated missions through the gantry environment and performance data was measured. Studies found performance increase with continued use of the indoor virtual UAV test facility.

- The novel application of UAV avionics with motion platforms to allow for the study of the "shared fate" effect on UAV pilot control and decision making (Chapter 4). The major contribution is the development of the multiple subsystems necessary for implementation.

- A Novel mixed-reality UAV piloting interface that improves situational awareness for UAV operations in near Earth environments (Chapter 5). The interface uses real world, real time avionics information to stabilize the onboard camera video feed. The position data is also used to enhance the limited field of view from the onboard camera with a virtual representation of the flight environment. Also integrated into the display is a virtual representation of the size and pose of the vehicle within the flight environment. Contributions also include human performance studies and cognitive workload assessment (Chapter 6 and Chapter 7). Results from flights using the indoor virtual UAV test facility showed that the mixed reality interface improved operator piloting performance in near Earth environments and decreased operator cognitive workload.

- Design and development of a field ready system for the implementation of the mixed reality interface in close range field operations (Chapter 8). A commercial Raptor 90 helicopter was modified and retro fitted with a wireless camera, wireless transmitter, and a inertial navigation system. A ground station gener-

ated the mixed reality interface in real time using wireless transmissions from the onboard avionics. Real world tests showed good performance of the mixed reality interface during field missions and the potential to enhance awareness during periods of degraded onboard camera video feed.

1.3 Thesis Organization

The rest of this work is organized in the following manner:

Human Factors of Unmanned Aerial Vehicles. Chapter 2 reviews the relevant human factors research conducted toward the development of teleoperation interfaces for UAVs which include Direct and Bilateral, Multisensor/Multimodal, Virtual Reality, and Mixed Reality/Augmented Reality interfaces. An analysis of the research literature is presented.

Methods for the Evaluation and Training. Chapter 3 demonstrates how a commercial flight simulation package is modified to serve as a UAV pilot training system. Also presented, is the integration of the software with Drexel's Systems Integrated Sensor Test Rig (SISTR) to create an indoor training and evaluation system that uses real world sensor hardware.

Motion Platform Integrated UAV Pilot Interface. Chapter 4 details the hardware and the integration methods for the design of a motion platform to UAV interface that addresses the issue of lack of motion feedback to a UAV pilot. Supporting literature for the benefit of motion feedback is presented.

Mixed Reality Interface for UAV Operations in Near Earth Environments. Chapter 5 presents two methods for generating the mixed reality chase viewpoint and the software and hardware integration methods for developing the pilot interface.

Exploratory and Development Stages Using the Systems Integrated Sensor Test Rig (SISTR). Chapter 6 details the experimental setup and procedure for the Exploratory

and Development Stages to assess the benefits of the chase view interface. Indoor flight trials using SISTR are presented. Results of these studies lead to the formulation of the main hypotheses for the Human Performance and Assessment studies.

Human Performance and Assessment Stage. Chapter 7 presents the human performance studies to test the formulated hypotheses. These studies were part of a collaborative effort with the Drexel Optical Brain Imaging Lab to integrate Functional Near-Infrared Spectroscopy (fNIR) into the assessment of pilot performance and cognitive workload while using various interface designs and flight environments. Statistical analysis of the behavioral and cognitive workload results from flight tests with the traditional onboard camera view and with the generated chase view are presented in detail. This chapter also discusses further testing with SISTR to evaluate pilot performance using rotorcraft. Also investigated is the effect of UAV position data accuracy on pilot performance using the chase view interface. Initial results and discussions of flight trials are presented.

Validation of the Chase View Interface in Near Earth Environments. Chapter 8 details the integration of software and hardware into a system capable of real world tests. Presented are results of the interface performance during flights in a near Earth environment.

Conclusion, Future Work and Enabling Technologies. Chapter 9 summarizes the work presented in this thesis. Further more, this chapter also discusses future development and enabling technologies that will help the mixed reality interface play integral role in the safe operations of UAVs in near Earth environments.

2. Human Factors of Unmanned Aerial Vehicles

2.1 UAV Accidents

In January 2006, a Los Angeles County Sheriff lost control of a UAV which subsequently nose-dived into a neighborhood. In April of the same year, a UAV crashed into the ground within several hundred feet of homes in Arizona. This was a civilian-version of the Predator B drone used by the U.S. Customs and Border Protection Agency. The operator had shut off its engine by mistake. Also in April, a Coast Guard Eagle Eye tilt-rotor UAV crashed in Texas after an unidentified radio signal triggered the self-destruct mechanism. A number of Predator systems have also been lost because of the difficulty in landing due to the narrow camera view.

Accidents are not isolated to directly piloted vehicles. Autonomous systems have also experienced a number of mishaps. In March 1999, operators at Nellis Test Range in Nevada, inadvertently sent a self terminate signal while Global Hawk was aloft and under the control of officials at Edwards Air Force Base in California. In December 1999, an operator for the fully autonomous Global Hawk incorrectly programmed the UAV to taxi at 155 nautical miles per hour. On November 4, 2000, the fully autonomous Fire Scout crashed due to a malfunctioning altitude sensor. The false reading indicated that the Fire Scout was at an altitude of 2 feet above the ground when, in fact, it was hovering at an altitude of 500 feet. The guidance and control system interpreted the incorrect altitude and shut down the engine as designed [31].

2.2 Human Factors Research

Certainly the high rate of UAV accidents raises much concern when discussing the integration of UAVs into the National Air Space (NAS). The benefit of UAV technol-

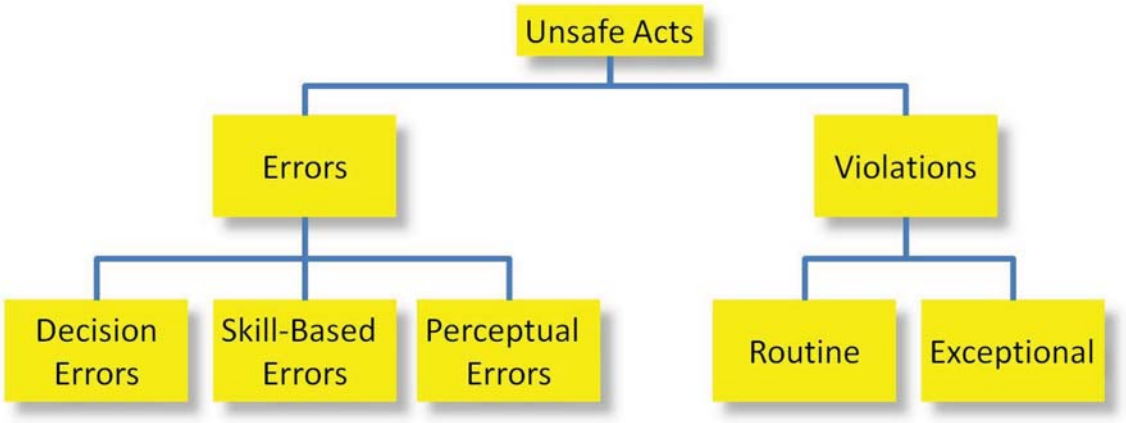

Figure 2.1: Human Factors Analysis and Classification System (HFACS) adapted from [3].

ogy has stimulated the development of many civilian applications. However, many of these applications will bring UAVs into areas that are high risk and have a higher probability of casualties due to a UAV mishap. Historically the main contributing factors of UAV accidents has been associated with electromechanical failures [32]. However, as the technology has matured and materials for various UAV parts have improved, human error is increasingly becoming a main factor in the cause of UAV mishaps [33]. Seagle et al. [34] studied 107 UAV accidents that occurred over the span of seven years and found that 43 percent were attributed to human error.

The army classifies accidents into 3 causal categories: human, material, and environmental [35]. Environmental causal factors are accidents associated with weather conditions, illumination, and noise. Material factors are events such as equipment failure. Human causal factors are accidents associated with human error. Human error can be further broken down into: Unsafe Acts, Preconditions for Unsafe Acts, Unsafe Supervision, and Organizational Influences. As seen in Figure 2.1, Unsafe Acts is expanded into Errors and Violations. Violations are errors corresponding to rules and regulations. Errors is further expanded into decision errors, perceptual errors and

skill-based errors [3]. Skill based errors can be attributed to a lack in training for a specific condition/task resulting in poor execution such as over control of the aircraft. Decision and perceptual errors are caused somewhat by a lapse in situational awareness where this lapse can result in inappropriate maneuvers, spatial disorientation, and poor decisions. To address these issues, human factors research must continue to investigate the causes of human error and produce valuable research leading toward the development of improved interfaces and procedures for UAV operations.

The high contrast in numbers between manned aircraft and unmanned aircraft accidents begs the question, "Why not apply the work developed to make manned aircraft safer to UAVs?" The answer to that question is difficult. For one, many of the smaller UAVs are not designed with the number of redundant safety systems that are currently onboard manned aircraft. Payload capacity at a smaller size is dramatically reduced, eliminating the ability to add on multiple redundant systems. Research findings from human factor research of manned aircraft has not been ignored completely. A lot of the work on the initial development of the flight controls and heads up displays used for systems like the Predator were designed based on human factor research for manned aircraft. Visual displays used for manned aircraft pilots are also being integrated into UAV displays such as synthetic vision [7]. There are also current efforts to replace the Predator HUD with a new design based on fighter aircraft HUD [31]. However, human factor research of UAVs presents challenges that are very different from manned aircraft. The main challenge, also being the main benefit of UAVs, is that the operator is not on board the operated aircraft. In addition to the issues stated in Chapter 1, the other challenges come from the myriad of ways that UAVs are operated. This stems from the large diversity of specialized missions that specific UAVs are designed for [36]. Because of this diversity, human factors research of UAVs spans a wide array of works. In general, these works can be

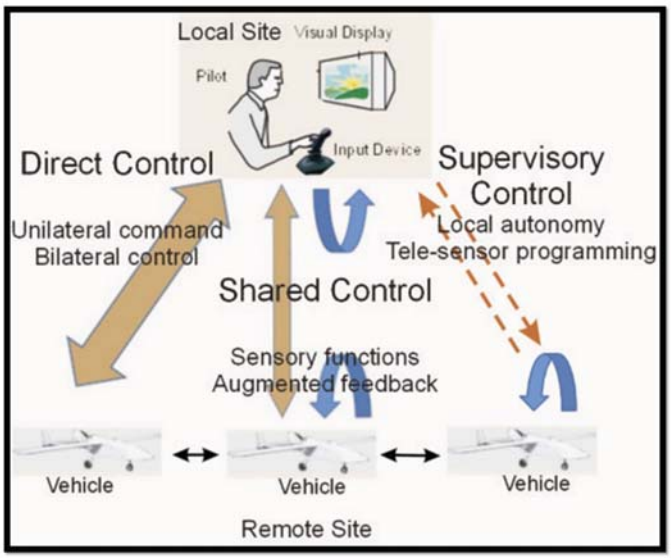

Figure 2.2: Teleoperation control schemes adapted from [4]

broken down into subsections dealing with automation, the human-machine interface, air traffic management, and crew operations. The bulk of this thesis focuses primarily on the human-machine interface for UAV pilots, so this topic is addressed in much greater detail in the following section.

2.2.1 Teleoperation Interfaces

Aerial robotic systems cover a very wide range of mission capabilities, operator requirements and autonomy. Because of this, there many types of interfaces developed for the multitude of systems. The control architecture of teleoperated vehicles can be organized into three categories, also illustrated in Figure 2.2 [4]:

- **Direct control** is probably the most common method for teleoperation as the vehicle motion is directly controlled by the operator using a joystick and monitoring the video feed from the onboard camera. There is no autonomy or intelligence in the system. This is appropriate for use when real-time human

decision or control is required [37]. However this technique does require very little delay in the data communication.

- **Shared control** is when there is some autonomy in the system or user feedback is augmented from virtual reality or other automatic aids.

- **Supervisory control** is when the supervisor (operator) gives high level directives to the robot and receives status information back [38]. This type of control requires the system to be autonomous and able to complete assigned tasks safely on its own. Systems under supervisory control are well suited for applications involving low bandwidth and high delay in data communications. A very successful application of supervisory control would be the Mars rover explorations [39].

The current state of the art UAVs are designed and operated to successfully complete tasks that commonly take place in higher altitude areas with very few obstacles to navigate around [1]. During a majority of these mission, most UAVs are operated under some level of supervisory control. These systems are not without their faults. In fully autonomous systems like the Global Hawk, Tvaryanas et al. [40] showed that because of the high level of automation, operators began to fall "out-of-the-loop" which lowered their situational awareness and increased their reaction time to system faults. Current autonomous systems are also not well suited for operations in cluttered environments. These require fast and accurate obstacle avoidance algorithms, fast object recognition, and quick adaptation to changing conditions. Few groups have successfully demonstrated autonomous low flight among obstacles but the vehicles still required predefined end goal locations [41, 42]. For potential civilian scenarios such as the monitoring of a car chase, these goal locations may not be defined prior to flight. In the event of a need to diverge from the predefined path, a

Figure 2.3: Left: Internal Pilot ground station for the Predator. Reprinted from http://spyflight.co.uk/Predator.htm. Right: An external pilot controlling a UAV during landing. Reprinted from [1].

human operator performance would be superior to an autonomous vehicle in obstacle avoidance and path finding. This scenario and many others will require critical and impromptu decisions that are beyond the current limits of state of the art artificial intelligence. For this reason, and others, this work focuses on improving the direct and shared control modalities of the teleoperation of UAVs. These control schemes keep a human in direct control of the flight of the vehicle. This allows for improved operations through the benefit of a human's ability to solve problems, ability to make rational decisions based on partial or incomplete information, and the experience and skills of the pilot. Teleoperation interfaces used in direct and shared control can be organized into four categories: Direct and Bilateral, Multisensor/multimodal, Virtual Reality, and Augmented/Mixed Reality.

Direct and Bilateral

While most of the current military UAVs have autonomous modes such as GPS waypoint navigation, there are still phases during operation where a pilot is in control of the vehicle using a direct teleoperation interface, such as during take off and

landing. Predator systems are a good example of this type of interface. In a direct control interface, the vehicle moves in direct relation to the input from the operator. The input device could be a joystick, or a replicated cockpit setup. Pilots of UAV systems such as the Predator, operate from ground stations that contain static pilot and payload operator consoles as seen in Figure 2.3 left. A pilot operating from this kind of station is known as an Internal Pilot(IP). The internal pilot directly controls the aircraft with a joystick, rudder pedals and views the remote environment through a monitor displaying images from an onboard camera. Alternatively to the IP, some UAVs such as the Mako from NAVMAR Applied Sciences, are flown during take off and landing stages using an External Pilot (EP). The EP controls the aircraft using a radio controller and views the vehicle through a line of site as seen in Figure 2.3 right, very similar to radio controlled (RC) model plane piloting.

Direct control interfaces are very susceptible to factors that degrade pilot performance. The limited field of view, delayed control response, and lack of sensory cues from the aircraft all lead to a low situational awareness for the pilot [43]. EP performance suffers from line of sight occlusion due to obstacles, control mapping difficulties, and a limited operational distance.

To address some of these issues, researchers have tried bilateral interfaces. In a bilateral interface, the vehicle also operates as a sensor and the operator input device also acts as a display. Ruff et al. [44] found that adding haptic feedback via the control stick improved pilot awareness to the onset of turbulence. Lam et al. [45] relayed force feedback to the control stick based on the location of the aircraft in relation to artificial force fields surrounding obstacles. This was shown to help decrease the number of collisions during flight especially during degraded visuals.

No prior work outside of the author's has been conducted on a bilateral interface to address the issue of lack of kinesthetic feedback to the UAV pilot. However, there

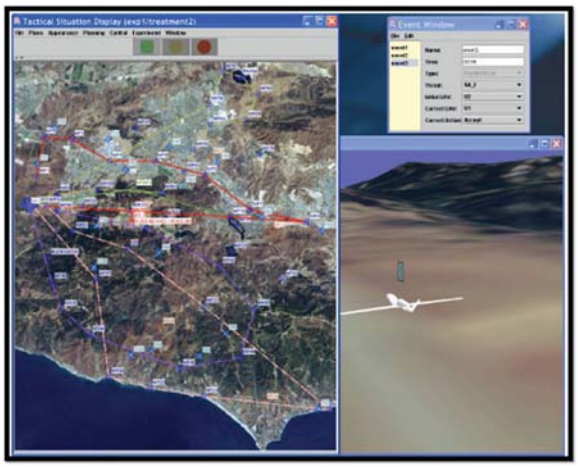

Figure 2.4: The multimodal immersive intelligent interface for remote operations (MIIRO). Reprinted from [5].

has been some work in the area of ground vehicles. Feng et al. [46] developed a motion platform interface to relay the motions of a construction tele-robot system to the operator. They hypothesized that for true telepresence when operating a construction robot, motion feedback was necessary.

While addressing some issues of decreased situational awareness for UAV pilots, many of these direct and bilateral interfaces do not address a number of the other issues such as data lag and limited field of view.

Multisensor/Multimodal

Multisensor interfaces combine data streams from multiple sensors to present an integrated view to the operator. Multimodal interfaces are designed to allow for changing control modes and displays based on context specific actions [37]. Most of the military ground stations in use today use these types of displays [1]. Tso et al. [5] developed a Multi-Modal Immersive Intelligent Interface for Remote Operation (MIIIRO) that is currently being used as a human factors test bed as seen in Figure 2.4. The system allows operators to control the UAV in manual, autonomous and

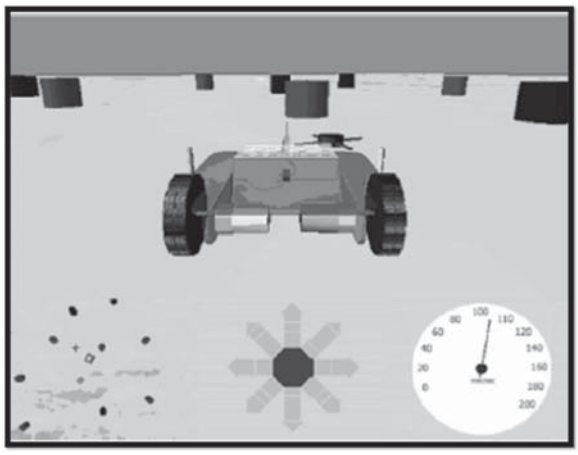

Figure 2.5: A virtual reality display for telerover navigation. Reprinted from [6].

shared control modes. The input from the UAV pilot comes from a joystick, motion tracker or voice commands. The display to the pilot includes mission plan view, virtual 3D view of the operation environment and instrumentation interfaces. While the increased amount of data has been shown to improve situational awareness, it comes at a cost of increased cognitive workload. The visual scanning between the different display windows can cause operators to rely and focus attention on only one part of the display. This is known as cognitive tunneling [47].

Virtual Reality

For virtual reality (VR) displays, the operator interacts with a virtual representation of the vehicle inside of a virtual representation of the remote environment as seen in Figure 2.5. In some cases, the remote vehicle is under direct control and follows the commands of the operator controlling the virtual vehicle. Otherwise, the remote vehicles are under supervisory control where the virtual environment and virtual robot are used as a high level task planner with some level of automation on the remote vehicle side. An added benefit of virtual reality is that the operator is no

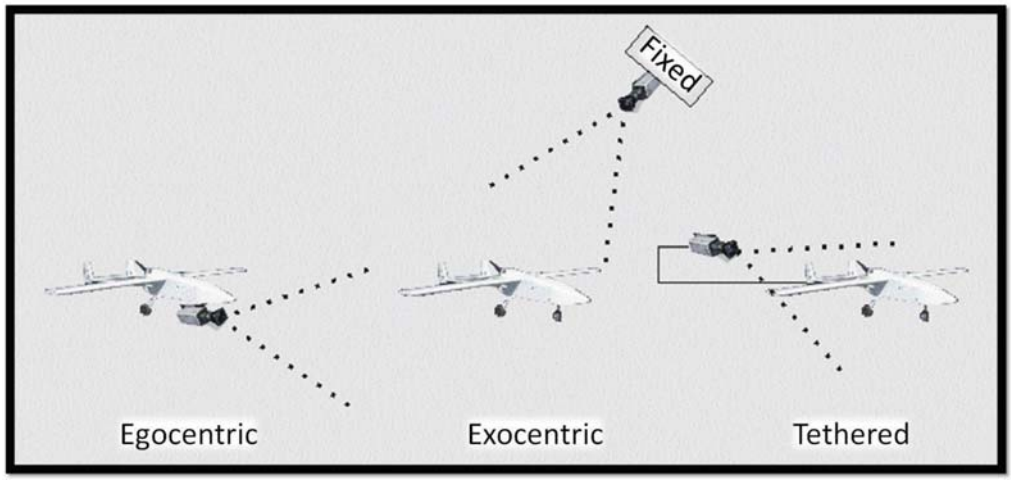

Figure 2.6: Egocentric, Exocentric and Tethered viewpoints for Teleoperation.

longer restricted to a standard viewpoint from the onboard camera. There are three possible viewpoints an operator can use during teleoperation:

- An **Egocentric View** in the teleoperated vehicle sense is the view from the onboard camera attached to the remote vehicle. It is also known as a first person viewpoint. For a forward facing camera, operator input always corresponds to the direction in which the vehicle is moving. Forward moves the robot forward with respect to the camera view, right moves the robot right with respect to the camera view, etc. Studies have shown that egocentric view is beneficial for local guidance which requires a strong understanding of the immediate surroundings of the vehicle [48]. However perception and visuomotor performance with this viewpoint does degrade as the field of view of the camera decreases [49].

- Using an **Exocentric View**, the operator views the robot and the environment from a fixed "bird's eye" view position. This has been shown to improve the global awareness of the operator which include tasks such as planning and problem solving [50]. Certainly with the much larger view of the environment,

understanding of the position and orientation of the vehicle with respect to its surroundings increase. However, performance in the control of the vehicle degrades due to control mapping issues. For a north up map view, if the operator is facing the display and the remote vehicle is facing north on the map, pushing forward will move the robot north on the display. However, if the robot is facing east, pushing forward on the remote will make the robot move east on the display which requires a mental rotation of the control mapping [51].

- A **Tethered view** is also known as a third-person view. This view is an external view from the vehicle but the perspective of the environment changes with the changing orientation and position of the vehicle. Salamin et al. [52] showed that this tethered view improved navigation through an environment when controlling a human avatar. Wang [51] presented extensive studies of moving a virtual object using multiple "styles" of tethered views with various distances and dynamic properties of the tether itself. The object was modeled as a point mass in the shape of an aircraft. It moved forward at a constant speed without any aerodynamic trajectory. The main goal was to keep the objects "wings" in the proper orientation with the floor of a long winding corridor. They showed that a tethered view produced better local guidance than an exocentric view but not as good as an egocentric view. Interviews of the subjects however showed that they preferred the use of a tethered view. The study however used a constant elevation of 30 degrees from the vehicle for the tethered view which may explain why egocentric view performed better. His future work recommends the study of different elevation angles for the tethered view point. His work also supports the results obtained by Wickens et al. [53] which showed that local guidance is better using egocentric displays but global awareness is better using increasing exocentric distances.

Vehicles under supervisory control can benefit from virtual reality interfaces as they are well suited for applications involving low bandwidth or high communication delays [37]. Virtual reality interfaces can also address the issue of telepresence. Systems such as CAVE use a wrap around display to facilitate immersion of the operator into the virtual environment [54].

Problems with virtual reality displays used for direct control can stem from degraded or delayed transmissions. In these cases, the virtual robot and virtual environment may not accurately represent what is actually occurring in real time at the remote site [39]. Kadavasal et al. address the issues of data communication delay during teleoperation by using a virtual reality interface and combining direct control and supervisory control [55]. During remote operation of a ground vehicle, the operator's commands are sent to a VR simulation that predicts the dynamic state of the ground vehicle. The simulation displays to the operator the dynamic movement of the vehicle in the modeled environment. While the operator is controlling the simulated vehicle, a series of waypoints are produced that the remote vehicle follows. If the vehicle encounters an obstacle that was not modeled in the virtual environment, it automatically breaks away from the commanded trajectory to avoid a collision and then returns to following the operator's commands. Through this type of control, the operator's performance does not suffer from communication delays because they receive instantaneous feedback of their commands from the virtual vehicle simulation. This method, however, still does not relay to the operator a real-time view of the operating environment and is technically still more supervisory control than direct control.

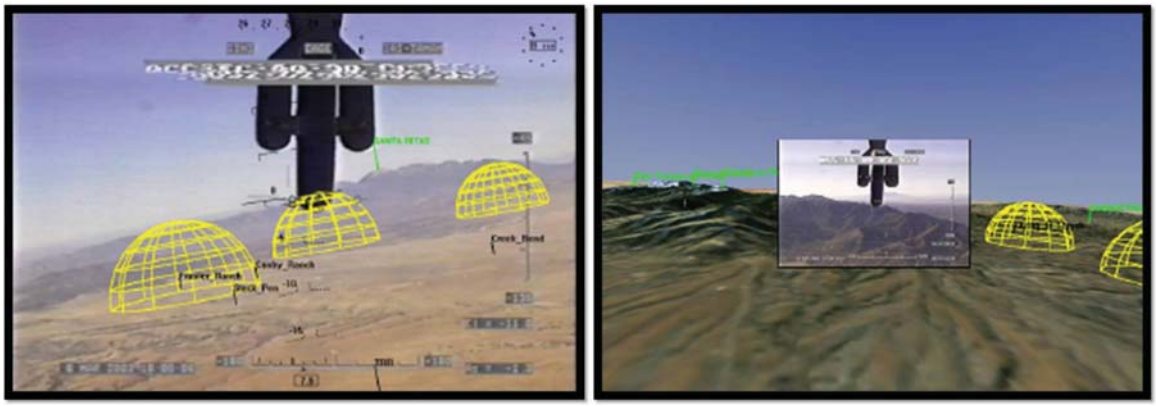

Figure 2.7: Left: Real world Onboard camera view with spatially reference computer-generated overlay symbology. Right: Picture-in-picture concept of real video imagery surrounded by sythetic-generated terrain imagery. Reprinted from [7].

Augmented/Mixed Reality

Augmented and mixed reality approaches have been recently developed to combine the advantages of both Virtual Reality and Multisensor displays. Mixed reality displays combine information from the real world and information from a virtual world together into a single integrated view of the environment. Augmented reality is essentially a subset of mixed reality in the sense that it involves the augmentation of a real world image with computer generated content. A commonly used Mixed Reality interface is Synthetic Vision, an example of which can be seen in Figure 2.7 left.

Synthetic vision, in recent years, has been studied and shown to improve situational awareness for remotely piloted vehicles [56]. Synthetic vision has a few key components. One display shows a far distance exocentric view of the UAV with a virtual representation of the terrain based on a database of elevation maps. This is mostly used to depict the planned trajectory from a 3D perspective for support in guidance and control. Another display shows the onboard camera video feed augmented with non-physical constraints such as threat volume depiction. More recently,

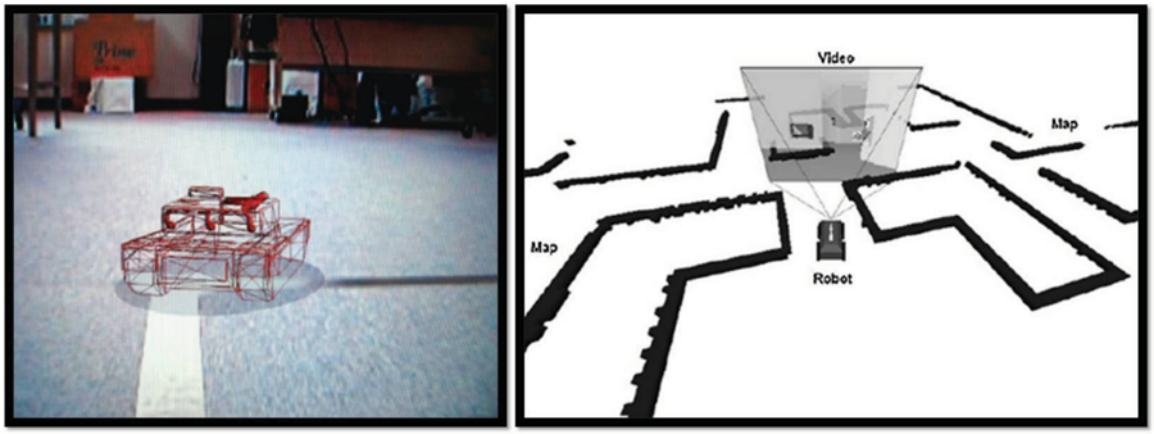

Figure 2.8: Left: Exocentric mixed-reality view using past onboard camera images. Reprinted from [8]. Right: 3D mixed-reality display with integrated onboard camera view. Reprinted from [9].

the field of view of the onboard camera feed has been enhanced with a virtual representation of the surrounding environment to compensate for sensor limitations such as limited field of view, range, and occlusion such as smoke or clouds. This is described as a "picture-in-picture" view by Draper et al. [57], an example of which can be seen in Figure 2.7 right. Synthetic vision has been used for higher altitude flight and requires prior knowledge of the terrain/elevation. It does not include obstacles other than the natural terrain data. Synthetic Vision displays have not previously been evaluated for near-Earth flight. The lack of integration of the 3D view of the vehicle with the onboard camera view requires the pilot to scan multiple displays causing a decrease in performance. Also, while the onboard camera view is augmented, a pilot can still struggle with the mental mapping of the environment. They may also struggle with vertigo due to the moving horizon.

A couple of research groups have investigated methods for viewing remotely operated ground vehicles from outside the vehicle; "Time Follower's Vision" by Sugimoto et al. [8] seen in Figure 2.8 left and tethered position by Nielsen et al. [9] seen in

Figure 2.8 right. Both methods produced a viewpoint that allowed an entire virtual visualization of the vehicle pose and real world images of the environment surrounding the vehicle itself. Both works presented studies showing that their methods improved remote operation of the vehicle in both speed of operation and accuracy of vehicle positioning. In the work produced by Sugimoto et al. [8] however, the surrounding environment is based on prior images from the vehicle camera so it is not suitable for use in a highly dynamic environment. It requires no roll motion from the camera image and still suffers from the limited field of view from the camera. Also, being purely a 2D image, it does not contain any 3D information about the surrounding environment. Nielsen et al. [9] generated a 2-D map of the environment as the vehicle drove around, using a laser range finder and simultaneous localization and mapping algorithms (SLAM). This map was relayed in a 3D perspective to the operator based on the tethered view from the vehicle. Integrated into the display was the onboard camera view which was adjusted and distorted to match the perspective of the created map. Their methods for obtaining this type of display is currently limited to indoor planar worlds. Direct adaptation of these methods for UAVs is not reasonable because UAVs can undergo large three dimensional translations and rotations in cluttered and urban environments. Also, obstacles can not be represented by infinitely high walls (often used in 2D ground vehicle maps) as UAVs can fly around, above, and in the case of overpasses, below obstacles. UAVs, especially those flown in urban environments, will be small so they can maneuver between obstacles with relative ease. The small size limits the payload capacity of the vehicle. Laser range sensors, like those used in [9], can be too heavy to add to a typical UAV sensor suite that already includes an inertial measurement unit (IMU), global positioning system (GPS) and an onboard camera.

Drury et al. [58] used simulated video data of a high altitude UAV flight and aug-

Figure 2.9: Left: "Wing-view" display for UAV control. Right: Mixed-Reality interface showing rotated onboard camera view and aircraft avatars of current and desired positions. Right: 3D mixed-reality display with integrated onboard camera view. Reprinted from [10].

mented it with pre-loaded map data (satellite imagery). The down-looking onboard camera view was rotated to match the preloaded terrain map and a silhouette of the UAV is displayed on the map showing its heading. Their results showed that the augmented image helped the observer's comprehension of the 3D spatial relationship between the UAV and points on the Earth. This study used simulation only and focused on observer tasks. It did not evaluate the effects of this type of display on the piloting performance of the UAV.

Quigley et al. [10] investigated the effects of displaying a simplified "wing-view" of the UAV to the operator via a hand held personal digital assistant (PDA) (Figure 2.9 left) that showed the roll and altitude of the aircraft. This display helped with the operator's understanding of the instantaneous relationship between the UAV and the ground. However, it does not relay enough information in the event that direct control of the vehicle is needed. Also presented by Quigley et al. [10] is a mixed-reality

interface that shows a transparent avatar of the remote aircraft ontop of an onboard camera view that has been rotated to level the horizon (Figure 2.9 right). Included in the display are two aircraft avatars of different colors. One color represents the desired commanded position of the aircraft and the other color represents the actual position of the aircraft. This type of display addresses all three levels of situation awareness for the pilot and simulation results showed that precision in orienting the vehicle and operator quickness in response to directed trajectory commands was high. However, this method only utilizes the visuals from the onboard camera so it suffers from the limitations stated earlier. The display also has not been tested when flying in near Earth environments and the study focused more on the reaction time of pilots to produce commanded positions rather than pilot overall flight performance. This interface was designed more for applications where the user has other pressing concerns as well as control of multi-agent teleoperation.

3. Methods for Evaluating and Training UAV Pilots for Near Earth Operations

The evaluation of pilot performance using various operating interfaces requires a system that allows for safe pilot training and evaluation. Field testing with actual UAVs can be dangerous and expensive, especially when evaluating and training beginning pilots. Also, to properly conduct field tests requires significant time and paper work to obtain a certificate of authorization from the Federal Aviation Administration (FAA) to fly in most airspaces. This is where the virtual world offers advantages. In the virtual world, we have full control of the conditions. It is certainly cheaper and less risky to operate virtually with the advantage of also being able to reconstruct accident scenarios and train pilots in those situations. There are a few commercial UAV simulators available and the numbers continue to grow as the use of UAV's become more popular. However, most of these simulators are developed to replicate the state of the art training and operations for current military type UAVs. Because this research focuses on UAV piloting in environments and scenarios not commonplace in current UAV operations, a new system needed to be developed.

3.1 Flight Simulator and UAV Model

Development of a new UAV training and evaluation system started with modifications to a commercially available flight simulation (sim) package. X-Plane from Laminar Research offers a low cost flight simulation program that uses blade element theory to quickly generate very accurate aerodynamic models. During calculations of the aircraft dynamics, X-Plane breaks the plane and the wings/stabilizers down into a number of small elements. It then calculates the velocity vector of those elements

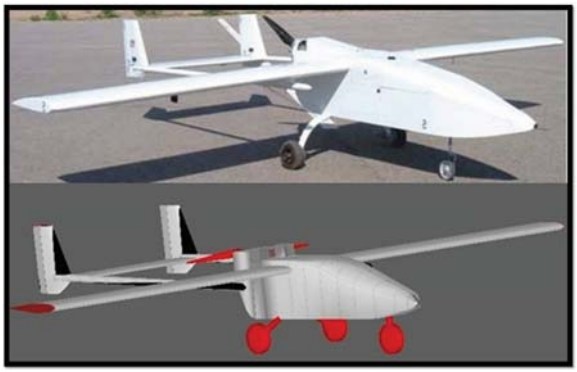

Figure 3.1: Top: MAKO UAV developed by NAVMAR Applied Sciences. Bottom: MAKO UAV recreated in X-Plane.

and determines coefficients such as lift and drag. Combining those values with the dynamic pressures surrounding the vehicle, it calculates and sums the forces on each of the elements. The summation of the forces is divided by the mass to obtain linear accelerations. The moments are divided by the moment of inertia to obtain angular accelerations. The accelerations are then integrated to obtain the velocities and again for positions. Although closed source, X-Plane is highly modifiable. It is also Federal Aviation Administration (FAA) certified. A very good description of X-Plane and how it works can be found from [59]. Users are able to control many aspects of the program and obtain a wide variety of data variables through user datagram protocol (UDP) connections and plug-ins.

A number of academics have utilized X-Plane for UAV research. Garcia et al. [60] built a small Maxi-Joker R/C rotor craft in X-Plane. They utilized the generated flight dynamics of the model of the rotor craft and used it to evaluate their autonomous flight controllers. Vidolov et al. [61] also used X-Plane to evaluate their fuzzy logic controller on a R/C helicopter model.

To start development of the training and evaluation system, a Mako UAV, seen in Figure 3.1, was modeled using the built in aircraft modeling program packaged

with X-Plane. The Mako is a military drone developed by Navmar Applied Sciences Corporation. It is 130 pounds, has a wingspan of 12.8 feet and is operated via an external pilot for takeoff and landings. It is under computer assisted autopilot during flight. For initial testing, this UAV platform was ideal as it could be validated by veteran Mako pilots in the author's local area. During the development of the training and evaluation system, a Mako pilot continually gave feedback on the fidelity of the system.

It is important to note that X-Plane is a flight simulation package originally developed to recreate manned aircraft pilot experience. Utilizing it as a tool for UAV operations takes some manipulation through user created plug-ins and external programs. Modifications began by developing view points and interfaces similar to interfaces in current UAV operations, specifically the internal and external pilot's viewpoints. These modifications were made using plugins written in C++. Plugins are small sections of code that can be run inside the main X-Plane program as opposed to external programs that run independently of the X-Plane program.

Internal Pilot View

The internal pilot operates the UAV from inside a ground station. The view from the wireless camera on board the aircraft is relayed to the internal pilot. The field of view is usually restricted due to the optics of the camera used. To recreate this viewpoint, the field of view needed to be restricted to match the real world camera specifications that is used on board the aircraft. The "blinders" were created using openGL graphics functions. This is shown in Figure 3.2. The position of the camera on board the aircraft is found in the global reference frame by applying a rotation and displacement to the camera frame as shown below. The camera is usually not positioned at the center of mass where the avionics are located so this displacement

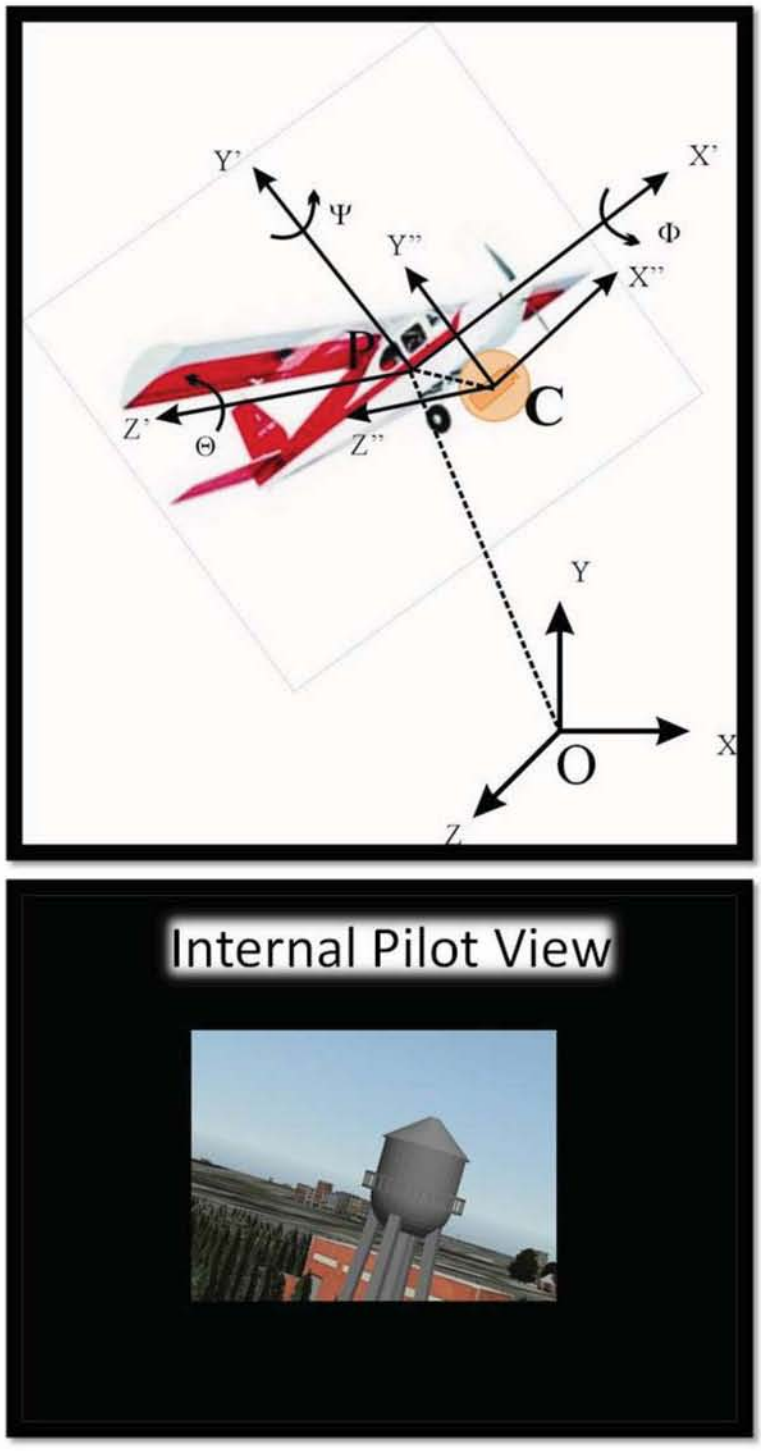

Figure 3.2: Top: Reference frames used for generating an internal view. Bottom: Example of a simulated internal pilot view.

must also be taken into account. The reference frames used to orient the camera are shown in Figure 3.2.

$$\begin{bmatrix} C_X \\ C_Y \\ C_Z \end{bmatrix} = \begin{bmatrix} R_{3x3} \end{bmatrix} \begin{bmatrix} C_{X'} \\ C_{Y'} \\ C_{Z'} \end{bmatrix} + \begin{bmatrix} P_X \\ P_Y \\ P_Z \end{bmatrix} \tag{3.1}$$

$$\begin{bmatrix} R_{3x3} \end{bmatrix} = R_\psi R_\theta R_\phi = \begin{bmatrix} c_\psi c_\theta & -c_\psi s_\theta c_\phi + s_\psi s_\phi & c_\psi s_\theta s_\phi + s_\psi c_\phi \\ s_\theta & c_\theta c_\phi & -c_\theta s_\phi \\ -s_\psi c_\theta & s_\psi s_\theta c_\phi + c_\psi s_\phi & -s_\psi s_\theta s_\phi + c_\psi c_\phi \end{bmatrix} \tag{3.2}$$

$$\begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} Roll_{aircraft} \\ Pitch_{aircraft} \\ Yaw_{aircraft} \end{bmatrix} \tag{3.3}$$

where "O" is the global reference frame for the flight simulator with X, Y and Z coordinates. "P" represents the aircraft reference frame with X', Y', and Z' coordinates. "C" represents the camera reference frame with X", Y", Z" coordinates. The variables "c_X" and "s_X" correspond to cosine(X) and sine(X) respectively.

External Pilot View

The external pilot as mentioned in Chapter 2, operates the aircraft using a line of sight with the aircraft from a static ground position. Usually the external pilot stands on or close to the runway next to the UAV during take off. The view, as seen in Figure 3.3, was created to maximize the ground peripheral vision of the UAV external pilot as this is used as a visual reference by the pilot to gather information on the speed and position of the aircraft. Another challenge was the nature of the computer screen itself. As the UAV traveled far away from the pilot, the vehicle

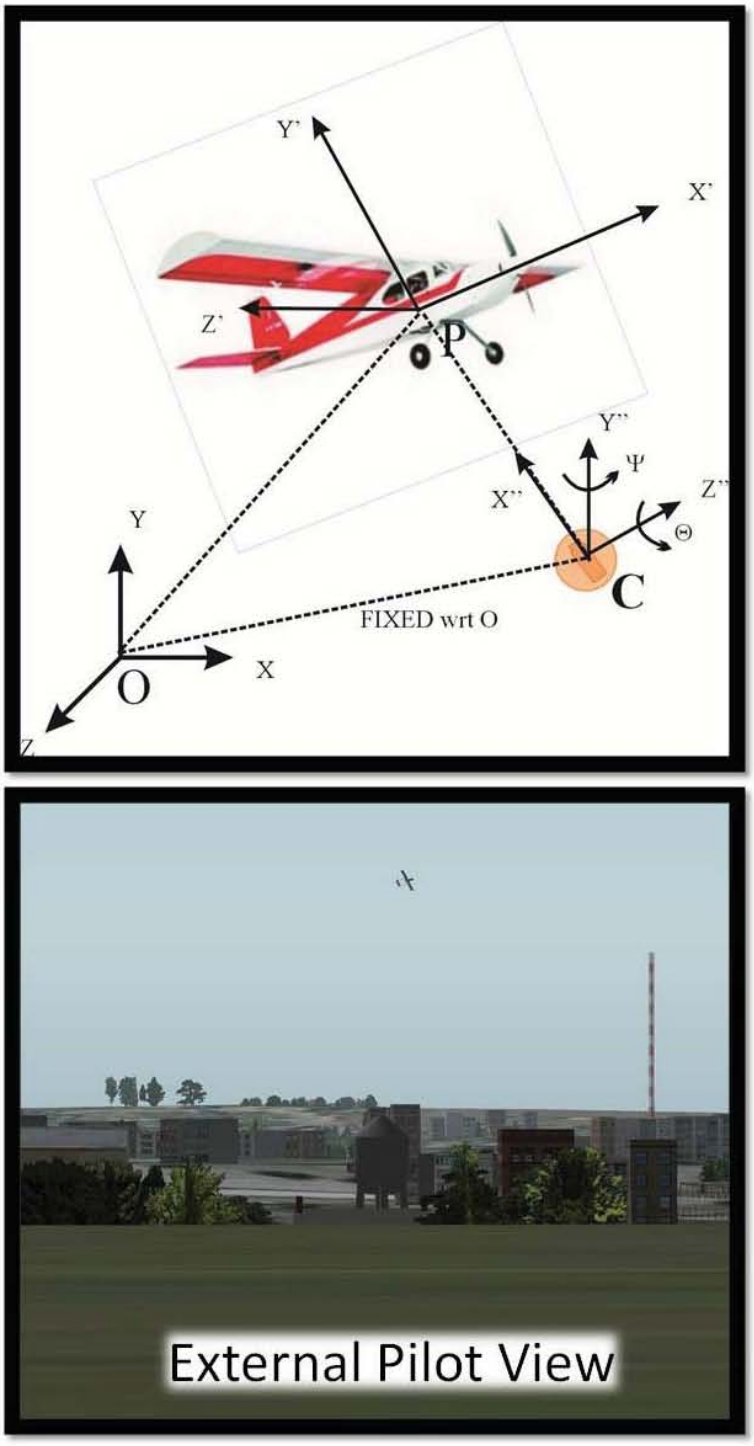

Figure 3.3: Top: Reference frames used for generating an external view; Bottom: Example of a simulated external pilot view.

tended to become pixelated and the pilot would lose sight of the orientation of the vehicle much sooner than they would in the real world. To alleviate this issue, an auto zoom function was created to keep the UAV from becoming pixelated in the image. The following equations were used to orient the virtual camera with respect to the global reference frame to produce the external pilot view. The reference frames are shown in Figure 3.3.

$$\begin{bmatrix} C_X \\ C_Y \\ C_Z \end{bmatrix} = \begin{bmatrix} C_X \\ C_Y \\ C_Z \end{bmatrix} \tag{3.4}$$

$$\begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} 0 \\ sin^{-1}((P_Y - C_Y) - 10/\,||P_{XYZ} - C_{XYZ}|\,|) - 10 \\ tan^{-1}((P_X - C_X)/(P_Z - C_Z)) \end{bmatrix} \tag{3.5}$$

$$Zoom \;= \begin{bmatrix} \;||P_{XYZ} - C_{XYZ}|\,|/thresholddist \end{bmatrix} \tag{3.6}$$

where "O" is the global reference frame for the flight simulator with X, Y and Z coordinates. "P" represents the aircraft reference frame with X', Y', and Z' coordinates. "C" represents the camera reference frame with X", Y", Z" coordinates. The camera distance from the global reference frame represents the location of the external pilot. The angles correspond to the angles of the camera and not the angular position of the aircraft. A value of ten degrees is subtracted from the pitch angle such that the aircraft is positioned higher in the field of view, thereby maximizing the ground/horizon in the pilot's peripheral view. The zoom function has a value called "threshold dist" which represents the maximum distance the aircraft can fly before it becomes pixelated. The zoom function stays at a value of one until that threshold is reached. Once the aircraft passes the threshold, the camera axis moves along the vector $\vec{CP}$ by a distance corresponding to the calculated zoom function. This ensures that the

Figure 3.4: Plugin demonstrating simulated catapult launch.

aircraft never becomes pixelated in the field of view. This adds an unrealistic effect where the aircraft does not get smaller in the field of view the farther it moves from the pilot (once the threshold is exceeded). External pilots do not typically operate the vehicles at extreme distances so this is not an issue.

Positioning the Aircraft

Two other functions were developed for the simulator that are necessary for its use as a training and evaluation tool. Figure 3.4 is an example of the developed position plugin that can place the UAV in any location, orientation and velocity. Currently the figure shows the UAV in a catapult launch situation. It can also be utilized to place the aircraft in different scenarios like landing approaches or in a situation just before an accident for training pilots on recovery techniques. This is important to current UAV systems such as the Predator where a large portion of accidents occur during the takeoff and landing phases of the mission.

Development of the position plugin required the use of quaternions. A common

way amongst the aircraft community for representing aircraft attitude is through the use of Euler angles, axis angles, and direction cosines. Many aircraft control engineers, roboticists and video game developers have shied away from using these types of representations because they are either computationally inefficient or prone to singularities at critical orientations such as an angle of 0 or 90 degrees [62]. At these singular points, the system loses a degree of freedom resulting in what is commonly referred as gimbal lock. Rather, many game developers utilize quaternions to provide smooth rotations and avoid the problem of gimbal lock. The developers of X-Plane chose this method of aircraft attitude representation.

Positioning the aircraft in an exact location and orientation in the X-Plane environment based on quaternions is not as intuitive as inputing the angular position based on yaw, pitch and roll angles. Therefore, the position plugin interface was designed to allow a user to input the position of the aircraft using yaw, pitch and roll angle representation. These angles are then converted to quaternion representation and written to the X-Plane program. A basic knowledge of quaternions is necessary to understand the conversion method used.

Quaternions encode rotations by four numbers, three of which have an imaginary component. The quaternion itself is defined as:

$$q = \begin{bmatrix} q_0 \\ q_1 i \\ q_2 j \\ q_3 k \end{bmatrix} = \begin{bmatrix} cos(\Theta/2) \\ sin(\Theta/2)cos(\beta_X)i \\ sin(\Theta/2)cos(\beta_Y)j \\ sin(\Theta/2)cos(\beta_Z)k \end{bmatrix} \tag{3.7}$$

$$i^2 = j^2 = k^2 = -1 \tag{3.8}$$

$$ij = -ji = k \tag{3.9}$$

$$jk = -kj = i \tag{3.10}$$

$$ki = -ik = j \tag{3.11}$$

where $\cos(\beta_X)$, $\cos(\beta_Y)$, and $\cos(\beta_Z)$ are the direction cosines representing the axis of rotation. Θ is the scalar angle of rotation about that axis. q_0 is also known as the scalar part of the quaternion and q_1, q_2, and q_3 is the vector part. The unit quaternion has the property such that:

$$q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1 \tag{3.12}$$

Successive rotations between frames, such as rotating from one coordinate frame to another, is described through the products of quaternions. For example a frame represented by the quaternion "a" is rotated using a quaternion representation of the rotation "b". The resulting quaternion is equal to the product of the a and b quaternions as shown below.

$$\begin{aligned} a \otimes b = (a_0 + a_1 i + a_2 j + a_3 k) \otimes (b_0 + b_1 i + b_2 j + b_3 k) = \\ (a_0 b_0 - a_1 b_1 - a_2 b_2 - a_3 b_3) + \\ (a_0 b_1 + a_1 b_0 + a_2 b_3 - a_3 b_2) i + \\ (a_0 b_2 - a_1 b_3 + a_2 b_0 + a_3 b_1) j + \\ (a_0 b_3 + a_1 b_2 - a_2 b_1 + a_3 b_0) k. \end{aligned} \tag{3.13}$$

If we have three Euler angles, such as the yaw(ψ), pitch(θ) and roll(ϕ) of the aircraft, we can form three independent quaternions:

$$Q_1 = [cos(\phi/2), sin(\phi/2), 0, 0] \tag{3.14}$$

$$Q_2 = [cos(\theta/2), 0, sin(\theta/2), 0] \tag{3.15}$$

$$Q_3 = [cos(\psi/2), 0, 0, sin(\psi/2)] \tag{3.16}$$

The rotation corresponding to $\mathrm{R}_\psi \mathrm{R}_\theta \mathrm{R}_\phi$ is:

$$\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = Q_3 \otimes Q_2 \otimes Q_1 = \begin{bmatrix} c_{\phi/2}c_{\theta/2}c_{\psi/2} + s_{\phi/2}s_{\theta/2}s_{\psi/2} \\ s_{\phi/2}c_{\theta/2}c_{\psi/2} - c_{\phi/2}s_{\theta/2}s_{\psi/2} \\ c_{\phi/2}s_{\theta/2}c_{\psi/2} + s_{\phi/2}c_{\theta/2}s_{\psi/2} \\ c_{\phi/2}c_{\theta/2}s_{\psi/2} - s_{\phi/2}s_{\theta/2}c_{\psi/2} \end{bmatrix} \tag{3.17}$$

To convert back from quaternions to Euler angles, the equation is:

$$\begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} atan2(2(q_2q_3 + q_1q_0), (-q_1^2 - q_2^2 + q_3^2 + q_0^2)) \\ arcsin(-2(q_1q_3 - q_2q_0)) \\ atan2(2(q_1q_2 + q_3q_0), (q_1^2 - q_2^2 + q_3^2 + q_0^2)) \end{bmatrix} \tag{3.18}$$

Real world latencies

An external program was developed in C# to control the amount of time lag between data communication. This allows a user defined lag time between the simulator sending/receiving information. The delay can represent real world communication delays in actual UAV operations. Depending on the distance of operation from the ground station, time lag can be present in the onboard camera feed, transmission of joystick commands, and transmission of state information from the onboard avionics. For realism, pilots must be introduced to real world delays associated with the specific mission during training.

3.2 Flight Environment

Unlike traditional high altitude environments common to military UAV use, near Earth environments are usually cluttered with obstacles such as people, trees, build-

ings, power lines, etc. Even more important, vehicles in these environments will most likely encounter situations where interaction with the surrounding civilian population is needed. An example of this would be external load transportation or rescue. These types of operations demand extreme situational awareness and quick adaptation to the ever changing dynamic environment. Whether or not these vehicles are directly controlled by a pilot or are fully autonomous, it is necessary to operate in similar environments and situations before actual testing at the final desired locations. These preliminary tests serve to train the pilots for flying the vehicle in specific conditions. For fully autonomous systems, the preliminary tests help to refine the control algorithms. For the preliminary work, field testing with all the hardware can be very time consuming and costly, especially in the event of an accident. It is also very difficult to control most of the environmental variables in the testing area. Simulation offers an advantage as it is cheaper to operate and the environmental conditions are more easily controlled.

Recently, simulators have been utilized in the unmanned aerial vehicle community to help develop more robust autonomous flight controllers. However, very few have utilized simulation tools for UAV pilot training and evaluation in near Earth and urban environments. Theodore et al. [63] utilized the Real-time Interactive Prototype Technology Integration/Development Environment (RIPTIDE) with a Yamaha RMAX helicopter dynamics model to develop a graphical environment that simulated and evaluated autonomous helicopter landing in an urban setting. Their parking lot scenario for landing included buildings, street lights, cars and trees. They showed that the simulation environment proved to be an effective tool for the performance evaluation of the machine vision algorithms even though the images were computer generated. Stoor et al. [64] have presented a paper on the development of a realistic urban simulation environment to study the performance of cooperative control

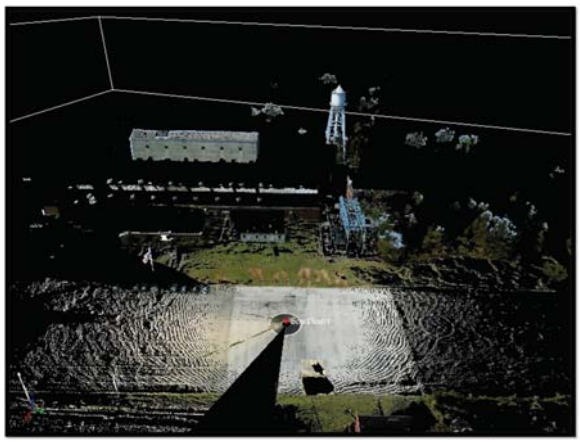

Figure 3.5: 3D laser scan of a near Earth environment

Figure 3.6: Left: Real satellite image of a near Earth environment; Right: Recreated in the virtual world

algorithms for UAVs in and around the urban landscape. As of 2006, their simulator included people, ground vehicles, buildings, flight dynamics models for UAVs and models of steady-state winds and turbulence.

X-Plane allows the importation of detailed terrain and environment obstacles. This is valuable to UAV training because of the ability to develop an environment exactly like the field testing arena. Laser scan data (Reigl LMS-Z210) as seen in Figure 3.5, physical measurements, and satellite imagery can be used to recreate a real

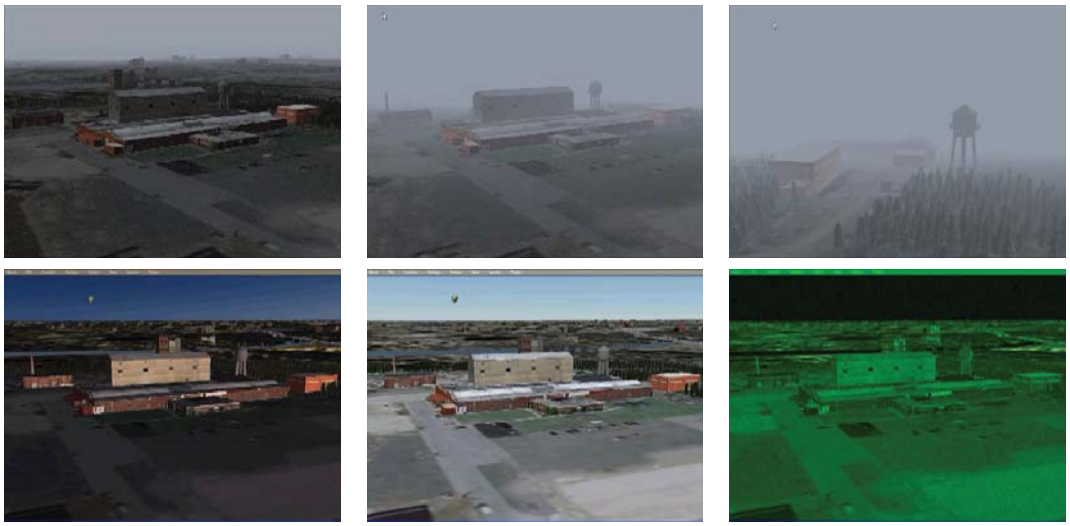

Figure 3.7: Top: Changes in weather from downpour left to increased fog right; Bottom: Changes in lighting conditions (Night vision far right)

world near Earth environment as seen in Figure 3.6. The area shown is the Piasecki Facility in Essington, PA. It is a good representation of a near Earth environment because of the buildings, trees, power lines, etc. With detailed texturing, the environment can look very realistic. As mentioned in Theodore et al. [63], simulated camera views can be used for vision algorithms such as feature detection which is important for tasks such as identifying safe landing zones for autonomous rotor craft.

UAVs are typically smaller and lighter than their manned counterparts. This makes them very susceptible to changing weather conditions such as wind, including turbulence, and precipitation. Operators of UAVs, both internal and external, are susceptible to changes in the visual field. Ground station operators utilize the view from the onboard UAV camera and external pilots rely on direct line of sight with the vehicle. X-Plane includes a comprehensive weather model that models fog, clouds, wind, turbulence, rain, snow, hail and thunderstorms. Users have full control of all these conditions. Also shown in Figure 3.7 top is an example of the Piaseki compound

Parameter	Value	Units
X-axis range (length)	17.72	ft
X-axis speed	2	ft/s
X-axis acceleration	2.95	ft/s^2
Y-axis range (width)	13.84	ft
Y-axis speed	2	ft/s
Y-axis acceleration	3.69	ft/s^2
Z-axis range (height)	5.43	ft
Z-axis speed	2	ft/s
Z-axis acceleration	7.40	ft/s^2

Figure 3.8: Left: SISTR workspace and specifications; Right: Image of the SISTR setup with a UAV sensor suite attached to the end effector. This image was adapted from [11].

under heavy rain conditions and in thick fog. Shown in Figure 3.7 is the environment under varying lighting conditions (different times of the day) and during night using night vision. It is valuable to train UAV pilots and test control algorithms under all possible weather and lighting conditions that could be encountered during real world tests.

3.3 Integration with SISTR

Simulation is only as good as the model being used to represent the object or event being simulated. It can be difficult to accurately model aspects of real world sensor performance in simulation. It has been shown in the previous section that through simulation, we can create very realistic weather conditions such as fog and rain,

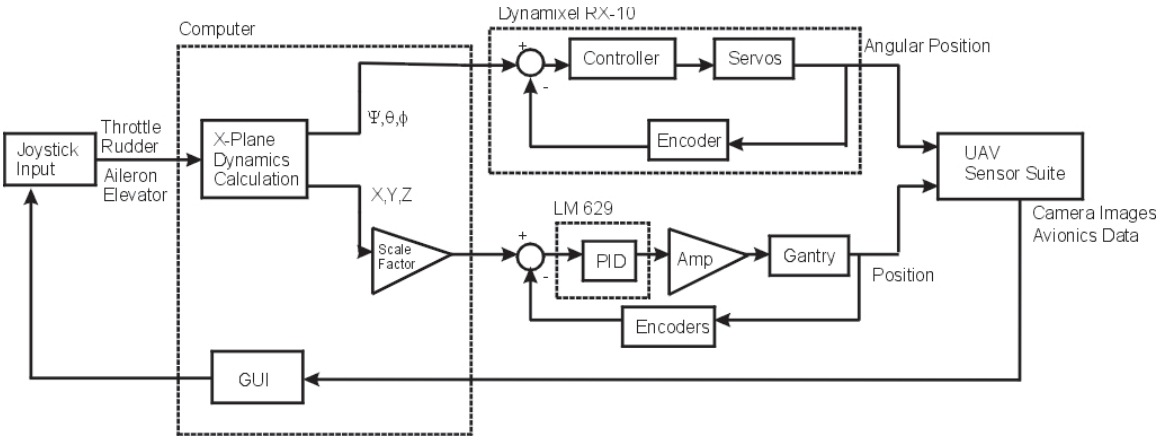

Figure 3.9: Block diagram for the training and evaluation system that is integrated with SISTR(gantry).

however accurately simulating a sensor's response to those conditions is challenging. The Systems Integrated Sensor Test Rig (SISTR) was developed to address these challenges. SISTR, as seen in Figure 3.8 is a three degree of freedom gantry system with a workspace measuring 18 feet long by 14 wide and 6 feet tall [11]. As seen in Figure 3.8, the gantry has ample workspace to allow construction of replicas of real world environments. In most cases, the real world environment is a scaled model to further augment the active workspace. SISTR was developed as a hardware-in-the-loop test rig and was designed to be used to evaluate obstacle detection sensors (Lidar, computer vision, ultrasonic, ultrawideband radar, millimeter wave radar, etc.), design sensor suites, and test collision avoidance algorithms. For this work, SISTR was integrated with the flight simulation software and was modified to encompass the training and evaluation of full UAV mission scenarios.

Figure 3.9 shows a block diagram of the integrated modified flight sim and SISTR system. SISTR's end effector is used to represent the location of an aircraft inside of the scaled environment. As seen in Figure 3.9, aircraft dynamics during operations are handled by the flight simulator and the scaled translational positions of the aircraft

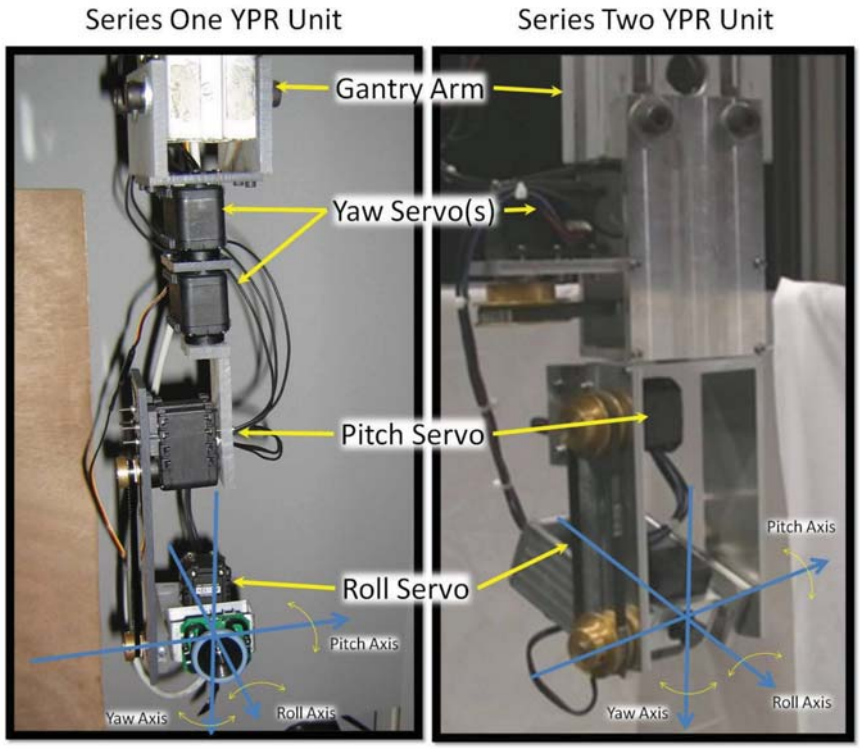

Figure 3.10: Yaw, pitch and roll unit used to recreate the angular position of the aircraft inside of SISTR. The unit is designed based on the Euler angles of the aircraft. Yaw is applied first, then pitch, then roll. Left: First series yaw, pitch, roll unit. Right: Second series yaw, pitch, roll, unit.

are relayed from the flight simulator to SISTR's controller via UDP at a rate of 20 Hertz . Currently SISTR uses a proportional, integral, derivative (PID) controller to drive the gantry end effector to the commanded positions.

The aircraft's control surface deflections are commanded by the subject (pilot) via a joystick. The resulting angular position of the aircraft, generated by the flight simulator, is relayed to a three DOF yaw, pitch and roll (YPR) unit attached to SISTR's end effector as seen in Figure 3.10. The YPR unit was specifically designed such that it moves according to the Euler angles of the aircraft; yaw is applied first, then pitch, then roll. It was also designed to have a small footprint due to operation in a scaled environment. A 640x480 resolution wireless camera with 70 degree field

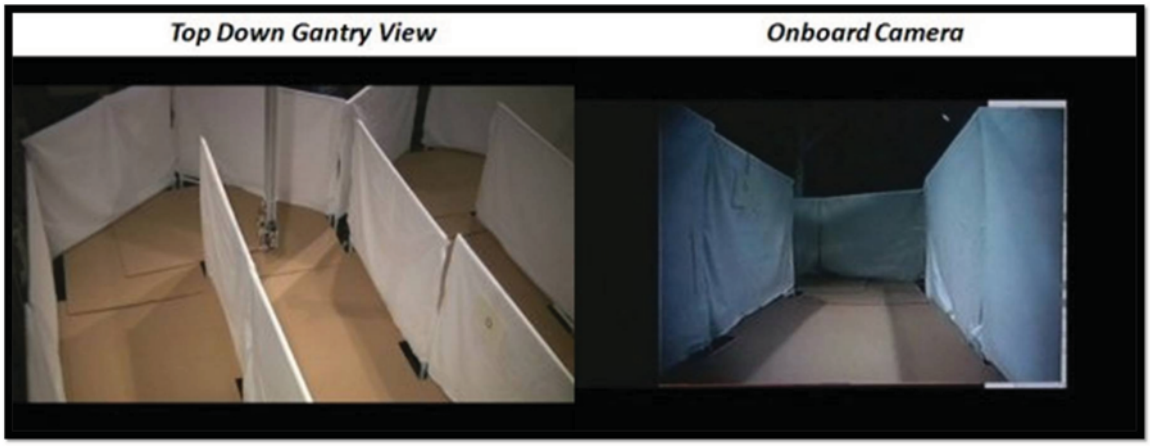

Figure 3.11: Left: Top down view of an example environment built inside of the gantry. Right: The onboard camera image of the environment.

of view was attached to the YPR unit as seen in Figure 3.10. The images from the camera represent the onboard camera view from the aircraft. The images are are fed back to the pilot located at the ground station. The second series YPR unit was designed to minimize vibrations and increase the angular workspace as compared with the first series. Figure 3.11 shows an example near earth environment built in the gantry and the resulting onboard camera image that is relayed back to the operator.

4. Motion Platform Integrated UAV Pilot Interface

The capability to train pilots for near Earth operations in mission type scenarios helps decrease the chances that a pilot will make a mistake due to inexperience. However, training alone can not address all causes of pilot mishaps. Situational awareness of the pilot can still be relatively low even with extensive training. This requires that we investigate approaches to enhance the situational awareness of the pilot that can be integrated into the training system presented in the previous chapter.

Accident reconstruction experts have observed that UAV pilots often make unnecessarily high-risk maneuvers. Such maneuvers often induce high stresses on the aircraft structure, accelerating wear-and-tear on the vehicle or even causing crashes. The motion platform integrated into the piloting system would recreate the sense of "shared fate" for the UAV pilot. Pilots of manned aircraft share the fate of their vehicle which includes feeling the motions, hearing sounds, and seeing the surrounding environment. They utilize this information for decision making and increased flight control. The motion platform offers a high fidelity flight experience to the UAV pilot and allows the unmanned aircraft to conduct tasks that commonly require direct human control. The hypothesis is that adding motion cueing to the pilot of a UAV can offer significant improvement over current piloting interfaces. The virtual immersion of a pilot inside the cockpit of the UAV will improve pilot reaction times, allow for more precise control and awareness of the aircraft, affect pilot decision making and risk taking behaviors, and decrease the number of UAV accidents.

The hypothesis is supported by previous research conducted on the effectiveness of motion cueing in flight simulators and trainers. The majority of the results show that motion cueing in the simulators does improve pilot performance over fixed-motion simulators. In rotorcraft especially, motion cueing in simulations have helped improve

pilot performance for a significant number of flight tasks. A study by Ricard and Parrish [65] showed that pilots performed best when performing a simulated helicopter hover with a moving motion base than with a fixed base. In Parish et al. [66] the authors compared a moving base to a fixed base simulation of a helicopter following a slalom course. Their results showed no differences in system error under the two conditions. However, more importantly, they showed that less control activity was present under motion conditions than under fixed based conditions. They attributed this to the pilots perceiving the realistic limitations of the machine due to the motion cueing. This is an important finding as pilots of UAVs can put the vehicle into extreme maneuvers (leading to crashes) due to the limited physical sense of the strain that they are putting on the vehicle. The benefits of motion cueing are not just limited to rotorcraft as any vehicle control will be improved by decreasing operator response time. Zacharias and Young [67] tested human subjects' response times to motion from a five degree per second step in angular velocity. They found that the vestibular system is able to detect acceleration much sooner than the visual system. This implies that a pilot would be able to correct for any disturbance in the flight sooner with motion cues than just visual cues alone. The reason for this is that the vestibular system can easily detect changes in acceleration but it can only detect constant motion for a brief period of time. The brain processes the visual information coming in and the visual system takes over for detecting constant motion. Naturally, our bodies utilize both the vestibular and visual systems together to optimize our reaction times and controls in dynamic environments.

4.1 Tele-operation setup

The tele-operated system is made up of five major parts: the motion platform, the aerial platform, the onboard sensors including wireless communication, the PC

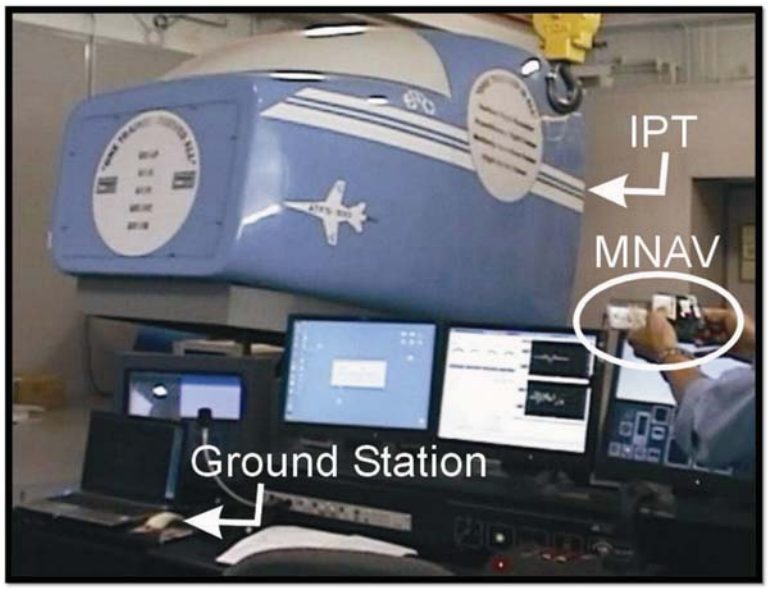

Figure 4.1: IPT 4-DOF motion platform from ETC being wirelessly controlled with the MNAV.

Table 4.1: Select ETC GYRO IPT II Motion System Capabilities

Degree of Freedom	Displacement	Speed	Acceleration
Pitch	± 25 deg	0.5 - 25 deg/sec	0.5 - 50 deg/sec^2
Roll	± 25 deg	0.5 - 25 deg/sec	0.5 - 50 deg/sec^2
Continuous Yaw	±360 degrees continuous	0.5-150 deg/sec	0.5-15 deg/sec^2

to remote control circuit and the ground station.

4.1.1 Motion Platform

To relay the motion of the aircraft to the pilot during both simulation and field tests, the authors utilized a commercially available 4-DOF flight simulator platform from Environmental Tectonics Corporation (ETC) shown in Figure 4.1. ETC designs and manufactures a wide range of full-motion flight simulators for tactical fighters, general fixed-wing aircraft and helicopters. For initial development, a 4-DOF

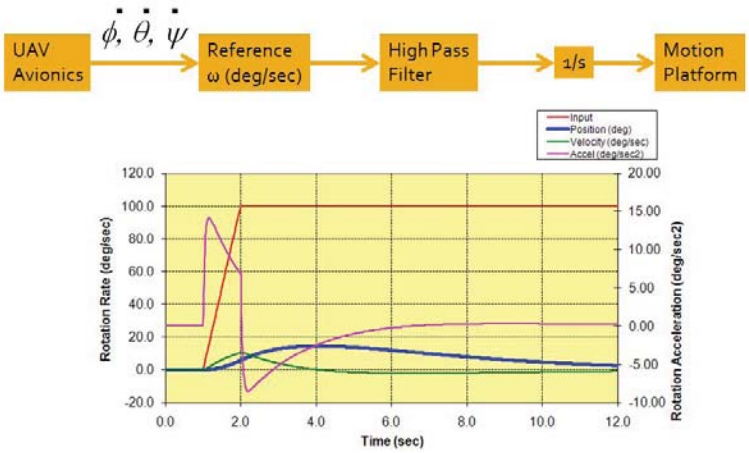

Figure 4.2: Top: Simplified block diagram of the UAV sensor and motion platform system. Bottom: Example data for one axis of the motion platform when an angular rate data is inputted into the system.

Integrated Physiological Trainer (IPT) system was employed because of its large workspace and fast accelerations that are needed to replicate aircraft flight. The motion system capabilities are shown in Table 4.1. The cockpit is modified for specific aircrafts offering a high fidelity experience to the pilot. The visual display inside the motion platform can handle up to a 120 degree field of view. Basic output from the motion platform utilized in this work are the flight commands from the pilot in the form of encoder positions of the flight stick (pitch and roll), rudder pedals (yaw), and throttle.

The motion platform generates the appropriate motion cues to the pilot based on the angular velocities that it receives from the ground station. Motion cues are brief movements in the direction of acceleration which give the sensation of constant motion to the pilot but are "washed out" before the motion platform exceeds its reachable workspace. Washout algorithms are commonly used by the motion platform community to return the platform to a neutral position at a rate below the threshold that humans can sense [68]. This allows the platform to simulate motions much

Figure 4.3: Left: The Sig Kadet model aircraft used as the testing platform. Right: MNAV and Stargate in the cockpit of the aircraft (top view).

greater than its reachable workspace. This is done through the use of low pass and high pass filters. In a classical washout algorithm, high pass filters serve to attenuate the low frequency accelerations that cause the motion-base to reach its limitations. The high frequency accelerations last for a small duration of time, and thus will not drive the motion-base to its physical limits. Low pass filters are used in generating tilt angles to simulate forces due to translational accelerations. Since the focus is on angular motion cues and not translational accelerations, the block diagram of the UAV motion platform interface can be simplified to what is shown in Figure 4.2. After passing through the high pass filter, the angular rate is integrated to produce angular position data which is fed into the motion platform. An example of how one axis of angular position of the motion platform would respond to an angular rate input is also shown. The response of the system can be tuned by adjusting the filter parameters. For the IPT motion platform in particular, pitch and roll rate data streaming from the onboard UAV sensor suite are washed out. The yaw rate is fed straight through due to the continuous yaw capabilities of the IPT motion platform.

4.1.2 Aerial Platform

UAV rotorcraft are of interest because they are well suited to fulfill missions like med-evac and cargo transport which demand hovering, pirouettes and precision positioning. For proof of concept, the immediate goal was to ensure a master-slave setup where the UAV's motions could be reproduced (in real-time) on a motion platform. To build system components, a fixed-wing UAV was used for initial demonstrations.

The Sig Kadet offers a cheap and quick crash recovery solution for initial tests. With the Sig Kadet, the proper sensor suite and communication issues can be worked out before switching to a commercial UAV. The Sig Kadet shown in Figure 4.3 left, is a very stable flight platform and is capable of carrying a sensor suite and camera system. It uses five servo motors controlled by pulse position modulated (PPM) signals to actuate the elevator, ailerons, rudder and throttle. With its 80 inch wingspan, it is comparable in size to the smaller back-packable UAVs like the FQM-151 Pointer and the Raven [1].

4.1.3 Onboard Sensors

On board the aircraft is a robotic vehicle sensor suite developed by Crossbow inertial systems. The MNAV100CA (MNAV) is a 6-DOF inertial measurement unit (IMU) measuring onboard accelerations and angular rates at 50 Hertz. It is also capable of measuring altitude, airspeed, GPS and heading. The MNAV is attached to the Stargate, also from Crossbow, which is an onboard Linux single board computer. The Stargate is set to transmit the MNAV data at 20 Hertz to the ground station via a wireless 802.11 link. As shown in Figure 4.3 right, the MNAV and Stargate fit inside the cockpit of the Sig Kadet close to the aircraft's center of gravity.

Onboard video is streamed in real time to the ground station via a 2.4 Giga Hertz wireless transmission link. The transmitter is held under the belly of the Sig Kadet

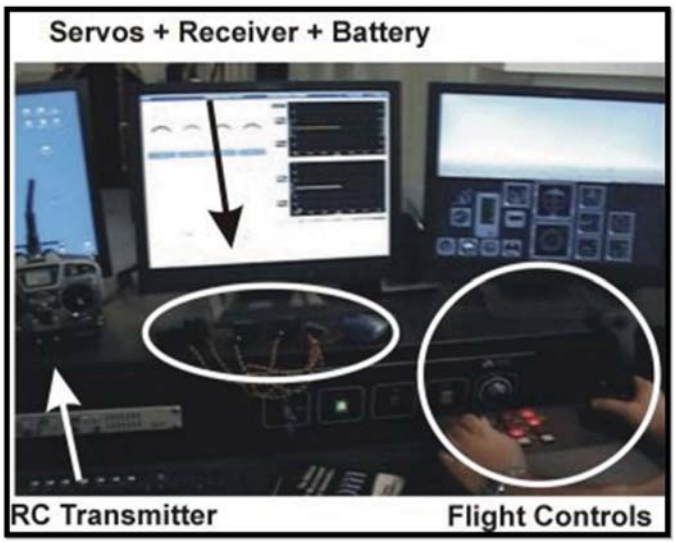

Figure 4.4: Computer to Remote Control configuration. Flight controls from the instructor stick, which map to the same controls from inside the IPT motion platform cockpit, are transmitted to the servo motors.

and the camera is located off the left wing of the aircraft. The current camera used has a 70 degree field of view and is capable of transmitting images at 30 frames per second (FPS) and 640 x 480 resolution to a distance of 1.5 miles (AAR03-4 / 450 Camera from wirelessvideocameras.net). This is relatively low quality as compared with high definition camera systems but it is inexpensive, making it a decent choice for initial tests. Future tests should include much higher resolution cameras and a more strategic placement of the camera to replicate a pilot's onboard view.

4.1.4 PC to RC

Position encoder data from the flight stick, rudder pedals, and throttle inside the motion platform are transmitted via an Ethernet link to the ground station. The signals are then routed through a PC to RC circuit that converts the integer values of the encoders to pulse position modulated (PPM) signals. The PPM signals are sent through the buddy port of a 72 Mega Hertz RC transmitter which then transmits

the signal to the RC receiver on board the aircraft. The PPM signals are routed to the appropriate servos to control the position of the ailerons, elevator, rudder, and throttle of the aircraft. The positions of the IPT flight controls are currently sent through the PC to RC link at a rate of 15 Hertz. The PC to RC setup can be seen in 4.4.

4.1.5 Ground Station

The ground station used for the tele-operation system is a version of the MNAV Autopilot Ground station freely distributed on SourceForge.net that was highly modified to fit this project's needs. The modified ground station does three things. 1) It receives all the data packets being sent wirelessly using UDP from the MNAV, decodes the packets and displays the relevant information such as velocities and attitude to the user operating the ground station. 2) It acts as the communication hub between the aircraft and the motion platform. It relays the MNAV information via Ethernet link to the motion platform computers and sends the flight control positions of the motion platform to the PC to RC circuit via USB. 3) It continuously monitors the state of the communication link between the motion platform and the MNAV. If something fails it will put both the motion platform and aircraft (via the MNAV/Stargate) into a safe state.

4.1.6 Field Tests

Current field tests have been conducted at a local RC flying field with the aircraft under full RC control. The field is approximately a half mile wide and a quarter mile deep. Avionics data such as angular velocity rates, accelerations and elevation was collected and recorded by the MNAV attached to the aircraft during flight. Video from the onboard camera was streamed wirelessly to the ground station and recorded.

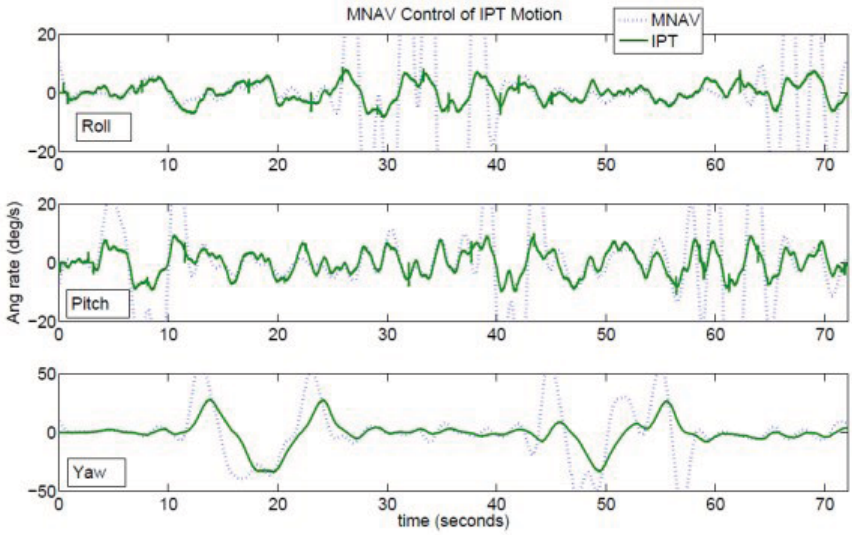

Figure 4.5: Comparison of the angular rates during MNAV control of the IPT.

During each flight, the RC pilot conducted take off, figure eight patterns and landing with the Sig Kadet.

4.2 Results and Discussion

In this section, initial test results from the hardware control portion of the UAV system are presented. In this prototyping stage, development was divided into three specific tasks that include: motion platform control using the MNAV, control of the aircraft servos using the IPT flight controls, and recording of actual flight data from the MNAV and replay on the IPT.

4.2.1 Motion Platform Control with MNAV

Aircraft angular rates are measured using the MNAV and this information is transmitted down to the ground station via a 20 Hertz wireless link. Task A demonstrated the MNAV's ability to communicate with the ground station and the IPT. The MNAV was held in hand and commanded pitch, roll and yaw motion in real time

to the IPT by rotating the MNAV in the pitch, roll and yaw directions.

Motions of the MNAV and IPT were recorded. Figure 4.5 shows a plot comparing MNAV and IPT data. The IPT is designed to replicate actual flight motions and therefore is not capable of recreating the very high angular rates commanded with the MNAV during the hand tests in the roll and pitch axis. The IPT handles this by decreasing the value of the rates to be within its bandwidth and it also filters out some of the noise associated with the MNAV sensor. Overall, the IPT tracked the motion being commanded by the MNAV fairly well. The IPT is limited by its reachable work space which is why the amplitude of the angular rates does not match at all times. Minimal lag between the commanded motion from the IMU and the resulting motion in the IPT is desired as significant differences between the motion cues from the IPT and visuals from the video feed will cause a quick onset of pilot vertigo.

4.2.2 Control of Aircraft Servos

Transmitting wirelessly at 15 Hertz, no lag was observed between the flight commands and the servo motor response. This is significant because it means that the pilot sitting inside the motion platform can control the aircraft through the RC link. This underscores fidelity of the system; the aircraft will respond as if the pilot was inside its cockpit and flying the aircraft. This has only been tested during line of sight control. RC is limited in range and as stated earlier, satellite communication links for long range distances can introduce delays in data transfer. However many near Earth UAV applications can be conducted with ground stations near the operation site.

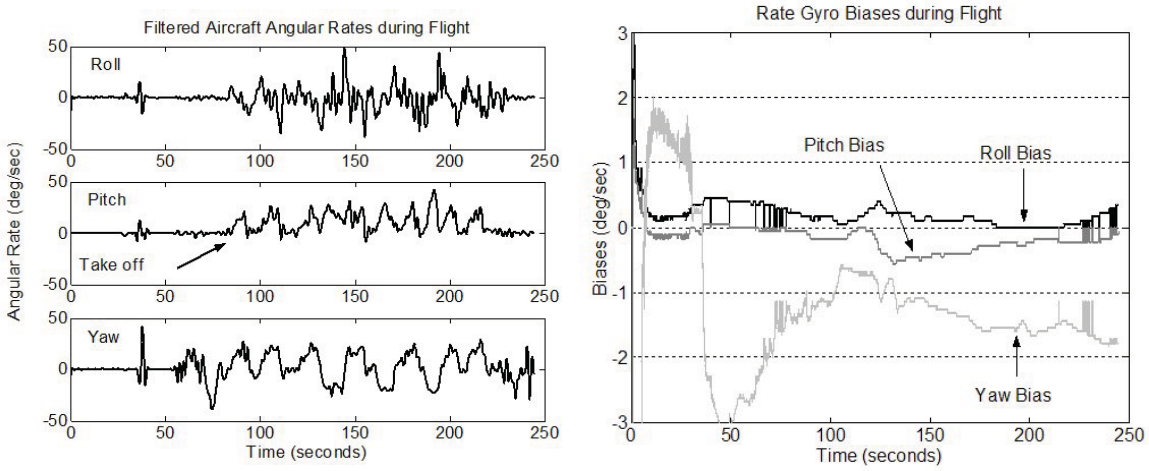

Figure 4.6: Left: Filtered angular rates during actual aircraft flight. Right: Rate gyro biases during actual aircraft flight.

4.2.3 Record and Replay Real Flight Data

It was demonstrated that the MNAV is able to transmit motion data to the IPT. During this next task the MNAV was subjected to extreme rates and poses. Such extremes are not representative of actual aircraft angular rates but serve to demonstrate master-slave capability. To test the IPT's ability to respond to actual aircraft angular rates being sent from the MNAV, angular rate data was recorded directly from a field flight of the Sig Kadet. This data was replayed on the IPT along with onboard flight video. The recorded video and flight data simulate the real time streaming information that would occur during a field tele-operation experiment. An example of the recorded angular rates from one of the field tests is shown in Figure 4.6 left and a still shot of the onboard video recording is shown in Figure 4.7 left.

Initial results showed errors in the angular rates between the observed motion and the recorded data. For example, the pitch rate (Figure 4.6 left), while it is oscillating, rarely goes negative. This means that the sensor is measuring a positive pitch rate during most of the flight. Analysis of the data proved that it was not a simple offset

Figure 4.7: Left: Onboard camera view off of the left wing during flight. Right: UAV cargo transport in a cluttered environment using a radio link that slaves robotic helicopter motions to the motion platform. Through a shared fate sensation the pilot flies by feeling the UAVs response to maneuvers commanded by the pilot.

fix. This was consistently the case for multiple flights. This phenomenon was only seen during flights. Hand held motions always produced correct and expected angular rates. The recorded flight data was replayed on the IPT motion platform. This caused the IPT to travel and remain at its kinematic joint limits as was expected because of the positive pitch rate.

The IMU was reprogrammed to output angular rates that reflect the bias correction made in the Kalman filter for the rate gyros [69]. A plot of the biases during a real flight is shown in Figure 4.6 right. The resulting biases were very small and did little to fix the positive pitch rate phenomenon during flights. Alternative IMUs should thus be explored. None the less, the integration of an IMU and motion platform was successfully developed. This underscores that the wireless communication interface and low-level avionics work as designed.

4.3 Summary

While the future of UAVs is promising, the lack of technical standards and fault tolerant systems are fundamental gaps preventing a vertical advance in UAV innovation, technology research, development and market growth. This chapter has presented the development of the first steps toward a novel tele-operation paradigm that employs motion cueing to augment UAV operator performance. This method has the potential to decrease the number of UAV accidents and expand the role of unmanned technology to more applications.

Leveraging this work, future development would include research to eliminate, reduce, or compensate for the motion lag in the motion platform. Also of interest would be to examine additional cues like sight, touch and sound that may improve UAV control. From such understanding, one can analytically design systems to better control UAVs, train UAV pilots and help eliminate UAV accidents.

The shared fate and motion cueing will have tremendous benefit in near Earth flying. Figure 4.7 right depicts a notional mission involving cargo pickup and transport through a cluttered terrain to a target location. The motion platform can be used to implement a virtual "shared fate" infrastructure to command a robotic helicopter. The visuals from the helicopter's onboard cameras would be transmitted to the motion platform cockpit. Added cues like audio, vibration, and motion would enable the pilot to perform precision maneuvers in cluttered environments like forests or urban structures. Future studies demand the look at rotorcraft because their potential applications extend beyond the capabilities of current fixed wing UAVs. Among these are applications such as search and rescue and fire fighting. Even cargo transport is still very difficult to achieve autonomously in non-optimal conditions and cluttered environments. These tasks require quick, precise maneuvers and dynamic mission plans due to quickly changing environmental conditions and close quarter terrains.

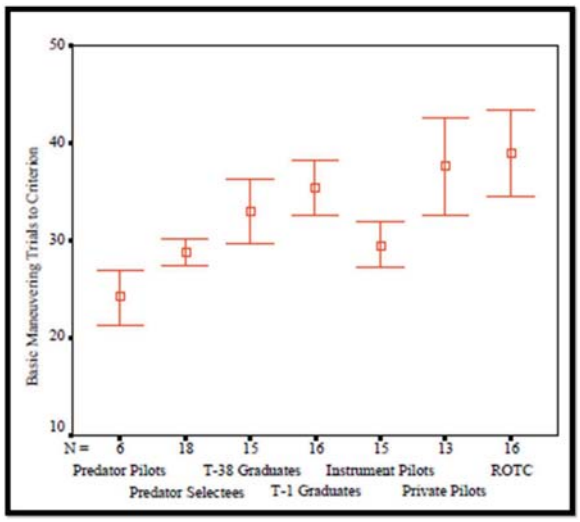

Figure 4.8: Left: Number of trials to achieve criterion performance on the Basic Maneuvering Tasks. Reprinted from [12].

To date these missions can only be flown by experienced, on board pilots, who still incur a great deal of risk.

It became apparent during field tests that transportation and integration of the motion platform at the field site can be prohibitively expensive for some. However, this does not eliminate the potential of motion platforms to enhance UAV operations. It may be beneficial to have UAV pilots train with simulated UAVs while inside of the motion platforms to get a feel for the motions the aircraft experiences due to their commands. When deployed to the field, they can operate the UAVs without the motion platform, operating purely from muscle memory of the motions they felt while training with the system. This claim is supported by research from Schreiber et al. [12]. In this work, they found that pilots who had manned aircraft experience in aircraft with similar handling characteristics of the Predator, performed better than pilots with other types of manned aircraft training and those without any manned aircraft training at all. Figure 4.8 presents an example from that research which shows pilots from the T-38 and civilian aircraft performing better than other aircraft

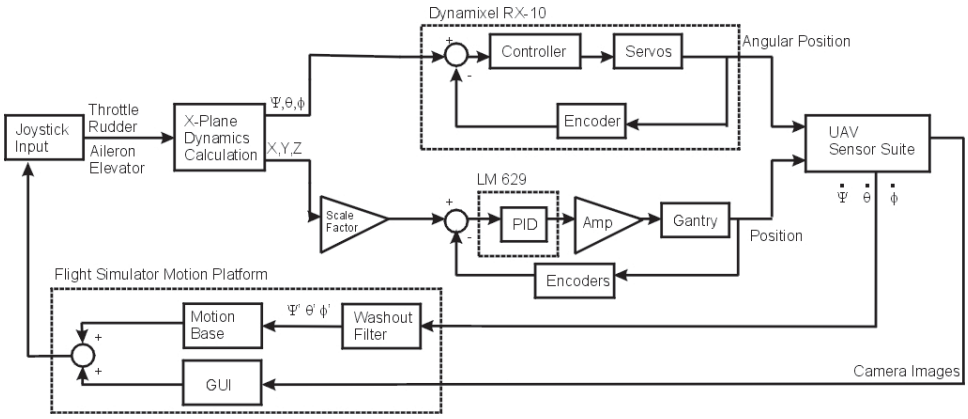

Figure 4.9: Block diagram for motion platform integration with SISTR.

and non aircraft pilots due to the similarities between their vehicles and the Predator UAV.

To address that thought, the motion platform can be easily integrated into the training and evaluation system presented in Chapter 3. Real world UAV avionics such as the interial measurement unit, can be added to the gantry endeffector. The captured motions from the sensors can then be used to drive the motions of the platform during flights. Sensor data which can not be captured indoors, such as GPS, can be simulated by modifying data being exported by the flight simulator. Figure 4.9 shows the block diagram detailing the integration.

This chapter has presented the development of the subsystems required for motion platform integration into UAV operations. However, it represents a solution to just one of the number of limitations to UAV pilot situational awareness presented in Chapter 2. The following chapter presents methods to address these additional concerns.

5. Mixed-Reality Interface for UAV Operations in Near Earth Environments

In this chapter, an approach is developed to improve UAV pilot situational awareness that utilizes sensor packages common on most UAV systems. The approach uses an onboard camera and an inertial measurement unit to generate a mixed-reality chase view to the operator as seen in Figure 5.1. There are two methods presented to generate the mixed-reality chase viewpoint. The mixed-reality notion comes from the fact that the surrounding environment displayed to the pilot (outside of the onboard camera field of view) is a virtual representation. This surrounding environment can be created in real-time or prior to flight. In method one, the surrounding environment is created by a real-time mapping of features extracted from the onboard camera view. In method two, the surrounding environment is created using a prior model of the environment. A prior model could be constructed using geospatial digital terrain elevation data (DTED), satellite imagery, or prior manned or unmanned forward observer reconnaissance missions. For the chase view, the onboard camera images are still relayed to the pilot but are rotated to keep the horizon level and the perspective consistent with the displayed chase viewpoint. This view allows the pilot to see the entire aerial vehicle pose and surrounding environment as if they were following at a fixed distance behind the vehicle. The benefits of this viewpoint include an increased awareness of the extremities of the vehicle, better understanding of its global position in the environment, mapping of the environment, and a stable horizon (which helps to reduce the chance for vertigo).

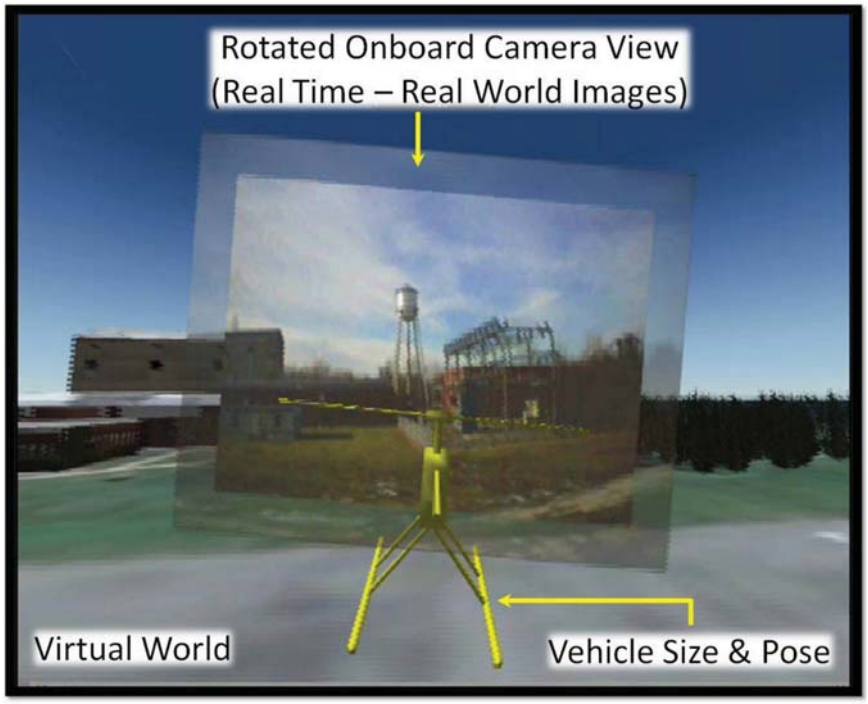

Figure 5.1: Screenshot of the graphical interface for the UAV pilot demonstrating the chase viewpoint during UAV operation in a near-Earth environment.

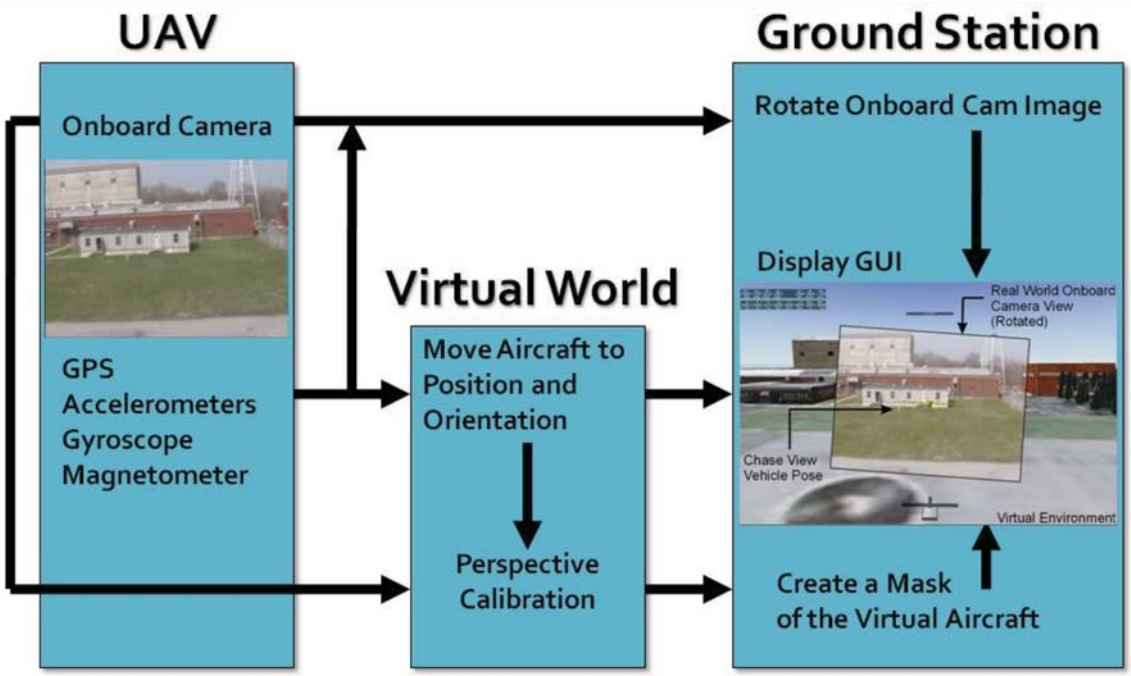

Figure 5.2: Diagram of the method used for generating a chase view.

5.1 Methods Toward Generating Chase View

Figure 5.2 shows the general methodology for generating the chase view in real time. On board the UAV, there is typically an inertial navigation system (INS) that outputs the real time location and orientation of the aircraft. This can include sensors such as an IMU, GPS, magnetometer, etc. The method for maintaining a level horizon requires counter rotating the onboard camera image based on the aircraft roll angle. The virtual world used to augment the field of view can be created in real time or *a priori* which is discussed in more detail in the following sections. Once a virtual world is established, the aircraft in the virtual world can be placed in the identical location and orientation as that of the real world based on the location and attitude information from the real world onboard sensors. Both the real world camera and the virtual world camera produce an image. These images are integrated together and the distance of the virtual camera from the virtual aircraft is adjusted until the surrounding virtual view matches the perspective of the onboard camera. Knowing the distance of the virtual camera from the virtual aircraft, data on the size of the aircraft (based on the perspective) can be extracted and the resulting avatar can be integrated into the GUI. The quality of the GUI is directly affected by the resolution and accuracy of the onboard sensor suites. Choosing an optimal sensor suite is important.

UAVs operating in urban and cluttered environments will most likely be limited to smaller back-packable and hand launchable vehicles that enable quick maneuvering and access to small spaces. With limited payload, choosing an optimal sensor suite can be difficult. The ultimate goal is to gather all data about the state of the vehicle and information from the surrounding environment using as few sensors as possible.

The advantage of Method one is that a map is created based on a very recent interaction with the environment and can be used without prior knowledge of the

operating area. It can also be adapted to work in areas without GPS availability by extracting vehicle state information using structure from motion methods. Method one comes at a cost of computation power, which limits the speed at which the UAV is allowed to fly in the environment. Method two allows for much faster flight as the environment is already mapped. Should the environment change, the pilot will be forced to mentally remap the surrounding environment during the flight using the onboard camera view.

5.1.1 Method One: Real Time Creation of the Environment

A chase viewpoint requires three dimensional measurements of the surrounding environment and accurate knowledge of the state of the vehicle. Researchers are currently working on methods to gather this information from only one onboard camera [70, 71] using Structure from Motion (SFM) methods. With this method, UAVs can be small and capable of map building in areas with no GPS signal. As these methods are currently computationally expensive, information from an onboard IMU, GPS, and camera is used toward developing the chase viewpoint. The technique for Method one is presented in the following sub sections.

Feature Detection and Tracking

Creating a map of the surrounding environment from the onboard camera view requires the extraction of three-dimensional information from multiple two-dimensional camera images. Features in each image must be identified and tracked from frame to frame. Following recommendations from the work of Shi et al. [72], a 7x7 feature detection window is used to calculate the spatial gradient matrix, H, as the window

moves across the image.

$$H = \Sigma \begin{vmatrix} (\delta I/\delta x)^2 & (\delta I/\delta x)(\delta I/\delta y) \\ (\delta I/\delta x)(\delta I/\delta y) & (\delta I/\delta y)^2 \end{vmatrix} \tag{5.1}$$

where I(x,y) is the gray level intensity and the summation is through the feature window. If the eigenvalues of H are greater than a chosen threshold, that particular area of the image is chosen as a feature point to track. Features were chosen based on the following criteria: they are the strongest features in the image, they do not overlap, and only a set number of features desired by the user are kept.

Tracking of the feature points is conducted using a pyramidal implementation of the Kanade Lucas feature tracker (KLT)[73]. The pyramidal implementation allows for much larger movement between two images. Currently this method uses a three level pyramid which can track pixel movement eight times larger than a standard KLT. In a traditional pyramidal KLT, feature points are chosen in the highest level of the pyramid. Using this method did not produce desired results. As such, the following modifications were made: features are detected on the highest resolution image which is currently at 640x480 (onboard camera resolution). A five by five gaussian blur is used before each re-sampling of the image down to the third level (80x60 resolution). The centroids of the chosen features are mapped to the location on the third level. For frame J to K, the previous and current onboard camera image respectively, the following calculations take place over ten iterations:
First an image difference $\delta I(x, y)$ is calculated:

$$\delta I_i(x, y) = J^L(x, y) - K^L(x + g_x^L + \nu_x^{i-1}, y + g_y^L + \nu_y^{i-1}) \tag{5.2}$$

where for level three (L = 3), the initial guess g_x, g_y is zero and the iteration guess

$\nu^0 = (0, 0)$. Then the image mismatch vector b_i is calculated for the feature window:

$$b_i = \Sigma \left| \begin{array}{l} \delta I_i(x, y) I_x(x, y) \\ \delta I_i(x, y) I_y(x, y) \end{array} \right| \tag{5.3}$$

The optic flow η^i is calculated:

$$\eta^i = H^{-1} b_i \tag{5.4}$$

And the guess for the next iteration becomes:

$$\nu^{i+1} = \nu^{i-1} + \eta^i \tag{5.5}$$

After the iterations are complete the final optic flow d^L for the level is:

$$d^L = \nu^{10} \tag{5.6}$$

The guess for the next lower pyramidal level g_x, g_y becomes:

$$g_x, g_y = 2(g^{L-1} + d^L) \tag{5.7}$$

And the process repeats until the final level (L^0), the original image, is reached. The final optic flow vector d is:

$$d = g^0 + d^0 \tag{5.8}$$

And the location of the tracked feature on image K is:

$$K(x, y) = J(x, y) + d \tag{5.9}$$

The tracking (50 features) is at sub pixel resolution and is currently running at ten

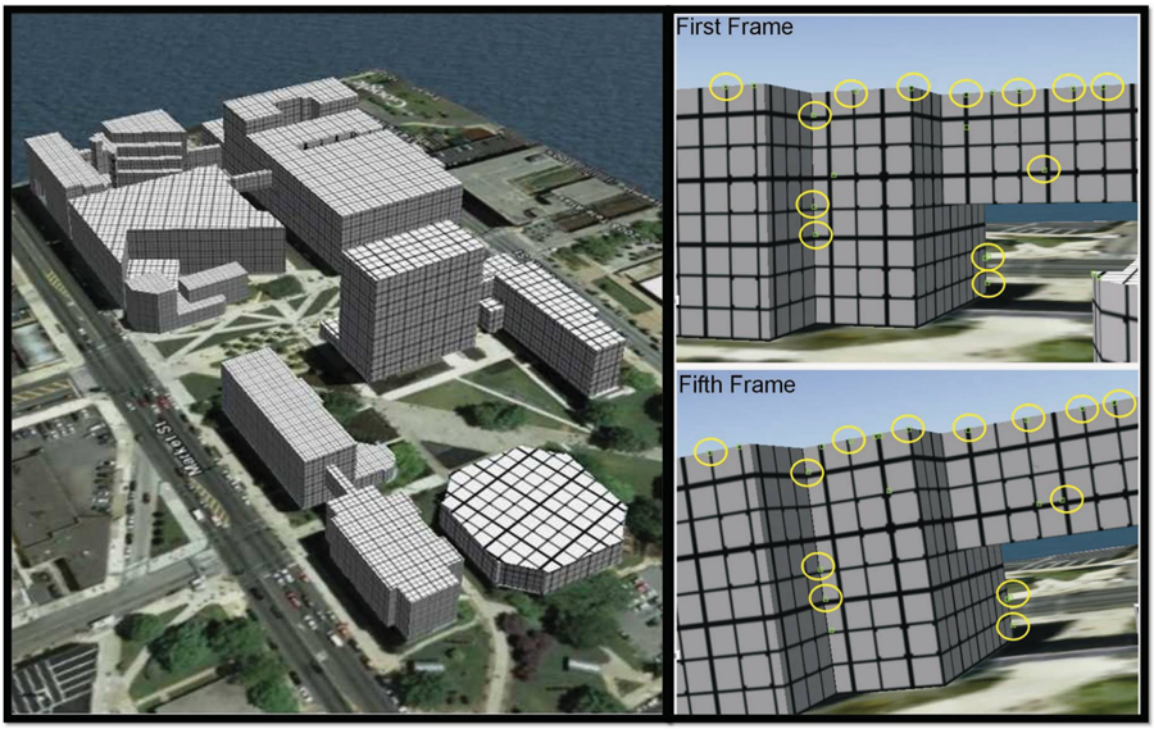

Figure 5.3: Left: Flight environment (Drexel University Campus) created in the virtual world for testing feature tracking and reconstruction. Initial textures were of grid patterns for easier development during initial stages. Right: Feature tracking across multiple frames. Features detected are surrounded by a small yellow box. The tracked features used in reconstruction are highlighted in this figure by yellow circles for better visualization. The screen captures contain a rotated view (aircraft is rolling) side of a building at Drexel. The texture of the walls were created with a grid pattern for easier feature detection/tracking during initial development.

frames per second on a 2.33 Giga Hertz dual core machine.

Reconstruction and Mapping

During the initial development stages, a simulated environment was modeled in the virtual world as seen in Figure 5.3. Since an IMU and GPS along with the camera are used, structure from motion methods are not needed and the three dimensional locations of the feature points can be found through triangulation. The extrinsic parameters for the camera are extracted from GPS and IMU measurements in the

X-Plane simulation. The intrinsic parameters of the camera are calculated prior to the tests using multiple images of a known grid pattern. Calibration tests found the focal length for the camera in the X-Plane environment to be 320.5 mm. Each feature point is stored in its initial frame and then tracked. If the feature point is successfully tracked for five frames, as seen in Figure 5.3, it is used in the reconstruction algorithm. The five frame difference was chosen to allow a greater distance between the two camera images before reconstruction is performed. The global frame of reference is chosen such that the axes lie on the latitude (Y), longitude (X) and altitude directions (Z) of the simulated environment. The origin of the axes are located in the simulated world where the vehicle is initially spawned. The distance to the aircraft camera from the global reference frame is calculated from GPS and IMU values. Locations of feature points in the camera image plane are transformed to the global reference frame using the following rotation and translation matrices:

$$\begin{aligned}
R_{1,1} &= cos(\alpha)cos(\gamma) - sin(\alpha)sin(\beta)sin(\gamma) \\
R_{2,1} &= sin(\alpha)cos(\gamma) + cos(\alpha)sin(\beta)sin(\gamma) \\
R_{3,1} &= -cos(\beta)sin(\gamma)
\end{aligned}$$

$$\begin{aligned}
R_{1,2} &= -sin(\alpha)cos(\beta) \\
R_{2,2} &= cos(\alpha)cos(\beta) \\
R_{3,2} &= sin(\beta)
\end{aligned} \tag{5.10}$$

$$\begin{aligned}
R_{1,3} &= cos(\alpha)sin(\gamma) + sin(\alpha)sin(\beta)cos(\gamma) \\
R_{2,3} &= sin(\alpha)sin(\gamma) - cos(\alpha)sin(\beta)cos(\gamma) \\
R_{3,3} &= cos(\beta)cos(\gamma)
\end{aligned}$$

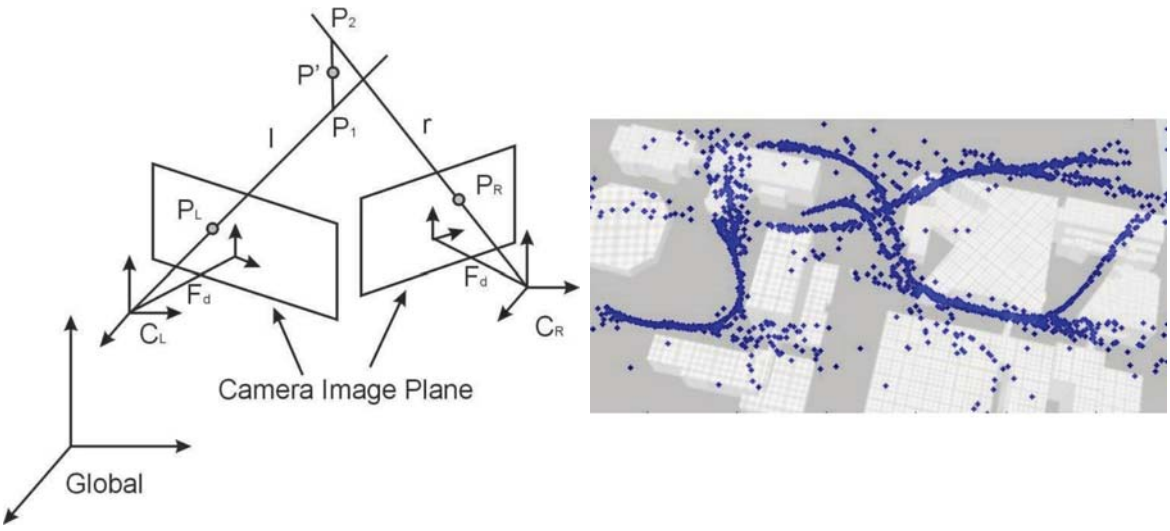

Figure 5.4: Left: Camera reconstruction geometry. Due to noise in the measurements, rays passing through the feature in the first and second camera image plane may not intersect. The midpoint of the closest point between the two rays is taken as the feature measurement. Right: Top down view of raw (non-filtered) reconstruction of feature points with flight environment overlayed over the data. Most data points far away from building edges are points reconstructed from features detected on the ground.

$$T = \begin{vmatrix} F_d cos(\beta) sin(\alpha) + Lon. - Lon.ofOrigin \\ F_d cos(\beta) cos(\alpha) + Lat. - Lat.ofOrigin \\ F_d sin(\beta) + Alt. - Alt.ofOrigin \end{vmatrix} \tag{5.11}$$

where α is the camera heading angle, β is the camera pitch angle, γ is the camera roll angle, and F_d is the camera focal length.

Reconstruction proceeds as follows:

Following Figure 5.4, the line running through the camera frame, C, and the feature point, P, in the image plane, to the feature point in the global frame is:

$$l = C_L + a(P_L - C_L) \tag{5.12}$$

$$r = C_R + b(P_R - C_R) \tag{5.13}$$

where a and b are values between 0 and 1 representing the length of vectors l and r respectively between C and P.

Ideally the two lines would intersect at the global location of the feature point, P, but due to noise in the measurements, they may not intersect. Therefore, it is determined that the feature point lies in the midpoint, P', between the line segment that is perpendicular to both of the rays.

$$P_1 = C_L + a_o(P_L - C_L) \tag{5.14}$$

$$P_2 = C_R + b_o(P_R - C_R) \tag{5.15}$$

$$P' = P_1 + 1/2(P_2 - P_1) \tag{5.16}$$

where a_o and b_o represent the values of a and b where the line P' crosses the l and r vectors respectively.

The orthogonal vector, w, to both lines, l and r, is:

$$w = (P_L - C_L) \times (P_R - C_R) \tag{5.17}$$

Therefore, the line going through P_1 to P_2 is:

$$P_2 = P_1 + c_o w \tag{5.18}$$

The unknowns a_o, b_o, c_o are found by solving the following equation:

$$a_o(P_L - C_L) - b_o(P_R - O_R) + c_o w = C_R - C_L \tag{5.19}$$

Currently the method is run without any filtering of the data so the results are noisy as seen in Figure 5.4. The method up to this point runs at approximately six frames per

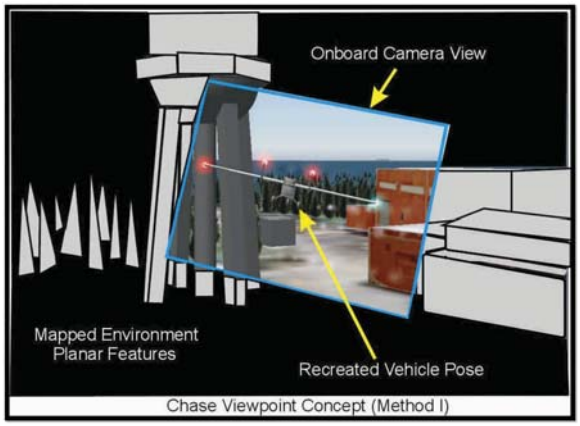

Figure 5.5: Conceptual graphic showing the chase viewpoint during UAV operation in a cluttered environment using Method I

second on a 2.33 GHertz dual core Windows laptop. The minimum desired operation speed is ten frames per second. The following steps presented describe how Method One would continue if the minimum frame rate is met.

Adapting a method similar to that presented by Watkins et al. [74] a three dimensional map of the environment can be created from a single camera viewpoint. This map can then be used in the chase view perspective of the UAV pilot. What the authors of [74] do differently from a number of single camera map making algorithms is that they merge feature points into planar regions for use in SLAM. The benefit is that it dramatically reduces the number of stored feature points needed to create a map. Much of urban terrain contains rectangular buildings. Therefore, many detected features can be turned into planar regions that represent building walls and rooftops. Once the mapping is complete and the planar regions have been represented by computer graphics (OpenGL), the chase viewpoint can be generated by integrating the UAV onboard camera view with the UAV perspective of the generated map. This concept can be seen in Figure 5.5. This method of generating the chase view allows for a current map of the environment to be relayed to the operator at the expense of

high computation requirements and limited flight speed.

5.1.2 Method Two: Pre-Built Environments

As stated earlier, Method Two requires much less computation during the flight as the operating environment is modeled prior to the flight. Again, one can easily generate such models from DTED data, satellite imagery and forward-observer reconnaissance. The details of this method are similar to techniques detailed in Chapter 3. There are a number of applications, such as surveillance or border patrol, where the environment will stay relatively static which makes Method Two valid. Aircraft position in the modeled environment is updated in real time using position data from the real world aircraft. The onboard camera view is rotated based on the roll angle received from the onboard IMU and is also surrounded by the simulated environment. An avatar of the aircraft is overlayed on top of the GUI, its size matching the perspective of the environment. This perspective is found by adjusting the virtual camera tether length behind the aircraft until the virtual environment correctly matches with the real world camera image. This "calibration" only needs to be done once for each camera system used.

To obtain the chase view images from the virtual world the following math operations are conducted. The reference frames used in the calculations are shown in Figure 5.6.

$$\begin{bmatrix} C_X \\ C_Y \\ C_Z \end{bmatrix} = \begin{bmatrix} R_{3x3} \end{bmatrix} \begin{bmatrix} C_{X'} \\ C_{Y'} \\ C_{Z'} \end{bmatrix} + \begin{bmatrix} P_X \\ P_Y \\ P_Z \end{bmatrix} \tag{5.20}$$

$$\begin{bmatrix} R_{3x3} \end{bmatrix} = R_{psi} R_{theta} = \begin{bmatrix} c(\psi)c(\theta) & -c(\psi)s(\theta) & s(\psi) \\ s(\theta) & c(\theta) & 0 \\ -s(\psi)c(\theta) & s(\psi)s(\theta) & c(\psi) \end{bmatrix} \tag{5.21}$$

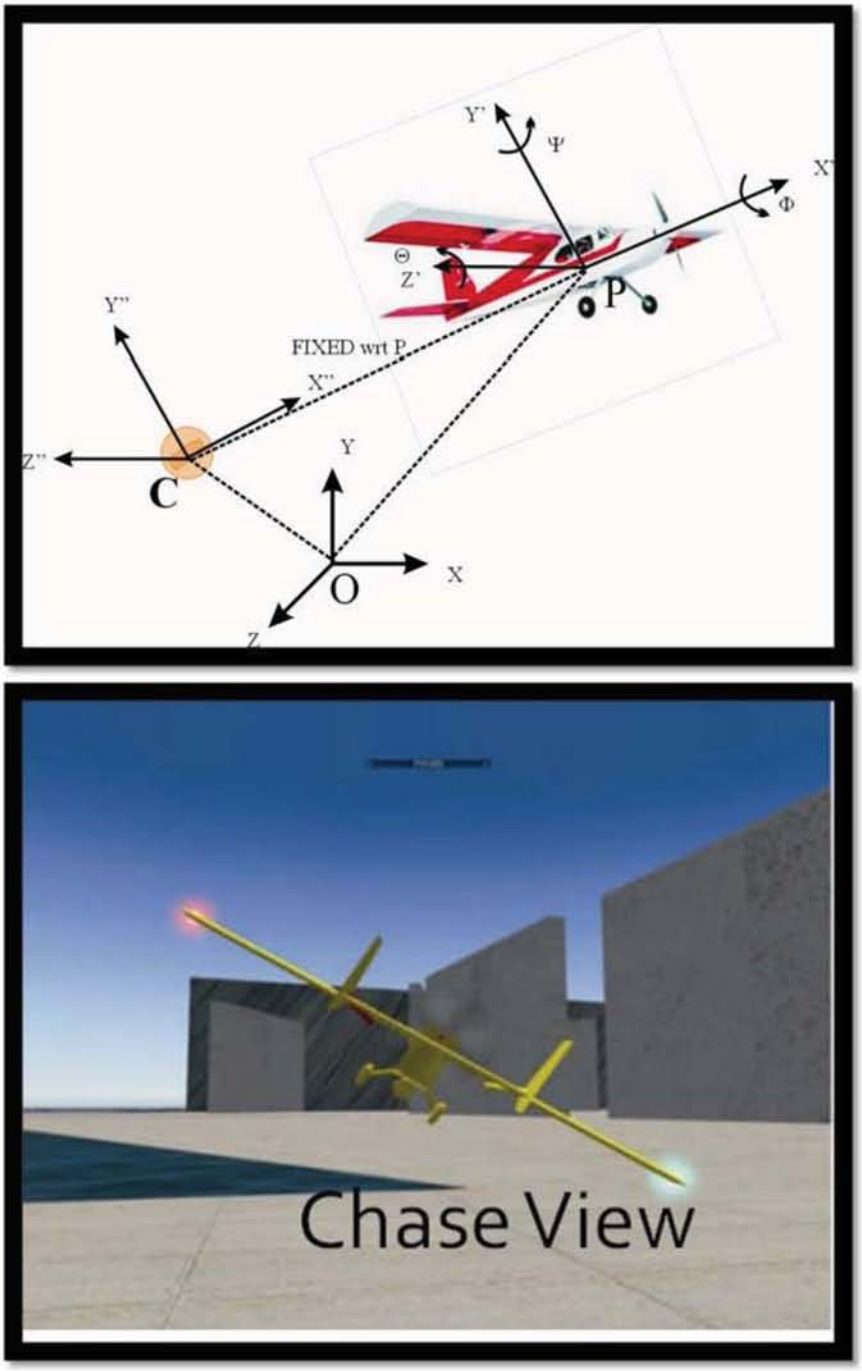

Figure 5.6: Top: Reference frames used for generating a chase view in the virtual world; Bottom: Example of the simulated world chase view.

$$\begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} 0 \\ Pitch_{aircraft} \\ Yaw_{aircraft} \end{bmatrix} \tag{5.22}$$

"O" is the global reference frame for the flight simulator with X, Y and Z coordinates. "P" represents the aircraft reference frame with X', Y', and Z' coordinates. "C" represents the camera reference frame with X", Y", Z" coordinates. The variables "*c*" and "*s*" correspond to cosine and sine respectively. As mentioned earlier, the tether distance CP in Figure 5.6 is adjusted until the virtual view matches the perspective of the onboard camera images. Once this tether distance is found, it is locked and stays that same distance with respect to frame P of the plane. The chase view roll angle stays zero throughout the flight regardless of the aircraft roll angle to ensure that the horizon stays level.

6. Exploratory and Development Stages Using SISTR

To test and evaluate pilot performance using the chase view interface, a series of experiments were developed to assess pilot skills operating in a cluttered environment using an onboard camera view and the chase view. For safe execution, each of these experiments utilized the indoor testing and evaluation facility described in Chapter 3. The experiments are presented in three stages. The first stage is the exploratory stage. This stage represents initial efforts to assess any observed differences in pilot performance using each view. Results from the first stage help direct the development of the chase view interface further and identified variables of interest for additional studies. The second stage is the developmental stage. This stage represents a more in depth study based on the results and changes made after stage one. Specific variables of interest identified from stage one were evaluated. The results and findings also helped to further develop the chase view interface and helped formulate the hypotheses for stage three, the Human Performance and Assessment Stage (Chapter 7).

The ideal scenario for this study would be to have the actual environment built in real time from sensor data. Method One was presented as the work done toward that goal. However, results are noisy and the update rate is slow. Method Two eliminates those variables from the analysis by using environment information gathered prior to flight. In all stages, Method Two for generation of the chase view is used.

6.1 Exploratory Stage

This stage was designed to help assess any possible change in pilot behavioral data while using the chase view interface. The goal was to determine if the chase view system produced enough benefit to warrant further development. As an exploratory

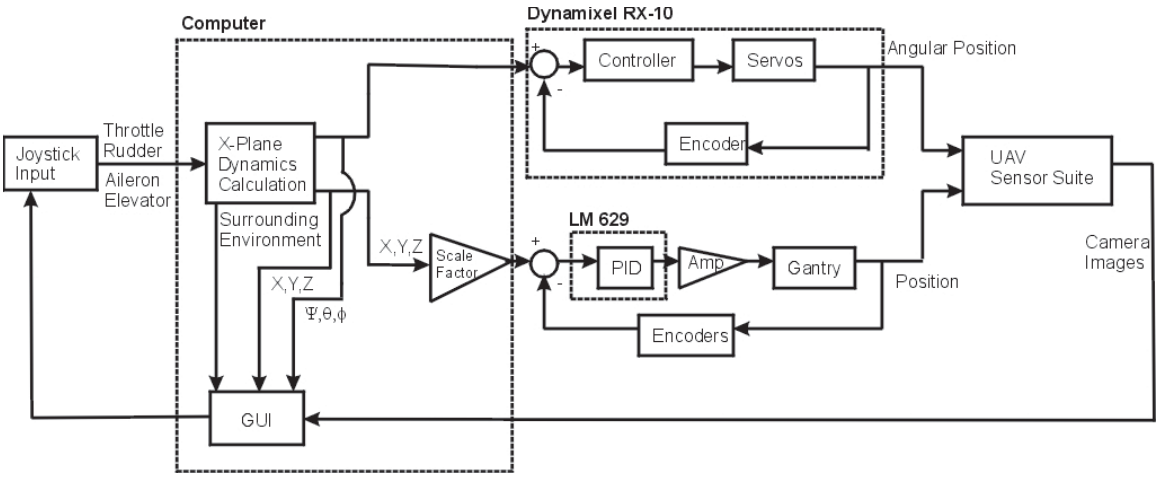

Figure 6.1: Block diagram of the experiment setup.

effort, a rigorous protocol and statistical analysis was not necessary as the results would be used to develop the hypotheses for future studies.

6.1.1 Experiment Setup

A block diagram of the experimental setup is shown in Figure 6.1. During the experiment, flight commands are input into the flight simulator by the subject via a joystick. The flight sim generates and sends the resulting translational and angular positions of the aircraft through UDP to the SISTR controller. Differently from what was presented in Chapter 3, the flight sim is also used in the chase view experiments to render the surrounding virtual view to the rotated onboard camera image as seen in Figure 6.1.

For these tests, a model of the UAV Mako, as seen in Figure 3.1 was used. For safety reasons, the simulated version of the Mako was modified so it had a lighter weight with less horsepower effectively decreasing its cruise speed to 45 miles per hour in the simulation which corresponds to 9 inches/second in SISTR motion at 1:87 (H0) scale.

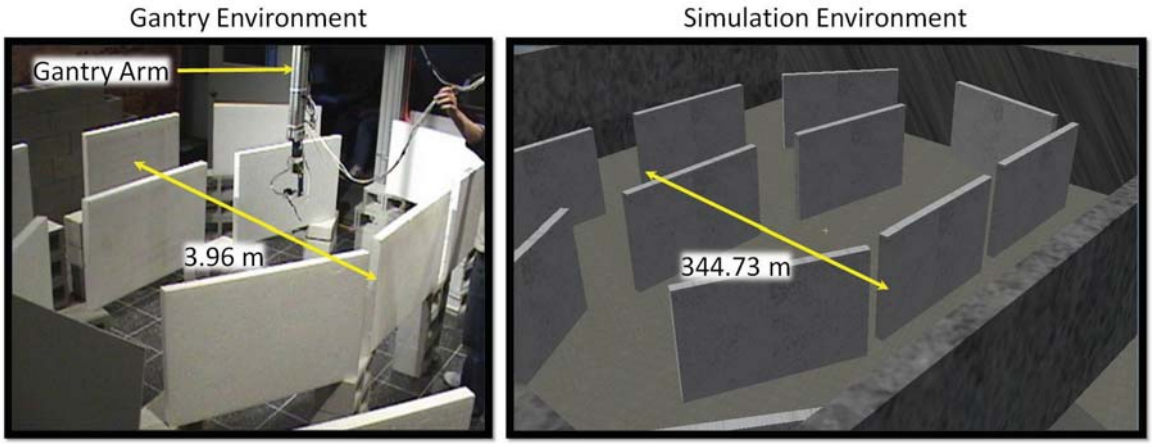

Figure 6.2: Comparison showing the real world scale flight environment with the H0 scale (1:87) SISTR environment. The white gates create narrow corridors representative of flight between large buildings in an urban environment. Left: Gantry environment 1:87 scale. Right: Simulated full scale flight environment.

To match the size of a reasonable real world UAV test environment, SISTR's workspace represented an H0 scale environment as seen in Figure 6.2. The flight environment consisted of corridors that can be representative of corridors between large buildings in an urban environment. The environment consisted of white foam boards with large gaps between each board. The walls were raised because the limitations of SISTR prevent the end effector from moving closer than 2 feet to the ground. This produced an environment with an imaginary floor.

A standard web camera (approximately 40 degree field of view) was used to represent the video feed from onboard the aircraft. The camera itself was attached to the series one YPR unit shown in Figure 3.10.

User Interface

The user interface was created using Visual C#. The program handled the visual presentation to the user and also the communication between the flight simulator and SISTR. The program collected translational and angular position data from the flight

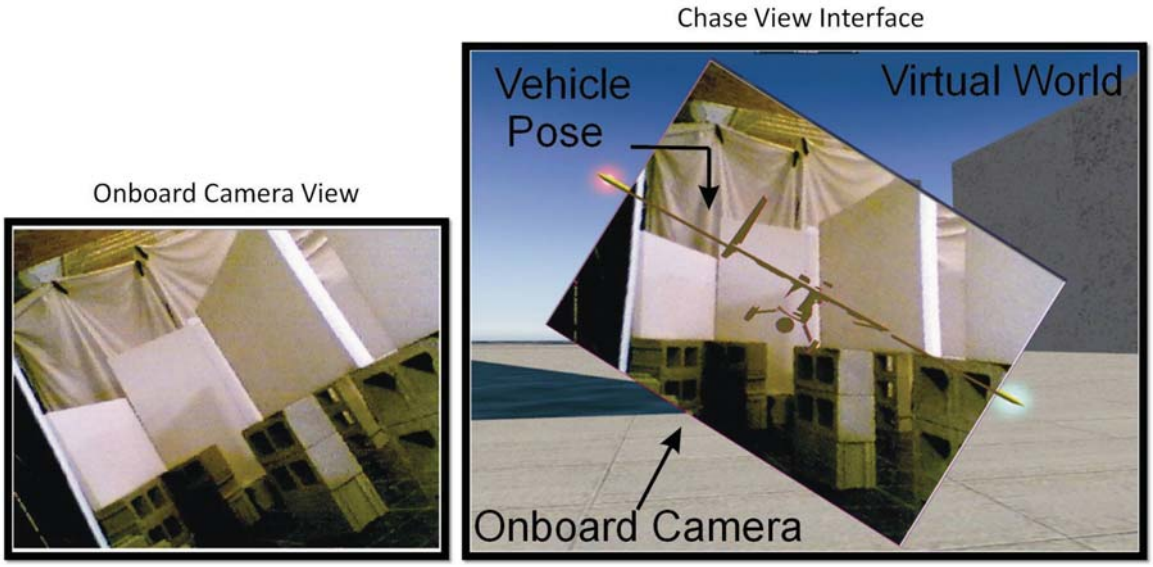

Figure 6.3: Left: Onboard camera view capture during H0 scale flight tests. This shows a view of the corridor environment during a turn maneuver by the aircraft. Right: Chase view interface during H0 scale flight tests.

sim, converted it to H0 scale and transmitted it through UDP to SISTR at 20 Hertz. During onboard camera tests, only the onboard camera view was shown to the pilots during flights through the environment as seen in Figure 6.3. During the chase view tests, the program displayed three main items to the pilot in real time:

1. Rotated onboard camera view so the horizon stays level
2. Virtual view of the surrounding environment based on aircraft location and prior model of the environment
3. Virtual representation of the aircraft pose to scale with the onboard camera view and surrounding environment

6.1.2 Procedure

Seven subjects were used for initial validation of the chase view concept. Each subject varied in flight simulator experience from no experience to five years worth

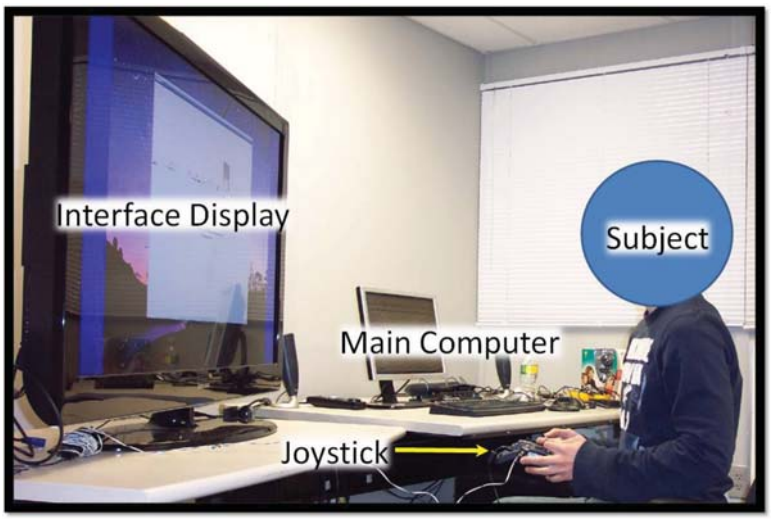

Figure 6.4: Subject operating setup

of recreational use. Prior to the tests, subjects were given time to fly the Mako in an open virtual environment using the flight simulator under both simulated onboard camera view and simulated chase view. This allowed them to become familiar with the controls and to get a feel for the response and size of the aircraft. When the subjects felt comfortable with the controls, the experiments began. As seen in Figure 6.4, the subjects were placed in a room, separated from the experiment environment, with a 52" monitor from which to view the user interface. Subjects underwent multiple tests where they flew the aircraft from an onboard camera view and a chase view. During onboard camera tests, the subjects were shown only the raw view from the camera and were asked to fly through the corridors of the environment while keeping a safe distance from the walls and keeping the aircraft as stable as possible. During the chase view tests, the subjects were shown the chase view and asked to fly through the corridors with the same emphasis on safe distance and stability. During each test, aircraft translational and rotational positions were recorded. If the subject crashed into the corridor walls, they were asked to continue their flight through the

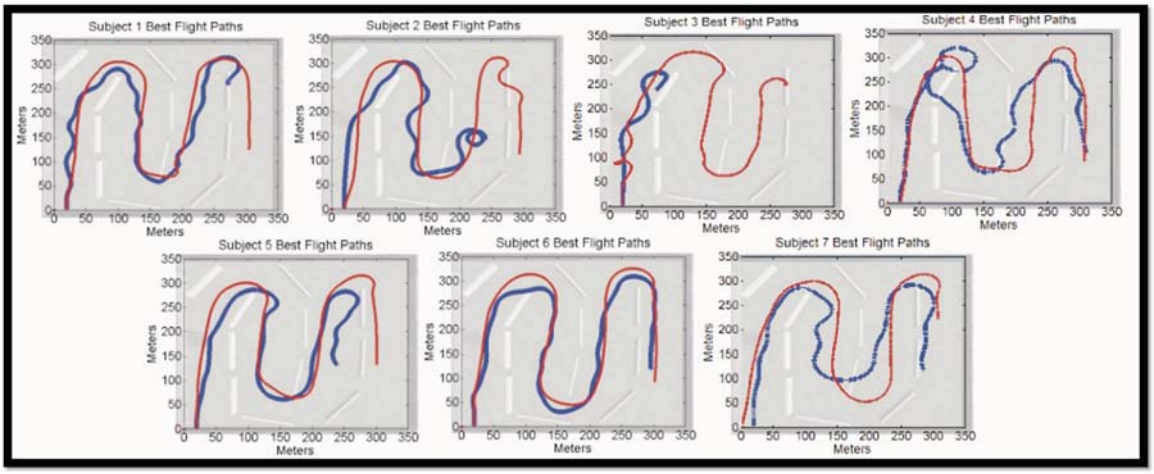

Figure 6.5: Top down view of the subjects best flight paths achieved using the onboard camera view (blue) and chase view (red). The flight environment is superimposed over the data.

corridors so data collection could continue. The walls of the SISTR environment were designed to easily collapse under contact. After each test, subjects were asked about their thoughts on the different modes of operation and how they felt it affected their performance.

6.1.3 Results and Discussion

Figure 6.5 shows the best flight paths out of all the tests achieved for each subject using a chase view and using an onboard camera view. The best flight was chosen by the following criteria: visually inspected straightness of the flight path, visually inspected distance from the obstacles, and farthest reached point in the environment. The data showed an improvement across all subjects leading to the conclusion that chase view does have a positive effect on the performance of pilots. However, the experimental setup itself added variables that may have affected the results. While using the onboard camera view, subjects showed much more oscillations in both translational and angular positions than when compared with the position results

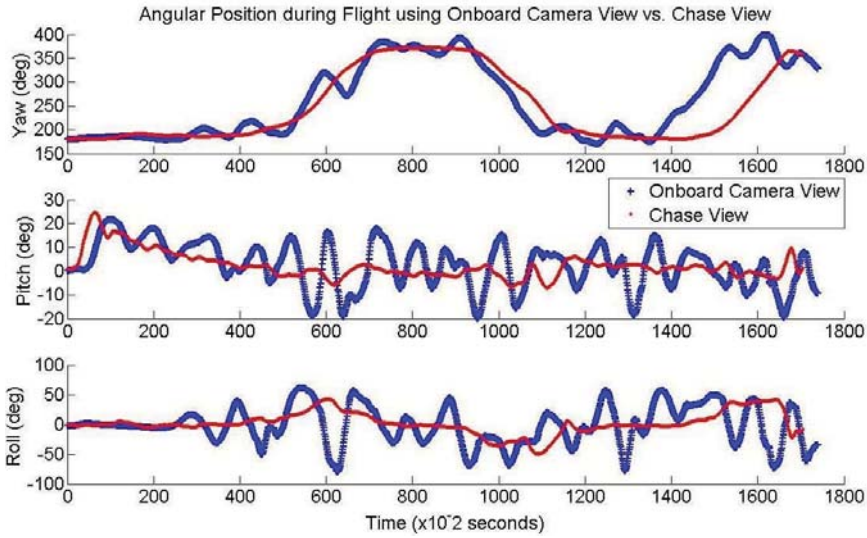

Figure 6.6: Example data of the aircraft angular positions during an onboard camera and chase view test from a single subject. The thicker blue line represents angles achieved using the onboard camera view and the thinner red line represents the angles achieved using the chase view.

using the chase view interface. This was attributed to two issues. During the onboard camera view tests the small field of view of the onboard camera would require subjects to continue to turn back and forth to bring the walls into view. This technique helped to gather enough information to establish their position in the environment. The second issue was the vibrations caused by the movement of the gantry arm and the cantilever design of the YPR unit. The vibrations caused subjects using the onboard camera view to overcompensate in their input commands leading to increased oscillations in the flight. The oscillations in the angular position can be seen by Figure 6.6 showing comparison data from one subject. Since the surrounding virtual view is immune to vibrations in the system, it did not shake, nor did the avatar of the aircraft. This resulted in a reliance on the virtual information during periods of high vibration which ultimately improved performance over the onboard camera view.

Discussions with the subjects after the tests revealed that the chase view system resulted in a better personal sense of awareness of the vehicle extremities and the aircraft's position in the environment. However, because the floor of the real world environment did not match the floor level of the virtual environment used in the surroundings for the chase view, the subjects at times were distracted and confused about the true height of the aircraft.

The results from the Exploratory Stage supported efforts to continue development of the chase view. Modifications needed to be made to eliminate factors such as vibrations and "virtual floors" from affecting the performance of pilots. The next section, Development Stage, presents those modifications and a further study.

6.2 Development Stage

This stage was developed to produce and refine the hypotheses for the chase view interface in near Earth UAV operations. Based on the findings from the Exploratory Stage, a number of modifications were made.

6.2.1 Experimental Setup

The block diagram of the setup and the overall flow of data is the same as the Exploratory Stage shown in Figure 6.1. Modifications were made to the YPR unit to eliminate the cantilevered design as seen in Figure 3.10 right. Along with increasing the rigidity of the YPR Unit, the gantry motion controller was also modified to produce smoother motions. These changes dramatically decreased the vibrations experienced at the end of the gantry arm.

The camera was changed to a commercially available wireless camera system with a 90 degree field of view. This camera is more representative of the type of cameras used on small UAV systems and is the actual camera used on board the UAV during

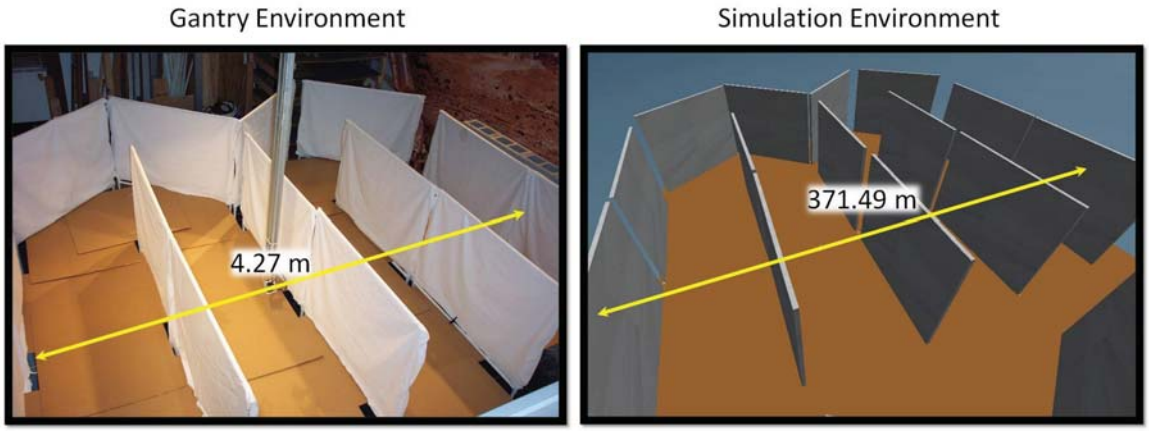

Figure 6.7: Left: Gantry environment built at 1:87 scale. Right: Simulated full scale replication of the flight environment.

field flights described in Chapter 8.

The environment was changed to focus on specific flight scenarios. As seen in Figure 6.7 the environment still consisted of corridors but was designed such that there were specific sections of straight flight, and specific sections requiring turning maneuvers Figure 6.8. Also added was a raised cardboard floor to match the lower limits of the gantry arm, eliminating the need for subjects to imagine a virtual ground.

User Interface

Modifications to the chase view interface were made to adjust for the wider field of view of the new camera system. The output of the camera was still at 640x480 resolution but due to the wider field of view, the virtual camera field of view had to be changed to match. This essentially increased information seen in the surrounding view as compared with the Exploratory Stage. Figure 6.9 shows the onboard camera view and the chase view interface used in these experiments.

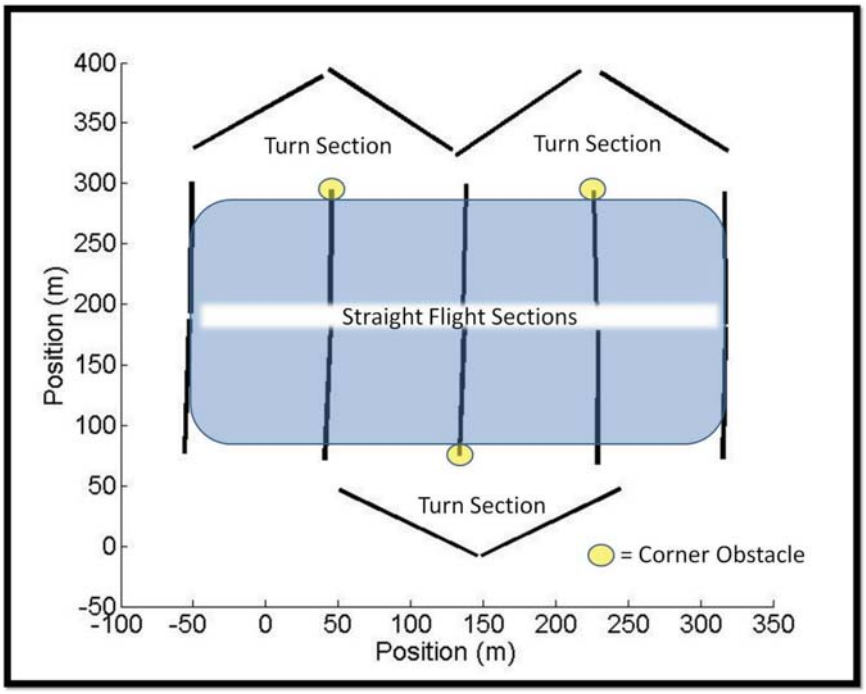

Figure 6.8: Top down view of the flight environment broken into a series of straight flight and turning sections.

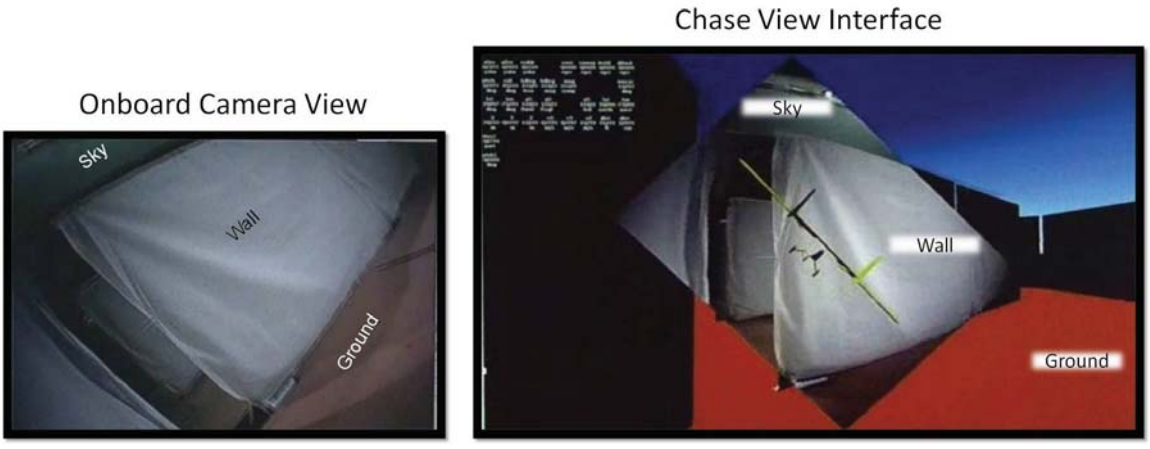

Figure 6.9: Left: Onboard camera view during a turn maneuver. The ground, corridor wall and sky are highlighted. Right: Chase view interface during the same turn maneuver.

Table 6.1: Mean Obstacle During Straight Corridor Flight

Subject	Chase View (m)	Onboard View (m)
1	37.44 ± 6.63	37.83 ± 6.95
2	34.77 ± 8.09	33.57 ± 6.92
3	37.57 ± 5.99	37.76 ± 5.21
4	30.88 ± 9.33	31.42 ± 7.71
5	33.46 ± 8.63	34.89 ± 8.14

Procedure

There were no major differences between the procedure in this stage and the one presented for the Exploratory stage. Five subjects different from the Exploratory Stage but with similar flight sim experience were also used.

6.2.2 Results and Discussion

For each flight, aircraft translational and rotational positions, velocities, and accelerations were recorded. The data for each subject was separated into the appropriate straight flight scenario and turning scenario. The minimum distance from the surrounding walls was also calculated at each position during the flight and separated in the the two flight scenarios.

Straight Flight

There was no difference among the subjects when using chase view and using onboard camera view analyzing the mean distance of the aircraft from the side walls (Table 6.1). The larger field of view of the onboard camera eliminated the need for subjects to move from side to side to establish awareness of the aircraft position.

The magnitude of the angular velocities during the straight section, shown in

Table 6.2: Mean Magnitude Angular Velocity During Straight Corridor Flight

Subject	Chase View (m/s)	Onboard View (m/s)
1	16.94 ± 24.77	20.48 ± 27.38
2	19.20 ± 18.72	18.70 ± 17.82
3	11.47 ± 11.89	11.17 ± 11.96
4	26.31 ± 42.70	31.37 ± 32.53
5	17.03 ± 21.48	13.94 ± 15.07

Table 6.2, gives a snapshot of how much movement there was during the straight flight sections. Two subjects (Subject 1 and Subject 4) produced higher average angular velocities during the straight away section while using the onboard camera view, while two subjects (Subject 2 and Subject 5) produced higher average angular velocities using chase view. Subject 3 showed little change in either one. The higher field of view of the onboard camera and the decrease of vibrations from the Exploratory Stage has helped to improve control during the onboard camera view trials. These results lead toward the conclusion that smoothness of the flight for each view is subject dependent. A larger data set would be needed to further assess pilot performance during straight flight.

Turn Sections

Table 6.3 shows that across all subjects, the mean obstacle distance during the turn portion was lower for chase view compared to onboard camera view. While chase view produced a closer distance to the corner obstacle, a value of 30.5 meters from the obstacle is well within the safe distance for an aircraft with a wingspan of approximately 4 meters. These results seems to show that while using chase view, the subjects had better awareness that the aircraft was clear of obstacles sooner (the corner obstacle is highlighted in Figure 6.8) and could take the turn tighter. This

Table 6.3: Mean Obstacle During Turn Section

Subject	Chase View (m)	Onboard View (m)
1	31.08 ± 11.76	37.59 ± 9.16
2	32.68 ± 12.43	39.56 ± 12.85
3	44.22 ± 10.74	47.83 ± 10.00
4	35.80 ± 20.25	39.59 ± 13.38
5	30.57 ± 13.40	33.70 ± 10.62

Table 6.4: Mean Magnitude Angular Rate During Turn Section

Subject	Chase View (m/s)	Onboard View (m/s)
1	64.55 ± 48.76	61.60 ± 45.17
2	40.88 ± 24.34	41.52 ± 30.09
3	37.21 ± 19.80	30.00 ± 16.21
4	61.73 ± 55.45	47.21 ± 31.11
5	52.01 ± 34.09	47.18 ± 27.42

assumes that the subject was using the same personal metric for "safe distance" during chase view flight as they did during onboard camera view.

Table 6.4 shows the mean magnitude angular velocity during the turning section for each subject. The mean angular rate was higher for four out of the five subjects with Subject 2 having close to the same mean angular rate in both chase view and onboard camera view. This higher angular velocity is a result of the decrease in turning radius (tighter turn).

6.2.3 Formulation of the Hypotheses

The results from the Exploratory and Development Stages have led to the following hypotheses:

Hypothesis 1

The chase view interface will produce greater awareness of the aircraft extremities over the traditional onboard camera view. This can be demonstrated by a closer (while still safe) distance with tighter turns around obstacles.

Hypothesis 2

During straight flight, chase view will help the pilot maintain a smoother flight resulting from seeing the aircraft pose in the image.

Hypothesis 3

The chase view interface will improve a pilot's understanding of the 3D spatial relationship of the aircraft and its surroundings. This can be demonstrated by a pilot's ability to fly directly over targets of interest and ability to notify when the vehicle is directly over a target of interest.

Hypothesis 4

Cognitive workload of the pilot will decrease using chase view. This is due to the stabilized camera image (horizon remaining level) and more of the environment displayed in the image. This will decrease the amount of processing the pilot needs to do for mental reconstruction of the environment and location of the aircraft within the environment.

The Human Performance and Assessment Stage was designed to test these hypotheses.

7. Human Performance and Assessment Stage

Exploratory and Development Stage results of the chase view system show pilot improvement over positioning of the aircraft as compared with a standard onboard camera view. These results support the efforts toward an extensive human factor study to validate the early claims. Human factor studies in general reveal a dizzying array of test issues, measurement methodologies, and analysis paradigms. This study is designed in collaboration with Drexel's Optical Brain Imaging team. The team has been one of the frontiers in use of the neuroimaging assessment tools, such as fNIR and EEG for the human performance and conducted many relevant research studies [14]. Typically in human factor tests, researchers consider what kinds of statements they want to make at the end of the tests (i.e. hypotheses). Then they develop the test measures necessary to test those claims.

In general, there are four broad categories that need to be represented in some degree to make sense of human factors results and to portray this information. These categories, seen in Figure 7.1, are Situation, Individual, Task, and Effect (SITE) [13]. The Situation category represents human factor issues that deal specifically with the environment in which the subject is placed for the experiments. Specifically, issues in this category address attributes of the operator setting such as software, hardware and environment conditions. The Individual category includes measures of the attributes of the individual users of the system such as user's experience and skills. It is also within this category that parameters such as the subject's cognitive workload and physical energy levels are addressed. The Task category addresses issues such as the accuracy of the subject's actions, the quality and speed of the performance, reaction times, and decision making. The final category, Effect, addresses issues on the consequences and effects of the overall results from the task category. Also

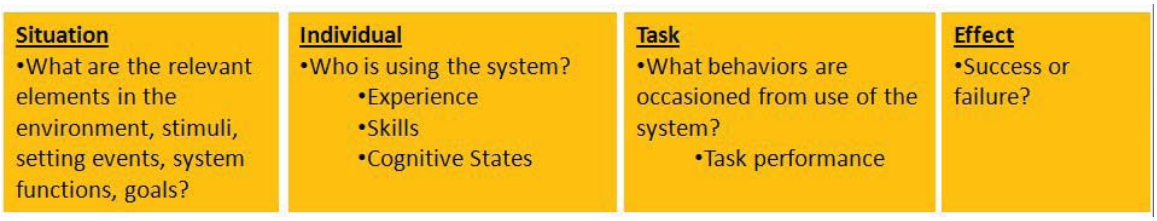

Figure 7.1: The SITE structure adapted from [13]. This represents the four categories which should be represented in some degree when conducting human factor tests.

included in this category is the evaluation of the user's assessment and satisfaction with the system.

The design of these human factor experiments was influenced in part by a collaboration with the Drexel Optical Brain Imagining Laboratory. The Optical Brain Imaging Laboratory was brought into this project because behavioral measures are not the only aspects important in the evaluation of a new piloting interface. Cognitive workload of the pilot plays just as an important role. If a pilot can perform well using the interface but requires a high level of mental processing to do so, they may not have a suitable level of mental resources available during the flight to safely handle unexpected events such as faults or warnings.

Current techniques in UAV training and pilot evaluation can be somewhat challenging for cognitive workload assessment. Many of these types of studies rely partly on self reporting surveys, such as the NASA Task Load Index (NASA-TLX). NASA-TLX was designed to reduce between-rater variability. The ratings are based on the demands imposed on the subject (Mental, Physical, and Temporal Demands) and the interaction of the subject with the task (Effort, Frustration, and Performance) [75]. The Task Load Index combines the subjects ratings of interaction with a weighted value of the demands. The demands are rated by importance to the subject on what he or she feels affected the work load level during the task. While this method does reduce between-rater variability, it doesn't eliminate it and it is also succeptable to

inconsistencies in the subject responses over a series of tests. The use of functional near-infrared (fNIR) brain imaging in these studies enables an objective assessment of the cognitive workload of each subject that can be compared more easily. The Drexel Optical Brain Imaging Lab's fNIR sensor uses specific wavelengths of light, introduced at the scalp. This sensor enables the noninvasive measurement of changes in the relative ratios of de-oxygenated hemoglobin (deoxy-Hb) and oxygenated hemoglobin (oxy-Hb) in the capillary beds during brain activity. Supporting research has shown that these ratios are related to the amount of brain activity occurring while a subject is conducting various tasks [14]. By measuring the intensity of brain activity in the prefrontal cortex, one can obtain a measure of the cognitive workload experienced by the subject [76, 77, 78]. Another added benefit is the design of the sensor itself which allows for ease in portability and enables the monitoring of subjects in actual or realistic environments. This is compared with other brain imaging modalities such as fMRI that require large specially designed rooms and minimal movement by the subject during tests [79].

As users of UAVs move toward newer and untested applications, data about operator cognitive workload and situational awareness become very important aspects of safe UAV operation. Low situational awareness requires higher cognitive activity to compensate for the lack of intuitive cues. Complex mission scenarios also inherently involve high cognitive workload. Adding some measure of brain activity to the selection, training, and operation of UAV pilots could greatly improve the resolution of any assessments involved therein. To that end, integration of the fNIR sensor into the Human Performance and Assessment Stage could produce an objective assessment of operator workload that can be used to enhance the self reported (subjective) workload, and help with further modifications to the chase view interface. Inline with Hypothesis 4, it is hypothesized that: *fNIR will detect a change in blood oxygenation*

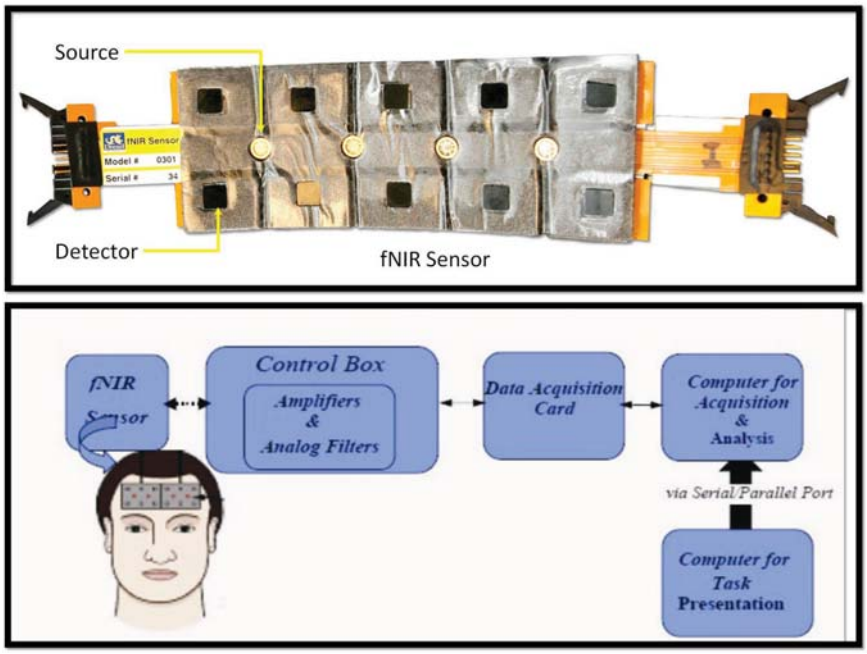

Figure 7.2: Top: fNIR sensor showing the flexible sensor housing containing 4 LED sources and 10 photodetectors. Bottom: fNIR Block diagram reprinted from [14]

(ie. cognitive workload) for Onboard camera view subjects that is significantly higher than Chase view subjects because of the increased mental mapping and prediction of aircraft position required due to the onboard camera perspective.

7.1 Experimental Setup

A majority of the experimental setup is the same as the setup described in the Development Stage (Chapter 6). Integration of the fNIR system, changes to the gantry environment, and changes to the chase view interface as well as the onboard camera interface are highlighted.

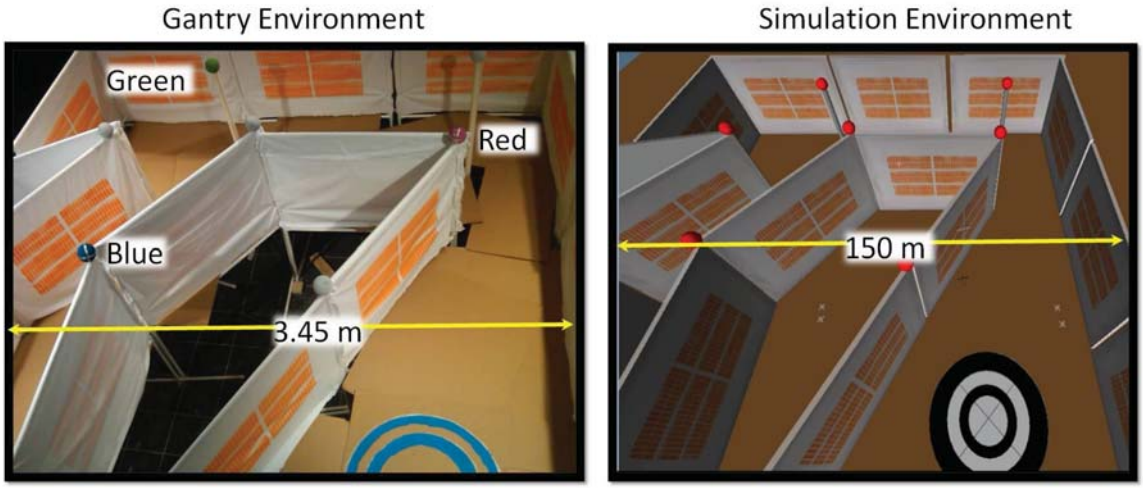

Figure 7.3: Left: Flight environment inside the gantry built at 1:43.5 scale. Highlighted in the image are the colored markers for the second level of the environment. Right: Simulated full scale environment.

7.1.1 fNIR

The fNIR sensor consists of four low power infrared emitters and ten photodetectors, dividing the forehead into 16 voxels. The emitters and detectors are set into a highly flexible rectangular foam pad, held across the forehead by hypoallergenic two-sided tape. Wires attached to each side carry the information from the sensor to the data collection computer. The components of the fNIR systems are seen in Figure 7.2.

7.1.2 Flight Environment

The gantry environment (Figure 7.3) consists of two flight levels. The lower level contains corridors and two tall pole obstacles. The upper level contains a series of colored spherical fiducials attached to the top of the corridor walls and obstacles. The physical workspace of the gantry environment is the same as in the Development Stage however the environment was built to half H0 scale (1:43.5) to allow for accurate

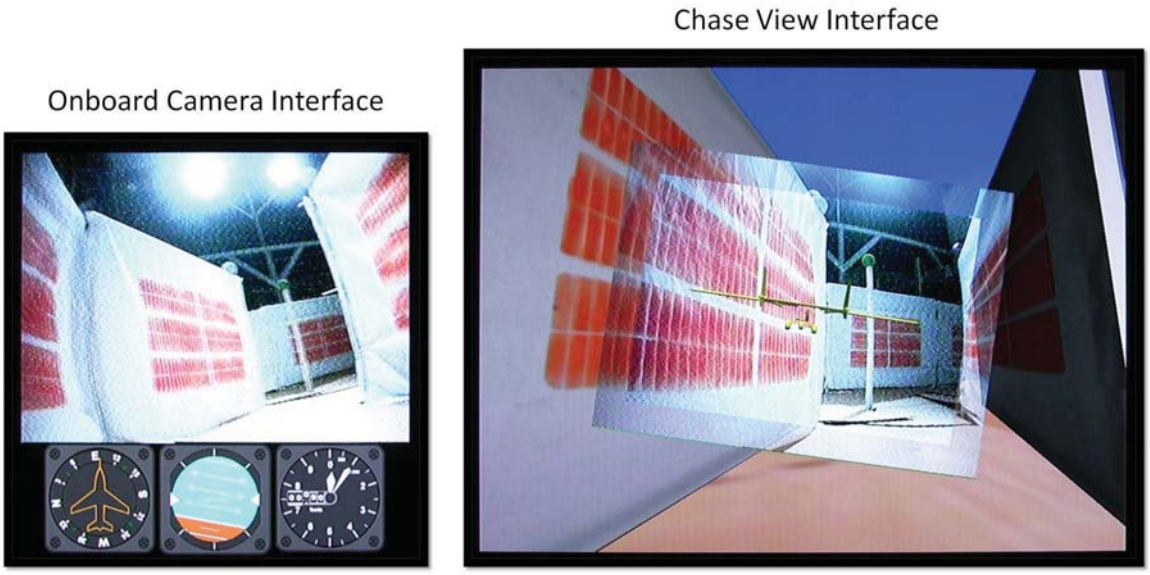

Figure 7.4: Left: Onboard camera view with virtual instruments positioned below the image to relay information about the vehicle state. Right: Chase view with alpha blended borders.

representation of the UAV wingspan with the width of the gantry end effector. Due to the temporal resolution of the fNIR sensor on the order of seconds, the environment was designed to continually require the pilot to update their path planning. The close quarters and multiple obstacles help to extract metrics during flights to test Hypotheses 1,2, and 4. The target and ball markers were added to help with testing Hypothesis 3.

7.1.3 Interface Modifications

Discussions with subjects from the Exploratory and Development Stages raised an issue about the border between the rotated onboard camera and the surrounding virtual image for the chase view interface. At times there was a high contrast between the border which distracted subjects and drew their attention to the border rather than the center of the interface. The new design for the chase view interface, shown

in Figure 7.4, addressed this issue with an added alpha blended border between the previous border of the rotated camera image and the surrounding virtual view. This helped to dramatically reduce the border contrast as well as increase subject immersion into the environment.

The onboard camera interface was modified to give a better representation of the information currently available to internal UAV pilots. Predator pilots have a heads up display superimposed onto the onboard camera images. This heads up display gives them a sense of the aircraft relative to the artificial horizon, bearing angle, and altitude. A generated heads up display integrated into the onboard camera image proved to be processor intensive for the approach taken in this thesis. As an alternative, the heads up display was replaced with virtual instruments as seen in Figure 7.4, similar to the instruments used on manned aircraft. These virtual instruments were placed directly below the onboard camera image, in clear view of the subject. The instruments displayed the aircraft relative to the artificial horizon, bearing angle, and altitude.

7.2 Procedure

A total of 12 subjects were used for these experiments, 1 female and 11 males. Subject 2 dropped out of the study after the second session so the data was not included into the analysis. Differently from the Exploratory and Development Stages, for these tests, the subjects were separated into two groups. Six subjects operated the aircraft using only the chase view interface (Chase view) and five subjects operated the aircraft using only the onboard camera interface (Onboard view).

All subjects were right hand dominant. No subject used video games for over six hours a week with six subject having no use of video games. No subjects had prior military training or manned aircraft training. More information about the subjects

Table 7.1: Subject Information and Prior Flight Experience. Number of subjects from each group is given.

Question	Chase Group	Onboard Group
Corrective Lenses	4	3
RC Aircraft Training	2	1
0 Hours Flight Sim	1	0
1-10 Hours Flight Sim	1	1
10-50 Hours Flight Sim	1	0
50-200 Hours Flight Sim	2	2
200+ Hours Flight Sim	1	2

and their prior flight sim experiences are found in Table. 7.1.

Each experiment session took approximately 45 minutes. There was a total of nine sessions, of which eight were actual flight sessions (the first is an intake/intro/consenting session). The fNIR sensor was placed on the participant's forehead during all eight flight sessions as seen in Figure 7.5. In all, 374 flights through the environment were recorded.

Before the beginning of each flight, an individual's cognitive baseline was recorded. This was a 20 second period of rest while the fNIR recorded oxygenation levels.

7.2.1 Session One

After the consenting process, each subject completed the Edinburg Handness questionnaire [80] and a brief questionnaire regarding previous flight and video game experience. After filling out the forms, the subjects had a a fifteen-minute introduction and free-flight session to get familiar with the dynamics of the aircraft and the flight controller. Differently from prior stages, subjects flew through the actual experiment environment (using SISTR) and their appropriate interface (chase or onboard camera view). After the free flight, subjects were given a small questionnaire to rate their

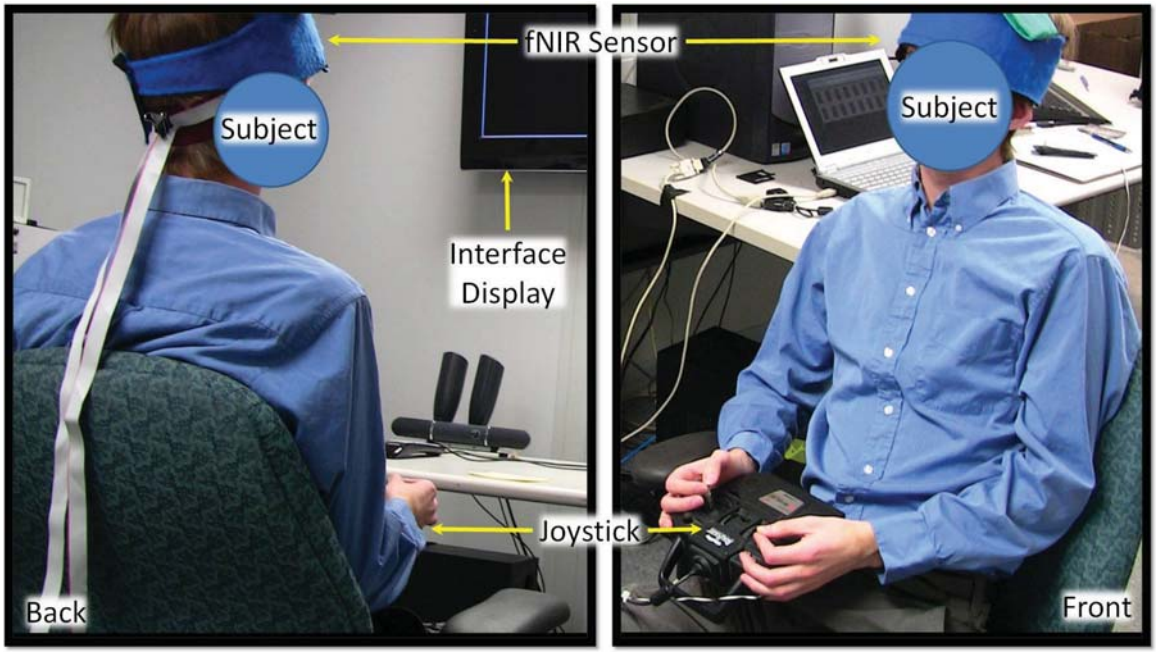

Figure 7.5: Subject operating environment. The fNIR sensor is shown strapped to the forehead of the subject with a blue felt cover to block ambient light.

confidence during the session (Appendix A.1).

7.2.2 Sessions Two through Nine

During each of these sessions (two through nine), the subjects conducted four flight trials. Each trial represented a different flight path to follow through the environment as well as a different marker setup for the second level. The four flight paths can be seen in Figure 7.6. An example of the marker setup can be seen in Figure 7.3 where the subject is required to fly over the blue marker, then the red marker and finally the green marker. All four paths were flown during each session but were presented to the subject in random order. The marker setup was also presented in random order, however there was a total of 20 possible marker combinations.

During the flight sessions, subjects had four goals. The first goal was to fly through the test environment while maintaining a safe distance from the corridor

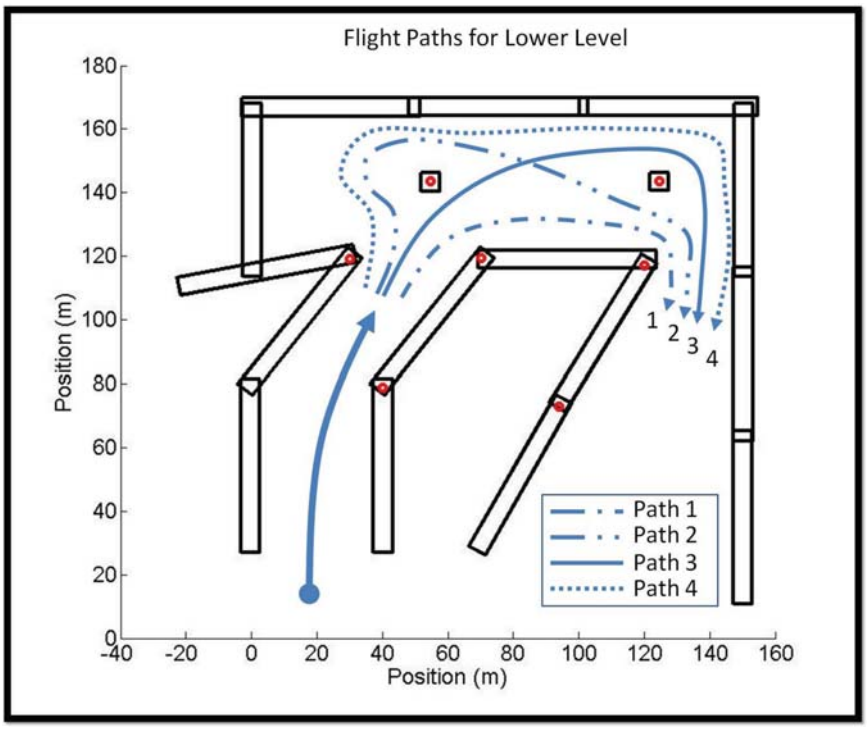

Figure 7.6: Top down view of the environment with the 4 flight paths through the lower level highlighted with different patterns.

walls and obstacles. The second goal was to correctly fly in the appropriate path around obstacles placed inside the environment. For the third goal, there was a ground target located near the end of the flight environment. The goal was to trigger a switch on the joystick when the subject felt that they were directly over the target. After the target is reached, the aircraft is automatically raised to the second level of the environment, above the corridor walls. The final goal was to fly directly over the center of the colored targets in the correct order supplied to them prior to flight. During all flights, the fNIR device was attached to the subject's head to measure cognitive workload during the flight. At the completion of each session (four flights in a session), the subject completed the NASA-TLX (Appendix B.1) and again filled out the confidence questionnaire (Appendix A.1).

Starting with session seven, subjects were shown a top down view of their flight

trajectory and target triggering location. This was introduced because it was noticed that most subject's performance was saturated after six sessions. For session one through six, there was no feedback given to the subjects about their performance other than the visuals received from the interface itself.

7.2.3 Session Ten

The final session (session ten) was performed immediately after session nine was completed. The subjects were asked to fly through the gantry environment using the interface from the group they were not a part of (e.g. onboard group used chase view interface). Every subject flew the same path (Path 2) and same marker setup for the two flights. The tasks were identical to the previous sessions. Distance to objects, target error and marker error were recorded for each flight. After the two flights, the subjects were asked to fill out a multiple choice questionnaire on their thoughts about the interface they just used. Extra opinions were also recorded for further analysis.

7.3 Data Analysis

7.3.1 Behavioral Data

The data analysis focused mostly on the assessment of a subject's behavioral data obtained through the measurement of aircraft positions, accelerations, and operator inputs during each flight. The following parameters were measured/cacluated in the analysis:

- Mean distance to the nearest obstacle.
- Planar distance from the center of ground target when button is triggered
- Deviation of flight path over the center point of the colored fiducial markers

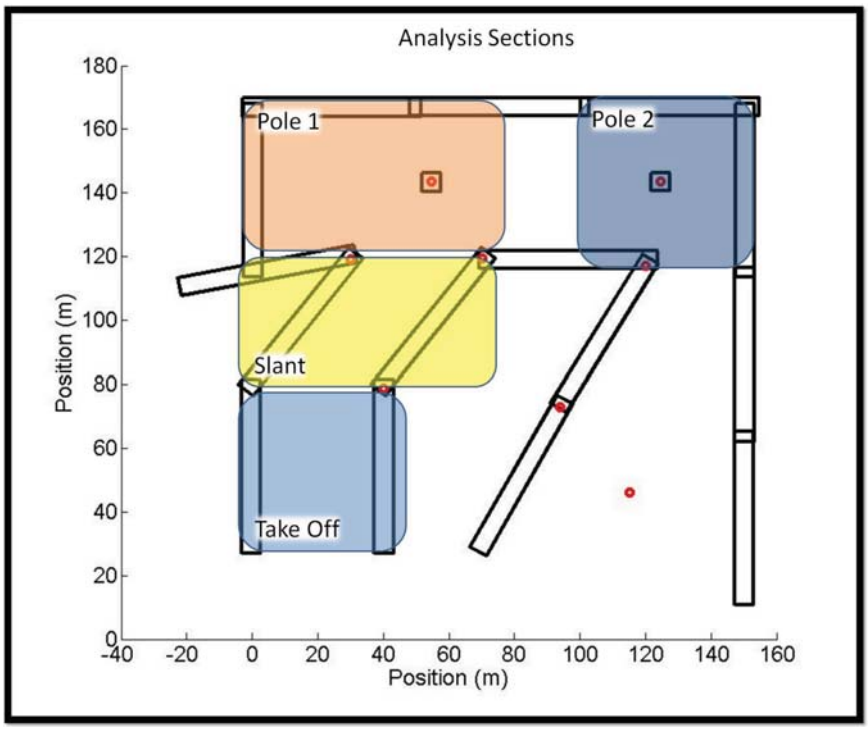

Figure 7.7: Top down view of the environment sectioned into four key analysis areas: Takeoff, Slant, Pole1 and Pole 2.

- Angular accelerations
- Operator control inputs

The environment was sectioned into four Locations(take off, slant, pole1, pole2) as seen in Figure 7.7. The flight variables [mean obstacle distance (ObDistance), mean magnitude angular acceleration(MagA), mean magnitude velocity(MagV), mean magnitude joystick velocities(jMagV)] were assessed for each flight path (1, 2, 3 and 4). The effects of View (Onboard, Chase) and Location (take off, slant, pole1, pole2) for each variable were evaluated using a Standard Least Squares model that evaluated each factor as well as the interaction between these factors using a full factorial design. In the event that significance was detected for location, multiple comparison Tukey tests were conducted ($\alpha = 0.05$).

In addition to the flight variables, the error variables [target error (TargetError), marker error (MarkerError)] were analyzed. The error variables contain the magnitude of the planar distance from the center of the target when the target switch is pulled (TargetError) and the magnitude of the planar distance from the nearest point on the flight path to the center of the markers (MarkerError). Chase and Onboard views were compared for each of the error variables using a Wilcoxon nonparametric test ($p<0.05$ for significance).

For all flight and error variables, a Spearman correlation was used to evaluate the relationship between the variable and session number for both Chase and Onboard view. JMP Statistical Software (Version 8, SAS Institute, Cary, NC) and $p<0.05$ was taken as significant for all statistical tests.

7.3.2 Subject Workload Data

The NASA Task Load Index (NASA TLX) gives a subjective workload assessment for each subject and each session. Chase and Onboard views were compared for each of the variables [adjusted weight rating, mental demand] using a Wilcoxon nonparametric test ($p<0.05$ for significance) to assess differences between the Onboard view and Chase view groups' subjective workloads.

The hemodynamic response features from the fNIR measures (i.e., mean and peak oxy-Hb, deoxy-Hb, oxygenation) were analyzed by the Optical Brain Imaging Laboratory. In their analysis, the fNIR measurements were first cleaned of motion artifacts [81]. A linear phase, finite impulse (FIR) low pass filter with a cut-off frequency of 0.2 Hertz was applied to the 16-voxel raw fNIR data for each subject to eliminate high frequency noise. For oxygenation calculations, a modified Beer-Lambert Law was applied to the data to calculate oxy-hemoglobin and deoxy-hemoglobin concentration changes. Analysis was run on all subjects and flights for session two through

Table 7.2: Significant effects and interactions for Path 1 using Standard Least Squares Model

Effect or Interaction	**ObDist**	**MagA**	**MagV**	**jMagV**
View		Yes		
Location	Yes	Yes	Yes	
View*Location	Yes	Yes	Yes	

Table 7.3: Significant effects and interactions for Path 2 using Standard Least Squares Model

Effect or Interaction	**ObDist**	**MagA**	**MagV**	**jMagV**
View		Yes		Yes
Location	Yes	Yes	Yes	Yes
View*Location	Yes	Yes		

session six. It is believed that the change in session seven through session nine (showing the subjects their results) would alter the fNIR analysis so these three sections were excluded from the current fNIR analysis. A repeated measures ANOVA was run across all flights, sessions two through six, and views for each voxel to determine if the data are not consistent with the hypothesis that all the samples were drawn from a single population. If this was the case, then a *Tukey-Kramer Multiple-Comparison test* was used to determine any significant differences between Chase and Onboard subjects ($\alpha = 0.05$).

Table 7.4: Significant effects and interactions for Path 3 using Standard Least Squares Model

Effect or Interaction	ObDist	MagA	MagV	jMagV
View	Yes	Yes		
Location	Yes	Yes	Yes	Yes
View*Location	Yes	Yes		

Table 7.5: Significant effects and interactions for Path 4 using Standard Least Squares Model

Effect or Interaction	ObDist	MagA	MagV	jMagV
View		Yes		Yes
Location	Yes	Yes	Yes	Yes
View*Location	Yes	Yes		

7.4 Results and Discussion

7.4.1 Behavioral Data

The results of the flight path analysis described earlier are shown in Figure 7.8 through Figure 7.22 and the results of the Standard Least Squares Model are shown in Table 7.2 through Table 7.5.

Mean Magnitude Velocity (MagV)

The results of mean magnitude angular velocity for each path are shown in Figure 7.8 and Figure 7.9. In Flight Path 1, the main effect of location was significant ($p<0.0001$) (Table 7.2). In addition, a significant interaction between view and location was observed ($p=0.04$). However, the interactions that are significant were not

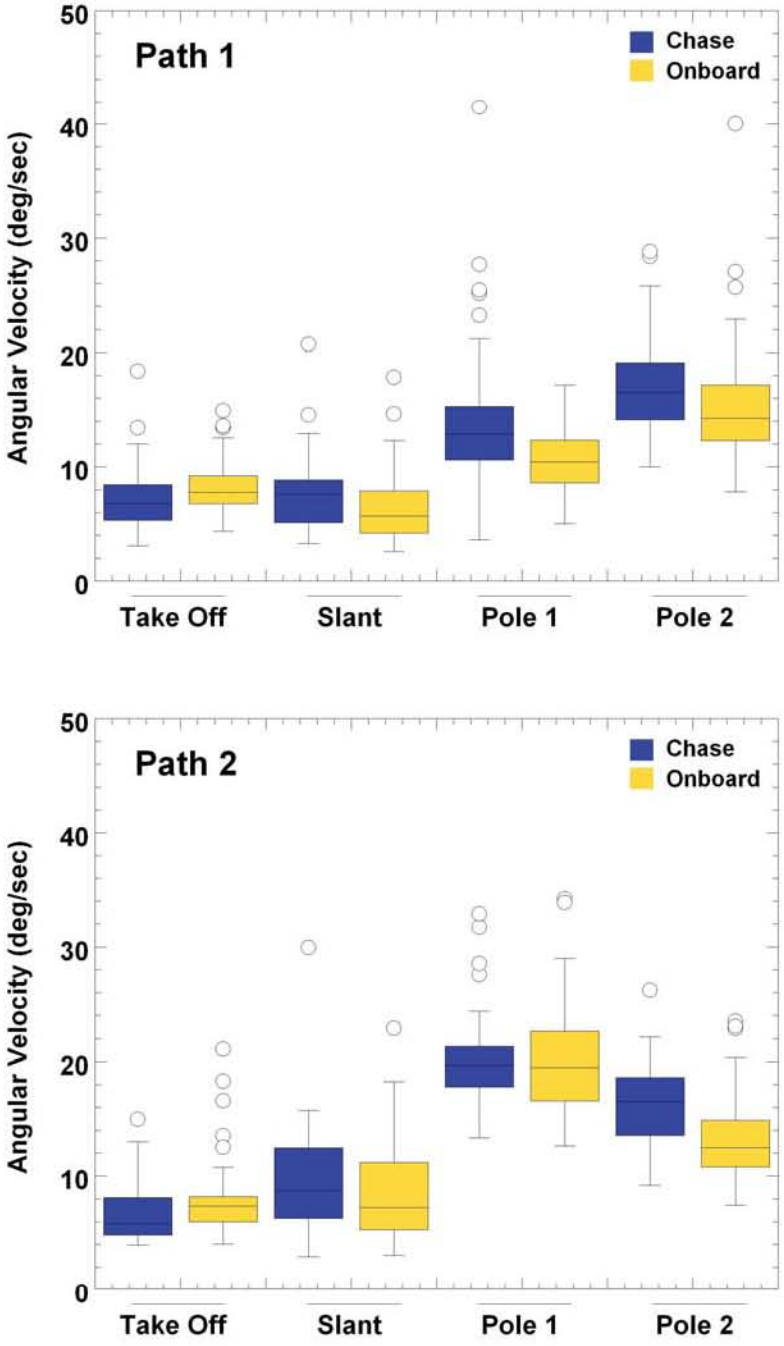

Figure 7.8: Mean Magnitude Angular Velocity for all locations (Take Off, Slant, Pole 1, and Pole 2). Significance, if any are, highlighted by an asterix with a line leading to the significant sets. Top:Path 1 Results Bottom: Path 2 Results

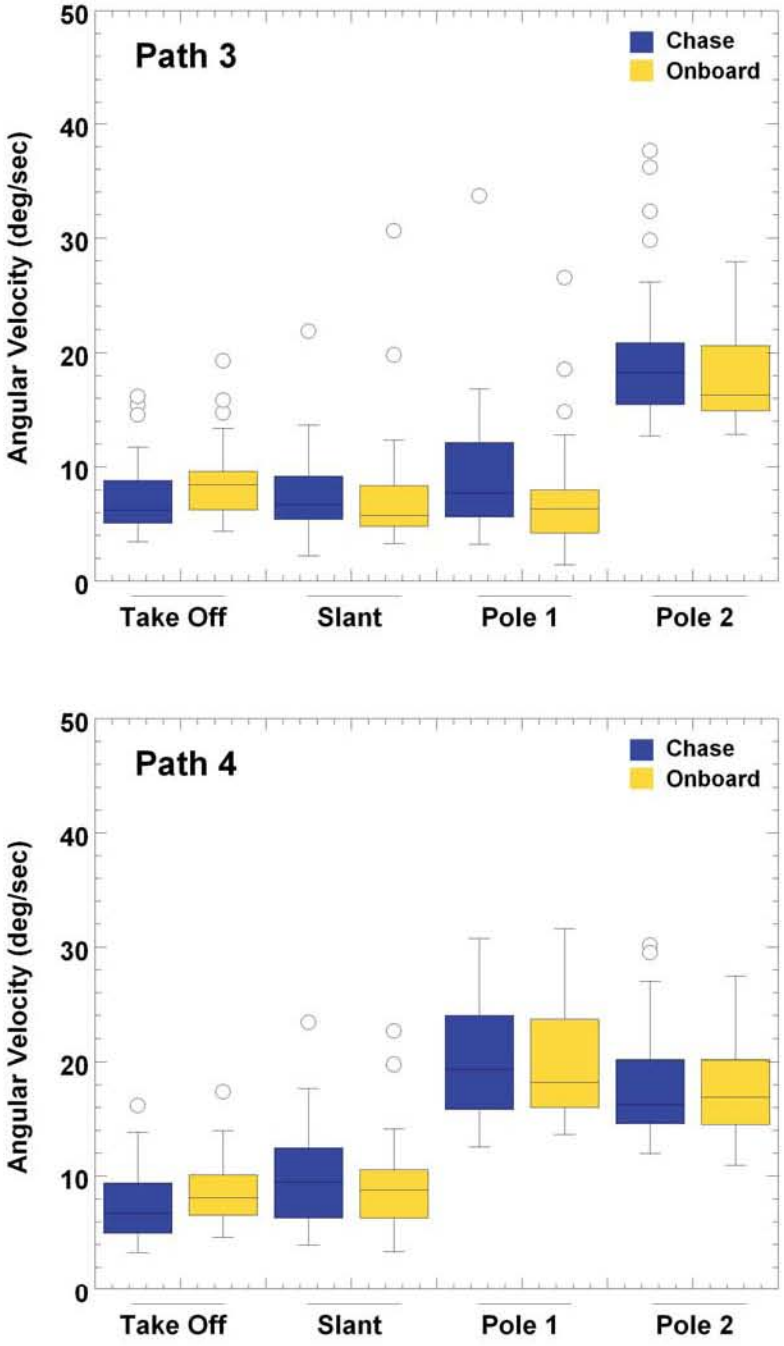

Figure 7.9: Mean Magnitude Angular Velocity for all locations (Take Off, Slant, Pole 1, and Pole 2). Significance, if any are, highlighted by an asterix with a line leading to the significant sets. Top:Path 1 Results Bottom: Path 2 Results

relevant to this study. For example, Chase view Pole 2 being significantly different from Onboard Take Off was not of importance. Although significant differences were not detected when comparing chase view and onboard camera view at specific locations of the flight, the angular velocities of the chase view were higher than those of the onboard view at Pole 2 for Paths 1 and 2 (Figure 7.8). Path 1 does not require a flight around the poles themselves but the pole 2 area does have a sharp turn. This result is consistent with the findings from the Development Stage studies that Chase view produces tighter and quicker turns.

For Flight Path 2, the main effect of location was significant (p=0.0005) as shown in Table 7.3 but no significant interaction was observed. Similar to Flight Path 1, shown in Figure 7.9 higher velocities are seen in locations such as Slant and Pole 2 for Chase View. Based on the hypotheses, it was expected that Pole 1 would have a significant difference since Path 2 takes the aircraft around pole 1 but this was not the case. It is investigated further later in this chapter.

Flight Path 3, the main effect of location was significant (p<0.0001) as shown in Table 7.4 but no significant interaction was observed. Again, similar to Flight Path 1, shown in Figure 7.8 higher velocities are seen in turning sections. Based on the hypotheses, it was expected that Pole 2 would have a significant difference since Path 3 takes the aircraft around pole 2 but this was not the case. It is investigated further later in this chapter.

Flight Path 4, the main effect of location was significant (p=0.009) as shown in Table 7.5 but no significant interaction was observed. The same analysis for Path 2 and Path 3 holds true here.

For all Flight Paths combined, a Spearman correlation indicated a significant negative relationship between mean magnitude velocity and session number for (Chase) subjects 7 (ρ = -0.38, p = 0.00) and 9 (ρ = -0.29, p = 0.00) and Onboard subjects 4

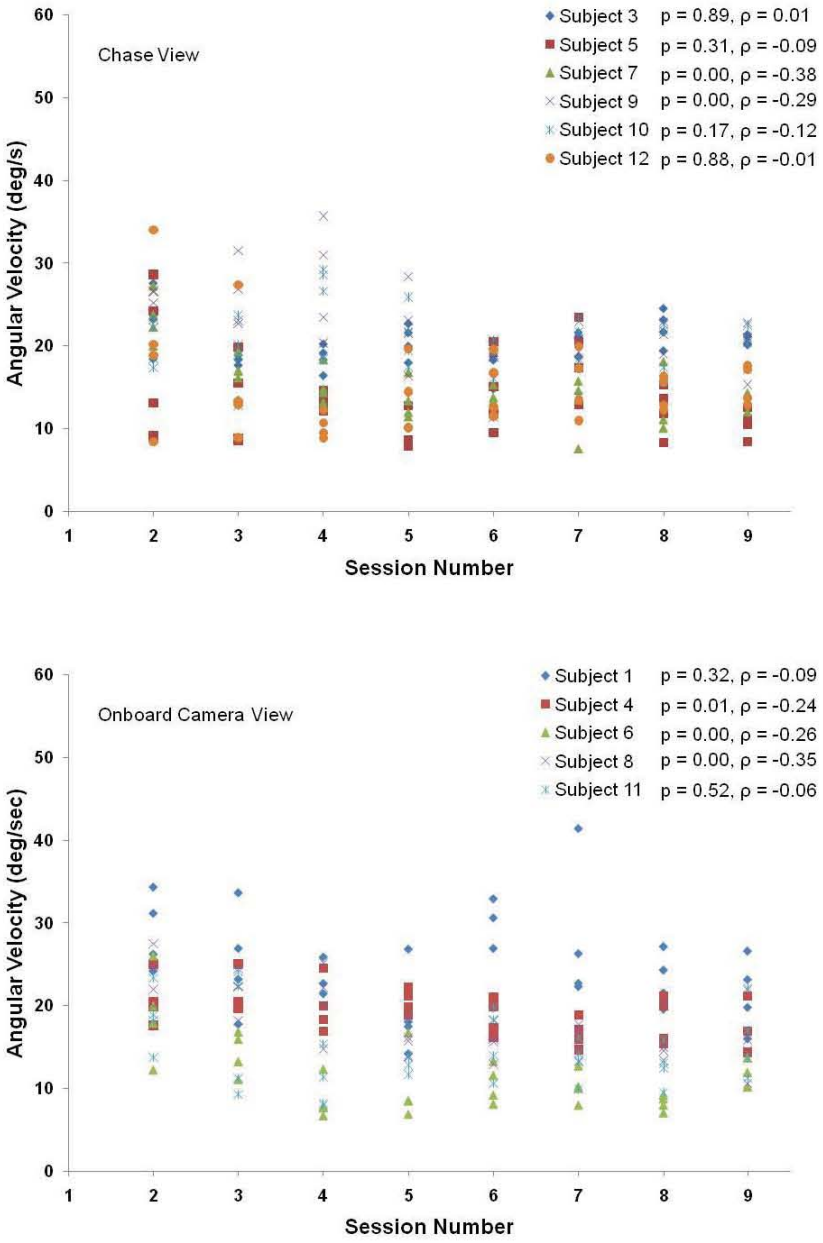

Figure 7.10: Spearman correlation of Angular Velocity and Session. Subjects with a $p<0.05$ show significant correlation. Top:Chase Subjects Bottom:Onboard Subjects

(ρ = -0.24, p = 0.01),6 (ρ = -0.26, p = 0.00), and 8 (ρ = -0.35, p = 0.00) as shown in Figure 7.10. This demonstrates an improvement in smoother flight over sessions for a subset of the subjects.

Mean Angular Acceleration (MagA)

The results of mean magnitude angular acceleration for each path are shown in Figure 7.11 and Figure 7.12. For all flight paths, the main effects of view (all p< 0.0001) and location (all p< 0.0001) were significant as shown in Table 7.2 to Table 7.5. In addition at a given view and location, significant interactions were observed (p=0.001, p<0.0001, p=0.007, p=0.004 for Path 1 to Path 4 respectively) as shown in Figure 7.11 and Figure 7.12. All paths showed a significantly higher angular acceleration at the locations of Pole 1 and Pole 2. Each of these locations requires a sharp turn which leads to an increase in the angular velocity. The higher accelerations can be explained by visual observations of the subjects' behavior during the flights. Onboard camera subjects would make very large sweeping roll maneuvers with a high amplitude in the angle. As a side result, they would overshoot their desired angle and would then proceed to make large and long roll maneuvers back to stabilize the aircraft. This occurred in a number of onboard subjects because most relied on optic flow to gain awareness of the aircraft roll angle rather than the artificial horizon instrument gage. The reliance on optic flow required a relatively large roll motion before the optic flow was large enough to gather awareness from. Chase view subjects on the other hand could easily see their aircraft angle as they rolled and more easily predicted their approach to the desired angle. This allowed for much faster and more minute motions to control the roll angle. An example plot (Figure 7.13) shows the larger sweeping roll angles by an onboard camera subject and the smaller and minute angle corrections of a chase view subject through a sharp turn.

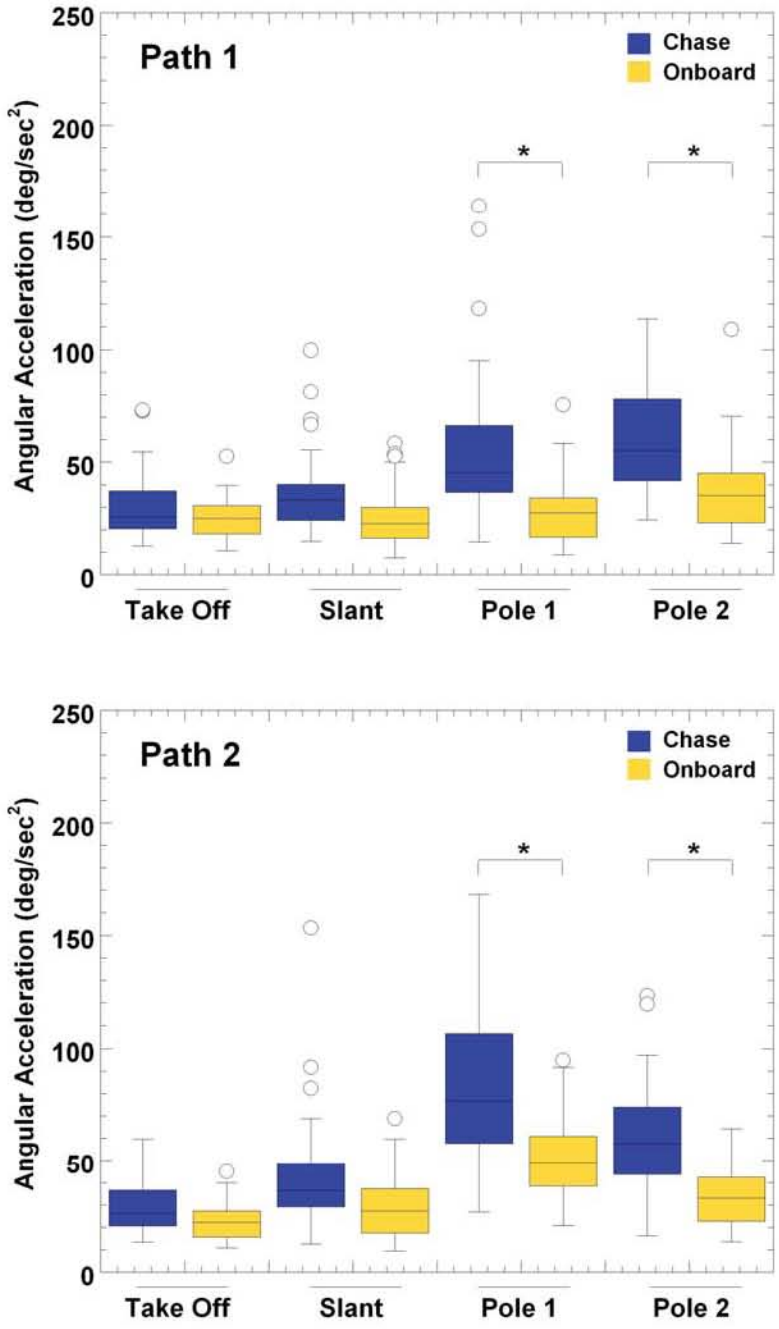

Figure 7.11: Mean Magnitude Angular Acceleration for locations Take Off, Slant, Pole 1, and Pole 2. Significance, if any are, highlighted by an asterix with a line leading to the significant sets. Top:Path 1 Results Bottom: Path 2 Results

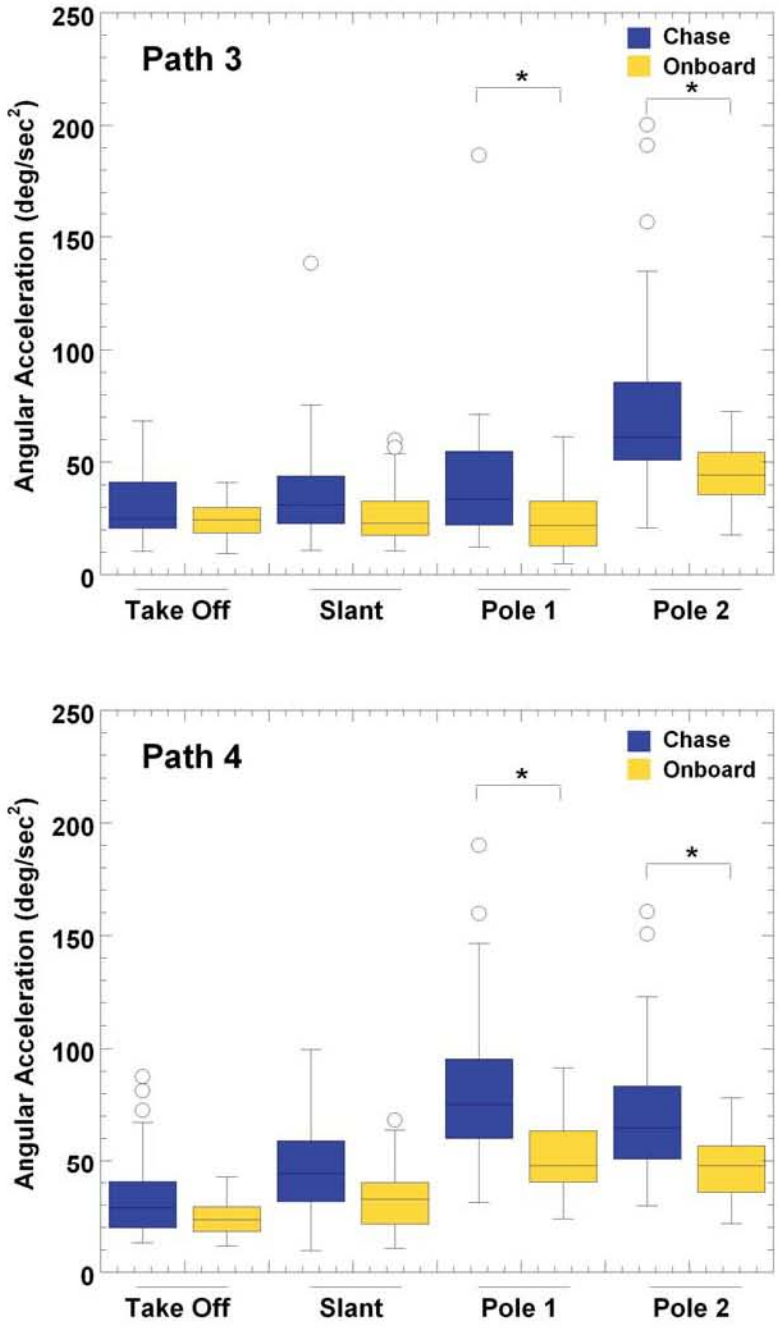

Figure 7.12: Mean Magnitude Angular Acceleration for locations Take Off, Slant, Pole 1, and Pole 2. Significance, if any are, highlighted by an asterix with a line leading to the significant sets. Top:Path 1 Results Bottom: Path 2 Results

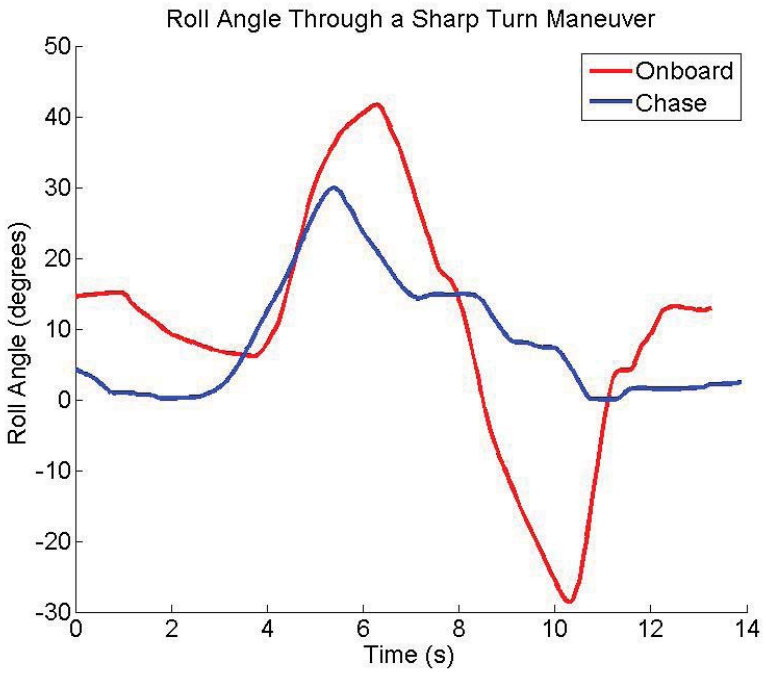

Figure 7.13: Example roll angle through a sharp turn for an onboard camera subject (red) and a chase view subject (blue). Onboard view subjects tended to take large motion turns, relying on optic flow to gather awareness of aircraft pose, while chase view subjects tended to take quicker turns with smaller intermittent angle corrections

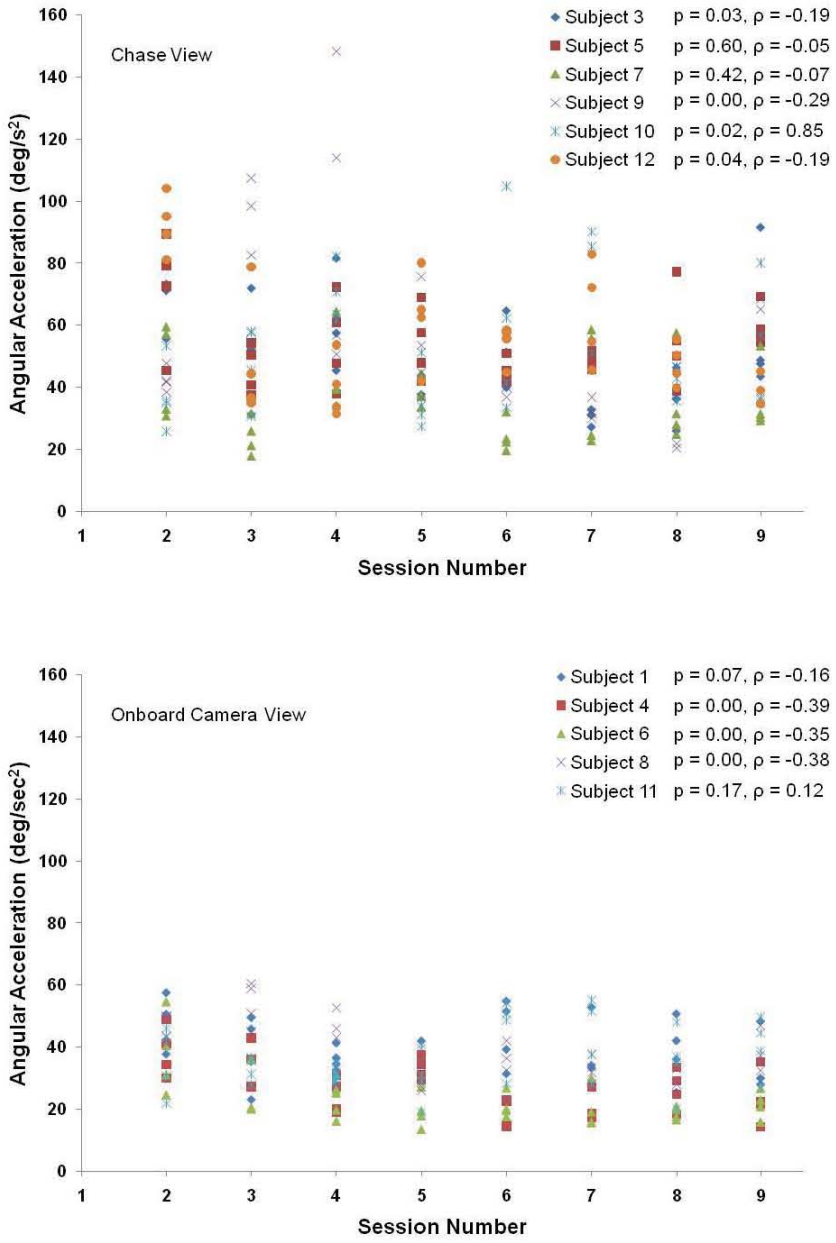

Figure 7.14: Spearman correlation of Angular Acceleration and Session. Subjects with $p<0.05$ show significant correlation. Top:Chase Subjects Bottom:Onboard Subjects

For all Flight Paths combined, a Spearman correlation indicated a significant negative relationship with Session for (Chase) subjects 3 (ρ = -0.19, p = 0.03), 9 (ρ = -0.29, p = 0.00), and 12 (ρ = -0.19, p = 0.04) and (Onboard) subjects 4 (ρ = -0.39, p = 0.00), 6 (ρ = -0.35, p = 0.00), and 8 (ρ = -0.38, p = 0.00) as shown in Figure 7.14. (Chase) Subject 10, however showed a significant positive relationship (ρ = 0.85, p = 0.02) with session however the values of Angular Acceleration are relatively consistent. This also helps to demonstrates an improvement in control over sessions.

Mean Joystick Velocity (jMagV)

The results of mean magnitude joystick velocity for each path are shown in Figure 7.15 and Figure 7.16. For all flights, no significant interaction was observed (p=0.32, p=0.58, p=0.34, p=0.98 for Path 1 to Path 4 respectively) (Table 7.2 to Table 7.5). For Path 2 and Path 4, the main effects of View (p=0.03, p=0.02 respectively) and Location($p<0.0001$ for both paths) were significant while Path 3 only showed the main effect of Location as significant ($p<0.001$). Path 1 had none (p=0.36) for both View and Location). Observing Figure 7.15 and Figure 7.16, while not significantly different, the Onboard Camera subjects mean magnitude joystick velocities were higher across all paths. This leads to the conclusion that Onboard Camera subjects were manipulating the joystick controls more than Chase view subjects. This might mean that Onboard camera subjects felt the aircraft was less stable, requiring more corrections.

A Spearman correlation for Mean Joystick Velocity and session number did not show a significant relationship with session. This demonstrates that subjects did not significantly change how they manipulated the joystick across sessions.

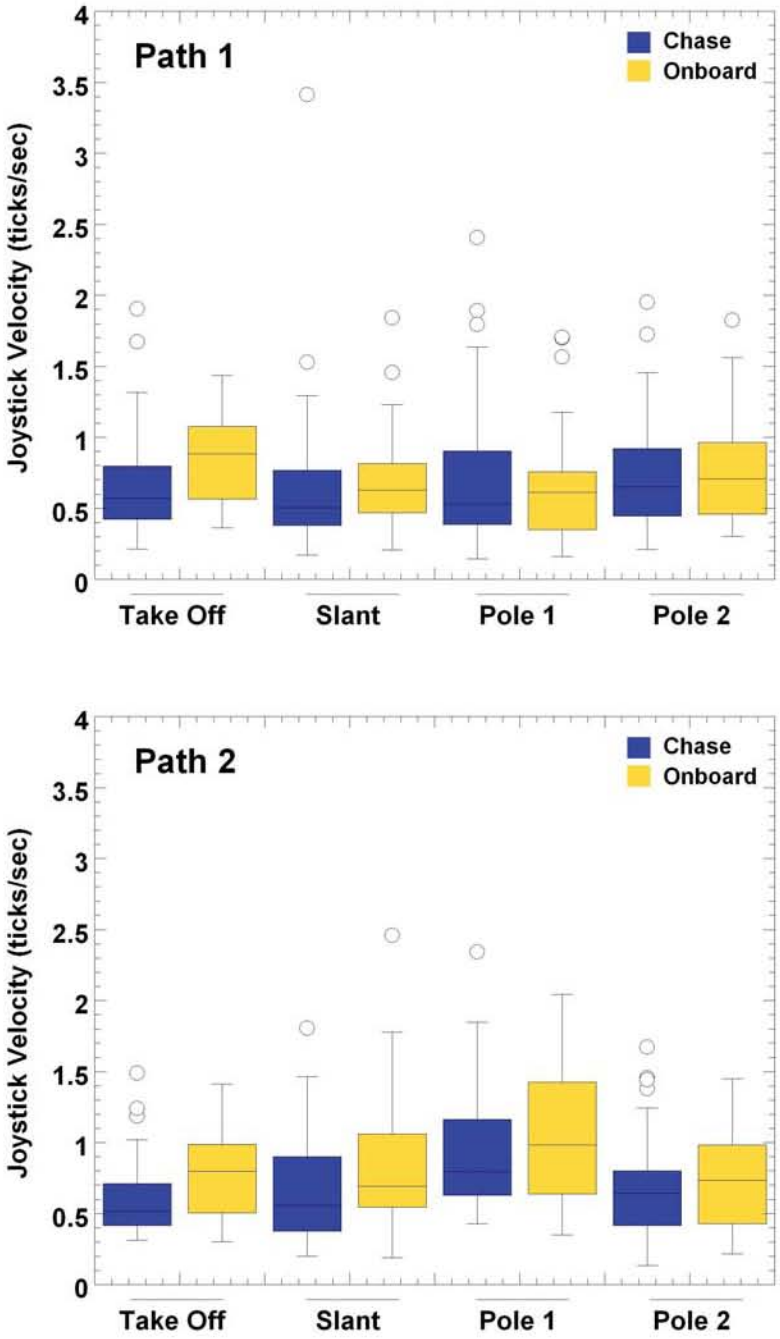

Figure 7.15: Mean Magnitude Joystick Velocities for locations Take Off, Slant, Pole 1, and Pole 2. Significance, if any are, highlighted by an asterix with a line leading to the significant sets. Top:Path 1 Results Bottom: Path 2 Results

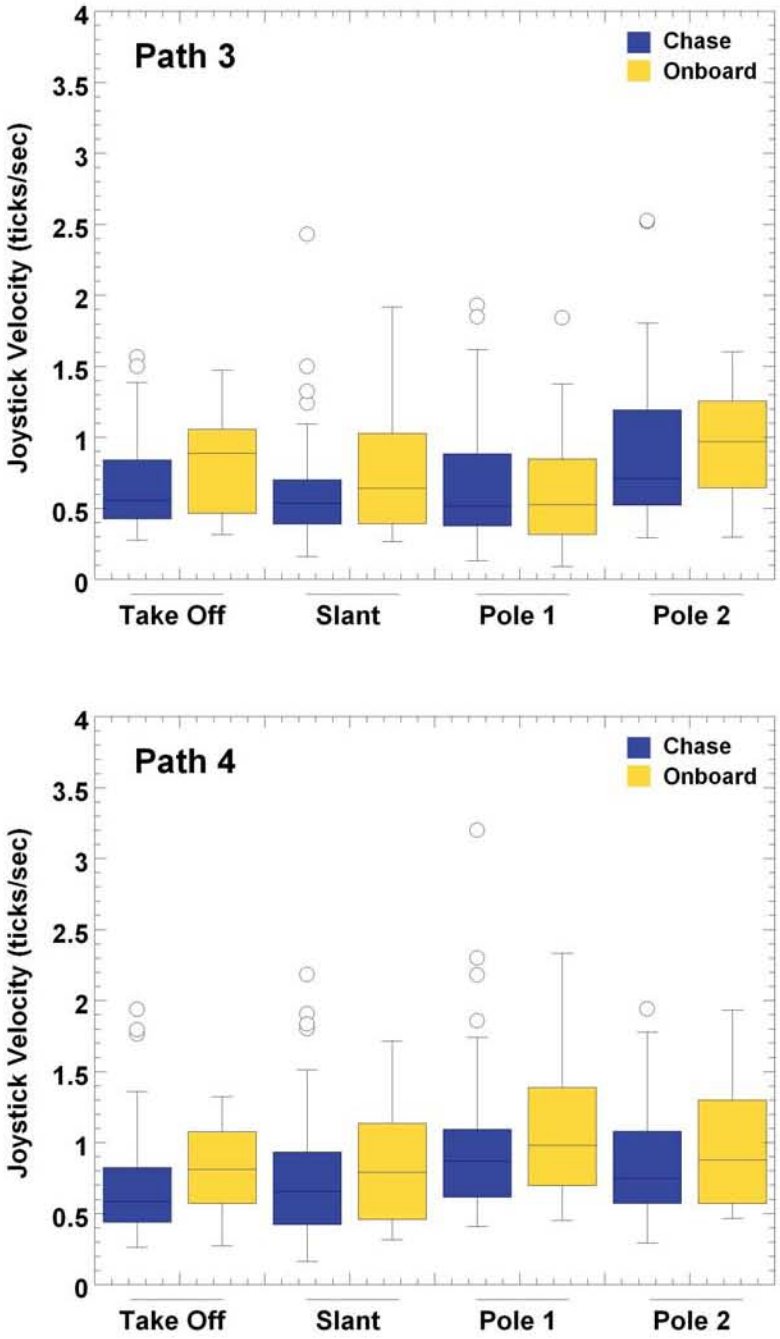

Figure 7.16: Mean Magnitude Joystick Velocities for locations Take Off, Slant, Pole 1, and Pole 2. Significance, if any are, highlighted by an asterix with a line leading to the significant sets. Top:Path 1 Results Bottom: Path 2 Results

Mean Obstacle Distance (ObDist)

The results of Mean Obstacle Distance for each path are shown in Figure 7.17 and Figure 7.18. For all flight paths, the main effect of Location was significant (for all paths $p<0.0001$) and at a given view and location, significant interactions were observed ($p<0.0001$, $p=0.004$, $p<0.0001$, $p=0.0005$ for Path 1 through Path 4 respectively). The effect of view was also significant for Flight Path 3 ($p=0.01$) (Table 7.2 to Table 7.5). The results for each flight path are shown in Figure 7.17 and Figure 7.18.

For Flight Path 1, Chase was found to be significantly lower at Slant and significantly higher at Pole 1 than Onboard. This supports Hypothesis 1 demonstrating a tighter turn around the corner in the slant section. Since Path 1 does not go around Pole 1, the nearest obstacle in the curve is pole 1 itself, so a higher distance represents a tighter turn around the corner.

Flight Path 2 showed a significance in the interaction of view and location however the resulting significance was not relevant based on the reasoning presented for Path 1's mean magnitude velocity. According to Hypothesis 1, it would be expected that Chase would have a significantly lower distance in the Pole 1 area representing a tighter turn. This however is not the case and is investigated further later in this chapter.

For Flight Path 3, Chase was found to be significantly lower at Slant and significantly higher at Pole 1 than Onboard. Flight Path 3 matches Flight Path 1 for the Take Off, Slant and Pole 1 areas so the analysis of Path 1 holds true here. Chase is significantly higher than Onboard in the Pole 2 area. This would seem to contradict Hypothesis 1 however this is discussed later in this chapter.

For Flight Path 4, for the location Pole 1, Chase was found to be significantly higher than Onboard. Path 4 takes the aircraft around both pole 1 and pole 2

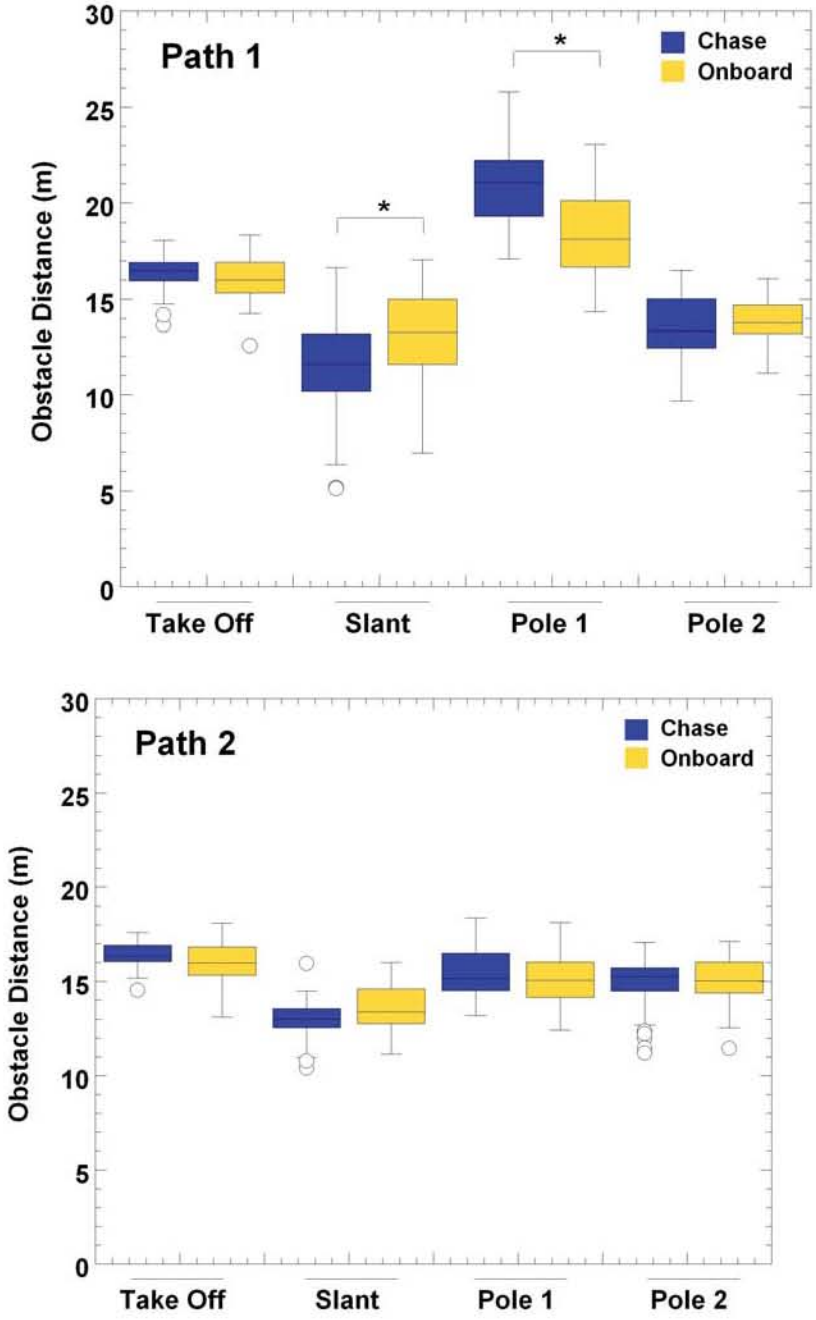

Figure 7.17: Mean Obstacle Distance of the Aircraft for locations Take Off, Slant, Pole 1, and Pole 2. Significance, if any are, highlighted by an asterix with a line leading to the significant sets. Top:Path 1 Results Bottom: Path 2 Results

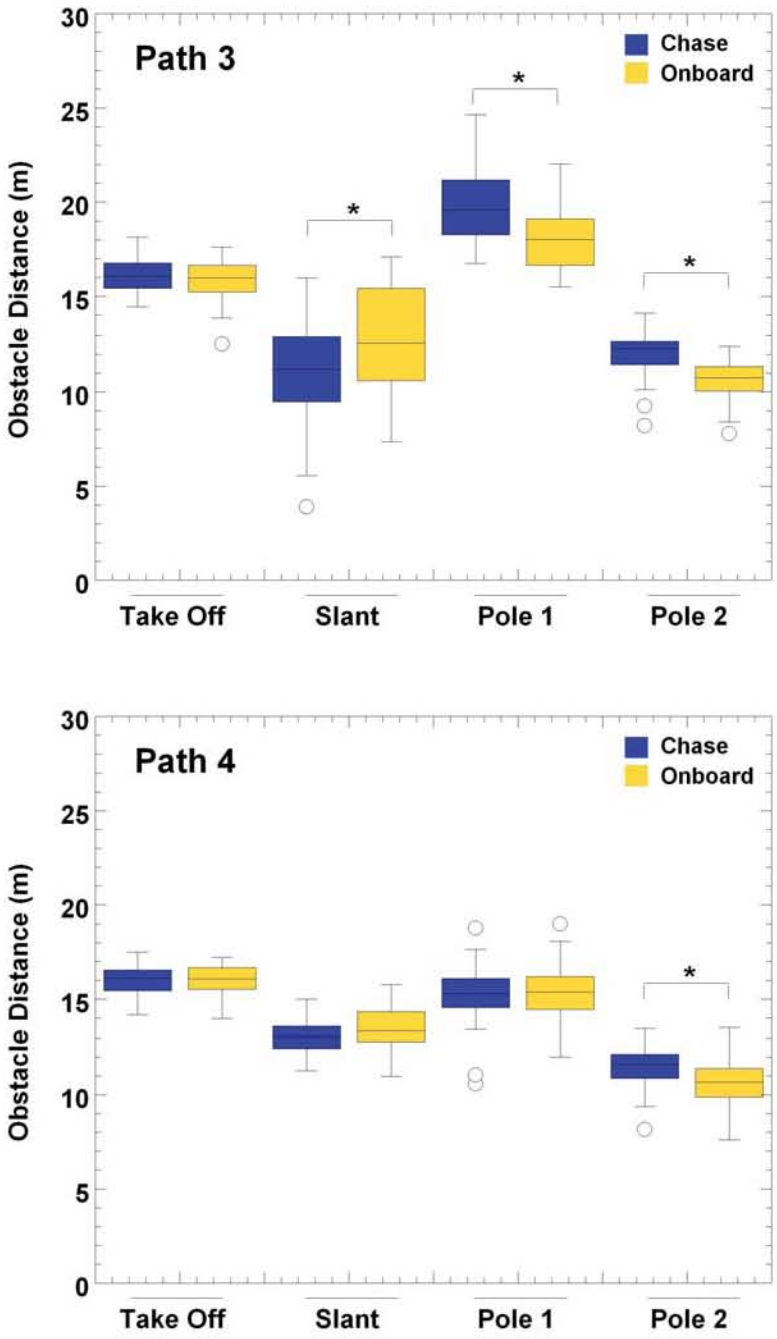

Figure 7.18: Mean Obstacle Distance of the Aircraft for locations Take Off, Slant, Pole 1, and Pole 2. Significance, if any are, highlighted by an asterix with a line leading to the significant sets. Top:Path 1 Results Bottom: Path 2 Results

themselves so it would be expected to see a significantly lower distance using Chase. The discussion of this is presented in the next section.

A Spearman correlation did not show a significant relationship with session for Mean Obstacle Distance. This is shown in Figure 7.19 as the plots show, the mean obstacle distance did not change significantly over session which means that pilots awareness of the aircraft extremities did not change across sessions. This further supports Hypothesis 1 that even with continued sessions, Onboard does not improve this awareness to match that of Chase.

Pole 1 and Pole 2 Further Investigation

Closer investigation into why the data in some cases did not support Hypothesis 1 revealed that the Pole 1 and Pole 2 areas include not only the pole itself but also the surrounding walls. Figure 7.20 shows the phenomenon where a Chase subject flew tighter to the pole but the Onboard subject flew closer to the walls around the actual Pole 1 and the actual Pole 2. This shows that Onboard subjects tended to take wider turns to go around the obstacle which ended up taking them closer to the wall. The pole 1 and pole 2 areas were further sectioned as highlighted by yellow boxes in Figure 7.20. The mean obstacle distance was calculated to the pole itself in these sections. Figure 7.21 shows that in all flight paths that go around the poles (Flight Path 2,3,4), Chase is actually significantly closer ($p<0.0001$ for pole 1 actual, $p<0.0001$ for pole 2 actual). The data now supports Hypothesis 1 that Chase view enhances awareness of the vehicle's extremities allowing for more efficient turn paths around obstacles.

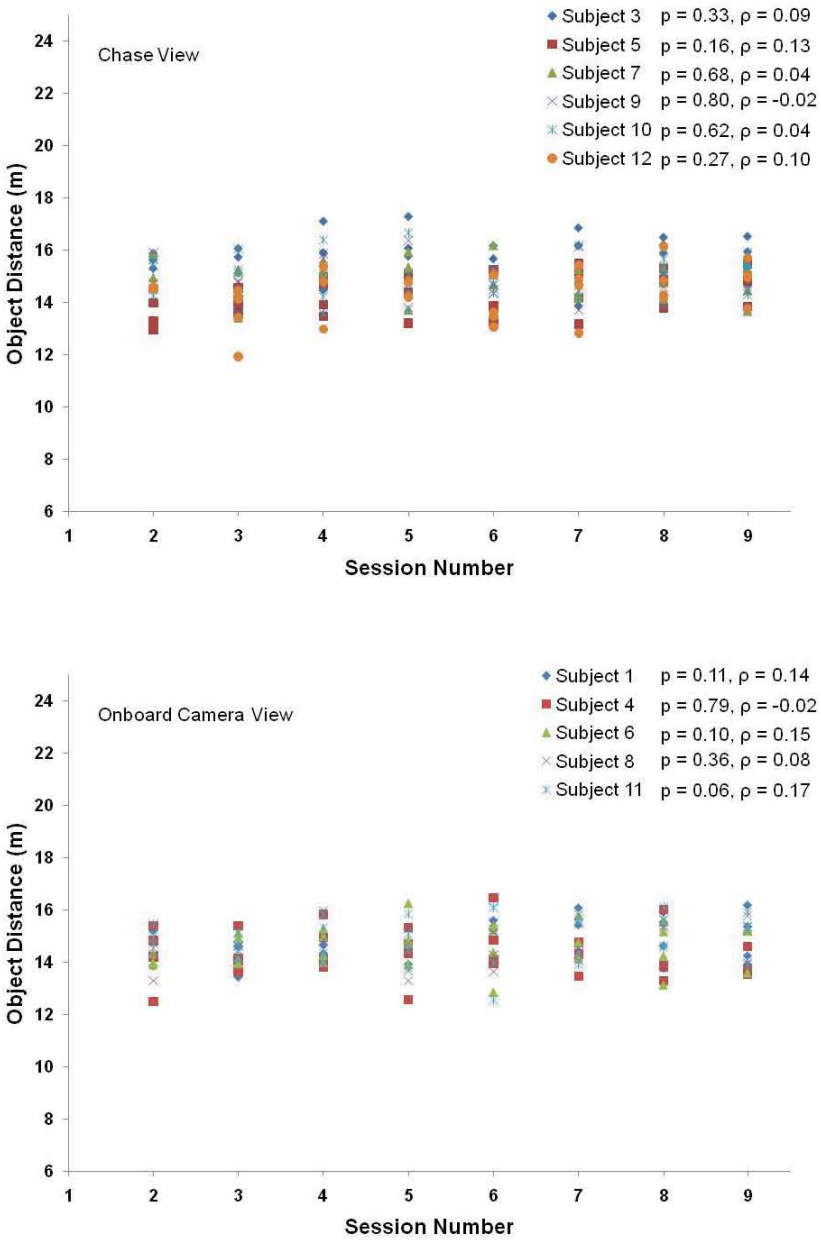

Figure 7.19: Spearman correlation of Obstacle Distance and Session. Subjects with $p<0.05$ show significant correlation. Top:Chase View Bottom:Onboard Subjects

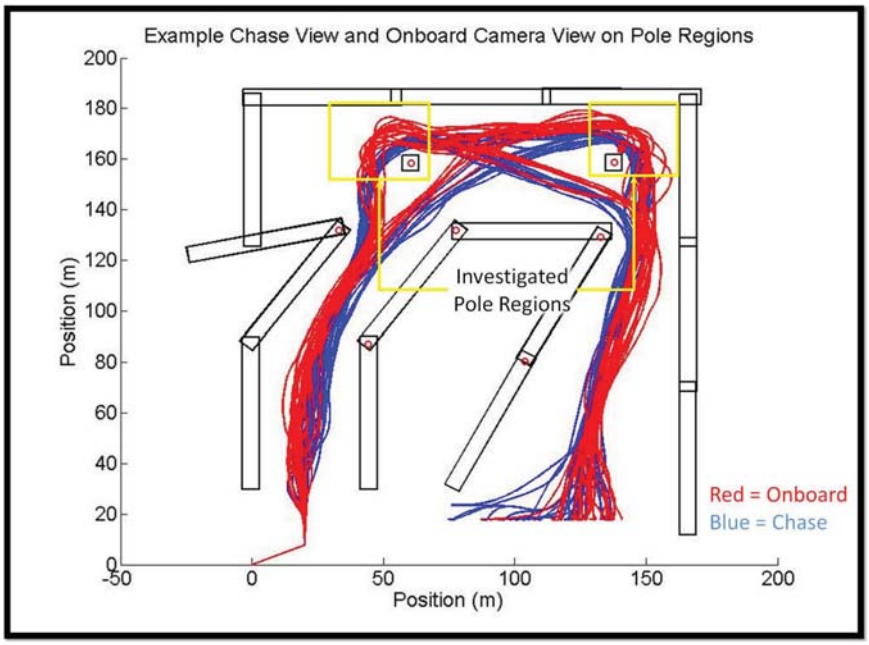

Figure 7.20: Top down view of the environment with the pole locations highlighted. The red line shows all the trajectories around the poles for an example Onboard View subject, the blue line shows all the trajectories around the poles for an example Chase View subject.

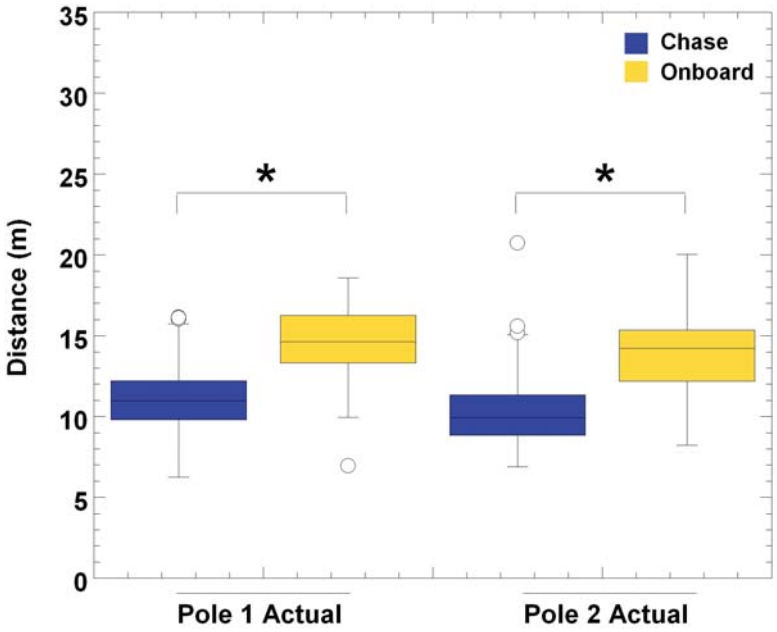

Figure 7.21: Obstacle Distance of the aircraft around the actual Pole 1 and Pole 2. Significant differences are highlighted by the asterix.

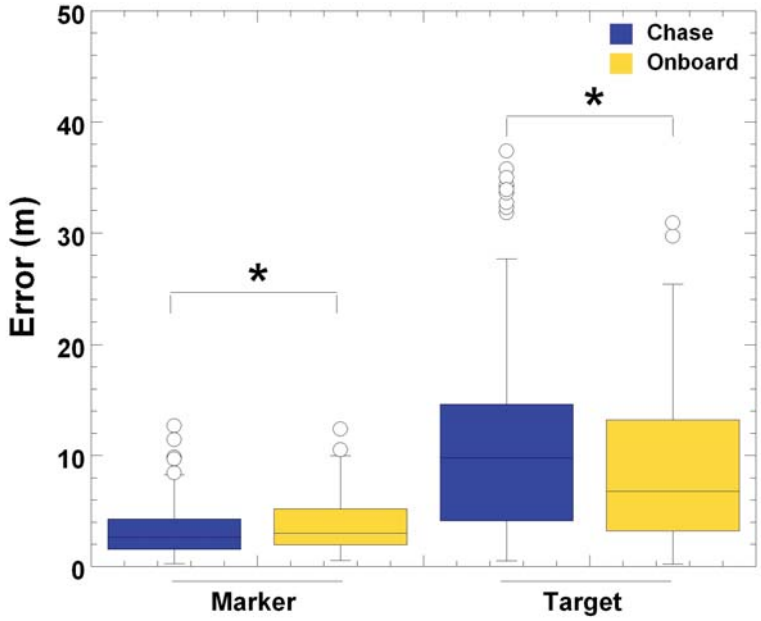

Figure 7.22: Magnitude error distance of the aircraft from the Target center and center of the Markers. Significant differences are highlighted by the asterix.

Target and Marker Error

Shown in Figure 7.22 are Chase and Onboard results of the Target Error and Marker Error. According to Hypothesis 3, one would expect significantly lower error with Chase versus Onboard. The Chase view would give a better 3D spatial awareness of the vehicle with respect to the surrounding environment. Only the data for Marker Error supports Hypothesis 3. The Marker Error was significantly higher ($p=0.02$) for the Onboard subjects when compared to the Chase subjects. The opposite was true for Target Error where the chase view group was significantly higher ($p=0.006$). This result can be explained by perceptual error and perspective.

As shown in Figure 7.23 when the object of interest passes out of the onboard camera image, Onboard subjects predict how long they have to wait until the aircraft is over the object. The higher up the aircraft, the longer they have to wait. Chase

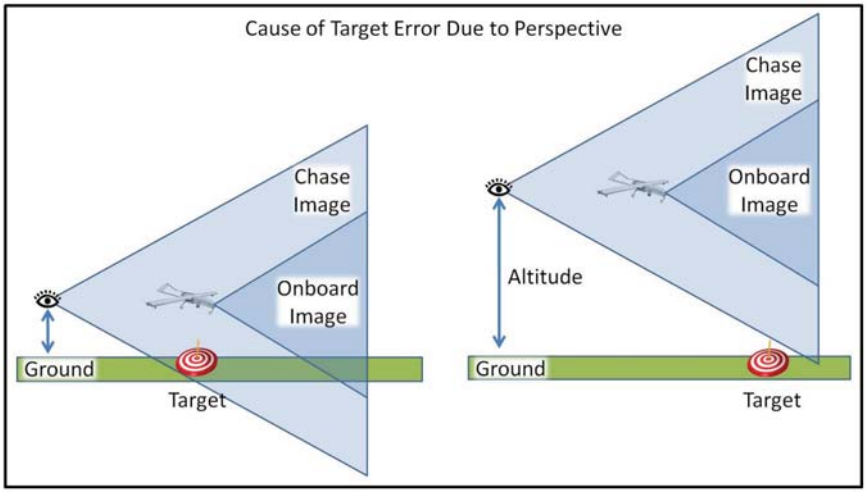

Figure 7.23: Left:Demonstration of how the target can be out of the onboard camera view but still in the chase view when under the aircraft. Right: Demonstration of how the target can be out of both views and still be ahead of the aircraft.

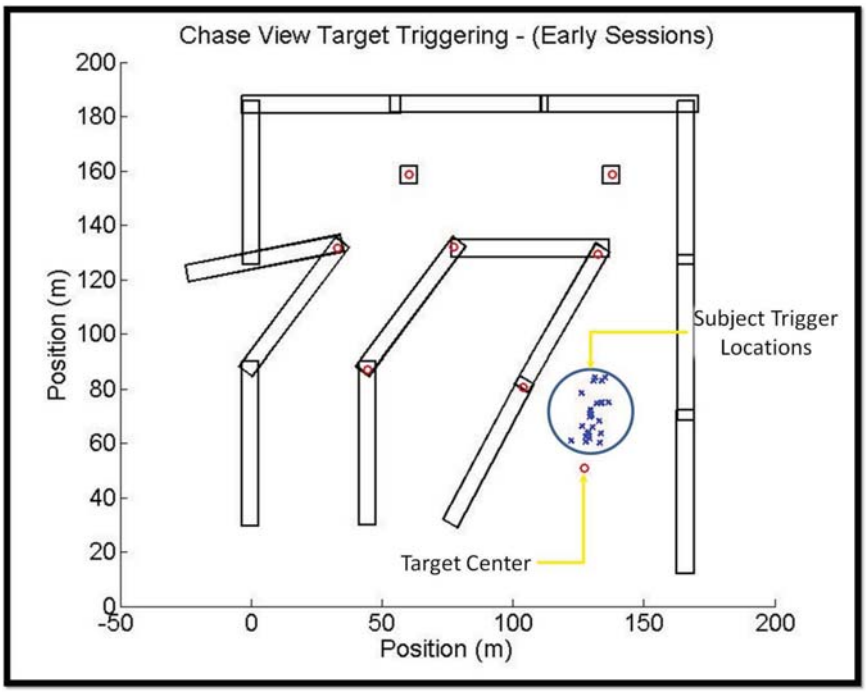

Figure 7.24: Top down view of the flight environment. Highlighted are all the locations from Session 2 where Chase view subjects triggered the target trigger signifying that they thought the aircraft was over the center of the target.

view subjects have the same requirement, however the object stays in view longer due to added virtual view. When low enough, the object can still be seen as it passes under the vehicle. However when higher, Chase view subjects still have to wait after the target has exited even the Chase view image. In early tests, Chase view subjects did not understand this perspective issue and tended to trigger over the target when the virtual image appeared under the the aircraft avatar, well before the actual target area. This can be seen in Figure 7.24 which shows the location of chase view subject target triggers in early trials. Not a single subject triggered after the target had already passed which supports the claim that misunderstanding of the perspective caused subjects to think the target was directly below the aircraft when it passed by the aircraft avatar in the chase view image. During the second level flights, all subjects were closer to the height of the markers, lessening the perspective error, and thereby improving the Chase subject's results. All subjects were told about the perspective issue after session 2 and results progressively improved.

For both Target Error and Marker Error, a Spearman correlation indicated a significant negative relationship with session for both Chase (ρ = -0.49, p = 0.00) and Onboard (ρ = -0.36, p = 0.00) as shown in Figure 7.25. As expected, a decrease in the amount of error is seen, after Session six, when the subjects were able to see their performance. This does not necessarily address any of the hypotheses but it does validate the use of the SISTR interface as a training system.

Workload Data

Hypothesis 4 would suggest that the task load of the subject, specifically the mental demand of the subject, would be statistically lower for Chase view. The NASA-TLX results are shown in Figure 7.26 and Figure 7.27. When comparing the task load and mental demand were not found to be statistically significant (p=0.103,

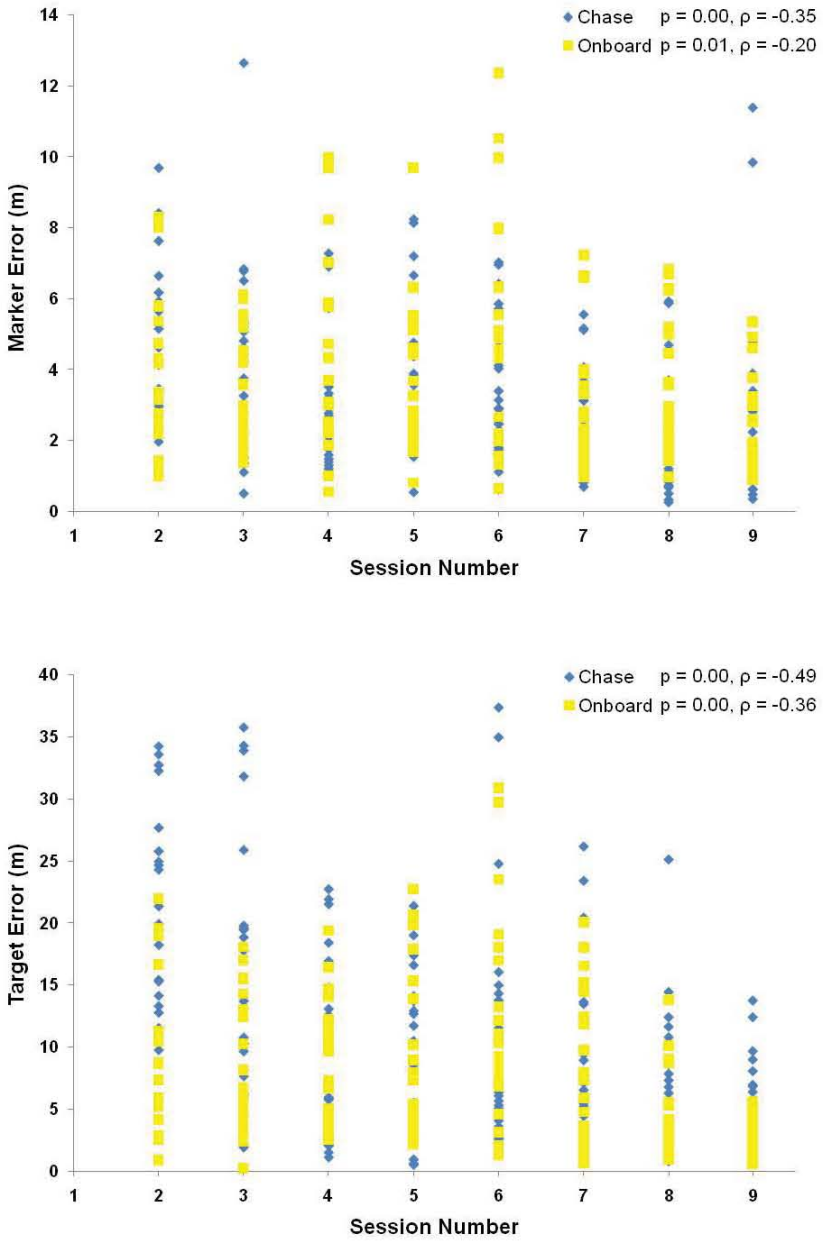

Figure 7.25: Spearman correlation of Error with Session. Subjects with $p<0.05$ show significant correlation. Top: Marker Error Bottom: Target Error

p=0.395, respectively) between Chase view and Onboard view. Further tests with more subjects as well as tasks that focus more on mental stimulation may help to support this hypothesis.

While the subjective tests showed no significance, the fNIR analysis showed otherwise. The difference of average oxygenation changes for all Chase and Onboard view groups were found to be significant ($F_{1,361} = 6.47, p < 0.012$). Post hoc analysis with Tukey-Kramer Multiple-Comparison tests also indicated that Chase and Onboard groups were different from each other. Onboard view was found to be significanly higher than Chase view. These results are shown in the top of Figure 7.28.

The difference of maximum oxygenation changes for chase view and onboard view groups were found to be significant ($F_{1,361} = 5.94, p < 0.016$). Post hoc analysis with Tukey-Kramer Multiple-Comparison test also indicated that Chase view and Onboard view groups are different from each other. Figure 7.28, bottom, shows that Onboard view group had higher maximum oxygenation change when compared with the Chase view group.

These comparisons were on voxel four. The location of the fourth voxel measurement registered on the brain surface is shown in Figure 7.29 [82]. Activation in the brain area corresponding to voxel four has been found to be sensitive during completion of standardized cognitive tasks dealing with concentration, attention, and working memory [83, 84, 81]. Higher oxygenation in this area is related to higher mental workload of the subject. Chase subjects' average oxygenation levels for voxel four was significantly lower than Onboard subjects, revealing that subjects using the onboard camera view were using more mental resources to conduct the flights. This result is most likely attributable to the narrower viewable angle and rolling of the environment in the onboard view, which require more cognitive processing by the subject to construct an accurate working mental model of the environment and the

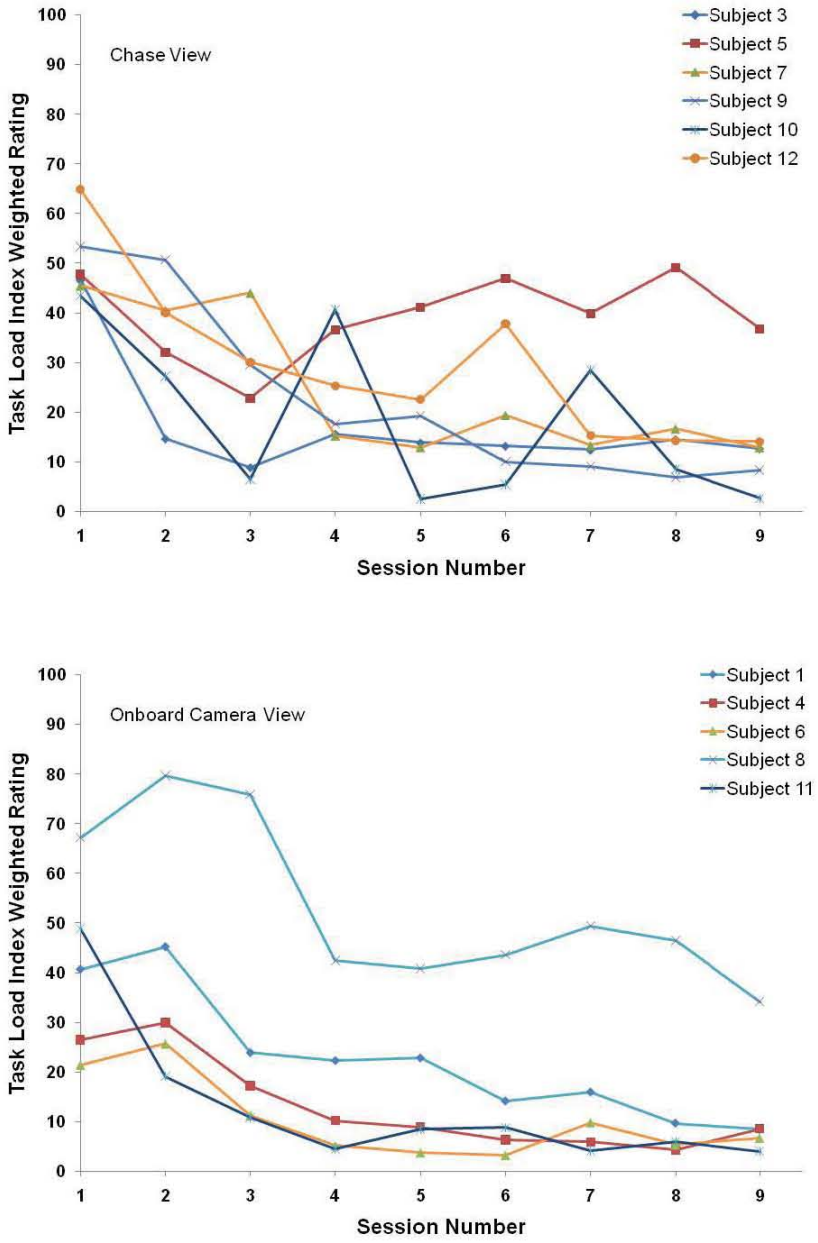

Figure 7.26: Task Load Index Weighted Rating across sessions. Top:Chase Subjects Bottom:Onboard Subjects

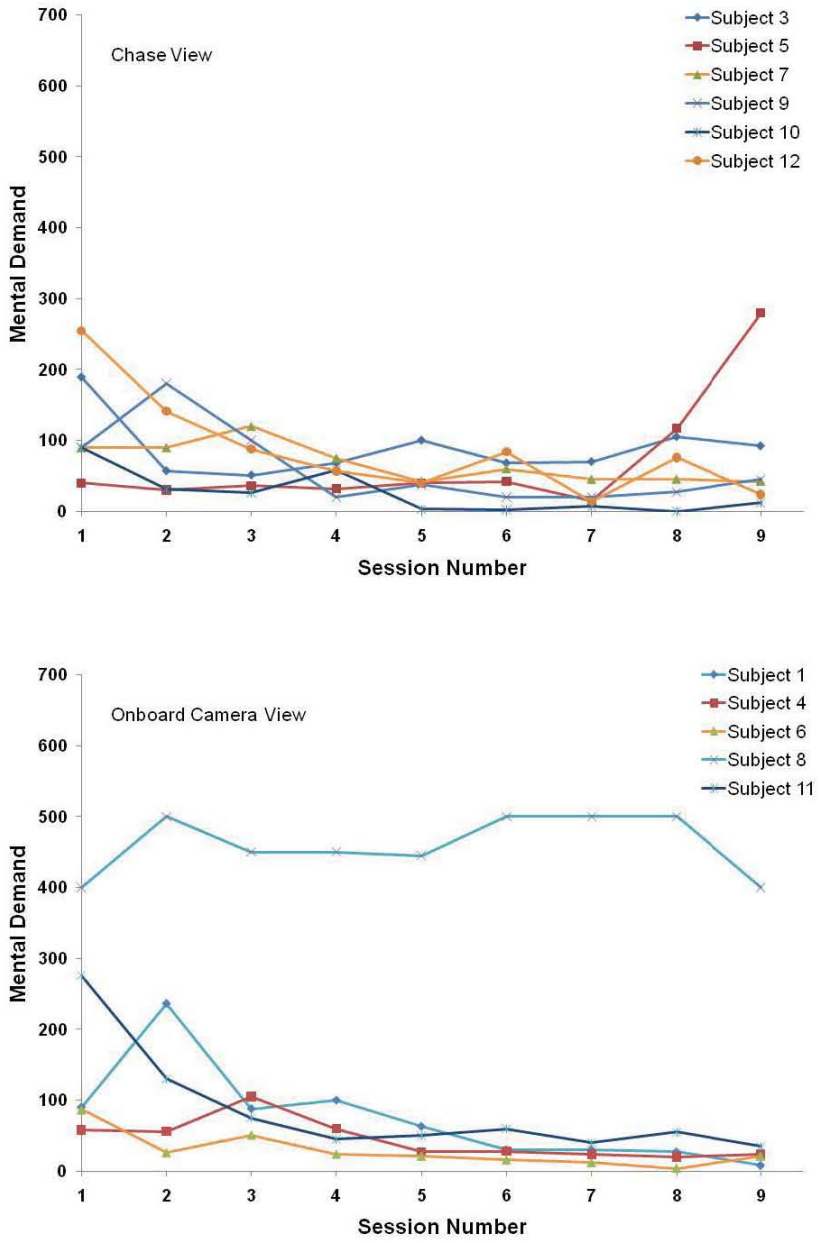

Figure 7.27: Mental Demand Rating across sessions. Top:Chase Subjects Bottom:Onboard Subjects

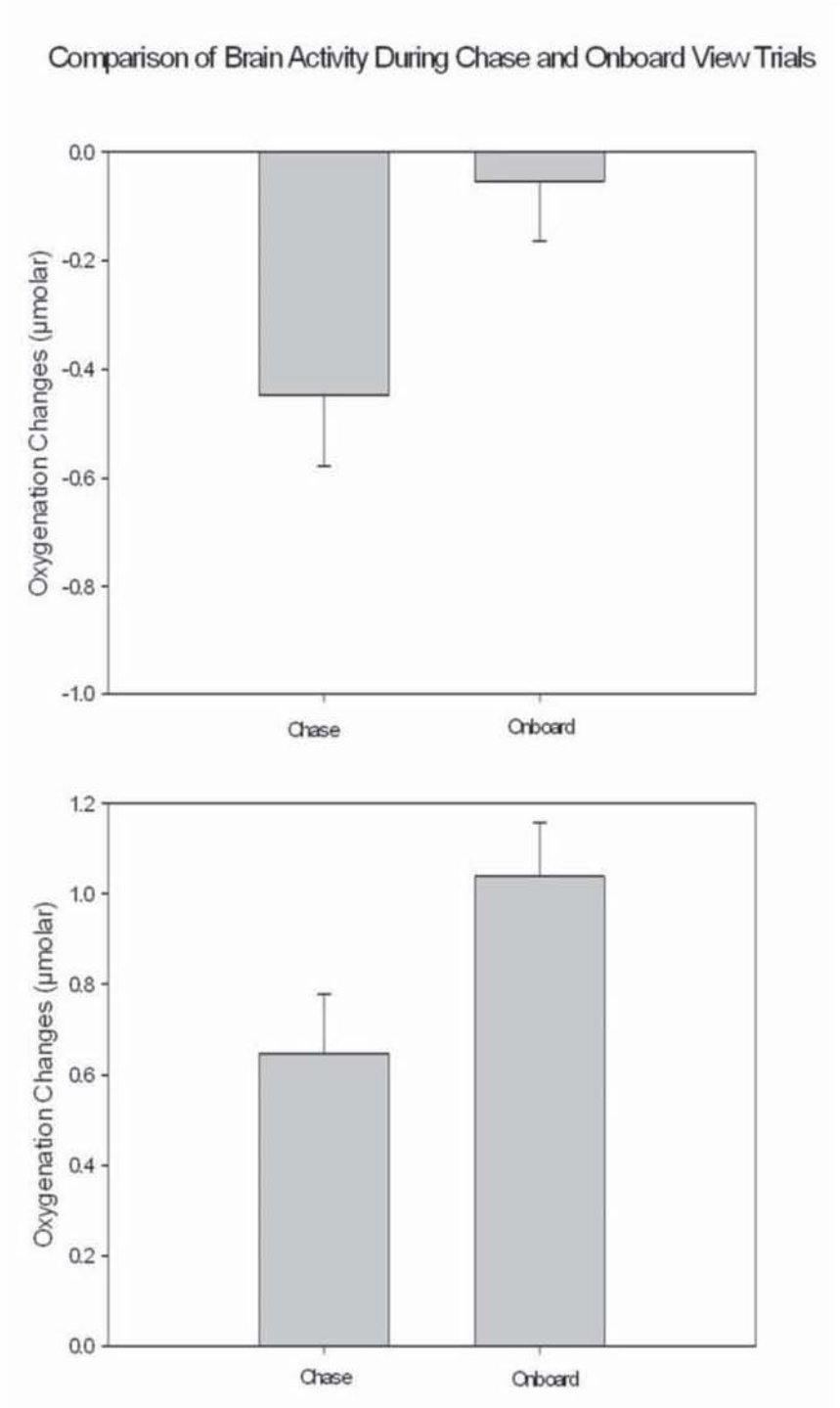

Figure 7.28: Average Oxygenation Changes for Chase and Onboard View Subjects. For comparison of the oxygenation changes, signal level is important. Top: Average Oxygenation changes for Chase view and Onboard view group. Plot shows Onboard view group's levels are higher. Bottom: Maximum Oxygenation changes for Chase view and Onboard view groups. Plot shows Onboard view group's levels are higher.

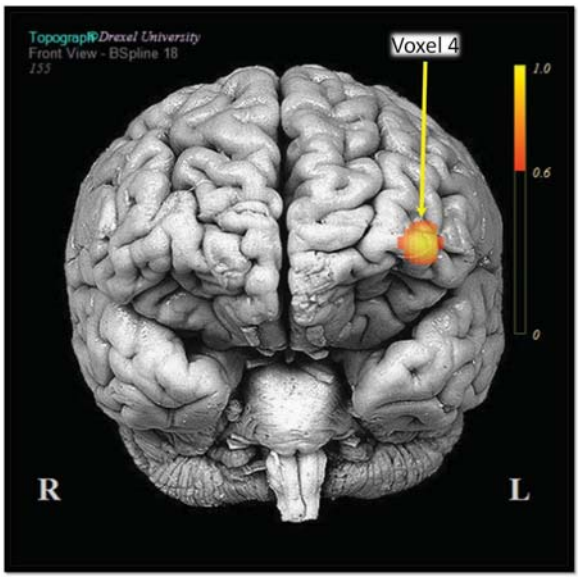

Figure 7.29: Location of the fourth voxel fNIR measurement registered on the brain surface.

aircraft's position in it. These results support Hypothesis 4.

For the Mental Demand and Overall Task Load (Weighted Rating) measures in the NASA-TLX, a Spearman correlation indicated a significant negative relationship with session for both Chase view(ρ = -0.30, p = 0.03) and Onboard view(ρ = -0.45, p = 0.00) as shown in Figure 7.30. Displaying results after session six, does not show a clear change in this negative trend. These results indicate that subjects became familiar and comfortable with the environment and tasks as the sessions progressed. In other words, workload seemed to decrease for all subjects as they learned what to expect and how to respond.

Session Ten

In session 10 the subjects performed two flights using the other view (ie. subjects in the chase view group used the onboard camera interface). The main purpose of this session was to gather opinions about the alternate view point. It was expected that

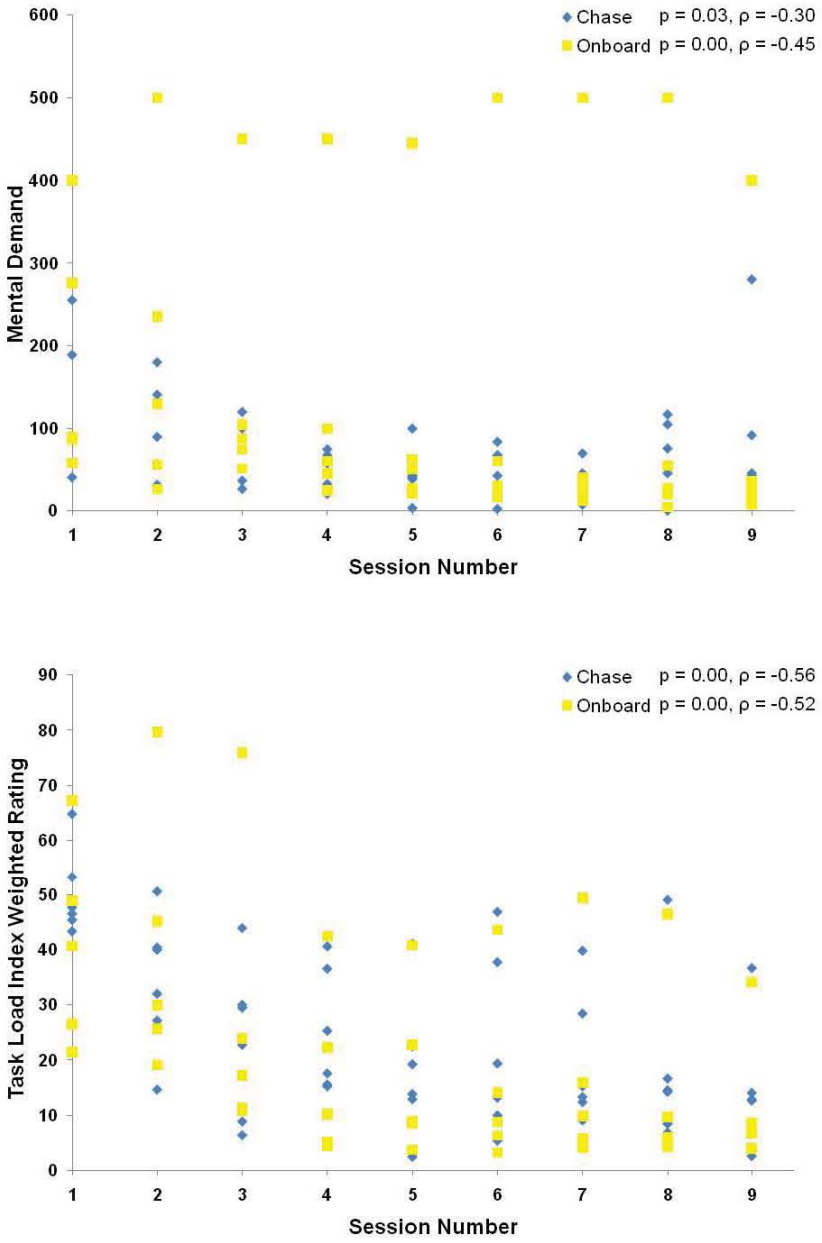

Figure 7.30: Spearman correlation of Task Load Index Weighted Rating and Mental Demand with Session. Subjects with $p<0.05$ show significant correlation. Top: Mental Demand, Bottom: Weighted Rating

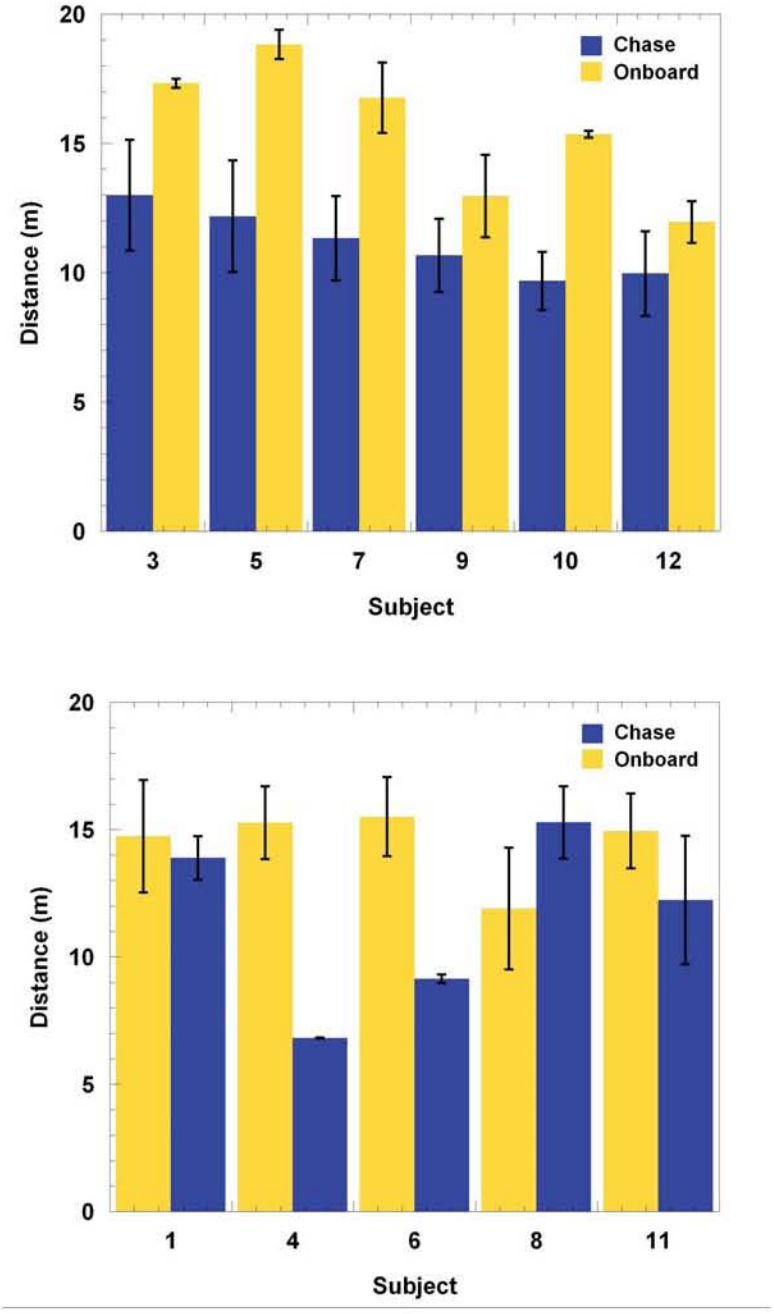

Figure 7.31: Mean distance from Pole 1 actual. The left bar represents the average distance from Pole 1 actual (during a turn around the pole) for the eight trials using the normal view, the right bar represents the average of the 2 flights using the alternate view. Top: Chase view subjects Bottom: Onboard view subjects

Table 7.6: % of Chase View Subjects Thoughts When Using Onboard Camera View

Difficulty in:	**More**	**Same**	**Less**
Completing the course	83.33(%)	16.67(%)	0
Proper altitutde	66.67(%)	33.33(%)	0
Safe distance	66.67(%)	33.33(%)	0
Smooth flight	83.33(%)	16.67(%)	0
Awareness of:	**More**	**Same**	**Less**
Extremities (eg. wings)	0	0	100
Pose (eg. roll)	16.67	16.67	66.67
Obstacle Locations	0	33.33	66.67
Interface Preference:	**Chase View**	**Onboard Camera**	
	83.33(%)	16.67(%)	

performance would decrease for each subject because they were used to operating the aircraft with their specific view point. Two flights is not enough to run a statistical analysis, however, some of the data showed an interesting trend. Hypothesis 1 has been supported by the fact that subjects took tighter turns around obstacles because of the greater awareness of the aircraft extremities. As Figure 7.31 shows, 4 out of 5 subjects who switched from an onboard camera view to a chase view (bottom of the figure) produced a tighter more efficient turn around the curve (closer distance to the obstacle). All of the chase view subjects when switching to onboard camera view (top of the figure) produced a much larger turn radius around the pole. This can be attributed to a lower awareness of the vehicle extremities and provides further support for Hypothesis 1.

After the tenth session, subjects filled out a survey about their thoughts on the view used during the session. These results are shown in Table 7.6 and Table 7.7. In summary, the majority of the subjects felt that the chase view produced better awareness of the aircraft extremities and a better awareness of obstacles in the surrounding environment. Eight out of the eleven subjects preferred the chase view interface. Two

Table 7.7: % of Onboard Camera View Subjects Thoughts When Using Chase View

Difficulty in:	**More**	**Same**	**Less**
Completing the course	60(%)	40(%)	0
Proper altitutde	80(%)	20(%)	20(%)
Safe distance	20(%)	0	80(%)
Smooth flight	60(%)	20(%)	20(%)
Awareness of:	**More**	**Same**	**Less**
Extremities (eg. wings)	100(%)	0	0
Pose (eg. roll)	40(%)	40(%)	20(%)
Obstacle Locations	60(%)	20(%)	20(%)
Interface Preference:	**Chase View**	**Onboard Camera**	
	60(%)	40(%)	

of the subjects who preferred the onboard camera view stated that they would prefer the chase view interface if it was further enhanced with similar instrumentation like the onboard camera interface had. They would also have preferred the chase view if they had more flights to get used to the change in perspective.

7.5 Indoor Tests Revisited with Rotorcraft

Small RC rotorcraft are well suited for flights in near Earth environments because of their hovering capabilities and payload capacity versus their fixed wing counterparts. However, rotorcraft are inherently unstable and much more sensitive to control inputs than most aircraft. This is especially true as the rotorcraft gets smaller in size. An average beginning RC pilot can understand the basics of fixed wing flight relatively quickly as each axis of the joystick controls the corresponding axis of the aircraft. For example, pulling down on the pitch control of the joystick will cause the elevator of the aircraft to move, resulting in the aircraft pitching up. Moving the roll axis of the joystick will move the ailerons of the aircraft, resulting in a roll of the

aircraft. Rotorcraft controls are much more tightly coupled. For example, increasing the collective causes a change in the altitude of the rotorcraft but also increases the amount of right rotation which requires a compensation with the tail rotor control (anti-torque). Precise control of rotorcraft requires a constant movement and coordination of all controls together. An introductory session and a few hours of flight time is not enough to become capable of traversing a flight path with rotorcraft in any safe fashion. Because this high skill level is difficult to attain, conducting human factor trials with a large number of subjects is challenging. In this section, studies are presented using simulated rotorcraft and the system presented in the previous section using two RC rotorcraft pilots.

7.5.1 Objectives and Hypothesis

The primary objectives of this rotorcraft study are two fold. 1) To understand how using the mixed reality interface during flights will affect the tele-operation of a small rotorcraft in a near Earth environment as compared with an onboard camera view. This is similar to the fixed wing tests however, the movement and control of the rotorcraft is very different from fixed wing flight. The findings in the fixed wing trials may not hold true for rotorcraft. 2) To understand how well a pilot would perform with the mixed reality interface under various aircraft position accuracies. Depending on what type of avionics are onboard the real world rotorcraft, and the quality of those avionics, the accuracy in the position can vary greatly, for example, from 10cm to 10m.

Rotorcraft Hypothesis 1

The first hypothesis is derived from results obtained during fixed wing trials presented in the previous sections. The hypothesis is: *The mixed reality interface will*

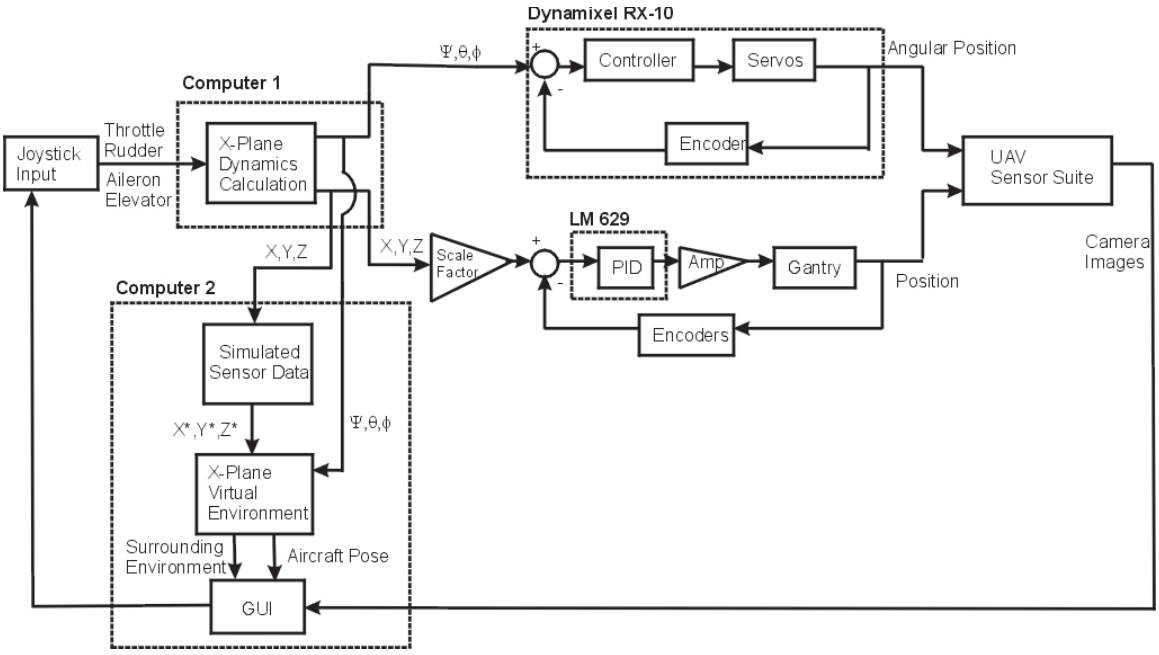

Figure 7.32: Block diagram of the indoor rotorcraft experiment system.

improve the pilot's positioning of the rotorcraft as they fly through the environment and hover over a target. Improvement consists of a safe flight path through the environment and more accurate positioning during hover flight.

Rotorcraft Hypothesis 2

The second hypothesis comes from a derivation of the results presented in [85] on the effect of conflicting cues on pilot performance. Reed showed that pilot performance decreased when the motion cues given to the pilot did not match the motion seen in the image. Based on those results the hypothesis is: *As the discrepancy of the surrounding virtual view with the onboard camera view grows, the performance of the subject will decrease.*

7.5.2 Experimental Setup

The environment for these tests is the same environment used during the fixed wing tests (Figure 7.32). The differences in the setup stem from the need to decouple the positions driving the gantry and the positions that drive the surrounding virtual view of the environment. This is necessary to be able to change the level of discrepancy between the surrounding virtual view and the onboard camera view to test Rotorcraft Hypothesis 2. For Rotorcraft Hypothesis 1, this discrepancy will be no different from the fixed wing tests.

Real Time Response and Simulated Sensor Data

As shown in Figure 7.32, the setup consists of two computers running separate executions of the flight simulation software. Computer 1 reads in operator control inputs and calculates the dynamics of the model rotorcraft during flight. It sends the rotational and scaled translational positions to the gantry controller and the YPR unit containing the camera. Computer 1 represents the real time response of the rotorcraft to operator commands. In parallel with sending data to the gantry, Computer 1 also sends the calculated translational and rotational accelerations, as well as the position data, to Computer 2 using UDP.

As mentioned in Chapter 3, SISTR was designed to integrate actual UAV hardware. In addition to the wireless camera, it is possible to attach UAV avionics to the YPR unit. Then the simulated aircraft accelerations driving the gantry arm and YPR unit could be captured by actual sensors and fed into Computer 2. The current avionics package used on our UAVs are commercial systems from Rotomotion. The avionics package integrate GPS and accelerometer data with an Extended Kalman filter to output position data. Without being able to replicate raw GPS signals to input into the avionics package, position data can not be accessed. Therefore, for this

study, it was necessary to simulate the position data that would be received from the system.

Computer 2 models the onboard sensor data of the simulated rotorcraft and feeds the data into the interface program, also running on Computer 2, to drive the virtual aircraft position and pose.

To limit the number of varying parameters during the study, the angular rotations were assumed to have been obtained from ideal sensors. Therefore, the rotorcraft pose in the mixed reality interface matched directly with the true rotorcraft pose calculated from Computer 1. Translational accelerations and position data were modified to represent different accuracies of various onboard sensor suites.

Translational Accelerations

The mixed reality interface uses position information obtained from onboard avionics to place the surrounding virtual image accordingly. Position data comes from a combination of integrating accelerometer measurements and measuring GPS position data. GPS provides accurate positioning (the level of accuracy depends on the quality of the sensor and satellite fixes) but at low update rates. The high rate of acclerometer measurements can provide accurate position information between GPS updates. However, due to noise in the signal and the inherent errors produced by integration methods alone, accelerometer measurements can lead to unbounded position error if the time interval between GPS updates becomes large.

To simulate noise in the acceleration measurements, a gaussian distributed random number generator was used with a $\mu = 0$ and $\sigma = 1$. The acceleration data was then modified as follows:

$$aNOISE_k = a_k + a_k * RAND; \tag{7.1}$$

where $aNOISE_k$ is the noisy acceleration value, a_k is the true acceleration value,

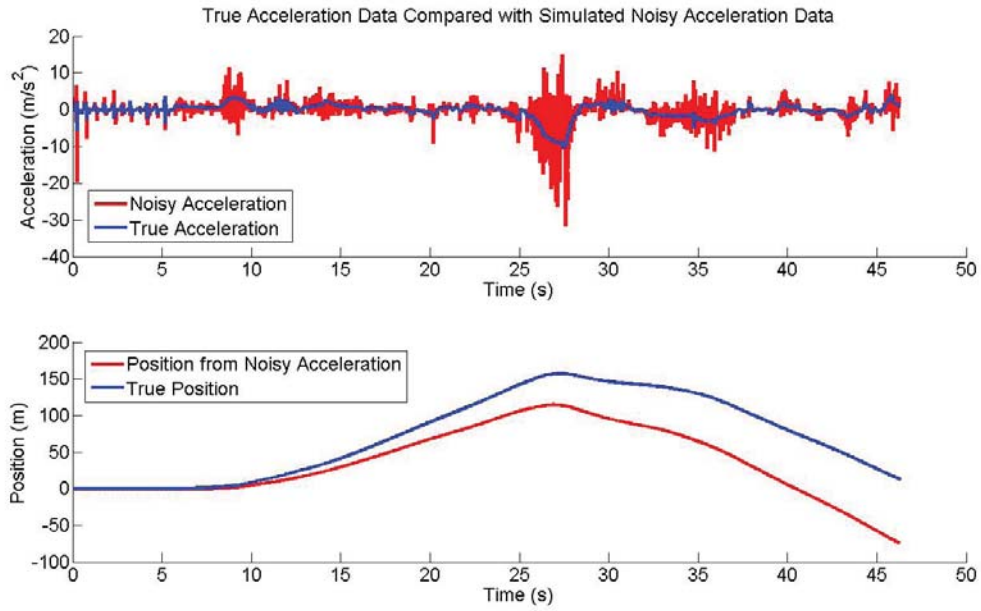

Figure 7.33: Top: True acceleration shown in blue is compared with the simulated noisy acceleration. Bottom: True position in blue is compared with the position obtained by integrating twice the noisy acceleration data.

and RAND is the random number generated. A plot of the true acceleration and the noisy acceleration can be seen in Figure 7.33.

To validate the simulated accelerations, the noisy data was integrated twice to obtain position measurements. As shown in Figure 7.33, integration of the modified accelerometer data produced drifts in the position measurement equivalent to measurements of real world accelerometers [86].

GPS

Commercial GPS sensors have a wide range of position resolutions. For example, the attitude heading and reference system sold by Rotomotion Inc uses a GPS system that has a resolution of 2 meters while certain Novatel systems can get achieve resolutions down to 10 centimeters. These values are often reported by manufacturers as

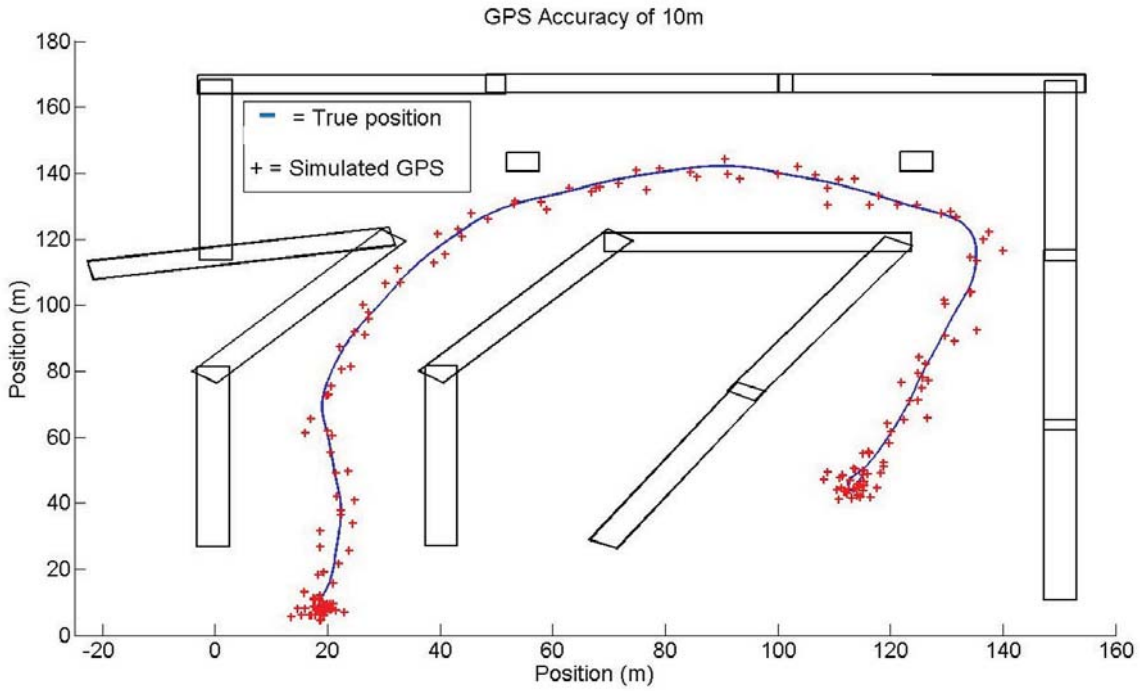

Figure 7.34: Simulated GPS position data representing an accuracy of 10 meters. The true position value is represented as a blue line and the GPS data is represented by red crosses.

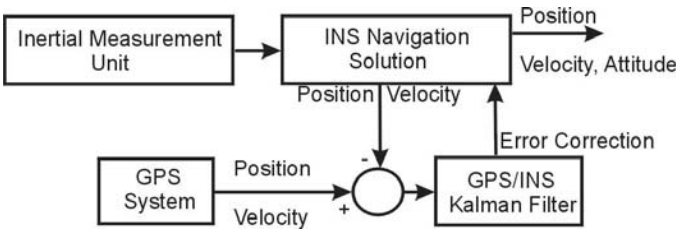

Figure 7.35: Block diagram showing a simple representation of a loosely coupled integration of GPS and Inertial Measurement Unit data.

a 95 percent probability that the position reading will fall within those limits. GPS measurements can be modeled as a gaussian distribution [87]. To simulate various levels of GPS accuracy, a gaussian distributed random number was added to the true position value with a $\mu = 0$ and various values of σ. An example of the simulated GPS data is shown in Figure 7.34.

GPS and Accelerometer Integration

Raw accelerometer and GPS data are integrated together using a number of techniques. The most common techniques are the loosely coupled method and the tightly coupled method. A block diagram of a loosely coupled integration is shown in Figure 7.35. Both methods of sensor fusion are used to obtain a better estimate of the position data than each individual sensor can give on its own [88]. A loosely coupled approach uses the output from the GPS receiver and the accelerometers as inputs to a Kalman filter. The filter outputs the estimates of the positions. A tightly coupled approach is more complex and uses multiple Kalman filters. The output of the Kalman filters are used to correct errors amongst raw GPS data and raw accelerometer data. The tight integration comes from the accelerometer measurements being used to aid in the GPS processing.

Because this study uses simulated data, it does not have information gathered

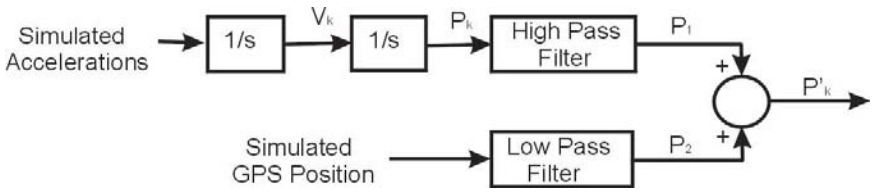

Figure 7.36: A block diagram showing the representation of a complementary filter.

from actual accelerometers and actual GPS data. The development of a Kalman filter to combine GPS and accelerometer data together requires a model of the sensors from which the data is obtained. Developing an accurate sensor model of accelerometer data and GPS data is beyond the scope of this thesis. To approximate the results of a loosely integrated GPS and accelerometer system, a complementary filter approach using the simulated GPS positions and acceleration data was used as seen in Figure 7.36.

The complementary filter is comprised of a low pass filter for the position data resulting from the simulated GPS measurement and a high pass filter for the positions obtained from the modified accelerations. The position data from GPS has a good low frequency response while the position data from the accelerometers has a good high frequency response. Each filtered output is added together to produce the final position signal as described below and in [89]:

$$P_k = G_1(s) * Pgps_k + s * G_2 * Pacc_k \tag{7.2}$$

where $Pgps_k$ is the position from the simulated GPS readings, $Pacc_k$ is the position given using the acceleration data, and:

$$G_1(s) = 1/(\tau s + 1) \tag{7.3}$$

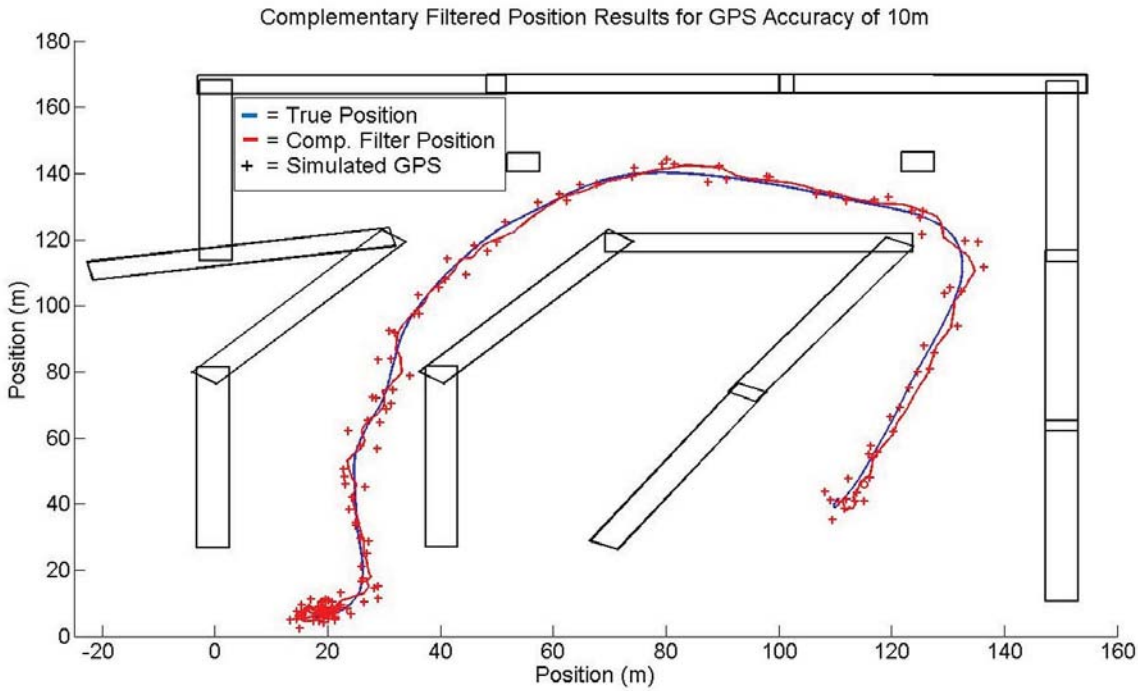

Figure 7.37: Complementary filtered position results using a simulated GPS accuracy of 10m. The true position is represented by the blue line, the complementary filter results by the red line and the simulated GPS by the red crosses. Example data is from a subject trial.

$$sG_2(s) = (\tau s)/(\tau s + 1) \tag{7.4}$$

and assuming ideal sensors:

$$G_1 + sG_2 = 1 \tag{7.5}$$

where τ is the time constant of the filters. In tests, satisfactory performance was found using a time constant of the complementary filter equal to 0.33 seconds. Results of the complementary filter can be seen in Figure 7.37 for a GPS precision of 10 meters. A screen capture highlighting the affect of the GPS accuracy in the chase view display is shown in Figure 7.38.

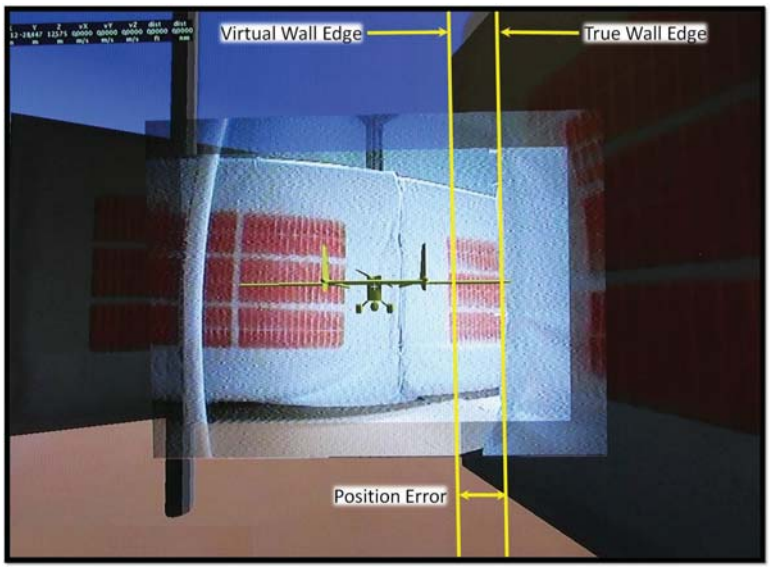

Figure 7.38: Example of position error shown in the interface due to noise and accuracy of the simulated onboard sensors. Results from ten meter GPS accuracy shown during a fixed wing test.

Rotorcraft Model

Two rotorcraft models were used for these tests. A commercially available Raptor 90 model for X-Plane was modified and used for the aerodynamic calculations representing the true aircraft flight during tests. The model was modified so the dynamics were similar to a larger size unmnanned rotorcraft. The larger size was necessary due to the physical size of the gantry arm representing the aircraft in the real world gantry environment. The aircraft displayed in the interface was dimensioned similarly to the large size unmanned rotorcraft known as the SR-200 from Rotomotion Inc. This aircraft has a 118 inch main rotor diameter.

7.5.3 Experiment

The mission is shown in Figure 7.39. The two RC helicopter pilots were asked to conduct the same task of flying through the lower level of the gantry environment

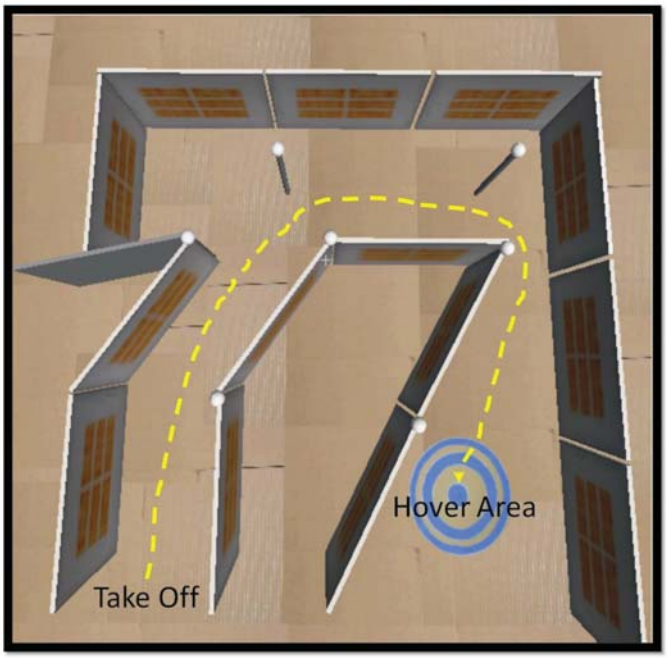

Figure 7.39: Top down view of the rotorcraft mission. The pilot is asked to take off and maintain a safe distance from the obstacles while heading toward the target. Once the pilot reaches the target, they were asked to maintain hover for at least 10 seconds.

Table 7.8: Mean Target Error in Meters for the 4 Flight Scenarios

Scenario	Pilot 1	Pilot 2
Onboard View	21.75 ± 18.06	10.83 ± 6.79
Chase View	12.47 ± 5.12	8.26 ± 1.39
2m Precision	6.14 ± 1.99	8.30 ± 6.09
10m Precision	8.89 ± 0.51	9.89 ± 9.30

with the goal of maintaining a safe distance from the obstacles. When they reached the target at the end of the lower level, they were tasked with hovering over it for 10 seconds. This task was conducted a number of times under the following scenarios: 1) Onboard camera viewpoint, 2) Mixed Reality Interface with no noise, 3) Mixed Reality Interface with GPS precision an accuracy of 2 meters, and 4) Mixed Reality Interface with GPS accuracy of 10 meters. During each flight, the positions of the rotorcraft were recorded. Scenarios were introduced in a random order as to minimize learning as best as possible.

Each pilot's performance was assessed using two measures: the average distance from the obstacles and the average distance from the center of the target during hover. Three flights per scenario for each pilot was recorded and analyzed.

Results and Discussion

For a small data set such as this, statistical analysis can not be used to prove the hypotheses. However, the data, session observations, and pilot opinions seems to lead to the following conclusions. 1) Chase view improves accuracy when hovering and 2) Discrepancies in the surrounding view due to various levels of GPS precision has little effect on performance.

Data supporting the first conclusion is seen in Table 7.8. Hovering over the target

Table 7.9: Mean Distance from Obstacles in Meters for the 4 Flight Scenarios

Scenario	Pilot 1	Pilot 2
Onboard View	15.30 ± 1.01	16.96 ± 0.87
Chase View	15.63 ± 1.51	17.53 ± 0.19
2m Precision	15.64 ± 0.57	18.89 ± 0.39
10m Precision	16.15 ± 0.59	18.16 ± 0.46

was found to be easier when using the chase view because the target was still seen on screen while hovering. When the rotorcraft is level during hover, the target does not appear in the onboard camera image unless hovering occurs at a very close distance to the ground. To check target position using the onboard camera view, the pilot had to pitch down to look at the ground which causes a movement forward and requires a counter pitch up to move back to the original position.

Data supporting the second conclusion is seen in Table 7.9. While there was a very slight change in the mean distance from the obstacles between onboard camera and chase view, the flight did not task the pilots to fly into tight areas of the environment like the fixed wing tests. More importantly, the data shows that the mean obstacle distance did not change with the degrading quality of the surrounding view. This result is slightly misleading in that it would seem the chase view interface performs well under degraded positioning. This is not the case. Discussions with the pilots revealed that the surrounding view was mostly ignored when the mismatch was high and pilot attention was focused purely on the center of the interface that contained the rotated onboard camera view. During high mismatch, the operator only used the virtual surrounding view during the hovering task to get a general approximation of where the target was located. The promising aspect of this result is that the pilot can still function at an acceptable level relying only on the rotated onboard camera

image during periods of high mismatch.

Another interesting observation, learned from discussions with the pilots after the tests, is that during flight, pilot awareness of the translational and rotational motion of the rotorcraft is obtained through optic flow in the image. Because of this coupling in the onboard camera image, it can make the mental separation of translational from rotational motion difficult at times. This is compared to the chase view where translational motion and rotational motion are decoupled in the interface by the rotation of the onboard camera image and vehicle pose.

8. Validation of the Chase View Interface in Near Earth Environments

The enhancement of situational awareness makes the chase view interface well suited for the direct piloting of unmanned aircraft for near Earth operations. This chapter describes the sensor suites and equipment platforms necessary for successful implementation of the interface in field tests. The setup varies differently from the indoor trials because real world environmental conditions, wireless data transfer, and real world aircraft dynamics are encountered. The following sections describe the test missions in more detail.

8.1 The Notional Mission

A group of UAVs are continually monitoring the borders around a top secret facility. Suspicious activity is reported at one of the building campuses. A security UAV pilot taps into the nearest UAV and flies in to survey the area. Nothing is found in the front parking lot so the UAV operator moves the aircraft to the back of the facility. Due to large structures in the rear of the facility, the UAV operator must safely fly between and around them to gather more information. Nothing out of the ordinary is found so the operator decides to place the aircraft down in an unexposed area to observe for a short while (Figure 8.1).

Airspace regulations and the inherent danger of initial tests of a newly developed rotorcraft piloting system in a populated environment required some modifications to validate aspects of the notional mission. Flight altitudes were restricted to below the tree line. Flights in the front of the facility and flights in the rear were allowed but not flights from the front of the facility to the rear. In the outdoor field test scenarios, a remote control rotorcraft pilot was in direct control of the rotorcraft and the ground

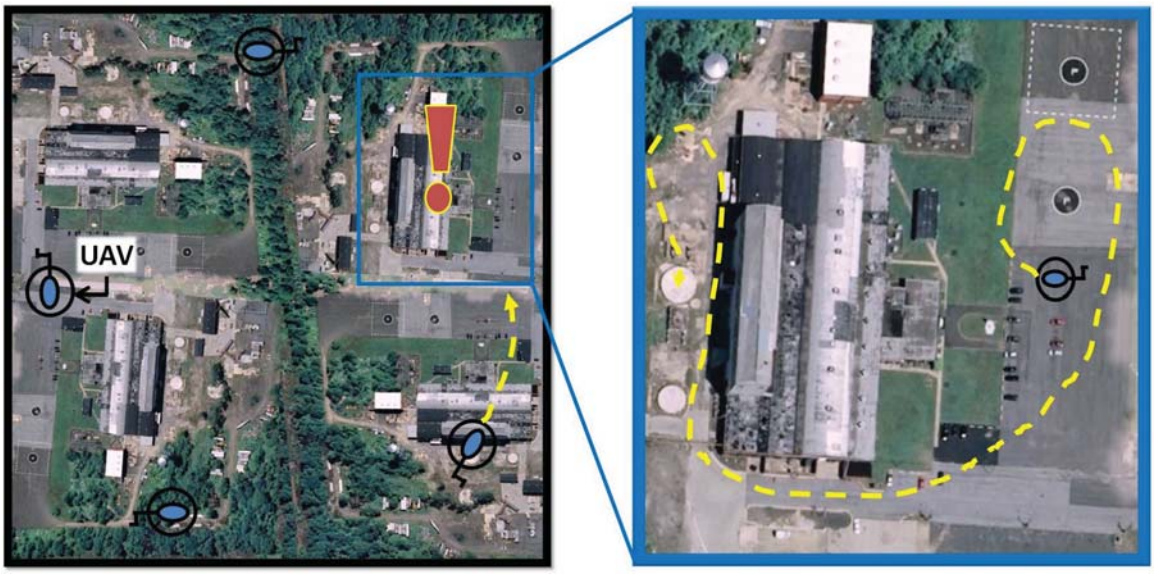

Figure 8.1: Notional mission for rotorcraft and the mixed reality interface.

station antennas maintained a line of sight with the rotorcraft at all times. Real time performance of the interface was recorded during every flight.

8.2 Field Test Equipment

8.2.1 The Aerial Platform

A Raptor 90 helicopter was used for the flight tests as seen in Figure 8.2. The Raptor 90 has a main rotor diameter of 64.75 inches and has been modified to run off of electric power. After batteries are installed, the Raptor has approximately a 15 pound payload capacity with about 20 minutes of flight time. The landing gear has been modified from stock to support the onboard sensor suite. Vibration damping mounts were installed between the helicopter and the landing gear and between the landing gear and the avionics. The pilot controls the Raptor through a nine channel 72 Mega Hertz transmitter that transmits a pulse position modulated (PPM) signal to a receiver on board the aircraft. The receiver controls the motor for the main and

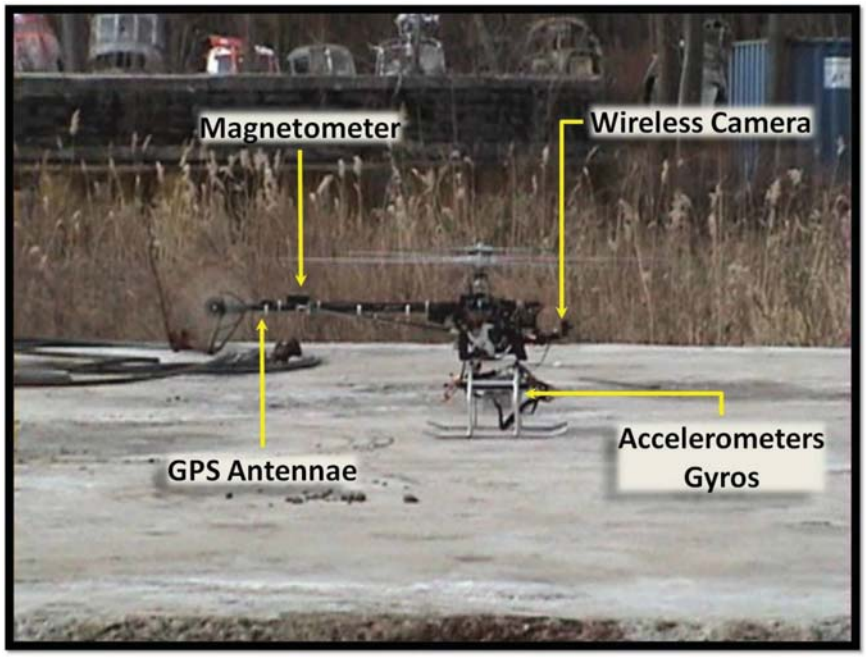

Figure 8.2: Modified Raptor 90 with new landing gear and installed avionics.

tail rotors and the servos controlling the rotorcraft swashplate.

8.2.2 The Sensor Suite

To achieve successful tele-operation using the mixed reality interface, real time aircraft state information and video images must be wirelessly transmitted to the ground station running the interface. Filtered translational and rotational positions of the aircraft in the Earth-centered, Earth-fixed frame are required.

Avionics

State information of the aircraft is obtained using an avionics package developed by Rotomotion Inc. The avionics package contains a GPS, accelerometers, gyros, and a magnetometer. All four are integrated together using Extended Kalman filters to produce accurate position and state information of the aircraft with out any drift.

Table 8.1: Choice Specifications of the Avionics Package

Specification	Value
Roll/Pitch Precision(deg)	0.5
Heading Precision(deg)	1.0
Absolute Location*(m)	2.0
*With good GPS coverage	
Relative Location(m)	0.015-0.025

In a very simplified explanation, the GPS and accelerometer data are combined such that the low frequency 2 Hertz, low resolution (2 meter) position information of the GPS is enhanced by the high frequency (100+ Hertz) and high resolution of the accelerometers. At the same time, position error caused from integrating the accelerometer data is bounded by the absolute data of the GPS. The gyros and magnetometer are integrated in a similar way to produce accurate angular position data. The current system specs can be seen in Table 8.1. The avionic package exports the Kalman filtered data at 25 Hertz.

Vision

There are a wide selection of cameras that can be used to send video images to the ground station. Some cameras like the Cannon XH-L1 HD can be mounted on board and can stream 1080i resolution video at 30 frames per second with very low latency in the video stream. These systems however can cost upwards of $50K and are not suitable for this work due to their relatively high weight. On the other side of the camera spectrum are small light weight cameras that are essentially webcams that can be configured to transmit video wirelessly. These cameras suffer from poor resolution, small field of view, high sensitivity to changing lighting conditions, and low frame rates. The camera used on board the Raptor 90 is a 70 gram, 450 line CCD

camera with a 90 degree field of view. The higher field of view causes a slight barreling distortion on the boundaries of the image, similar to a fish eye lens. However, the distortion was not found to dramatically affect pilot performance.

Data Transmission

Because the UAV is operated a distance away from the ground station, data obtained from onboard sensors is transmitted wirelessly. The video stream is transmitted using a 5.8 Giga Hertz, one Watt transmitter with a range of three miles. The avionics data is transmitted through a 2.4 Giga Hertz 802.11b Senao Multi-Client Bridge wireless bridge with an operating range of at least 180 meters line of sight (found experimentally).

Sensor Data Latency

Because the mixed reality interface fuses sensor data with the graphical interface, ideal conditions would be zero delay in the sensor data. While actual raw sensor data is relatively instantaneous, processing the data and processing through the interface program produce latencies in the system. To test the latencies in the real world system, the sensors onboard the aircraft transmitted data wirelessly to the interface while the aircraft underwent hand held rotations. A video camera was placed such that it could record both the motions of the aircraft and the results of the mixed reality interface in the same frame. The time delay measured was the time it took for the aircraft to experience the motion, and the virtual aircraft model to display the resulting motion. Analysis of the individual frames of the recorded video showed an average delay of 200 $\pm$ 33 milliseconds. The time delay for the onboard camera was tested in a similar fashion. The camera was rotated and its image was displayed using a video capture device and a direct link with the computer. The time it took

for the image integrated into the display to show the rotation was measured as the time delay. The result was 170 ± 33 milliseconds.

To following is used to illustrate how these time delays can affect a flight. A time delay of 170 milliseconds in the video image means that flying at 20 miles per hour would result in a maximum offset in the true position and the displayed position in the image of 4.93 feet. As the pilot slows down when nearing an obstacle (for safety it should be well before 4.93 feet), the maximum offset between the true position and the position displayed in the video image decreases accordingly. The indoor tests presented in earlier chapters, which were run with a 200 millisecond delay, showed the delay did not cause uncontrolled and unsafe flights. Important to note is that the delays do not cause a growing accumulation of error in the interface. At each time cycle, the interface uses the data packets that arrive at the moment it requires one. It does not run on a first input first output queue.

Certainly non-line of sight missions will require an extended network of radio towers and/or satellite communication links. These will most likely add extra latencies to the system. Chapter 9 addresses these issues and presents technologies and approaches that can be used to alleviate some of the problems that arise during long delays in data transmission. The mission experiments presented in this chapter show results that demonstrate successful operations of the interface in close range scenarios.

8.2.3 The Ground Station and Data Input

The core of the ground station does not change dramatically from the setup described previously for the indoor trials. As seen in the block diagram shown in Figure 8.3, translations and angular data received from the aircraft is used as input to the interface. This is different from the indoor analysis section where data input came from state information produced by the flight simulator. The other difference

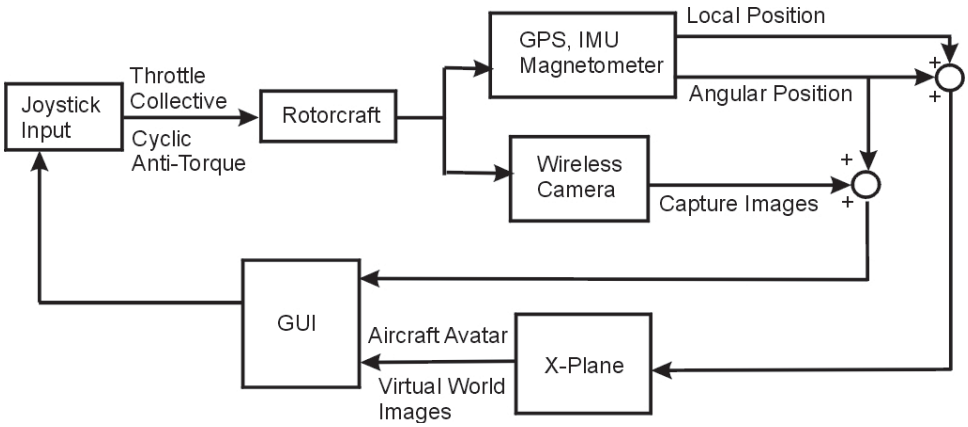

Figure 8.3: Block diagram of the Field Test system.

is that the operator joystick transmits commands directly to the aircraft and not to the flight simulation software.

The state information from the onboard avionics enters the ground station as roll, pitch and yaw in radians, latitude and longitude in degrees, altitude (mean sea level), North, East, Down position in meters. The roll, pitch, and yaw are converted to quaternions and fed into the flight sim as discussed in Chapter 3. The origin of the North, East and Down positions are located at the position where the avionics were turned on. These positions are converted to OpenGL coordinates and also fed into the flight sim to position the aircraft. Since the same wireless camera was used in the indoor trials as in the field tests, there is no difference from the indoor trials in how the signal is processed.

8.3 Virtual Models

8.3.1 Flight Environment

The notional mission represents a scenario where major obstacles such as buildings, trees, and power lines would be well known before UAV flights into the environ-

Figure 8.4: Left: Real World Environment, Right: Virtual Environment

ment. In this case, a virtual model of the Piasecki Aircraft facility can be modeled prior to the field experiments in line with the notional mission. As described in Chapter 3, the Piasecki facility was modeled by integrating satellite imagery, 3D laser scan data, and physical measurements to obtain an accurate 3D virtual representation of the flight environment. A comparison of the virtual world and real world is shown in Figure 8.4.

8.3.2 Aircraft Avatar

To represent the appropriate size and pose of the aircraft in the interface, a simplified model of the Raptor 90 with modified landing gear was created as seen in Figure 8.5. While the goal of the indoor tests was to match the physical size and dynamic response of the real world aircraft, the goal of the model used in real world tests is to accurately match the locations of the rotorcraft extremities. Aerodynamic

Figure 8.5: Model of the converted Raptor 90 used as the rotorcraft avatar in the chase view interface.

calculations from the flight simulator are not used in real world field tests and are therefore turned off in the flight sim. The flight simulator is used purely to render the orientation and location of the surrounding obstacles and vehicle pose. This is driven by position data being input into the simulator from the aircraft avionics.

8.4 Walking Trials

Before conducting flight tests, an number of experiments were conducted by walking the aircraft platform in set patterns and analyzing the avionics data. Plots of the walking patterns can be seen in Figure 8.6. As expected, satellite coverage effected the accuracy of the position. With seven or more satellites available for a fix, position data was well within the 2 meter accuracy specification posted by the manufacturer. The results collected during rectangular pattern walks and returning to the exact starting locations can be seen in Figure 8.6. The bottom figure shows an example trial at Drexel University with less than five satellite coverage and left

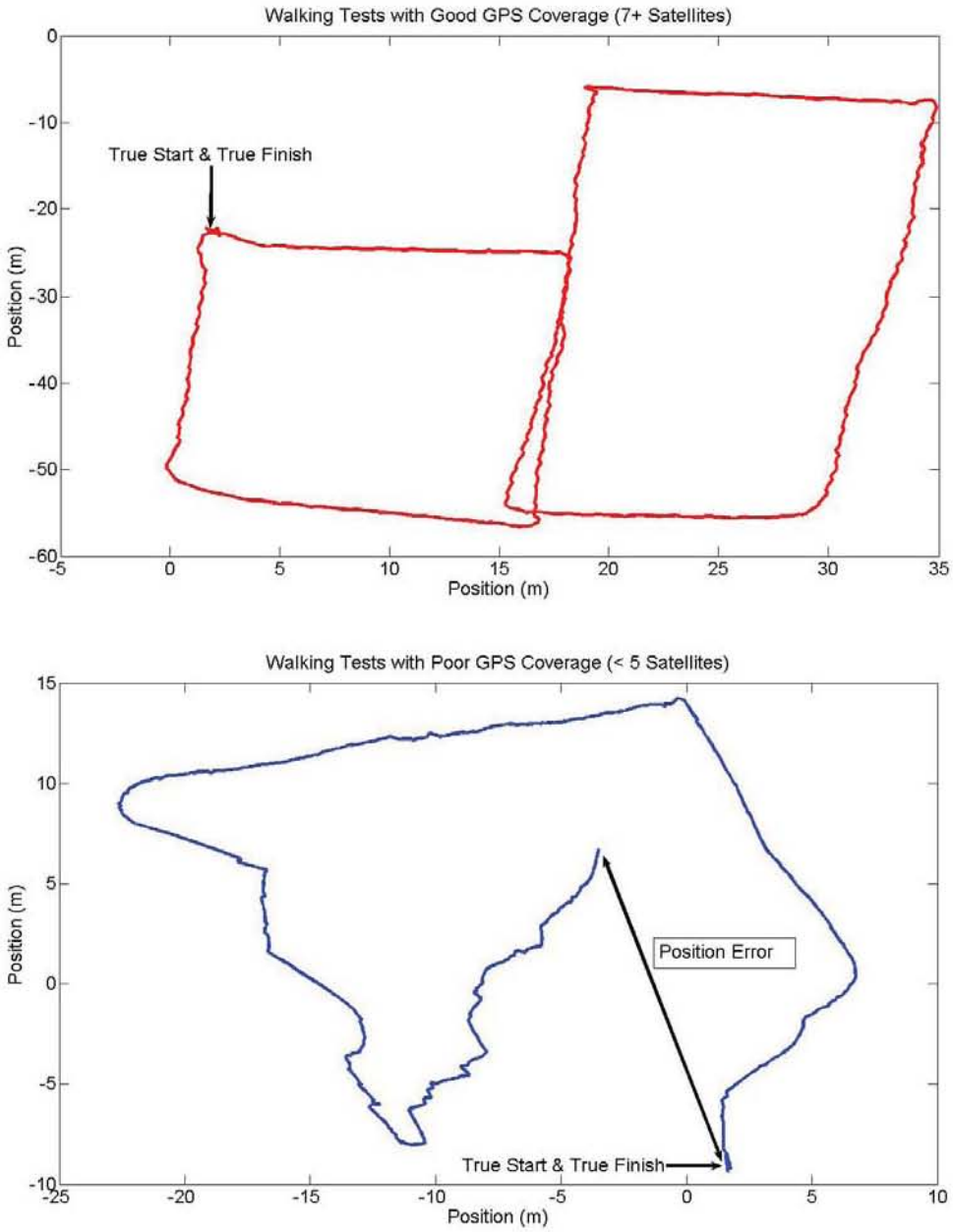

Figure 8.6: Top: Plot of position during a walking test with good GPS coverage. Bottom: Plot of position during a walking test with poor GPS coverage.

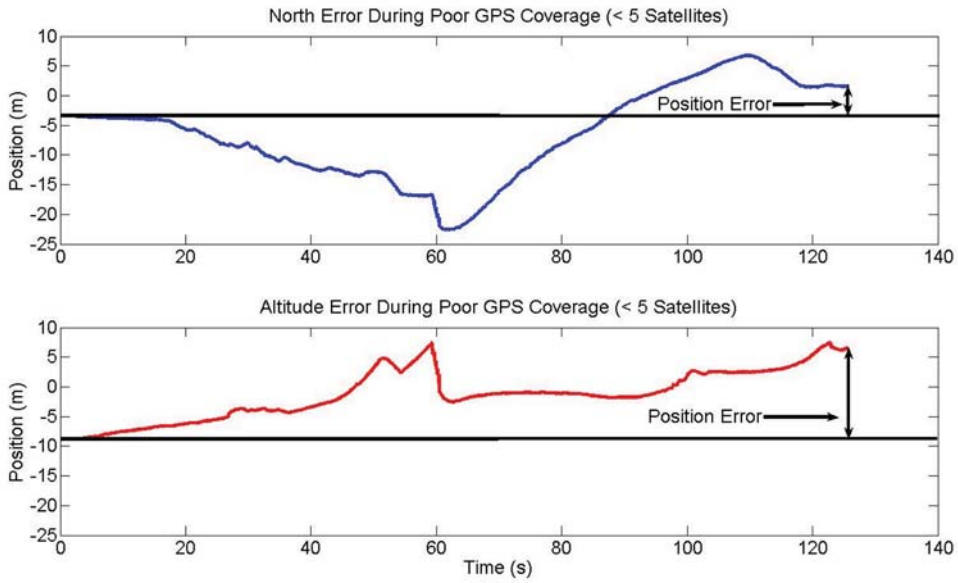

Figure 8.7: Position errors during poor GPS coverage (less than five satellites available for a fix). Data comes from rectangular pattern walking tests where the start and finish are at the same location.

shows an example trial at Piasecki Aircraft with more than seven satellites.

As shown in Figure 8.7, during periods of poor satellite coverage, the position data has a greater error in the altitude measurement than the North and East directions. Because the current avionics package only uses GPS as the absolute measure of altitude, the accuracy of the altitude can at times be greater than 1.5 times worse than the accuracy of the GPS latitude and longitude values. This mostly has to do with geometry and the way altitude is calculated from GPS satellite information. A detailed explanation of this can be found in [90].

To eliminate much of the drift and position variation prior to rotorcraft liftoff, North, East, and Down data from the avionics was ignored by the interface until the a change in yaw was detected above a defined threshold. Figure 8.8 shows an example plot of the time when the threshold is reached during take off and the the local frame of reference is set. During lift off with the rotorcraft, a quick movement in

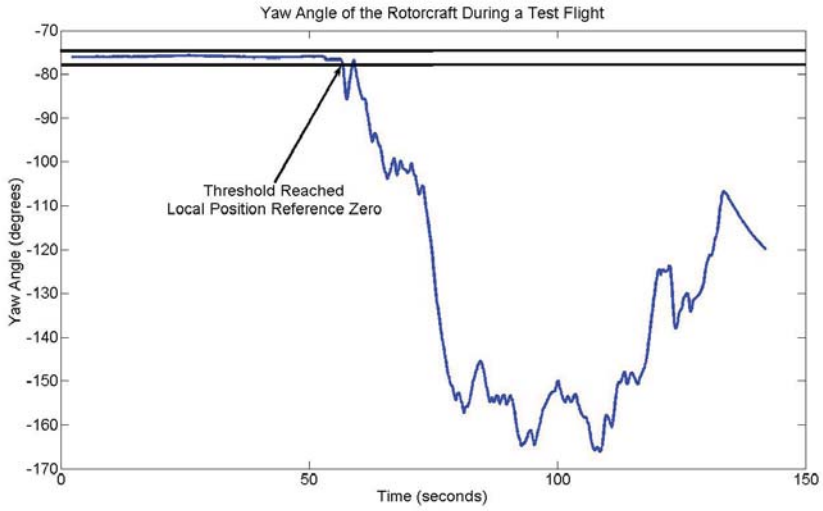

Figure 8.8: Yaw angle of the rotorcraft during test flight. The point at which the yaw angle passes a threshold value denotes the time when the local frame of reference is set.

the yaw direction occurs as soon as the aircraft lifts off the ground due to the torque produced by the main rotors. At the moment the yaw threshold is met, the North, East and Down origin is set to the current location and all further flight information is referenced from that point.

8.5 Flight Procedure

Each flight begins with the registration of the aircraft position in the virtual world with the position in the real world. With good satellite coverage, the absolute location of the aircraft with its current sensor package can be found from the GPS output to within two meters. The start location of the aircraft in the virtual world is then manually modified to a more accurate position within the two meter GPS reading. The tether distance of the virtual camera is also adjusted such that the virtual surrounding view matches the perspective of the onboard camera. Once registration is complete, all data coming into the ground station is referenced from the fixed registered frame.

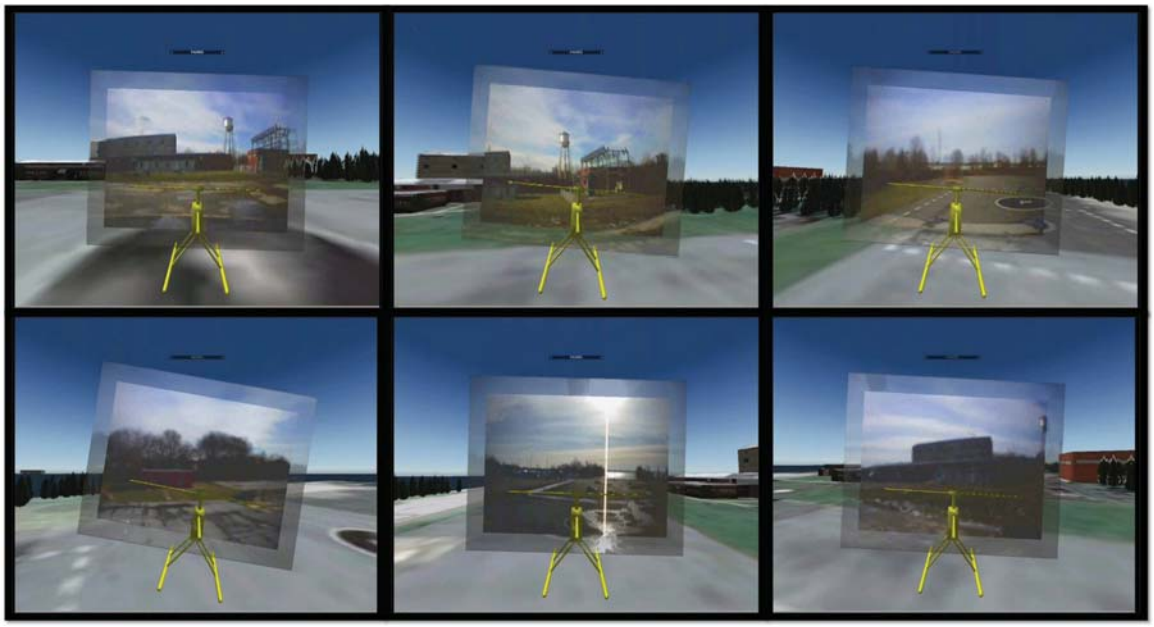

Figure 8.9: Screen captures of the chase view interface during a 360 degree pan around the front of the test facility. The sequence of snapshots goes from top left to right then bottom, left to right.

8.6 Mission Experiments

8.6.1 Open Flight

The first experiment represents the initial portion of the notional mission where the UAV pilot conducts an area surveillance in the front of the facility. The aircraft lifts off while facing the front of the main building. It then travels parallel to the front of the facility while looking toward the front of the main building. At the right side of the facility, the aircraft does a full 360 degree pan of the area, and then proceeds to land. Figure 8.9 shows screen captures of the interface during the mission.

8.6.2 Obstacle Flights

The next series of flights took place in the rear of the Piasecki facility. These flights represented the second portion of the notional mission. Multiple flights were

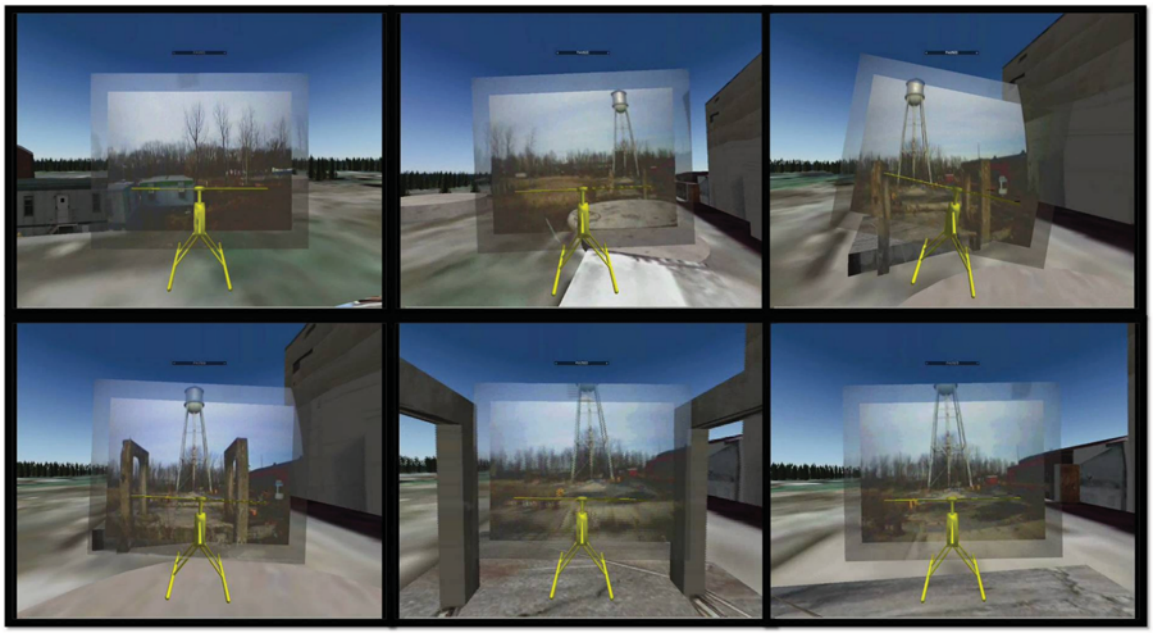

Figure 8.10: Screen captures of the chase view interface during flight between obstacles in the rear of the test facility. The sequence of snapshots goes from top left to right then bottom, left to right.

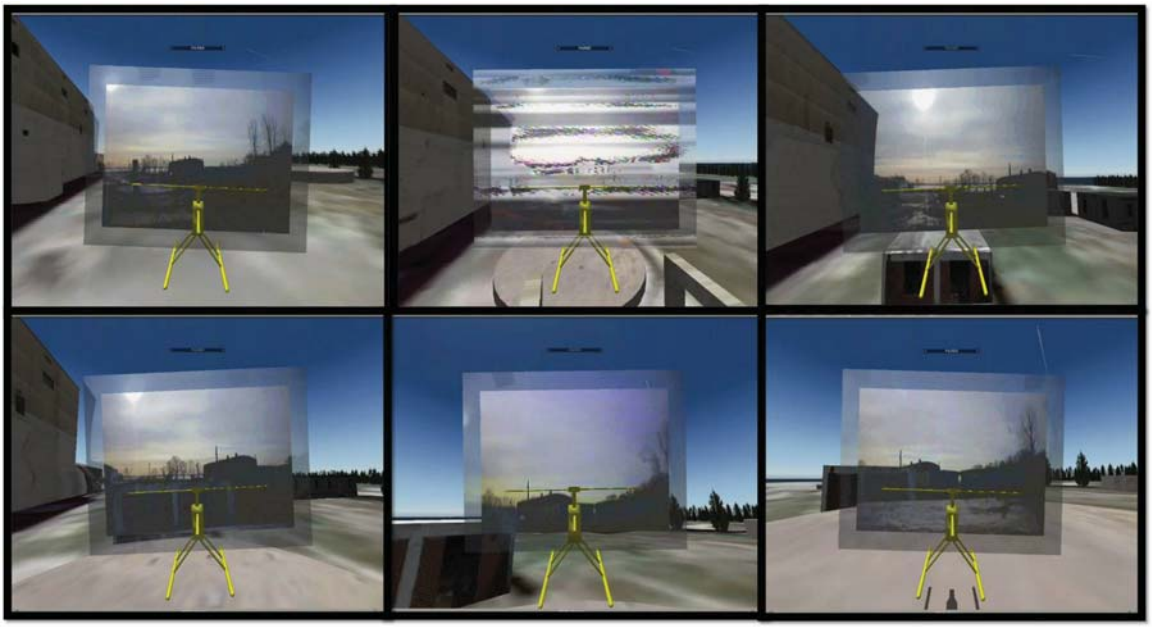

Figure 8.11: Screen captures of the chase view interface during flight around obstacles and landing on an unexposed area in the rear of the test facility. The sequence of snapshots goes from top left to right then bottom, left to right.

conducted where the operator flew over, around, and in between obstacles. In the final flight, the operator landed on the large concrete pad surrounded on three sides by obstacles, representing landing in an unexposed area. Screen captures of the interface during these missions is shown in Figure 8.10 and Figure 8.11.

8.7 Results and Discussion

Five out of six tests showed successful real time integration of the virtual world and real world data. In these tests, there was a very low degree of mismatch between the surrounding virtual view and the onboard camera. In all flights, little to no observable lag was noticed in the recreated vehicle pose or the rotated onboard camera image. This resulted in a maintained level horizon and accurate aircraft pose through out the flight as seen in Figure 8.10.

The sixth test failed due to errors in the positioning data. North, East, and Down positions are sensitive to GPS coverage even though the Extended Kalman filter fuses the accelerometer data with GPS to maintain a higher level of position accuracy. If GPS is degraded for a significant period of time, the errors of the accelerometer integrations are not as tightly bounded and the measured positions can drift toward the low resolution bounds of the GPS data. North and East measurements are more robust to specific satellites during a fix. Altitude measurements can be affected dramatically depending on which specific satellite fix is lost. Only during one flight did the number of satellites drop resulting in degraded height accuracy. This caused the virtual aircraft to drop below the surface of the ground (in the virtual view) while the true position was a few feet above. As demonstrated during the indoor trials presented in Chapter 7, a noticeable discrepancy such as this would require the pilot to focus only on the rotated onboard camera image to maintain control of the rotorcraft until position data was correctly updated. Chapter 7 tests would suggest

that while the situational awareness decreases, the control of the rotorcraft can still be maintained.

Another benefit of the chase view was observed during field tests. As shown in Figure 8.11 (top center), there were periods where the video feed would drop out or become very noisy during the flights. Using the onboard camera alone, the pilot would be flying "blind" during these outages, some of which lasted multiple seconds. However, with the chase view interface, the pilot is still able to gather awareness of position and orientation of the rotorcraft based on the pose of the avatar and the surrounding viewpoint, both of which are immune to static and video dropout. They are not however, immune to errors in the avionics signals as was described in the previous chapter.

9. Conclusions, Future Work and Enabling Technologies

9.1 Summary and Achievements

The successful record of unmanned aerial vehicles (UAVs) in the military has fueled a strong desire to adapt and increase their use in civilian applications. This will bring UAVs into near Earth environments and situations that are currently very difficult to conduct autonomously, requiring a pilot to be in direct control of the vehicle. This expanding role makes the challenges of UAV operation and the speed and efficiency of UAV pilot training more important than ever.

This work addressed the issue of UAV pilot training for operations in near Earth environments through the integration of a large indoor robotic gantry, UAV hardware, and flight simulation software. There are few training systems outside of the military for UAV operations and even fewer that focus on near Earth operations. The SISTR environment allows for the safe training of pilots and the evaluation of pilot performance in an indoor controlled atmosphere while using UAV sensor hardware and scaled models of real world flight environments. The flight simulation system was also modified and designed to recreate the various piloting viewpoints for current UAV operations. The training system was used as part of an human factors study to evaluate the effect of the pilot interface on pilot performance. The standard onboard camera results from the human performance tests (Chapter 6) can be representative of training current UAV internal pilots. Analysis of subject performances using the standard onboard camera piloting scheme, results showed a significant correlation with pilot performance variables as sessions increased using the training system. This validated the system's training and evaluation purpose.

To address the limitations associated with operating the UAV from a static ground

station, this work has presented the development of the subsystems required for the integration of motion platforms into UAV operations and training. The subsystems required for field tests are: 1) The motion transfer system which includes wireless data communication between the UAV avionics and the motion platform to relay motion of the UAV to the pilot, 2) The operator control system which includes the communication between the operator joystick inputs inside the motion platform and the control surface servo motors on the UAV, and 3) The ground station interface which includes relaying onboard video information and other vehicle state information to the pilot.

To address the limitations associated with low situational awareness caused by the onboard camera perspective, this work presented a "chase view" approach toward improving pilot situational awareness during UAV operations in near Earth environments. The chase view system enhances the limited imaging area of the onboard camera with a virtual representation of the flight environment. It also enhances pilot awareness of the vehicle by displaying a virtual representation of the size and pose of the aircraft in real time. Three dimensional spatial awareness is enhanced by horizon stabilization incorporated into the view which rotates the onboard camera image based on the roll angle of the aircraft. Rolling motion is no longer represented by rotation of the flight environment in the onboard camera image but instead by a direct view of the roll angle of the aircraft with respect to a horizontal horizon. A real time environment reconstruction method using vision was analyzed and found to be computationally expensive. An alternative approach using prior built environment models was presented and is suitable for applications where the environment does not change much from modeling to flight.

The main hypothesis for the chase view interface is that it enhances a pilot's awareness of the vehicle's extremities and three dimensional spatial location in the

flight environment. A series of studies ranging from exploration to human performance assessment were developed to test the hypothesis. Results of the studies show a significant difference between the flight paths taken by pilots using the chase view and those using the onboard camera view. The enhanced awareness allowed pilots to fly a more efficient path in a near Earth environment. Self reported preferences showed that the majority of subjects preferred the chase view interface over the traditional onboard camera perspective. All subjects reported that chase view gives a better awareness of the aircraft extremities in the flight environment and the majority report a greater awareness in the aircraft pose.

Included in these studies was a collaboration with the Drexel Brain Optical Imaging Laboratory that introduced the fNIR sensor into the evaluation and analysis of pilot performance. During the study, the fNIR sensor measured a subject's brain activity and produced an objective assessment of the subject's cognitive workload. This result was used to enhance the self reported (subjective) workload surveys. Analysis of the fNIR data found that Chase view subjects' average oxygenation levels for voxel four was significantly lower than Onboard view subjects, revealing that subjects using the onboard camera view were using more mental resources to conduct the flights. This result is most likely attributable to the narrower viewable angle and rolling of the environment in the onboard view. This requires more cognitive processing by the subject to construct an accurate working mental model of the environment and the aircraft's position in it. The benefit of a lower cognitive workload using the chase view interface is that a pilot would have more mental resources available to handle any warnings, system faults, or other unexpected events that might occur during the flight.

The indoor tests of the chase view interface were validated with successful field tests using UAV hardware. The mock mission flights showed good performance of

the chase view interface during periods of good GPS coverage. When the GPS signal dropped in quality, discrepancies in the surrounding view of the interface became high. Indoor trials showed that pilots can overcome the discrepancies by focusing on the center, real world image portion of the chase view interface until the signal improved.

The mixed reality approaches presented in this thesis follow studies on human factors performance and cognitive loading. The resulting designs serve as test beds for studying UAV pilot performance, creating training programs, and developing tools to augment UAV operations and minimize UAV accidents during operations in near Earth environments. Also, as users of UAVs move toward newer and untested applications, data about operator cognitive workload and situational awareness become very important aspects of safe UAV operation. Adding some measure of brain activity to the selection, training, and operation of UAV pilots could greatly improve the resolution of any assessments involved therein.

9.2 Future Work and Enabling Technologies

Since UAV operations in near Earth environments is still a young area of research, there are many avenues that can be explored. The motion platform UAV training system described in Chapter 4 certainly has potential to become a significant study of its own. The human performance and assessment study presented in this thesis could also be continued to evaluate the affects of pilot performance with data lag, environmental effects, and other mission scenarios. The following sections present a few areas for modifications to the current Chase view system that will further improve it's benefits.

9.2.1 Interface Improvements

Because three dimensional information (aircraft location in the environment) is being relayed on a two dimensional display, some perspective problems occur. The perspective errors are overcome through training, but they can be further eliminated through the use of a three dimensional interface. Since the chase view is created from three dimensional data, the surrounding virtual environment and aircraft pose can easily be presented in three dimensions using a 120 Hertz refresh rate monitor and shutter goggles. Essentially, depth in the view is created by displaying the left eye perspective in one frame and the right eye perspective in the second frame resulting in a three dimensional image at 60 Hertz. This is a very promising direction. Without 3D goggles, the perspective error can be decreased in the 2D image by presenting a drop down line/shadow from the aircraft avatar.

With respect to rotorcraft control, the chase view interface might produce improved control using a lose virtual camera tether instead of a rigid tether that it currently has. RC pilots, in general, are experienced with controlling the rotorcraft from an external view. Chase view displays an external view that maintains a "tail in" view of the rotorcraft at all times. However, being able to see the rotorcraft pitch with respect to a stationary horizon gives them a feel for how much the rotorcraft will move. With the current chase view, the virtual camera moves with the pitch of the aircraft so the horizon moves and the viewpoint with the rotorcraft stays the same. The modification would require that the onboard camera image, which currently is fixed to the center of the display, be able to move around the display (as well as warp) to match the direction the rotorcraft is pointing.

9.2.2 Sensor Suites

With a slightly larger rotorcraft (payload capacity of 30 pounds or more) than the one used in this study, a high-speed 3D laser scanner can be carried onboard for fast and accurate real time 3D point cloud mapping of the surrounding environment. Systems such as the VQ - 480 from Riegel can produce high accuracy 3D scans up to 100 scans per second. Granted, all the data would have to be streamed wirelessly to the ground station computer which would require tremendous bandwidth. Alternatively, if the 3D point clouds are converted into planar features similar to what was described in Chapter 5, the required bandwidth could be reduced significantly. This would produce an accurate current map of the flight environment for creating the chase view interface.

The chase view interface itself relies on accurate position information from the aircraft sensors. Accurate sensor information is heavily reliant on the quality of sensors being used. The sensors used in the field study for this work utilized a commercial grade IMU system with commercial grade GPS. This led to a global accuracy of two meters under good GPS with a local accuracy of 20 centimeters from the IMU. The interface performed well under these conditions however when GPS signal degrades, the position accuracy degrades as well, especially in the altitude readings. There is the option of using a navigation grade inertial navigation system (INS) that is much more immune to GPS outages but are almost prohibitively expensive as well as too heavy for a small UAV. There are also options to use higher grade GPS antennas that can give global accuracy readings down to ten centimeters or less but are also very expensive. An alternative approach, specifically for better altitude measurements, would be height aided positioning through the integration of a barometric altimeter into the UAV suite. While a barometric altimeter is susceptible to errors caused by changes in air density, it is not affected by GPS coverage. Therefore, the two sensing

modalities can complement each other.

For significant GPS outages, a large amount of research has been conducted on how to still maintain accurate positioning. Operations in near Earth environments have the potential of GPS disruptions due to buildings blocking the direct view of the satellites. At times the GPS signal may become unreliable eliminating correction in the position measurement of the IMU, causing the virtual world image and real world image to become misaligned. If the GPS dropout is infrequent and occurs for only short periods of time, the pilot may be able to operate successfully during the misalignment periods. However, if the GPS dropout is for a significant period, the error between the images would hinder the chase view system useless. Fortunately, there has been plenty of research toward navigation without GPS which could be leveraged during GPS dropout.

There are multiple methods for navigation during GPS dropouts. These methods are based either on a known flight environment or and unknown flight environment. They however, require much more computational power than when GPS is available so fast processing speeds are necessary for safe operations.

GPS Denied In a Known Environment

When the environment of operation is known, meaning there is some type of map data available, there are a number of methods that can be used to fix the IMU drift. Cruise missile systems use the Terrain Contour Matching system which essentially uses onboard radar altimeter readings with pre-stored digitized terrain elevation maps to account for the IMU errors [91]. A more suitable method for this mixed reality chase view system would be a scene/image matching correlation system which would use the onboard camera image and compare it with the image from the virtual world map to adjust for error. This could be integrated with visual odometry and the IMU

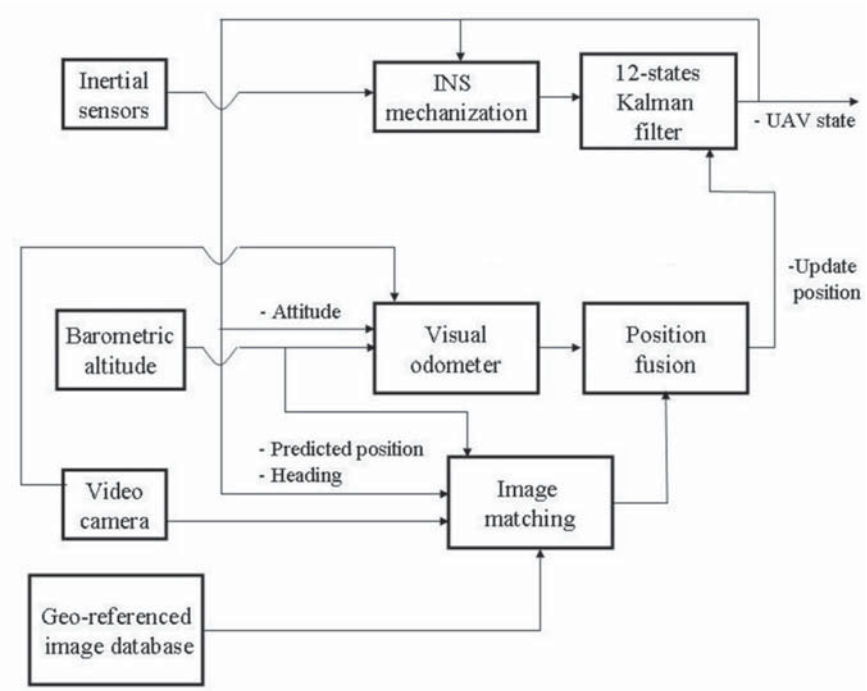

Figure 9.1: An example of a method to obtain aircraft position in a GPS denied environment. Adapted from [15]

measurements as seen in Figure 9.1 to produce accurate positioning results until the GPS signal becomes available [15].

GPS Denied in an Unknown Environment

If ever a pilot was in the situation where GPS is denied and he or she was operating in an unknown environment, methods are still available for navigation. With the chase view interface, the operator still has the real world camera view that can be used to traverse through the unknown environment. The flight would be difficult and dangerous because they would be limited to only the view from the onboard camera which suffers from the limitations described in this thesis. However, with incredible pilot skill and fast reflexes it could be done. Otherwise, one could implement a simultaneous and localization and mapping (SLAM) algorithm. This would not only

generate the map but also give the position of the aircraft [92]. The difficulty with SLAM however is that it requires "closing the loop" meaning that you must revisit landmarks for the accuracy of the map to converge.

Sensor Latencies

While this system performed well in close range tests, in its current state, it will experience issues during delays in the data transmission. Lag in the visual and control information can cause pilot performance to decrease. This chase view interface can be further developed to account for possible delays. A potential solution would be to utilize the flight simulator aerodynamics calculation capabilities rather than utilizing it only as a visual platform. An accurate model of the aircraft can be developed and used to predict the aircraft's motions and location in the environment during periods of delay. When data information finally arrives, a Kalman filter or some other type of algorithm can be used to update the predicted aircraft position. If the visual feed is delayed, the aircraft displayed in the interface can "fly" into the previous image in a third person perspective until the frame is updated. This would be similar to the approach used for ground vehicles presented by Sugimoto et al. [8].

Bibliography

[1] DepartmentofDefense, August 2005. Unmanned aircraft systems roadmap 2005-2030. Tech. rep.

[2] Weibel, R. E., and Hansman, R. J., 2005. Safety considerations for operation of unmanned aerial vehicles in the national airspace system. Tech. Rep. ICAT-2005-1, MIT International Center for Air Transportation.

[3] Shappell, S., and Wiegmann, D., 2000. The human factors analysis and classification system - hfacs. Tech. rep., Office of Aviation Medicine.

[4] Siciliano, B., and Khatib, O., 2007. *Handbook on Robotics*. Springer-Verlag, Berlin Heidelberg.

[5] Tso, K. S., Tharp, G. K., Tai, A. T., Draper, M. H., Calhoun, G. L., and Ruff, H. A., 2003. "A human factors testbed for command and control of unmanned air vehicles". Vol. 2 of *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, Institute of Electrical and Electronics Engineers Inc., pp. 8.C.1/1–8.C.1/12.

[6] Sheik-Nainar, M. A., Kaber, D. B., and Chow, M. Y., 2005. "Control gain adaptation in virtual reality mediated human-telerobot interaction". *Human Factors and Ergonomics in Manufacturing,* **15**(3), pp. 259–74.

[7] Calhoun, G. L., Draper, M. H., Abernathy, M. F., Delgado, F., and Patzek, M., 2005. "Synthetic vision system for imroving unmanned aerial vehicle operator situation awareness". In Enhanced Synthetic Vision, J. G. Verly, ed., Vol. 5802, pp. 219–230.

[8] Sugimoto, M., Kagotani, G., Nii, H., Shiroma, N., Matsuno, F., and Inami, M., 2005. "Time follower's vision: a teleoperation interface with past images". *IEEE Computer Graphics and Applications,* **25**(1), pp. 54–63.

[9] Nielsen, C. W., Goodrich, M. A., and Ricks, R. W., 2007. "Ecological interfaces for improving mobile robot teleoperation". *IEEE Transactions on Robotics,* **23**(5), pp. 927–941.

[10] Quigley, M., Goodrich, M. A., and Beard, R., 2004. Semi-autonomous human-uav interfaces for fixed-wing mini-uavs, September 28 - October 2.

[11] Narli, V., and Oh, P. Y., 2006. "Hardware-in-the-loop test rig to capture aerial robot and sensor suite performance metrics". In IEEE International Conference on Intelligent Robots and Systems, p. 2006.

[12] Schreiber, B. T., Lyon, D. R., Martin, E. L., and Confer, H. A., 2002. Impact of prior flight experience on learning predator uav operator skills. Tech. rep., Air Force Research Laboratory Human Effectiveness Directorate Warfighter Training Research Division.

[13] Charlton, S., and O'Brien, T., 2002. *Handbook of Human Factors Testing and Evaluation*. Lawrence Erlbaum Associates, Mahwah, NJ.

[14] Izzetoglu, K., 2008. "Neural correlates of cognitive workload and anesthetic depth: fnir spectroscopy investigation in humans". PhD thesis.

[15] Conte, G., and Doherty, P., 2008. An integrated uav navigation system based on aerial image matching, March.

[16] Cook, K. L. B., 2007. "The silent force multiplier: the history and role of uavs in warfare". 2007 IEEE Aerospace Conference, IEEE, pp. 1–7.

[17] Anderson, J. D., 2000. Samuel pierpont langley - america's first aeronautical engineer.

[18] DepartmentofDefense, 2007. Unmanned systems roadmap 2007-2032. Tech. rep.

[19] Oh, P. Y., Valavanis, K., and Woods, R., 2008. Uav workshop on civilian applications and commercial opportunities.

[20] Wong, K., Bil, C., Gordon, G., and Gibbens, P., 1997. Study of the unamnned aerial vehicle market in australia. Tech. rep.

[21] Zaloga, S., November 2004. "Special report: World market for unmanned systems". *Unmanned Systems,* **22**(6), pp. 22–33.

[22] Sayers, C., 1999. *Remote Control Robotics*. Springer-Verlag, New York.

[23] Endlesy, M., 1988. "Design and evaluation for situation awareness enhancements". In Proceedings of the Human Factors Society 32nd Annual Meeting, pp. 97–101.

[24] Dixon, S. R., and Wickens, C. D., 2003. Control of multiple-uavs: A workload analysis.

[25] Kaber, D. B., Onal, E., and Endley, M. R., 2000. "Design of automation for telerobots and the effect on performance, operator situation awareness, and subjective workload". *Human Factors and Ergonomics in Manufacturing,* **10**(4), pp. 409–430.

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

[27] Wiegmann, D. A., Goh, J., and O'Hare, D., 2001. "The role of situation assessment and flight experience in pilot's decisions to continue visual flight rules flight into adverse weather". *Human Factors,* **44**(2), pp. 189–197.

[28] Erp, J. B. v., 1999. Controlling unmanned vehicles: the human factors solution, 26-28 April 1999.

[29] Glumm, M., Kilduff, P., and Masley, A., 1992. A study on the effects of lens focal length on remote driver performance. Tech. rep., Army Research Laboratory.

[30] Kim, W. S., Hannaford, B., and Fejczy, A. K., 1992. "Force-reflection and shared compliant control in operating telemanipulators with time delay". *IEEE Transactions on Robotics and Automation,* **8**(2), pp. 176–85.

[31] Williams, K. W., 2004. A summary of unmanned aircraft accident/incident data: Human factors implications. Tech. Rep. DOT/FAA/AM-04/24, US Department of Transportation Federal Aviation Administration, Office of Aerospace Medicine.

[32] Schmidt, J., and Parker, R., 1995. Development of a uav mishap factors database, July.

[33] Rash, C. E., Leduc, P. A., and Manning, S. D., 2006. "Human factors in u.s. military unmanned aerial vehicle accidents". *Human Factors of Remotely Operated Vehicles,* **7**, pp. 117–131.

[34] Seagle, J., 1997. "Unmanned aerial vehicle mishaps: A human factors approach". PhD thesis.

[35] Manning, S., Rash, C., LeDuc, P., Noback, R., and McKeon, J., 2004. The role of human causal factors in us army unmanned aerial vehicle accidents.

[36] McCarley, J., and Wickens, C. D., 2005. Human factors implications of uavs in the national airspace. Tech. Rep. AHFD-05-05/FAA-05-01, Aviation Human Factors Division Institute of Aviation, University of Illinois.

[37] Fong, T., and Thorpe, C., 2001. "Vehicle teleoperation interfaces". *Autonomous Robots,* **11**(1), pp. 9–18.

[38] Ferrell, W. R., and Sheridan, T. B., 1967. "Supervisory control of remote manipulation". *IEEE Spectrum,* **4**(10), pp. 81–88.

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

[40] Tvaryanas, A., 2004. Usaf uav mishap epidemiology, 1997-2003.

[41] Scherer, S., Singh, S., Chamberlain, L., and Elgersma, M., 2008. "Flying fast and low among obstacles: Methodology and experiments". *International Journal of Robotics Research,* **27**(5), pp. 549–574.

[42] Griffiths, S., Saunders, J., Curtis, A., McLain, T., and Beard, R., 2007. *Obstacle and Terrain Avoidance for Miniature Aerial Vehicles*, Vol. 33 of *Advances in Unmanned Aerial Vehicles*. Springer Netherlands.

[43] Williams, K. W., 2006. "Human factors implications of unmanned aircraft accidents: Flight-control problems". *Human Factors of Remotely Operated Vehicles,* **7**, pp. 105–116.

[44] Ruff, H. A., Draper, M. H., Poole, M., and Repperger, D., 2000. "Haptic feedback as a supplemental method of altering uav operators to the onset of turbulence.". In Proceedings of the IEA 2000/ HFES 2000 Congress, pp. 3.14–3.44.

[45] Lam, T. M., D'Amelio, V., Mulder, M., and van Paassen, M. M., 2006. "Uav teleoperation using haptics with a degraded visual interface". 2006 IEEE Conference on Systems, Man, and Cybernetics, IEEE, pp. 2440–5.

[46] Feng, S., Xu, M., Zhao, D., Zhang, H., Tang, X., and Weng, G., 2006. "Research on rbf-pid control for the 6-dof motion base in construction tele-robot system". IMACS Multiconference on "Computational Engineering in Systems Applications", CESA, Inst. of Elec. and Elec. Eng. Computer Society, pp. 1915–1920.

[47] Chen, J. Y. C., Haas, E. C., and Barnes, M. J., 2007. "Human performance issues and user interface design for teleoperated robots". *IEEE Transactions on Systems, Man, and Cybernetics-Part C (Applications and Reviews),* **37**(6), pp. 1231–45.

[48] Wickens, C. D., 1999. *Frames of Reference for Navigation.* MIT Press, Cambridge, MA.

[49] Alfano, P., and Michel, G., 1990. "Restricting the field of view: Perceptual and performance effects". *Perceptual and motor skills,* **70**, pp. 35–45.

[50] Lasswell, J., and Wickens, C., 1995. The effects of display location and dimensionality on taxi-way navigation. Tech. Rep. Tech.Report ARL-95-5/NASA-95-2, University of Illinois Aviation Reseach Lab.

[51] Wang, W., 2004. "Human navigation performance using 6 degree of freedom dynamic viewpoint". PhD thesis.

[52] Salamin, P., Thalmann, D., and Vexo, F., 2006. "The benefits of third-person perspective in virtual and augmented reality". Proceedings of the ACM Symposium on Virtual Reality Software and Technology, VRST, Association for Computing Machinery, pp. 27–30.

[53] Wickens, C., and Prevett, T., 1995. "Exploring the dimensions of egocentricity in aircraft navigation displays". *Journal of Experimental Psychology: Applied,* **1**(2), pp. 110–135.

[54] Demiralp, C., Jackson, C. D., Karelitz, D. B., Zhang, S., and Laidlaw, D. H., 2006. "Cave and fishtank virtual-reality displays: a qualitative and quantitative comparison". *IEEE Transactions on Visualization and Computer Graphics,* **12**(3), pp. 323–30.

[55] Kadavasal, M. S., and Oliver, J. H., 2009. "Sensor augmented virtual reality based teleoperation using mixed autonomy". *Journal of Computing and Information Science in Engineering,* **9**(1), p. 014502 (5 pp.).

[56] Tadema, J., Koeners, J., and Theunissen, E., 2006. "Synthetic vision to augment sensor-based vision for remotely piloted vehicles". In Enhanced and Synthetic Vision, Vol. 6226, SPIE-Int. Soc. Opt. Eng., pp. 62260D–1–10.

[57] Draper, M., Calhoun, G., and Nelson, J., 2006. Evaluation of synthetic vision overlay concepts for uav sensor operations: Landmark cues and picture-in-picture. Tech. Rep. AFRL-HE-WP-TP-2006-0038, Human Effectiveness Directorate Warfighter Interface Division Wright-Patterson AFB.

[58] Drury, J. L., Richer, J., Rackliffe, N., and Goodrich, M. A., 2006. Comparing situation awareness for two unmanned aerial vehicle human interface approaches. Tech. rep., Defense Technical Information Center OAI-PMH Repository [http://stinet.dtic.mil/oai/oai] (United States).

[59] Landrum, D. B., Cerny, J., Warden, L., and Meyer, A., 2007. "Affordable flight simulation in an educational environment". In American Helicopter Society 63rd Annual Forum, Vol. 3, pp. 2241–2251.

[60] Garcia, R., and Valavanis, K., 2008. The usl autonomous helicopter testbed, June 23-24.

[61] Vidolov, B., Miras, J. D., and Bonnet, S., 2006. "A two-rule-based fuzzy logic controller for contrarotating coaxial rotors uav". In IEEE Conference on Fuzzy Systems, pp. 1563–1569.

[62] Hart, J., Francis, G., and Kauffman, L., 1994. "Visualizing quaternion rotation". *ACM Transactions on Graphics,* **13**(3), pp. 256–276.

[63] Theodore, C., Shelden, S., Rowley, D., Dai, W., McLain, T., and Takahashi, M., 2005. Full mission simulation of a rotorcraft unmanned aerial vehicle for landing in a non-cooperative environment, June 1-3.

[64] Stoor, B. J., Pruett, S. H., Duquette, M. M., Subr, R. C., and MtCastle, T., 2006. Urban simulation environment, August 21-24.

[65] Ricard, G. L., and Parrish, R. V., 1984. "Pilot differences and motion cuing effects on simulated helicopter hover". *Human Factors,* **26**(3), pp. 249–256.

[66] Parrish, R. V., Houck, J. A., and Jr., D. J. M., 1977. "Empiracle comparison of a fixed-base and a moving-base simulation of a helicopter engaged in visually conducted slalom runs". *NASA Technical Report,* **D-8424**, pp. 1–34.

[67] Zacharias, G. L., and Young, L. R., 1981. "Influence of combined visual and vestibular cues on human perception and control of horizontal rotation". *Experimental Brain Research,* **41**, pp. 159–171.

[68] Nahon, M. A., and Reid, L. D., 1990. "Simulator motion-drive algorithms: A designer's perspective". *Journal of Guidance, Control, and Dynamics,* **13**, pp. 356–362.

[69] Jang, J. S., and Liccardo, D., 2006. "Automation of small uavs using a low cost mems sensor and embedded computing platform". In 25th Digital Avionics Systems Conference, pp. 1–9.

[70] Webb, T. P., Prazenica, R. J., Kurdila, A. J., and Lind, R., 2007. "Vision-based state estimation for autonomous micro air vehicles". *Journal of Guidance, Control, and Dynamics,* **30**(3), pp. 816–826.

[71] Prazenica, R. J., Watkins, A. S., Kurdila, A. J., Ke, Q. F., and Kandae, T., 2005. "Vision-based kalman filtering for aircraft state estimation and structure from motion". In AIAA Guidance, Navigation, and Control Conference, Vol. v 3, American Institute of Aeronautics and Astronautics Inc., Reston, VA 20191, United States, pp. 1748–1760.

[72] Shi, J., and Tomasi, C., 1994. "Good features to track". In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, IEEE Comput. Soc. Press, pp. 593–600.

[73] Bouguet, J. Y., 2002. Pyramidal implementation of the lucas kanade feature tracker: Description of the algorithm. Tech. rep., Intel Corporation Microprocessor Research Labs.

[74] Watkins, A. S., Kehoe, J. J., and Lind, R., 2006. "Slam for flight through urban environments using dimensionality reduction". In AIAA Guidance, Navigation, and Control Conference, Vol. v 8, American Institute of Aeronautics and Astronautics Inc., Reston, VA 20191, United States, pp. 5018–5029.

[75] Hart, S. G., 2006. Nasa-task load index (nasa-tlx); 20 years later. Tech. rep., NASA-Ames Research Center.

[76] Milton, J., Small, S., and Solodkin, A., 2004. "On the road to automatic: Dynamic aspects in the development of expertise". *Clinical Neurophysiology,* **21**, pp. 134–143.

[77] Kelly, A., and Garavan, H., 2005. "Human functional neuroimaging of brain changes associated with practice". *Cerebral Cortex,* **15**, pp. 1089–1102.

[78] Poldrack, R., 2000. "Imaging brain plasticity: conceptual and methodological issues - a theoretical review". *Neuroimage,* **12**, pp. 1–13.

[79] Izzetoglu, M., Bunce, K., Ayaz, H., Devaraj, A., and Onaral, B., 2005. "Functional near-infrared neuroimaging". *IEEE Trans Neural Syst Rehabil Eng,* **12**(2), pp. 153–159.

[80] Oldfield, R. C., 1971. "The assessment and analysis of handedness: the edinburgh inventory". *Neuropsychologia,* **9**(1), pp. 97–113. Oldfield, R C England Neuropsychologia Neuropsychologia. 1971 Mar;9(1):97-113.

[81] Izzetoglu, K., Bunce, S., Onaral, B., Pourrezaei, K., and Chance, B., 2004. "Functional optical brain imaging using near-infrared during cognitive tasks". *International Journal of Human-Comptuer Interaction,* **17**(2), pp. 211–227.

[82] Ayaz, H., Izzetoglu, M., Platek, S., Bunce, S., Izzetoglu, K., and Pourrezaei, K., 2006. "Registering fnir data to brain surface image using mri templates". In IEEE Eng Med Biol Soc, Vol. 1, pp. 2671–2674.

[83] Ayaz, H., Shewokis, P., Bunce, S., Schultheis, M., and Onaral, B., 2009. "Assessment of cognitive neural correlates for a functional near infrared-based brain computer interface system". *Foundations of Augmented Cognition. Neuroergonomics and Operational Neuroscience*, pp. 699–708.

[84] Izzetoglu, M., Bunce, S., Izzetoglu, K., Onaral, B., and Pourrezaei, K., 2007. "Functional brain imaging using near infrared technology for cognitive activity assessment". *IEEE Engineering in Medicine and Biology Magazine, Special issue on on the Role of Optical Imaging in Augmented Cognition,* **26**(4), pp. 38–46.

[85] Reed, L., 1977. Visual-proprioceptive cue conflicts in the control of remotely piloted vehicles. Tech. Rep. AFHRL-TR-77-57, Brooks Airforce Base, Air Force Human Resources Laboratory.

[86] Flenniken, W., Wall, J., and Bevly, D., 2005. "Characterization of various imu error sources and the effect on navigation performance". In Proceedings of the 2005 ION GNSS.

[87] Diggelen, F. v., 2007. "Gnss accuracy lies, damn lies, and statistics". *GPS World*, January, pp. 26–32.

[88] Biezad, D. J., 1999. *Integrated Navigation and Guidance Systems.* AIAA Education Series. American Institute of Aeronautics and Astronautics, Reston, Virginia.

[89] Yun, B., Peng, K., and Chen, B., 2007. Enhancement of gps signals for automatic control of a uav helicopter system, May 30 - June 1.

[90] Grewal, M., Weill, L., and Andrews, A., 207. *Global Positioning Systems, Intertial Navigation, and Integration*. John Wiley And Sons, Inc, Hoboken, New Jersey.

[91] Baker, W., and Clem., R., 1977. Terrain contour matching (tercom) premier. Tech. rep., Aeronautical Systems Division, Wright-Patterson.

[92] Kim, J., and Sukkarieh, S., 2004. Slam aided gps/ins navigation in gps denied and unknown environments.

Appendix A. Confidence Questionnaire

Appendix A

Confidence Questionnaire

Subject number_____

Date_____

Session number_____

How well do you think you performed at the tasks given to you today (circle one)?

1	2	3	4	5	6	7
very poorly		poorly	Ok	fairly well		extremely well

How confident were you doing the tasks (circle one)?

1	2	3	4	5	6	7
Not Confident		A little confident	average confidence	fairly confident		completely confident

For each task rate how difficult the task was from 1 (very difficult) to 5 (very easy) as well as how well you think you did in this mission from 1 (very poorly) to 5 (almost perfectly).

Task	Difficulty	Performance
Keeping the plane under control (e.g. holding a course, altitude, or speed)		
Maneuvering the plane accurately (turning, pitching, rolling)		
Keeping track of your speed		
Keeping track of where you are flying		
Keeping track of your next objective		
Deciding where to fly next		
Choosing the correct path to your objective		
Locating objectives visually		
Deciding when you are directly above a target		

For each task, do you feel your ability to do this task has improved? (Yes/No/It's Worse)

Please comment on your improvements.

Figure A.1: Confidence Questionnaire Page 1

Keeping the plane under control (e.g. holding a course, altitude, or speed) and reacting to crosswinds and updrafts or downdrafts

Maneuvering the plane accurately (turning, pitching, rolling)

Keeping track of your speed

Keeping track of where you are flying

Keeping track of your next objective

Deciding where to fly next

Choosing the correct path to your objective

Locating objectives visually

Deciding when you are directly above a target

Figure A.2: Confidence Questionnaire Page 2

Appendix B. NASA Task Load Index

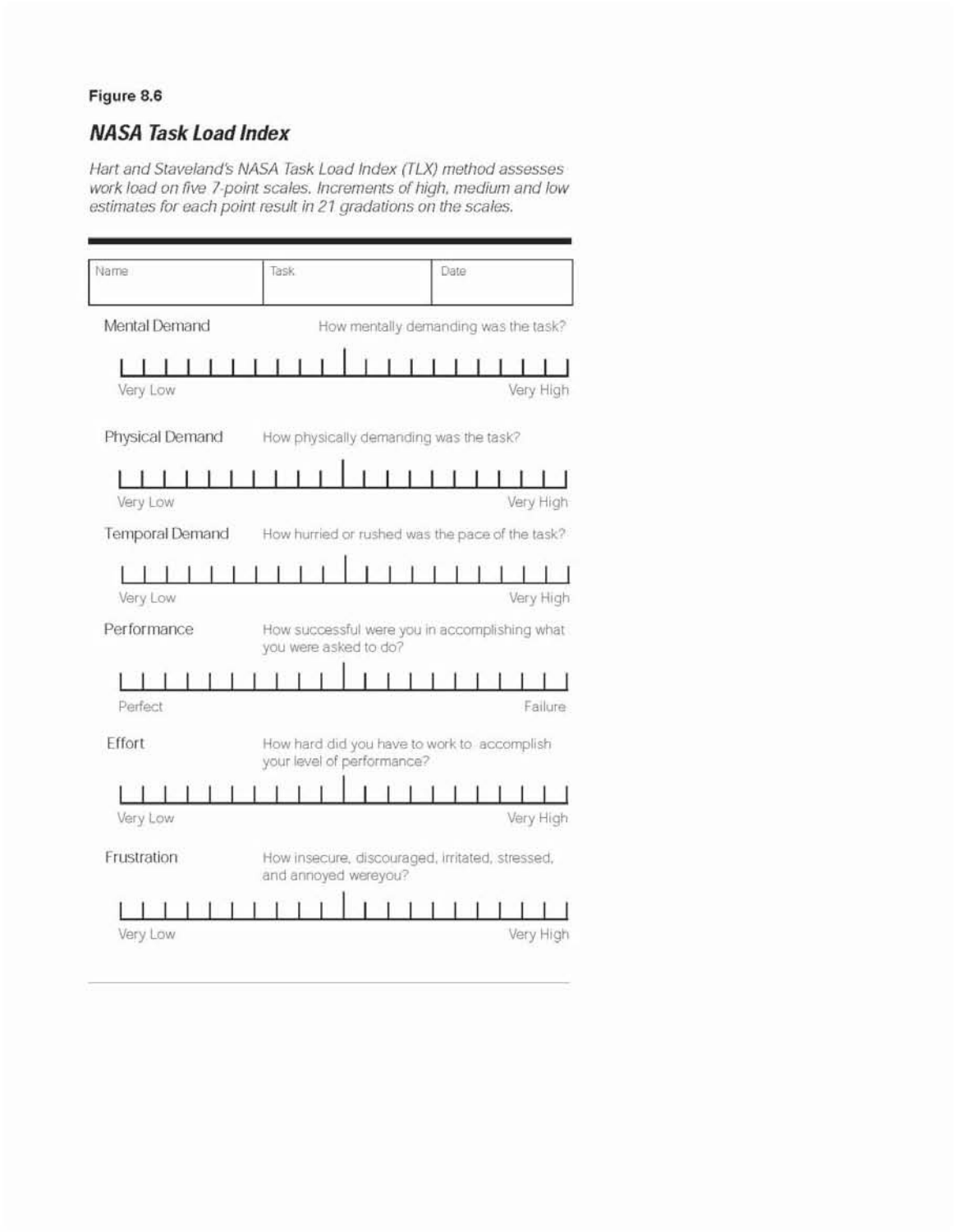

Figure 8.6

NASA Task Load Index

Hart and Staveland's NASA Task Load Index (TLX) method assesses work load on five 7-point scales. Increments of high, medium and low estimates for each point result in 21 gradations on the scales.

Name	Task	Date

Mental Demand — How mentally demanding was the task?
Very Low — Very High

Physical Demand — How physically demanding was the task?
Very Low — Very High

Temporal Demand — How hurried or rushed was the pace of the task?
Very Low — Very High

Performance — How successful were you in accomplishing what you were asked to do?
Perfect — Failure

Effort — How hard did you have to work to accomplish your level of performance?
Very Low — Very High

Frustration — How insecure, discouraged, irritated, stressed, and annoyed wereyou?
Very Low — Very High

Figure B.1: NASA Task Load Index

Appendix C. Background Questionnaire

Appendix C

UAV OPERATOR STUDY BACKGROUND QUESTIONNAIRE

Date_____ Participant #_____

Age____ Sex M F

Do you wear corrective lenses (circle one)? NO GLASSES CONTACTS

Highest education level attained:
_____ High School
_____ Some college, but no degree
_____ Two-year degree
_____ Four-year degree or higher

If you are in the military:
Grade: _____
Specialty (if applicable) _____
Time in Present Job _____yrs _____mos
Total Active Federal Military Duty _____yrs _____mos
Time in Career Field _____yrs _____mos

Pilot Experience
What military pilot training have you had? (please check all that apply)
_____ None
_____ Glider
_____ Ground school
_____ T-3
_____ T-37
_____ T-38
_____ T-1
_____ Initial qualification in an operational aircraft
_____ Mission qualification in an operational aircraft
_____ Other (please specify _____)

What private pilot training have you had? (please check all that apply)
_____ None
_____ Enrolled in ground school now
_____ Ground school for private pilot's certificate
_____ Private pilot's certificate
_____ Instrument rating
_____ Commercial pilot certificate
_____ Airline transport pilot certificate
_____ Other (please specify _____)

Figure C.1: Background Questionnaire Page 1

Miscellaneous Interests

On average, during the last six months, how many hours per week did you typically spend playing flight simulation video games? _____ (enter 0 if you did not use flight simulation games)

Please check the amount of lifetime experience you have playing flight simulation games.

_____ None
_____ 1-10 hours
_____ 11-50 hours
_____ 50-200 hours
_____ Over 200 hours

On average, during the last six months, how many hours per week did you typically spend playing 3-D action, sports, or driving video games (such as Halo, Half-Life, Assassin's Creed, Warcraft, Madden NFL, Gran Turismo — not flight simulation games)?

_____ (enter 0 if you did not play 3-D action, sports, or driving games)

Figure C.2: Background Questionnaire Page 2

Vita

Education

Ph.D. Mechanical Engineering,2010, Drexel University, Philadelphia, PA

M.S. Mechanical Engineering,2006, Drexel University, Philadelphia, PA

B.S. Mechanical Engineering,2003, Drexel University, Philadelphia, PA

Research Experience

Graduate Research Assistant, DASL, Drexel University, 2006-2010

Graduate Research Assistant, PRISM Lab, Drexel University, 2003-2006

Teaching Experience

Instructor/T.A., MEM Dept., Drexel University, 2003-2007

Honors and Awards

Gates Millennium Scholar 2000 - 2007

Department of Education GAANN Fellow 2006, 2007

Deans Fellow, Drexel University 2003-2007

Honorable Mention, NSF Graduate Research Fellowship 2004

Drexel University Athletic Scholarship, Golf, 1998-2002

Selective List of Publications

1. James Hing, Keith Sevcik, Paul Oh, "Improving Unmanned Aerial Vehicle Pilot Training and Operation for Flying in Cluttered Environments", IEEE Internation Conference on Intelligent Robots and Systems, St. Louis, MO, October 11-15.

2. James Hing, Paul Oh, "Development of an Unmanned Aerial Vehicle Piloting System with Integrated Motion Cueing for Training and Pilot Evaluation", Journal of Intelligent and Robotic Systems, Vol 54, pp.3-19 March 2009.

3. James Hing, Ari D. Brooks, and Jaydev P. Desai, A Biplanar Fluoroscopic Approach for the Measurement, Modeling, and Simulation of Needle and Soft tissue Interaction, Medical Image Analysis, Vol 11, Issue 1, pp. 62-78 Feb 2007.