# Quadditch: An Augmented Reality, Multiplayer, Aerial Robotics Game for Outreach, Education, and Research


**Matthew Foran** Graduate Student, Virginia Tech, Bradley Department of Electrical and Computer Engineering, 24061, Blacksburg, VA, USA. mjforan@vt.edu

**Naveed Nekoo** Graduate Student, Virginia Tech, Crofton Department of Aerospace and Ocean Engineering, 24061, Blacksburg, VA, USA. nnaveed2@vt.edu

**Wade Foster** Graduate Student, Virginia Tech, Crofton Department of Aerospace and Ocean Engineering, 24061, Blacksburg, VA, USA. jwade109@vt.edu

**Jean-Michel Fahmi** Graduate Student, Virginia Tech, Crofton Department of Aerospace and Ocean Engineering, 24061, Blacksburg, VA, USA. fahmi@vt.edu

**Nazmus Sakib** Graduate Student, Virginia Tech, Crofton Department of Aerospace and Ocean Engineering, 24061, Blacksburg, VA, USA. nazmus.sakib@vt.edu

**Craig Woolsey** Professor, Virginia Tech, Crofton Department of Aerospace and Ocean Engineering, 24061, Blacksburg, VA, USA. cwoolsey@vt.edu



***ABSTRACT***

**Quadditch, or Quadrotor Quidditch, is a multi-player game in which human operators each control a drone using a gaming controller and an augmented reality, first-person-view (AR/FPV) interface. The game provides an engaging mechanism for education and outreach using aerial robotics. Quadditch is also a useful tool for human factors research. As a competitive game, it provides direct, objective measures for the effectiveness of technology features. Combined with subjective measures, such as those familiar to the flying qualities community, such data can be used to improve AR/FPV robotic system interfaces and the training required to use them effectively. The Quadditch hardware and software architecture is presented along with plans for continuing research.**



**Keywords:** Multirotor flight control, augmented reality, first person view, human factors, gaming


## 1 Motivation

As scaffolding for robotics outreach, education, and research, we have developed Quadditch – "quadrotor Quidditch" – an aerial robotic team competition reminiscent of the fictional sport invented by J. K. Rowling [1]. A key attribute of Quadditch is each player's use of first person view (FPV) video imagery augmented with virtual artifacts to enable game play and to improve player performance and overall system safety. The human competitive aspect of the game induces an operational pace that exposes challenges in the use of augmented reality FPV (AR/FPV) for the control of aerial robots. These challenges include low-level concerns such as stability augmentation to tune drone performance to an operator's skill level [2], accurate georectification of augmented reality artifacts within the FPV imagery [3], and delay compensation in the augmented video stream [4], as well as high-level concerns such as user workload, performance, and system trust [5]. Exposing and addressing these challenges will advance the utility of AR/FPV for civil and commercial aerial robotics applications like inspection [6, 7], delivery [8], construction [9], and disaster response [10]. Hence, Quadditch serves both as an engaging game for education and outreach and as a research test bed to advance the underlying technologies. Below, we provide a bit more context for the motivating challenges listed here, but this paper is primarily intended to provide an overview of the Quadditch concept, including the hardware and software architecture. To the authors' knowledge, Quadditch is a first-of-its-kind effort to implement real-time synchronization among multiple, flying drones in a multiplayer, AR/FPV game using the Unreal Engine environment.

Aircraft operator interfaces are rapidly evolving in the era of unmanned aircraft and it is important to develop effective tools for their design and evaluation. Aircraft design optimization and performance evaluation are well documented [11–13], but the operator is often considered only to a limited extent, e.g., through stability and flying qualities requirements obtained from human studies. Exceptions include a strand of research on aircraft stability and control with a pilot-in-the-loop, as in the work of McRuer [14] and others. UAVs are accessible and operable by a much larger population than are piloted aircraft, however, and the expectations for pilot behavior are very different. We assume that the typical Quadditch participant is not a trained UAV pilot and that the control system performance must be adapted to match the operator's proficiency. This tunable control system performance makes the game of Quadditch more accessible and reflects a need to adapt aerial robot interfaces for use by non-specialists in inspection, delivery, disaster response, and other tasks.

The implementation of Quadditch requires the accurate georectification of augmented reality (AR) artifacts, such as the goal posts, in a real-time visual image from a maneuvering UAV. Methods and comparisons for several georectification approaches for video and camera imagery are presented in [15], [16] and [17]. Padró *et al* [17] showed that georeferencing with ground control points (GCPs) measured using an accurate real-time kinematic (RTK) GPS is, on average, 40 times more accurate than raw GNSS data without RTK corrections. The Quadditch system provides an environment for exploring the effects of positioning inaccuracies on the performance of an operator who is executing a precision control task using an AR/FPV interface.

The Quadditch system also allows researchers to investigate methods for mitigating delays in telerobotic operation. A plethora of hardware and software tools have been integrated to create the game, but these subsystem components introduce computational, transmission, and mechanical delays between an operator's input and the perceived output. Since most participants will not be trained pilots, these delays could cause aversion to the game and its supporting technologies. Sakib *et al* [4] developed a predictive display methodology to mitigate telerobotic delays, although that methodology assumes there are two cameras, including one with a wide field of view (FoV). Brudnak [18], on the other hand, developed a single camera based predictive display system for a ground vehicle and tested its effectiveness in simulation. Quadditch provides an opportunity to test a single camera based real-time predictive display algorithm for use with UAVs, which is the subject of ongoing work.

Finally, and perhaps most obviously, Quadditch can be used for game-based human-subjects research. Human subjects testing requires both objective measurements (parameters that provide impartial and quantifiable outcomes of a process) and subjective measurements (parameters that depend on the human's experience). Objective measurements can be obtained from common game statistics like the score, the number of shots on goal, the number of successful "hits" with a bludger, etc. Subjective measurements can be obtained using established opinion scales like the Cooper-Harper (C-H) rating scale [19], the task-load index (TLX) [20], and the system usability scale (SUS) [21]. Combining objective measurements obtained from game statistics with subjective measurements from player ratings provides a more complete assessment of proposed innovations in AR/FPV robot interfaces.

This paper provides an overview of the Quadditch concept and gives a sense of the game's utility for outreach, education, and research. The primary focus is on describing the system and its current and future capabilities. Section 2 gives a brief conceptual overview of the game. Section 3 describes the coupled physical and virtual environments in which the game evolves, and provides more details about game play as well as some safety features. Section 4 describes the drone hardware while Section 5 describes the software and hardware infrastructure that are used to generate and superimpose AR artifacts into the players' FPV video streams. Section 6 describes continuing efforts to develop and use Quadditch as a framework for outreach, education, and research.

# 2 Game Overview

In Rowling's Quidditch, young magicians fly through the air on broomsticks, jockeying for position as they seek to score or to defend [1]. Two types of projectile – the "quaffle" and the "bludger" – play important roles in the sport. To score points, for example, a player hurdles the quaffle through one of three fixed hoops. Other players bat bludgers at their opponents to disrupt their concentration. (A third element – the "golden snitch" – evades capture by the "seeker" on either of the two teams. The golden snitch and seekers are omitted from this initial re-imagining of Rowling's sport.)

There are no flying magicians in Quadditch. Multirotor drones are controlled by individual players using AR/FPV interfaces. Virtual artifacts such as the goal posts, quaffle, and bludgers are superimposed on this video imagery, along with operational cues such as elements indicating other players' positions. Beyond enabling the basic game functionality, the primary purpose of these cues is to help each player operate the drone safely and effectively.

In its initial incarnation, Quadditch involves four human players (two on two), each of whom operates a single drone. A single offensive player can launch the (virtual) quaffle in an attempt to score points, while the remaining players can each launch a (virtual) bludger to "stun" a competitor, momentarily freezing that player's drone in space. This concept of aerial "freeze tag" is managed through a supervisory system that enables game play and ensures safety. (An administrator oversees the game, authorizing takeoff and monitoring system performance until the game's conclusion.) The implementation involves a parallel, simulated environment in which the actions of virtual agents match those of the actual drones. Each physical drone is mirrored by an avatar in the virtual environment, with the avatar's position and orientation updated using data obtained from an autopilot system on-board the physical drone. Important artifacts within this "digital twin" environment, such as the goal posts, the quaffle, and the bludgers, are rendered in each player's FPV scene from the perspective of their particular drone. Screenshots of the player interface from a developmental flight test in Virginia Tech's Drone Park are shown in Figures 1(a) and 1(b). Footage of various system elements can be viewed in a publicly available video [22].

The stability and control characteristics of the drones are modified to suit the player skill level, the game tasks, and the operating environment. For novice players, the drones respond to velocity commands with a suppressed closed-loop dynamic response so that FPV imagery from the airframe-fixed camera is not disorienting. (All drones participating in a given match have identical performance characteristics.) For players with a higher skill level, this artificial suppression of the dynamic response can be relaxed to the point of allowing direct control of the bare airframe dynamics.

As interest in Quadditch increases, along with operator skill level, the game can be scaled to include more players on each team. In keeping with J.K. Rowling's invention, for example, one might include a dedicated keeper to defend each team's goal and even expert pilot "seekers" tasked with capturing a (real or virtual) golden snitch [1].


(a) Screenshot from the Quadditch FPV interface. (b) Quadditch drone and game controller.

**Fig. 1 A screenshot and photograph for a Quadditch drone operating in Virginia Tech's Drone Park.**

# 3 Game Design

Quadditch is designed to engage players through AR/FPV interfaces that provide the real-time information needed to play the game. In addition to live, analog video imagery transmitted from each aircraft, these interfaces provide each player with operational and mission data (e.g., goal locations, time remaining, and the score). Following, we describe the AR scenario in terms of both the physical environment and the virtual environment.

## 3.1 Physical Environment

Initial developmental testing of Quadditch has taken place within the Virginia Tech (VT) Drone Park, a 300-foot by 120-foot lawn enclosed below a net that is supported by twenty 80-foot steel posts around the perimeter. Physically segregated from the US national airspace system (NAS), the facility enables outdoor drone testing beyond the purview of the Federal Aviation Administration (FAA).

The Quadditch pitch is a sub-volume of the netted VT Drone Park, with twelve-foot buffer zones between the boundaries of the virtual pitch and the physical net. All personnel (players, supervisors, and audience members) remain outside the net while the Quadditch drones are in operation.

Low-level communication with the drones, for monitoring and control, is provided by a multiple-access telemetry link to a central supervisory computer referred to as the *ground control station (GCS)*. The GCS runs the multi-agent control interface *QGroundControl* [23] which communicates with each drone's PX4 autopilot [24] using the *MAVLink* protocol [25]. Details are provided in Section 5.

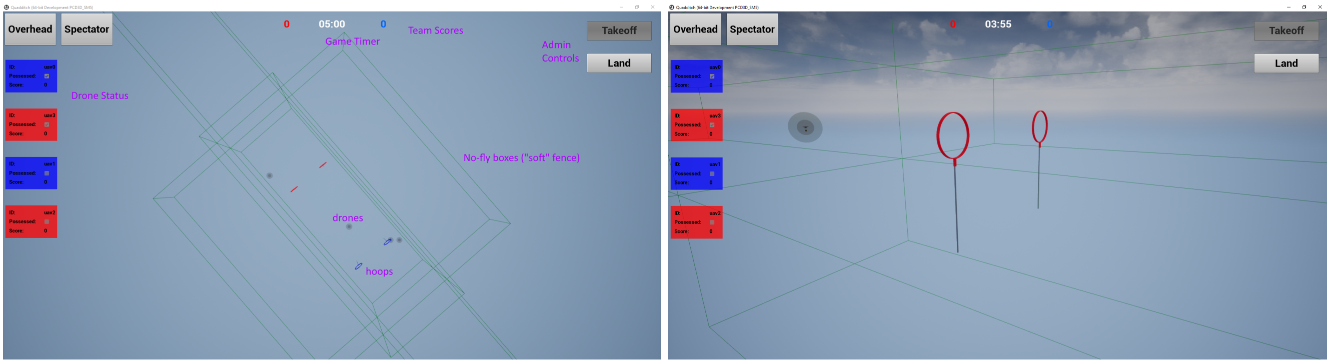
Higher-level communication between the GCS and the drones is provided by a dedicated wireless LAN. A Raspberry Pi co-computer running on each drone runs a set of ROS nodes, which communicate with a ROS master running on the central computer. Information communicated over the WLAN supports game play by enabling the physical environment to be reconciled with the virtual environment described next.

## 3.2 Virtual Environment

In order to properly render virtual artifacts in each player's live FPV stream, a virtual clone of the physical environment is instantiated using Unreal Engine (UE) [26]. A UE game server running on the GCS maintains a virtual representation of the environment encompassng all aspects of the game: the pitch, with its virtual boundary planes and the goal posts; the drones, with pose and status (offense or defense, frozen or free, etc.); and the game state (score and time remaining). The game server presents all of this information to a human supervisor and spectators in a separate administrative interface; see

Figure 2. In addition, the game server provides vital information to the UE game clients which run on each of the dedicated player computers.

Using knowledge of the game state obtained from the UE game server, each player's computer generates virtual artifacts that are overlaid on the live FPV stream from the associated drone. These artifacts include georectified representations of any fixed or moving virtual objects that should appear within the current field of view (FOV), such as goal posts or projectiles, and virtual representations of the drones operated by teammates or opponents. These artifacts also include relevant operational and mission data that does not need to be rectified within the FOV, such as the game score and the time remaining. A challenge of system design is to accurately synchronize the presentation of artificial information with the video imagery obtained from the drone cameras, as presented to the operators, despite the fact that critical information must pass through multiple layers of communication and processing along the way.




(a) Overhead view in administrator interface

(b) Spectator view in administrator interface

**Fig. 2 Two views provided by the game administrator interface**

## 3.3 Game Play

A Quadditch match has a time limit of ten minutes, which is well within the endurance of the Quadditch drones. Four players – two on each team – are each given control of a drone at the beginning of the match by the game administrator. A signal indicates when the drones may take off from the ground. One team is randomly designated as the offense, making the other team the defense, and one player on the offense is randomly selected to carry the quaffle. Each of the remaining three players then carries a bludger. These elements can be launched by players as virtual projectiles with a limited range.

If the player with the quaffle is within range of the goal for that player's team, the goal is highlighted to indicate this fact and the player may score one point by successfully firing the quaffle through the goal. If the player misses the goal, a new quaffle is spawned and assigned to the opposing team's drone that is farther from that team's goal. (The quaffle replaces that player's bludger.) Thus, the roles of offense and defense are reversed.

If any player is within range of any other player, the fact is highlighted to indicate that these players may launch bludgers at one another. If a player launches a bludger and succeeds at hitting an opponent, that opponent's drone is frozen at its current position and heading. A "miss" is followed by a period during which the assailant is without a bludger, after which a new bludger is spawned. To accommodate the relatively small size of the Quadditch drones, a spherical region of influence is centered on each drone which serves to trigger a successful bludger hit.

The winning team is the one with more points when time expires.

### 3.4 System Safety

To ensure system safety, virtual walls and a virtual ceiling are imposed on the Quadditch pitch. For operation within a netted facility, these virtual boundaries are offset some distance from the physical boundaries (12 feet for the VT Drone Park). The drone's proximity to these virtual boundaries is indicated in the interface. The boundary has two components: a "soft" fence marking the boundary of the Quadditch pitch and and a "hard" geofence intended to protect the drone and the netted facility. The soft fence prevents drones from passing through, however the player is free to continue moving in other directions (e.g., sliding along the virtual wall). The hard geofence is placed between the soft fence and the physical net. Hitting the hard geofence means the drone somehow passed through the soft fence, which might be caused by a strong gust or a game malfunction. Colliding with the hard geofence revokes player control and causes the drone to land, a process that is governed by the higher priority flight control system rather than the game environment.

A soft boundary sphere also encloses each drone to prevent it from colliding with other drones. In the game, these soft boundaries appear as translucent bubbles containing each player. These collision events trigger situational awareness icons to appear on the screen, letting the player know that their drone has collided with a virtual obstacle.

## 4 Quadditch Copters

The drone chosen to serve as the game platform is the commercially available DJI Flame Wheel F450 quadrotor kit. The Flame Wheel was chosen for its modularity and payload capacity, to accommodate the required instrumentation. Each aircraft is powered by a four-cell 14.8V 5200mAh lithium-polymer battery providing approximately 25 minutes of endurance. While the kit provides a number of essential components, the Quadditch drones have been customized to include important additional components, as indicated in Fig. 3.

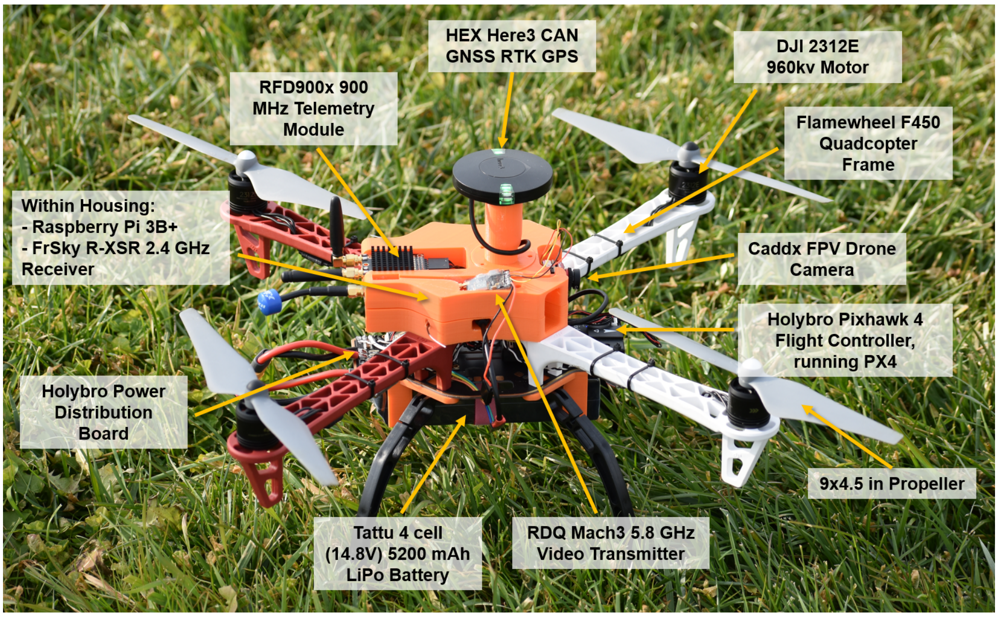



**Fig. 3 A Quadditch drone**

A Pixhawk flight computer, specifically a Holybro Pixhawk 4 flight controller running PX4 [24] autopilot software (firmware v1.11.3) serves as the primary flight control system (FCS) for each drone. The PX4 provides motion data including filtered measurements of three-dimensional position, attitude, velocity, and angular velocity as well as health-related data such as GPS and communication signal strength. While the default Pixhawk 4 hardware and software provides stable altitude and position estimates, the position accuracy is not adequate to enable Quadditch. Because the drones are projected onto the AR overlay based on their position, uncertainties no greater than a few centimeters can be tolerated for effective game play. To achieve this position accuracy, real-time kinematic (RTK) GPS units were integrated with the Pixhawk 4. Each drone's HERE3 RTK GPS receiver provides centimeter-accurate lateral position estimates and communicates using the Pixhawk's UAVCAN port and protocol. The RTK system uses a separate GPS module on the ground to calculate position errors, sending them to the vehicle via the telemetry link.

The PX4 firmware provides safety features such as a programmable return-to-base behavior that can safely end the flight if the telemetry link is lost. Additionally, an on-board co-computer is connected via Universal Serial Bus (USB). The co-computer is a Raspberry Pi 3B+ running the Ubuntu 18.04 Linux distribution, communicating with the Pixhawk using MAVLink protocols and using the Robot Operating System (ROS) Melodic framework. The co-computer receives commands from the host game computer and relays velocity commands to the Pixhawk, simultaneously sending pose updates back to the game host. The Pixhawk in turn controls the aircraft's propellers to attain the commanded velocity.

# 5 Augmented Reality Infrastructure

We describe the AR infrastructure in terms of the overall system architecture in Section 5.1 and provide a brief overview of the software in Section 5.2.

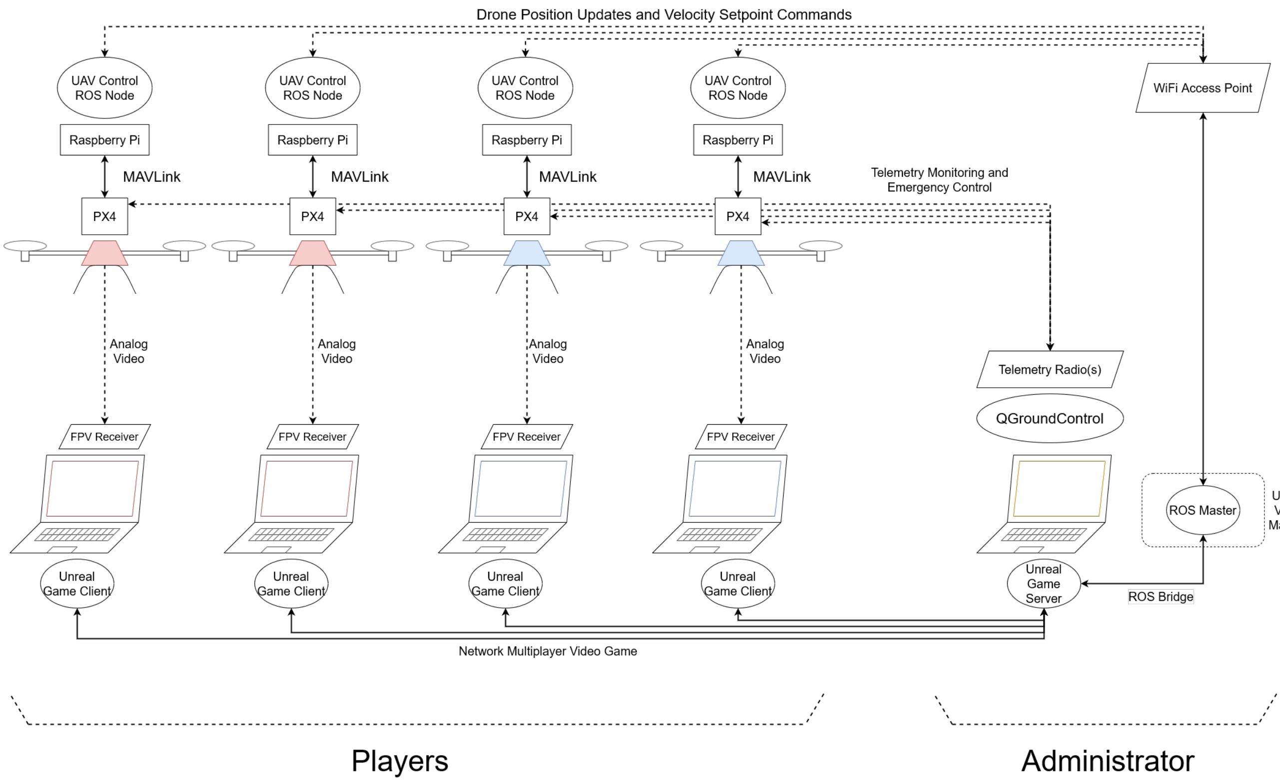



**Fig. 4 Quadditch architecture**

## 5.1 System Architecture

The system architecture is depicted in Figure 4. The lowest level computer on a given drone, the Pixhawk flight computer, handles control algorithms and peripheral interfaces for all hardware components onboard the drone. The Pixhawk flight computer connects to a Raspberry Pi co-computer via the MAVLink protocol using a USB port. (A special parameter must be set to enable flight while a USB cable is connected.) This Raspberry Pi runs ROS Melodic on Ubuntu 18.04. The MAVROS package is used to connect to the Pixhawk. This library exposes a wide variety of flight controller functions through ROS topics, services, and parameters. These are fed over a WiFi link to the ROS master running on a virtual machine hosted by the GCS computer. This computer also runs the QGroundControl GCS software and the Unreal Engine (UE) host game. QGroundControl allows monitoring of telemetry data and remote commands, in case of emergency. The Unreal Quadditch host game connects to the ROS master using a UE plugin called ROSIntegration. This plugin connects to rosbridge and allows entities in the game to publish and subscribe to topics.

The host game includes an administrative interface which displays the status of all game drones and allows the administrator to start and end games. Finally, this information is mirrored to all of the game client instances, each of which runs on a separate laptop. Each laptop is granted control of a single drone using an XBox controller (Figure 5) and displays the corresponding FPV feed through a USB video receiver. Ideally, this FPV feed could be displayed directly within the UE4 game, but this proved unworkable with the available video types. As a workaround, we use the ManyCam software to open the video stream and replay it in a wrapper compatible with UE4. A prototype of the player view can be seen in Figure 1(a).


**Fig. 5 Game control interface**

Developing the Quadditch architecture involved integrating a number of trusted hardware and software components. For example, the Pixhawk flight controller hardware and the PX4 firmware are capable of accurately estimating the position and orientation of the drone, making it unnecessary to develop a custom navigation system. A great deal of effort focused on the creative (and often tedious) integration of existing elements. The integration of aerial robots, in particular, distinguishes the effort from other work on *simulating* drones with Unreal Engine [27]. To our knowledge, Quadditch is the first effort to implement real-time synchronization among multiple, flying drones in a multiplayer, AR/FPV game using Unreal Engine.

## 5.2 Software

Two software systems were written for this project: `quadditch-ros` and `quadditch-unreal`. The `quadditch-ros` software runs on the Raspberry Pi aboard each drone and handles the interface with MAVROS. Its primary role is to abstract complicated functions like the startup sequence and takeoff commands, allowing the game component to issue simple commands. The `quadditch-unreal` software contains the bulk of the game content, with everything from the object meshes to the overarching game logic. These packages (and auxillary components) can be found in our GitHub organization [28].

# 6 Continuing Development and Research

Quadditch was created to advance and promote augmented reality, first-person-view (AR/FPV)interfaces to enable the broader use of aerial robots for a wide range of applications. As a competitive game Quadditch provides direct, objective measures of an AR/FPV interface's effectiveness. A developer or researcher can easily introduce or modify virtual artifacts within the player's view and examine the effect of these changes by comparing game statistics (the score, the number of bludger hits, etc.) before and afterward.

Ongoing efforts are aimed first at finalizing an intuitive baseline AR/FPV interface and then at answering related engineering research questions. In these efforts, human subjects studies are conducted involving separate groups of novice and expert volunteers who play the game as designed. Volunteers first play games to develop their proficiency, as measured by convergence of the total game score, and then play additional games to generate research data. All games are recorded, along with player dialog. Game play is followed by focus group discussions to review and document player experiences.

The human factors research opportunities related to AR/FPV interfaces that are enabled by Quadditch are wide-ranging. By making the details of the system architecture publicly available, the authors hope that researchers who study human/technology interaction might replicate and adapt Quadditch to support their own investigations. In the narrower realm of aerospace human factors, the authors have a particular interest in the effect of time delay on operator performance when using an AR/FPV interface. Accumulated time delays due to computation, signal transport, data encryption, data compression, and error correction can interfere with operator performance and may require active mitigation such as a predictor [4, 29]. As it stands, the Quadditch system does not exhibit noticeable delay. Since all critical components are connected via a dedicated high-speed network (or analog video link), the full-path delay is on the order of only a few milliseconds. It is possible to introduce artificial delays, however, in any combination of the video display, AR artifact rendering, and command and control.

The initial research question we will answer using Quadditch is the following: What modifications to the AR/FPV interface enhance player training and performance? Again, the fact that Quadditch is a competitive game provides direct, objective measures of system performance, such as shooting accuracy, in addition to subjective measures such as the Cooper-Harper (C-H) rating scale [19] or the system usability scale (SUS) [21]. Subjective measurements representing perceived system performance can be compared with objective measurements representing actual system performance to understand how various design elements influence interface effectiveness and user acceptance.

At the time of this writing, human subjects testing had not commenced, but the test procedures have been developed. The flight dynamics of the Quadditch drones can be modified to provide three levels of performance in response to user inputs – sluggish, normal, and fast. Two categories of operators are being recruited – experienced and inexperienced. All players will begin using the system with the drones in their 'sluggish' mode. Objective and subjective measurements will be tracked across several games. With sufficient practice, it is expected that the operators' preferences will shift towards the 'normal' or 'fast' mode. We hypothesize that experienced drone operators will adopt the most responsive mode, and perform better, by both objective and subjective measures, more quickly than inexperienced operators. We want to understand if adding AR augmentations to the FPV image stream can speed up this learning process for both categories of users. Within both categories – experienced and inexperienced – one group will be presented with an AR display system while the other (control) group will operate without any supporting augmentations. We will document the subjective and objective measures described earlier to see how these measures correlate with changes in the AR display. For example, for one group of players, augmentations will be included that enhance the visibility of the other drones. We hypothesize that these modifications will improve the ability of players to "stun" one another with bludgers.

# CONCLUSION

Augmented reality, first-person-view (AR/FPV) interfaces are an enabling technology for a wide range of aerial robotics applications, but it is important to ensure that these interfaces promote effective human/robot collaboration. Quadditch provides a useful tool for developing AR/FPV interfaces, and measuring their effectiveness, and also for promoting science, technology, engineering, and mathematics (STEM). In this multiplayer game, the human competitors each control a drone using an AR/FPV interface. The physical state, including drone positions and orientations, is replicated synchronously in the Unreal Engine environment which in turn presents virtual elements, such as bludgers, goal posts, and the scoreboard, to the players from their drone camera's perspective. Beyond using Quadditch as a tool for STEM education and outreach, ongoing efforts involve human subjects research focused on player training and performance using AR/FPV interfaces. The Quadditch software and hardware architecture have been carefully documented and the software is freely available, with the exception of licensed third-party content, through an online repository.

# ACKNOWLEDGMENT

The authors gratefully acknowledge Kyle Rearwin, Kushal Patel, Sabrina Wang, and Alan Huang for contributing to the development of Quadditch. Thanks also to Sam Woolsey for input concerning the game rules and to Michael Marcolini for suggesting the name. We are also grateful to the anonymous reviewers for their constructive input. This effort was sponsored in part by NSF Grant No. IIS-1840044.

# References

[1] J. K. Rowling. *Harry Potter and the Philosopher's Stone*. Bloomsbury Children's Books, London, U.K., 1997.

[2] S. M. Herrington, M. J. H. Zahed, and T. Fields. Pilot training and task based performance evaluation of an unmanned aerial vehicle. In *AIAA Scitech 2021 Forum*, 2021. DOI: 10.2514/6.2021-0595.

[3] A. Çöltekin, I. Lochhead, M. Madden, S. Christophe, A. Devaux, C. Pettit, O. Lock, S. Shukla, L. Herman, Z. Stachoň, et al. Extended reality in spatial sciences: A review of research challenges and future directions. *ISPRS International Journal of Geo-Information*, 9(7):439, 2020. DOI: 10.3390/ijgi9070439.

[4] N. Sakib, J. Gresham, and C. A. Woolsey. Usability studies of a predictive heterogeneous vision system in mitigating the effects of visual display delay. In *AIAA Scitech 2021 Forum*, 2021. DOI: 10.2514/6.2021-0017.

[5] J. van Dam, A. Krasner, and J. L. Gabbard. Augmented reality for infrastructure inspection with semi-autonomous aerial systems: An examination of user performance, workload, and system trust. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pages 742–743. IEEE, 2020. DOI: 10.1109/VRW50115.2020.00222.

[6] J. Seo, L. Duque, and J. Wacker. Drone-enabled bridge inspection methodology and application. *Automation in Construction*, 94:112–126, 2018. DOI: 10.1016/j.autcon.2018.06.006.

[7] A. Das and C. A. Woolsey. Workspace modeling and path planning for truss structure inspection by small UAS. *Journal of Aerospace Information Systems*, 16(1):37–51, January 2019. DOI: 10.2514/1.I010634.

[8] K. Dorling, J. Heinrichs, G. G. Messier, and S. Magierowski. Vehicle routing problems for drone delivery. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(1):70–85, 2016. DOI: 10.1109/TSMC.2016.2582745.

[9] Y. Li and C. Liu. Applications of multirotor drone technologies in construction management. *International Journal of Construction Management*, 19(5):401–412, 2019. DOI: 10.1080/15623599.2018.1452101.

[10] S. Chowdhury, A. Emelogu, M. Marufuzzaman, S. G. Nurre, and L. Bian. Drones for disaster response and relief operations: A continuous approximation model. *International Journal of Production Economics*, 188:167–184, 2017. DOI: 10.1016/j.ijpe.2017.03.024.

[11] D. Raymer. *Aircraft Design: A Conceptual Approach, Sixth Edition*. 09 2018. DOI: 10.2514/4.104909.

[12] J. Roskam and C. T. E. Lan. *Airplane Aerodynamics and Performance*. Airplane design and analysis. Design, Analysis and Research Corporation, 1997.

[13] S. Kontogiannis and J. Ekaterinaris. Design, performance evaluation and optimization of a UAV. *Aerospace Science and Technology*, 29:339–350, 08 2013. DOI: 10.1016/j.ast.2013.04.005.

[14] D. T. McRuer. Development of pilot-in-the-loop analysis. *Journal of Aircraft*, 10:515–524, 1973. DOI: 10.2514/3.44389.

[15] Y. Angel, D. Turner, S. Parkes, Y. Malbeteau, A. Lucieer, and M. McCabe. Automated georectification and mosaicking of UAV-based hyperspectral imagery from push-broom sensors. *Remote Sensing*, 12:34, 12 2019. DOI: 10.3390/rs12010034.

[16] H. Xiang and L. F. Tian. Method for automatic georeferencing aerial remote sensing (rs) images from an unmanned aerial vehicle (UAV) platform. *Biosystems Engineering - BIOSYST ENG*, 108:104–113, 02 2011. DOI: 10.1016/j.biosystemseng.2010.11.003.

[17] J.-C. Padró, F.-J. Muñoz, J. Planas, and X. Pons. Comparison of four UAV georeferencing methods for environmental monitoring purposes focusing on the combined use with airborne and satellite remote sensing platforms. *International Journal of Applied Earth Observation and Geoinformation*, 75:130–140, 03 2019. DOI: 10.1016/j.jag.2018.10.018.

[18] M. J. Brudnak. Predictive displays for high latency teleoperation. In *NDIA Ground Vehicle Systems Engineering Technology Symposium*, 2016.

[19] G. E. Cooper and R. Harper. The use of pilot rating in the evaluation of aircraft handling qualities. Technical Report TN-D-5153, NASA, 1969.

[20] S. G. Hart and L. E. Staveland. Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. *Advances in Psychology*, 52:139–183, 1988. DOI: 10.1016/S0166-4115(08)62386-9.

[21] J. Brooke. SUS: a retrospective. *Journal of Usability Studies*, 8(2):29–40, 01 2013. DOI: 10.5555/2817912.2817913.

[22] Elements of Quadditch. `https://youtu.be/wTZ9vGiRTss`, Accessed March 2022.

[23] QGroundControl Website. `http://qgroundcontrol.com/`, Accessed December 2021.

[24] Pixhawk 4 Flight Controller. `https://docs.px4.io/master/en/flight_controller/pixhawk4.html`, Accessed December 2021.

[25] MAVLink Website. `https://mavlink.io/en/`, Accessed December 2021.

[26] Unreal Engine Website. `https://www.unrealengine.com/`, Accessed December 2021.

[27] AirSim Wiki. `https://microsoft.github.io/AirSim/`, Accessed March 2022.

[28] Quadditch GitHub. `https://www.github.com/Quadditch`, Accessed March 2022.

[29] S de Vries. UAVs and control delays. Technical Report TNODV3-2005-A054, TNO Defence Security and Safety, 2005.