TECHNICAL DESIGN PAPER

CNU UAS Team - Christopher Newport University

ASSOCIATION FOR UNMANNED VEHICLE SYSTEMS INTERNATIONAL
STUDENT UNMANNED AERIAL SYSTEMS COMPETITION

Figure 1: CNU UAS Anaconda Aircraft

Abstract

Christopher Newport University Unmanned Aerial Systems' (CNU UAS) goal for the AUVSI SUAS 2018 competition was to design a working modular system that can withstand human error, complete regulated tasks as well as follow and execute daunting missions. In preparing for competition, CNU UAS' sole goal was to build a system that surpassed previous years. CNU's Unmanned Aerial System (UAS), Anaconda, was the creation of over 40 undergraduate students from various fields of study including Electrical Engineering, Computer Science, Aerospace Engineering, Computer Engineering, Business, and Psychology. The combined knowledge allowed CNU UAS to account for the perceptions of multiple fields regarding the aircraft's design and risk mitigation analysis. The Anaconda is a combination of methodical stability airframe with modular parts, a potent power system, an immaculate circulated imagery system, autopilot, and a specially designed ground station.

Table of Contents

1. Systems Engineering Approach

1.1 Mission Requirements Analysis

An analysis of the overall purpose is important in order to understand the mission and ultimately succeed at the competition. The mission simulates a search and rescue mission that involves several components including adequate Mission Duration, Obstacle Avoidance functionalities, autonomous flight, and Object Classification and Detection executed through a joint manual and autonomous image recognition process.

Task	Description	Requirements for Mission Success
Timeline (10%)	● Mission Time (80%) ○ 45 minutes ○ Flight time ○ Post processing time ● Timeout (20%) ○ Used in unsafe or other circumstances	● Practice and rehearse the flight line setup and system pre-flight checklist. ● Run the mission several times to rehearse for the flight on the day of the mission ● Completion of the mission on the day of the flight in accordance with the rehearsal
Autonomous Flight (30%)	● Autonomous flight (40%) ○ No manual control ● Waypoint capture (10%) ● Waypoint accuracy (50%)	● Run several test runs on flight field to make sure that the plane is tuned correctly ● Conduct full autonomous flight tests
Obstacle Avoidance (20%)	● Stational obstacle avoidance (50%) ● Moving obstacle avoidance (50%)	● Utilization of specific pathing algorithm to inject path around obstacle ● Integration with Mavlink to insert waypoint
Object Detection, Classification, Localization (20%)	● Geolocation (30%) ● Actionable (30%) ● Autonomy (20%)	● Run several test photos to train the feature vector in the Convolutional Neural Network ● Test the system using images from test flights to upload data to a sample competition server ● Test the camera gimbal with flights
Air Delivery (10%)	● Delivery Accuracy	● Not attempting
Operational Excellence (10%)	● Professionalism ● Communication ● Reaction to unexpected events ● Safety	● Practice under competition conditions with each member at their respective role ● Utilize mentors and veteran members as sample judges for test flights ● Perform two forms of communication based on status updates and anomalies

Table 1: Mission Requirements Analysis

Task Analysis

CNU UAS identified the requirements for successful mission execution as shown in Table 1. Because the *Autopilot Task* requires the use of an airframe and a working aircraft, the necessary conduct of research and development dictates that the task be a priority. This is the rationale behind a high priority behind this task. However, given that the development for successful flight is routine, not much research is required to conduct a successful flight; A flight which includes autonomous takeoff and landing. Given the Flight Subteam's experience in this area of interest, The CNU UAS Team is confident in its capabilities to conduct a fully autonomous flight. Handling the mission duration is another matter.

The *Mission Duration* requirement necessitates several test flights to insure we are within the threshold of the time allotted to conduct a successful flight. Testing this requirement is essential to ensure the mission is successful as many of this year's members have not flown at previous competitions. To properly train new members of the CNU UAS Team, the team studies the system in groups referred to as subteams. Each subteam is led by an experienced member who serves as mentor. Due to strenuous flight testing and training methodology, the CNU UAS Team is confident in the capabilities and skills of the team's new members to ensure a successful mission.

The *Obstacle Avoidance* task requires a Software System capable of calculating waypoint paths for the plane, as well as operating the plane based off waypoints needed for those calculations. To calculate waypoints, the Software Subteam's design was based off an A* structure mechanism involving the use of a variable heuristic taking into account distance from an obstacle, its size, as well as whether the obstacle is moving or not. The heuristic was developed to incorporate all the aforementioned elements to insure the plane does not stray too far away from an Obstacle.

The *Object Detection, Classification and Localization (ODCL)* task is jointly handled by the team's Payload and Software Subteam. As a result, the primary difficulty in preparing for the task was organizing the two subteams' duties, requisitions, and responsibilities. Designating roles to the subteams helped ensure project management was coordinated between the subteam leads. Due to their combined efforts, the software subteam was able to utilize infrastructure the payload subteam had to simplify testing their modules. This facilitated on-site testing to mitigate redundant testing or lost flight time on the field.

To further assist with task completion, the team took advantage of a Convolutional Neural Network (CNN) to handle all the required image recognition necessary for the automation process. The CNN involved a feature vector that needed to be trained with several sample photos taken from various flights. As a result, several test flights with photos being taken continuously throughout the flight were required. Numerous flights ensure the CNN is adequately trained. Since training the CNN is essential, this task also required a high-resolution camera, the Canon G15, in order to conduct the execution of the task in an acceptable manner.

1.2 Design Rationale

The CNU UAS Team has designed the system to maximize functionality and performance in accordance with the information specified in Table 1. The each task's value is also specified to provide an indication as to the level of attention given to each task to be completed.

Mission Requirements

The core requirements for a successful flight revolve around adequate battery life, adequate level of power consumption and flight worthiness of the UAS. Through thorough analysis of core requirements, the Team decided to select a carbon fiber reinforced foam airframe. In addition to the analysis, a major aspect taken into account when making the final decision was were lightweight design of the airframe. The design of the airframe makes the plane easy to transport, and provides more payload capacity then previous models. Increased payload assists with the completion of the ODCL task as well as the Interoperability task by letting us to carry the DSLR camera and the on-board computational systems in the UAS; these components are critical because they permit transmitting information required by the judges on competition day. Another factor taken into account was the role aircraft maintenance plays in mission success.

The Anaconda's foam exterior is easy to repair in the case such services are necessary as it expands when heated water is poured on it. This permits quick repair in the event that a crash is experienced during testing. Additionally, the ease of repair limits the amount of mission time that would be lost should the aircraft incur damage.

A few other design features were taken into account when selecting this airframe. The factors were considered to enhance safety during the mission as well as reduce the amount of flight time needed during the mission. For example, one of the central means of control for the plane is a yaw axis tilt, so a rear propellor was used to enhance flight stability and maneuverability during the search-area task.

The CNU UAS Team believes the decision to use the current airframe for the aforementioned reasons meets the requirements to allow for mission success.

1.3 Programmatic Risks and Mitigation

Prior to development of the RMRC Anaconda System for mission success, programmatic risks were assessed and evaluated by the team. Past experiences from previous competitions played an essential role in evaluating possible modes of failure for the system. Potential instances of failure were brainstormed, utilizing the team's experience to prepare for worst-case scenarios. We compiled all topics into a list of potential failure modes using a mechanism of failure modes and effects analysis.

Mode of Failure	Likelihood	Impact	Risk Mitigation
Failure to meet competition requirements	Very Low	Very High	Mitigation of this risk happens long term. It requires constant testing of the system built into the development time over a the entire development period. At the same time, we also strive to maintain backwards-compatible components to the system. This ensures that we meet the most basic requirements.
Subsystem integration problems	Medium	High	Due to the nature of the software written for individual components and the ability to transfer information easily each component is developed with a care API. We also archive legacy systems as a means of utilizing old mechanisms for use in the worst case scenario.
Insufficient training for new recruits	High	Medium	In order to mitigate against lack of experience we require all members attending competition to attend mandatory test-flight sessions to gain flight line experience as well as operational experience regarding mission conduct.
Component damage during testing	Very Low	Medium	Due to the carbon fiber reinforced foam structure of the plane, the repair, and reconstruction process is very easy given the few tools needed to conduct a successful repair.

Table 2: Failure Mode Effective Analysis Overview (FEMA)

The CNU UAS team assessed each of the proposed risks that were detrimental to mission success. To insure the risks were mitigated, routine checks against the overall risk were assessed emphasis was placed on tasks deemed to highly impact mission success. Subsystem team leads were selected to minimize delays by permitting concurrent system development. The subteam lead leads also served as safety officers. As safety officers, the individual's main responsibilities were to ensure proper training of each team member. In addition to system knowledge, training included flightline responsibilities and etiquette.

2. Systems Design

2.1 Aircraft

Airframe

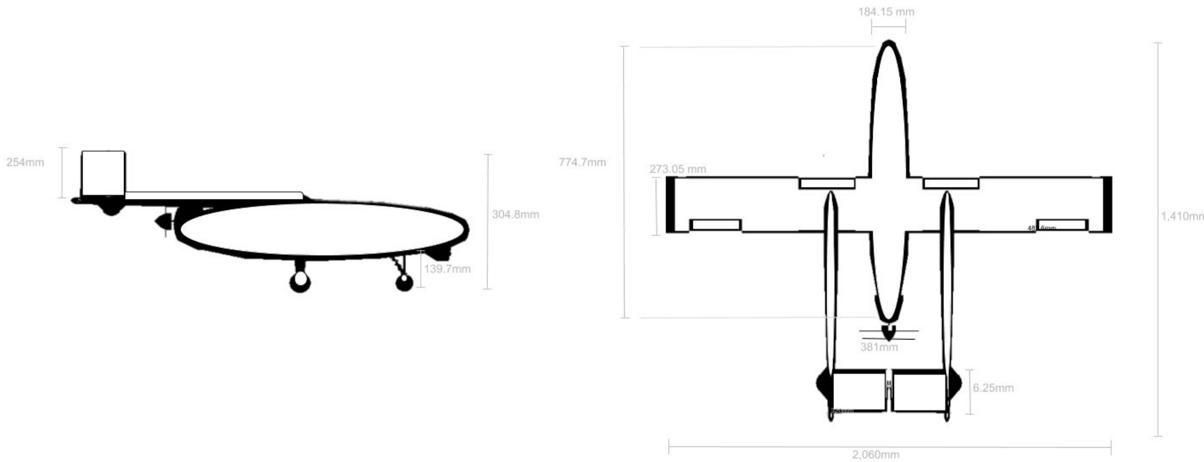

Figure 2: Specification View and Measurement of the Plane

The aircraft is based on a heavily modified ReadyMade RC Anaconda airframe. As a fixed-wing plane, it features a robust carbon-fiber reinforced structure with a spacious center fuselage pod, and two carbon-fiber reinforced foam wings. The narrow wings enhance maneuverability of the aircraft. The usage of a plane rather than a rotary copter aircraft was based on two criteria: payload and flight time. The aircraft carries a lot of equipment that weighs down on the airframe. Considering payload capacity was critical as inadequate capacity could result in mission failure.

In addition to payload capacity, a low flight time was essential to mission success. A rotary flight system decreases maneuverability thereby increases flight time. The Anaconda has a typical speed of 60 knots which letting us cover extensive portions of the search area in a short timespan. For these reasons, the Team decided a fixed-wing aircraft provided an increased likelihood of task completion than a rotary copter system.

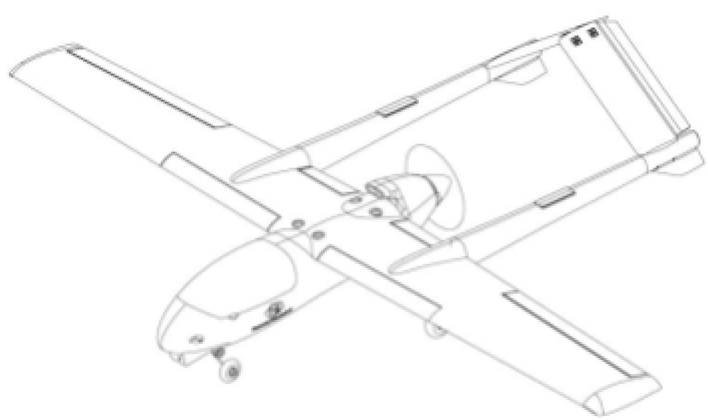

Figure 3: Design and View of the External of the Plane

The team decided fixed landing gear was an important component of the aircraft. The tricycle design provided an effective means for touchdown. Since the payload is heavy, it was important that the design be able to withstand the stress of rough grass runways. The landing gear design permitted simple takeoff and landing procedures. Since the design can withstand the force of impact with the ground, the aircraft did not need a wide area to land in.

The tricycle design was also beneficial in take-off scenarios and enhancing vehicle speed. Overall, it was decided that this design permitted a much simpler takeoff and landing process which could be utilized at the competition. This feature is crucial to mission success because autonomous takeoff and landing are critical to the competition.

Another important feature of the plane that the Team considered was the wing profile. The wing profile provides no inherent aerodynamic self-righting ability which permits gimbal angle corrections for the autopilot. It also lets us tune the autopilot software between flights and calibrate the correction software properly for a smooth and successful flight. The extended tail section's flat-bottomed wing design and a thick airfoil with no dihedral contributes to the aircraft's maneuverability.

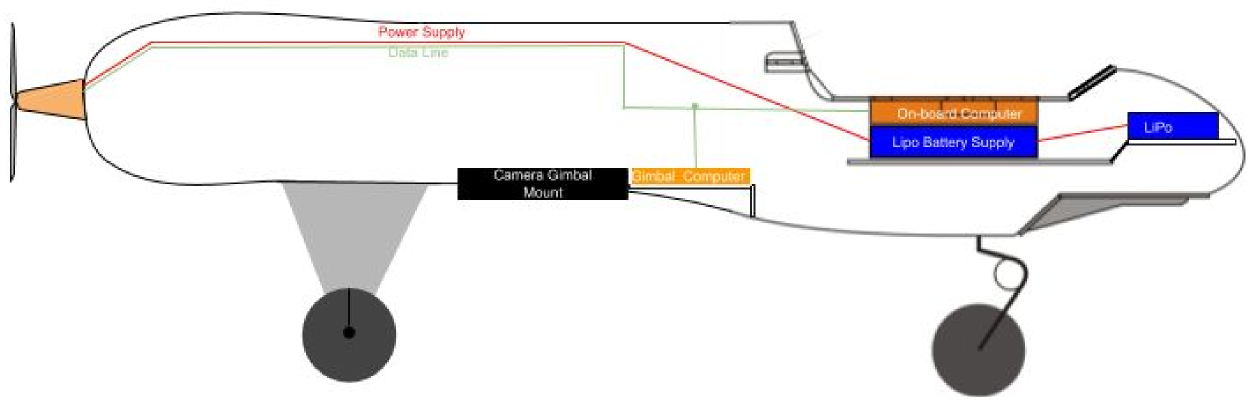

Figure 4: Specification of internal Hardware Placement (RMRC Anaconda)

Power Supply

The aircraft is powered through a combination of two of lithium polymer (LiPo) batteries. The main flight and payload subsystem batteries are comprised of four cell LiPo batteries containing a nominal capacity of 4000 milliampere-hour (mAh) and nominal voltage of 14.8 volts. We used two four cell LiPo batteries positioned in the nose of the aircraft to maintain a proper center of gravity of the aircraft (see the position of the LiPo Batteries displayed in Figure 4).

Propulsion

The primary source of propulsion is provided by a 800kv Tiger Brushless Outrunner AT3520-5 Motor combined with a 15x6 static propeller and Tiger Motor 80A Electronic Speed Controller. Conclusive testing demonstrated that the system has power suitable for sustaining 60 degree climbs.The aircraft has proven capible of maintaining a cruising speed of 30 meters per second (m/s), or 60 Knots-Indicated Airspeed (KIAS) and a normal cruising speed of 16 m/s (31 KIAS). During normal cruise, the power system draws only 6A. This gives us a theoretical flight time of 1 hour. Positioning the rotor in the back of the fuselage let us maximize propulsion of the entire aircraft (see Figure 4).

2.2 Autopilot

The aircraft uses a Pixhawk 2.1 with multiple sensors for autonomous navigation between waypoints. Through conducting an alternative analysis, the team concluded that continuing to use the Pixhawk Flight Management Unit (FMU) was ideal. As mentioned in the *Aircraft* section of the *System Design*, the self-righting ability of the aircraft provided gimbal angle corrections for the autopilot. Self-righting let the plane navigate its way to different waypoints. Table 3 shows a comparison between various autopilots. Each was considered for usage in the system. The three flight control systems provide the same level of mission capability and flight functionality. Each of the three systems are priced at under five-hundred US Dollars. The CNU UAS Team has a wealth of knowledge on all three options for the Autopilot system.

Hardware	APM 2.6 System	Pixhawk 2.1 System	Lisa/M V2.0 System
CPU	ATMega2560 Standard Arduino	STM32F427 ARM Cortex M4	STM32F105RCT6 ARM Cortex M3
RAM	8Kb - SDRAM	256Kb	64Kb
Flash	128Kb (124Kb Usable)	2Mb (1Mb Usable)	256 Kb
Redundancy	None	- STM32F103 failsafe coprocessor - Redundant power supply inputs - Backup mixing systems	Multiple R/C Receivers
Software	Ardupilot (Deprecated Hardware)	Ardupilot	Paparazzi
Cost	$65.00 (owned)	$350.00 (owned)	$250.00 (owned)

Table 3: Available Autopilot Systems for Implementation

Table 3 shows that the APM 2.6 hardware is deprecated and no longer supported by Ardupilot. As such, the APM 2.6 hardware is no longer a viable option for safe flight. While the Lisa/M 2.0 based on the paparazzi autopilot software is a capable autopilot, the system had several drawbacks noted by the team. The paparazzi software has very little documentation, insufficient online support, and is difficult to operate. The hardware required on board connections that are hard to build. At the same time, the connectors themselves are difficult to find. All of these drawbacks diminish the financial and operative feasibility of the Lisa/M V2.0. Diminishing the technical feasibility of the plane, is seen that the plane's voltage level, and the required voltage of the Lisa/M V2.0 don't match. While the input/output logic voltage of the Lisa device is 3.3-Volts, the on-board logic voltage is 5.0-Volts. This discrepancy creates a need for additional hardware to convert the logic voltage.

Due to the easy-to-use nature, the ease of configurability, and the ease of operation, the Pixhawk autopilot using ArduPlane firmware was seen as the best autopilot to suit our mission requirements. Coupled with a good level of documentation and it's open source nature, the Pixhawk 2.1 system is the most feasible and thus the best choice to achieve mission success at competition.

2.3 Obstacle Avoidance

The CNU UAS Team developed The *Dynamic Realtime Obstacle Pathing System* (DROPS) for *Obstacle Avoidance.* The DROPS program, through the use of Mavlink protocols, is able to autonomously adjust waypoint paths for the purpose of navigating around obstacles. DROPS calculates these paths with it's built-in algorithms.

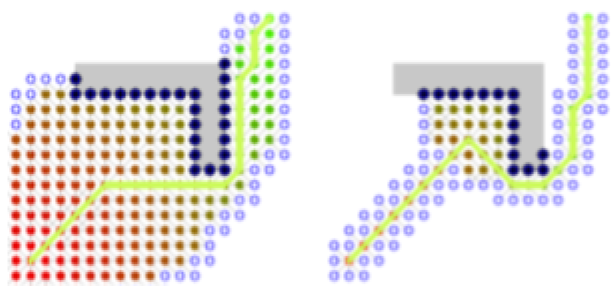

Figure 5: A (right) Weighted A* (left) Pathing around Obstacle*

DROPS calculates paths based on an A* exploring algorithm that finds the best fit to fit a path along around obstacles. Doing this requires a cost function to be attributed to the search algorithm. Three separate algorithms were debated in finding out which was optimal. Dijkstra's Algorithm, A* Search and Weighted A* Search. Dijkstra's Algorithm is not optimal in either the best fitted solution, or the fastest solved. While Weighted A* does not compute the most optimal solution, it is the fastest algorithm to compute a path around an object. Carnegie Mellon's Search Based Programming Library (SBPL) provides a framework in which to calculate paths around obstacles. The comparison in the number of possibilities explored and the path taken around an obstacle is displayed in Figure 5. This Figure is provided by the SBPL framework.

Despite the framework provided, the algorithm calculations are very computationally expensive. For these algorithms to be effective utilization of a ground station computer is needed to calculate these paths and send them to the plane. As a result the plane needs to be constantly updated with new pathing mechanisms for avoidance waypoints. All this causes latency and delay in the entire system. For this reason, the algorithm doesn't have a 100% success rating for all obstacles at the moment, and is best suited for stationary obstacles.

2.4 Image System

Camera

We selected the Canon G15 camera (Figure 6) as our camera due to several important features. An advantage of the G15 is that it is based on the DIGIC 5 processor, which has faster image saving and transferring speeds than most other Canon PowerShot processors. The camera also has a 12 megapixel image sensor. Due to the sensor and lense configuration we can identify features as small as ¾ inch at an altitude of 200 feet. The camera is able to shoot continuously, three frames per second, or one image every four seconds when triggered over USB. The resolution and rate of pictures are required to fully cover the ground when flying our cruising speed, approximately 32 knots at an altitude of 200 feet.

Figure 6: Canon PowerShot G15 Specification

An important decision criterion element was the camera's compatibility with the Canon Hacker Developer Kit (CHDK). The CHDK software framework facilitates the interface between the camera and payload computer. In particular, CHDK permits operations and control through an extended Picture Transfer Protocol (PTP). Though CHDK only supports point-and-shoot cameras, it was not considered disadvantageous as it corresponded to the team's goal of maintaining a minimal payload weight.

Gimbal

The gimbal mount to hold the camera is a composed model of a custom 2 axis gimbal which has been modeled in a 3D CAD System (Figure 7) to facilitate a 3D printed result. To keep the imaging camera parallel to the ground and improve image quality, we designed a two-axis gimbal mount actuated by two HITEC HS-85MG micro servos. The gimbal mount went through several iterations of design. The early iterations provided the rotational freedom desired, however, the volume and weight of the 3D printed result proved burdensome on the payload space. Later iterations improved tolerances, reduced size, and eliminated unnecessary weight. The final design of the gimbal is shown in Figure 7. with an affixed mounting block, two rotational mounts, a camera plate, and servos.

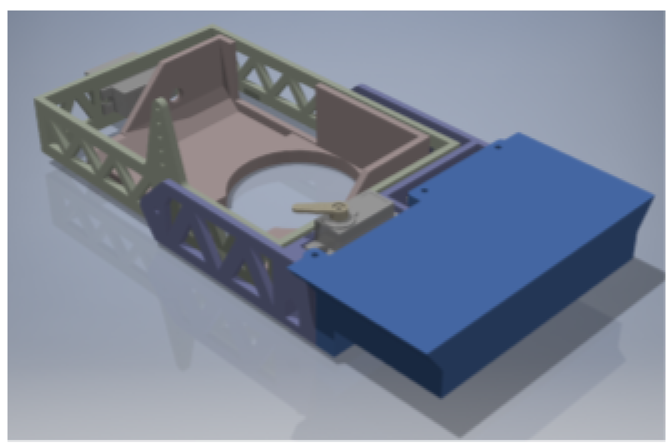
Figure 7: Custom 2 Axis Gimbal

Raspberry Pi

This year we upgraded the Raspberry Pi 3B from version 2. Maintaining a price of thirty-five US dollars, and size, 85x56mm. The performance of 3B is efficient and reliable with a high processing capability. The operating system Linux (Raspbian) is installed on the Raspberry Pi 3 due to the large support base, and open source origins. The on-board chip allows for control of several components with different modules. In order to control the gimbal system on the aircraft, MavProxy is used to level the gimbal, therefore stabilizing camera. The Raspberry Pi 3B offers greater flexibility in usage, a higher feature count out of the box, and a greater ease of use. The Raspberry Pi 3 Model B operates a 64-Bit ARM Cortex-A53 CPU 1.2GHz. The system on chip (SoC) of version is equipped with Broadcom BCM2837 quad-core 64-bit. The GPU of the Raspberry Pi has a Broadcom VideoCore IV with a frequency of 400MHz. Compared to a microcontroller, the Raspberry Pi offers more standard interfaces, including USB ports and Ethernet. Given the team's budget and size constraints, the Raspberry Pi 3B is an effective and efficient solution for use on the UAS.

2.5 Object Detection, Classification, Localization

The image recognition system allows for manual detection, localization, and classification of targets in images. The targets recognized by the imaging system are recognized through a module that utilizes a Convolutional Neural Network (CNN). Those images are uploaded to the competition server through an module embedded into the interoperability server. Images are processed by the machine vision module and then passed onto the interoperability buffer with all relevant information and uploaded to the competition server. In order to make sure that data is consistent with the correct information that pertains to the images, a line of data analysts are assigned to the flight line to make sure that the information is correct, and handled properly upon upload to the competition server. The CNN has a 99% accuracy rating that is associated with recognition (Figure 8).

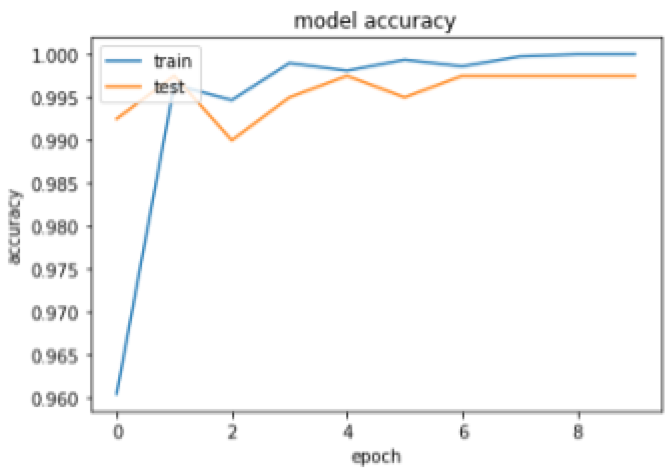

Figure 8: Model of Accuracy for CNN

The CNN operates all of the autonomous image recognition handled by the image recognition module. The mission module has an image upload mechanism built into the inter IMG-Mission API. Given that the images are recognized by the image recognition module and that there are users that are actively verifying the specification of each image on the flight line, The CNU UAS Team is confident in its capabilities with regards to uploading relevant targets to the competition server.

2.6 Communications

Radio Communications

Figure 9 provides a visual representation of how the communications are interconnected. The Spektrum DX6i controller held by the safety pilot operates via 2.4GHz RC. With the intent of full autonomous flight, the link will not be utilized under normal circumstances except for use of failsafe settings for the aircraft in flight. The on-board receiver we use is the Spektrum AR7610 that utilizes DSMX (Digital System Multiplexer), a digital control protocol. There are many protocols used to control the operation of the drone. The DSMX protocol is very robust against noise and interference and provides incredibly fast input-to-output response with an 11-millisecond frame rate. The controller is capable of supporting six channels to achieve complete control of the control surfaces. Due to the safety nature of this link, it operates fully independent of other systems and communications onboard.

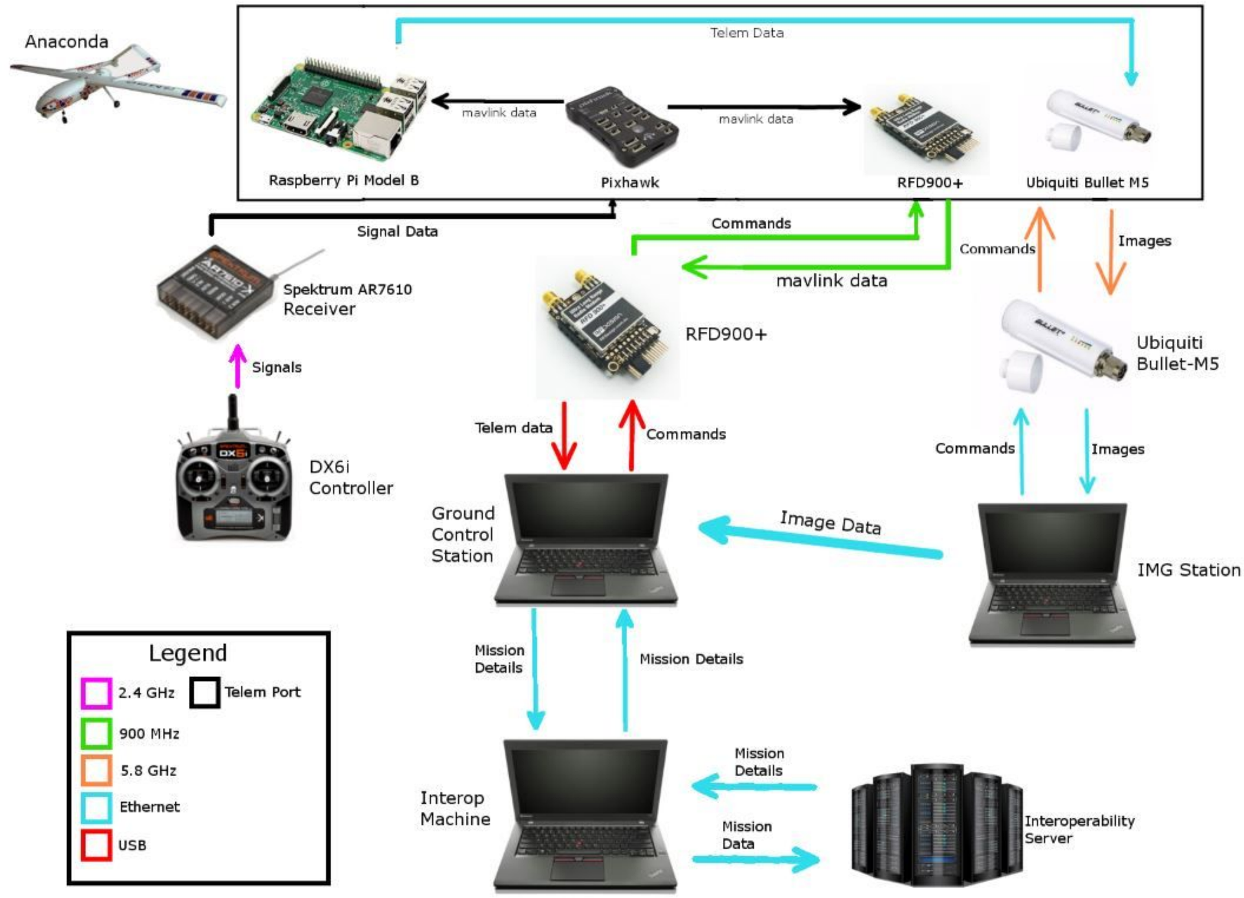

Figure 9: Communications System

Telemetry Communications

A 913MHz telemetry radio provides a real time data link for transmission of MavLink flight data to and from the mission control station. The data is utilized for the interoperability task and submission to the competition server. To achieve a greater transmission rate, we subsequently decreased transmission range. To account for this, we switched to using RFD900+ telemetry modules in place of the standard 3DR radio modules. The RFD900+ module is capable of 1.0 watt transmit power or 30dBm whereas the previous used modules are rated for 0.1 watts of power or 20dBm. The FCC (Federal Communications Commission) allows operations up to 4 watts EIRP (Effective Isotropic Radiated Power), which is 1.0 watt device output power plus 6dBi of gain. The reason higher EIRPs are acceptable is that the higher gain antennas are more directive, which reduces the possibility of Radio Frequency interference with other systems. With the enhanced setup, the radio utilizes dual polarization to prevent loss of communication. Due to enhanced features, the aircraft is capable of greater transmission rates than previous years and minimal packet loss.

Payload Communications System

A data link consisting of two Ubiquiti Bullet-M5's is utilized for communication to the payload systems onboard. The use of 5.8GHz instead of 2.4GHz ensures minimal interference with the RC safety link. Since the Ubiquiti Bullet-M5 supports standard wireless-networking standards, we are able to utilize the 802.11n protocols. The standard supports both 2.4GHz and 5 GHz frequency bands. 802.11n can reach up to speeds of ~300 Mbps and a range of 175 feet. 802.11n uses Modulation and Coding Scheme (MCS) to determine the data rate of a wireless connection using high-throughput orthogonal frequency division multiplexing (HT-OFDM). There are eight mandatory types of MCS (MCS0 - MCS7) that are used for 20MHz channels. The transmit rate can range from 19 dBm to 25 dBm depending on the MCS used. The 802.11n networking standard is chosen rather than the 802.11a for performance reasons. The Bullet is modified to have an omnidirectional whip antenna for high gain. The weight of the Ubiquiti Bullet-M5 is roughly 0.18kg which doesn't increase the overall carry weight. The Bullet has a max power consumption of 6 Watts and has a power rating of 24V, which equates to 0.4 A. With two four-cell LiPo batteries, the Bullet power consumption will not significantly impact flight time. The telemetry data is also routed through this data link to provide a greater transmission rate than achievable with the 900MHz radios. This ensures quick in flight retasking.

2.7 Ground Station

Mission Control

Out team chose the Mission Planner software as our autopilot grounds control station. Due to it being an open source software, it allows the team to modify it to fit the necessities of our system. We chose this software due to it already being implemented and tested with the Pixhawk. It also provides the pertinent flight information including the plane's speed, climb rate, altitude and waypoints through a map that indicates the position of the aircraft at any given time. The software encompasses a window that provides access to all the tuning constraints for effective tuning of the plane in flight.

Flight And Guidance

The flight control computer is the main control system in the GCS. It runs essential software such as MAVProxy, Mission Planner, and Interoperability. MAVProxy is a lightweight software used to manage inputs and outputs for telemetry data. The software allows the distribution of telemetry data across multiple systems. The link from the plane is fed into MAVProxy which then routes the data to appropriate destinations. Mission Planner is a full-featured ground station application for the ArduPilot open source autopilot project. Ground Control Station uses Mission Planner as a configuration utility for the autonomous drone. Having predefined functionality allows for waypoint mission planning and monitoring of the plane. Prior to being on the flight line, a base mission is created and all future tasks are added as waypoints; points received on the flight line are generated and added before flight. To permit maximization of versatility on flight day, four autonomous takeoff and landing patterns are made in advance. The patterns are made for each direction on the two runways.

Planning and Execution

This year, teams will be provided the search grid (as well as airdrop position, off-axis position, and the emergent object's last known position) at setup time via the Interoperability System. To complete the *Object Detection, Classification, and Localization* portion of the competition, we implemented functionality which auto-generated a set of waypoints covering the search area. Waypoints will be executed with Mission Planner as seen in Figure 10. While previous years permitted the generation of a search area beforehand, current regulations require it to be generated on the flight line.

We created a Mavlink Module to generate waypoints; The script was written using Python to conduct computational logic. The module takes in coordinates of the search grid, received from the Interoperability System, and generates a set of waypoints that efficiently covers the search area. The waypoints form a pattern which effectively moves back and forth down the search area grid, accounting for constraints of the aircraft. This efficient pattern minimizes how frequently the plane has to cover the same ground again.

Figure 10: Mission Planner

Interoperability

The system handling interoperability for this year includes a modular system comprised of several different components. The main components revolve around the front end, Systems interaction, and Competition server handling sections. The overall system uses several systems to pass around data and utilize it.

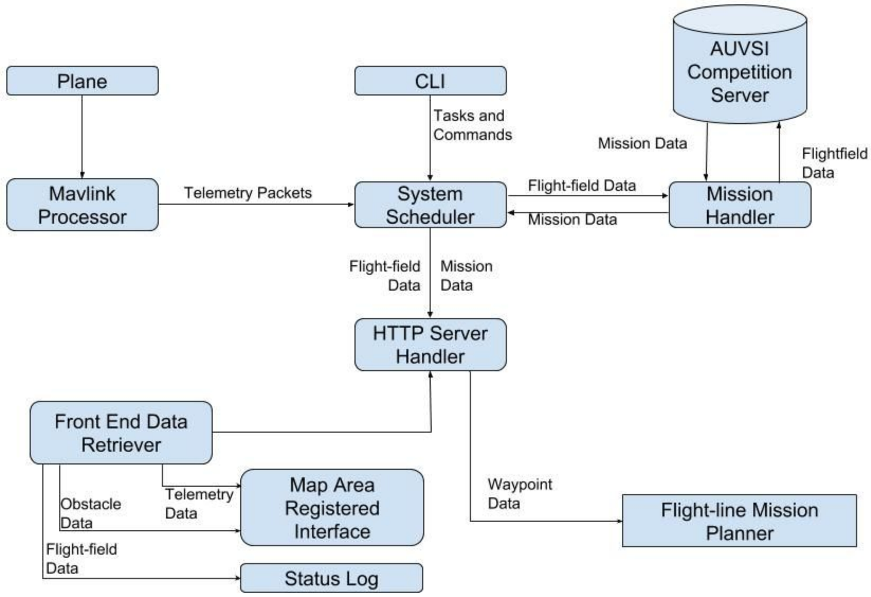

Figure 11: Interoperability Breakdown.

As Figure 11 demonstrates, the system uses several modules in order to handle data effectively. There are two people on the flightline operating the Ground Station-Plane Interface while the competition is in effect. This interface is mainly comprised of both the mission planner as well as the interoperability interface. The system starts out with an initialization call to the AUVSI Competition Server. The server sends mission data to the System scheduler for use by the entire ground station. To fulfill the mission requirement as well as the necessity of pulling mission information from the competition server, a data transfer system was implemented with the intended goal of storing the mission waypoints in the system and serving it on a large scale for the entire system. To fulfill requirements for both Obstacle Avoidance, as well as the Target Image System, functionalities implemented in the Mission Data System help to facilitate quick uploads of required images and their information. At the same time the front end system helps demonstrate the Obstacle Avoidance patterns, and helps verify that obstacles are being avoided properly by the plane.

The front end system includes safety precautions and features that allow the operators to know if the plane is functioning according to the team's expectations. Indicators present themselves to help the flight line determine if a mission abort, or an emergency landing is necessary. If an emergency landing is necessary, procedures will be carried out to ensure safe flight of the plane.

Image Processing & Submission

To have an autonomous image recognition solution requires a result that can be implemented on a limited hardware environment. This limited environment stems from the spatial and power limitations of an airborne drone. To construct an effective solution, the present challenges must be incorporated into the implementation for a useful and applicable outcome. This solution will be required to overcome these main design hazards, first recognizing and distinguishing between the complex differences between targets within images. Then implementing it on a small hardware device for airborne installation. The actual images that will be sent to identify from will be required to be 150x150 pixels, but the exact size of the target's pixels can vary. This variance in pixel size stems directly from the altitude the aircraft will acquire the images from, with the larger the altitude, the smaller the overall pixel size of the target will be. The average size of these targets within the 150x150 pixel images is about 50x50 pixels. Differences in input can result from changes of altitude, camera angle change from the aircraft turning, as well as flight conditions such as weather.

The present challenges regarding the development of an effective image recognition solution require a system that can handle varying inputs and output probable classifications, as discussed before. Considering these requirements, an Artificial Intelligence (AI) utilizing machine learning was selected as the best approach to the challenge. Three methods were analyzed and selected from. A deep neural network has become the standard for classification based problems where inputs are dynamic and outputs are more probable based. A convolutional neural network (CNN) implements many of the same features of a normal deep neural network but specializes in image recognition solutions. These features can include curves, edges, colors and locality among these different features. The last design alternative is a derivative of the CNN structure discussed earlier. A CNN with transfer learning is a CNN that implements the learned numeric weights found by another CNN. The CNN has been constructed using the VGG16 pre-trained neural network as a base for feature detection. This model was trained on the Image-Net collection of about 1.2 million samples, and is recognized as a one of the most successful image recognition bases. This model has been adapted to accept 150x150 pixel images while still utilizing the pre-trained weights provided with the VGG16 model. The data required to train the CNN was generated using augmentations on the samples collected in flight of the targets. Due to the difficulty in constructing and acquiring data points, the shapes have been prioritized for learning at the moment, with future plans to implement color detection. All the shapes chosen have been physically constructed and sample images have been collected containing these targets for the data point set. After ten training cycles, the CNN was able to distinguish, with 99% accuracy, the differences between all the shapes.

2.8 Cyber Security

Cyberattacks are a real and diverse threat. In the world of influential systems, such as UASs, they have the budding ability to result in serious harm to life and technology. Thus, the security of our team's systems is always considered in the design process.

GPS spoofing or jamming is a tangible threat. The costs of a signal takeover is grave because of what UAS technology implies. A loss of control of the UAS due to lack of knowledge by the controllers can lead to a damaged aircraft or even one that gets stolen. As a college organization, our UAS technology is built with consumer grade hardware. Due to this, we cannot defend against attacks such as spoofing. Because the competition is within a closed environment and the probability of such an event happening is low, we will be mindful of the situation but not hostile towards this type of attack. If such a threat does occur, our UAS will follow necessary procedures that imply it returns to the ground station.

3. Safety, Risks, and Mitigations

3.1 Developmental Risks and Mitigations

The development of drones and the hardware that they require is not always harmless. Human error is a qualifying factor that all teams need to consider and take into account. Due to this, we require all of the team members to follow the following rules when actively working in the lab.

Risk	Mitigation Strategy
LiPo battery fire	All members on the team are trained on the proper handling of LiPo batteries and the dangers they pose. Each member must pass a practical test before being able to handle charging of the batteries. For each plane test, a fire extinguisher is on hand. No longer operational batteries are properly marked and disposed of.
Injury due to power tool usage	All members must go through a workshop on tool handling annually.
Injury from plane hardware	All members are taught the purposes as well as the expectations when handling drone parts and equipment.

Table 4: Developmental Risks and Mitigations

3.2 Mission Risks and Mitigations

During missions, human error is one of the most prevalent risks to discuss. While the Anaconda might be at optimal performance and accuracy, there are still many risks that come out of an unmanned device. We recognize that every system has a flaw. Because of this, we pose the following actions taken to mitigate the most dangerous cases that may occur during a mission.

Risk	Mitigation Strategy
Abrupt aircraft behavior due to obstacle avoidance	The safety pilot maintains full authority to override autonomous aircraft control. Stationary obstacle avoidances must be approved by the GCS before sending the information back to the plane.
Loss of control over the aircraft	The mission commander will verify control is not due to autonomous flight mode. Safety pilot will then attempt to take control. Ground station will monitor flight path and a pre-designated spotter will maintain visual with arm extended in direction of aircraft to reference point. If no pilot control may be obtained, the pilot will relay information to the mission commander. The fail safes are designed so that if the pilot loses RC connection to the vehicle, it will return to a pre-set home location and loiter. If no control is regained, the autopilot will terminate the flight.

Table 5: Mission Risks and Mitigations