

A Testbed for Exploring Human-Robot Interaction with Unmanned Aerial and Ground Vehicles

Jaime H. Flores¹, Glenn A. Martin¹, and Paula J. Durlach²

¹ University of Central Florida, Institute for Simulation and Training, 3100 Technology Parkway, Orlando, FL 32826

² U.S. Army Research Institute for the Behavioral and Social Sciences, 12350 Research Parkway, Orlando, FL 32826

{jflores, martin}@ist.ucf.edu, Paula.Durlach@us.army.mil

Abstract. Over the last twenty years, the emerging roles of unmanned aerial/ground vehicles in the U.S. military presented a number of different research opportunities in usability and training, ranging from robotic control interfaces to human-robot team collaboration. In this paper we present a testbed that we developed as a flexible software platform to explore a variety of training and coordination issues with UXVs for military application.

Keywords: Interface Usability, Unmanned Vehicles, Team Collaboration.

1 Introduction

Effective operation and control of UXVs (unmanned aerial/ground vehicles) is a major usability question, as the U.S. military is heavily reliant not only on technological superiority but also superior training in the use of that technology. [1]

In order to effectively study the impact of the UXV user interface on performance and collaboration of an operator with a commander, a testbed should provide interchangeable input mechanisms for controlling a vehicle, multiple types of sensor feedback regarding the vehicle state, and control over the presentation of vehicle data. A flexible feedback system should also be included to present experimenter feedback to the operator.

Beyond single-operator interface questions, team training and collaboration aspects are also investigated due to the ubiquity of squad-level missions in the military. This centers on the interaction of an operator-commander pair of participants, allowing the pilot to control one or more UXVs while a co-federate provides mission plans with detailed routes, areas of interest, and specific points over a geographic area. Communication capabilities between the participants should also be taken into consideration as part of a testbed, since the participants may need to work in a distributed setting.

As part of research opportunities with the Army Research Institute for Behavioral and Social Sciences, the testbed was developed for the purpose of studying the effectiveness of current and experimental robotic control interfaces and their use within joint human-robot teams. The rest of this paper presents an in-depth discussion

of our testbed for this research, with emphasis on the versatility in the design and future expansions on this work.

2 Platform

Our platform consists of three software applications that have been developed over the last six years. Each application is designed to provide a specific, yet flexible feature set geared toward research in UXV usability and training. SimOCU (Simulated Operator Control Unit) is the main simulation application presented as the operation control interface for one or more UXVs. This application includes interchangeable models for controlling the motion of the unmanned vehicle, as well as a camera model based on a pan/tilt/zoom metaphor. C2Node (Command and Control Node) provides an interface for a commander participant to work along with the UXV operator, providing mission planning and support both before and during a mission. AFS (Automated Feedback System) adds a fully configurable rule-based feedback generation system to the platform, using an Action-Response interface metaphor.

2.1 Simulated Operator Control Unit

SimOCU is the core application for simulation and testing of different user interface and control configurations for an unmanned vehicle. It is designed to either work in standalone or as part of a distributed simulation, with multiple control and motion models, and configurable frame-based layout to allow for wide range of control in experimentation.

Virtual Environment. The simulation space is a virtual environment created using the Virtual Environment Simulation Sandbox (VESS) [2] and supports up to three different UXVs per simulation. Each vehicle can be configured to simulate a variety of sensors, notifications, and behaviors, including number of cameras, out-of range warnings, flight status sensors, and transmission quality. Warnings and flight status can be configured to specific distances or commands relative to the UXV base station, and the transmission latency can be set to a specific delay to simulate real-world communication variables. The environment can further be configured for a variety of real world areas and scenarios through loaded graphical models, as well as control over time of day, fog/visibility, and some basic weather patterns.

The camera model is based on the pan/tilt/zoom metaphor, and can be configured to allow a range of movement (or disabled entirely) on each axis per UXV. Up to three cameras are supported per vehicle, and each can be positioned on the simulated model to present an accurate display of the orientation and view each camera is expected to show. The controls also allow snapshots to be taken from any camera and can be transmitted to (as well as receive from) the C2Node application.

SimOCU is also built on the GEMINI library [3], making this application both DIS and HLA compatible. As such, this application can be used as part of a larger distributed simulation with entities capable of being displayed in the SimOCU application.



Fig. 1. Available feedback frames for UXV operation in SimOCU

Multiple Control and Motion Models. SimOCU can support different physical interfaces for controlling the vehicles, as well as different motion models for each vehicle. In particular, a participant can control any UXV via a traditional computer keyboard and mouse, or via a USB game controller (similar to those included with video game consoles).

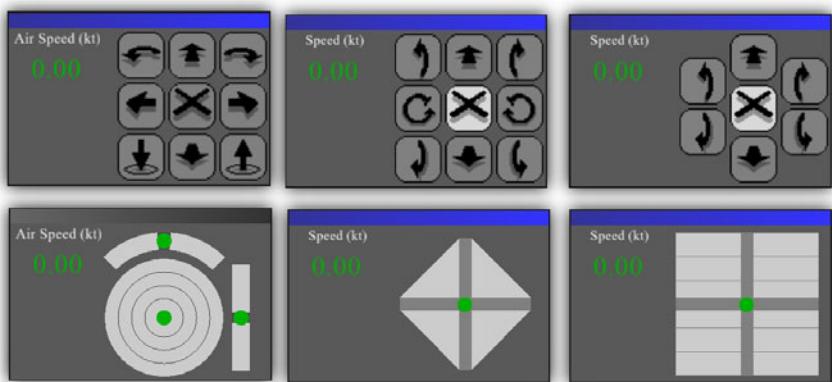


Fig. 2. Manual control interfaces in SimOCU

Motion models supported for UXV control include a continuous control scheme (where a user can control the exact speed of the vehicle on multiple axes) and a discrete control scheme (where a user can issue movement commands for a single axis). Figure 2 shows the manual control interfaces for these models currently in SimOCU for aerial, wheeled and tracked UXVs respectively, though the application is designed to easily expand support to other motion models (as well as physical control interfaces).

Beyond manual control, the simulation also supports automated navigation of the UXV through a waypoint-based spline path system. SimOCU includes an editor for creating these automated routes, as well as having assigning specific actions at each waypoint, such as, patrolling an area or taking a camera snapshot.

Configurable Frame Layout. The UXV interface is based on a number of distinct frames (shown in Fig. 1) displaying information received from the simulated vehicle. This model allows for the most flexibility in interface control as these frame elements can be moved, reconfigured, or hidden on an individual basis to allow for precise control of the size, position, and information shown in each one. The layout of these frames can be saved and reloaded on demand for any given simulation.

2.2 Command and Control Node

C2Node was designed as a complement to SimOCU by providing an interface for a commander to collaborate with the UXV operator. The application can display camera and position data for the unmanned vehicles and provides tools for creating mission plans that can be sent to UXV operator on demand, as well as text and voice communication.

Streaming UXV Data. C2Node (shown in Fig. 3) can receive UXV camera and position data transmitted through a network as it also supports DIS and HLA protocols through GEMINI [3]. It can handle data streams from any active UXV from a single SimOCU at a time. It can also send images to the operator as well as receive camera snapshots from any of the UXVs. A transmission delay can also be specified for receiving all data from the operator to simulate real world communication issues, as well as simulate loss of camera feed should the UXV move out of range from the base station, as specified in the SimOCU simulation.

Mission Planning. C2Node includes the capability to create, edit, and send a mission plan to an UXV operator. A mission plan consists of a number of markers that appear on the Map Frame to indicate a point/area of interest or to define a route the operator should take with the vehicle. This serves as a complete and effective way to communicate instructions and goals to the UXV operator beyond simple text or voice messages. Both text and voice channels however are supported for more traditional communication.

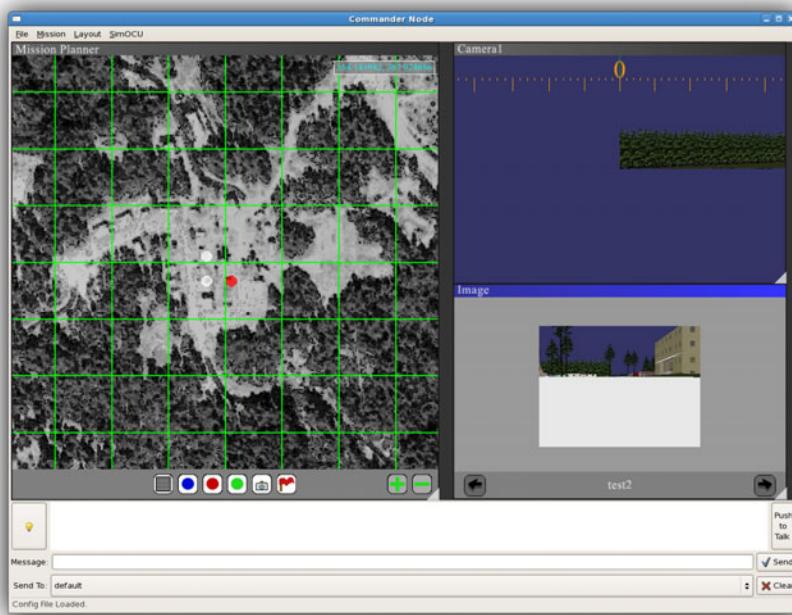


Fig. 3. The main interface in C2Node

2.3 Automated Feedback System

AFS is a rule-based feedback system that, once configured, can generate text feedback based on a user-specified set of questions used to generate raw scores. Feedback can then be transmitted to a C2Node or SimOCU user over a network.

Rule System. The generation of feedback in AFS is based on a graded Objective-Action-Response system. An Action is a collection of Responses uniquely tied to that action, and an Objective is a collection of Actions. Each level (Objective, Action, Response) can be graded separately to provide a raw score. From the various scores, Feedback is conditioned through user-specified Rules that define what kind of Feedback is generated. Objectives, Actions, Responses, Feedback, and Rules are entirely configurable, and can be saved and reloaded on demand into AFS (Fig. 4 shows an example).

Feedback Generation. Feedback is generated by the experimenter at the end of a mission and transmitted to a C2Node or SimOCU user. Along with the feedback, AFS also supports text communication to both SimOCU and C2Node. When feedback is generated, each set is normally generated completely independent of previous missions and scores. The Rule system, however, includes comparison options that allow current feedback scores to be compared to previous mission scores for more targeted and precise feedback.

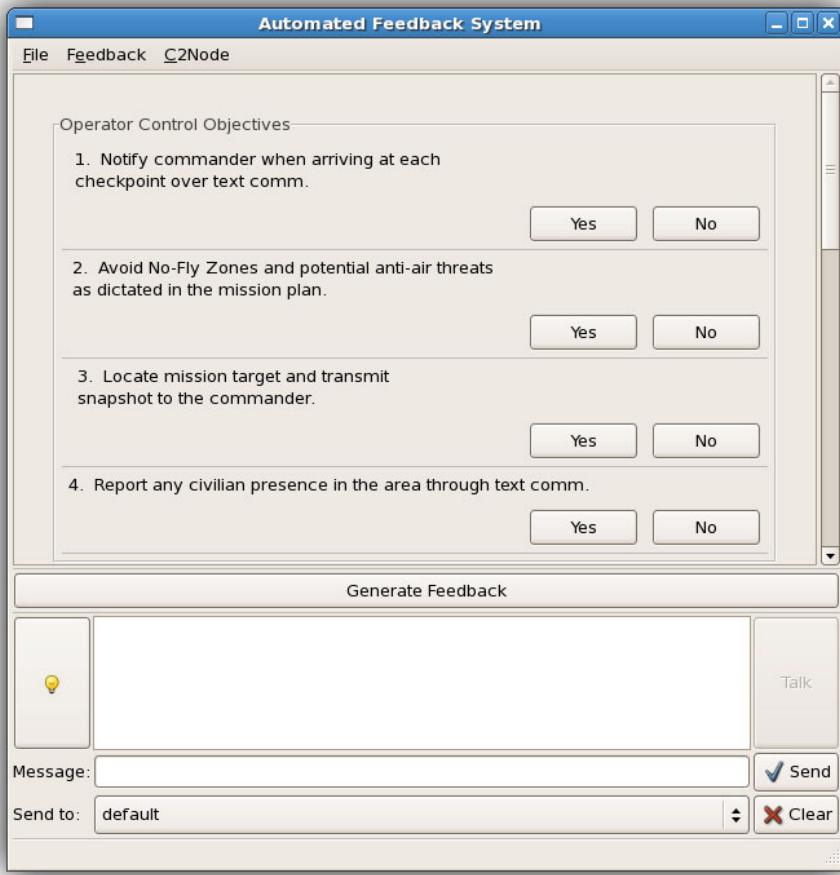


Fig. 4. The main interface in AFS

3 Research and Usage

Development of our experimental testbed has always been research-driven through work with the Army Research Institute for the Behavioral and Social Sciences (ARI). Completed studies employed the testbed to examine methods and metrics for operator training, and aspects of UXV interface usability [1] [5] [6].

Initial work focused almost entirely on interface questions regarding the intuitiveness of UXV operator control unit [1]. A number of different elements including physical control (mouse vs. game controller), motion model (discrete vs. continuous), and sensor arrangement (one combined camera display vs. two simultaneous camera displays) were compared to determine their affect of each on training and performance. The findings

suggested that given certain patterns that arose in the data, a heuristic could be used for designing an effective UXV control interface. Subsequent work established how different interface elements affected different aspects of performance during operator training [5] [6].

Future work will use the AFS system to examine the relative training effectiveness of providing feedback adapted to the trainee's skill level and team training issues.

In order to provide a setting to examine communication and coordination issues between UXV operator and commander, a separate virtual environment simulation for running dismounted infantry missions was added. This allowed for a collective exercise with a UXV and multiple friendly, hostile, and civilian entities, which provided the commander a mission context in which to employ the UXV. The initial findings support the potential for effective training using the UXV while on a mission [4], and after some further development on the platforms, would allow more complete exploration of team coordination issues and the training which would address them. Additional questions are now also being explored surrounding the factors involved in effective human-UXV teamwork.

4 Limitations

The testbed presented here was designed with some flexibility in mind for future expansion. SimOCU in particular was developed such that the additional UXV platforms, interface elements, control schemes or simulation elements can either be done by the researcher in the application configuration, or would only require a short development cycle.

Despite this intent, the applications still have some limitations. This has been more recently apparent in AFS. While generating feedback is automated from the marked responses, it cannot automatically detect actions in the simulation that might trigger a particular kind of feedback, which is why a researcher must currently enter all the information manually. This is due to lack of infrastructure and established protocol for detecting such specific actions as taking a snapshot of a particular target, or identifying a specific object. This limitation, however, is not insurmountable.

Limitations in C2Node and SimOCU have largely been related to integration with other simulation systems. Integrating Research Network Inc.'s GDIS2 military training simulation (described in [4]) presented a number of entity synchronization issues within each application. This was in large part due to the differences in the terrain models used by each application. GDIS2 in particular uses a proprietary data format for terrain models, which prevented the model's use with SimOCU and C2Node. This meant that a custom terrain model for SimOCU and C2Node was required. Differences in these models caused problems in matching camera views and entity positions from the UXV simulation with those views seen in GDIS2. Circumventing these issues for [4] required careful adjustments to the simulation and use of the terrain models to match all the necessary views and entities. Recent developments with VESS [2] should now allow the GDIS2 terrain models to be used with SimOCU and C2Node, thus eliminating most of these problems in future research.

References

1. Durlach, P.J., Neumann, J.L., Billings, D.R.: Training to Operate a Simulated Micro-Unmanned Aerial Vehicle with Continuous or Discrete Manual Control. Technical Report 1229. U.S. Army Research Institute for the Behavioral and Social Sciences (2008)
2. Daly, J., Kline, B., Martin, G.A.: VESS: A Testbed for Virtual Reality Research and Application Development. In: Proceedings, IEEE Virtual Reality 2002, pp. 289–291. IEEE Press, Los Alamitos (2002)
3. Martin, G.A., Daly, J., Thurston, C.: Interaction within Multimodal Environments in a Collaborative Setting. In: First International Conference on Virtual Reality (2005)
4. Durlach, P.J., Priest, H., Martin, G.A., Saffold, J.: Developing Collective Training for Small Unmanned Aerial Systems Employment. In: Proceedings, MODSIM World Conference, pp. 235–240 (2009)
5. Billings, D.R., Durlach, P.J.: How Input Device Characteristics Contribute to Performance During Training to Operate a Simulated Micro-Unmanned Aerial Vehicle. Technical Report 1273. U. S. Army Research Institute for the Behavioral and Social Sciences (2010)
6. Billings, D.R., Durlach, P.J.: Effects of Input Device and Latency on Performance While Training to Pilot a Simulated Micro-Unmanned Aerial Vehicle. Technical Report 1234. U. S. Army Research Institute for the Behavioral and Social Sciences (2008)