Demo Preliminary Implementation Study

Matthew Paternoster, Matthew Wyatt, Tutku Gizem Guder, Brenden Martins, Tarek Kayali

Group 32

Florida Atlantic University
EGN 4950C Engineering Design 1

Drs. Waseem Asghar, KwangSoo Yang, Xiangnan Zhong

Target Detection Subsystem (Tutku Gizem Guder)
The Target Detection Subsystem is a critical component of the multi-drone coordination system for addressing challenges presented in disaster scenarios such as hurricanes and power outages. This system will enable drones to rapidly locate and identify individuals or hazards within a realistic 3D simulated environment using Microsoft AirSim. The enhanced detection capability from this subsystem would improve the effectiveness of disaster response, including rescuing, scientific research, and future real-world applications.

A huge deal of progress has been made for establishing the Target Detection Subsystem. The first phase is completed, which involves installation and configuration of Microsoft AirSim for the purpose of creating a solid environment for simulation to mimic a disaster scenario. In addition, an enormous amount of research regarding the different models and algorithms was conducted; based on that, YOLOv5 was chosen for object detection due to its real-time performance and high accuracy. YOLOv5 is very good at handling various scales of objects, making it suitable for the identification of human victims and hazardous elements in complex environments. Besides, tracking algorithms such as SORT have been selected to monitor detected objects across frames, enhancing target localization during dynamic rescue operations.
The simulation currently focuses on the detection process, integrating YOLOv5 with AirSim to process sensor data like camera feeds and validate initial outputs in simpler disaster scenarios. Further implementation will involve fine-tuning of the model by hyperparameter adjustment and transfer learning of the model on disaster-specific datasets to enhance the precision of detection. It will then be followed by the addition of more complex scenarios, like occlusions, low-light conditions, and diversity in terrains, in this simulation in order to check on the robustness of the system. Finally, comprehensive testing will be carried out on the accuracy, speed, and reliability to ensure the subsystem is prepared for real-world applications.
Time Table for the Subsystem

Task ID	Task Name	Process
1	Install and Configure Microsoft AirSim	Done
2	Research Models and Algorithms	Done
3	Simulate the Process	Doing
4	Fine tune the model	To do
5	Add complex scenerios	To do
6	Testing	To do

User Interface Subsystem (Matthew Paternoster)

As of now, the User Interface (UI) Subsystem for the Multiple Drones Coordination System (MDCS) is in the initial development phase. My primary focus is on designing a a platform for the user interface. Before, my initial idea was to use React Native, however, I now plan on using Electron or PyQt instead of creating a mobile app which will have a more seamless integration with AirSim.The planned interface will enable operators to monitor drone activities, view real-time data, and send control commands seamlessly. Unlike the original implementation plan, which outlined basic monitoring and control features, I am incorporating additional functionalities such as historical mission logs for better analysis and decision-making. Furthermore, I intend to integrate push notifications to alert operators of critical events instantly, enhancing responsiveness during emergencies. These enhancements go beyond the initial scope by providing more comprehensive data visualization and improving user engagement.

Another key difference is the emphasis on user authentication and role-based access, ensuring that only authorized personnel can control the drones and access sensitive information. While the original plan focused on fundamental real-time updates, the updated approach aims to deliver a more robust and user-friendly experience by adding advanced features and security measures. Moving forward, I will be working on optimizing the app’s performance to handle high-frequency data streams and ensuring smooth integration with the backend systems for reliable data flow. This proactive approach will address potential challenges early on and help create a more effective and efficient user interface for the MDCS.

Drone Control Subsystem (Tarek Kayali)

The Drone Control Subsystem is a pivotal element in the multi-drone coordination system, ensuring smooth navigation, real-time stability, and robust communication for autonomous operations in complex scenarios. Designed to enhance efficiency and safety, the subsystem employs advanced algorithms to coordinate multiple drones simultaneously. This subsystem is particularly crucial in disaster response missions, where precise movement and reliable decision-making can significantly impact the success of operations.

Significant progress has been made in developing the Drone Control Subsystem. The initial phase involved setting up a simulation environment and implementing a basic pathfinding algorithm to enable drones to traverse from one point to another efficiently. Following this, extensive research was conducted to compare and evaluate different control methods, leading to the adoption of Model Predictive Control (MPC) for dynamic trajectory optimization. MPC offers a balance of computational efficiency and adaptability, ensuring drones can react swiftly to environmental changes while maintaining optimal paths. To complement this, a collision avoidance mechanism leveraging sensor data and real-time processing was integrated to prevent mid-air incidents.

Currently, the focus is on testing and refining the control algorithms in a simulated disaster environment. This involves evaluating drone behavior in scenarios involving obstacles, varying terrains, and unexpected disruptions. The subsystem is being further enhanced by incorporating machine learning techniques to improve its ability to learn from past missions and make better decisions over time. Future steps include fine-tuning the system for real-world applications and introducing features like task reassignment and autonomous recharging. Comprehensive testing will be carried out to ensure reliability, scalability, and readiness for deployment in diverse scenarios.

Simulation Management Subsystem - Brenden Martins

As the team member responsible for the Simulation Management Subsystem, my primary focus was to develop a foundational 3D simulation environment for the project demo, serving as a proof of concept rather than a finished product. This involved creating a basic placeholder environment using Blender and Unreal Engine 4 (UE4) to utilize previously created assets and texture/map them out. Blender was used to model simple terrain features, while UE4 provided a framework for rendering and interaction. Although the environment currently lacks the complexity envisioned in the original implementation plan, it effectively demonstrates the system's potential to support drone operations in a realistic simulation.

One of the initial goals was to incorporate real-time terrain mapping using data from platforms such as Bluemarblegeo - Global Mapper 26.0 and the USGS 3D Elevation Program (3DEP). While these tools were explored and integrated conceptually, their full functionality was not utilized in this demo. Instead, a static terrain modeled in Blender and imported into UE4 serves as the foundation for the demo environment. Dynamic features like real-time obstacle placement and weather simulation were scoped out of this phase and remain as planned future enhancements.

The subsystem’s real-time data streaming capabilities, which are critical for its integration with other subsystems, were also addressed at a conceptual level but have not yet been fully implemented. This decision reflects the project's focus on showcasing the overall framework in the demo rather than completing all technical components. The environment, while functional, serves as a placeholder that will be replaced with a more complex and interactive system in later iterations.

Despite these limitations, the work done thus far aligns with the project’s goal of demonstrating the potential of the Multiple Drones Coordination System. Learning to use tools like Blender, UE4, and real-world datasets such as those from USGS, provided valuable insight into the processes needed for future development. This demo lays the groundwork for a scalable and dynamic simulation environment, with clear next steps to refine and expand its functionality.

Data Management & Communication Subsystems (Matthew Wyatt)

For the Data Management Subsystem, my initial effort involved setting up a MySQL database on FAU's lamp server at lamp.cse.fau.edu. This database serves as the backbone for storing and managing the operational data securely. The starting phase included configuring the database server. Although the database tables have not been set up yet, this will be done once we have the data flowing from the other subsystems. This current setup provides a robust foundation for data management, aligning with our goal of using a reliable and scalable database system.

For the Communication Subsystem, my primary effort involves setting up the network infrastructure and wireless communication modules to facilitate reliable data exchange between drones and control systems. This setup ensures stable connectivity and low-latency communication, which is crucial for real-time operations. For now, I have established the initial files for basic API connections using Express.js to facilitate data exchange between the drones and the database. These APIs allow for real-time data insertion, retrieval, and updates, ensuring that the system can handle the dynamic nature of drone operations. I also implemented basic security measures during setup, including encryption protocols to safeguard data transmission. The current situation provides a robust foundation for communication, aligning with our goal of ensuring secure and efficient data exchange. The network infrastructure and wireless communication modules were selected for their reliability and performance under high-load conditions.

Comparing these implementations to the initial plan, we have successfully built robust foundations for both data management and communication. The MySQL database setup for the Data Management Subsystem aligns with our goal of using a reliable and scalable database system. The network infrastructure, wireless communication modules, and API connections for the Communication Subsystem ensure stable and low-latency communication. Moving forward, the focus for the Data Management Subsystem will be on defining and creating the necessary database tables, optimizing database performance, and ensuring seamless integration with other subsystems. This will involve implementing advanced indexing techniques and conducting thorough testing to ensure data integrity and security under various operational scenarios. For the Communication Subsystem, the focus will be on enhancing the security mechanisms, including multi-factor authentication (MFA) and role-based access control (RBAC), to verify user identities and prevent unauthorized access. This will involve refining the encryption protocols, optimizing network performance, and conducting thorough testing to ensure the system's responsiveness and reliability under various operational scenarios.