ArIES IIT ROORKEE
Technical Design Paper
AUVSI SUAS 2021

Abstract:

The AUVSI SUAS Competition is designed to stimulate interest in Unmanned Aerial Systems (UAS) technology and even careers, and more importantly, to engage students in a challenging UAS mission. This report describes the modus operandi of Team ArIES to achieve the tasks at hand, laid down by the 2020 edition of the same. The whole report can be divided into five major parts: The first section talks about the requirements and the criteria of acceptance for the mission. The second, labeled as System Design, contains information about the structure and functionality of the UAV. All the major components and their work have been discussed in detail here. The third section, named Alternative Considered, describes the alternatives that were considered but not used and the rationale behind them. Fourth, Testing and Evaluation give an insight about our approach to test and evaluate the performance of all the parts and functionality of the UAV. Fifth, Safety, Risks, & Mitigations consider the risk involved in the development and mission process and mitigations to counter them, which has been discussed in detail.

1. Requirements and Acceptance Criteria

Upon takeoff, we must immediately fly the waypoint path from waypoint 1, before attempting other tasks, thereby simulating the trip to the operation area. We are allowed to attempt other tasks while flying the waypoints. We must successfully takeoff and go above 100ft MSL within the first 10 minutes of the mission clock, or the demonstration will be terminated. And, we will be provided with 40 minutes to complete the mission, which is broken down into two periods: flight time and post-processing time. Flight time is when we occupy the runway or airspace. We shall complete the flight task within 20 minutes. We have planned to capture the waypoints in about 13-14 minutes with a speed of 5 m/s. And while capturing the waypoints or after capturing these (depending on the 3 proximity of the object detection space from the waypoint path), we shall take 5-6 minutes for capturing the images and simultaneously storing them in our memory card fitted on the onboard computer. Post-processing time starts once the UAS has landed. It ends when we stop processing imagery, stop uploading data through interoperability, and return the interoperability network cord to the judges. The limit is set to 10 minutes and we shall utilize the whole of it.

Through the Interoperability System, we shall be given a set of stationary obstacles. Each stationary obstacle will be a cylinder, with height axis perpendicular to the ground, and bottom face on the ground. The cylinders will have a radius between 30ft and 300ft, and there is no constraint on height. There can be up to 30 stationary obstacles. We shall detect, classify, and localize two types of objects: standard and emergent. A standard object will be a colored alphanumeric (uppercase letter or number) painted onto a colored shape. The standard object will be at least 1 foot wide with 1 inch thick lettering. One of the standard objects will be located outside the flight boundary.

The emergent object is a person engaged in an activity of interest. There may be up to 20 objects. We shall utilize the post-processing time of 10 minutes to crop an image containing the identified object.

For standard objects, we have to identify 5 characteristics: shape, shape color, alphanumeric, alphanumeric color, and alphanumeric orientation. For emergent objects there is one characteristic: a description of the person in need of rescue and the surrounding scene.

We shall design an Unmanned Ground Vehicle (UGV) that can be air dropped to a specified location. The UGV will carry a standard 8oz water bottle that will be provided by the judges at setup time. Upon landing, the UGV will drive to another location with the water bottle. This task is planned to be performed within 1 minute.

2. System Design:

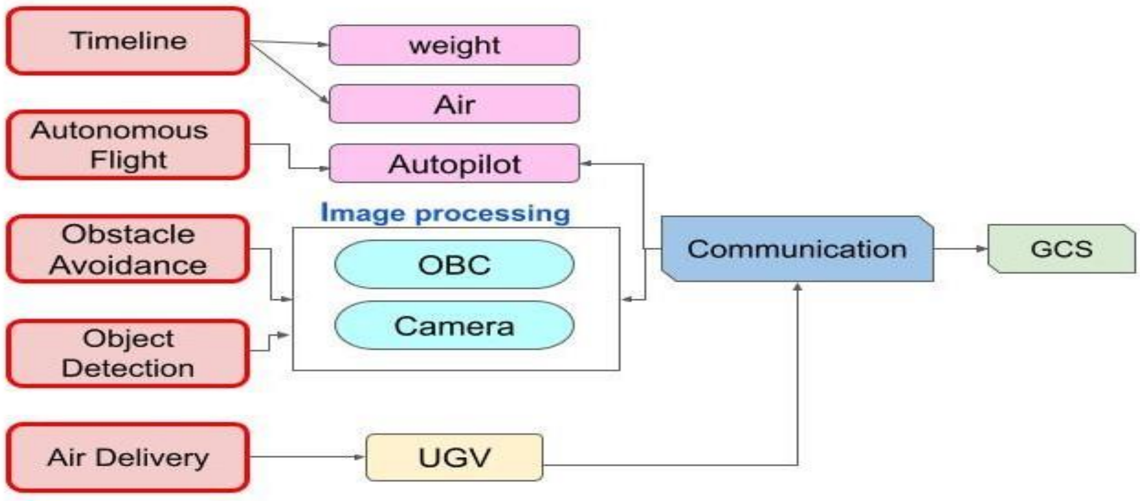
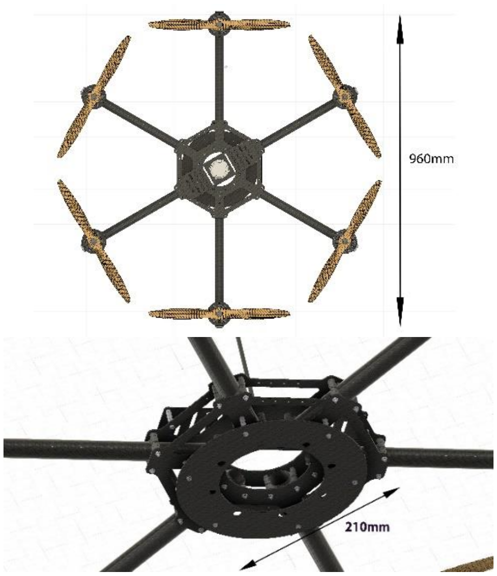


Figure 1 Decison Diagram for hexacopter

2.1 AIRFRAME

In this paper, the discussion focuses on the construction of a heavy hexacopter to be used independent control. It is difficult to design a large hexacopter with the ability to lift heavy weights. In the design of a hexacopter, it is necessary to consider the following: the exact weight can be raised, pressure, car type, propeller type and size, material structure frame, and flight time. Because of these factors, the design is expected to be able to achieve objectives and overcome air traffic control problems and load distribution changes due to the additional load. The next step is to customize the design by supporting objects such as Electronic Speed Controller (ESC), Brushless DC motor, battery, flight controller, GPS and compass and remote control. The frame should be made of heavy materials but is strong enough to support its operational weight and load structure. Therefore, it is necessary analysis of the flexibility and strength of the frame structure.



Dimensions of Fuselage

2.1.1 Hexacopter Configuration:

Hexacopter uses six motors that will rotate six propellers. This hexacopter lifting movement utilizes the thrust that is generated by the propeller combination of the hexacopter frame. The frame configuration is generally recognized as two types: the Plus (+) and X configurations as in Fig. 1. The Hexacopter has 6 degrees of freedom (DOF), where the six degrees of freedom are affected by the rotational speed of each rotor, thus both frames will have different motion dynamics models. Both frame's Configurations are capable of performing different tasks. Since the mission requirement is to carry a heavy load, experimentally it has been proven that X configuration is more preferable for heavy load tasks over plus (+) configuration. The X configuration provides more stability as compared to plus (+) configuration since it has more number of rotors for stabilizing on each side of the vertical bisector. An X Configuration can have a camera pointed forward without obstruction from the frame more easily than a plus (+) configuration. Moreover, the rolling, yawing, pitching moments are easier and flexible in X configuration. Considering these factors, it can be concluded that X configuration is suitable for the mission.

2.1.2 MOTORS

Compared the loose winding motor, the instant current would be lower 15% and the load current would be lower 5-10%, have higher efficiency, and more torque. Standard and delicate winding process. Under the model of fixed connection with ESC and with the requirement that the motor should turn in the same direction .

The bearings used for T-MOTOR MN5212 series are of double size compared with those used for other motors of the same size it increases stability.

Maximum instantaneous current: 31.0 A for each motor,

Each ESC has rating: 40 A(continuous rating);

Burst Current (10s): 60A, So it is compatible with our motor. Voltage rating: 4-6 S LiPo battery i.e. 13.6 V to 20.4

V Our ESC has 2-6S LiPo operating voltage,so it works accordingly with the battery also.Thus we decided to use 40A 2-6S Brushless ESC With 5V 5A SBEC

2.1.3 Propeller

Number of blades: Experiments indicate that efficiency (denoted by thrust/power) of two-blade propeller is better comparing with other propellers. So we are using a two-blade propeller Material: Material includes carbon fiber, plastic, wood and so on, while the propellers made of carbon fiber cost almost twice as much as those made of plastic. Propellers made of carbon.

• Lighter and stronger.

Specifications			
Diameter/pitch	18” ‘5.5	Working Temp	-40℃ to 65℃
Weight	31.5g	Storage Temp	-10℃ to 50℃
Material	CF+Epoxy	Storage Humidity	<85%
Surface Treatment	Polished	Optimum RPM	3000-6000 RPM/min
Propeller type	2blades-integrated	Thrust Limitation	8.2kg

carbon fiber propellers

Thrust Test				
Propeller: 1855 Carbon propeller				
Thrust	**Voltage**	**Amps**	**RPM**	**Watt / kg**
1.32	24	5.7	3596	104.32
1.61	24	7.4	3958	110.48
1.90	24	9.3	4310	116.78
2.26	24	11.6	4622	123.15
2.84	24	16.5	5226	139.51
3.48	24	22.1	5751	152.75
4.36	24	31.0	6358	171.00

2.1.4 Battery

Supplies power to motors and autopilot system via power distribution board.

Operating voltage= 22.2V (nominal) to 25.2 (maximum) i.e. 3.7 x 6 to 4.2 x 6 V Thus, 6S Lipo battery will be optimal.Capacity: For Tarot-5212 motor, We used MATLAB script to obtain the motor characteristics:

Current as a function of throttle:

$$f(x) = -4.5637 \times 10^{-7}.x^4 + 8.9964 \times 10^{-5}.x^3 - 0.0016x^2 + 0.0260x + 0.005$$

(Bi-quadratic expression was chosen because it gave the least standard error of 0.0545 calculated by polyval function)

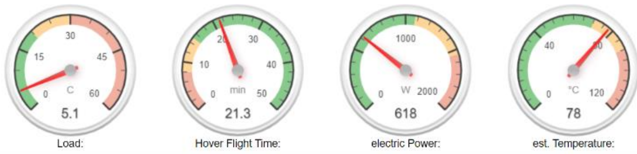
$$Average\ current = 9.3570$$

$$Battery\ capacity = \frac{(Avg.Current\ drawn\ by\ motors\ and\ components) \times (flight\ time)}{(battery\ discharge)} \times 1000$$

$$= (9.3570 \times 6 + 2) \times \left(\frac{30}{60}\right) \times \frac{1}{0.8} \times 1000$$

$$= 36406.25\ mAh$$

This gives us approx. capacity of battery required. Further calculations have been done via ecalc.

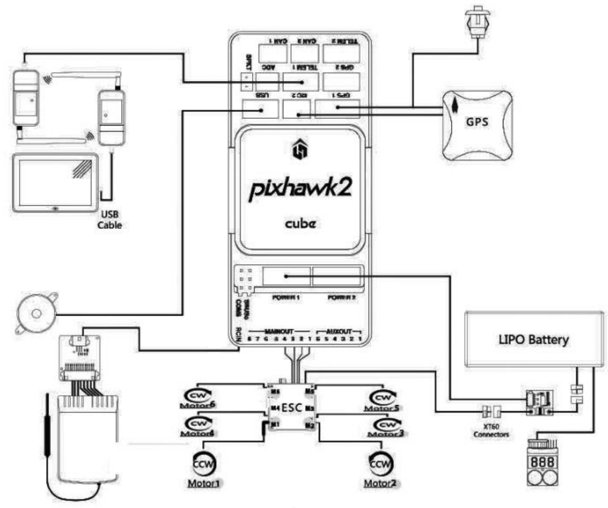


Battery		Motor @ Optimum Efficiency		Motor @ Maximum	
Load:	5.09 C	Current:	22.37 A	Current:	30.57 A
Voltage:	20.23 V	Voltage:	21.11 V	Voltage:	20.22 V
Rated Voltage:	22.20 V	Revolutions*:	6606 rpm	Revolutions*:	6095 rpm
Energy:	799.2 Wh	electric Power:	472.2 W	electric Power:	618.1 W
Total Capacity:	36000 mAh	mech. Power:	402.0 W	mech. Power:	520.1 W
Used Capacity:	30600 mAh	Efficiency:	85.1 %	Power-Weight:	340.8 W/kg
min. Flight Time:	10.0 min				154.6 W/lb
Mixed Flight Time:	19.1 min			Efficiency:	84.1 %
Hover Flight Time:	21.3 min			est. Temperature:	78 °C
Weight:	7200 g				172 °F
	254 oz				
				Wattmeter readings	
				Current:	183.42 A
				Voltage:	20.23 V
				Power:	3710.6 W

Data for ToMAP 6S 18000 mAh 20/40C 2120g battery used as 6S2P

2.2 Auto Pilot

PixHawk's Cube is the autopilot used by the team. It is open-source, which allows us to develop new functions and tasks with great ease. Our flight controller, running the PX4 firmware can change the motor's layout through the signal mixer. The MAVLink protocol makes it easy to send missions to the drone and it supports ''hot mission swapping'' so we can adapt rapidly if we change our mission priority during the flight. It integrates well with QGroundControl (QGC), our current ground station software. Based on QT C++ framework. It is also cross-platform and can run on Linux, macOS, ios, Windows, and Android. With the help of QGC, we can generate a mission that can be executed by the aircraft. Since we are using a multirotor, the multirotor can do a 180 degrees turn on itself. In the goal of increasing precision, we will be using an RTK (Real Time Kinematic) with EKL which allows having a maximal error of 6.5 ft. on the position. It can be integrated into the QGC software. With each drone iteration, we have to reconfigure our autopilot. Here's our procedure: As soon as the drone is assembled for the first time, we connect to it via QGC. Following that action, we start the necessary calibrations such as the gyroscope, accelerometer, level horizon and the compass. Next, we upload the configuration of the prevision version of the autopilot and we test our new platform in our inside flight cage. We can thus directly see and correct details and adjust parameters in order to obtain the desired behavior We use this interior flight cage each time that we need to test a new parameter. We can also use it to test different platforms as well as to train new pilots and ground station operators. Before each flight, we simulate the accurate version of the way in which we will going to fly our drone for testing and repeat the calibration of our autopilot to be sure that it still calibrated and so we avoid many potential problems. The below-shown area is the total flight boundary in which our drone has to flight.
We will use Pixhawk Cube for our drone. The Pixhawk cube is the most advanced open auto pilot system designed to meet the needs of the drone which utilises the incredible ArduPilot firmware.Autopilot offers great redundancy on many sensors. It has triple IMU and ability to blend two GPS signals and the ability to have three magnetometers also provide temperature regulation.

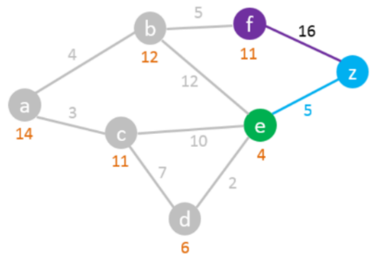


Advanced Pixhawk Hexacopter Wiring Chart

2.3 Obstacle Avoidance

We will be dealing with two major obstacle avoiding scenarios, for one we need to dodge the predefined stationary obstacles and for the other we need to dodge the dynamic obstacles,namely the other drones.For the first case,the gps locations of the objects will be used for deciding the shortest path using the mission planner software.For detecting the incoming drones,we will be using a Lidar based system and we will develop a dodging algorithm where we climb or descend depending on the position of other drone to dodge.

A * is a computer-based algorithm that is widely used in graph traversal and pathfinding, which is a process of finding a path between multiple points, called "nodes". It enjoys widespread use because of its functionality and accuracy. However, in practical navigation systems, it is usually outperformed by algorithms that can process the graph beforehand getting better performance, even though some work has found that A * is better than other methods. A * is a knowledgeable search algorithm, or advanced search, which is based on weighted graphs: starting at a specific point of the graph, aims to find the path to a given goal node with the least cost (minimum distance, shorter distance, etc.). It does this by keeping a tree of paths starting from the root node and extending those paths one edge at a time until the termination condition is satisfied.



Node	Status	Shortest Distance From A	Heuristic Distance to Z	Total Distance*	Previous Node
A	Visited	0	14	14	
B	Visited	4	12	16	A
C	Visited	3	11	14	A
D	Visited	10	6	16	C
E	Visited	12	4	16	D
F		9	11	20	B
Z	Current	17	0	17	E

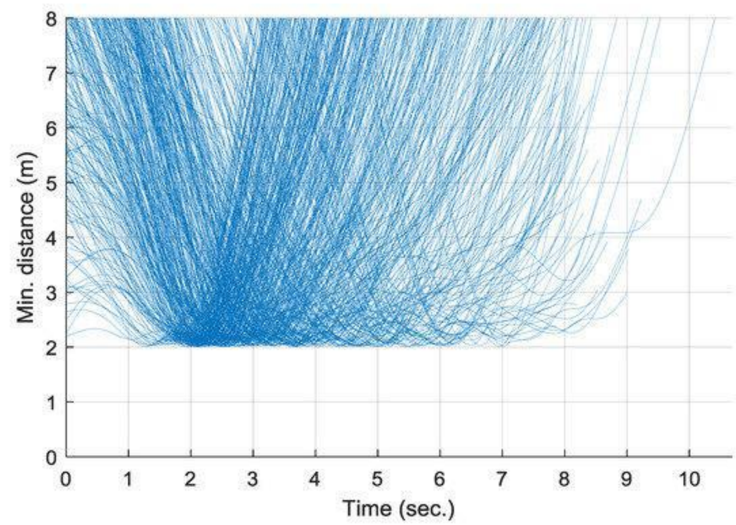
* Total Distance = Shortest Distance from A + Heuristic Distance to Z

Working of A algorithm*

LIDAR (Light Detection and Ranging) consists of a laser, a scanner, and a specialized GPS receiver. We will be using two LiDARs one mounted 6.098 degrees below the horizontal axis and another at 6.098 degrees above the horizontal axis to detect the whole geographical view.To scan the entire 360 degree view the two Lidars will be rotating in phase at the rpm of around 833, scanning at gaps of 10 degrees after 0.001s, which is the data transfer/scanning rate of the lidar we are using. The specifications of the same are as follows.

Range of LiDAR	40m
Operating Voltage	5V
Maximum speed of drone	9 ms^{-1}
Average velocity of approach	14 ms^{-1}
Maximum critical sensing time	0.036 s
Wire latency	0.0535 s
Critical interrupt time after detection	0.0895 s
Critical distance for dodging	2.8532 m
Angle between lidar and axis	6.098 degrees

We have simulated Lidar sensors using these values through Gazebo + ROS. We designed a set of paths during navigation to take into account the obstacle velocity, which was estimated by a Kalman filter. The estimated velocity was used to predict the obstacle's position within a tentacle-based approach. We developed nonlinear geometric guidance and differential geometric guidance, based on aiming point guidance. The guidance law of the UAV was made applicable for moving obstacles by incorporating the Point of Closest Approach (PCA). It is a formation control problem where other UAVs are treated as moving obstacles. A nonquadratic avoidance cost function was constructed by an inverse optimal control approach so that the optimal control law was obtained in an analytical form.



Time history of minimum distance between UAV and obstacle

2.4 Imaging System

2.4.1 Camera

For the camera Sony A5100:
Sensor height = 15.6 mm, sensor width = 23.5 mm, Height of camera = 150 feet = 4572 cm focal length = 16-55 mm Camera resolution = 6000px x 4000px weight = 281 grams.
So the GSD comes out to be So the nominal GSD would be 1.11 cm/px which is very close to our desired value of 1.06 cm/px. We also searched for some other cameras such as GoPro HERO8 12MP with GSD = 0.42 cm/px,Panasonic Lumix DMC G85H 16MP with GSD = 1.15 cm/px but GSD do not matched our desired value. Thus the final conclusion comes out to be SONY A5100.

2.4.2 Gimbal

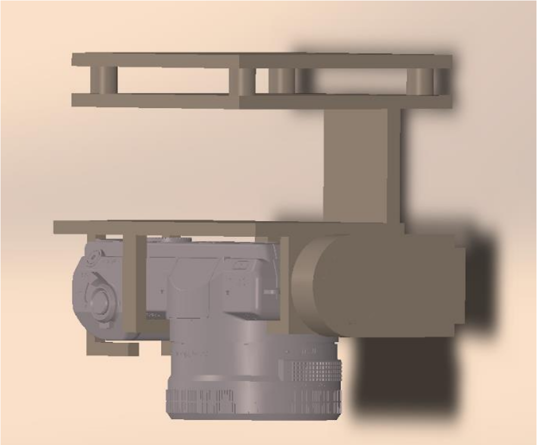
A gimbal is a support system that allows an object to remain horizontal regardless of the motion around it.

Choosing the right gimbal:

Number of axes: Gimbals for drones are either 2-axis or 3-axis. 2-axis gimbals do not compensate for yaw, which means that there will be slightly more bump in your images. Gimbals with three axes are generally heavier

than their 2-axis counterparts due to more components. A heavier gimbal means shorter flight times.
Camera support: Gimbals are designed with specific cameras in mind. Our gimbal was designed to hold the Sony A5100 camera.

Material Used: Material should be Lightweight, have high strength, easy availability and 3D printable. . PLA and ABS both suited the following criteria. ABS was chosen due to its better resistance to wear and temperature fluctuation.



3-D model of Gimbal

2.4.3 Gimbal Controller:

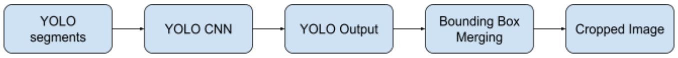
Gimbal controller is mainly used for controlling the motion of gimbal. It depends on the axes of the gimbal used. For eg. 2 axis gimbal controller is compatible with 2 axis gimbal and 3 axis gimbal controller is compatible with 3 axis gimbal.
It is simply a driver that controls motion with the help of a software suitable for that particular gimbal controller. So, as per our requirement and weight limitation, we will be using 2 axis gimbal that is 3D printed by ABS and 2 axis gimbal controller available commercially. The software used will be simple BGC to operate a gimbal controller.

2.5 Object Detection, Classification, Localization.

2.5.1 Detection

For detection of the region of interest, we used **YOLO** (You only look once). Images from the camera are sent through the YOLO detection process which outputs cropped images to be sent through the **CNN** array for classification.

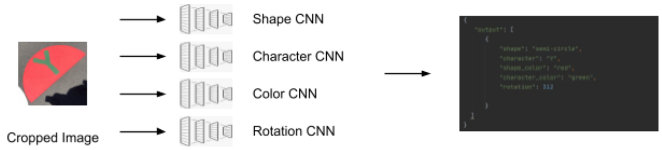
The input image (6000X4000 pixels) is divided into square segments of 400 pixels. These segments are passed through YOLO CNN. This returns a bounding box for each segment and a confidence value for whether there is a shape contained in the segment. If a shape is contained in multiple YOLO segments then the predicted bounding box extends out of the boundary of the segment. Later these bounding boxes are merged to form a single bounding box containing the shape (the region of interest). Finally, a 120 pixel cropped square image is produced which is to be further passed to the CNN array for classification.



Working of YOLO

2.5.2 Classification

For shape, color, orientation, and letter classification. We use **separate CNNs** that take in the cropped image from the YOLO network as input. To get the CNNs to detect, classify, and localize in **real-time** on the OBC, **knowledge distillation** (Knowledge distillation is the process of creating a large CNN that undergoes standard training and then using the large CNN to teach a smaller CNN) was utilized to reduce the network size and execution time. The teaching session involves executing the large, trained CNN on a particular input; The same input is then given to the small CNN which is trained on the large CNN's result.



Classification of CNN Array Diagrams

2.5.3 Localization

For the localization of the object the actual location of the UAV and center point values of the original frame is certain, then the localization algorithm must continue with determining the center point of ROI. After identifying flight direction, pixel difference between the center points of original photo and ROI, and equivalent actual location value of pixel difference, real coordinates are found.

Distance Calculation

The distance between the target and the UAV is calculated in terms of pixels. Using Euclidean geometry, the slope and the angle of the target relative to the current heading are calculated.

$$horizontalDistance = (targetX - currentX) * gsd$$
$$verticalDistance = (targetY - currentY) * gsd$$
$$totalDistance = \sqrt{}(horizontalDistance^2 + verticalDistance^2)$$

Bearing Calculation

We then calculate the angle the target makes with the UAV's heading.

$$slope = \frac{targetY - currentY}{targetX - currentX}$$
$$angle = \tan^{-1}(slope)$$

Location Calculation

To calculate final coordinates, we need the current coordinates, distance to target, and bearing of the target. We can calculate the bearing relative to the UAV and then add it to the UAVs heading to get absolute bearing. This data is sent to a python library called geopy whose ellipsoid function can be used.

$$bearing = angle + currentHeadingOffset$$
$$coordinates = Ellipsoid(currentLat, currentLong, totalDistance, bearing)$$
$$targetLatitude = coordinates.latitude$$
$$targetLongitude = coordinates.longitude$$

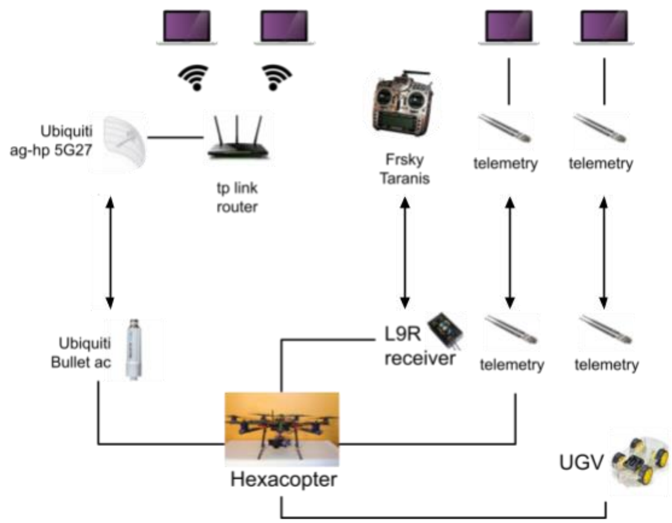
2.6 Communications:

Implementing a solid communications infrastructure has proven to be the most crucial part as it includes the connection between the ground station and the UAV and in turn submitting the data to the interop server received from UAV. With efficient design, and correct use of automation only, the reliability of ground and air equipment can be ensured.

2.6.1 Aircraft communication:

We are using a Ubiquiti bullet AC(Model: B-DB-AC) for communicating with the GCS.The module will be used to send the processed(cropped) image obtained by Processing from OBC at the frequency of 5GHz.The cropped image has a size of 300 X 200 pixels,which translates to a size of about 1.44Mb.The data transmission rate of Bullet ACis 300+Mbps,which gives the transmission time less than 0.0048s. But when we include the transmission latency of the wires from Camera to OBC and OBC to ubiquity,we get the time as 0.011s. Taking in account the processing time and latency of the module itself the time we get is much less than 0.1s. The module can also act as a backup communication channel for telemetry in case of failure of the rfd900+ module. At the GCS, we are using Ubiquiti Airgrid M5 HP (Model: AG-HP-5G27) with 27dBi sector antenna

We are using Telemetry RFD 900+ operating at 915 MHz frequency range to send and receive data between our drone and the ground station. The ardupilot autopilot system allows android devices to be used as your ground control station. It is compatible with pixhawk ,holds data link at over 40 km and have a universal asynchronous receiver transmitter baud rate of 57600 and air data rate of 64kbps.The time taken for transmission between each packet is 0.033s,while transmission time in air is about 0.028 s which puts total transmission latency at around 0.061s.



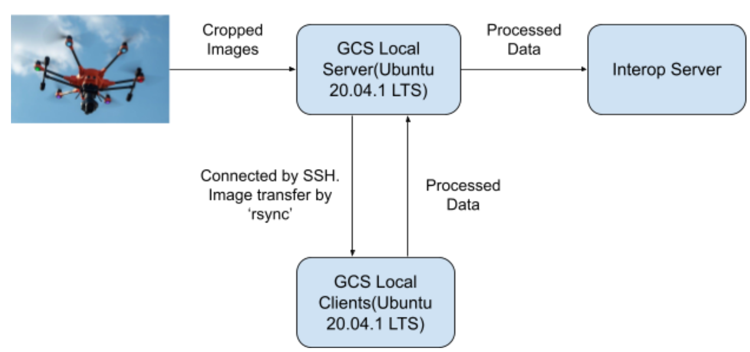
Communications Block Diagram

2.6.2 Antenna Tracker:

Team Aries developed an antenna tracker to ensure that it consistently points towards the UAV. It is able to move in horizontal and vertical planes using two servos. Telemetry RFD 900+ will be sending us GPS coordinates of the drone though which, we will get the height from ground level and distance of the drone from the GCS. Using this information and applying Pythagoras Theorem, we can get the aerial distance and angle of elevation of the drone and point our antenna tracker automatically.

2.6.3 Software Communications:

Image files from the UAV will be received at the ground server station. Client computers of the local network will be connected to the local server using SSH(secure shell) communication protocol. Sampled images will be automatically synced to client computers for further processing. The local server will be connected to the interop server and the processed information according to the required format will be provided to the interop server. Till now, we are able to manually submit the data and the automatic solution is under test phase.

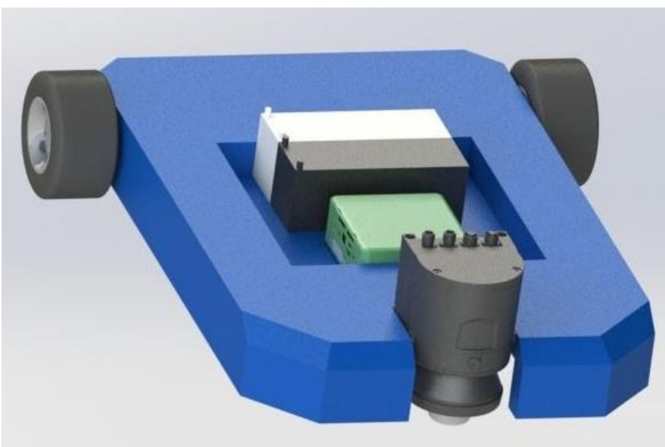


Software Communications

2.7 Air Drop

2.7.1 UGV(Unmanned GROUND VEHICLE):

The UGV is a custom-designed vehicle holding a bottle up to the delivery location. It is the frame is made from carbon fibre to have minimum weight with maximum strength. It utilizes DC motors to transport the 8 oz(236gm). water bottle. The water bottle is held firmly in place by the custom 3d printed supports and by some rubber bands as an extra precaution. It would be a three-wheeled ugv with a wheel size of 7x2 cm each. In order to reduce the weight, the team decided to use a 600mAh LIPO battery, giving us the minimum power required to perform the mission. An 8-bit Arduino type microcontroller will be controlling the motors, GPS/IMU module as well as the telemetry module. Once on the ground, the GPS will report the position of the rover and the telemetry will allow communication at a distance. Servo-motors allows the rover to separate itself from the rope hooked to the drone once it is on the ground. The UGV drive up to its destination is completely autonomous. The aforementioned UGV components were once again chosen to reduce weight and overall dimensions all while keeping the solution as versatile as possible



3-D model of UGV

2.7.2 AIR DELIVERY:

UGV will be airdropped with the help of the winch mechanism attached to UGV. The drone will be hovering over drop location while UGV is being released. A DC motor attached to winch will be used to control the descending speed of UGV, to prevent damage due to excessive speed. A nylon thread will be used for descending UGV. It will be attached to the drone with the help of an actuator and pin which will release pin after UGV has touched the ground. An ultrasonic sensor attached to UGV would be used to detect the correct time for the release of a pin. The DC motor is selectively chosen to have a higher stall torque to sustain heavy loads. The drawback of a heavy DC motor having a low RPM was compensated by the higher stall torque as well as precise drop delivery by having a controlled drop mechanism.

3. Alternative Considered:

3.1 Copter configuration

The differences in the multirotor type related to the difference in the ability in lifting the load due to the addition of a motor, while poses a new problem that causes the system to become more nonlinear and diffcult to control. Since the mission is lifting a heavier load, thus the main choice of the multirotor type is hexacopter or octocopter. Although the octocopter has the ability as a hexacopter, it cost more price due to the more number of motors. Therefore, hexacopter is selected to be designed as a heavy-lift carrier.

3.2 Autopilot

	Pixhawk 2	PIXHAWK CUBE
Support Fixed Wing and Multi-copter, VTOL	YES	YES
Open development Environment	YES	YES
Modular Cube Design	NO	YES
Dampened IMU	NO	YES
3 times redundancy IMU	NO	YES
CM level GPS	NO	YES
Multiple GPS system	NO	YES
Allowing flights at extreme temperature	NO	YES

3.3 Propellers

Adding more blades to a propeller also increases its drag, and mass making the motor work much harder to spin the prop, which results in a higher amp draw, lower flight time, and greater strain on the battery.

3.4 Camera

Other reason for not selecting the mentioned cameras because:
1. The weight should be as low as possible.
2. We do not need GSD value to be lower than 1.09 cm/px at the cost of weight.
3. Too low GSD values are also not appreciated as more than one image might be needed for the detection of a single object. Plus, the field view would be too less

4. Testing and Evaluation Plan

4.1 Development testing

4.1.1 Motor testing: To fully characterize a motor, you need to measure the following parameters: Voltage, Current, Throttle input, Motor load or torque, Speed .We will use RCbenchmark software that automatically calculates the Mechanical power, Electrical power, Motor Efficiency.

4.1.2 Propeller testing:
For extracting useful propeller data, we need to measure the following parameters:Speed, Torque, Thrust. We will use RCbenchmark software to calculate the Mechanical power, Propeller efficiency. Notice that the mechanical power is the same for the motor and propeller. That is because all the motor's mechanical power output goes into the propeller, since it is directly coupled to the motor's shaft.

4.1.3 Flight Testing:
For initial Flight testing we divided the flight into different part landing and takeoff ,hover and autonomous flight.All these modes will be tested separately first in simulation(Gazebo in ROS) and then in the open field. Flight time will be evaluated in each mode test battery.

4.1.4 Obstacle avoidance:
Both static(A*) and dynamic(LiDAR) obstacle avoidance system will be first simulated in Gazebo. If this works, the system will be tested in open field against stationary obstacles and a test drone

4.1.5 ODCL:
We use open field images with random shapes superimposed on top containing different letters with different rotation. This formed our test set on which YOLO algorithm is run whose output is then used as input to our CNN mode. Also we will test our localization algorithm indoors for the development phase.

4.1.6 Communication:
We would test our system by sending images from the UAV and measure the time delays in data transfer. Latency issues in processing images at the GCS will be considered. Forwarding the tasks provided by the interop server to the UAV needs to be very efficient. We have already set the dummy interop server (with the code available on github) and a complete test of receiving and sending missions is in the pipeline. Completing these tasks in a given time limit is our first priority.

Air Drop:.The UGV will be dropped from a height of 15 meters and time would be noted. From this estimated time required for UGV to descend from actual drop height would be calculated. Stability of Drone for air drop would be determined by hovering the drone at 15 meter for estimated time of actual airdrop.

4.2 Mission testing

4.2.1 Drone flight testing: On the test run drone will takeoff above 100ft MSL within the first few minutes keeping the mission clock in mind we have planned to capture the waypoints in about 13-14 minutes with a speed of 5 m/s and during this time interval we try to perform all the mission tasks.

4.2.2 Obstacle avoidance: The whole system will be tested at a height of about 150 feet against ten stationary obstacles and two times against a test drone

4.2.3 ODCL: Objects similar to our test set will be made and placed randomly in an open field. Using drone we perform the task of localisation and detection on the objects whose results will be then sent to our communication team in real time to maximize our score for conducting the mission as much as possible.

4.2.4 Communication: we will check range and bandwidth under different weather conditions and in the presence of obstacles like trees and buildings to get the most pessimistic range and data speed.

4.2.5 Air Drop: It would be conducted in open ground without any obstacles .For safety purposes, UGV would be dropped from a height of 150ft. Hover time of drone over drop location = drop time determined from testing + 10 seconds (buffer time).
Various test drops of UGV would be conducted to check closeness of dropped UGV to drop location.

5. Safety, Risks, & Mitigations

Safety is fundamental to the development and operation of a UAS. This section describes potential safety risks and details steps taken to mitigate them. Risks are classified by likelihood of occurrence: Rare, Infrequent, and Frequent. Severity is classified by: Low, Medium, High.

5.1 Developmental Risks & Mitigations.

The development process poses several risks. Following table summarizes the safety issues and mitigation measures adopted to tackle them.

Development risks	Occurrence	Severity	Mitigation
Ineducate training of new team members	Rare	High	Pre-training and safety demonstration for new members. Wearing protective equipments is compulsory for all the new members. Senior members of the team always present to supervise their work.

Budget Issue	Infrequent	Medium	Once the needs of the competition have been identified, we plan what we will need for the rest of the year. We create an accurate budget based on the needs of every subteam leaders with a planned buffer to anticipate any surprise costs.
Injury due to hazardous components	Rare	High	Wear PPE kits including gas masks, protective glasses, anticut safety gloves, etc. Batteries stored in metal cabinets.
Drone crashes/collisions	Rare	High	Testing was mostly done inside close rooms free of any humans. Policies including requirements for operator training, recurrent pilot proficiency examinations, vetting and auditing, equipment inspection and maintenance, and detailed record keeping are implemented.

5.2 Mission Risks & Mitigations.

While operating an aircraft, the team must consider many direct and indirect risks. A list of the most common risks and the solutions required to mitigate them are presented in the following table

Mission risks	Occurrence	Severity	Mitigation
Loss of telemetry or RC link with UAV	Infrequent	Medium	Pre-mission range test and Return to Land failsafe
Aircraft not following the predetermined path	Rare	High	Calibrate our autopilot as we arrive on the flight zone. Confirm that the real position of the aircraft correspond to the position we have on our monitor. If it drifts while in the air, take control back from the autonomous mode and manually operate it.
Battery die out	Rare	High	If battery level falls below the threshold of 15 percent, Activate Return to land failsafe.
Pilot error	Rare	High	Air and temperature conditions should be checked before flying. Pilot should not be distracted before and during drone flight.
Injury from propeller	Rare	High	Strict pre-flight safety procedures. Safety switches are incorporated including a mechanical one.
Accidental UGV release	Rare	Medium	A series of locking mechanisms that prevent accidental release of the UGV

6. References

1. https://www.movable-type.co.uk/scripts/latlong.html
2. Jongho Park, N. C. (2020). Collision Avoidance of Hexacopter UAV Based on LiDAR Data in Dynamic Environment. *Trends in UAV Remote Sensing Applications*, 20.
3. https://www.rcbenchmark.com
4. https://www.ecalc.ch