# Brown Water Group – Aerial Mapping System for Urban Environments

**Capstone Design, ME 4182-F**

**Critical Design Review Report**

Prepared for


Dr. Joe Brown
Associate Professor, School of Civil and Environmental Engineering
Georgia Institute of Technology
joe.brown@ce.gatech.edu


and


Senior Design Adviser
Dr. Kathryn Wingate
Academic Professional, George W. Woodruff School of Mechanical Eng.
Georgia Institute of Technology


by


Preston Culbertson
Nicholas Selby
David Steinhoff
Santiago Vivanco


Edited By: David Steinhoff
4 May 2016


Georgia Institute of Technology
North Ave NW,
Atlanta, GA, 30332

# Contents

## Executive Summary

This report presents the design, manufacturing process, and verification of a turnkey aerial imaging system developed by the MAPSAN team for ME-4182 in Spring 2016. The project sponsor, the Brown Water Group, is currently conducting sanitation studies in informal communities in Maputo, Mozambique, but was unable to calculate the population densities of the studied communities, an essential measure for their study. While the Brown Group is currently able to measure the population of the communities via a census, they are unable to use either ground-based or satellite-based methods to measure compound areas.

In this project, a drone-based imaging system was developed that can quickly and accurately create a composite aerial photo of the area being studied from which compound area may be measured. The delivered system includes a hardware flight system, based on a 3DR IRIS+, that uses a GoPro Hero4 to capture a large number of images that may be stitched together to create a composite photo. Further, a software suite was delivered that allows the user to plan flights over a desired area, stitch the large number of captured photos into one high-resolution composite photo, and accurately measure area from the composite photo.

The system was verified during flights at Cobb County RC field, during which the drone system followed a series of user-generated flight paths, and demonstrated a variety of safety features, such as auto-landing and come-home commands, which ensure the system will remain under the control of the operator in emergency situations. In addition, the image-stitching and area measurement software was verified using sample images generated from the Google Maps aerial photography dataset.

Future work includes a full system test, which demonstrates the system following user generated waypoints, capturing a series of photos that define the desired area, stitching the aerial photos, and accurately measuring area from the aerial photography. This test was not performed due to a manufacturing defect with the purchased Tarot gimbal used for camera stabilization, and will be completed before final delivery to the project sponsor. In addition, the Brown Group plans to extend the functionality of the delivered system via other sensor packages to use it as an aerial platform to capture previously unavailable datasets.

## Nomenclature

Ah: Amp-hours

BOM: Bill Of Materials

BWG: the Brown Water Group

COTS: Commercial Off-The-Shelf

DSLR: Digital Single-Lens Reflex Camera

D/W: Demand or Want

GIS: Geographic Information System

GPS: Global Positioning System

ID: Identification

iOS: Operating System found on Apple devices

LiPo: Lithium Polymer battery

MAPSAN: MAPuto SANitation study

MP: Mega-Pixel

NGO: Non-Governmental Organization

OS: Operating System

PDR: Preliminary Design Review

SSR: Systems Requirement Review

UAV: Unmanned Aerial Vehicle

w/: With

w/o: Without

# Main Body

## 1. Introduction and Background

Residents of informal urban communities often suffer from poor health outcomes due to lack of proper sanitation. These communities often lack latrines and basic sanitation practices, enabling the spread of diseases and microbes between residents. To improve health outcomes for these communities, many NGOs have proposed using sanitation “interventions” in which latrines and anti-microbial medicine are provided to residents. Problematically, the improvements in health outcomes from these interventions have not been quantified and it is unclear whether these methods are adequate or effective for improving the health of these communities.

Researchers, such as the project sponsor Dr. Joe Brown, are currently partnering with local NGOs to quantify the health gains from sanitation interventions. Dr. Brown is currently conducting studies in Mozambique and Nicaragua. An essential variable in these studies is the population density of the studied communities which may only be measured by determining the areas of the compounds that make up informal communities. Dr. Brown’s team is currently unable to measure the size of their studied compounds effectively and thus have no way to accurately measure the compound’s population density. Walking the perimeter of each compound is difficult and time-intensive meaning ground-based area measurement methods are not suitable for this application. Furthermore, satellite-based imaging methods are prohibitively expensive and cannot provide the resolution necessary to accurately identify or measure the compounds.

This project aims to develop a comprehensive solution for measuring the area of a compound to determine the population density of communities being studied by Dr. Brown’s team. This first involves designing, fabricating, and verifying an aerial imaging system capable of capturing high-resolution aerial images of the compounds. The images must have a resolution of approximately 20 cm (8 in.) to be capable of detecting important features such as compound walls. The imaging system will be used by locals with little training or experience necessitating that the system be simple and quick to learn.

This project will develop a turnkey solution that enables Dr. Brown’s team to image, measure, and visualize the areas of the compounds they are studying. To date, no existing imaging system can provide the necessary resolution and ease proposed in this report at an acceptable cost. Additionally, the hardware and software developed in this project may be used by other researchers, governments, and NGOs to conduct similar studies and collect valuable data to inform policy and health decisions.

An important constraint on this system is the privacy concerns of the communities' residents. Many people are uncomfortable being imaged by a drone system meaning the system must be non-intrusive, quiet, and respectful of residents' privacy. Another constraint is that regulations exist on flight above the cities being mapped (*e.g.* airspace restrictions, drone laws). These regulations may be exceptionally strict, as some of the areas studied by BWG are located near airports and may be subject to additional rules. Fortunately, these constraints may sometimes be mitigated by the lax regulatory climate in countries like Mozambique which do not currently have any laws specifically covering drone flight.

Upon its completion the deliverables for this project will include a demonstrated aerial imaging system that can quickly image an area of at least 4 $km^2$ with little user input, a flight platform that is capable of lifting a payload of ~400 g, a software system that can process and store this aerial imagery, and an accurate and easy-to-use interface for identifying and measuring objects in the captured images.

## 2. Prior Art

The various systems for aerial mapping that are currently on the market range from drone systems to satellite imaging. Google Maps maintains and updates an almost worldwide satellite imaging system that allows for high resolution mapping of certain areas like the United States, but they do not conduct similar high resolution mapping of Southern Africa. Another option is specialty satellite imaging companies like Satellite Imaging Corporation or Digital Globe that are also capable of taking satellite imagery of Southern Africa, but not at the 20cm resolutions required for this project[1][2]. Fixed-wing, surveying drones like the senseFly eBee cover areas with better resolutions but are prohibitively expensive (*e.g.* the eBee costs $12,000) and are generally used for larger areas than the communities in this project[3]. Rotary-wing drones like the DJI Phantom 3 or Yuneec q500 4k come with cameras built in that are capable of meeting the hardware specifications for this project but don't contain the programming to perform the necessary software tasks[4][5].

The selected design makes use of an existing drone system like the above but with a custom attached camera system to collect aerial images of informal communities. Existing image processing software may be used but it will have to be open-source to allow for customization and optimization for the specific requirements of this project. Patent research finds numerous examples of patents for drones, cameras, and drones with cameras, but the usage of the camera is typically for remote controlling of the drone rather than data collection. There is a low level of concern of patent infringement in this design project given the academic nature of BWG and the noncommercial usage of the images collected.

## 3. Customer Requirements

Stakeholders for this project include BWG, the governments of developing countries along with their citizens. BWG has the highest importance and influence on the overall development of the design and is the main source of the provided customer requirements. An example stakeholder analysis for this project is collected in Figure 1.

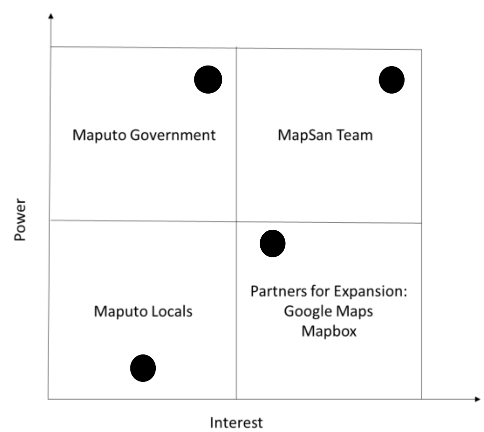



**Figure 1. Stakeholder Analysis Example for Maputo.**

Some of the requirements include mapping and calculating the area of large compounds, identifying cinderblock walls within each compound, and maintaining an easy to use interface. The imaging must be done from an aerial view in order to capture a suitable picture for processing and calculations. The final results obtained must be presented in a matter simple enough for anyone to interpret.

Functions the design must accomplish include carrying a camera in order to capture images at the desired height, as well as providing full control of the frequency and location of images. Constraints in this design relate to developing a system from open sourced hardware and software in order to provide the required controls to complete the required tasks, as well as minimizing the total cost of the system and fulfilling flight regulations of Atlanta along with developing countries. To satisfy the minimum resolution required for this project, the camera selected must have at least 12 MP resolution with a focal length of 14 mm. The design needs to be able to travel a horizontal distance of 2 km and complete its mission in less than 6 hours. The battery requires a minimal storage capacity of 5Ah, providing enough energy to complete one compound per battery and reducing the number of change outs required during use. A complete list of specifications can be found in Table 7 in Appendix A.

## 4. Market Research

There was not a large amount of market research performed for this project. The sponsor provided a set of requirements and desired functionality that directly drove the design of the final imaging system. An external market for the imaging system could be other public health organizations that are in need of health data of similar form to the population density found in this project. Since the project has been in development there have been several groups that have shown interest in using the flight and software platforms developed for this project for things other than imaging but they were considered too far outside the scope of the project.

## 5. Design Concept Ideation

A complete functional breakdown of the design concept is found in Figure 11 in Appendix B. The following sections break down the ideation by subsystem.

### 5.1 Drone:

The drone for this project needs to be able to ascend, traverse a large area holding a camera, and return to home if commanded.

### 5.2 Controller:

The controller needs to be well-documented and open-source to allow programmers maximum flexibility and support for custom programming, interfacing, and flight planning. In addition, the controller must interface with the drone selected.

### 5.3 Camera:

The controller will also operate a camera to photograph areas at predetermined waypoints. The camera needed to be of high enough resolution and fast enough shutter speed to identify 8 in. cinderblocks at an altitude of 200 ft. Also, because any added mass significantly decreases battery life, lighter cameras are preferred.

5.4 Gimbal:

Without a gimbal, the vibrations and camera view angle inherited from the drone would make discernable imagery impossible. To mitigate both effects, a gimbal will be the interface between the drone and the camera. It will also need to interface with the controller.

5.5 Microcontroller:

A microcontroller will be used both to store photos taken by the camera. Thus, it will need to have the storage capacity capable of storing all 6,800 photos and interface easily with the controller and desktop. Additionally, a high level of documentation and flexibility is preferred to facilitate future work such as the potential to communicate with a mobile app.

## 6. Concept Selection and Justification

6.1 Drone:

In order to spend more time troubleshooting algorithms, so-called "do it yourself" drones were not selected in favor of heavily tested, well documented, off-the-shelf solutions. The 4 $km^2$ area eliminated the possibility of using the least expensive hobbyist drones which were often severely limited in range due to hardcoded distance limits and/or battery life. Additionally, the desired robustness and scalability of the project made open source control essential, eliminating many professional drones which have heavily guarded and un-editable control algorithms. The minimum payload requirements needed to hold the camera and gimbal further eliminated less powerful drones on the market. The drone which meets all engineering requirements for the lowest cost is the 3DR IRIS+, a highly popular drone among hobbyists and professionals alike. The IRIS+ advertises 16 minute flight times, a 400 g payload limit, and an open-source controller.

6.2 Controller:

The 3DR IRIS+ offers a well-documented, open-source interface through a built-in Pixhawk controller and impressive engineering specifications for its price. The Pixhawk is easily programmable with predetermined GPS waypoints and can interface with almost any hardware. Selecting the IRIS+ forced the choice of using a Pixhawk controller as it is not removable from the drone.

6.3 Camera:

The microcontroller will also operate a GoPro Hero 4 camera to photograph areas at predetermined waypoints. The GoPro Hero4 has a 3000 pixel linear resolution, a 8192 Hz shutter speed, and weighs only 88 g. The camera specifications implied that it would be able to precisely identify key points on the map, and a drive-by test using a car and a square checkerboard confirmed that several pixels spanned an 8-in. distance. Thus, images taken by a GoPro Hero4 will show the cinderblock walls separating compounds.

6.4 Gimbal:

A Tarot T-2D gimbal was selected to be the interface between the drone and the camera. The Tarot's specs and cost are both average for two-axis gimbals on the market, but the Tarot in particular has been coupled with Pixhawk controllers in several well-documented applications including use on the IRIS+ drone. As a general component selection philosophy, this team chooses components that are already extremely well tested and 3DR heavily recommended the Tarot T-2D gimbal.

6.5 Microcontroller:

The GoPro will store photos in a Raspberry Pi before they are transported onto the desktop for processing. Of the various microcontrollers available, Arduino, Raspberry Pi, and ODROID are all technically capable of handling the processing and storage required, but Raspberry Pi is well documented and costs the least.

## 7. Industrial Design

The final physical design consideration was not strongly influenced by industrial design. The project only calls for the creation of a single functioning imaging system and the base components (*i.e.* the drone, the camera, the gimbal, and the pump) were all bought off the shelf and adapted into the system based on their functionality alone. The bright coloring of the drone itself was the aesthetic choice of the drone's manufacturer, but for the purposes of the aerial imagining system the bright coloring is desirable because it makes the drone easier to spot and thus harder to lose. The branding on the aerial imaging system reflects the respective brands of the components and there is no plan to add a new logo for this project other than maybe an identifying tag for the BWG.

## 8. Engineering Analysis and Experiments

The testing of the aerial imaging system design was done in three main stages, a preliminary camera testing stage, a software simulation stage, and a hardware stage. In consideration of the legal requirements discussed later in this report, discrete tests to validate individual specifications were not performed. The system was instead validated using high level systems tests that stressed all major points of functionality. Using a rapid, iterative troubleshooting process allowed the aerial imaging system to get as close to fully operational as possible during testing within only a few hours. Specifications of interest that needed to be tested were validated where possible from the general flight data collected during hardware testing. Some significant hardware failures and testing limitations prevented some specifications from being validated requiring that further testing be performed.

### 8.1 Preliminary Camera Testing:

The camera testing was completed before the IRIS+ was selected as the drone for the design. An in-detail analysis of the testing is collected in Appendix C. The GoPro was tested on the ground using a pattern with 8 in. square patterns on it simulating the features that would need to be visible in the final design. The pattern was placed at a known distance of 200 ft apart and pictures were taken while the camera was static and moving 25mph in a car. The static test case was a complete success and each square of the test pattern was easily visible by eye. The 25 mph test was also a success in that the squares of the test pattern were visible by eye, but there was slight aliasing around the edges of the picture. By increasing the overlap between pictures in the design software the aliasing issue was resolved and the 12 MP resolution of the GoPro was validated as sufficient for the needs of the aerial imaging system.

### 8.2 Software Testing:

Before physical testing began, the design software was tested using a simulation suite designed to replicate the complete conditions of the aerial imaging system in flight. jMAVsim is the specific software used in this project and was chosen because it was created to support the Pixhawk controller found inside the IRIS+ drone. The benefit of using the simulation software was the comprehensive suite of tests it could perform on the drone without requiring actual flight. The basic functionality of jMAVsim is to take any outputs the Pixhawk might send to the drone and divert them into a computer simulation that comprehensively accounts for most predictable flight conditions. To run jMAVsim for this project a computer with the simulation suite pre-installed was connected directly into the hardware setup for the design via an access port on the drone. jMAVsim performs the initial activation sequence for the software and the Raspberry Pi takes over the auto-navigation afterwards. Using the simulation software to thoroughly debug the code for the project demonstrated that by the beginning of real-life testing the software side of the aerial imaging system would, in theory, perform to spec for the design.

### 8.3 Hardware Testing:

Hardware testing was performed at the Cobb County RC test field located roughly 45 minutes by car away from Georgia Tech's campus. All hardware testing was conducted while a licensed RC pilot held the remote controller for the IRIS+ drone and a representative from Georgia Tech's legal department observed. Initial flights of the design were to allow the pilot to become familiar with operating the drone and its emergency failsafe systems. Once the pilot was ready, the testing of the autonomous functions of the aerial imaging system began. Detailed flight data was relayed to the remote controller and recorded by the Pixhawk controller before being collected after each flight. The critical specifications that needed validating are collected in Table 1.

**Table 1. Validation Testing Results**

| Specification | Validation Result | Explanation |
|---|---|---|
| Top Speed >25mph | PASS | Drone exceeded 25 mph multiple times |
| Hover Time >15min | PASS | Hover time was 15.5 minutes |
| 12 min flight time per battery | PASS | Average flight time of each battery was ~15 minutes |
| Max range >1km unobstructed | INCONCLUSIVE | Field too small for test |
| Works in wind speeds <10mph | PASS | Wind speed during testing was 13 mph and drone functioned |
| Lose contact with drone for 20 sec and still complete mission | INCONCLUSIVE | Pilot prevented any testing that tampered with fail-safes |
| 8 in. features must be visible during flight | INCONCLUSIVE | Gimbal manufacturing defect prevented aerial testing of camera |

The flight speed, hover time, and flight time of the system were above the specifications required for this project. The wind speeds the day of testing happened to be around 13mph which was high enough to validate the wind speed specification required for the imaging system. Several key specifications could not be validated due to several reasons. A manufacturing defect in the gimbal that was purchased made it unable to point downwards preventing testing of the camera in the air. The field where testing occurred was not large enough to test the maximum operating range of the system. The RC pilot would not allow for modifications to the built-in abort features that came with the IRIS+ preventing testing of the delayed return feature also referred to as the heartbeat timeout. There will need to be future testing before the system is ready for use. The plan for this future testing is to acquire a replacement gimbal from 3DR, locate a larger RC test field, and develop a heartbeat functionality that does not require disrupting the drone's abort settings.

## 9. Final Design

### 9.1 Final Software Design

To plan the mission, the user begins by opening a path planning JavaScript app on the desktop. The app runs on the Google Maps API and allows users to view satellite imagery of the regions they wish to map. Users begin by inputting photography specifications, including the altitude at which the drone will be flying and the technical specs of the camera. Then, the user draws a polygon over the area they wish to photograph on the map and selects the home point to which the drone will return after every flight as seen as the red pin in Figure 2. The app uses a scan-line fill algorithm to populate the polygon with nodes spaced just near enough such that the camera will be able to image the entire area. The GPS locations of each node is then stored in a CSV.


**Figure 2. Path Planning User Interface**

After clicking “Download CSV,” the user runs a MATLAB function on the generated CSV. Because of the large area, the drone is unable to visit all nodes in one flight. Thus, the function uses a novel, iterative path planning algorithm to calculate the set of paths which link every node in the least total round trips, generating a single CSV for each path containing the GPS coordinates of every point. The algorithm takes a heuristic approach to a variant of the symmetric distance-constrained vehicle routing problem which this for is designated the symmetric, distance-constrained, maximally interconnected vehicle routing problem due to the completeness of the map in a graph theory sense. Example results of the algorithm are visualized in Figure 3.

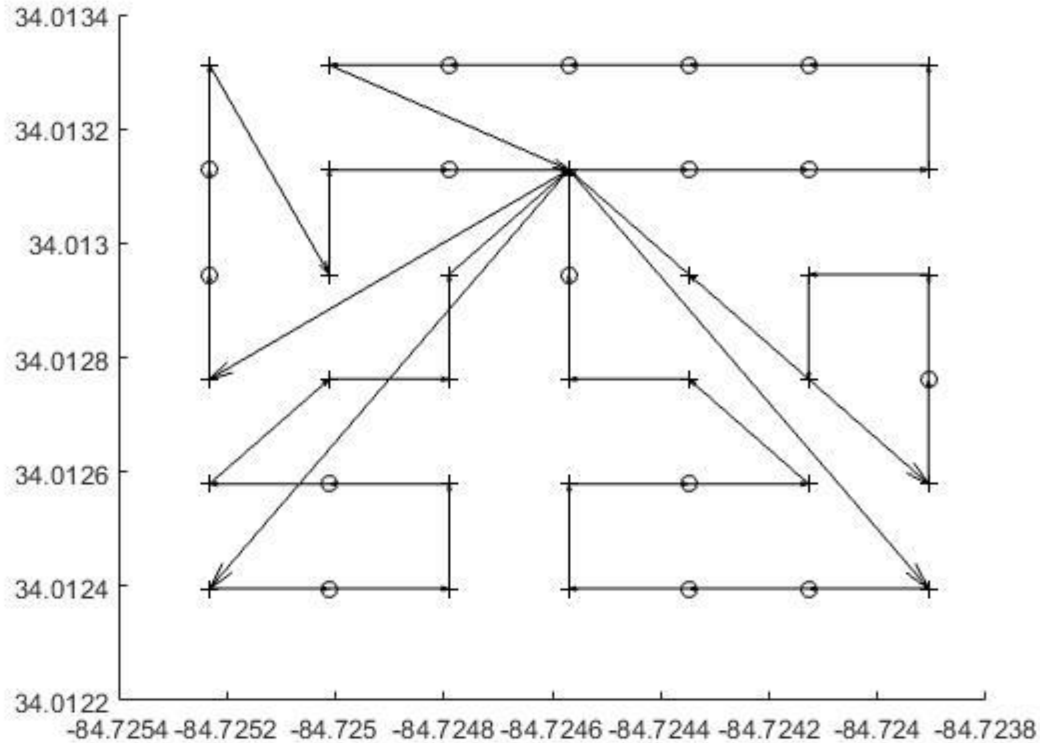



**Figure 3. Example Results from the Path Planning Algorithm**

Next, the paths are uploaded to the Raspberry Pi on the drone. The Raspberry Pi interprets the coordinates from the CSV as instructions for the PixHawk controller. In flight, the Raspberry Pi also instructs the GoPro to take pictures whenever the drone enters a predefined proximity radius of the desired waypoint.

After the flight is complete, the user downloads the pictures from the GoPro onto the desktop and runs a second MATLAB script which stitches the images into a single panoramic. The stitching algorithm is based off of MATLAB's feature based panoramic image stitching algorithm from the Computer Vision System toolbox. However, MathWorks' algorithm does not take image order as an input and, as a result, is inaccurate and inefficient for large image sets. The algorithm used in this design recursively divides the image set into quadrants, stitches those subsets individually, then stitches the quadrants together into a single panorama using MathWorks' standard algorithm. An example taken from the validation testing is shown in Figure 4.

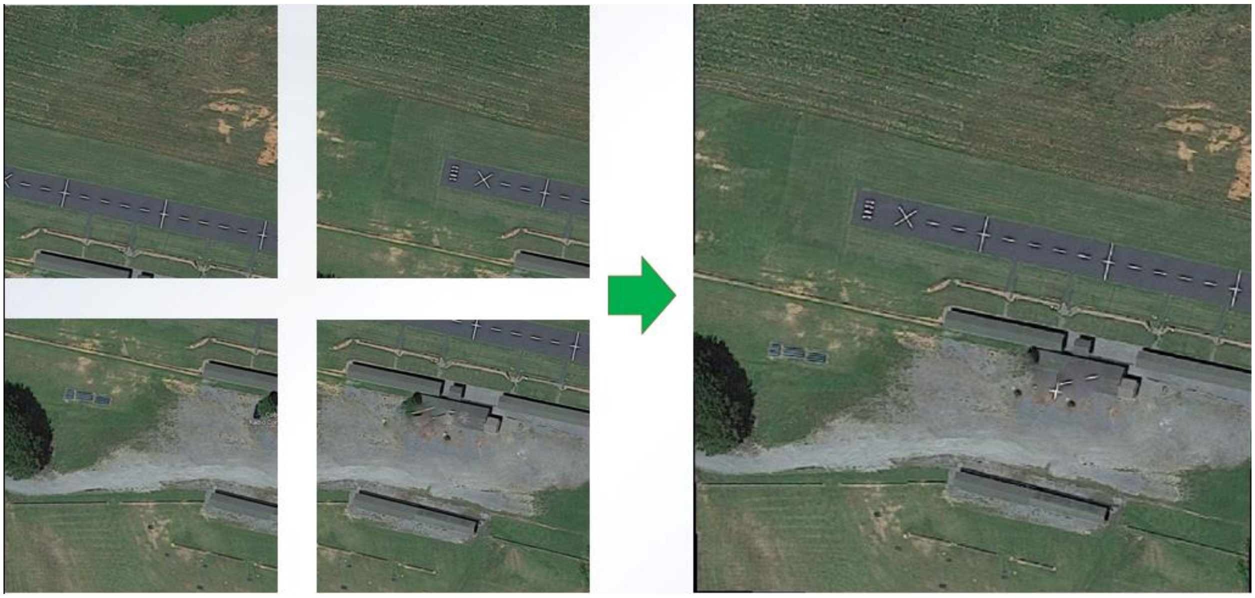


**Figure 4. Example Panoramic Stitching**

Finally, with a single panorama of the entire area, a combination of computer vision and user input is used to identify compound boundaries. Users import the panoramic image back into the JavaScript UI and help the computer identify the cinderblock walls separating compounds. From this, the app calculates the area of each compound and outputs the resulting population density calculation results.

9.2 Final Hardware Design

The final design on the hardware side is comprised of four subsystems: the drone, the gimbal, the controller, and the camera. Due to the academic nature of this project only one final product is being created as a deliverable. There was no prototype created and there are no current plans to produce any more copies of the aerial imaging system developed for this project. Although there were some desired features that had to be cut from the design over the course of the project, the final hardware design is completely capable of performing the required task of collecting aerial images at areas that measure in the square kilometers.

9.2.1 Drone:

The drone is a 3DR IRIS+ that serves as the flight vehicle for the imaging system. A Pixhawk controller inside the drone handles the stabilization and control of the system during flight and collects location information using GPS. Communications with the drone are handled using either a 915 MHz antenna in the United States or a 433MHz antenna anywhere else in the world. During the use of the drone a remote controller, that is built for the IRIS+ and provided by

3DR, is held by the drone operator to keep an eye on the flight information and to abort the flight if necessary. The drone is turned on by inserting and connecting the battery which should only be done once the drone is already at its launch position. Once turned on the drone has a safety button that needs to be held for several seconds to arm the motors and trigger the auto-navigation code. The default legs of the IRIS+ are replaced with provided extended plastic legs that allow the drone to stand without the rest of the system touching the ground. The power source for the drone is a 5100 mAh, three cell, LiPo battery that sits inside the IRIS+'s shell next to the Pixhawk. As part of the imaging system's demand for multiple flights, six additional batteries were purchased each of which lasts for about 15 minutes of flight and takes 90 minutes to recharge. To shorten the net charge time a HobbyKing Quattro 4x6S multi-charger was included in the design to reinforce the 3DR provided charger allowing for up to five batteries charging at a time. The multi-charger requires a power source as well so a HobbyKing 350W 25A power supply was added to the design however a car battery can also work when a reliable power grid is not available. The drone is highlighted in purple in Figure 5.

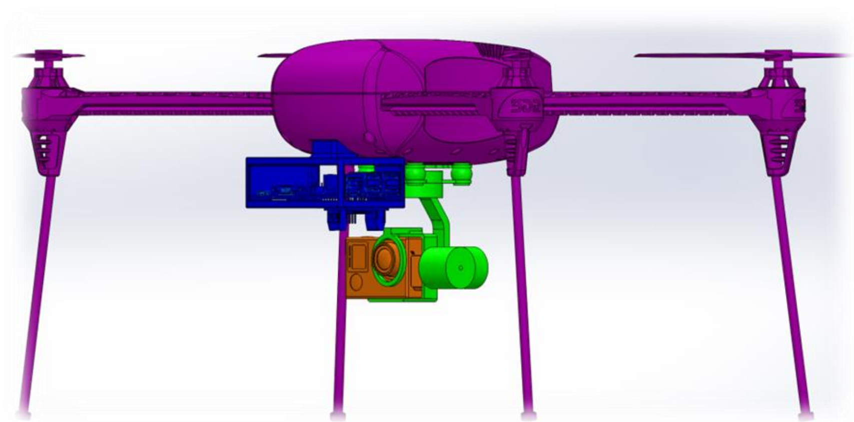


**Figure 5. System View with 3DR IRIS+ Drone Highlighted in Purple**

<u>9.2.2 Gimbal:</u>

The gimbal is a Tarot T-2D that is built specifically to mount a GoPro to an IRIS+ drone in a stable manner so that the camera always faces straight downwards. There are two electrical connections between the gimbal and the drone. The first is a power connection and the second is a reference position signal that is set by the remote controller. The stock power connection uses a quick-disconnect for easy installation of the removal and for the aerial imaging system that is supplemented with a multi-connector that allows the connection to also power the Raspberry Pi.

Whenever the gimbal is powered-on by the drone it defaults to facing forward and has to be angled towards the ground via a dial located on the remote controller before operation can begin. As long as the gimbal is roughly placed towards the ground the imaging shouldn't be affected. The gimbal is seen in Figure 6.

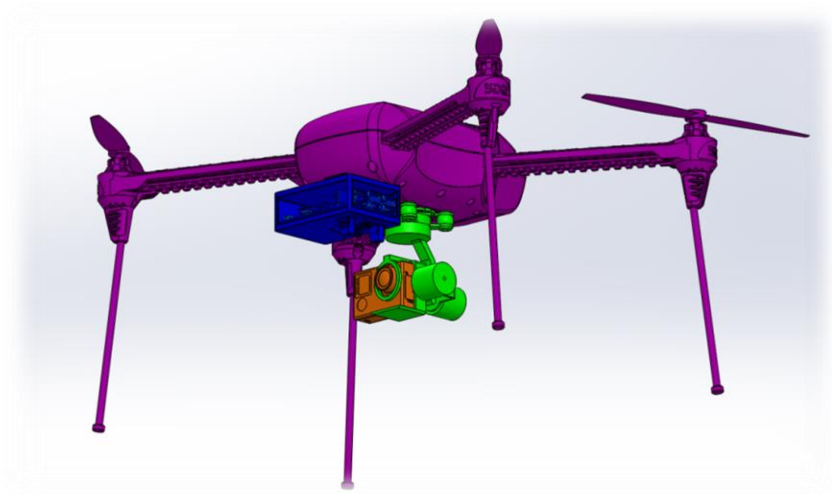


**Figure 6. System View with Tarot T-2D Gimbal Highlighted in Green**

9.2.3 Microcontroller:

The microcontroller is a Raspberry Pi 2 that is used as the central brain of the design. The Raspberry Pi communicates with the ground station to receive the coordinates for each flight and talks with the drone both to upload those coordinates for navigation and to check when coordinates are reached to take pictures. The Pixhawk on the IRIS+ is capable of receiving autopilot navigation coordinates on its own but it needs the Raspberry Pi to keep track of when picture waypoints are missed at which point the latest navigation waypoints are reset to attempt another picture. Whenever the drone's GPS signal tells the Raspberry Pi that it is within an acceptable distance of a picture waypoint a trigger signal is sent via Wi-Fi to the camera causing a picture to be taken. The Wi-Fi connection between the camera and the Raspberry Pi is secured via password to prevent any accidental tampering from an observer with a GoPro app on their phone. The Raspberry Pi is powered directly from the drone battery via the gimbal power connection and a voltage regulator. Great care has to be taken not to power the Raspberry Pi with an external source unless the drone is also powered as the power connection is two-way. The Raspberry Pi is mounted to the IRIS+ via a custom 3D-printed mount that attaches with two shoulder screws that come with the drone. The screws were originally used to hold a component of the drone that secures cameras in place without a gimbal but that is unnecessary for this design and the screws were repurposed to conserve on weight. The mount is placed on the front of the drone because that is where the existing holes are, and serves mostly as a place for the Raspberry Pi to sit during flight and the large number of wires attached to the Pi is enough to secure it from falling out. The microcontroller and its mount are seen in Figure 7.

**Figure 7. (Left) System View with Raspberry Pi and Mount Highlighted in Blue and (Right) a Close-up Rendering of the Raspberry Pi in its Mount**

9.2.4 Camera:

The camera is a GoPro Hero4 Black Edition that is used to take and store all of the pictures for the imaging system. The GoPro sits securely inside the gimbal and faces straight towards the ground throughout the mission. Every picture taken is saved locally on a microSD card that can be quickly offloaded and backed up between flights. The camera acts a secure access point that any Wi-Fi enabled device can access with the correct password. Whenever the GoPro receives a trigger signal over Wi-Fi it takes a picture and records the timestamp of when that photo was taken. The GoPro has its own battery that needs to be recharged periodically through a standard miniUSB charging cable. The camera is seen in Figure 8.

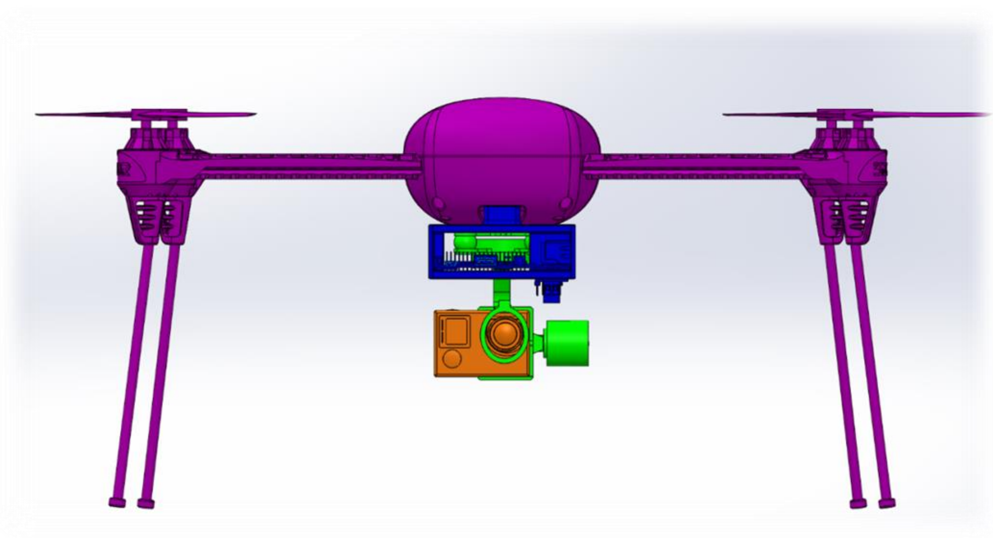


**Figure 8. System View with Camera Highlighted in Orange**

9.2.5 Complete System:

Some high quality renders of the entire system are seen in Figure 9 and an actual photo of the finished system is seen in Figure 10.

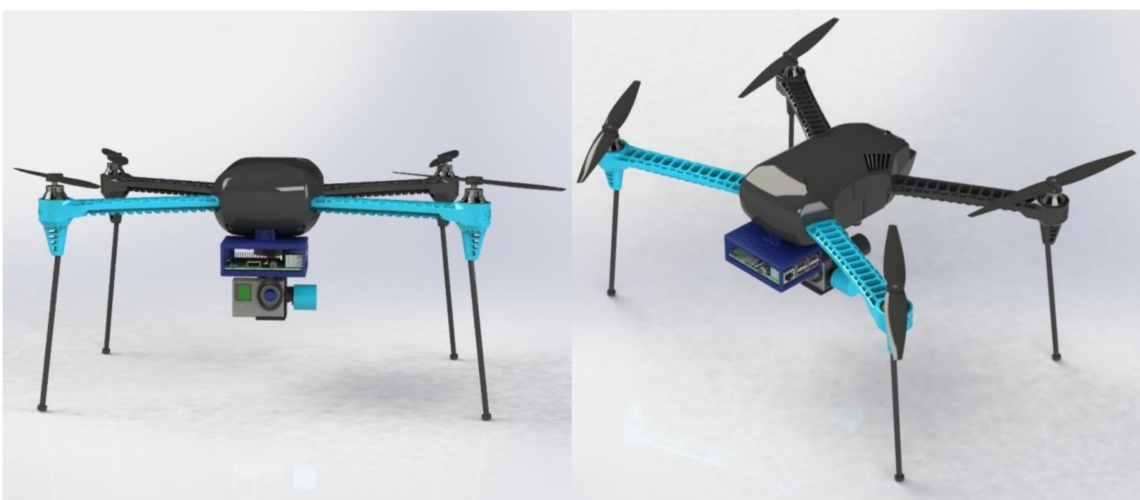


**Figure 9. High Quality Renderings of Complete Aerial Imaging System Hardware**

**Figure 10. Photo of Complete Final Hardware Design on Launch Pad**

The way the system works from a hardware perspective is that a predetermined flight plan is uploaded to the Raspberry Pi with both navigation coordinates and picture waypoints. The drone battery is disconnected and the system is placed at the launch point clear of any aerial obstacles. To launch the drone, the remote controller is set to autopilot mode, the drone battery is connected, the safety button is pressed until it beeps, and as soon as the operator turns the motors on the preprogrammed imaging flight will begin. There is a ten second delay built into the program for people to get out of the way before the propellers start spinning and the system will always begin its flight by flying to a relative height of 200 ft straight up. Once the flight height is reached the drone will hover for ten seconds and then move towards the first navigation coordinate. As the system is in flight the drone will send the GPS location to the Raspberry Pi and whenever a picture waypoint is reached a signal will be sent to the camera to take a picture. When a picture is taken the timestamp and actual GPS location will be recorded for later use. The system will repeat this process until it is finished at which point it will return to a point directly above the launch point and slowly descend. After the flight is completed the battery will be replaced, the pictures will be taken off the camera, the flight data will be taken off the drone, and the next flight will begin until the entire area is mapped.

The entirety of the software side was developed using available, open-source programs so the costs of this project fell on the hardware side. Table 2 is a Bill of Materials for the hardware collected for the final system design.

**Table 2. Hardware BOM**

| Part Name | Quantity | Cost | Individual Total |
|---|---|---|---|
| 3DR IRIS+ | 1 | $599.99 | $599.99 |
| Tarot T-2D Gimbal | 1 | $210.00 | $210.00 |
| 3DR IRIS+ Battery Pack | 6 | $39.99 | $239.94 |
| GoPro HERO4 Black Edition | 1 | $499.99 | $499.99 |
| microSD Memory Card 64 GB | 1 | $17.95 | $17.95 |
| Raspberry Pi 2 | 1 | $40.00 | $40.00 |
| HobbyKing™ Quattro 4x6S Lithium Polymer Multi Charger | 1 | $99.99 | $99.99 |
| HobbyKing™ 350w 25A Power Supply (100v~120v) | 1 | $43.94 | $43.94 |
| Charge Cable w/ Male XT60 <-> 4mm Banana plug | 4 | $3.23 | $12.92 |
| Cables Unlimited 6-feet Mickey Mouse Power Cord | 1 | $4.49 | $4.49 |
| | | **Total** | $1,769.21 |

## 10. Manufacturing

The scope of this project is limited to the delivery of only one complete aerial imaging system that has already been built. There are no plans to produce any more of the final product and as such not much consideration was given to the viability of mass production. The majority of the system is made from off-the-shelf parts meaning there was not a large amount of material selection considered. The one mount that was made specifically for this project was 3D-printed because plastic is light and it could be 3D-printed quickly. Again, only one of this part will be made and it is already finished so there is not a large concern for how to make another. The largest deliverable for this project from a manufacturing standpoint is a set of instructions for the assembly of the components into the complete system. The specific items needed for assembly that will be provided are: instructions for mounting the gimbal which were provided by 3DR, instructions for mounting the camera which were provided by GoPro, instructions for mounting the Raspberry Pi, an Allen wrench set, and an electrical wiring guide.

## 11. Codes and Standards

FAA regulations regarding academic drone flights require special registration of drones. In order to avoid this months long process it is possible to fly a university owned drone under Georgia Tech's general Certificate of Authorization although this can only be done at a certified drone testing field and a representative from Georgia Tech's legal department must be present as well as a licensed RC pilot. Some other codes and standards that are applicable to this project are collected as follows:

Title 49 U.S.C. § 40102(a)(41)(C)[6]

"public aircraft" means: An aircraft owned and operated by the government of a State, the District of Columbia, or a territory or possession of the United States or a political subdivision of one of these governments, except as provided in section 40125(b).

Title 49 U.S.C § 44103(a)(1)[7]

On application of the owner of an aircraft that meets the requirements of section 44102 of this title, the Administrator of the Federal Aviation Administration shall register the aircraft.

Mozambique Drone Laws

Do not fly your drone near airports or in areas were aircraft are operating[8].

Nicaragua Drone Laws

Drone use is currently banned in Nicaragua and drones are not allowed to be taken into the country[9].

## 12. Societal, Environmental, and Sustainability Considerations

Due to the current use of UAV's by the Department of Defense and the prevalence of military drones in the media, many people around the world associate their use with the invasion of privacy and even death. Thus, it is critical that the implementing team develop relationships within the communities over which this application will be deployed, informing the community of the UAV's academic purpose and reassuring their safety and privacy. To protect the privacy of individuals whose homes will be photographed, the pictures will be analyzed solely by the computer for population density measurements alone, then deleted without release to any third party.

## 13. Risk Assessment, Safety, and Liability

Of the various risks needing mitigation, the greatest risk is legal. Maputo is too close to an airport to allow drone flights, and the first backup location of Nicaragua confiscates drones at the airport. Thus, the MAPSAN administration (Joseph Brown, et al.) is working with the government officials in those regions to attempt to acquire permission. Simultaneously, testing and validation is being performed in conjunction with the Georgia Tech Research Institute, who has express permission from the FAA to conduct drone testing.

The risk of operator failure and injury, while low, would be catastrophic. Thus, extensive testing is required for all operators of the craft, and an intuitive procedure has been developed. For testing within the United States, only licensed pilots are legally allowed to fly the drone, further decreasing risk of operator failure.

During normal operation, the operator simply acts as a backup in case software fails. The controller is programmed to prioritize operator input over preprogrammed waypoint navigation. Thus, in the event of software failure of any kind, the operator will still be able to return the drone to home.

Most hardware failure risk, including loss of servo, rotor, gimbal, or battery, can only be mitigated via extensive testing. Battery fire, a potentially disastrous risk for Lithium-ion batteries, can be prevented with protective circuitry.

The collision of the drone with birds or airplanes would be obviously disastrous for the mission. While birds generally avoid other flying objects, a large flock flying near the drone, as well as low-flying airplanes, are predictable but dangerous risks. Thus, if the operator perceives a threat to the drone in the air, they shall immediately signal the drone to return to home. A risk assessment matrix collecting these and other risks are collected in Table 3.

**Table 3. Risk Assessment Matrix.**

| Hazard # | Hazard | Frequency | Severity | Initial Risk Level | Mitigation | Final Risk Level |
|---|---|---|---|---|---|---|
| | **Collision** | | | | | |
| 1 | Collision with airplane | A | 3 | High | Come Home button to tell drone to abort and return | Low |
| 2 | Collision with bird | B | 3 | Medium | Come Home button to tell drone to abort and return | Low |
| | **Hardware Failure** | | | | | |
| 3 | Drone failure over passerby | D | 2 | Medium | Live diagnostics with "Go Home" feature for emergency use. | Low |
| 4 | Loss of rotor | D | 3 | Low | Heavy testing prior to mission | Low |
| 5 | Loss of servo | D | 3 | Low | Heavy testing prior to mission | Low |
| 6 | Battery failure | D | 3 | Low | Heavy testing prior to mission | Low |
| 7 | Battery fire | D | 2 | Medium | Protective circuitry. Battery monitoring. | Low |
| 8 | Gimbal failure | D | 4 | Low | Heavy testing prior to mission | Low |
| | **Software Failure** | | | | | |
| 9 | Failure to communicate "Come Home" | C | 3 | Medium | Remote control redundant manual control | Low |
| 10 | Done flyaway | D | 4 | Low | Remote control redundant manual control | Low |
| 11 | Failure to recognize compound boundary | B | 4 | Medium | Communication with locals to identify compound boundary | Low |
| | **Operator Failure** | | | | | |
| 12 | Operator injury during liftoff | C | 1 | High | Well documented startup procedure. Delay before launch. Protective gear and training for Operator. | Low |
| 13 | Operator injury during landing | C | 1 | High | Well documented landing procedure. Protective gear and training for Operator. | Low |
| | **Miscellaneous** | | | | | |
| 14 | Legally stopped from flying | A | 4 | High | Coordination with Maputo legal offices beforehand through liaison. Optional change of venue to rural area. | Low |

## 14. Patent Claims

This project is not specifically intended for commercialization and no patent claims are currently planned. The algorithm developed for minimizing the number of flights required is novel and there are plans to publish it.

## 15. Conclusions and Future Work

This report has presented the design and testing data of a turnkey aerial imaging system developed by the MAPSAN team. The system manufactured consisted of a 3DR IRIS+ drone, a Raspberry Pi-based flight computer, a Tarot T-2D gimbal for image stabilization, and a GoPro Hero4 camera for imaging. A software suite was also developed that successfully generated flight waypoints and paths, as well as stitching and processing real distance measurements from the aerial photos. An integrated systems test was successfully performed to demonstrate the system successfully navigating to software-generated waypoints, as well as many of the safety features of the flight platform.

Due to a manufacturing defect in the COTS gimbal used for camera stabilization, a fully integrated systems test that verified navigation, camera triggering, and image stitching remains outstanding, and will be completed in Summer 2016 before the project is delivered to the sponsor. In addition, some software improvements will be made to the user interface to improve its ease of use and design. The system is slated to complete its first in-field flight in Mozambique in Summer 2016.

Much future work remains on this project, as the project sponsor plans to use the flight platform developed by the MAPSAN team for a variety of other projects. Some novel applications include lifting air sampling equipment with the drone to capture air quality samples at a variety of altitudes and locations to better understand disease vectoring. Further, the user interface developed in this project could be extended to map and visualize much of the study data, including markers for compounds, latrines, water sources, etc. In addition, a machine vision algorithm could be implemented to allow for automated compound identification and measurement that would require no user input.

## 16. References


[1] Our Image Processing Services. (n.d.). Retrieved February 9, 2016, from http://www.satimagingcorp.com/services/image-processing/

[2] Standard Imagery. (n.d.). Retrieved February 9, 2016, from https://www.digitalglobe.com/products/standard-imagery#resource-section

[3] EBee. (n.d.). Retrieved February 9, 2016, from https://www.sensefly.com/drones/ebee.html

[4] Phantom 3 Professional. (n.d.). Retrieved February 9, 2016, from http://www.dji.com/product/phantom-3-pro

[5] Typhoon 4K. (n.d.). Retrieved February 9, 2016, from https://www.yuneec.com/products/aerialuav/q500_4k

[6] https://www.law.cornell.edu/uscode/text/49/40102

[7] https://www.law.cornell.edu/uscode/text/49/44102

[8] https://uavsystemsinternational.com/drone-laws-by-country/mozambique-drone-laws/

[9] https://uavsystemsinternational.com/drone-laws-by-country/nicaragua-drone-laws/

# Appendices

## Appendix A: Extra Tables

**Table 4. Mass Budget for Imaging Configuration.**

| Part | Mass (g) | Quantity | Mass Total (g) | Max Payload (g) | Payload Remaining (g) |
|---|---|---|---|---|---|
| Iris + | - | 1 | - | - | - |
| 3DR Pixhawk | - | 1 | - | - | - |
| Battery | - | 1 | - | - | - |
| TOTAL | 1282 | 1 | 1282 | - | - |
| GoPro No Housing | 89 | 1 | 89 | 400 | 311 |
| Raspberry Pi 2 B | 40 | 1 | 40 | 400 | 271 |
| Tarot Gimbal | 200 | 1 | 200 | 400 | 71 |
| | | | Mass Total (g) | Max Payload (g) | Payload Remaining (g) |
| | | | 1611 | 400 | 71 |

**Table 5. Power Budget.**

| Part | Quantity | Power (W) |
|---|---|---|
| Iris + | 1 | 166.80 |
| Tarot Gimbal | 1 | 6.00 |
| Pump | 1 | 1.48 |
| GoPro Hero 4 Black | 1 | 3.31 |
| Raspberry Pi 2 B | 1 | 2.50 |
| | | |
| | Max Power (W) | Power (W) |
| | 489.60 | 180.21 |
| Power Remaining (W) | 309.39 | |

**Table 6. Data Budget.**

| Part | Quantity | Frequency |
|---|---|---|
| **IRIS+:** | | |
| GPS | 1 | 5 Hz |
| Telemetry radio | 1 | 915 MHz or 433 MHz |
| 3DR Pixhawk | 1 | 168 MHz |
| **Gimbal:** | | |
| Support for remote devices | 1 | PPM/ PCM/2.4G |
| Control | 1 | 2000 Hz |
| Motor drive | 1 | 20 kHz |
| **Raspberry Pi 2 B** | | |
| Processor | 1 | 2GHz |
| Infrared (IR) receiver | 1 | 37.9 KHz |

**Table 7. Specification Sheet.**

| Drone | Specification | | Source |
|---|---|---|---|
| | Horizontal travel | 2 km | BWG |
| | Non-hover flight time | ~15 min | Testing |
| | Hover time | ~15.5 min | Testing |
| | Expenses | ~$1800 | BWG |
| | 100% of trained staff able to use without error | | BWG |
| | Communications frequency | 915 MHz (USA) 433 MHz (Rest of World) | 3DR |
| | Battery | 5100 mAh (3S) | 3DR |
| | Load capacity | 400 g | 3DR |
| | Life | >= 100 flights | BWG |
| | Mass | < 2 kg | 3DR |
| | Flight height | < 200 ft., > 15 ft. | Maputo Government |
| | Radio range | >= 1 km | 3DR |
| | Stability in winds | >= 7 m/s | Testing |
| | Automatic come home | | FAA |
| | Battery swap time | <= 1 min | MAPSAN Team |
| | Tunable controller gains | | BWG |
| | Horizontal speed | 11 m/s | Testing |
| | Propeller length | 0.22 m | Measured |
| | Drone Height w/ long legs | 0.37 m | Measured |
| | Drone Length w/o propellers | 0.26 m | Measured |
| | Drone Length w/ propellers | 0.49 m | Measured |
| | Drone Width w/o propellers | 0.41 m | Measured |
| | Drone Width w/ propellers | 0.61 m | Measured |
| **Camera** | **Specification** | | **Source** |
| | Field of view | 70 m | BWG |
| | Size of identifiable features | >= 8 in. | BWG |
| | Linear resolution | 3000 pixels | GoPro |
| | Battery life | 22 min | Testing |
| | Resolution | 12 MP | GoPro |
| | Focal Length | 14 mm | BWG |
| | Mass | <= 200 g | GoPro |
| | Wi-Fi enabled w/ password | | MAPSAN Team |
| | Internal memory swap time | <= 1 min | MAPSAN Team |

**Table 7. Specification Sheet (Cont.)**

| **Electronics** | **Specification** | | **Source** |
|---|---|---|---|
| | Boot time | < 5s | BWG |
| | Geotag accuracy | <= 1 m | BWG |
| | Computation time | < 5s | BWG |
| | Failsafe Features | | BWG |
| | Sensor Bits | >= 10 Bits | BWG |
| | Path Planning Efficiency | 80% | BWG |
| | Allow ground-based control override | | FAA |
| | Minimum functional battery charge | 15% | Testing |
| | Battery charge time | <= 90 min | 3DR |
| | Open-source firmware | | BWG |
| | Accurate drone simulation software | | BWG |
| | OS runs off of microSD card | | BWG |
| | Powered by drone battery | | BWG |
| | Battery charger voltage | 12 V | HobbyKing |
| | Battery voltage | 11.1 V | 3DR |
| | Battery charger amperage | 4.5A per battery | HobbyKing |
| **Gimbal** | **Specification** | | **Source** |
| | Min angular rate | 1500 °/sec | Tarot |
| | Control accuracy | 0.5 ° | Tarot |
| | Control precision | 0.5 ° | Tarot |
| | Control angle range | -45 ° +45 ° (roll) and (tilt) | Tarot |
| | Mass | <= 200 g | Tarot |
| | Manual pitch angle control | | Tarot |
| **System** | Specification | | |
| | Individual flight time | 12 min | BWG |
| | Total mission time | ~6 hours | BWG |
| | Heartbeat timeout | 30 sec | BWG |
| | Total number of batteries | 7 batteries | MAPSAN Team |
| | Batteries chargeable at once | 5 batteries | HobbyKing |

## Appendix B: Extra Figures

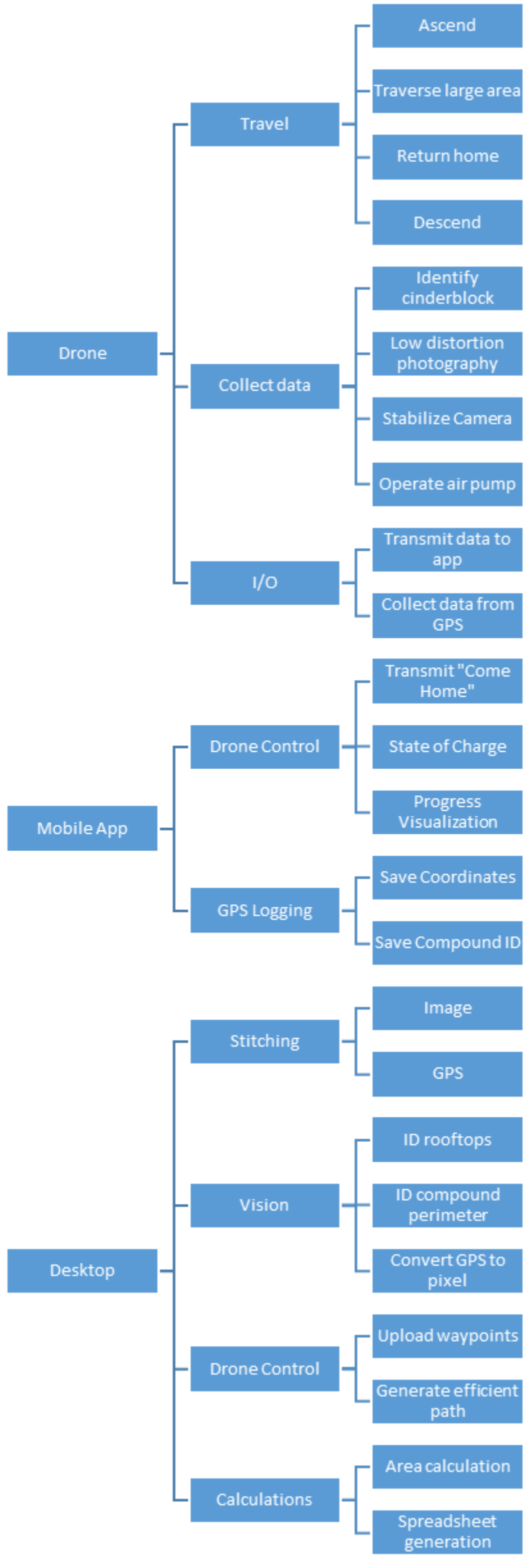



**Figure 11. Complete Functional Breakdown.**

## Appendix C: Tests

GoPro Hero4 Camera Testing:

The pattern of alternating green and white 8 in. squares seen in Figure 12 was made to simulate the 8 in. features the final imaging system would need to be able to detect.

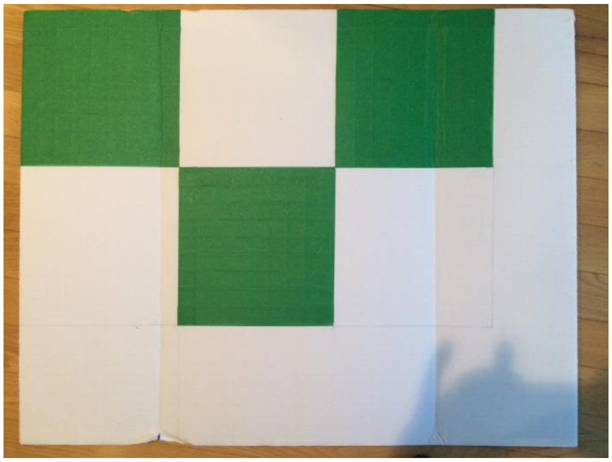


**Figure 12. Testing Pattern.**

The minimum threshold for a successful test was that each square must appear as at least 1 pixel from 200 ft away. An additional criteria for complete success was that the camera must be moving at a relative velocity of 20 mph compared to the testing pattern. To perform the static test, the pattern was held perpendicular to the ground 200 ft across flat terrain from the GoPro. A picture was taken and the close up of the testing pattern in that picture is seen in Figure 13 which clearly shows multiple pixels for each square and even some of the color of the pattern.

**Figure 13. Close Up of Testing Pattern Image from 200 ft while Static.**

The moving GoPro testing was performed by holding the same testing pattern perpendicular to the ground and facing it towards a road 200 ft away. The GoPro was held by a person in a car and when the car drove past the pattern at 25mph a photo was taken through an open car window. The close up of the testing pattern in that photo is seen in Figure 14 which shows some slight rolling shutter at the edges but still had multiple pixels per square on the pattern.

**Figure 14. Close Up of Testing Pattern Image from 200 ft while Moving at 25 mph.**

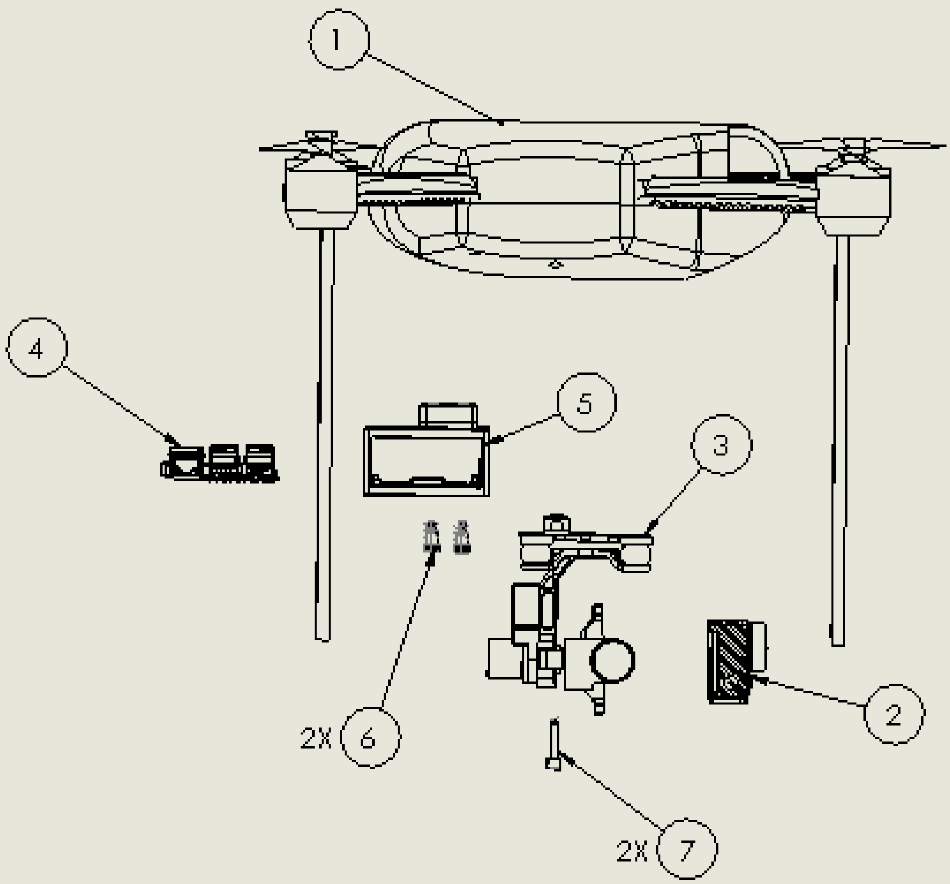
The conclusion drawn from the camera testing is that the GoPro Hero \4 has a high enough linear resolution to satisfy the requirements of the imaging system because it can clearly show objects that are 8 in. wide from 200 ft away. Testing may be performed again with colors that contrast less than green and white, such as gray and white, to see if the camera performance is affected.

**Appendix D: Budget**

No additional money was spent beyond the hardware found on our BOM. All fasteners and spare electronics were provided with the purchase of the equipment from 3DR, HobbyKing, or Amazon. As a result out of the starting $6000 budget only $1787.16 was spent and that is directly funded by the Brown Water Group who made purchases directly on behalf of our senior design team.

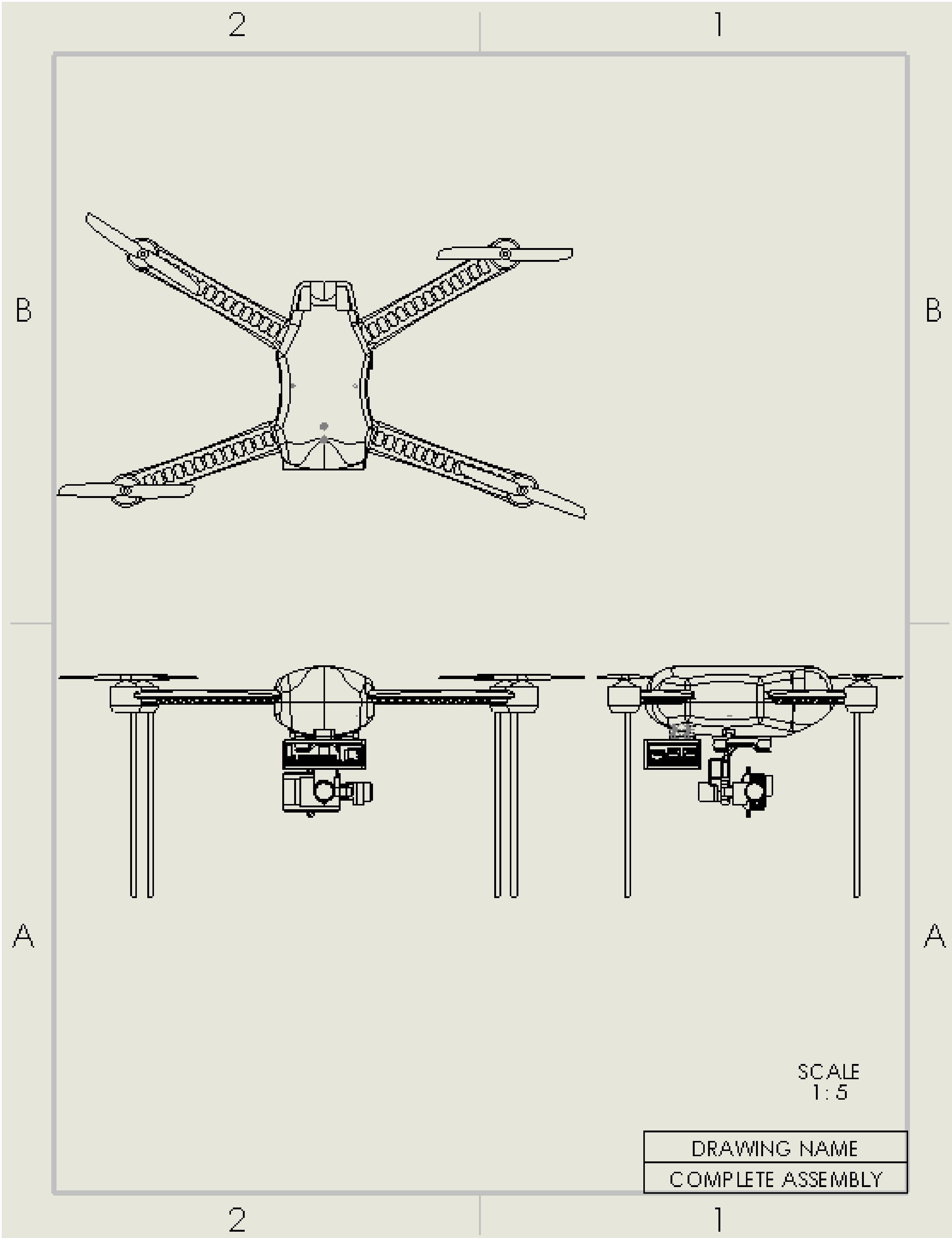
## Appendix E: Fabrication Package

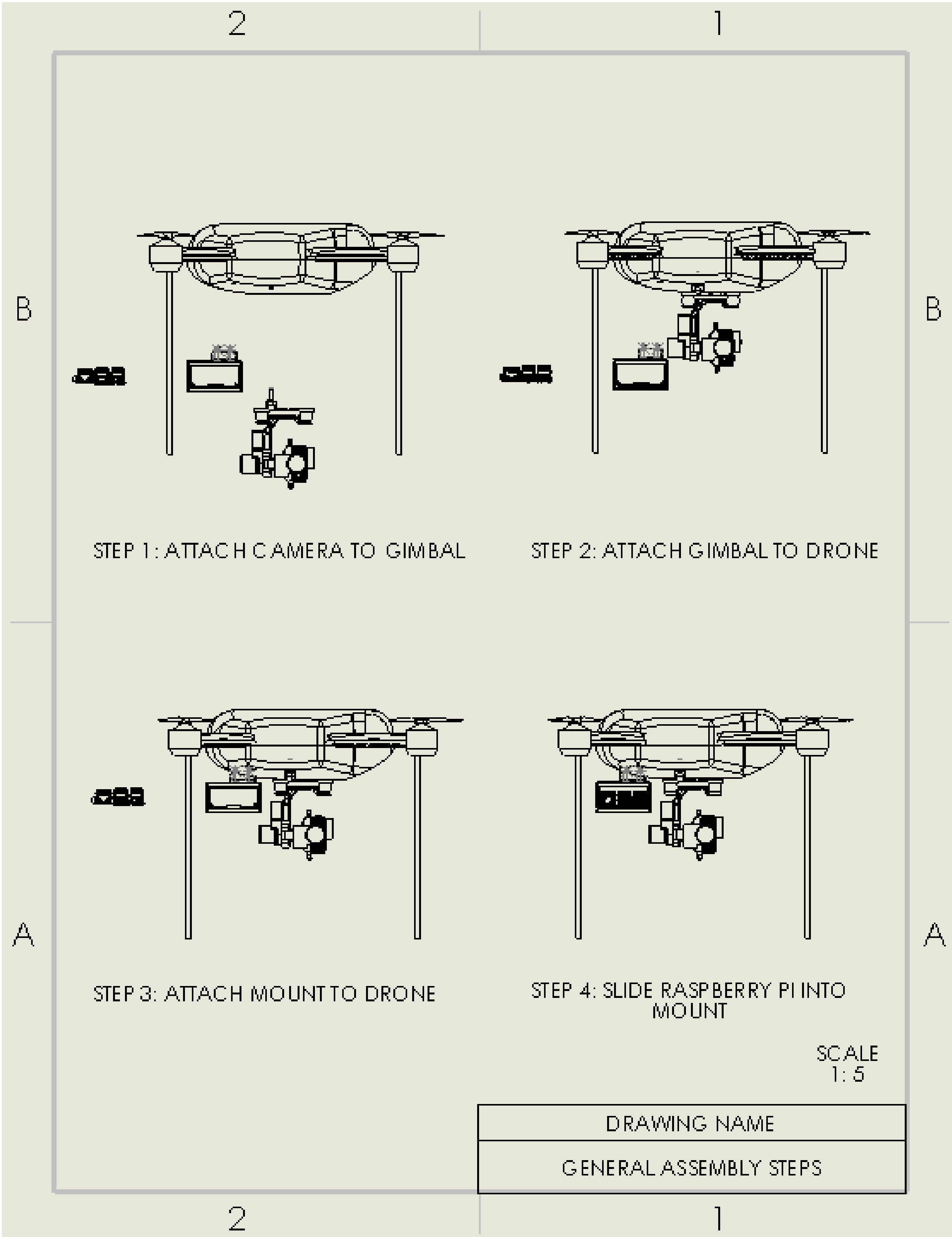
| ITEM NO. | PART NUMBER | DESCRIPTION | QTY. |
|---|---|---|---|
| 1 | 3DR IRIS+ | DRONE | 1 |
| 2 | GOPRO HERO4 | CAMERA | 1 |
| 3 | TAROT T-2D | GIMBAL | 1 |
| 4 | RASPBERRY PI | CONTROLLER | 1 |
| 5 | PI MOUNT | ADAPTER | 1 |
| 6 | SHOULDER SCREW | 8-32X3/16" DIAX1/4" LENGTH | 2 |
| 7 | SOCKET CAP SCREW | 8-32X.875" | 2 |

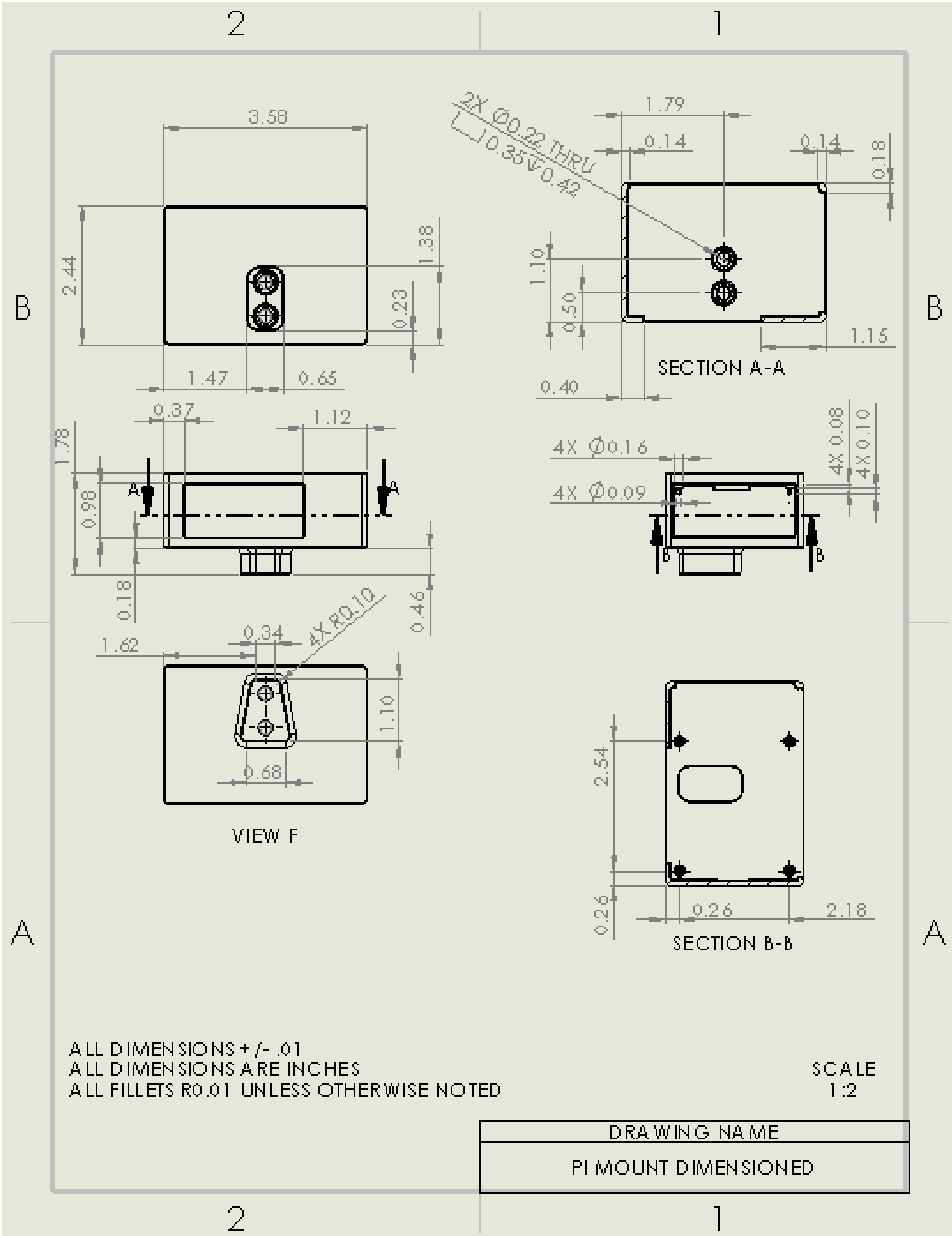



SCALE 1:3

| DRAWING NAME |
|---|
| EXPLODED COMPLETE ASSEMBLY |


2
1
B
A
SCALE
1 : 5
DRAWING NAME
COMPLETE ASSEMBLY


2
1
B
A
STEP 1: ATTACH CAMERA TO GIMBAL
STEP 2: ATTACH GIMBAL TO DRONE
STEP 3: ATTACH MOUNT TO DRONE
STEP 4: SLIDE RASPBERRY PI INTO MOUNT
SCALE 1:5
DRAWING NAME
GENERAL ASSEMBLY STEPS



3.58
2.44
1.38
0.23
1.47
0.65
2X Ø0.22 THRU
0.35 ∇ 0.42
1.79
0.14
0.14
0.18
1.10
0.50
1.15
SECTION A-A
0.40
0.37
1.12
1.78
0.98
0.18
0.46
4X Ø0.16
4X Ø0.09
4X 0.08
4X 0.10
1.62
0.34
4X R0.10
1.10
0.68
VIEW F
2.54
0.26
0.26
2.18
SECTION B-B
ALL DIMENSIONS +/- .01
ALL DIMENSIONS ARE INCHES
ALL FILLETS R0.01 UNLESS OTHERWISE NOTED
SCALE
1:2
DRAWING NAME
PI MOUNT DIMENSIONED