THE ROBOTICS INSTITUTE

Conceptual Design Review

Team F

Falcon Eye

Yuchi Wang, Rahul Ramakrishnan,
Danendra Singh, Pulkit Goyal, Prathibha Tripathi

sponsored by
Katia Sycara

October 4, 2017

Contents

1 Project description

Natural disasters are massively damaging events; from 2004 to 2015, an average of 200 million victims were affected by disasters each year. In 2010, the Haiti earthquake claimed a total of 225570 lives, displaced 1.5 million people and caused 8$ billion worth of economic damage. Communities affected by disasters take 20-30 years to recover and each day that first aid is delayed extends the recovery period by 1000 days.

Robin Murphy, renowned professor at UT and rescue robotics expert, states that the main challenge in disaster relief is not the rescue component but rather the lack of situational information available to rescue teams. Currently, first aid responders rely on high altitude satellite images or ground views to gain an understanding of the disaster area. However, images from these viewpoints typically do not give rescue teams sufficient information about the hazards, physical obstructions, and locations of survivors in the local area. Without the required situation awareness, responders are unable to approach a disaster zone as safely and quickly as they otherwise could, thus delaying the overall relief effort.

To address this problem, Team F proposes the FalconEye **aerial surveillance system**. Falcon Eye is a heterogeneous robotic system that operators can safely deploy from outside the affected disaster areas. Once deployed, FalconEye will autonomously proceed to a target location inside the disaster areas and launch a fleet of UAV (unmanned aerial vehicles) that is able to provide low-altitude reconnaissance of desired target zones and structures. The aerial sensor data is relayed in real-time back to the operator, who provides locational commands to Falcon Eye to set the next target destination. This control architecture absolves the operators from the cognitive workload required by direction teleoperation and enables them to focus on the high-level coordination and planning of the rescue operation.

2 Use case

In 2020, Pittsburgh is ravaged by a magnitude 3.0 earthquake and suffers heavy infrastructural damage to the buildings, pipelines, and electrical grid. Crumbled buildings, displaced streets and the resultant debris leave many locations inaccessible from the ground. Some of the buildings remain standing but rescue personnel are uncertain about the damage sustained by the infrastructure and the location of any safe access points. Satellite images lack the fidelity and the angle to show the detailed damage sustained by the structures and ground obstructions and hazards prevent relief personnel from approaching certain locations. Many people are trapped amidst the debris, time is of the essence, and disaster relief teams urgently need more information about the landscape of the disaster zones.

The city of Pittsburgh dispatches Captain Dolan with FalconEye, the heterogeneous aerial surveillance system, to survey the disaster zones and provide crucial aerial information to the rescue teams. Dolan arrives at the outskirts of Pittsburgh and safely sets up Falcon Eye. A map of pre-disaster Pittsburgh that also shows the estimated areas of damage is loaded onto Dolan's computer. Dolan inputs a GPS location on the map to indicate the approximate area that he would like to conduct aerial surveillance.

The AGV fleet plans a path to the location using the GPS and path planning algorithm and begins to autonomously transport the UAV's there. Since debris, water, and other physical obstructions may be present, the AGV's detect obstacles and attempt to navigate around them when possible. Once the AGV's have determined that they are unable to proceed further, it automatically deploys the UAV's and notifies Dolan via the computer that aerial surveillance is ready.

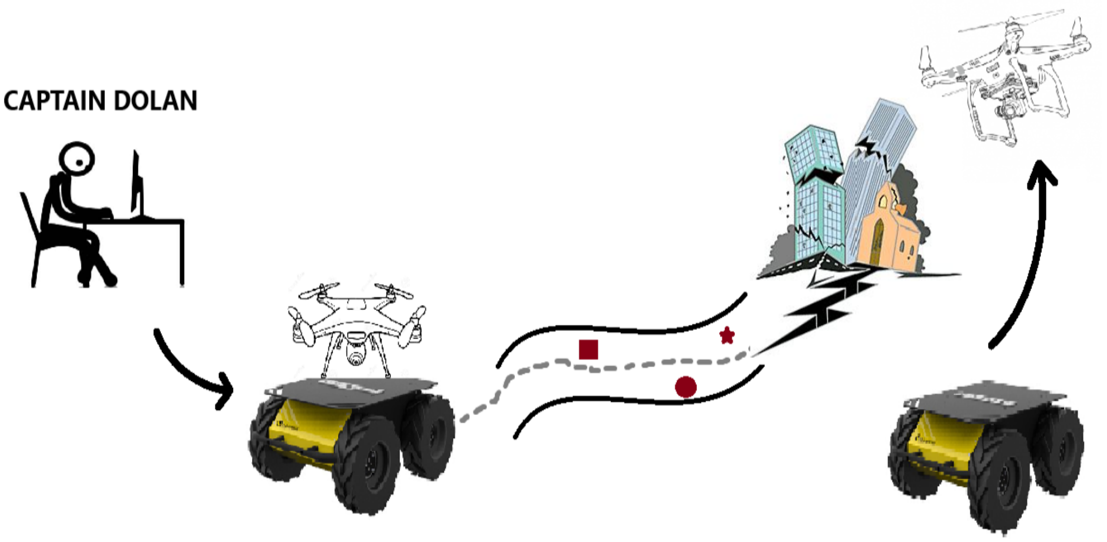


Figure 1: Use case illustration

The aerial sensor feed is relayed in real time from the drones to Dolan's computer. Dolan notices an anomaly at a certain location on the video feed and he wishes to gain a closer look. He instructs the UAV to travel to that location by inputting the corresponding GPS coordinate into the computer. The UAV fleet receives this GPS location and autonomously travels there. Once they reach the location, they plan a surveillance pattern for the area and begin sweeping the area. When the UAV's have finished the surveillance, Dolan has gained a sufficient understanding of that specific area and the survivors and indicates to the rescue team the safest path to proceed based on AGV's sensor information as well as the UAV's feed.

3 System level requirements

3.1 Mandatory performance requirements

M.P.1: The AGV will avoid obstacles 30 × 30 cm in dimensions.
The AGV is expected to avoid reasonably sized pieces of debris.

M.P.2: The AGV will travel over obstacles 5 cm in height.
Rough terrain will be expected in a disaster zone.

M.P.3: The AGV will have a carrying capacity of at least 15kg.
The UAV will have to be transported to the disaster zone.

M.P.4: The AGV will travel at an average speed of 2 mph.
The AGV is expected to move at a certain speed.

M.P.5: The AGV will operate for 4 hours on a single charge.
The AGV will be out on the field for long periods at a time.

M.P.6: The UAV will travel at an average speed of 30 mph.
The system may need to survey many different areas in under a certain time.

M.P.7: The UAV will travel to the target location within 20m radius
This is the acceptable range of error for conducting aerial surveillance.

M.P.8: The UAV will survey at a minimum height of 50m.
Aerial surveillance needs to be conducted close to the ground to see important details.

M.P.9: The UAV will relay video back to the user with a max latency of 1 s.
Latency cannot be too high to reduce operator inefficiency.

M.P.10: The UAV will capture video at a minimum resolution of 720p.
The resolution needs to be high enough for the operator to see important details on the ground.

3.2 Mandatory non-functional requirements

M.N.1: The system will cost a total 5000$.
Budget set by MRSD project - does not include material already available.

M.N.2: The system will have a form factor that can be transported in a container truck.
Reasonable form factor for deployment purposes during disaster relief.

M.N.3: The system will have a maximum combined weight of 70kg per UAV & AGV pair.
The system needs to be decently portable for ease of transportation.

M.N.4: The system will consist of at least 1 UAV and 1 AGV.

3.3 Desired performance requirements

D.P.1: The AGV will avoid obstacles 10 × 10 cm in dimensions.

D.P.2: The AGV will travel at an average speed of 3 mph.

D.P.3: The UAV will travel at an average speed of 40 mph.

D.P.4: The UAV will travel to the target location within 10m radius.

D.P.5: Video streamed will have a max latency of 0.5 s.

D.P.6: Video streamed will have a min resolution of 1080p

3.4 Desired non functional requirements

D.N.1: The system will have a maximum combined weight of 60kg per UAV & AGV pair

D.N.2: The system will consist of at least 3 UAV and 3 AGV.

D.N.3: The system will be able to operate effectively in heavy snow conditions.

4 Functional Architecture

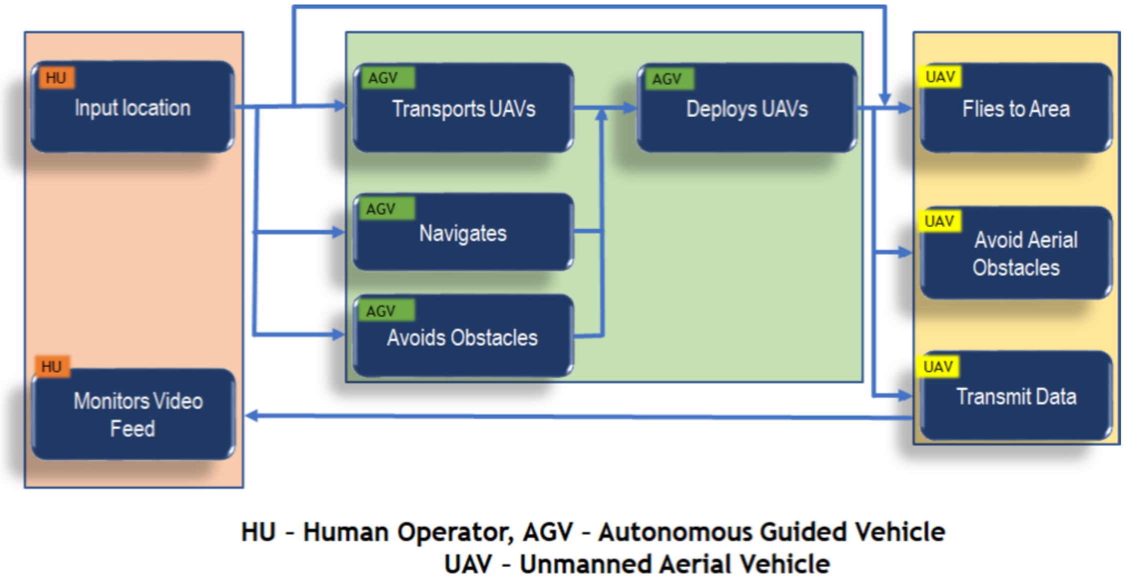


Figure 2: Functional architecture

The functional architecture for Falcon Eye can be divided into two overarching subsystems: the AGV and the UAV. To start with, the AGV is active while the UAV is being transported to the disaster zone by the AGV. The user will provide an input location to the AGV in the form of a GPS coordinate. Using this target location and GPS, the AGV plans a probable path to the target. This path is probable because it doesn't take into account the presence of obstacles in its path. While the AGV is proceeding according to its calculated path, it constantly monitors and processes information from its onboard sensors to perceive the surrounding environment and avoid obstacles. If it senses an obstacle obstructing its planned path, it will update its internal map and recalculate another probable path. Once it has either arrived at the target location or determined that it cannot proceed further, the AGV releases the UAV's from its transportation mechanism.

Once the AGV releases the UAV's, the operator is notified of this event. The user proceeds to give locational commands to the UAV in the form of GPS coordinates. With the GPS location, the UAV's use a path planner to calculate a trajectory to the location and translates the necessary motion into high-level yaw, pitch and roll commands. These commands are fed to the control system onboard the drone which will maintain the desired state while auto stabilizing. While the UAV is flying, it is continuously relaying the sensory data back to the user, who visualizes it in real time. At any point when the drone is flying, the operator is able to input a new location.

5 System level trade studies

5.1 AGV platform

The AGV platform plays a major role in the operation of the entire system and needs to be chosen wisely based on several factors. The AGV has to be rugged enough to sustain the rough and unpredictable environmental conditions associated with disaster zones. The platform needs to be agile enough to avoid large physical obstructions in its path and while maneuvering over smaller obstacles without getting stuck. It needs to be sufficiently mobile to reach the target location in a reasonable time and while transporting the UAV payload. The AGV should be able to operate for hours at a time while performing its surveillance mission. Lastly, the cost for these platforms is considered - please note that if the platform is in inventory at CMU, it is given a cost of 0 for the purposes of the trade study. Considering all these factors, we have narrowed down upon the **Clearpath Husky**.

AGV Platform	Carrying capacity	Ruggedness	Speed	Operation time	Cost	Total
Clearpath Turtlebot	1	1	1	3	5	11
Clearpath Jackal	2	3	4	4	2	18
Clearpath Husky	**4**	**4**	**3**	**5**	**5**	**21**
Clearpath Warthog	5	5	5	3	1	19
Boston Dynamics BigDog	3	3	3	3	0	12
Boston Dynamics LS3	4	4	3	5	0	16

Table 1: Trade study for the AGV platform

5.2 AGV onboard sensors

The effectiveness of path planning and object avoidance depends heavily on the sensors that are used. For this application, the sensors needs to have sufficient accuracy and sensor range such that obstacles can be detected precisely and with enough range for the AGV platform to make the necessary avoidance maneuvers. The usefulness criteria in Table 2 is indicative of how well the type of sensor data will help the path planning and localization subsystems. For instance, a 2D lidar or a stereo camera that performs suboptimally in bright light will receive a lower score than a 3D lidar. The sensor should also have a wide angle of coverage and should be easy to integrate with our system. These requirements led us to finalize on using **Velodyne VLP 16** sensor to map the environment.

Onboard sensors	Accuracy	Sensor range	Usefulness	Coverage angle	Cost	Total
Velodyne VLP 16	**3**	**5**	**5**	**5**	**1**	**19**
Sick LMS111	3	3	2	4	1	13
Microsoft Kinect	4	1	2	2	5	14
ZED 2k Stereo Cameras	5	3	3	3	4	18
Hokuyo URG-04LX-UG01	4	5	2	4	3	18

Table 2: Trade study for the AGV onboard sensors

5.3 UAV platform

The UAV platform forms the basis of the aerial surveillance system and therefore needs to be robust. Its speed should be high enough to enable quickly reach the disaster zone its deployment location and cover the surveillance area in a reasonable time. It should have high flight time to cover the maximal area and stable enough to provide clear video coverage of the target location. The package support for any given drone should also enable quick software integration and deployment. All these factors helped us to decide on **Bebop 2** as our UAV platform.

Drone platform	Speed	Equipped sensors	Operation time	Developer support	Stability	Cost	Total
Clearpath Pelican	2	5	4	4	5	2	22
Parrot AR Drone 2.0 Power	3	4	2	5	2	5	23
Parrot Bebop 2	**5**	**3**	**5**	**4**	**4**	**5**	**26**
DJI Phantom 2	3	3	5	5	4	5	25
DJI Mavic Pro	5	3	5	5	4	3	25

Table 3: Trade study for the AGV onboard sensors

6 Cyberphysical architecture

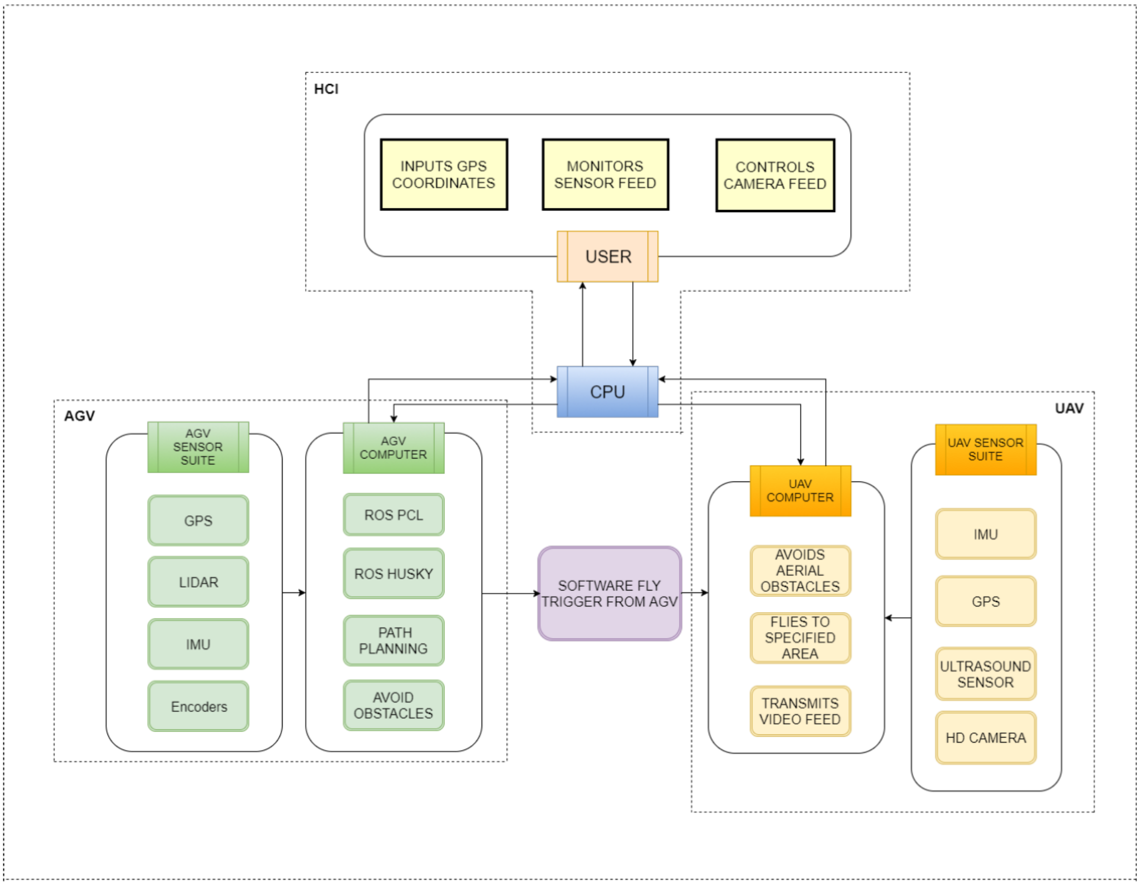


Figure 3: Cyberphysical architecture

The cyber-physical architecture depicted in the Figure 3 details the specific functionality and components of the main subsystems.

We have divided our cyber-physical architecture into 4 main sub-sections:

1. AGV (Software and sensor suite)
2. UAV (Software and sensor suite)
3. Central Processing Unit (CPU)
4. Human-Computer Interaction

6.1 Autonomous Ground Vehicle(AGV)

Our AGV will be performing the following functions:

- Take input from its sensor suite and perform onboard processing on Mini ITX computer.
- Autonomously plan path and reach target destination using GPS inputs from the operator.
- Avoid and static and dynamic obstacles on its way.

- Deploy drones on reaching the desired location.

6.2 Unmanned Aerial Vehicle(UAV)

The UAV has the primary task of aerial surveillance and transmitting live video feed and sensor data to the base station computer. In addition, the UAV performs the following operations:

- Avoid collisions with other UAVs and other static objects.
- Send real time data via wiFi channel to the operator.
- Execute predefined path flying.
- Receive new waypoints from user.

6.3 Central Processing Unit(CPU)

The central processing unit is essentially a computer at the base station that interacts with all other subsystems. It has a two way communication channel between the user, AGV and UAV. It is used to pass GPS waypoints to the AGV and UAV. It also receives the video feed back from the UAV and pass it forward for display on a computer screen.

6.4 Human Computer Interaction(HCI)

The HCI subsystem is defined as follows:

- This subsystem comprises of user interface with the overall system.
- It incorporates a monitor for displaying the live feed from the UAVs.
- A CPU for getting input commands from the user and transmitting them to the robots

7 Subsystem description

7.1 AGV Subsystem

The main objective of AGV is drive to the closest location to the disaster zone and then trigger the UAV flying. While moving to the disaster zone the AGV avoids obstacles on the way and incorporates path planning. We will be using Clearpath's Husky (Figure 4) mobile platform that has been provided by Professor George Kantor.

Figure 4: Clearpath Husky AGV

7.1.1 AGV sensing

The AGV generates environment map using a LIDAR sensor. A LIDAR usually consists of a scanner and a laser. The laser emits a laser beam at a certain frequency and also receives the reflected laser beam from an object on the path of the laser. This can be used to estimate the distance between the LIDAR sensor and the object by estimating the time difference between the transmitted and received laser. This method is a very reliable method of estimating the distance upto a certain limit depending upon the LIDAR specifications.

We have selected Velodyne VLP 16 sensor (Figure 5) due to its excellent online support and reliability.

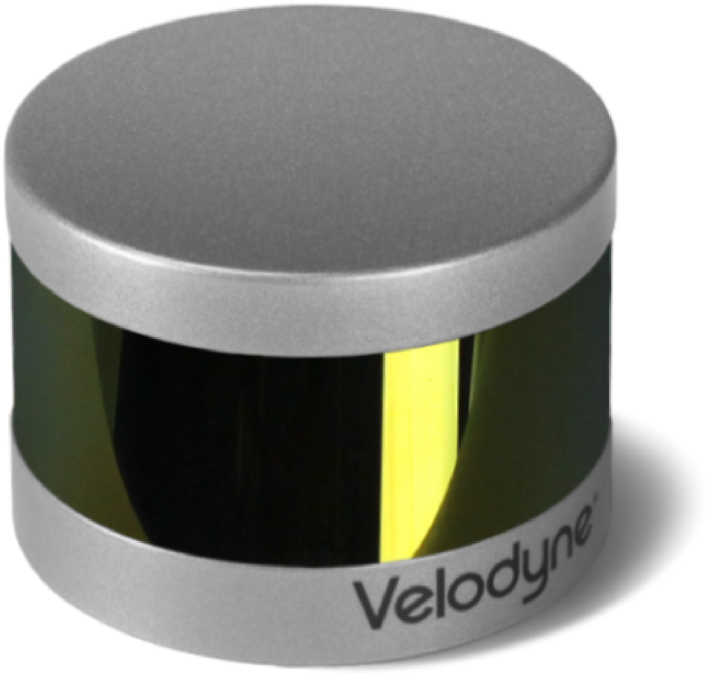


Figure 5: Velodyne VLP 16

7.1.2 Localization

We will be using a GPS system on the AGV for global localization of the vehicle. Using global waypoint generation we can navigate in a large environment space using individual GPS waypoints. We will use ROS package for husky to transfer GPS coordinate input by the user. Conventional GPS modules have an accuracy of around 5-8 meters which can be used to localize the AGV. Please refer to Figure 6 for a visualization of our localization architecture.

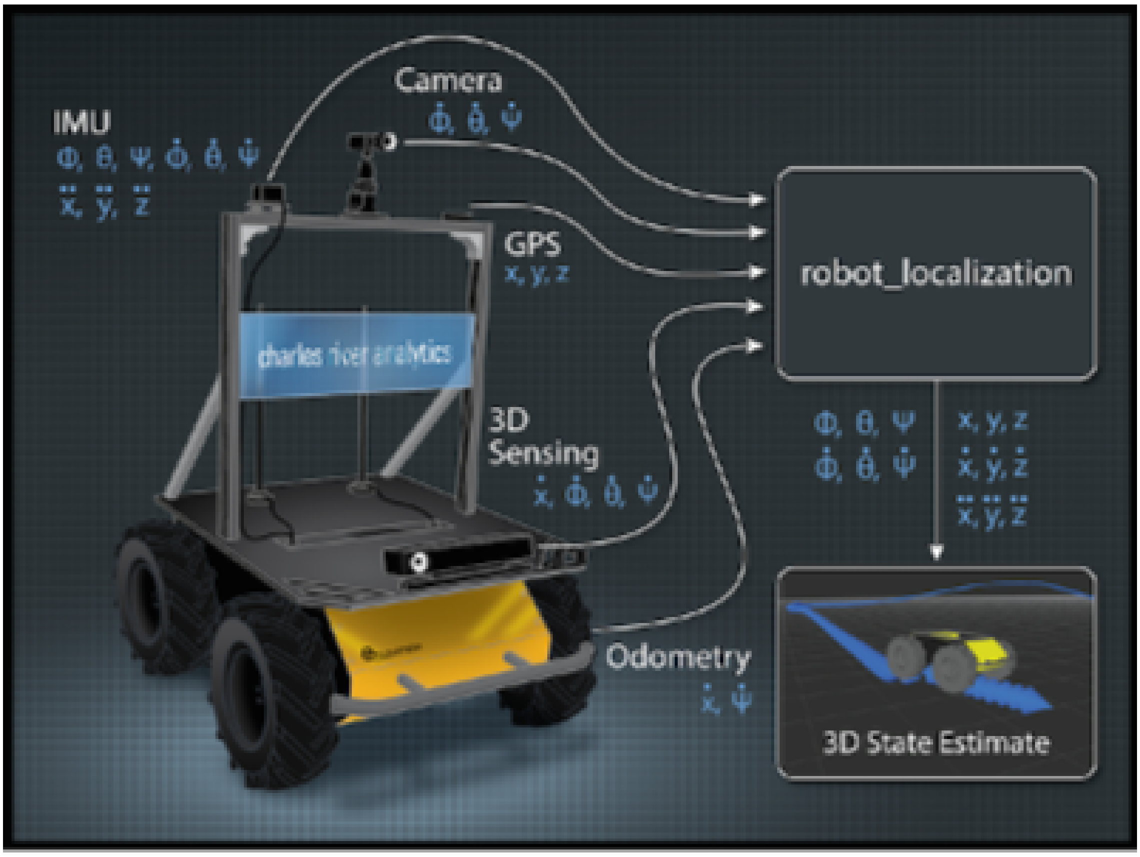


Figure 6: Localization methodology

7.1.3 Object Detection

LIDAR is used to detect ground obstacles that lie in the path of the AGV. We can classify the reflected lasers that have the same bouncing distance from the LIDAR and our neighbours as one object and using centroid estimation techniques, the center of the object can be fairly estimated for object avoidance.

7.1.4 Path Planning

We are going to develop an algorithm for optimized localized-navigation. Search based planning algorithms like A*, D* and D* Lite have proven to be very effective for motion planning of a mobile robots. We have decided to use search based algorithm and test them on the AGV, finalizing the one which does not depreciate the computational capabilities of the AGV while at the same time accurately plans the correct path.

7.2 Central Processing Unit

A base computer is being used as a centralized unit for data transfer between all subsystems. It will also be responsible for receiving data inputs from the user and display live video feed from the UAV via an easy to use GUI. The specifications of the base unit are as follows:

- I7-7700HQ Intel Processor @2.8 GHz
- 16 GB DDR4-RAM
- NVIDIA GeForce GTX 1060 6GB GDDR5 GPU

- 1TB Hard Disk Drive, 5400rpm, 2.5”, SATA3
- 512GB Solid State Drive M.2 PCIe

7.3 Communication

The communication system is critical for the mission and hence should be robust, reliable and efficient. Since it will encounter multiple communication channels, the module should not interfere with other channels, avoid latency and give excellent range.

We have decided to use Ubiquiti BULLET M-5 outdoor 5 GHz WiFi radio for our communication needs.

7.4 UAV Subsystem

After getting deployed from the AGV, the UAV will perform autonomous aerial surveillance of the area and transmit High-Definition video frames back to the operator. We will be using Parrot Bebop 2 (Figure 7) drones, that are provided to us by the sponsor. The drones come preinstalled with a HD camera and software image stabilization.

Figure 7: Parrot Bebop 2 Drone

7.4.1 UAV Sensing

The UAV comes with a High-Definition fisheye camera. This can be used to transmit live video feed from the UAV to the base station. The fisheye lens can be used as a software gimble and hence transmit video feed frames from different angles.

7.4.2 Localization and Path Planning

GPS based localization and path planning will be incorporated for autonomous surveillance of the disaster area. Multiple UAVs help in covering more area in the same time, hence allowing the user to identify area of maximum interest and focus the surveillance operation in those areas.

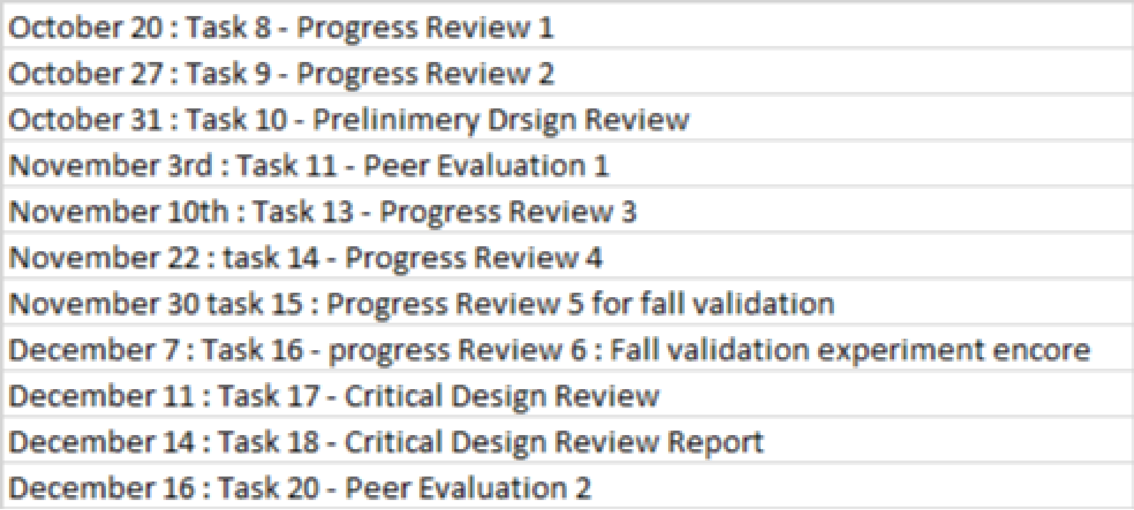
8 Project management

8.1 Work plan

Major Tasks	Milestones
1) System component finalization	1. Finalize the platform for AGV. 2. Finalize the perception system sensors : Lidar, Camera. 3. Finalize the sensors for localization system : GPS, IMU, Odometry. 4. Procure the batteries, spares and the above finalized hardware.
2) AGV Sensor Calibration and Data Capture	1. Outdoor testing of the AGV platform with RC. 2. Checking the Odometry data of the AGV. 3. Statically testing the standalone Lidar sensor for the accuracy of data. 4. Statically testing the GPS data in outdoor environment. 5. Statically checking the IMU for accurate sensor data.
3) AGV Basic ROS sensor Software Development	1. Finish CAD design and start fabrication. (a) For mounting the sensors on AGV (b) For mechanical platform to place UAV over AGV. 2. ROS driver setup for the different sensors (IMU, GPS, LIDAR, Odometry). 3. Setup ROS on the on-board PC of AGV. 4. Test he batteries for on-board PC, and different sensors integration.
4) Sensor Integration	1. Software development and testing for different sensors (IMU, GPS, LIDAR, Odometry) with ROS. 2. Obstacle detection for AGV using the Lidar. 3. Path Planning : considering virtual obstacles in environment. 4. Localization : Integration of GPS and wheel odometry
5) Major Sub-system integration wilh fall demo vehicle	1. Integrating obstacle detection with path planning 2. Motion Planning and control for AGV movement integrating all the software stacks. 3. Behavioural model for autonomous navigation. 4. Save the map for SLAM. 5. Testing and validation the entire sub-system.
6) UAV Sensor Calibration and Data Capture	1. Outdoor free flight test of the UAV platform with RC. 2. The transmitter and receiver pairing. 3. Statically testing the GPS data in outdoor environment. 4. Statically checking the IMU for accurate sensor data.

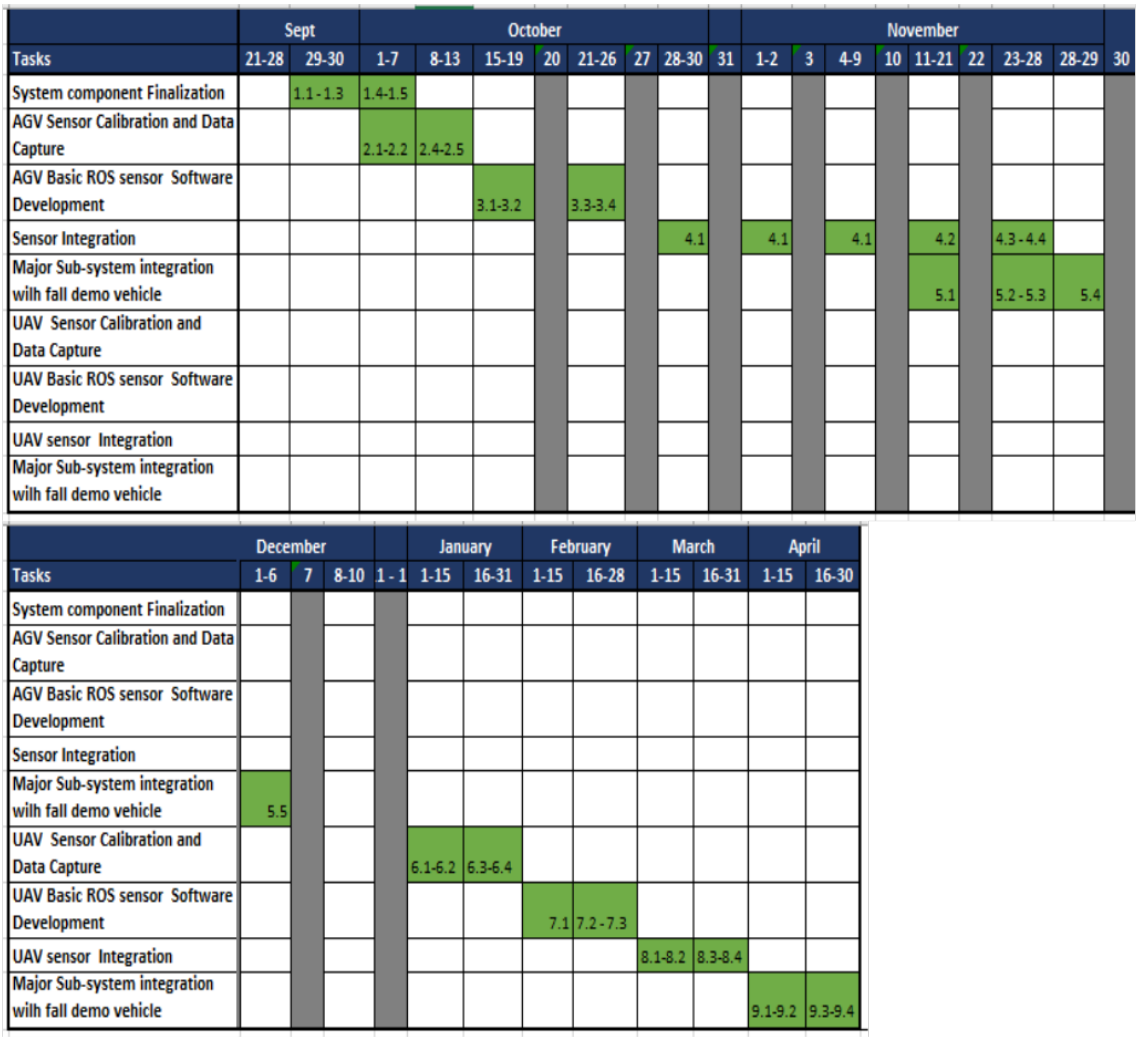
7)UAV Basic ROS sensor Software Development	1. ROS driver setup for the different sensors (IMU, GPS, Camera). 2. Setup ROS for the UAV. 3. Test the batteries with different sensors.
8) UAV sensor Integration	1. Setup the PC to control UAV remotely. 2. Take off UAV from the AGV platform 3. Detect obstacles 4. Path Planning with Virtual obstacles
9) Major Sub-system integration wilh fall demo vehicle	1. Path Planning with obstacle avoidance 2. Motion Planning and control for UAV movement integrating all the software stacks. 3. Video feed from the camera to remote PC. 4. GUI for the system.

8.2 Schedule and milestones



October 20 : Task 8 - Progress Review 1
October 27 : Task 9 - Progress Review 2
October 31 : Task 10 - Prelinimery Drsign Review
November 3rd : Task 11 - Peer Evaluation 1
November 10th : Task 13 - Progress Review 3
November 22 : task 14 - Progress Review 4
November 30 task 15 : Progress Review 5 for fall validation
December 7 : Task 16 - progress Review 6 : Fall validation experiment encore
December 11 : Task 17 - Critical Design Review
December 14 : Task 18 - Critical Design Review Report
December 16 : Task 20 - Peer Evaluation 2

Figure 8: Schedule



	Sept		October								November								
Tasks	21-28	29-30	1-7	8-13	15-19	20	21-26	27	28-30	31	1-2	3	4-9	10	11-21	22	23-28	28-29	30
System component Finalization		1.1 - 1.3	1.4-1.5																
AGV Sensor Calibration and Data Capture			2.1-2.2	2.4-2.5															
AGV Basic ROS sensor Software Development					3.1-3.2		3.3-3.4												
Sensor Integration									4.1		4.1		4.1		4.2		4.3 - 4.4		
Major Sub-system integration wilh fall demo vehicle															5.1		5.2 - 5.3	5.4	
UAV Sensor Calibration and Data Capture																			
UAV Basic ROS sensor Software Development																			
UAV sensor Integration																			
Major Sub-system integration wilh fall demo vehicle																			

	December				January		February		March		April	
Tasks	1-6	7	8-10	1 - 1	1-15	16-31	1-15	16-28	1-15	16-31	1-15	16-30
System component Finalization												
AGV Sensor Calibration and Data Capture												
AGV Basic ROS sensor Software Development												
Sensor Integration												
Major Sub-system integration wilh fall demo vehicle	5.5											
UAV Sensor Calibration and Data Capture					6.1-6.2	6.3-6.4						
UAV Basic ROS sensor Software Development							7.1	7.2 - 7.3				
UAV sensor Integration									8.1-8.2	8.3-8.4		
Major Sub-system integration wilh fall demo vehicle											9.1-9.2	9.3-9.4

Figure 9: Gantt chart

8.3 System validation experiments

8.3.1 Fall validation Experiment

The fall validation experiment will be done with autonomously moving UGV, that simply navigates from point A to point B on the basis of path planning algorithm with obstacle avoidance. This will take place in an outdoor setting at Carnegie Mellon University in Robotics Institute. It will perform following tasks:

1. Localization: Test 1
   (a) Procedure: The AGV will localize in the surroundings.
   (b) Performance metrics: The AGV will localize with accuracy of 80
2. Path Planning: Test 2
   (a) Procedure: Give the destination GPS location with no obstacle in the path.
   (b) Performance metrics: Drive from the initial point to the user provided GPS location with the accuracy of 20m.
3. Obstacle Avoidance Test 3

    (a) Procedure: Give the destination GPS location with obstacle in the path.
    (b) Performance metrics: Avoid static obstacles of 3 × 30 cm.

8.3.2 Spring validation Experiment

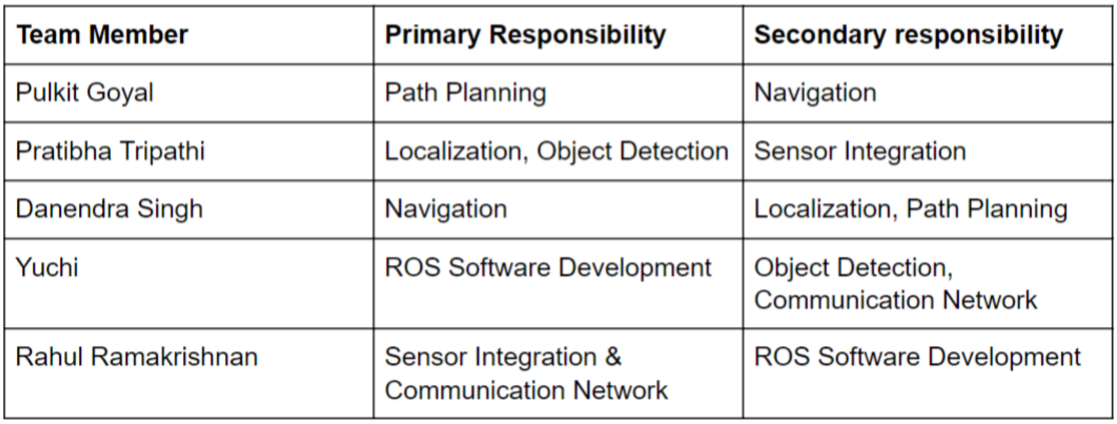
The spring validation experiment will be done with autonomously moving UGV to the user given location and then flying UAV from top of it. The UAV will fly and transfer the video stream to the remote PC. This will take place in an outdoor setting at Carnegie Mellon University in Robotics Institute. It will perform following tasks:

1. Mapping the environment with UAV
    (a) Procedure: The AGV will map the localized environment around it.
    (b) Performance metrics: The AGV will do the loop closure with 80
2. UAV flying trigger: Test 1
    (a) Procedure: AGV will trigger the flying of UAV after reaching the user given GPS location
    (b) Performance metrics: AGV will trigger the flying of UAV with 90% success rate. location with the accuracy of 20m.
3. Obstacle Avoidance Test 3
    (a) Procedure: Place obstacle in path of UAV.
    (b) Performance metrics: UAV will not collide with the placed obstacle in its environment with accuracy of 90

8.4 Budget

S. No.	Component	Sub-Components	Quantity	Unit Cost	Total Cost
1.	Parrot Bebop 2 drone	HD Camera	1	$400	$400
		On-board IMU			
		Flight Controller			
S. No.	Component	Sub-Components	Quantity	Unit Cost	Total Cost
2.	Husky	GPS	1	Sponsored	–
		IMU			
		On-board Processor			
		Li-Ion-Phosphate Battery	2	$405	$810
3.	Velodyne Puck		1	$8000	$8000
4.	Ubiquiti BULLET	M5-HP Outdoor	2	$89	$178
	Total			$9388	

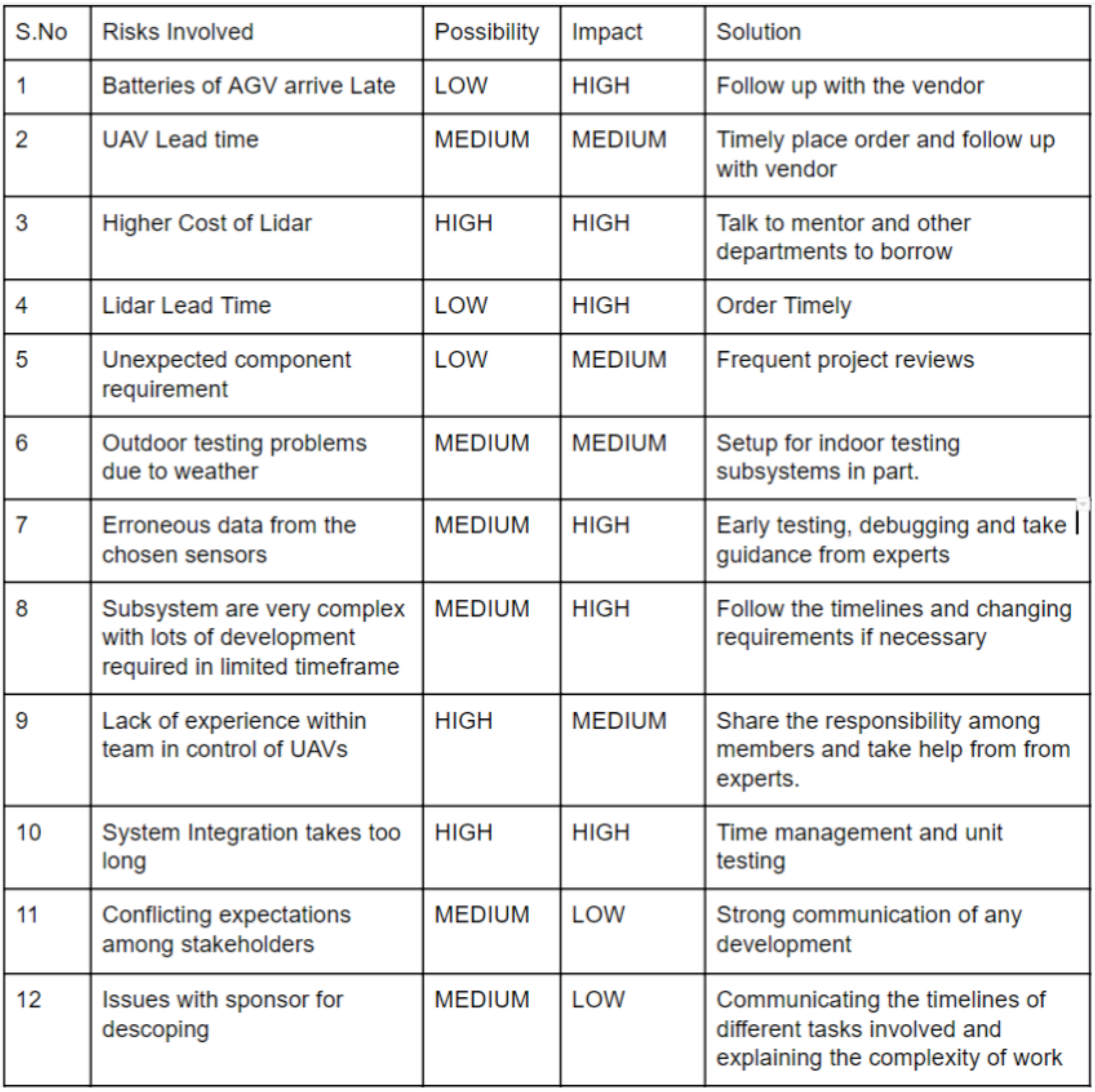
8.5 Team member responsibility



Team Member	Primary Responsibility	Secondary responsibility
Pulkit Goyal	Path Planning	Navigation
Pratibha Tripathi	Localization, Object Detection	Sensor Integration
Danendra Singh	Navigation	Localization, Path Planning
Yuchi	ROS Software Development	Object Detection, Communication Network
Rahul Ramakrishnan	Sensor Integration & Communication Network	ROS Software Development

Figure 10: Team member responsibility

8.6 Risk management



S.No	Risks Involved	Possibility	Impact	Solution
1	Batteries of AGV arrive Late	LOW	HIGH	Follow up with the vendor
2	UAV Lead time	MEDIUM	MEDIUM	Timely place order and follow up with vendor
3	Higher Cost of Lidar	HIGH	HIGH	Talk to mentor and other departments to borrow
4	Lidar Lead Time	LOW	HIGH	Order Timely
5	Unexpected component requirement	LOW	MEDIUM	Frequent project reviews
6	Outdoor testing problems due to weather	MEDIUM	MEDIUM	Setup for indoor testing subsystems in part.
7	Erroneous data from the chosen sensors	MEDIUM	HIGH	Early testing, debugging and take guidance from experts
8	Subsystem are very complex with lots of development required in limited timeframe	MEDIUM	HIGH	Follow the timelines and changing requirements if necessary
9	Lack of experience within team in control of UAVs	HIGH	MEDIUM	Share the responsibility among members and take help from from experts.
10	System Integration takes too long	HIGH	HIGH	Time management and unit testing
11	Conflicting expectations among stakeholders	MEDIUM	LOW	Strong communication of any development
12	Issues with sponsor for descoping	MEDIUM	LOW	Communicating the timelines of different tasks involved and explaining the complexity of work

Figure 11: Risk management

References

[1] `http://library.isr.ist.utl.pt/docs/roswiki/Robots%282f%29Husky.html` .

[2] `http://velodynelidar.com/vlp-16.html` .

[3] `https://www.clearpathrobotics.com/tag/husky/` .

[4] `https://www.amazon.com/Ubiquiti-BULLET-M5-HP-Outdoor-802-11n-M5HP/dp/B002SYTPMU` .

[5] MRSD Fall 2016 Conceptual Design Review document of Team Rescue Rangers (F).