Autonomous Aerial Assistance for Search and Rescue

CRITICAL DESIGN REVIEW

Team F Rescue Rangers:
Karthik Ramachandran
Sumit Saxena
Juncheng Zhang
Xiaoyang Liu

Sponsor:
Near Earth Autonomy
Advisor: Paul Bartlett

Mentors:
Dr. John Dolan
Dr. Dimitrios Apostolopoulos

December 15, 2016

Abstract

This report summarizes our progress on the development of an autonomous aerial system, which can be used to assist search and rescue operations by analyzing the likely location of human beings through sensor data and dropping the rescue package there.

For start-off, we present an overall description of the project and the use case followed by the system-level requirements and performances. Then, the functional and cyber-physical architectures are shown to describe how our system will work and how it is designed.

Next, the targeted system requirement is discussed, as well as the progress achieved to meet these requirements in three subsystems, including autonomous flight system, sound detection and package drop system, and signature detection and analysis system. Furthermore, the current status for each of the above subsystems is detailed.

After that, the project management methods that we used for planning and tracking our progress are included. The final part of the report consists of the conclusions and references.

Table of Contents

1. Project description

1.1 Motivation

A typical search and rescue mission has very stringent requirements on time and the operating environment. This makes direct human involvement in the operation difficult and expensive, and has led to the use of automated vehicles to conduct the first wave of search. In such hazardous operations, where little information is available about the environment, aerial vehicles have a unique advantage of being able to quickly cover ground and gain an overview of the situation.

However, most of the existing approaches to SAR(Search and Rescue) using aerial vehicles currently rely heavily on teleoperated drones with minimal autonomy, which increase the risk for the rescue team and the cost of SAR operations. Apart from the huge cost, current approaches also impose strict piloting requirements on the operator, which limits the pervasiveness with which such technologies can be deployed. In addition, the capabilities of a teleoperated mission are extremely limited to certain categories of local terrain that always allow a link between the vehicle and the operator. All these issues in addition to the fact that there are roughly 11 SAR incidents each day at an average cost of $895 per operation[1], underscore the need for building systems that are as autonomous as possible.

1.2 Objectives

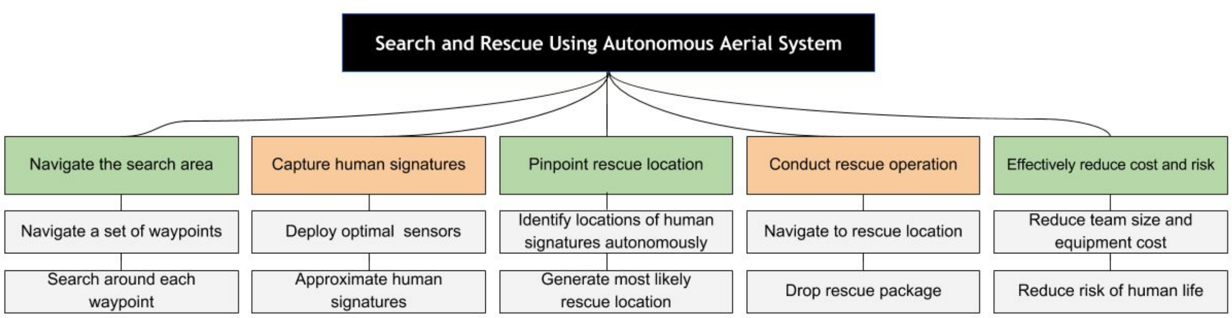


Fig 1.1 Objective Tree

As part of our quest to solve this challenging problem, we propose an autonomous aerial system for search and rescue, in order to effectively reduce rescue team size, equipment cost, as well as risk to human life. A programmable hexrotor will be used to autonomously navigate the search area and collect data through sensors. A system will be built which can then analyze the data to detect human signatures and pinpoint the most beneficial location to conduct a rescue operation efficiently and reliably.

Due to the time constraint and limited team size, a readily available drone with inbuilt maneuvering capabilities will be provided by our sponsor, Near Earth Autonomy. Due to this, most of the efforts will be focused on planning strategies for generating and searching around waypoints, and using machine learning and vision algorithms for capturing human signatures and pinpointing rescue locations. Also, a mechanical actuator system will be designed to hold and drop a rescue packet when conducting rescue operations.

2. Use case

Jamie is the Team coordinator at the Yosemite Search and Rescue (YOSAR)[6] team. His team is mainly responsible for conducting SAR activities in the Tuolumne Meadows region. He is proud that his team is one of the most well-oiled SAR machines in the world. But that comes at a cost. His team employs only well-trained individuals with strong alpine skills and also makes use of helicopters to search in large areas.

YOSAR team conducts rescue operations on a “pay by mission” basis and the rescue operations involve thorough searches in the meadows, forests and mountains. The operations are expensive and each rescuer in the team has to be paid $ 23-34 per hour. Given the large area to traverse, most operations require at least a couple of helicopters, if not more. Moreover, finding good rescuers is a challenge in itself due to the stringent requirements posed by the task. Browsing over the internet, he stumbles on a video showcasing the “Rescue Rangers” drone with the ability to search for human beings in relatively un-occluded environments. He immediately sees the value in using it for his missions as a cheaper option with much a faster response and decides to own it.

When the package arrives, he is excited to open it and finds the set-up pretty simple – assembling the parts of the drone – not any more difficult than assembling a furniture these days. He installs a mandatory software on his laptop to help give the drone inputs and goes through a few tutorials provided to help operate the system. It takes him less than a couple hours to set everything up. Happy with his new gizmo, he wraps up the day unaware of the situation that awaits him the next day.

At 6 in the morning, while nicely tucked in his bed, he receives an emergency SAR SOS from the Yosemite Emergency Communications Center. Two hikers had wandered off the trail and had gone missing for the past two hours. The agency managed to get their approximate location two hours ago and there was also a mention of one of them being severely bruised as well. Jamie immediately sends an alarm to gather the team and prepare for rescue. While preparing for his mission, he glances at the drone he had set up the previous night and wonders if this is the right opportunity to put the drone to test , since he also recollects that the area around the location provided is not covered with trees. “Half an hour for assembling the team should be enough to test the drone”, he thinks to himself.

He fires up his laptop and switches the drone’s power ON in the lawn outside. On the software suite, he is able to see Google maps of the nearby area with GPS coordinates. He selects the option “Search and map” and feeds in the three GPS coordinates (Waypoints W1, W2 and W3 in Figure 2.1) around the provided location where he reckons the hikers might be. The drone takes off (from location S in Figure 2.1) and he resumes his preparation for the operation. 20 minutes later, his team is assembled with all the equipment and he briefs them on the mission and they are all set to leave. Just as they can start, the drone comes back to the base. Jamie connects the drone to the laptop with a USB cable and data is transferred by the software in 5 minutes. In another 3 minutes, the software pops up with some relevant pictures found and their locations on the map. Jamie is awestruck with the capability. He could easily see the two hikers in one of the pictures. He immediately communicates the location to his team. Figuring that the team still might take some time to reach the location, he attaches a first aid package to the drone and switches ON its power. He selects the option "Drop package" on the software and specifies the chosen location. The drone takes off and comes back in 10 minutes. After another 30 minutes, the team brings the two hikers via a helicopter to the base to treat and tells Jamie that the victim had already got first aid before the team could find them. ! MISSION ACCOMPLISHED !

Figure 2.1 Illustration of how Rescue Rangers aerial system works (waypoints, terrain and visible distances only for illustrative purpose only) ***[Background image sourced from:[7]]***

3. System-level requirements

3.1. Mandatory requirements:

Mandatory requirements were arrived at after exhaustive research on the needs of search and rescue missions, numerous discussions with the sponsors and carefully considering what is achievable in the given timeframe. They were further modified based on feedback received on the Conceptual Design Review document.

Table 3.1 Mandatory Functional and Performance Requirements

Functional Requirements The system shall:	**Performance Requirements** The system will:
M.F.1. Autonomously navigate through a set of provided locations of interest	**M.P.1.** Accurately reach the locations of interest with a tolerance of +-5m
M.F.2. Complete the search within limited time	**M.P.2.** Complete one iteration of search in an un-occluded operating area of 200m x 200m in <25 minutes
M.F.3. Explore the surroundings around each location of interest	**M.P.3.** Attain up to 80% coverage of the desired local search areas around each location of interest
M.F.4. Collect perceptual data while navigating	**M.P.4.** Collect perceptual data limited to 3 types - IR radiation, visual imagery, and sound
M.F.5. Process the data to identify human signatures	**M.P.5.** Identify at least 67% of the locations with human signatures
M.F.6. Analyze the identified signatures to estimate human location	**M.P.6.** Estimate potential human signature location with +-5m tolerance
M.F.7. Navigate to the rescue location carrying the rescue package	**M.P.7.** Carry a rescue package weighing 100g
M.F.8. Drop the rescue package	**M.P.8.** Drop the package at the rescue location with a tolerance of +-5m

Table 3.2 Mandatory Non Functional Requirements

Mandatory Non-Functional Requirements The system will:
M.N.1. Reduce the search team size required to <=2
M.N.2. Reduce risk to human lives
M.N.3. Reduce equipment cost required

3.2. Desired requirements:

Table 3.3 Desired Functional and Performance Requirements

Functional Requirements The system shall:	**Performance Requirements** The system will:
D.F.1. Optimize initial path planning based on geography/terrain of the given search area	**D.P.1.** Reduce the initial navigation plan duration by at least 20%
D.F.2. Adaptively generate navigational waypoints during the flight based on the sensor data	**D.P.2.** Reduce the overall search and rescue duration by at least 20% in cases where the actual human location is far away from the initial waypoints provided

Table 3.3 Desired Non Functional Requirements

Non-Functional Requirements The system shall:
D.N.1. have an interactive GUI to make it operable by an untrained human being ● Receive inputs from the user on a map ● Show live navigation on a map.

4. Functional architecture

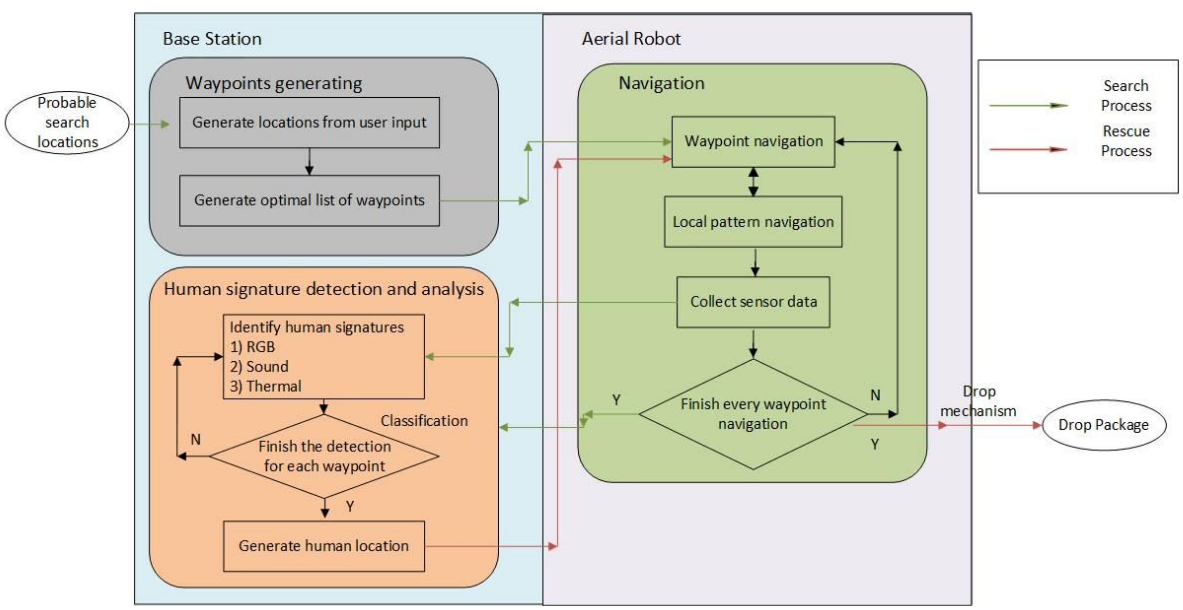


Figure 4.1 Functional Architecture

The architecture is described below as a sequence of functions:

1. A mission begins with a rescue agent providing a list of geographic zones where the system should focus the search on. This information is then translated to GPS coordinates by the system and an optimal navigation path is generated as a list of ordered waypoints.
2. The aerial system navigates the list of waypoints and initiates a localized navigation pattern at each of the waypoints. The localized pattern is specifically designed to enable capturing reliable sensor data at each waypoint.
3. Once the waypoints are navigated and sensor data is collected, the drone returns to the ground station and initiates a data transfer.
4. Once the data is available, the ground station runs sophisticated algorithms to identify human signatures from the data and their precise locations.
5. The aerial system then navigates to the rescue location and drops a rescue packet as accurately as possible.

5. Cyber-physical architecture

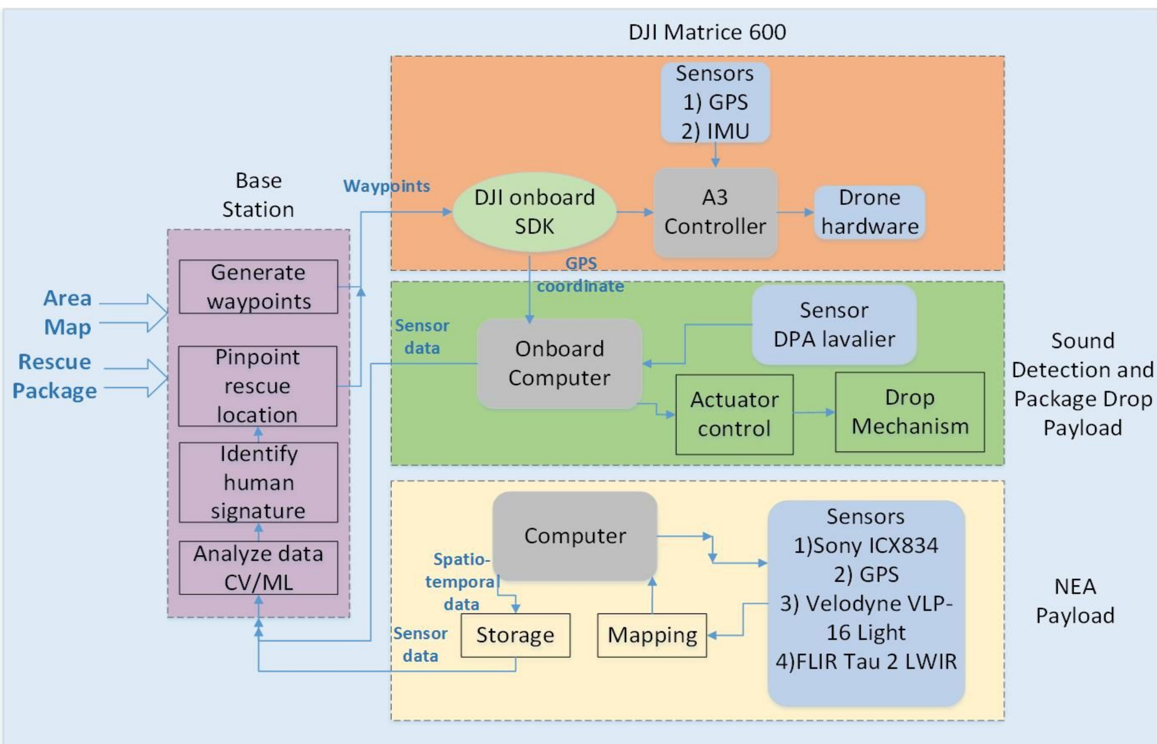


Figure 5.1 Cyber-physical Architecture

5.1 Autonomous Flight System

The autonomous flight system is based on DJI Matrice 600 platform. The GPS and IMU sensor embedded in Matrice 600 will primarily be used for navigation. The sensor data will be sent to both the flight controller and the onboard computer so that they can analyze and control the real-time flight of the drone.

There are two key components in this system: the Global Waypoint Generation component and the Local Pattern Navigation system. The Global Waypoint Generation component accepts fuzzy region information from a Local terrain expert and converts it into waypoints represented as GPS coordinates. It also generates an optimal ordering of the waypoints for the aerial system to navigate. The Local Pattern Navigation system generates a localized pattern at each global waypoint so that the drone can maneuver in a way that enables the system to collect quality sensor data with both resolutions as well as coverage.

5.2 Sound Detection and Package Drop System

The Sound Detection and Package Drop Payload System(SDPD) consists of the following components:

- **Drop Mechanism:** Rescue packet drop mechanism to be custom designed and fabricated for the drone.

- **Additional Sensors:** The system will also have an additional sound sensor (DPA lavalier) for detecting human sound which will be quantified by a certain magnitude and frequency.

- **Onboard computer:** Currently we use Raspberry Pi as the onboard computer. The onboard computer is responsible for collecting microphone information and transfer it to the base station once it flies back to the station. Also, this onboard computer is able to receive commands from DJI SDK and give the drop mechanism the instruction to drop packages.

5.3 Sensing System

The sensors on the NEA payload consist of RGB camera, GPS, Lidar and thermal sensor. The rationale behind using multiple types of sensors is so that the system can recognize different human signatures, and thus increase the possibility of finding humans. The Lidar will predominantly be used for precise altitude information while doing a rescue.

After doing the trade study, we decide to choose the types of sensors as follows:

- RGB camera: Sony ICS 834
- Lidar: Velodyne VLP-16 Light
- Thermal: FLIR- tau2- LWIR

5.4 Human Detection and Analysis System

The Signature Detection and Analysis system resides in the base station and is responsible for analyzing all the sensor data and detecting human signatures.

Once the signatures are available, the software will generate a ranked list of candidates which will be presented to the operator to pick the best candidate. Once the candidate is available, the payload map can be used to lookup the coordinates of the location which will then be used by the drone for the rescue mission. Finally, those coordinates will be transferred to DJI onboard SDK for the next rescuing flight.

6. Current System Status

The targeted system requirements for the fall semester are:

1. Autonomously navigate through a set of provided locations of interest,
2. Process the data to identify human signatures,
3. Navigate to the rescue location carrying the rescue package,
4. Drop the rescue package

In order to achieve these requirements, the subsystems that were worked on for the Fall semester are:

1. Autonomous Flight Subsystem,
2. Signature Detection Subsystem,
3. Sound Detection and Package Drop Subsystem.

Though some work was done for the sensing subsystem in terms of evaluating and finalizing the type and model of the sensors, most of the work is planned for the Spring semester. The Fall work accomplishments for each of the above subsystems is detailed in the rest of this section.

6.1 Autonomous Flight Subsystem

6.1.1 Descriptions

The autonomous flight subsystem comprises of the autonomous navigation and the waypoint generation system.Autonomous Navigation deals with being able to navigate the drone based on predefined set of waypoints provided as GPS coordinates. In this semester, the DJI mobile SDK was used to control the drone. Their app framework was extended and modified to add additional functionality to track and accept input GPS locations through a map. The app also provides functionality to set mission parameters like altitude and speed. A couple of screenshots from the app are shown below in Figure 6.1 and Figure 6.2.

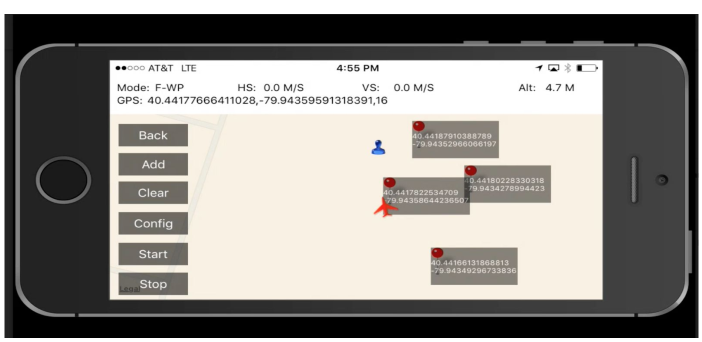


Figure 6.1. Screen for entering waypoints

Figure 6.2. Screen for entering mission parameters.

The purpose of the waypoint generation component is to generate a set of waypoints that ensures that the sensing subsystem is able to capture data with maximum coverage and resolution. To achieve this goal a spiraling pattern was decided upon as shown in Figure 6.3.

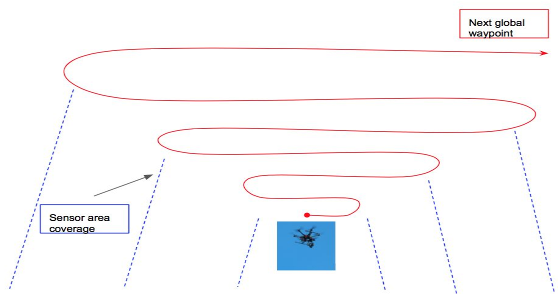


Figure 6.3. Spiral pattern for maximum sensor coverage and resolution.

In this semester, the waypoint generation component was implemented as an independent system and various tests were done by projecting the generated waypoints on a map and ensuring their accuracy. The system is implemented in such a way that the various parameters that govern the actual path of flight like height and width of the spiral, descent increments between highest and lowest points, the number of local waypoint increments between two global waypoints etc, are all configurable. This ensures that the system can be launched in a specific configuration based on the requirements of the mission. Figure 6.4 shows the result of one such path as a 3D plot.

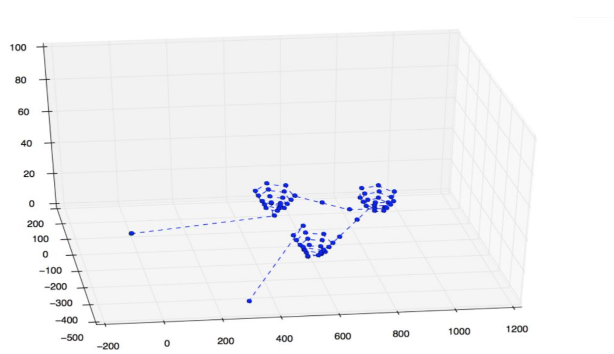


Figure 6.4. 3D plot of waypoints

6.1.2 Validation and testing

The autonomous navigation component was initially tested using the DJI simulator and then subsequent tests were done through outdoor flights. A sample simulator test screenshot is shown in Figure 6.5.

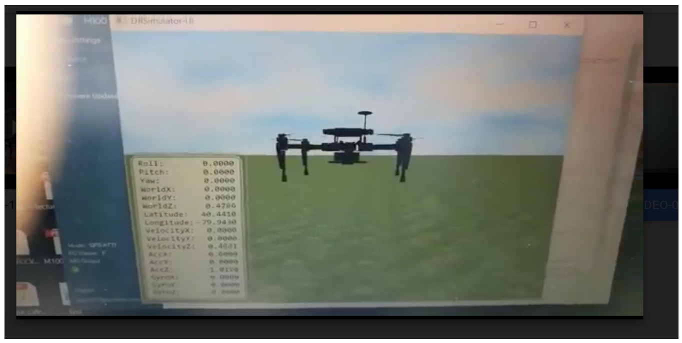


Figure 6.5. DJI Simulator test screenshot

For validating the GPS accuracy of the flight system, GPS from the two different sources were compared against each other. The GPS locations from the Drone's inbuilt GPS was compared and tallied with the GPS coordinates as reported by the smart phone. In addition, to understand any random errors within individual GPS sources, experiments were done to record GPS coordinates reported by each device for a given location and the standard deviations of the readings were noted. The two sources were found to be fairly accurate in terms of exhibiting any random errors. The one thing that was not tested extensively was the possibility of any static bias in the two sources. Though this is addressed to some extent by comparing and tallying the sources with each other, a more comprehensive test will help understand and address any static bias the systems might have.

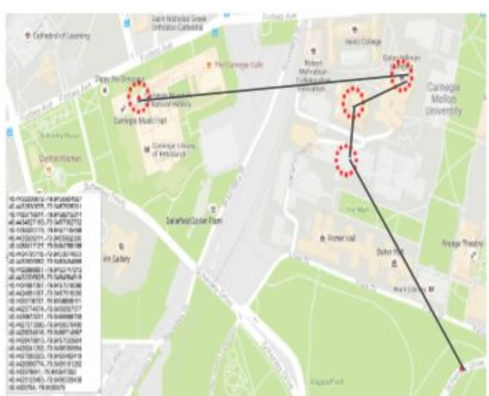


Figure 6.6. Waypoints on a map

The waypoint generation system was tested by plotting the actual waypoints on a map and ensuring that the path generated by the system indeed goes through each of the required waypoints as shown in Figure 6.6.

6.1.3 Performance evaluation

The Fall requirements for the Autonomous Flight System are shown in Table 6.1 below.

Table 6.1

Test A	***UAV Waypoint Navigation Test***	
Description	Validates the autonomous flight control and waypoint navigation capability of the UAV	
Location	Open 50m x 50m area with GPS access and normal wind conditions	
Equipment	UAV, Laptop for waypoint control	
Steps	**Step Description**	**Performance Measures**
A.1.	Place UAV on the ground. Feed GPS locations as waypoints	
A.2.	UAV takes off and goes to the first GPS location	Accuracy in reaching desired height (+-1m tolerance)
A.3.	UAV navigates from one waypoint to another	Accuracy in reaching the waypoints (+-5m tolerance)
A.4.	UAV returns to the starting location	Accuracy in reaching the starting location (+-5m tolerance)

The evaluation results against the above requirements are listed below:

- A.1 Implemented iOS App based GPS entry functionality
- A.2. Height provided = **5M.** Min height = **4.8M**, Max height = **5.3M**
- A.3 Accuracy: **<3M** from each waypoint
- A.4 Back to start location: **<3M**

As can be seen, all requirements for Fall validation were satisfied.

6.1.4 Conclusions

While developing against the DJI SDK, we found it to be extremely flexible and rich in terms of the API to control the drone. This made it easy to rapidly iterate on the iOS App and add additional functionality within a short period of time. The UX that was developed was intuitive and easy to use in terms of being able to enter GPS coordinates but needs some work to display exact distances between the drone and waypoints. In the next semester, the main focus will be to integrate the waypoint generation component with the flight navigation component and implement changes in the UX to make it easier to use.

6.2 Sound Detection and Package Drop (SDPD) Subsystem

Sound Detection and Package Drop (SDPD) subsystem, as the name suggests, consists of the payload we are developing, which will have components required for sound detection and package drop. It will be a single unit with its own power distribution circuit and would be designed to enable easy mounting on the sponsor's (Matrice 600) drone.

Currently, only two of its components have been developed - (1) Package drop Mechanism, and (2) Power distribution system.

6.2.1 Package Drop Mechanism

Description

To effectively fulfill the mandatory system functional requirements M.F.7 and M.F.8, our package drop mechanism is required to meet the following basic requirements:

The mechanism should:

- Be able to carry a 100g package
- Enable easy package attachment
- Have a good grip on the package throughout the flight
- Release the package easily without causing any damage

Keeping these requirements in mind, we designed a simple yet robust mechanism, shown in Figure 6.7. The mechanism is basically a slider-crank mechanism mounted on a 3-D printed ABS body, facilitating required motion of the slider. Various components of the mechanism are highlighted in Figure 6.7.

The mechanism consists of the following main components:

1. Servo motor: To actuate the mechanism
2. Servo motor attachment: acts as the crank
3. Connecting rod: 3-D printed (ABS); to connect crank to slider
4. Slider rod: made of wood to keep it lightweight while providing sufficient strength
5. Body: 3-D printed (ABS); to provide mounting for the servo motor, passage for the slider rod, and space for attaching a package to the mechanism

Control of the mechanism for FVE was implemented using a Raspberry Pi mounted on the drone. Raspberry Pi and its housing are shown in Figure 6.8. We created a server on Raspberry Pi and sent HTTP calls to it, with required servo angle as the parameter, using a phone. Raspberry Pi, in turn, commanded the servo motor to rotate by the specified angle to open/close the mechanism.

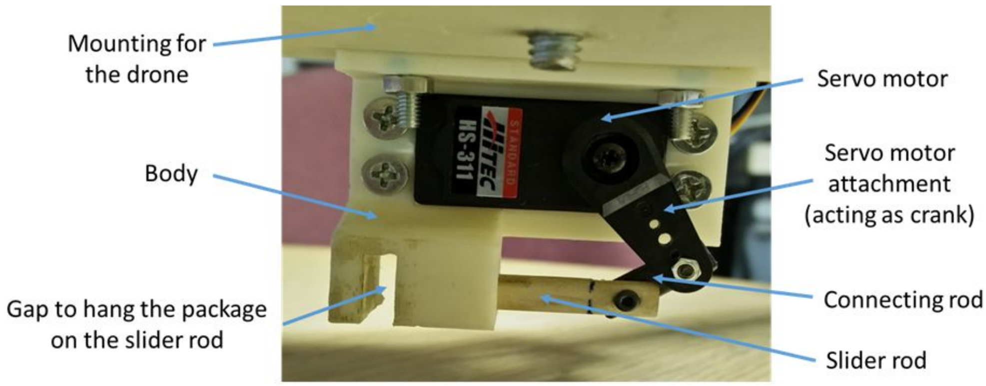


Figure 6.7. Package Drop Mechanism

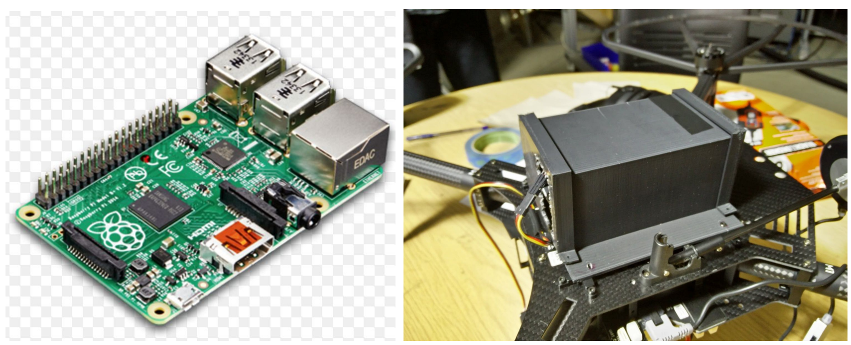


Figure 6.8. (a) Raspberry Pi used to command the mechanism
(b) Housing for Power supply and Raspberry Pi

Modeling and Testing

The mechanism was first modeled in SolidWorks to:

- To test feasibility
- To finalize exact design specifications of different components to be manufactured
- To estimate the angles the servo motor needs to rotate to open/close the mechanism

Figure 6.9 shows the SolidWorks model for the mechanism.

Rigorous testing was done to ensure that the mechanism worked, as required. A package weighing ~160 g and dimensions 10cm x 10 cm was made for this testing. Typically, two types of tests were done to ensure good performance:

1. **In-flight testing:** Package was attached to the drone and the drone was flown at speeds reaching up to 3 m/s. This test was conducted 5-6 times before FVE. This test not only ensured us of the grip of the mechanism but also of the stability of the flight with the package

2. **Attach/release testing:** Ease of package attachment and release was tested by performing this sequence 20-25 times.

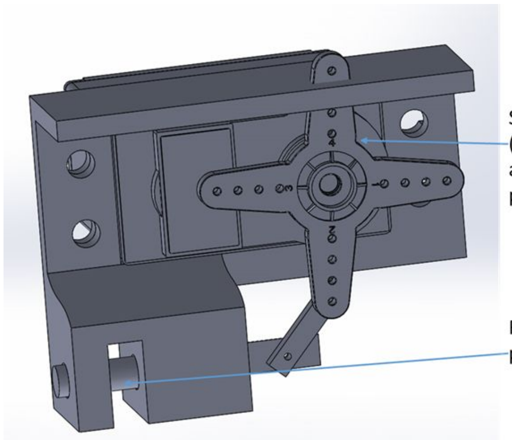


Figure 6.9. SolidWorks model for the Package Drop Mechanism

FVE Performance Evaluation

The following table describes the FVE test for the package drop mechanism and our mechanism's performance on the performance measures:

Table 6.2. Package Drop Mechanism: FVE performance evaluation

FVE Test C: Package drop mechanism prototype test				
Steps	**Step Description**	**Performance Measures**	**FVE**	**FVE - Encore**
C.1.	Validate package size and weight. Secure the package in the mechanism	Should be able to hold package of weight 100g, and size 10cmx10cm	**Successful (package weight ~160g)**	**Successful (package weight ~160g)**
C.2.	Subject the mechanism to accelerations in x, y and z directions manually	Should not lose grip of the package for drone velocities under 2 m/s	**Successful**	**Successful**
C.4.	Manually demonstrate the mechanism's ability to release the package	Should release the package safely without any damage, 3 times in a row	**Successful**	**Successful**

Our mechanism was successful on all the three performance measures.

Conclusions

The package drop mechanism worked very well for us and we did not face any issues with it. Following are some of its strengths we identified:

- Simple and easy to control: this prevents any complex issues
- Compact and lightweight: this enables easy handling and will make migration to Matrice 600 easy
- Grip on package not dependent on any electrical system, but rather on structural strength: this ensures we never lose grip on the package even if other systems fail

Despite these strengths, the fact that we hang the package through a string and don't secure it firmly in a place may be treated as a weakness for the system since it might affect the drone's dynamics. But given our requirements for the package weight and dimensions, we do not expect any effects on the drone's dynamics and this is what we have observed also, so far.

Further work:

Since finally, the mechanism has to be mounted on our sponsor's Matrice 600 drone, we need to design a mounting for the mechanism which can be easily attached to the drone.
Although there are no concerns with the mechanism's structural strength and load carrying capacity, since this mechanism is built with ABS and wood, we might investigate the need for metal fabrication.

6.2.2 Power Distribution System

Description

Our power distribution system needs to provide the electric energy both to the payload from our sponsor, Near Earth Autonomy, and to our own designed payload for sound detection and package drop. The input is from the battery of the drone, whose voltage output is 22.2V. More specifically, the battery model is TB475, and its type is LiPo 65, with 4500mAh capacity.

The main power need of our aerial system is the payload provided by Near Earth Autonomy, including an RGB Camera, a thermal camera, a LiDAR and an onboard computer. The power distribution system for those sensors is also inside the payload, and the team only needs to design the power distribution system for our SDPD system. Specifically, our system is designed to power an onboard computer for storing data of the sound sensor and communicating with the DJI flight controller, as well as a DC motor to drop the rescue package.

The conceptual design for the power distribution system is shown below:

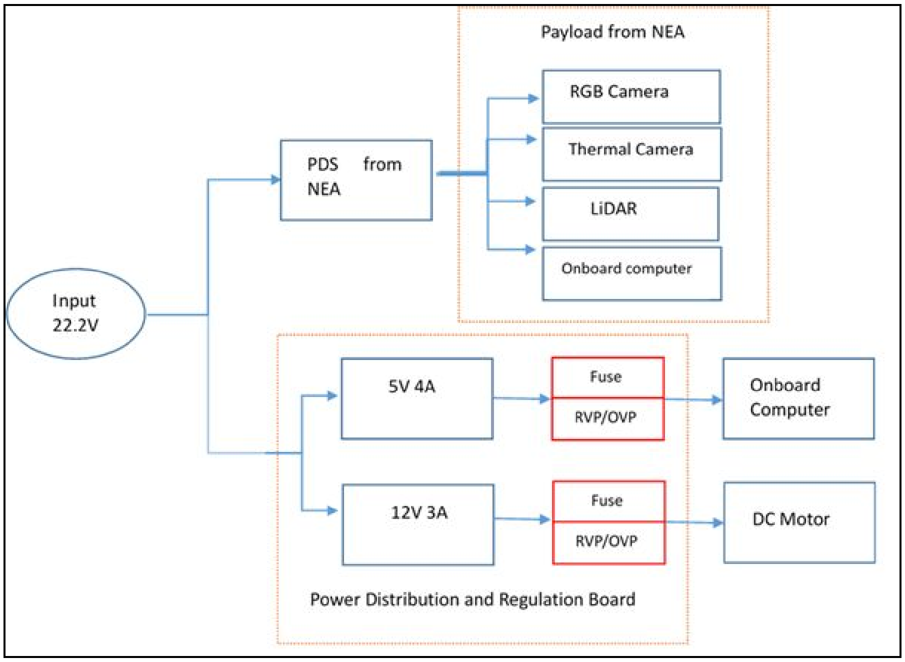


Figure 6.10. Conceptual Design for PDB

The schematic design and the layout design of our power distribution system are shown respectively in Figure 6.11 and Figure 6.12.

To make sure that our PDB is printable, the manufacturing files are created and checked by www.freedfm.com.

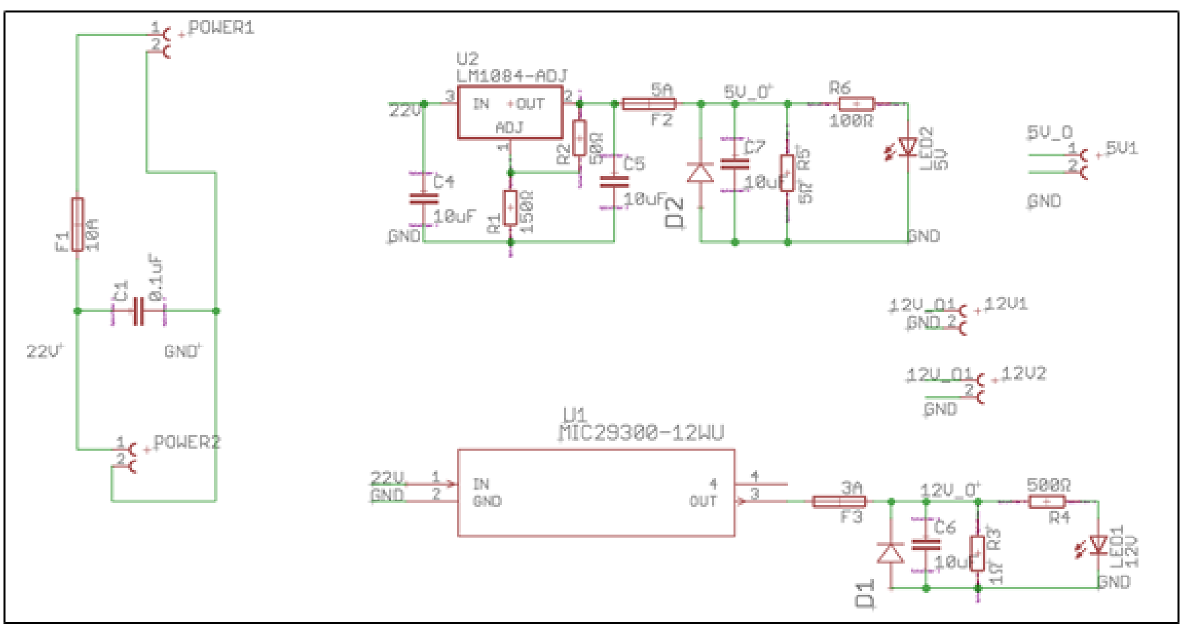


Figure 6.11. Schematic Design for PDB

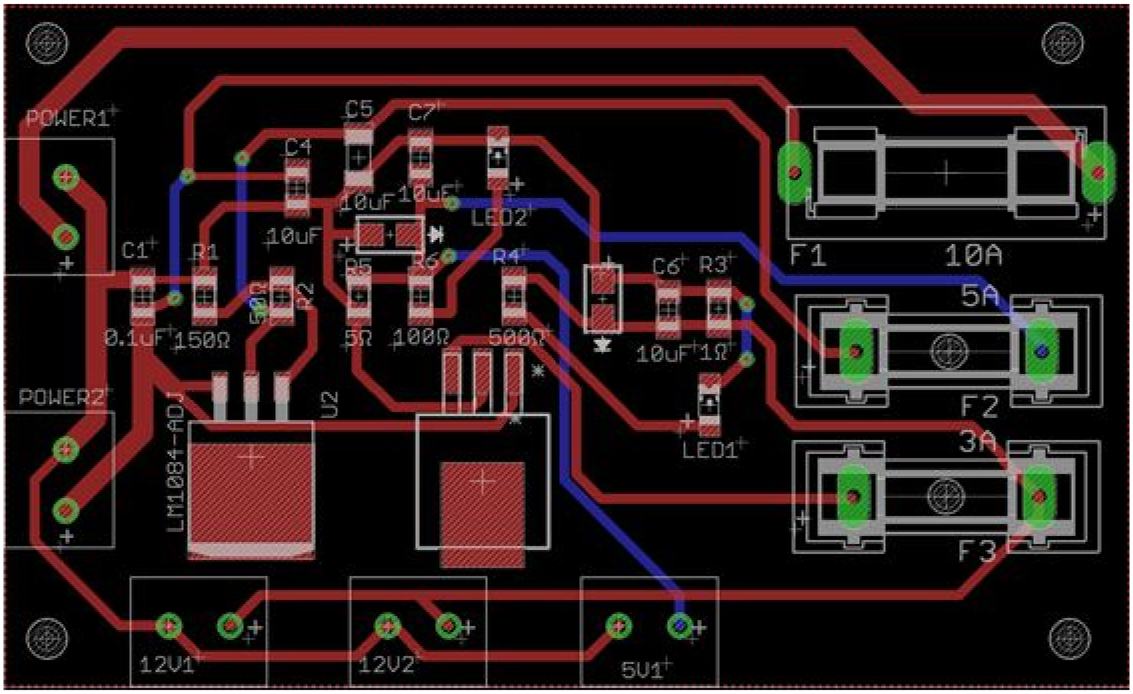


Figure 6.12. Layout Design for PDB

Performance Evaluation

The fabricated power distribution board is shown in Figure 6.13.

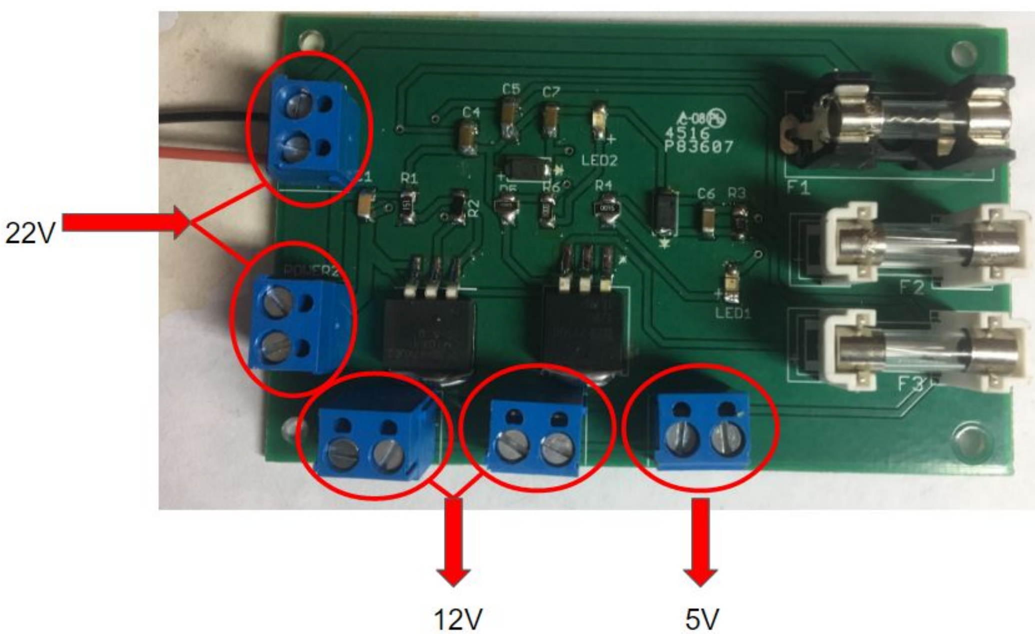


Figure 6.13. Fabricated Power Distribution Board

We connected our PDB with the battery of Matrice 100, and measured the voltage of each output. It turned out that both the 5V output and the 12V output worked perfectly.

Strength and Weakness

The power distribution board we designed have several advantages:

1. Is able to protect the circuit from misoperation
2. Is compact and lightweight
3. Have additional connectors for backup use

However, there is still a weakness. If the battery is fully charged, the output voltage of the battery increases up to 25V, which is quite close to the maximum input voltage of the 5V regulator. In order to mitigate this risk, we need to reduce the input voltage of the regulator by adding a step-down transformer before it. In addition, we can change the type of the regulator so that it has a higher maximum input voltage.

6.3 Signature Detection and Analysis

6.3.1 Description

Detecting humans in images is a challenging task owing to their variable appearances and wide range of poses. Our motivation behind developing an algorithm to detect the presence of human beings is that it can be used in various scenarios. More specifically, it can be applied in autonomous search and rescue operations through aerial platforms, which can effectively reduce the equipment cost and risks of injuries of humans. In this project, we firstly implemented several different methods for capturing human candidates, the namely region of interests(ROIs). Then we utilized HOG to extract features and classify whether there are human beings inside the ROIs based on linear support vector machine(SVM)[4]. Plus, we visualized our classification results to further illustrate the cardinal principles. At last, we applied our method to series of images and computed the accuracy of the overall performance.

The overall performance of the algorithm is shown in Figure 6.14. As you can see, the bounding boxes in frames are the ROIs, and red boxes represent for those who are classified as ROIs without humans while green ones are human candidates after classification.

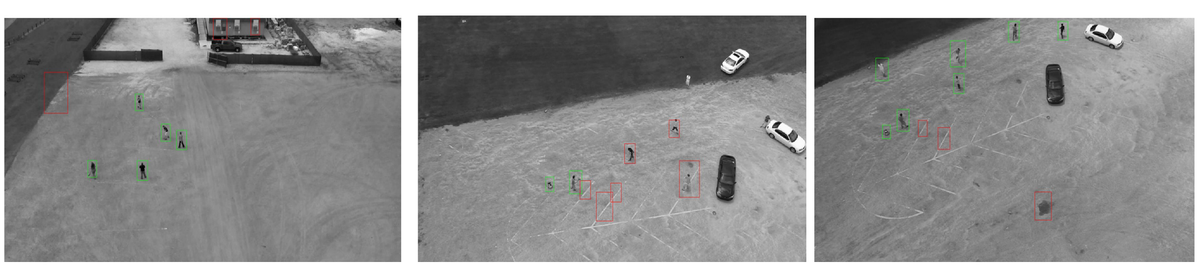


Figure 6.14 Overall performance of human signature detection

6.3.2 Modeling and Analysis

Modeling of Edge Detection and Blob Detection

The first row of Figure 6.15 shows the modeling of Blob Detection, and the second row shows the modeling of Edge Detection.

Blob detection

- Find interest points by applying Gaussian pyramids and difference of Gaussian(row 1, col 1)
- Use Dilate and Erode operations to connect adjacent interest points.(row 1, col 2)
- Rule out several improbable candidates based on the shape of connected pixels.(row 1, col 3)

Edge detection

- Use Sobel method[3] for edge detection(row 2, col 1)
- Use Dilate and Erode operations to fill the inner areas of edges and find connected components which exceed a minimum number of pixels (row 2, col 2)
- Rule out several improbable candidates based on the shape of connected pixels.(row 2, col 3)

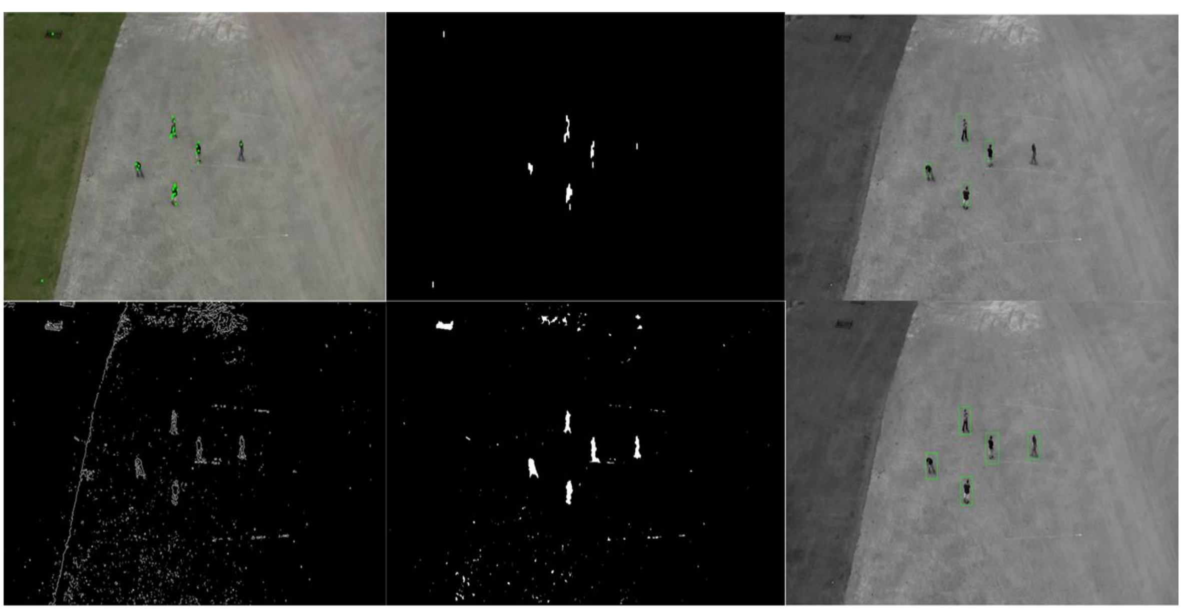


Figure 6.15 Description of Edge and Blob detection

Modeling of Background Subtraction

Figure 6.16 shows the modeling of Background Subtraction.

Background Subtraction Strategy:

- Input a series of video frames(a).
- Segment moving objects from the background by using Vibe[2] (b).
- Outline the potential human candidates based on the area of connected foreground pixels in segmentation masks(c)

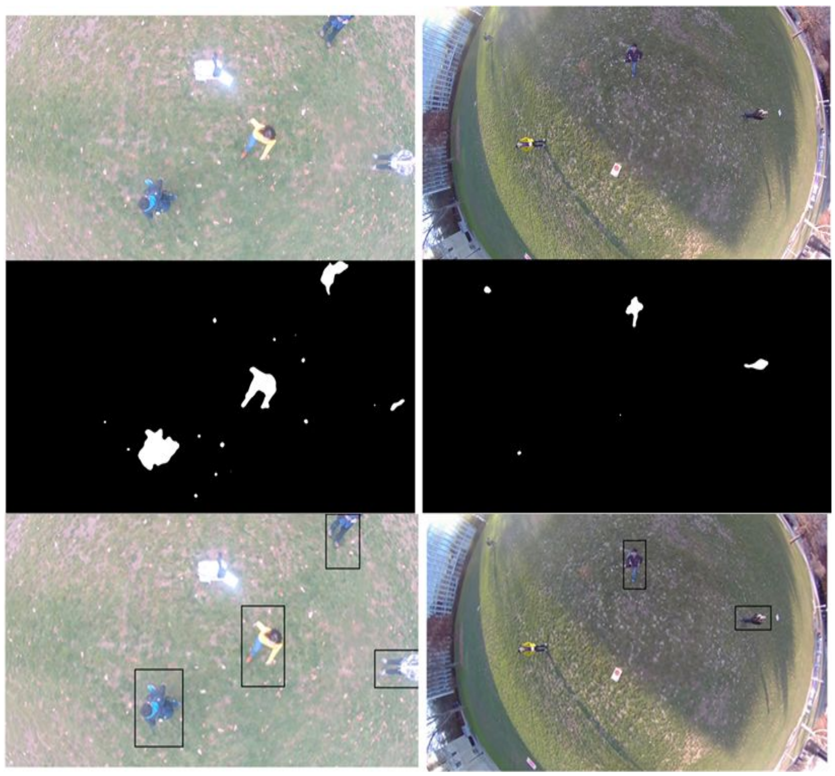


Figure 6.16 Description of Background Subtraction Strategy

Analysis of HoG + SVM for classification

HoG+SVM are very efficient classifying pedestrians[5]. However, we cannot confirm the feasibility of using this method before the analysis on aerial samples by using this algorithm. Therefore, we collect 299 pictures containing humans and 372 pictures without humans, and use them as positives and negatives to train the HoG+SVM classifier. It is very important to make all the pictures in the training set have the same size as the ROIs we will be used in the test set. Then, we used ROIs which are captured by applying two algorithm mentioned above as our test set. Before doing the final test, we labeled the ROIs with humans as positives and those without humans as negatives for the reason that it's easier to calculate the accuracy by comparing the test labels and predicted labels.

Data Information:

- Training set:
  299 positive images, 372 negative images
- Test set:
  111 positive images, 108 negative images

Table 6.3 Confusion Matrix of SVM+HOG classification

	Negative(Predicted)	Positive(Predicted)
Negative (Actual)	98	10
Positive (Actual)	26	85

Testing Results

According to the Confusion Matrix shown in Table 7.4.2.1, we can get that

- The Predicted Positive Value is:
  85/(26+85) = 76.6%
- The Predicted Negative Value is:
  98/(98+10) = 90.7%

Conclusion:

The results demonstrate the feasibility of using HOG and SVM which can efficiently classify ROIs into the right classes.

6.3.3 FVE Performance Evaluation

FVE Requirement

Table 6.4 FVE Requirement of Human detection subsystem

Test B	***Human detection algorithm test***	
Description	Validates the capability of the algorithm to detect human signatures in RGB images	
Location	Lab	
Equipment	Laptop to run the detection algorithm, pre-stored images with relevant human signatures	
Steps	**Step Description**	**Performance Measures**
B.1.	Run the algorithm on the set of images	Ability to detect at least 60% of valid human signatures in images

Evaluation Results

- For 20 images from 2 different online videos, there are 64 humans in those images, and 42 of them are detected. Accuracy: **65.6%**
- For 20 images from 2 our collected videos, 42 of 65 humans in those images are successfully detected. Accuracy: **64.6%**

● Overall accuracy for 40 test images: **65.1%**

In conclusion, the human signature detection and analysis subsystem meets the FVE requirement.

6.3.4 Conclusion

The strengths of human signature detection and analysis subsystem are as follows:

- Since we use pre-processing algorithms to find ROIs that contain humans, it will be very efficient in finding potential human candidates in an image.
- Because we utilize edge detection as well as background subtraction to find human candidate, we are able to detect both the motionless and moving human beings.

The weaknesses of human signature detection and analysis subsystem are as follows:

- There some constraints in edge detection algorithm which may lead to a lower accuracy when dealing with human beings who are not vertical.
- Due to the limitation of HOG+SVM method, we can only detect humans with similar poses as those in the training set. Since we can only collect limited samples for the training set, our method cannot cover all poses. Based on the weaknesses mentioned above, our further work should mainly focus on
    - Improving the training set to include humans with different poses.
    - Using more advanced methods for finding human candidates and classification

7. Project management

7.1 Work Breakdown Structure

The work breakdown structure is shown as below:

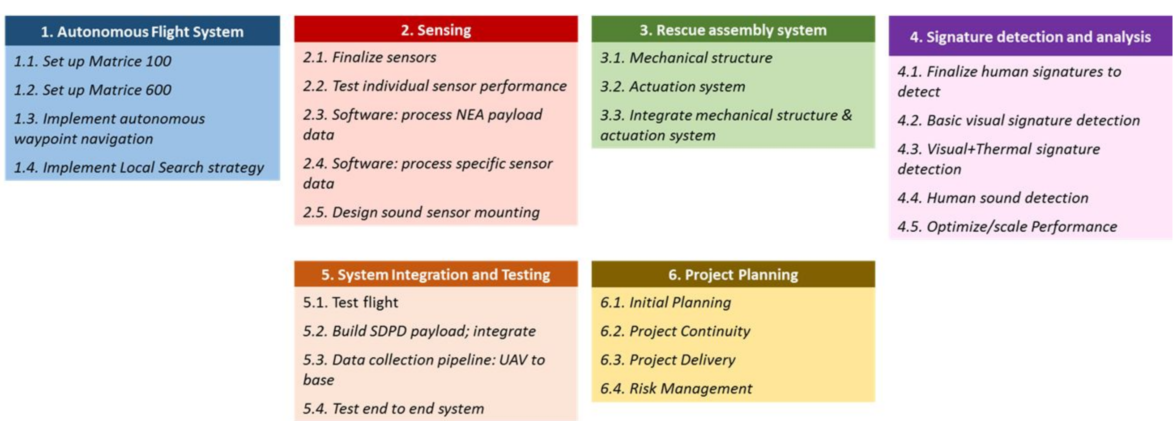


Figure 7.1. High-Level Work Breakdown Structure

As you can see, our work breakdown structure is created based on our subsystems. We have four subsystems and each of them will be a part of the work breakdown structure, including autonomous flight system, sensing system, rescue assembly system and signature detection and analysis system. Additionally, system integration and testing, as well as project management are also important parts of our work breakdown structure. A more detailed version of work breakdown structure is shown in Figure 7.2.

Figure 7.2. Detailed Work Breakdown Structure

7.2 Schedule

For each task in the detailed work breakdown structure, we assigned expected time to accomplish it and made the whole schedule. The high-level schedule is shown in Figure 7.3.

In our schedule, the blue blocks represent the plan for the project and give us a general idea about when we will work on this subsystem. Except for that, the green blocks are the tasks that we have already finished, while the yellow blocks stand for what we haven't achieved yet.

According to colors of the blocks in the schedule, we are basically on schedule. In the fall semester, we have made much progress on the autonomous flight system, rescue assembly system and signature detection and analysis system. For the sensing system, we have finalized types of three different sensors, and we will test their performances in spring. Also, more integration and testing tasks will be involved in the spring semester as well. Lastly but not the least, we will continue working on the project planning to make sure that tasks in every progress review will be finished on time.

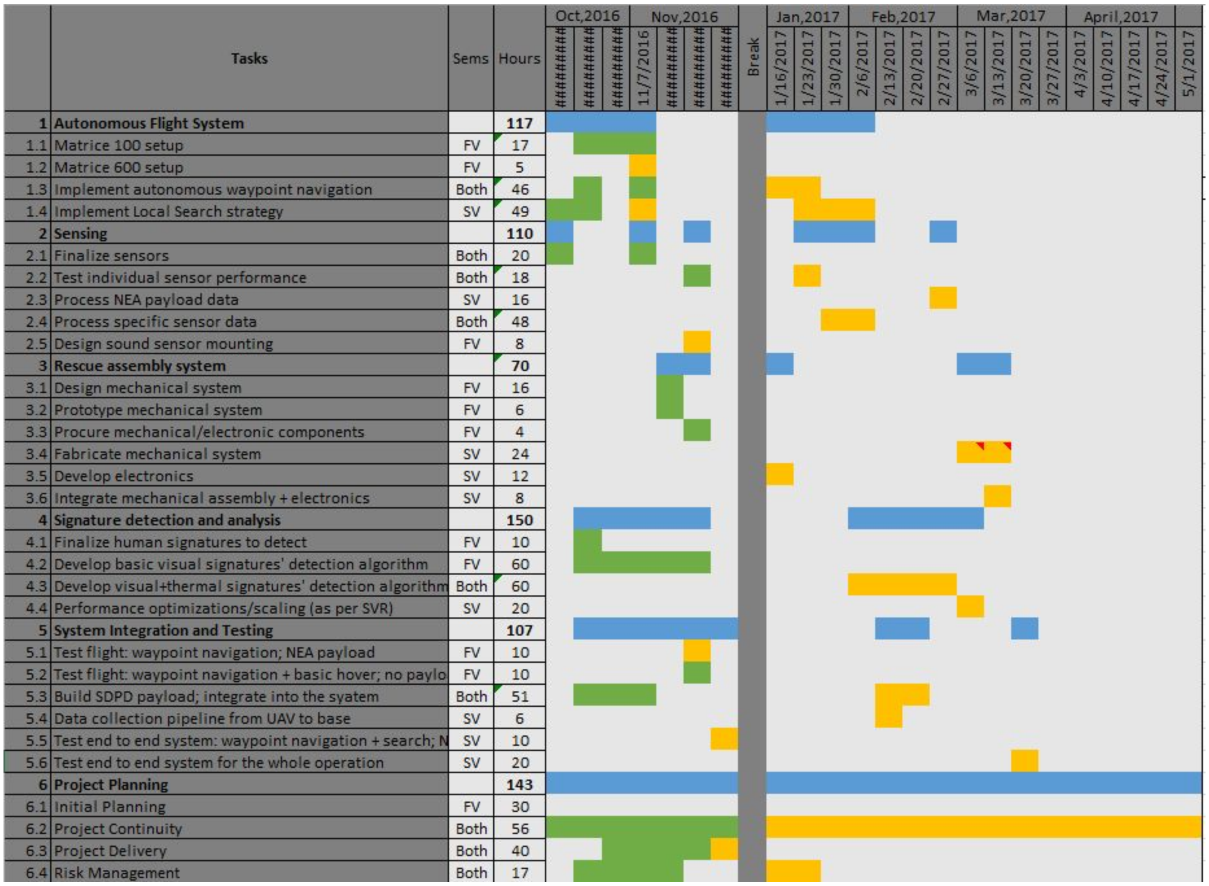


Figure 7.3. Schedule

7.3 Test plan

7.3.1. Capabilities & Milestones

Table 7.1. Capabilities & Milestones, Spring 2017

Milestone Date	Functionality

01/30/2017
PR 7
	• Autonomous Flight System
	    ○ Demonstrate Localized Pattern Navigation for best sensor coverage and resolution on a simulator.
	• Sensor
	    ○ Demonstrate human voice detection algorithm on sample data
	• Sound Detection and Payload Drop System
	    ○ Demonstrate design.

02/15/2017
PR 8
	• Autonomous Flight System
	    ○ Demonstrate Localized Pattern Navigation for best sensor coverage and resolution outdoors on dev drone.
	    ○ Demonstrate ability to run existing navigation app on sponsor's drone.

	○ Achieve 1 flight with actual sponsor drone and payload system. ● Sensor ○ Demonstrate the first version of thermal data based signature detection ○ Demonstrate improvements in RGB based signature detection. ○ Demonstrate software to extract and process payload data from sponsor's drone flight. ● Sound Detection and Payload Drop System (SDPD) ○ Implement CAD based prototype for SDPD ○ Demonstrate PDB integration with SDPD
02/28/2017 PR 9	● Autonomous Flight System ○ Demonstrate navigation to Rescue location and payload drop outdoors on dev drone. ● Sensor ○ Demonstrate improvements to thermal data based signature detection ○ Demonstrate first version of sound based signature detection ● Sound Detection and Payload Drop System ○ Demonstrate end to end SDPD system with PDB, sound system, and rescue package drop.
03/15/2017 PR 10	● Autonomous Flight System ○ Demonstrate one flight on sponsor drone using our waypoint navigation system. ● Sensor ○ Demonstrate Improvements to sound based signature detection. ○ Demonstrate end to end data collection and processing pipeline to transfer data from sponsor's payload and run on our signature detection. ● Sound Detection and Payload Drop System ○ Demonstrate end to end data collection and processing pipeline for the sound system. ○ Demonstrate mounting SDPD on sponsor drone.
04/01/2017 PR 11	● Autonomous Flight System ○ Demonstrate software to detect likely search locations in the absence of operator. ● Sensor ○ Demonstrate software to combine results from sensing subsystems and provide the most likely location of rescue. ● Sound Detection and Payload Drop System ○ Demonstrate flight with SDPD system and end to end data collection on sponsor drone and SVE site
04/15/2017 PR 12	● Integration and Testing ○ Demonstrate and test end to end system with all payloads and rescue subsystem fulfilling SVR requirements on sponsor's drone at SVE site.

7.3.2. Spring Validation Experiment (SVE):

Test D: Full System Test

Objective:

To validate the system's ability to autonomously search for a human in a search and rescue scenario and also dispatch a rescue package

Test conditions:

Location	Open 200m x 200m area with GPS access and normal wind.
Equipment needed	UAV; Laptop; Rescue package; 3 Mannequins (filled with hot water, wearing red shirt and with a speaker) /other representations of human signatures

Test Sequence:

Table 7.2. Spring Validation Experiment

Step	Description	Performance Measures
D.1.	Place UAV on the ground. Feed GPS locations as waypoints	
D.2.	UAV takes off and reaches the desired altitude for navigation	- Smoothness of takeoff - Accuracy in reaching desired height - (+-1m tolerance)
D.3.	UAV Reaches the first waypoint and performs localized search	- Accuracy in reaching the waypoint (+-5m tolerance)
D.4.	UAV Flies from one waypoint to another performing localized search	- Accuracy in reaching the waypoint (+-5m tolerance) - coverage as a percentage of planned search area
D.5.	UAV flies back and lands near the starting point after covering all the waypoints	- Accuracy in reaching the waypoints (+-5m tolerance)
D.6.	Transfer data from the UAV to base station	- Ability to collect the three types of perceptual data with spatial-temporal information
D.7.	Process the data to identify any human signatures (mannequin in our case)	- Ability to identify the mannequin
D.8.	Based on the identified human signatures, select the best location for rescue	- Accuracy of the human location conveyed (is it close to the mannequin?)
D.9.	UAV flies to the selected rescue location	- Accuracy in reaching the rescue location (+-5m tolerance)
D.10.	UAV performs localized search to get as close as possible to the human	- Ability to reach the desired altitude and close to the human (mannequin in our case)
D.11.	UAV releases the rescue package	- Ability to release the package

D.12.	UAV flies back to the base station	- Accuracy in reaching the starting location (+-5m tolerance)

System requirements validated:

- All mandatory performance and non-functional requirements

7.4 Parts list and budget

Table 7.3 Part list 1(Sponsored by Near Earth Autonomy-Parts still need to be finalized)

Description	Manufacturer	Model	Unit	Weight (g)	Cost
LWIR	FLIR	Tau 2	1	72	$7000
RGB Camera	Pointgrey	Grasshopper	1	520	$2,399
Lidar	Velodyne	VLP-16	1	590	$7,999
Flying platform	DJI	Matrice 600	1	9,600	$4,599
Panorama video camera	360fly	360fly 4k video camera	1	172	$399

Table 7.4 Part list 2 (Not sponsored by Near Earth Autonomy)

Description	Manufacturer	Model	Unit	Weight (g)	Cost
Aerial Platform	DJI	Matrice 100	1	680	$3250
Battery Heater	DJI	Inspired 1	1	100	$20
Propeller Guard	DJI	Matrice 100	1	200	$39
Insulation Sticker	DJI	Matrice 100	2	N/A	$4

Table 7.3 shows the items provided by our sponsor Near Earth Autonomy, and table 7.4 lists those that need to be purchased using our own budget. In all, we have a $5000 budget, and our total cost in the fall semester was 3313$, around 66.2% out of the total budget. The key item for us was the DJI Matrice 100, which cost $3250.

7.5 Risk management

One of the key risks in the previous iteration was in being able to spend enough time on a drone to demonstrate the various requirements for the Fall. The initial plan was to use the Sponsor's drone for demonstrating the capabilities but there was a big risk in being able to get time on the Sponsor's drone and being able to make iterative progress on it. This risk was mitigated by purchasing a dev drone and using that as the platform for Fall demonstration.

The risk analysis for the project is listed in Table 7.5 below. There are 3 major risks related to the availability of the drone and payload from the sponsor, ability of our signature detection techniques to detect humans accurately, and impact of weather on our drone components. Mitigation strategies have been devised for each of them and are being pursued aggressively.

Table 7.5 Risk Analysis

ID	Description	Likelihood of Occurrence	Level of Impact	Area of Impact	Mitigation Strategies
1	Difficult to schedule outdoor flying tests with the drone/payload provided by the sponsor.	3	High	Time, Reliability	1. Track requirements and dates for testing with Sponsor in high detail. 2. Request sponsor for past data for offline processing. 3. Schedule outdoor flying tests in advance with the sponsor. 4. Explore options to use continue using dev drone for final demo.
2	Difficulty in achieving high accuracy with signature detection	2	Medium	Reliability	1. Collect more sensor data to improve training sets. 2. Explore alternative algorithms for improving accuracy. 3. Explore options of getting sound sensor closer to the ground.
3	Impact of weather conditions on drone components.	3	High	Time, Reliability, Cost	1. Order additional backup components 2. Ensure all components and backups are preheated and ready before each flight.

Table 7.6.1 RiskID: 1

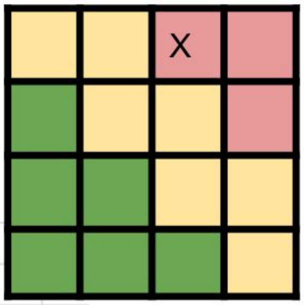


Table 7.6.2 RiskID: 2

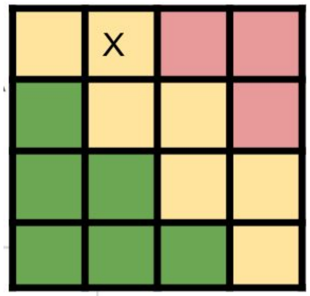
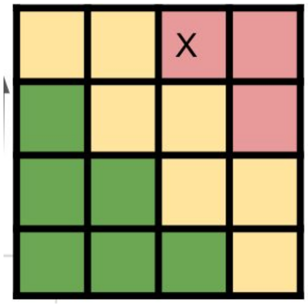


Table 7.6.3 RiskID: 3



8. Conclusions

In the fall semester, we have met all the FVE requirements we set in the beginning of the semester. However, there are still some lessons we need to learn from the fall semester.

- Requirements should be as clear as possible:
  For example, Human detection requirements should be more specific. To illustrate, we should put some constraints on the source of testing images.
- We should give more time to discussions:
  There will always be varied opinions, it's better to discuss them ahead of time.
- Keep things simple:
  For example, when designing package drop mechanism we faced no issues because of the simplicity.
- Carefully consider all the spares required
  For example, although expensive, a spare pre-heated battery for drone could have worked well for FVE Encore

Also, for the upcoming spring semester, the plan is to resolve the problems mentioned above and embark on additional key activities to finish the project.

- Requirements should be as clear as possible:
  To illustrate, we need to
  - Define human detection requirements properly
  - Define area coverage requirements properly
  - Think about ways to validate them
- Give time to discussions:
  For example, to set aside time to discuss whenever starting on a new thing.
- Keep things simple:
  We need to keep this in mind whenever designing something new
- Carefully consider all the spares required:
  For example, we can

- Order a new battery for the drone
- A spare pre-heated battery is a great idea for flights in cold weather conditions

9. References

[1]https://www.researchgate.net/publication/26795932_Dead_Men_Walking_Search_and_Rescue_in_US_National_Parks

[2] http://www.telecom.ulg.ac.be/research/vibe/doc2/index.html

[3]https://www.mathworks.com/help/images/examples/detecting-a-cell-using-image-segmentation.html

[4] http://www.pyimagesearch.com/2014/11/10/histogram-oriented-gradients-object-detection/

[5] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. Computer Vision and Pattern Recognition (CVPR), 2005. 1

[6] Point cloud image, Courtesy, Near Earth Autonomy, http://www.nearearth.aero

[7] Use case has been designed after reading cases "Yosemite National Park, California". Refer to links: https://www.nps.gov/yose/blogs/psarblog.htm, https://www.nps.gov/yose/getinvolved/sar_jobs.htm, https://www.nps.gov/yose/blogs/Three-Distressed-Hikers-Rescued-Tuolumne-Meadows-Area.htm, http://www.climbing.com/news/10-things-you-didnt-know-about-yosar/