Final Report

The Red October

Matthew James
Andrew Hendricks
Michael Boren
Kevin Perkins
Samuel Farnsworth
Thomas Townsend
Tyler Hansen

Executive Summary

The Department of Defense has given several universities the assignment to develop simple, inexpensive and non-traditional methods for gathering surveillance of a known source as well as methods to prevent this surveillance. BYU's engineers have been selected to participate in the design and implementation of possible solutions that satisfy these requirements.

The Red team is tasked with providing reconnaissance information related to militarily significant items including measurement and signature intelligence, imagery intelligence, and technical intelligence. Two systems, mission and flight, have been developed and integrated to successfully achieve our objectives.

The mission system was designed to provide two types of intelligence, passive Wi-Fi packet collection and visual data collection. To acquire this information, a single board computer (Raspberry Pi) is used for it's ease of use and lightweight structure. A wireless network detection program called Kismet is run on the Raspberry Pi to passively collect Wi-Fi packets and report signal strength in real time to the base station. To gather visual data, a pipeline-based multimedia framework called GStreamer is used. Gstreamer allows visual data to be transferred from the Raspberry Pi camera to the base station. A 4G dongle will be used to transfer data from the Raspberry Pi to the base station via cell phone towers. The Raspberry Pi camera is mounted on a gimbal allowing for a dynamic camera angle. Power for the mission system will be provided by a 2200 mAh USB battery pack.

The flight system consists of a hexacopter, an advanced autopilot system (Pixhawk) and wireless communication channels including a telemetry radio, an RC controller, and a GPS module. The telemetry radio, GPS module and autopilot system will be used to provide waypoints to the hexacopter for autonomous flight. Commands from the base station will be received via telemetry radio and the 4G dongle. The RC controller will be used for manual control of the hexacopter for safety precautions. Power for the flight system comes from a 6000 mAh lithium-polymer battery.

An Ubuntu 14.04 LTS operating system is used to receive information from the mission and flight system. This base station sends commands to the gimbal to change the angle of the camera, and is used for other functionality of the total system. Information coming from Kismet and GStreamer will be viewed and processed here.

With the integration of these systems, the project objectives were completed, and effective solutions were provided to the Department of Defense.

Table Of Contents

- Executive Summary
- Table Of Contents
- Table Of Figures
- Index Of Tables
- Introduction
  - Project Introduction
  - Background
  - Standards
  - Important Specifications
  - Critical Assumptions
  - Contest Location and Obstacles
- Block Diagram of the Final Design
- Project Breakdown
  - Mission Systems
    - Raspberry Pi Model B+
    - Video Stream Communications
    - Wi-Fi Snooping
    - GPS Integration
    - 4G Communications Link
    - PKCell USB Battery
  - Flight Systems
    - Unmanned Aerial Vehicle: 3DR Y6
    - PixHawk
    - Communications
    - Battery
- Base Station
- Cost Summary
  - Budget
  - Purchased Items
  - Future purchases
- Problems Encountered and Lessons Learned
- System Performance and Tests
- Future Tasks
  - Implement Software Defined Radio
  - Automated Flight
  - Connect Raspberry Pi and Pixhawk via Mavlink
  - Gyro-Stabilized Gimbal
  - Tests/Simulations
- Conclusion
- APPENDIX A - Functional Specifications Document

- Project Requirements
- Project Specifications
- Linking Project Specifications and Needs
- Conclusion

APPENDIX B - Concept Generation and Selection
- Body of Facts
  - Specifications
  - Known Facts
  - Critical Assumptions
- Design
- Concept Generation for Microcomputer
  - Description of the Alternatives
  - Decision Criteria and Weight
  - Scoring
  - Review of the Results
- Concept Generation for Sensor Types
  - Description of the Alternatives
  - Decision Criteria and Weight
  - Scoring
  - Review of the Results
- Conclusions

APPENDIX C - Project Schedule

APPENDIX D - Tutorials/Instructional Documents

Table Of Figures

Figure 1: Contest Location……………………………………………………………………8

Figure 2: System Diagram……………………………………………………………………9

Figure 3: Types of Intelligence Gathering…………………………………………………10

Figure 4: Raspberry Pi………………………………………………………………………10

Figure 5: USB Battery Pack…………………………………………………………………11

Figure 6: Y6 and Pixhawk Autopilot…………………………………………………………12

Figure 7: Y6 Battery……………………………………………………………………………12

Figure 8: Wi-Fi Heat Map……………………………………………………………………16

Figure 9: PWM board to Raspberry PI pin connections……………………………………28

Index Of Tables

Table 1 : Items Purchased……………………………………………..……………...…..14

Table 2: Customer Requirements and Priorities……………….....................................19

Table 3: Metrics of Project Requirements……………………………….………..…...…....20

Table 4: Correlation between interpreted customer needs and specified metrics..........21

Table 5: Concept Scores: Microcomputer…………………………….………………….23

Table 6: Concept Scores: Sensors……………………………………..…...………...24

Introduction

Project Introduction

This document describes the final report for the Fall 2014 Red October UAV Surveillance Project. Our team consists of Matt James (team leader), Thomas Townsend, Michael Boren, Andrew Hendricks, Samuel Farnsworth, Kevin Perkins, and Tyler Hansen. A project description as well as project requirements, project specifications, financial specifications and total system design will be discussed in this document.

Background

The Department of Defense is concerned with the growing popularity of UAVs and the potential threat they pose to national security. They have selected several universities to participate in a project that will help them better understand and plan for these potential threats in the future. Two teams were created for this project, one focusing on surveillance and gathering information on an unknown object using a multi-rotor aircraft (Red team), and the other focusing on counter measures to impede the other team from accomplishing their task (Blue team). Each team will have the opportunity to display their solutions during a demonstration that will be held in April at the United States Air Force Academy. The object which the Red team is surveillancing will be referred to as the "Yugo II" throughout this document.

Standards

Several standards were set by the Department of Defense to determine if project specifications were met. Red team's primary tasks and restrictions are to:

"Provide reconnaissance information related to militarily significant items including measurement and signature intelligence (MASINT), imagery intelligence (IMINT), and technical intelligence (TECHINT). All Red Team UAS must pass safety tests performed by USAFA personnel and adhere to the FAA and FCC rules (see Section 2.7 and Appendices for rules and regulations) before they are allowed to participate in the flight demonstration. Should safety of flight, FAA or FCC restrictions preclude any of the academic teams from operating their UAVs at USAFA, the USAFA staff will operate previously approved USAFA UAVs in support of the demonstration."

-- Preliminary Control Document from DoD

Success of the project will be determined by the following objectives:

- Detection: Demonstrate an ability to search over the protected area without being detected
- Collect intelligence of target: Significant and useful target intelligence is collected
- Communication: Timeliness of reporting, real-time versus post-flight provision of target intelligence to the Red Team personnel

Important Specifications

The Department of Defense has laid out several specifications that must be observed by the Red team during the final demonstration.

- One multi-rotor in the air at a time
- Solution Demonstration will occur during the day
- The UAS must transmit location, health, and sensor status to a Base Station
- An RC pilot must be within 500 ft. of the UAS at all times
- The Red team will have 5 minutes to enter enemy airspace and 30 minutes to gather intelligence.

These key specifications played an important role in guiding our team to our final solution. For example, since we know that the demonstration will take place during the day, we spent less time considering and implementing solutions that would work better at night. Similarly, due to the 30 minutes of mission time to gather intelligence on the Yugo II, one of our main goals was to make the payload on our UAS system as small as possible. This approach will allow for the longest possible flight time, and in turn, the highest probability of mission success.

Critical Assumptions

While implementing our UAS system, we operated on two critical assumptions concerning the Yugo II and the demonstration area at the United States Air Force Academy.

- The Yugo II will be emitting in the 2.4 GHz ISM band
- There will be sufficient 4G cell coverage demonstration area

If these assumptions were proven to be incorrect, our entire solution would have to be reinvented.

Contest Location and Obstacles

The location for the final demonstration is shown on the right. We know that the target will reside somewhere in the blue area, however we have to navigate there from inside the red circle. To avoid detection, we will want to fly low. Manual control is needed to avoid obstacles. To succeed, we need to fly low enough to be undetected, high enough to avoid trees, but no higher than 500 feet to comply with FCC regulations.

Figure 1: Contest Location

Block Diagram of the Final Design

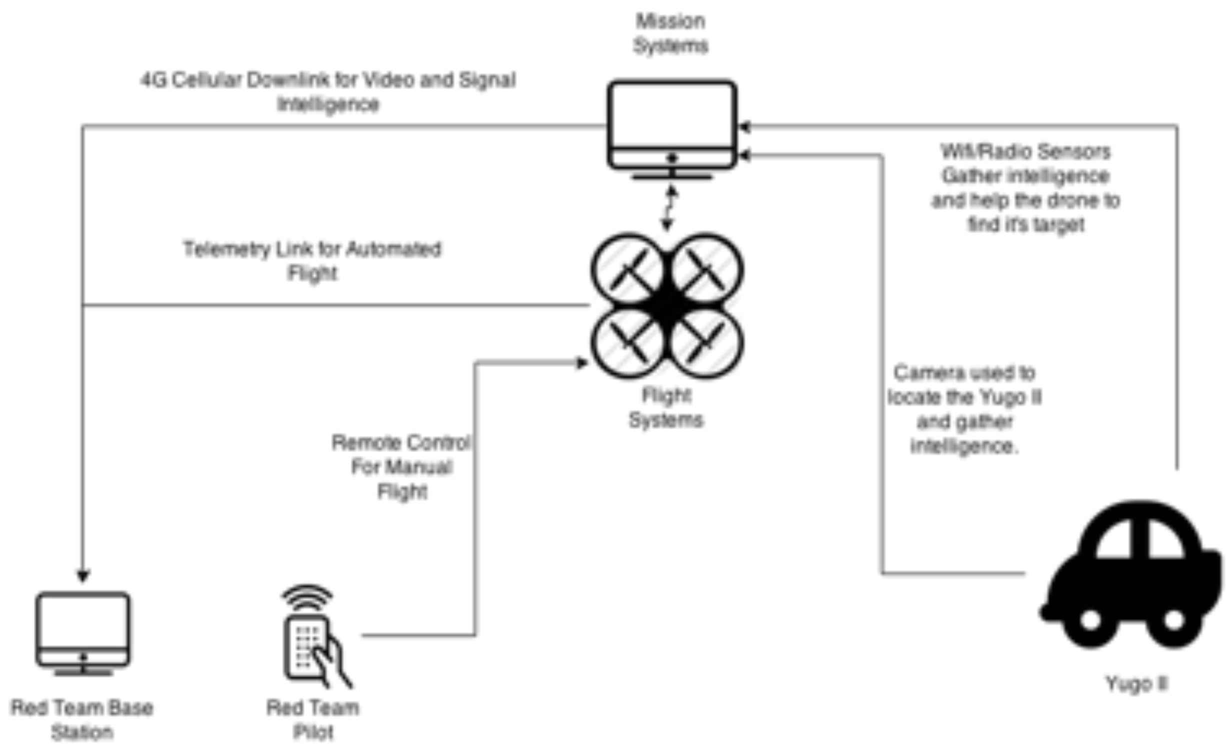

Figure 2: System Diagram

Project Breakdown

Mission Systems

The purpose of the Mission Systems is to gather intelligence. The Department of Defense is specifically looking for Imagery, Measurement and Signal, and Technical Intelligence. In more general terms, this may comprise pictures and videos of, RF emissions from, and intent of the Yugo II. Below is a diagram of the types of intelligence that we feel would be most useful.

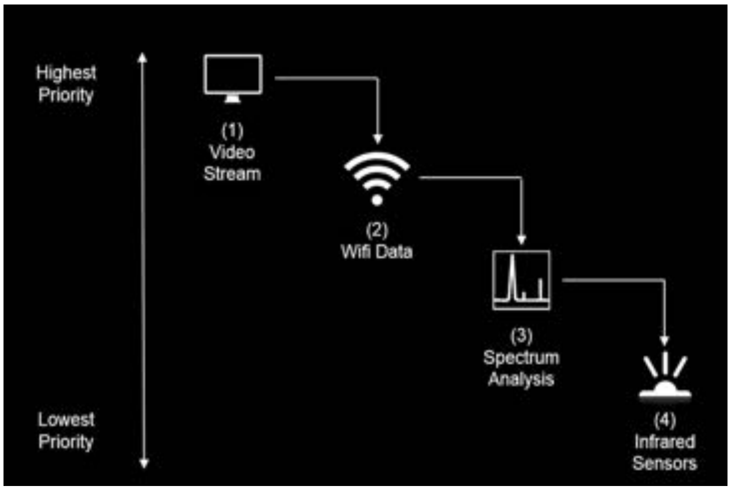

Figure 3: Types of Intelligence Gathering

As can be seen, our primary focus will be on gathering pictures and video of the target. Next, we will attempt to “snoop” Wi-Fi packets and other frequency signals being emitted from the target. The final method of intelligence gathering, infrared sensors, will primarily be used for night surveillance. However, since the demonstration will occur during the day, it is the lowest priority of the four. The mission will be considered a success if any one of these methods yield useful intelligence.

The Mission Systems will be controlled primarily by a Raspberry Pi (model B+) Microcomputer. This is an ARM based, credit card sized, computer that runs a compact version of debian linux. A further description of the Raspberry Pi, along with a list of software and hardware components that will interface with it, is detailed below.

Raspberry Pi Model B+

- 700 MHz Low Power Broadcom SOC
- 512 MB Ram
- VideoCore video processing, including hardware HD encoding/decoding
- 40 GPIO Pins

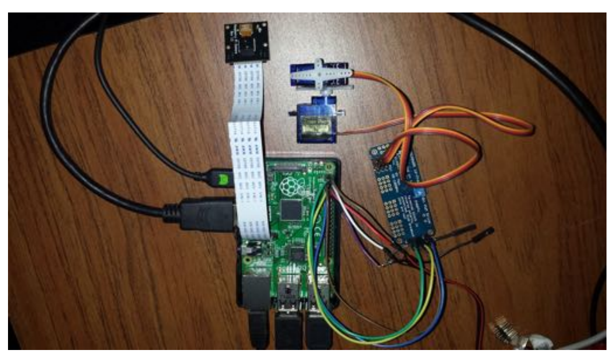

Figure 4: Raspberry PI

- Camera Slot
- HDMI output
- 4 USB Ports

Video Stream Communications

Real-time video stream at customisable resolution and frame rates is taken from a camera mounted to a 2 axis gimbal. Constant video stream from RPi to base station will be achieved using Gstreamer, an open source software. The gimbal consists of a PWM board and 2 servos. Communication between the PWM board and RPi is through a standard I2C protocol. Real-time commands to the gimbal can be sent from the base station with keystrokes ‘a’, ‘s’, ‘d’ and ‘w’. High resolution images are taken and stored with the keystroke ‘p’.

Wi-Fi Snooping

Using Kismet, another open source software, and an off-the-shelf Wi-Fi adapter, we can use the Raspberry Pi to detect Wi-Fi signals. These Wi-Fi signals are recorded and can be used as a form of Signal Intelligence. The relative signal strength of these Wi-Fi signals can also be used to help form a map that will be used to assist in locating our target.

GPS Integration

Using the built in linux GPS daemon, we are able to use the Ublox GPS that came with our drone and feed it directly into the Raspberry Pi, this allows us to record the latitude and longitude at which we are receiving our signal intelligence. In the case of Wi-Fi Snooping, this will allow us to construct a heat map that shows relative signal strength vs GPS location.

4G Communications Link

Although the Pixhawk has a small telemetry radio, the amount of intelligence we are gathering necessitates another means of communication with the ground station. We will be using a 4G LTE modem to send video and Wi-Fi data to the base station. We hope this high speed link will enable us to send back high quality video and images, as well as allowing us to control the camera gimbal, and receive other intelligence we may need to gather.

PKCell USB Battery

A Small USB battery will be used to power both the Raspberry Pi and the servos. A USB splitter will be used to connect both the Pi and the PWM board to the battery. The battery can last almost 2 hours, much longer than our drone will be able to fly.

- 2200mAh Capacity
- Output: 5V @ 1000mA

Figure 5: USB battery pack

Flight Systems

Unmanned Aerial Vehicle: 3DR Y6

The 3DR Y6 provides the mobile platform for all intelligence gathering systems. The Y6 configuration provides a good thrust to weight ratio, and hence good carrying capacity. The three arms also give a wide field of view for cameras and other sensors.

PixHawk

The PixHawk autopilot is the heart of the flight systems. It receives data from GPS, compass, and onboard accelerometers and gyros, and controls the drone's motors. The data received is used to stabilize flight when under RC control and to follow given waypoints in autonomous flight. Flight data is also transmitted to the base station over telemetry radio.

Figure 6: Y6 and Pixhawk autopilot

Communications

Flight systems make use of two communication links. Manual control is maintained through the RC link between a Spektrum DX6i transmitter and an onboard DSM receiver. Waypoints are passed to the PixHawk autopilot and flight data is received at the base station across a 915 MHz telemetry radio link.

Battery

Flight Systems are powered from a single Lithium-Polymer battery. The trade-offs of including a second battery are under consideration.

Figure 7: Y6 Battery

Specifications:

- 6000mAh
- 14.8 V, 4S
- 680 g

Base Station

Critical to the operation of both mission systems and flight systems is the base station. The base station will receive and process telemetry data from the Pixhawk, a video stream from the Raspberry Pi, and a Wi-Fi snooping stream from the Pi. In addition, it will also transmit waypoints to the Pixhawk, control the camera gimbal angles, and be the failsafe communication system for the Remote Control link. It will run on an Ubuntu 14.04 LTS operating system. A description of the software powered by the base station is listed below.

Base Station Software

- APM Planner - Used to send/receive telemetry to/from the drone.
- GStreamer - Used to receive the video stream from the drone.
- Kismet Server - Used to process the Wi-Fi stream from the drone.
- GoogleEarth - Used to view signal heat maps.
- Heat map scripts - Will be used to construct Wi-Fi/Radio signal strength maps.
- SSH - used to connect to and execute code remotely on the Raspberry Pi

Cost Summary

Budget

A generous budget of $15,000 is available for the development and prototyping costs of both the red and blue team of a single university. The final working solution for both teams may not exceed $15,000, and no single team's solution may exceed $10,000. Items such as student laptops do not count towards the final solution.

Purchased Items

Item	Cost
2 DIY6 Kits with Telemetry Radios	$1,437.35
2 Aero Battery Packs (6000 mAh)	$156.65
2 Spektrum RC receivers	$38.16
ODroid U3 with Wi-Fi Module and USB Camera	$115.00
Wi-Fi Adapter(1), T-Mobile Rocket, USB Hub	$80.77
Raspberry Pi B+ with Normal and NoIR Cameras	$109.32
Spektrum RC Transmitter with Free Receiver	$129.99
2 Test Quadcopters	$75.47
Wi-Fi Adapter(2), DSMX Remote Receiver	$53.15
Wi-Fi Adapter (that works), price is estimate before shipping	$22.00
Accelerometer, Gyro, PWM (price before shipping)	$35.00
5 pair of spare props	$29.85
Total	**$2,282.71**

Table 1: Items Purchased

Future purchases

The greater part of expected expenses are displayed above. The main future expense will be a cell phone data plan to allow testing of the 4G link for mission systems communication. We will also need another 4G modem for the Base Station. Furthermore, we also expect future expenses for redundant and replacement parts as tests are performed .

Problems Encountered and Lessons Learned

Working on this project was a challenge for a lot of different reasons. Having such a freeform project has allowed each member of our team to be creative and come up with their own solutions. The downside to this is that we had a lot of good ideas we had to turn down simply because of time constraints.

One thing we all learned was to overplan the amount of time we need to finish things. Even the things we thought would take no time at all, like getting the drone to fly or binding the remote control, ended up taking weeks or months. This didn't necessarily set us back--we had other systems to work on as well--but if we had to do it again we would probably overestimate the time that it takes, rather than assuming that everything will work exactly as it should right out of the box.

Another lesson learned with this project is that engineering consists of not only design, but also taking the work of others and being able to integrate pieces of other projects into a working system. We were able to do quite a bit of research and find other ideas and designs that we could use to create a system that met our specifications. We learned that it is important to document your work, when possible, so that others can benefit from your findings.

Even when we take off the shelf parts that are supposed to work together they often don't. We learned that just as important as being able to design and build a working system, is the ability to troubleshoot it. We had software errors, hardware errors, wiring errors, and installation dependency errors. A lot of these errors were tricky to track down and hard to fix. Just because someone else got a setup to work a certain way, it doesn't mean you will. We didn't know that we would have to go through 3 different Wi-Fi adapters until we found one that worked with Kismet. All of these Wi-Fi adapters had drivers that should have worked, but only one of them ended up with the capability we were looking for.

System Performance and Tests

Over the course of the semester we performed tests to ensure that the mission systems would function correctly. One of our first tests involved streaming video from the Raspberry Pi to the base station to ensure that latency was low and the amount of dropped packets was few. The first programs we used was netcat. When we tested the streaming using netcat, latency was on the order of 3-4 seconds. By switching to GStreamer software and tweaking the frame-rate, resolution, encoding type, and other parameters, we were able to achieve a video stream with a latency of less than ½ of a second.

Next we tested the capture of Wi-Fi packets and the creation of a heat map associated with this information.This test was a success as can be seen by the figure to the right, which shows the Wi-Fi signal strengths on a simple walking path at BYU.

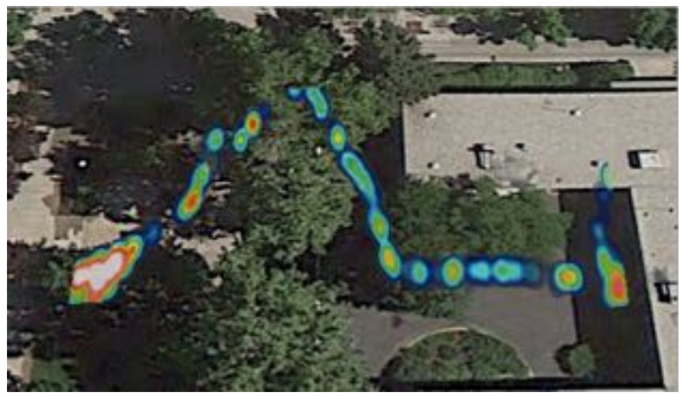

Figure 8: Wi-Fi Heat Map

The next issue was related to the processing power of the Pi. We were unsure at first that there would be enough power to do everything we needed. But, once we connected everything to the Pi, we were able to run all the required sensors, video, gps, and Wi-FI, as well as stream video back with no problems. We still have the ability to overclock the Pi by an extra 300MHz if performance becomes an issue.

Future Tasks

Implement Software Defined Radio

- While Video and Wi-Fi are great methods of gathering intelligence, we feel that it may make the solution more robust to include a Software Radio that can detect if the Yugo II is emitting frequencies besides those in the Wi-Fi Bands. To implement this, we would simply find an inexpensive radio, program a spectrum analyzer in a software package called GNU Radio and look for signals.

Automated Flight

- Due to complications, we were only able to get the Y6 flying with an RC pilot this semester. The next step is to automate the flying process by sending waypoints to the Pixhawk.

Connect Raspberry Pi and Pixhawk via Mavlink

1. Send GPS data from the Pixhawk to the Pi to avoid the need for two GPS units
2. Read the attitude log from the Pixhawk to gyro stabilize the camera.
3. Send waypoints from the RPi to the Pixhawk as a failsafe in case the telemetry link is compromised, or to limit all communication with UAV to 4G bands in order to avoid detection.

Gyro-Stabilized Gimbal

1. Read in data from accelerometer, gyroscope, compass.
2. Convert raw data to angle.
3. Alternatively, the Pi can read in telemetry data from the Pixhawk as well.
4. Use a new keystroke to toggle camera lock i.e., when camera lock is on, any change in roll or pitch on the Y6 is countered by an opposite change in roll or pitch on the servos. (See Appendix D for more information)

Tests/Simulations

In order to prepare for the demonstration at the Air Force Academy in April, several tests and simulations still need to be performed. These tests will serve a variety of purposes including: benchmarking critical design criteria, evaluating how well the solution performs overall, and evaluating how well the system performs while being targeted by the blue team. A list of questions that tests will need to answer can be seen below.

- What is the maximum range that can be achieved with the telemetry radio?
- What is the maximum range of the Remote Control?
- What is the maximum flight time of the Y6 multi-rotor with all hardware attached?
- What changes must be made to allow use of the 4G link to stream data?
- What is the latency of 4G communications?
- How does the drone react when it enters a failsafe mode? (i.e. What happens if the drone loses GPS/RC in flight?)

Conclusion

Through the integration of the flight system, mission system and base station, we have designed and implemented an inexpensive, simple solution to gain intelligence of a known source. All important specifications established by the Department of Defense and those participating were followed. Additions to each subsystem can be made for improved functionality to the total system. Our success in this project proves that a small group with enough resources could design a UAV that could capture military intelligence.

While we did not have enough time to completely finish all aspects of the project, we have advanced the systems to the point that they all have basic functionality. The challenge for next semester will be to integrate the systems together and test and fine-tune the devices until everything works properly. The biggest upcoming challenges will be communication with the ground station from the mission systems, and autonomous flight for the flight systems.

APPENDIX A - Functional Specifications Document

Project Requirements

The Department of Defense, our primary customer, stated that, “The objective of Perseus III is to enhance the understanding of the capabilities and limitations of inexpensive UAS and how they may place DoD personnel, equipment, and infrastructure at risk. In addition, it will generate information, through non-traditional lenses, providing valuable provide data points to assist DoD and partner organizations to appropriately address the low-cost “hobby” UAS in the future”. We are to design, build, and demonstrate one Unmanned Air System (UAS) solution, but can propose other solutions that we may not be able to fly.

To encourage creativity and out of the box solutions, product requirements were, to an extent, kept deliberately vague. However, from the overarching goal and details in the documentation provided, we have extracted the following project requirements:

#	Customer Requirement	Priority
1	Provide reconnaissance information related to militarily significant items	high
2	UAS must adhere to FAA and FCC rules and regulations	high
3	UAS should be functional in noisy, low-visibility conditions in day or night	low
4	The UAS shall not fly above 500 ft	high
5	UAS should be able to search over a protected area without detection	low
6	UAS should report intelligence in a timely fashion (in real-time if possible)	medium
7	Only one UAS may be airborne at a time	low
8	Undergraduate students must be responsible for 100% of solution	high
9	UAS must be able to be shut down remotely on command	high
10	Solution development process must be well documented	high
11	Solution should be low cost, off-the-shelf solutions are encouraged	medium
12	UAS should be a multi-rotor platform	low
13	UAS autonomy is encouraged	medium
14	UAS must be able to enter, search and exit the target area for recovery	high

Table 2: Customer Requirements and Priorities

Project Specifications

The objective of the assignment is to develop and showcase a multi rotor aircraft that can autonomously maneuver, find an object of interest and relay images of the object back to base. Given the competitive nature of the assignment, product specifications will be chosen to perform approximately the same as other teams in some aspects of the assignment while outperforming the competition in other areas.

Flight specifications will include the following:

- flight time (battery life)
- carry weight
- top speed
- number of rotors

Most of these specifications are consistent across the online retail market and thus will not constitute distinguishing features of our final product.

Mission directed specifications include the following:

- radius of communication
- camera resolution and frame rate
- aircraft size and RF emissions

In this category we have more freedom to choose allowing us to set ourselves apart. We will seek a very large range of communication, a high resolution camera, and we will keep the aircraft size and RF emissions to a minimum to avoid detection where possible.

Linking Project Specifications and Needs

We can measure each of the needs and specifications detailed above by the use of metrics. Each metric found in table 2 provides a benchmark for every aspect of our project that is critical for mission success. Table 3 shows exactly how each metric aligns with the interpreted customer needs.

#	Metric	Min/Max Value	Ideal Value
1	Total cost of parts	< 10,000 USD	< 5,000 USD
2	Flight time	15 min	20 min
3	Hardware platform	4 rotors	6 rotors
4	Overall size of system	< .5 m^2	< .3 m^2
5	Carry weight	2 lbs	5 lbs

6	Top speed	15 km/hr	20 km/hr
7	Total power consumption	< 50 W	< 40 W
8	Communication type	Wi-Fi	3G/4G
9	Radius of Communication	300 m	1 km
10	Camera Resolution	720p	1080p
11	Camera Frame Rate	20 frames/sec	30 frames/sec
12	GPS autopilot error radius	10 meters	5 meters

Table 3: Metrics of project requirements

	1	2	3	4	5	6	7	8	9	10	11	12	13
Functional flying platform		X	X	X	X	X	X						
Picture taking and real-time transmission ability							X	X	X	X	X		
Autonomous flight, GPS control													X
Not easily detected				X								X	
Low cost - “hobby” UAS	X												

Table 4: Correlation between interpreted customer needs and specified metrics (numbers correspond to the metrics found in table 2)

Conclusion

Our primary objective for this project is to design a UAS that can conduct surveillance effectively. The specifications that we have selected were chosen based on what we feel will be able to fill this mission, and what is currently available on the market. These specifications will allow us to meet the requirements of the customer, and allow us room for improvement should the project requirements change or be clarified in the future. Our current timeline should allow us to have a flying, semi-autonomous platform by the end of the semester. Another group of students will finalize the development and finish the project by the competition date in April of next year.

APPENDIX B - Concept Generation and Selection

Body of Facts

Specifications

Build a simple, low cost Unmanned Aerial System.

Ability to fly into an area, take pictures or take sensor readings of an object and return.

Ability to fly and take readings during day or night.

Ability to navigate in a one nautical mile radius.

Total cost of final solution cannot exceed $10,000.

Known Facts

Our market/customer is the DoD.

We must submit any hardware to be approved for flight by the fifteenth of November.

Intelligence that we submit can consist of measurements, technical intelligence and pictures.

Best results will consist of real-time transmission of data.

We can only operate below 500 feet.

We must be in line of sight of the UAV at all times during the demo.

We must have a fail-safe mode for the aircraft.

We can't talk to teams from other schools.

The demonstration will happen during the day.

We (the undergraduates) must contribute 100% of the solution.

Non flight ideas can be demonstrated by paper study or ground demo.

Timeline is two semesters, field demo will take place in April 2015.

Critical Assumptions

We can use any multi-rotor aircraft.

All our off the shelf parts will interface.

Our battery will last longer than five minutes.

No time limit for the demo.

We will not be given GPS coordinates.

We know the bounds the object will be inside of.

Design

Our project design is centered around our multi-copter platform pictured below. The multi-copter is flown either autonomously by an on board computer, or manually via an RC remote. While flying is obviously of paramount importance to our goal, many of these decisions have already been made. We will use the Pixhawk autopilot to control our autonomous flight. We have chosen to use a Y6 multicopter as our aircraft. Second to our goal of flight is the array of sensors that we can use to collect intelligence. To facilitate this, we will use a microcomputer, such as a Raspberry Pi or Odroid, coupled with various sensor types. This computer, along with it’s sensors, will be mounted onto the hardware platform along with the auto-pilot.

Concept Generation for Microcomputer

Criteria	Weight	Raspberry Pi	Odroid
Compatibility	50	8	6
Processor	15	5	10
Memory	15	2	10
Ease of Use	20	7	5
Total	100	645	700

Table 5: Concept Scores - Microcomputer

Description of the Alternatives

Both of these devices are miniature system on a chip computers. We will need to choose one that can drive our sensors and communicate with our ground station.

Raspberry Pi - A very popular micro computer, inexpensive, but not very fast. Also has limited RAM.

Odroid - Another inexpensive micro computer, more powerful, but members of our team are not as familiar with running them.

Decision Criteria and Weight

Cost was nearly the same for both devices, so the only factors that are currently influencing our decision are

Compatibility - Whichever platform we choose, we want to be able to quickly and easily install or program drivers for our sensors. *Weight: 50.*

Processor - Depending on the sensors that we use, we may need to devote lots of processing time to collecting and sending the data However, the efficiency of our algorithm greatly affects this. *Weight: 15.*

Memory - Along with processing speed, memory can help us handle larger amounts of data. *Weight: 15.*

Ease of Use - Because of the semester break in our project, whatever we choose must be easy for another group to learn and develop. *Weight: 20.*

Scoring

Each microcomputer has certain strengths and weaknesses. These scores represent the feeling of our team members, as well as some of the technical specifications of each computer.

Compatibility - *Raspberry Pi: 8. Odroid: 6.* While each computer runs a similar operating system, the Raspberry Pi seems to have more compatibility with some of the sensors we have studied.
Processor - *Raspberry Pi: 5. Odroid: 10.* In terms of computing power, the Odroid has a little more than double the clock speed of the Pi, and is dual core. This might be a large advantage for us, but only if we can write or find software that is efficient enough to utilize it.
Memory - *Raspberry Pi: 2. Odroid: 10.* Memory is also another advantage for the Odroid. It has 2GB, the Pi has 512MB. This may be a large advantage as well, but depending on the sensors we choose, may not be quite as relevant.
Ease of Use - *Raspberry Pi: 7. Odroid: 5.* The Pi scores slightly higher here because of its active developer community. In addition, more of our team members have worked with the Pi in the past, meaning the learning curve will be lower.

Review of the Results

The results of our decision making process are close. For the time being, we have decided to run tests using both computers with our sensors to determine which computer will make it onto the final hardware. Luckily, both computers have similar architectures making it possible for us to run tests on both without unnecessary loss of time. We anticipate having a decision made within two weeks.

Concept Generation for Sensor Types

Criteria	Weight	Wi-Fi	Infrared Near	Infrared Far	Video	Laser Mic	LIDAR	RADAR	Radio
Cost	10	8	5	1	7	8	1	2	8
Weight	15	9	8	4	7	7	4	6	7
Power Consumption	15	6	6	7	7	7	5	2	7
Ease of Use / Robust	25	10	8	7	10	1	1	1	5
Detection Performance	25	7	5	4	8	1	3	3	5
Build Size	10	10	9	6	8	7	6	6	5
Sum/Rank	100	830	675	510	810	410	305	300	590

Table 6: Concept Scores - Sensors

Description of the Alternatives

The following are proposed hardware solutions to be added to the UAV in order to gain intelligence on the object of interest.
Wi-Fi A usb Wi-Fi dongle can be used to intercept Wi-Fi packets from the target.

Infrared-near imaging A solid state digital camera without the infrared filter. Can be used for night imaging, if the subject is illuminated with infrared light.
Infrared-far (Thermal) imaging A sophisticated imaging device that images heat signatures. Can be used for night imaging without infrared illumination.
Video Standard visible light camera.
Laser Mic DIY solution to intercept acoustic signals from the target by bouncing a laser off the target and filtering the incoming optical signal.
LIDAR Laser based range and motion detection system.
RADAR Radio based range and motion detection system.

Decision Criteria and Weight

Cost - There is a final amount of money that is allowed for the final solution ~$10,000. Replacement parts are also purchased for redundancy which places more emphasis on low-cost solutions. *Weight: 10.*
Weight - Weight is an important issue as it directly impacts the maneuverability and battery life of the multi-rotor. *Weight: 15.*
Power Consumption - Battery life is a critical factor of the competition, so power consumption must be kept to a minimum. *Weight: 15.*
Ease of use - Due to a limited amount of time and the requirement to train the next group of seniors, ease of use is paramount. Simple, robust solutions are sought after, customer desires off-the-shelf solutions. *Weight: 25.*
Detection performance - This metric estimates each solution's performance at determining the location of the object of interest, and obtaining valuable information. *Weight: 25.*
Build Size - Hardware size is to be minimized to maintain aircraft stability. *Weight: 10.*

Scoring

Cost - Alternatives with a low score (1 or 2) indicate the cost could be in the thousands of dollars. Scores of 3 or 4 indicate hundreds of dollars. Other scores indicate the cost of a cheap solution, losing one point for every $10.
Wi-Fi: 8 IR Near: 5 IR Far: 1 Video: 7 Laser Mic: 8 LIDAR: 1 RADAR: 2 Radio: 8
Weight - Weight under 100 grams is considered insignificant. One point deduction for every 100 grams.
Wi-Fi: 9 IR Near: 8 IR Far: 4 Video: 7 Laser Mic: 7 LIDAR: 4 RADAR: 6 Radio: 7
Power Consumption - All systems but LIDAR and RADAR have roughly negligible power consumption.
Wi-Fi: 6 IR Near: 6 IR Far: 7 Video: 7 Laser Mic: 7 LIDAR: 5 RADAR: 2 Radio: 7
Ease of Use - One point deduction for every 5 hours of predicted work to produce a robust solution.
Wi-Fi: 10 IR Near: 8 IR Far: 7 Video: 10 Laser Mic: 1 LIDAR: 1 RADAR: 1 Radio: 5

Detection Performance - One point given for 10% predicted probability of detection of useful information. Note, thermal imaging will likely perform well at detecting heat, but is less likely to offer valuable target specific information.
Wi-Fi: 7 IR Near: 5 IR Far: 4 Video: 8 Laser Mic: 1 LIDAR: 3 RADAR: 3 Radio: 1
Build Size - one point deduction for every cubic inch of final build size.
Wi-Fi: 10 IR Near: 9 IR Far: 8 Video: 6 Laser Mic: 7 LIDAR: 6 RADAR: 6 Radio: 5

Review of the Results

After working out all of the multiplications we arrive at conclusions that we had partly anticipated. Solutions such as the laser mic, RADAR, and LIDAR were demonstrated as not viable given a number of issues with many of the scoring categories. The solutions that appear to be viable are the camera, IR near camera, and usb Wi-Fi dongle. These solutions received a high score, because they are simple, easy solutions that offer the desired performance.

Conclusions

We have chosen a Y6 multi-rotor UAV to gather data about an known source. Using the Department of Defense's design specifications and making our own critical assumptions, we have determined how to best gather this data. Camera and Wi-Fi transmission appear to be the best solutions for gathering intelligence. We have yet to decide between the Odroid and the Raspberry Pi for the computing platform, though we have decided on using the Pixhawk autopilot.

APPENDIX C - Project Schedule

- September 2014
    - 1st: Initial Planning and Design Decisions
    - 14th: Hardware Ordered
    - 25th: Assembled Y6 Frame
- October 2014
    - 11th: Raspberry Pi – Capturing and streaming video
    - 21st: Decision made to use Raspberry Pi
    - 28th: Began Preflight testing (motors etc.)
- November 2014
    - 4th: Coupled video stream and Wi-Fi snooping on Raspberry PI
    - 12th: Preliminary Design Review with DOD
    - 28th: Pixhawk Remote Control Flight
    - 29th: Gimbal servo code
- December 2014
    - 5th: GPS mapping
    - 10th: Final Presentation
- January 2015
    - Integrating Mission System Platform and Flight Platform
    - Testing flight capabilities, flight duration with payload.
    - Test Software Radio.
    - Testing Mission Systems link with ground station wirelessly.
- February 2015
    - Testing the solution. Battery Power, Flight Time, Maneuverability etc.
- March 2015
    - Simulation for the final demonstration.
    - Final Report
- April 2015
    - Final Demonstration

APPENDIX D - Tutorials/Instructional Documents

PWM and Servos

https://learn.adafruit.com/adafruit-16-channel-servo-driver-with-raspberry-pi/overview

Backup code if board breaks:

http://abyz.co.uk/rpi/pigpio/

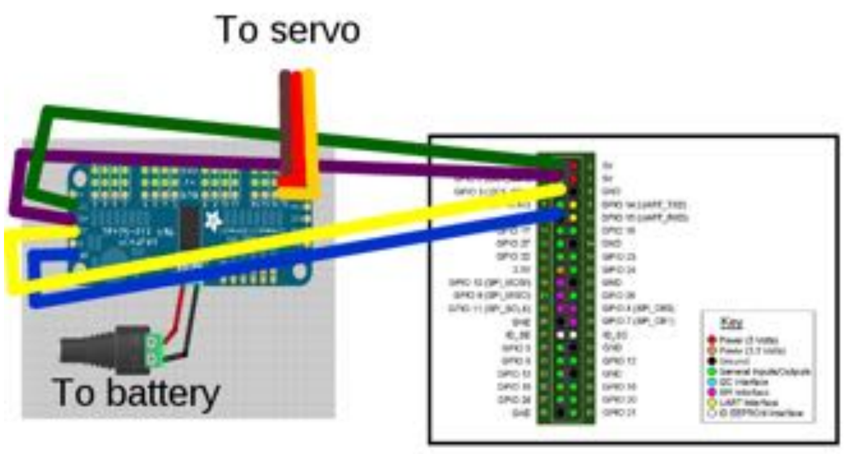

Figure 9: PWM board to Raspberry PI pin connections

Setting up the accelerometer, gyro, compass. Option 1

http://ozzmaker.com/2013/04/29/guide-to-interfacing-a-gyro-and-accelerometer-with-a-raspberry-pi/

(Note this tutorial uses an older version of gyro, acc, compass and there are some differences which are documented on the wiki) .

Setting up the accelerometer, gyro, compass. Option 2

https://github.com/mpolaczyk/Gyro-L3GD20-Python

(Works, but currently only reads Gyroscope values)

Connect Raspberry Pi to Pixhawk.

The Raspberry Pi can be connected to the Pixhawk via mavlink.

http://dev.ardupilot.com/wiki/raspberry-pi-via-mavlink/

Starting Gstreamer

http://pi.gbaman.info/?p=150

Basic RPi / Linux

http://www.raspberrypi.org/documentation/remote-access/ssh/

http://www.raspberrypi.org/documentation/installation/installing-images/

http://www.raspberrypi.org/wp-content/uploads/2013/07/RaspiCam-Documentation.pdf

Backing up the Pi
http://www.raspberrypi.org/documentation/linux/filesystem/backup.md -- Linux
http://lifehacker.com/how-to-clone-your-raspberry-pi-sd-card-for-super-easy-r-1261113524 -- Windows

Basic Pixhawk
This website is good for the hardware side.
http://www.pixhawk.org/
This website is good for the software/firmware side.
http://ardupilot.com