

Urban Air Mobility System Testbed using CAVE Virtual Reality Environment

Panadda Marayong¹, Praveen Shankar¹, Jessica Wei²,
Hanson Nguyen², and Thomas Z. Strybel³

¹Department of Mechanical and Aerospace Engineering

²Department of Computer Engineering and Computer Science

³Department of Psychology

California State University, Long Beach
1250 Bellflower Blvd., Long Beach, CA 90840

Vernol Battiste

San Jose State University Research Foundation

NASA Ames Research Center

Moffett Field, CA 94035

Abstract—Urban Air Mobility (UAM) refers to a system of air passenger and small cargo transportation within an urban area. The UAM framework also includes other urban Unmanned Aerial Systems (UAS) services that will be supported by a mix of onboard, ground, piloted, and autonomous operations. Over the past few years UAM research has gained wide interest from companies and federal agencies as an on-demand innovative transportation option that can help reduce traffic congestion and pollution as well as increase mobility in metropolitan areas. The concepts of UAM/UAS operation in the National Airspace System (NAS) remains an active area of research to ensure safe and efficient operations. With new developments in smart vehicle design and infrastructure for air traffic management, there is a need for methods to integrate and test various components of the UAM framework. In this work, we report on the development of a virtual reality (VR) testbed using the Cave Automatic Virtual Environment (CAVE) technology for human-automation teaming and airspace operation research of UAM. Using a four-wall projection system with motion capture, the CAVE provides an immersive virtual environment with real-time full body tracking capability. We created a virtual environment consisting of San Francisco city and a vertical take-off-and-landing passenger aircraft that can fly between a downtown location and the San Francisco International Airport. The aircraft can be operated autonomously or manually by a single pilot who maneuvers the aircraft using a flight control joystick. The interior of the aircraft includes a virtual cockpit display with vehicle heading, location, and speed information. The system can record simulation events and flight data for post-processing. The system parameters are customizable for different flight scenarios; hence, the CAVE VR testbed provides a flexible method for development and evaluation of UAM framework.

TABLE OF CONTENTS

1. INTRODUCTION	1
2. SYSTEM OVERVIEW	2
3. VIRTUAL UAM WORKSPACE SIMULATION	3
4. UAM USE CASE	4
5. CONCLUSIONS	5
ACKNOWLEDGEMENTS	5
REFERENCES	6

1. INTRODUCTION

In recent years, advancements in unmanned aircraft technology have drawn interest and investment from industry and government agencies on smart unmanned aerial vehicle (UAV) designs for passenger and cargo transportation in a metropolitan area. Based on the announcement of the NASA Strategic Implementation Plan in 2017, the Urban Air Mobility (UAM) concept emerged as a major area of research under NASA's Aeronautics Research Mission Directorate (ARM D). UAM refers to a system for air passenger and cargo transportation within an urban area, inclusive of small package delivery and other urban Unmanned Aerial Systems (UAS) services that support a mix of onboard- or ground-piloted and autonomous operations [1-2]. The UAM concept is illustrated in Figure 1. Market studies funded by ARM D identified use cases of UAM for passenger and cargo transport that are viable and potentially profitable. Additionally, the UAM operation framework, with small electric-powered vertical-takeoff-and-landing (VTOL) vehicles, offers an innovative environmentally friendly transportation option for metropolitan areas.

NASA, in partnership with the UAM community, established the vision and framework for UAM that address several key



Figure 1. Illustration of UAM concepts (credit NASA)

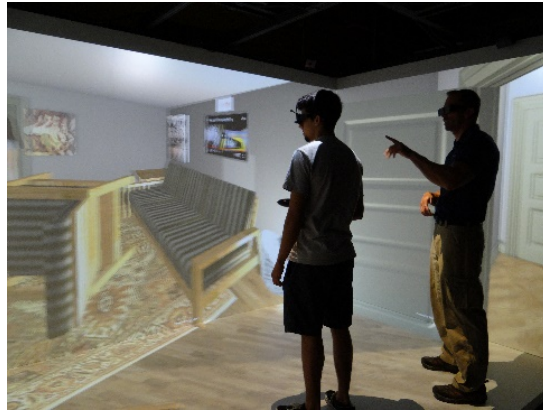


Figure 2. (Left) VisCube™ M4 CAVE virtual reality system (Right) Multiple users interaction inside the CAVE.

developments and barriers, including vehicle development and production, airspace system design and implementation, community integration, air traffic and fleet operations and management, and individual vehicle management and operations. UAM will require increasing autonomous systems, which is known to improve the capacity, safety and efficiency of complex systems. While many industry partners, including Uber and Boeing [3-4], have advanced the design of smart air vehicles, the concept of airspace operations and human-automation teaming in UAM framework remains an important area of research. For example, concepts developed for UAM airspace integration must take into account the wide range of performance that can be achieved by the diverse range of UAM aircraft types. In addition, performance variables, such as hover capabilities and wake around similar or dissimilar vehicle types, transition times between vertical and horizontal flights, will have significant implications on airspace design and operating procedures for airspace integration. Given these differences, traditional air traffic management (ATM) strategies may not support efficient UAM operations in urban environments. Both NASA and the FAA expect near-term operations to draw heavily on current airspace management technologies and procedures, as well as recent procedures developed for UAS traffic management (UTM). Hence, there is a need for an integrative approach to test and evaluate various aspects of UAM operation.

Virtual Reality (VR) provides a customizable and cost-effective platform to study human interaction with automation in complex environments. VR simulations have well-established use in aeronautic system analysis and training, such as flight simulators, aircraft design, and human-in-the-loop air traffic management [5-8]. In this paper, we describe our work that utilizes a Cave Automatic Virtual Environment (CAVE) system to create a VR testbed for human-automation teaming and airspace operations research for UAM. The CAVE environment consists of a four-wall projection system, a motion capture system, and control software for generation of immersive virtual environments with real-time full body tracking capability. In this initial phase of development, we created a virtual simulation of the San Francisco city with a passenger UAM

aircraft modelled after an enlarged quadcopter drone. The system components, the developed VR simulation, and research plans are described in the following sections.

2. SYSTEM OVERVIEW

CAVE System Hardware

A CAVE offers advantages over VR produced with head-mounted displays in that it provides better presence through a greater field of view, can be shared with multiple users, and allows an operator to interact immersively in a realistic volume. Our system is the VisCube™ M4 CAVE Immersive 3D Display (Visbox, Inc., IL), which offers an approximate 3D volume of 12.5'W x 8' D x 9' H. Figure 2 shows our CAVE system and users operating in a virtual room inside its simulation volume. The system consists of front-projected four panel displays (3 walls and a floor) fully integrated with an eight-camera ARTTrack5 motion capture system, a surround sound system, and a graphics workstation. The center and floor projectors have 1920 x 1200 pixel resolution while the side projectors have a 1316 x 1200 pixel resolution. The graphics workstation uses 12-core Intel Xeon E5-2650 v4 processor with Nvidia Quadro P5000 graphics card. The ART motion capture system utilizes the set of cameras to track targets with active or passive markers worn by the user to determine his/her motion in space, see Figure 3L. The ARTTrack5 optical tracking system provides a sensor resolution of 1.3 megapixels at full frame mode with a frame rate up to 300 Hz capable of full-body tracking for the CAVE workspace volume and the resolution needed for dynamic tracking (Advanced Realtime Tracking (ART) GmbH, Germany).

User Interfaces and Virtual Environment

To create immersive 3D displays, the user wears active stereo glasses with reflective targets (Figure 3R) to track his/her head movement through the ART motion capture system. User's head movement is used to update the view generated in VR according to his/her movement. The user can interact with virtual objects through the ART's Flystick2, which is a

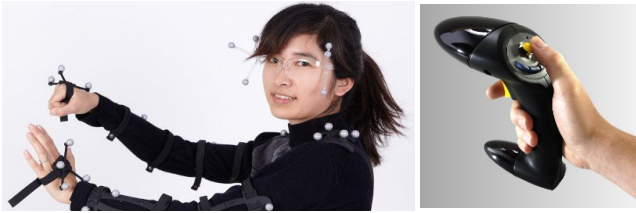


Figure 3. (Left) user with marker targets (Right) Flystick2 user input device (credit ART GmbH).

hand-held wireless interaction device with a protected passive target for real-time tracking of its position and orientation (Figure 3R). The Flystick2 consists of six buttons and an analog 2D joystick that can be programmed for different actions. Additional users can enter the CAVE space and wear shutter glasses to observe the view of the virtual environment as experienced by the main user with the active stereo glasses.

The VR simulation is developed on the Unity3D real-time game engine (Unity Technologies, San Francisco, CA) [9]. The CAVE system uses MiddleVR plugin (MiddleVR, Paris, France) to allow 3D contents generated in Unity to display appropriately across multiple screens [10]. The motion tracking system, including the cameras, interaction devices, and targets, are controlled through the ART DTrack2 software. ART-Human software is also available for full-body motion capture of a human subject using a specially-design set of 17 individual 6-DOF targets that are attached to various parts of the body, and the three-finger Fingertracking2 system with vibrotactile feedback. The MiddleVR plugin supports multiple 3D interaction devices across different platforms for navigation and manipulation of the virtual world, including the ART DTrack2 and the Flystick2. However, the ART-Human and the Fingertracking software are not supported currently by MiddleVR. We are currently working on the integration of these additional targets with our system.

3. VIRTUAL UAM WORKSPACE SIMULATION

The majority of UAM operations will be within urban environments initially using visual flight rules (VFR) right-of-way rules, and/or segregation of specific vertiport access by vehicle classes with compatible operating envelopes [11]. Additionally, since UAM is planned for urban metropolitan areas that are located proximal to major airports, most UAM operations will require coordination with air traffic control (ATC) for some portions of their mission. VR environment provides flexibility to add and customize simulation parameters, such as other aircrafts flying in the scene and traffic modeling, for real-world scenarios. However, in this initial phase of development, we start with the city of San Francisco and the San Francisco airport as the virtual workspace for testing concepts of UAM operations. The simulation includes a single aircraft that is maneuvered by a single operator using a flight control joystick. The sections

below provide a description of the simulation and the user interface.



Figure 4. User inside the virtual aircraft overlooking San Francisco city.

Simulation of San Francisco City

The city of San Francisco displayed within the 3D environment is created with an open-source WrlD3D API (WRLD, Los Angeles, CA) [12], which allows inputs of pre-made real-world locations into the Unity3D environment. WrlD3D API is implemented in our environment for which the entire San Francisco city to SFO Airport is loaded through a large preset boundary. Buildings and streets were proportionally scaled to the real-world environment. The city scene generated in WrlD3D are mapped into the Unity environment based on specified longitude and latitude of a selected origin in the SF city. The boundary of the city that is viewable can be changed in the simulation setting. One of the limitations of WrlD3D is that the graphics are generated at a lower resolution using low quality polygons and gray-scaled graphics to accommodate the rendering in real time. Another limitation is that the scenes are created and deleted based on the current location and orientation of the user. This process relies heavily on the download and upload speed from both the CAVE graphics system and the WrlD3D server. However, the scenes can be rendered without a significant lag in our simulation.

Aerial Vehicle Model

We modelled the virtual aircraft as an enlarged quadcopter that can take-off and land vertically. The vehicle was modified from a hexacopter 3D drone model available through the Professional Drone Pack (Professional Assets, Unity Asset Store [13]). The interior and the exterior of the original drone model was modified using open-source 3D computer graphic software, Blender, to create a quadcopter that is scaled to fit 2-4 passengers with cockpit displays for the pilot. It is important to note that the dynamics of the vehicle are based on a simple rigid body model. Several modifications were made to the exterior of the vehicle including moving the propellers above the pilot's line of sight

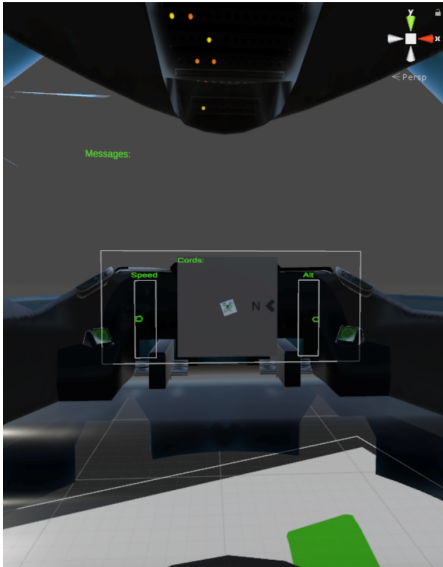


Figure 5. Cockpit display of the virtual aircraft.

and using a darker color to increase its visibility when flying around and over buildings. While the interior space of the original drone was empty, we added the cockpit display and the cabin space using cockpit components available on aircraft models in Silantro Flight Simulator Toolkit (Oyedoyin, Unity Asset Store [13]). The dimension of the cockpit display relative to the overall vehicle dimension was kept to accommodate a seated pilot. The aircraft can be programmed to fly autonomously or manually by a single pilot.

Cockpit Display and User Input Interface

During this initial implementation, the virtual cockpit display consists of a mini-map with a bird's eye view of the aircraft, a speed display, an altitude display, and a message board for incoming messages from an external source to simulate communications with an Air Traffic Controllers and Dispatchers (Figure 5). Additional displays can be added or customized for future testing. The pilot can maneuver the virtual aircraft in real time using a Logitech Attack3 USB joystick. The joystick has 11 programmable buttons and force feedback functionality. The operation of the joystick is configured to mimic the functionality of a flight yoke system. For example, the pilot can move the aircraft forward by pushing the joystick forward to change its forward speed relative to how far the joystick is pushed. The aircraft will also rotate and bank through the tilt of the joystick. As depicted in Appendix A, the aircraft is also able to climb, alter its speed, and turn. It is to note that the joystick and the user's hand position are currently not tracked by the system.

4. UAM USE CASE

To develop an infrastructure for testing of operational concepts and human-automation teaming for UAM, we created a use case of a flight from a downtown San Francisco location to the SFO airport international terminal. The route

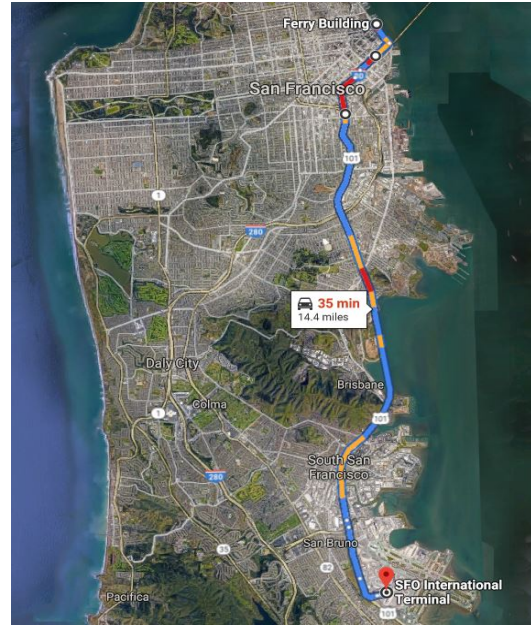


Figure 6. Flight route along highway 80 and 101 from downtown to SFO airport. (credit: Google).

is initiated from the Port of San Francisco, Ferry Building Clock Tower and end at the international terminal at SFO airport, as shown in Figure 6. The aircraft heads southeast following the Embarcadero Road toward Highway 80 to merge on the Highway 101 southward to the SFO airport. Upon reaching the exit on Hwy 101 for SFO airport, the aircraft turns eastward to land on a marked vertipad on top of the SFO International terminal building (Figure 7). The highways can be seen through the aircraft windows and on a cockpit display (mini-map) with the zoomed out view of the area below the aircraft.

The system provides two modes of flight: manual mode and autonomous mode. In the manual mode, the pilot has full control over the speed, altitude and direction of the aircraft using the joystick. The pilot must rely on his/her vision to follow the appropriate flight path to the SFO airport – in aviation terms, pilotage. In the manual mode, the pilot has an option to operate in the “Toggle” mode, which allows the pilot to fly the aircraft at a pre-set speed without continuously pushing the joystick for a more comfortable flight experience. In the autonomous mode, the aircraft is programmed to cruise at a constant speed along twenty waypoints that are generated along the described route between downtown to the SFO International terminal building. To follow the flight path, the aircraft move between consecutive waypoints using a linear interpolation function available in Unity. In the autonomous mode, the aircraft climbs to the altitude of 600ft during flight. The system can be configured to record multiple simulation variables for post processing and analysis, such as aircraft position, speed, and events. In the next step, we will perform validation test of our Phase-I VR system. The validation tests will consist of scenarios that involve take offs from downtown locations in San Francisco, flights to San Francisco Airport (SFO) and landing in designated airport areas. These will be flown by certified private pilots who will

provide feedback on the simplicity of the controls, and realism of the external (out-the-window) and internal (cockpit) environments.

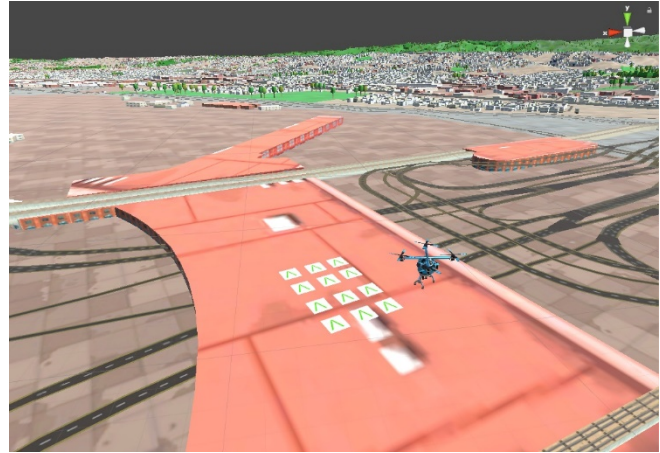
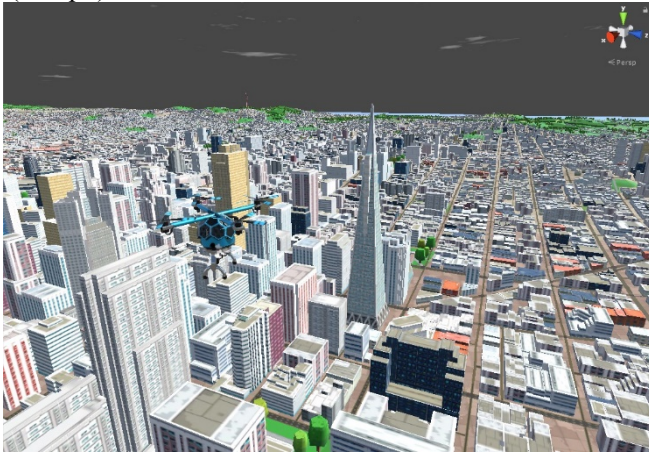


Figure 7. (Left) Aircraft flying over the city of San Francisco (Right) Landing at a vertipad at SFO airport.

5. CONCLUSIONS

In this work, we present a virtual environment for UAM concepts for the city of San Francisco using a CAVE VR system. In the current system, the vehicle is modelled as an enlarged quadcopter for 2-4 passengers. The use case scenario is taking off and landing between a downtown location and a SFO airport terminal. The aircraft can be operated in real-time manually by a single pilot using a flight control joystick or flown autonomously. Several simulation parameters are configurable to create different experimental scenarios. The VR simulation provides a flexible testbed to conduct research on various aspects of UAM including concepts of operation, human-automation teaming, vehicle and cockpit design, and passenger experience.

We will continue to enhance the functionalities of the VR environment including an integration of Air Traffic Management operations, inclusion of the finger-tracker for manipulation of virtual control objects, enhanced vehicle flight dynamics reflecting real-world scenarios, and other traffic in the SFO area. It is our near-term goal to use the system to conduct research on air traffic management operations in and around class B airspace to develop procedures and guidelines for efficient UAM transitions and operations in the airspace. For additional information on the CAVE system including videos of the UAM simulation, please visit the CSULB CAVE website: www.beachcave.org

ACKNOWLEDGEMENTS

The acquisition of the CAVE system was funded by the NSF Major Research Instrumentation award (Award #1626655). Students received funding support from San Jose State Research Foundation Grant #2116145736 (PI: Sean Laraway). The authors would like to also acknowledge Chris Senner for his work on the initial development of the VR simulation.

REFERENCES

- [1] <https://www.nasa.gov/aero/nasa-embraces-urban-air-mobility> (Nov 7, 2017)
- [2] <https://www.nasa.gov/uam>
- [3] <https://www.uber.com/us/en/elevate/uberair/>
- [4] www.boeing.com/NeXt
- [5] <https://www.x-plane.com/>
- [6] N. Dorighi, and B. Sullivan, "FutureFlight Central: A Revolutionary Air Traffic Control Tower Simulation Facility," AIAA Modelling and Simulation Technologies Conference and Exhibit, August 11-14, Austin, TX, 2003.
- [7] D. Giovannetti, "NASA Ames Research Center Simulation Laboratories," Journal of Modeling and Simulation Information Analysis Center (MSAIC), 2003. Retrieved from: <https://www.aviationsystemsdivision.arc.nasa.gov/publications/category/simulation.shtml>
- [8] F. Zuniga, S. Cliff, D. Kinney, S. Smith, V. Hawke, and C. Tang, "Vehicle Design of a Sharp CTV Concept Using a Virtual Flight Rapid Integration Test Environment," AIAA Atmospheric Flight Mechanics Conference and Exhibit, August 5-8, Monterey, CA, 2002.
- [9] <https://unity.com/>
- [10] <https://www.middlevr.com/>
- [11] B. Lascara, A. Lacher, M. DeGarmo, D. Maroney, R. Niles, and L. Vempati, "Urban Air Mobility Airspace Integration Concepts." The Mitre Corporation, 2019. Retrieved from; <https://www.mitre.org/publications/technical-papers/urban-air-mobility-airspace-integration-concepts>
- [12] <https://www.wrld3d.com/>
- [13] <https://assetstore.unity.com/>

BIOGRAPHY



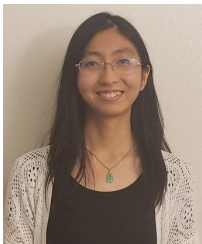
Dr. Panadda (Nim) Marayong is a Professor and the director of the Robotics and Interactive Systems Engineering (RISE) Laboratory in the Department of Mechanical and Aerospace Engineering at California State University, Long Beach (CSULB).

She received a B.S. in Mechanical Engineering from Florida Institute of Technology and a M.S. and a Ph.D. in Mechanical Engineering from Johns Hopkins University before joining CSULB in 2007. Her research areas are in robotics, human-machine collaborative systems, and haptics.



Dr. Praveen Shankar is an Associate Professor in the Department of Mechanical and Aerospace Engineering at California State University, Long Beach. Dr. Shankar's research expertise is in the analysis and design of control systems for complex dynamic systems.

He serves as the director the Collaborative Autonomous Systems Laboratory at CSULB which focuses on the development and testing of advanced motion planning and control technologies for autonomous robotic systems.



Jessica Wei is a sophomore in the Computer Science Program at California State University, Long Beach. She has been working on the Urban Air Mobility research project for over a year. Her contributions include building a data collection system and developing drone control and

movement. She currently serves as the Vice President of the Video Game Development Association at California State University, Long Beach and has been a part of its board since 2018.



Hanson Nguyen is a sophomore in the Computer Science Program at California State University, Long Beach. He has been working on the Urban Air Mobility research project for over a year. His contribution involves scripting and system integration

between Unity and Wrl3D. He serves as a General Officer for the Video Game Development Association at California State University, Long Beach since 2018.



Dr. Thomas Strybel has over 30 years of experience in human factors that has been focused on assessing human performance in complex work environments, and in training and mentoring human factors students. He is Professor Emeritus of Psychology and Human Factors at California State

University Long Beach, and Director of the Center for






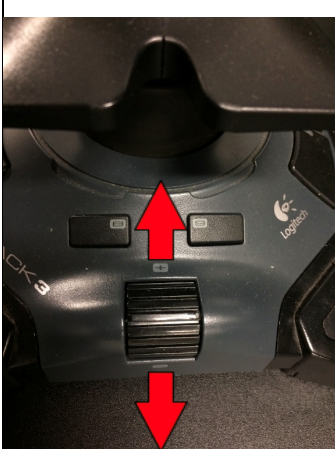
Human Factors in Advanced Aeronautics Technologies (CHAAT), A Group 5 NASA University Research Center. His research interests include human systems integration in cockpit and air-traffic-management systems, driver distraction, training methods, audio-visual displays, human factors and perception. He is an Associate Editor of the journal *Auditory Perception and Cognition*.



Vernol Battiste is one of the founding members of the NASA-TLX development team and a former Co-PI for the FDDRL at NASA-ARC. He is currently a Senior Research Associate for

the San Jose State University Research Foundation at NASA Ames in the Human Autonomy Teaming (HAT) laboratory. He was designated as one of 50 Most Important Blacks in Research Science for 2004, and received the Black Engineer of the Year Award in 2001. He has extensive knowledge of human factors issues relating to the operation of manned and unmanned systems. He received his BA in Experimental Psychology from California State University, Hayward, and his MA from San Jose State University. He is also a graduate of the FAA's Air Traffic Control Academy, and was fully Certified as an ATC Specialist at the Oakland Air Route Traffic Control Center. He completed the USAF Air Traffic Control Academy and the Jet Engine training program and spent 4 years as an Air Force Jet Engine Mechanic, and 5 years as an Air Force Air Traffic Controller.

Table 1. Mapping of Logitech Attack3 joystick to the motion control of the virtual aircraft.

Joystick Function	Control Function	Joystick Function	Control Function
	<p>Push the joystick forwards (90° max) to fly forwards and turn as necessary.</p> <p>Pull the joystick backwards to slow the speed. If the vehicle's speed is less than 0.5 unity length unit/s, the vehicle will slowly move backwards.</p>		<p>Press buttons 2 (top) and 3 (bottom) to climb or descend the vehicle, respectively.</p> <p>Press buttons 4 (left) and 5 (right) to rotate the vehicle when travelling at < 0.5 unity length unit/s.</p>
	<p>Press button 7 to switch from Manual/Toggle to Autonomous mode.</p>		<p>Press button 11 to turn the vehicle on or off.</p>
	<p>Press the trigger to switch from manual speed control to set the current forward speed (Toggle mode).</p>		<p>The scroll wheel controls the maximum speed preset for the vehicle.</p>