Mavericks UAS

Fr. Conceicao Rodrigues College Of Engineering

University Of Mumbai

Hex-Wife Technical Design

AUVSI SUAS 2019

Abstract

Team Mavericks UAS of Fr. Conceicao Rodrigues College of Engineering (Aff. University of Mumbai) has designed an efficient, modular UAV system to participate in the AUVSI SUAS 2019 competition, called the Hex-Wife. It is a multi-rotor with an excellent power to weight ratio, a precisely tuned autopilot and a state-of-the-art communication system. The Hex-Wife has been engineered by a young team of only seven members over a span of 18 months. Mavericks UAS has rigorously tested the Hex-Wife in various scenarios to ensure its stability and reliability while complying to all safety regulations.

1. Systems Engineering

The Hex-Wife was designed to be power efficient and cost effective while being able to perform autonomous missions with great reliability. The system is also able to achieve static obstacle avoidance, object detection, classification and localisation of ground objects. The following section looks at the mission requirements and gives an overview of the various decisions and trade-offs that were considered on the road to building the final model.

1.1 Mission Requirement Analysis

Table 1.1.1 Mission Requirement Analysis

Task + Marks	Description	Requirement Analysis
Timeline (10%)	• Complete the mission within the directed time (80%) • Minimize any timeouts taken (20%)	• The UAS should be able to travel 6-7 miles in 30 mins or less • The ODLC system should be efficient and computationally powerful enough to finish the post processing in 10 mins or less
Autonomous Flight (20%)	• Achieve minimum 3 minutes of autonomy (40%) • Waypoint within 100ft. (10%) • Waypoint accuracy (50%)	• An autopilot capable of navigating through waypoints, auto take-off and land • An agile airframe that can efficiently manoeuvre through extremely small angles while minimizing the distance travelled
Obstacle Avoidance (20%)	• Avoiding stationary obstacles (100%)	• An Algorithm that adjusts the path that the UAV takes to navigate through the waypoints such that the UAV avoids any obstacle it might collide with
ODLC (20%)	• Identify characteristics such as shape, shape colour • alphanumeric, alphanumeric colour, alphanumeric orientation (20%) • Provide GPS location of object (30%) • Object Submission via Interop server (30%) • Autonomous Submission(20%)	• An accurate classifier for classifying objects which should be computationally cheap enough to be run on the ground station. • The IP system should be capable of resolving images at minimum height of 150ft. • The UAV should be very stable in flight to ensure minimal vibrations for the IP system. • The system must provide accurate data to calculate the GPS coordinates of the objects • Image processing server should autonomously submit objects to the judges.
Air Drop (20%)	• Accuracy of Dropping (50%) • UGV Driving to Location and reaching within 10ft (50%)	• The UAV must be equipped with a mechanism which releases the package upon receiving confirmation from the judges
Operational Excellence (10%)	• Operational Professionalism, Communication, System Failure Reaction, Attention to Safety	• The team should be ready and well equipped to deal with any system failure • The team should have professional conduct and good communication between members and the organisers • Multiple practise drills should be conducted to be ready for the actual mission to be performed in the competition

1.2 Design Rationale

1.2.1 Environmental Factors

The seven member team comprises of six computer science and engineering students and one production engineering student. This makes any task under the field of computer science fall under the team's area of expertise and while the team does possess CAD and modelling skills, they are not substantial enough. Our institute also does not have a Mechanical or aerospace engineering department. The Hex-Wife was designed under these constraints to maximize mission performance.

1.2.2 Mission Requirements and Decision Flow

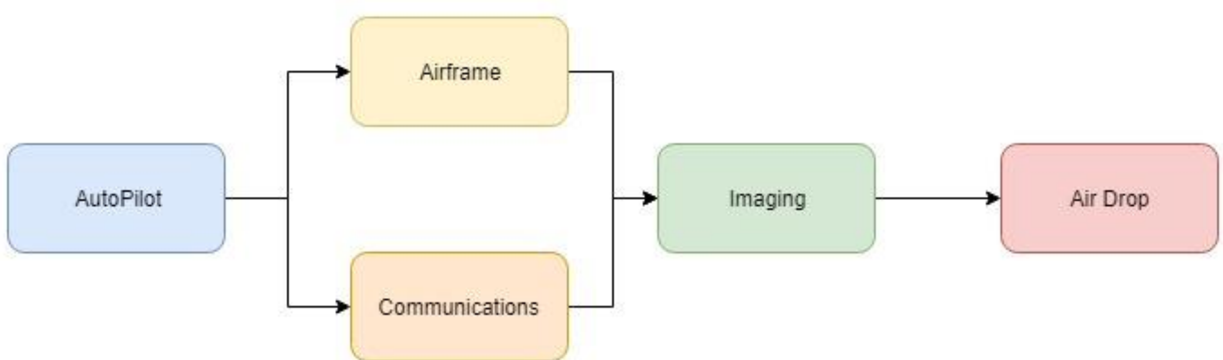
The autopilot will be responsible for the entire autonomy task which would entail Waypoint Navigation, Obstacle Avoidance, Telemetry Communication and Safety Measures. Prior to SUAS 2019, the team had been working with the Pixhawk and the PX4 firmware for an extensive period of time. This coupled with the fact that Pixhawk and PX4 are open source hardware and software platforms which offer a reliable community support made it an excellent choice for the autopilot system. This was the primary decision around which all the other systems have been designed.

The second decision that was taken was the choice of frame. Given that waypoint navigation is a major task in the competition, the choice for airframe was heavily influenced by it. The team decided to use a multirotor over a fixed wing as described in section 2.1. Briefly, multi-rotors offer more agility over fixed wings and provide additional capabilities such as hovering which is particularly useful in imaging.

The third decision was the architecture of the communication system as it is the backbone of all flight systems. Image processing is carried out on the ground station on the image processing server which made it critical to have a reliable bidirectional radio link. The team has been experimenting with long range 5GHz AC Wi-Fi systems for the past year and it proved to be adequate for the task. The main telemetry link is implemented using a 433Mhz UHF radio system which was found to be more than capable and reliable for transmitting mission critical data between the UAV and the GCS. The safety pilot can take control over the UAV using the 915MHz UHF self-healing radio system which is the most reliable link in the whole system. (Refer section 2.6)

Having a robust and precise imaging system was a crucial requirement of the mission. It was hence decided to use a Raspberry Pi Camera V2 as it is light weight, energy efficient and has a good resolution. A camera with a larger sensor would have drawn more power and would require a bulkier gimbal. Having a multi-rotor over a fixed wing also gave the advantage of hovering which improved our image quality by minimizing camera vibrations. Thus, the fourth decision was the architecture of the imaging system.

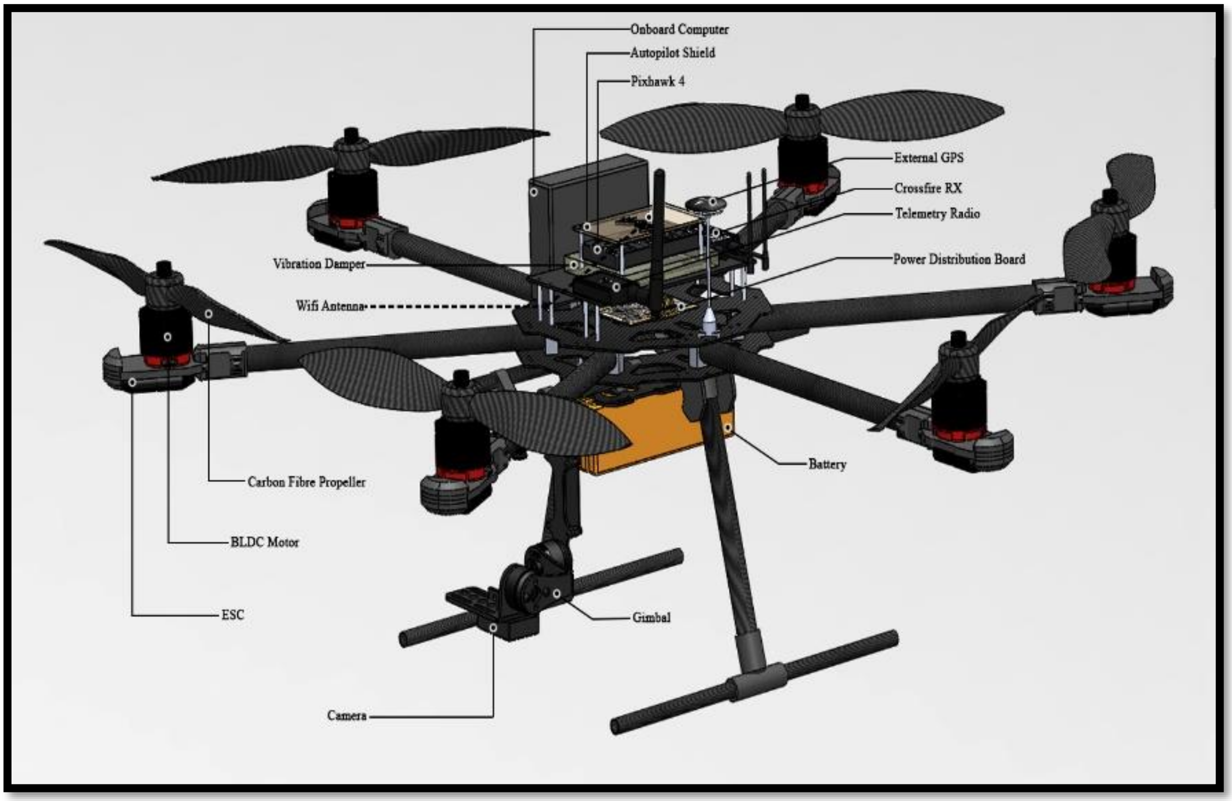
Finally, the team decided to not attempt the airdrop task i.e. the actual drop and the UGV tasks this year for the reasons mentioned in section 2.7.



Decision Flow

2. Systems Design

2.1 Aircraft



The Hex-Wife Model

2.1.1 The Frame

The TAROT 680 Pro frame was chosen due to its modular nature - flight components can be swapped and tested very easily. Additionally, the frame is primarily made of carbon fibre which has an excellent strength to weight ratio, thus reducing the thrust required for lift off. While a fixed wing aircraft would excel in flight endurance and aerodynamic efficiency, the team decided to use a multi-rotor due to its superior manoeuvrability, redundancy in the case of motor failures, stability during imaging and general ease in any fabrication need .The multi-rotor is equipped with two carbon fibre booms with a length of 10.826" further referred to as main arms. The main arms are stationary and are perpendicular to the heading of the UAV. The remaining four arms are also made from carbon fibre with a length of 10.315" further referred to as auxiliary(aux) arms. The arms converge to the main body of the UAV which consists of two plates. The top plate has a riser upon which the Autopilot, Communication and Imaging Systems have been mounted (Ref. figure 2.1). The section between the riser and the top plate houses the power distribution module. Due to the strong electromagnetic effects, no sensors are placed here. The GPS and the compass are mounted on an elevation, thus isolating it from the power distribution module. The bottom plate consists of landing gear mounts, rails and a gimbal for the camera. The underside of the bottom plate has clamps for mounting the rails. The battery plate is clamped to these rails. The battery is secured with two Hook and Loop fasteners. This system has been vigorously tested and it has been determined that the battery will not fall off even if the UAV is travelling at a speed of 27 Knots. The system ensures quick swapping of batteries. The camera will be mounted on a 2-axis 3D printed gimbal. The landing gear consists of foldable arms with each arm measuring 7.283" in length.

2.1.2 Propulsion System

The propulsion system consists of electronic speed controllers (ESCs) and Brushless DC (BLDC) motors. The system is powered by a 6 cell Lithium polymer (LiPo) battery system. The battery is connected to the power distribution module(PDM) capable of monitoring and reporting power utilisation of the onboard systems to the GCS. The BLDC motors chosen for the UAV are the EMAX MT3515 650kv along with the hobby wing 40A ESC. Different BLDC-ESC pairs were put through various thrust, power draw and vibration tests. These tests concluded that chosen combination of BLDC and ESC gave the best performance per Watt.

Table 2.1.2.1 BLDC_ESC Test Results

BLDC Motor	ESC	Max Thrust (lbf)	Current Draw (A) at 4.409lbf	Vibrations
EMAX MT3515 650kv	HobbyWing 40A ESC	6.173	23.8	Low
SunnySky V3508 700kv	HobbyWing 40A ESC	5.071	26.3	Medium
SunnySky V4006 740kv	HobbyWing 40A ESC	5.732	23.4	Medium

All tests were done with 15 inch Carbon Fibre propellers and a 4 Cell Lithium Polymer Battery. It was later decided to move to a 6 Cell LiPo system as it made the system even more efficient. The 6 Cell LiPo battery was chosen after comparing multiple battery systems for the Hex-Wife. It was observed that the current flowing through all the onboard wires reduced significantly by increasing the operating voltage (moving from 4 Cell to 6 Cell LiPo).

The batteries used are covered in duct tape for protection and bright orange tape for locating it in case of a crash. They are always kept at storage voltage in LiPo safe cases.

Table 2.1.2.2 Battery Test Results

Battery	Capacity (mAh)	Weight (lb)	Max Discharge (A)	Capacity/Weight (mAh/lb)
TATTU 6S	6000mAh	1.840	210	3260.87
TATTU 3S x 2 (In series)	10000mAh	3.086	250	3240.44
TATTU 4S	5300mAh	1.208	185.5	4387.41

2.2 Autopilot

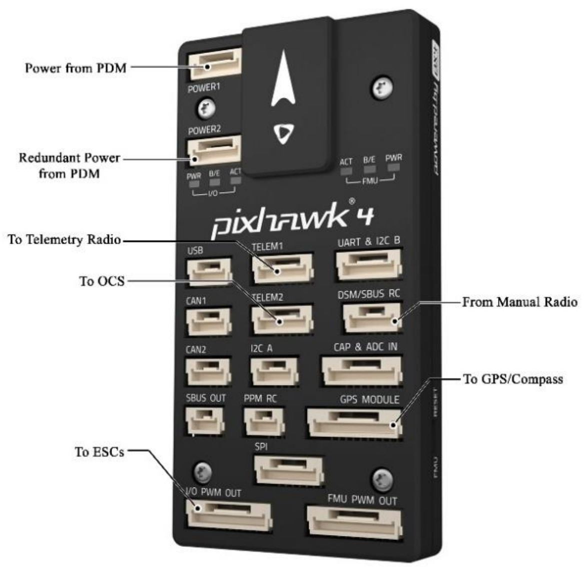


Figure 2.2.1 The Pixhawk

The Pixhawk uses a 32-Bit Arm Cortex M7 processer for as the main Flight Management Platform (FMU) processor and a 32-Bit Arm Cortex M3 as the IO processor. These two processors combined with an array of high accuracy sensors and low latency intercomponent communications ensure good performance at all times.

The Pixhawk supports both ArduPilot and PX4 firmware. Both of these are capable of autonomous take off, land and waypoint navigation thus eliminating the penalty incurred for a manual takeover. After testing both of them extensively for flight stability and waypoint accuracy, it was observed that they perform similarly in those aspects. Because multi-copters are highly manoeuvrable, both the firmware consistently trace all the waypoints with zero misses.

.

Figure 2.2.2 Ground Control Station

The PX4 firmware also supports DronecodeSDK which has been used to send data to the interop system. Also, Mission Planner, the companion GCS for ArduPilot, is not available for macOS or Linux which was also a minor factor in choosing the firmware. Finally, PX4 missions can be stored as JSON files which can be directly imported into QGroundControl which was a crucial factor as it makes converting mission items from the interop system into the firmware supported format very convenient and ties up perfectly with the implementation of the obstacle avoidance algorithm. Hence PX4 Firmware was chosen.

2.3 Obstacle Avoidance

The obstacle avoidance algorithm has been designed by the Autonomy team under the guidance of the Professors of the Dept. of Computers. The Algorithm has been extensively tested for various test cases.

The algorithm has been designed to take mission data from the interop system and check for collisions with static obstacles and find a route around them w.r.t. the constraint of navigating through all the waypoints. The algorithm outputs a mission file that has the rerouted paths. This file is then loaded in the GCS which uploads it to the Autopilot.

2.3.1 Detecting Collision

We have a line segment from $\boldsymbol{p_1}$ to $\boldsymbol{p_2}$ and we want to find the points of intersection with a circle centred at $\boldsymbol{q}$ and radius $\boldsymbol{r}$

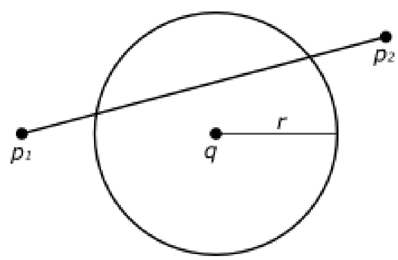


Figure 2.3.1

Any point on the line segment can be written as $\boldsymbol{p_1 + t * (p_2 - p_1)}$ for a scalar parameter $\boldsymbol{t}$ between $\boldsymbol{0}$ and $\boldsymbol{1}$
Assume $\boldsymbol{v = p_2 - p_1}$
Now, a point x is on the circle if its distance from the centre of the circle is equal to the circle's radius, that is, if

$$|\boldsymbol{x - q}| = |\boldsymbol{r}|$$

So, the line intersects the circle when

$$|(\boldsymbol{p_1 + t * v}) - \boldsymbol{q}| = |\boldsymbol{r}|$$

Squaring both sides gives

$$|(\boldsymbol{p_1 + t * v}) - \boldsymbol{q}|^2 = \boldsymbol{r^2}$$

and now we can use a property of the dot product (namely $|A|^2 = A.A$ for any vector A) to get

$$(\boldsymbol{p_1 + t * v - q})(\boldsymbol{p_1 + t * v - q}) = \boldsymbol{r^2}$$

Expanding the dot product and collecting powers of t gives

$$\boldsymbol{t^2(v.v) + 2t\big(v.(p_1 - q)\big) + (p_1.p_1 + q.q - 2p_1.q - r^2) = 0}$$

which is a quadratic equation in t with coefficients

$$\boldsymbol{a = v.v}$$
$$\boldsymbol{b = 2v.(p_1 - q)}$$
$$\boldsymbol{c = (p_1.p_1 + q.q - 2p_1.q - r^2)}$$

and solutions

$$t = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

If the discriminant $\boldsymbol{b^2 - 4ac}$ is negative then the line does not intersect the circle, if it is 0 then the line is a tangent to the circle and if it is positive then the line intersects the circle.

If the value of t is in the interval, $[\mathbf{0}, \mathbf{1}]$, then the line intersects the circle in the line segment $\boldsymbol{p_1 p_2}$ otherwise the line intersects the circle when extended.
Hence the UAV collides with the cylindrical obstacle if and only if the value of $\boldsymbol{t}$ we get is in the interval $[\mathbf{0}, \mathbf{1}]$

2.3.2 Path Finding

In case it is determined in the previous step that a collision may happen, the height of the waypoints is compared with the height of the obstacle. If the Height of both waypoints is greater than the height of the obstacle then no action is taken as there is no collision. However, if the height of either of the waypoints is less than the height of the obstacle then the following cases are considered

Case 1: Both waypoints are outside the boundary of the obstacle

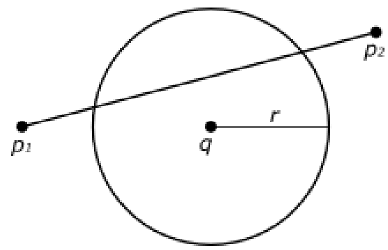


Figure 2.3.2.1

In this case, the height of the obstacle is compared with its radius.

Case 1.1: $height_{obstacle} \gg radius_{obstacle}$

The parameter t calculated in the previous step is then used to calculate the points p_{i1}, p_{i2} at which the path intersects the obstacle.
Now, consider the vector $\vec{a} = p_1 p_{i1}$
Now we want to rotate $\vec{a}$ by an angle θ and choose a θ such that the new transit point clears the obstacle by $6\ feet$
We then rotate the vector using the rotation matrix and get the new transit point $p_t = (x_t, y_t)$ as

$$\begin{bmatrix} x_t \\ y_t \end{bmatrix} = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix} * \begin{bmatrix} x_{i1} \\ y_{i1} \end{bmatrix}$$

We then check if p_t lies outside the geofence boundary or inside any other obstacle [*]
If p_t passes this check then it is added as a transit point.
If it does not, then we rotate by an angle $-\theta$ and repeat the procedure.
If it does not pass the test again then this case is handled as case 1.2
Hence the path now is $p_1 \rightarrow p_t \rightarrow p_2$
The method is then repeated for the path $p_t \rightarrow p_2$ and if it collides, the next transit point is calculated. The method is repeated till p2 is reached and there is no collision.
If it does not pass the check then we rotate by an angle - and repeat the procedure.
If it does not pass the test again then this case is handled as case 1.2

Case 1.2: $height_{obstacle} \leq radius_{obstacle}$

In this case the transit waypoint is added by along the same line in the $xy - plane$ but the height is increased. The number of transit points added depends on the height of the waypoints as shown in the diagram.

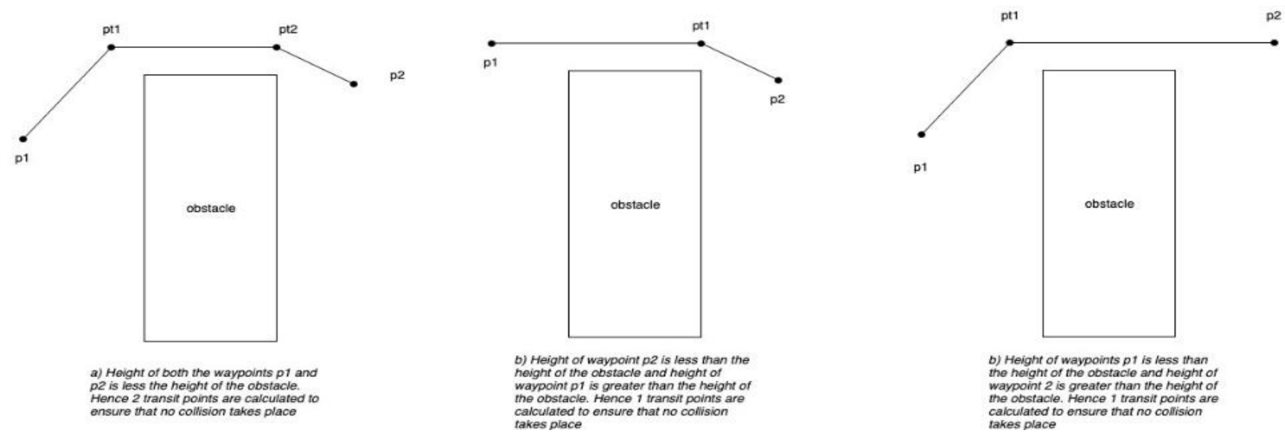


Figure 2.3.2.3

Case 2: One of the waypoints is inside the boundary of the obstacle

In this case, the transit waypoint is added as per the height as in figure 2.3.2.5. A transit waypoint is added so that the UAV does not collide with the obstacle while gaining altitude to travel to the next waypoint.

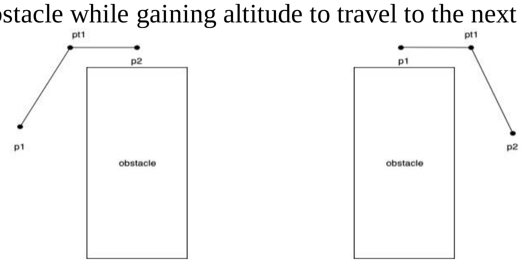


Figure 2.3.2.4

This is done to ensure that collisions of the nature as described in figure 2.3.2.6 are avoided

(*) - This method fails in cases of extreme clutter and will not generate a path in the cases as in figure 2.3.2.7

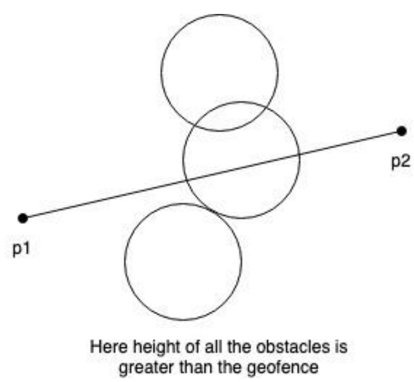


Figure 2.3.2.5

To tackle such cases, the GCS operator also manually checks the path and adds corrections if needed. The GCS operator also manually corrects the path if there are obstacles inside the search grid for ODLC as this algorithm is not applied on the search area and only on waypoints

2.4 Imaging System

The system consists of an onboard ARM based computer system (OCS) running Linux. The OCS connects to the camera via the Camera Serial Interface (CSI). The chosen camera is the Raspberry Pi Camera V2. The Camera is fast and has a very high resolution to size ratio which made it the ideal aerial camera. Other compact higher resolution camera options were not explored due to financial constraints. The autopilot triggers the shutter by sending a digital HIGH signal to the OCS once it enters the search area. The OCS continuously clicks pictures until the Autopilot sends a digital HIGH again, signifying the end of the survey. The pictures are saved to the OCS's memory and are transmitted to the GCS (Ground Control System) once the survey is over. The system receives the location information of the UAV from the autopilot via the UART (Universal Asynchronous Receiver Transmitter) port on the OCS. The location information is crucial for localisation (finding the location of the objects on the ground after they are detected and classified), the location information is written to a JSON file and the file is transmitted to the GCS along with the captured image.

The camera is mounted on a two-axis gimbal controlled by the autopilot, the gimbal is driven by a brushless control system that controls motion in one direction each. The UAV is very stable in the air as it is a multi-rotor, adding a gimbal for stabilisation increases the camera's stability and makes sure the camera is always at a nadir angle (facing the ground).

The OCS uses asynchronous capturing and transmission (ACAT) which makes sure that delay in capturing images does not affect transmission. This is implemented by using parallel processing algorithms. A similar system is used on the ground at the Image Processing Server (IPS) for asynchronous reception and processing (ARAP).

2.5 Object Detection, Classification and Localisation

As the images captured and transmitted by the UAV's IP system are received by the image processing system (IPS) in the GCS. The detection algorithm based on Maximally Stable Extremal Regions (MSER) finds regions of interest (ROI). The ROIs are then exported to the classification algorithm after some contrast stretching, sharpening and noise reduction. The classification algorithm classifies the ROIs with respect to alphanumeric character, shape, colour and orientation. In the last step the location is estimated using the localisation algorithm, local coordinates of the ROI in the image are used to calculate the global coordinates (GPS coordinates) of the object.
Classification and Localisation is carried out parallelly by implementing multithreading in the IPS.

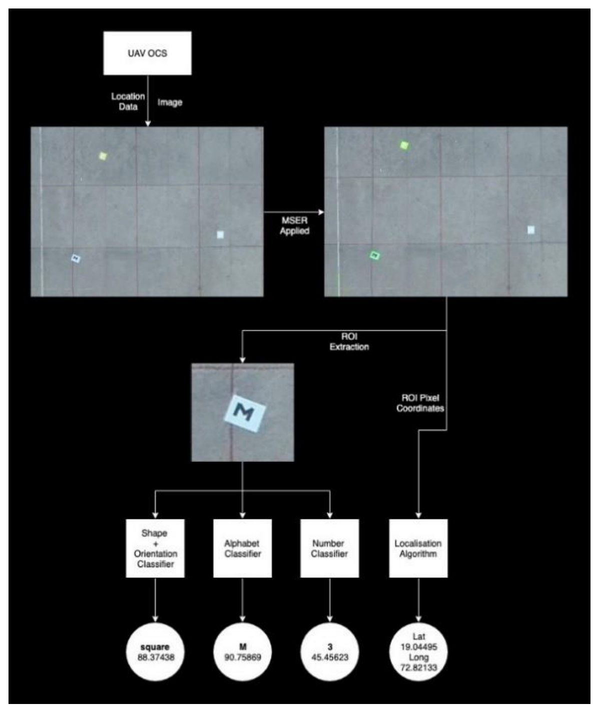


Figure 2.5.1 ODLC Flow

2.5.1 Detection

The MSER algorithm extracts from an image several co-variant regions, called MSERs: an MSER is a stable connected component of some grey-level sets of the image. The original algorithm of is $O(n\log(\log(n)))$ in the number n of pixels. It proceeds by first sorting the pixels by intensity. This would take $O(n)$ time. After sorting, pixels are marked in the image, and the list of growing and merging connected components and their areas is maintained using the union-find algorithm. This would take time. In practice these steps are very fast. During this process, the area of each connected component as a function of intensity is stored producing a data structure. A merge of two components is viewed as termination of existence of the smaller component and an insertion of all pixels of the smaller component into the larger one. In the extremal regions, the 'maximally stable' ones are those corresponding to thresholds where the relative area change as a function of relative change of threshold is at a local minimum, i.e. the MSER are the parts of the image where local binarization is stable over a large range of thresholds.[1]

2.5.2 Classification

The exported ROIs are passed through an image classification system which outputs the confidence of the ROI belonging to a class in probabilities. The image classification system consists of three main classifiers, the first and second ones based on Convolutional Neural Networks (CNNs) to classify alphabets and numbers and the third one to classify shapes based on Google's Inception V3 network. Once the ROIs are classified, their local coordinates (pixel locations) are sent to the localiser to calculate the location of the object in the image.
The images taken from the onboard camera are then sent to a classifier that returns the pixel location of the objects. This location is then sent to a python script that extracts relevant data like the angle of the target relative to the heading of the UAV, the distance of the object from the UAV as well as its latitude and longitude.

2.5.3 Localisation

GSD Calculation

To calculate the distance of the target from our current location, a function that maps each pixel to a distance is required. The function is called as Ground Sampling Distance (GSD) which depends on the camera specifications and the altitude of the UAV. For maximum accuracy the camera must be in a nadir position.
Depending on the accuracy requirements of the objective GSD in metres/pixel can be calculated using two formulas

- $gsd = ((pixelSize * 1000) * height)/focalLength$
- $gsd = \max((\frac{height*sensorWidth}{focalLength*resolutionWidth}), (\frac{height*sensorHeight}{focalLength*resolutionHeight}))$

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.

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

[1] – Source: Wikipedia

2.6 Communications

2.6.1 Ground - UAV Communication

The communication system is divided into two main subsystems accompanied by the manual radio system, the telemetry subsystem and the data subsystem. The telemetry subsystem is responsible for communication between the UAV's autopilot and the ground station, it is a bidirectional link and uses the MAVLink protocol to send and receive data packets. The data subsystem is responsible for the communication between the UAV's IP system and the ground control system, it is a bidirectional link and uses socket programming to send data.

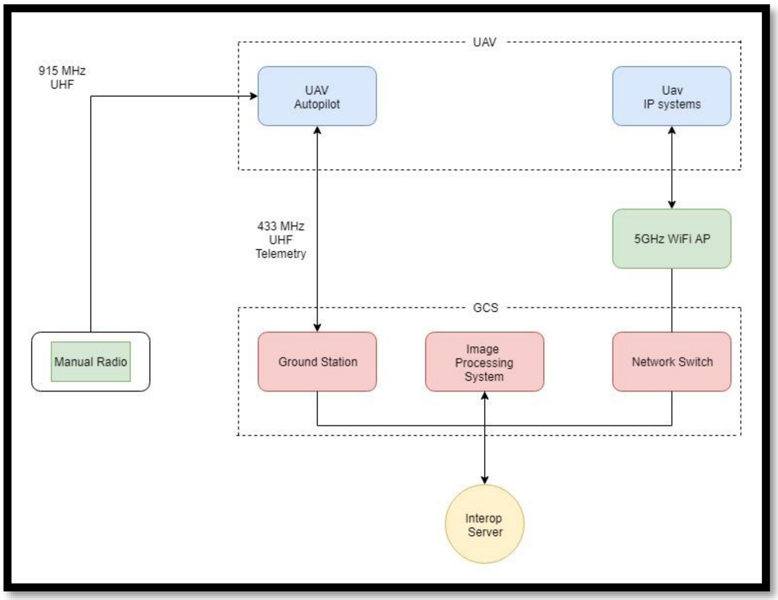


Figure 2.6.1.1 Ground – UAV Communication

The Manual Radio subsystem uses the TBS Crossfire long range UHF self-healing frequency hopping system, this was chosen such that it is the most reliable link between the UAV for the GCS. The TBS Crossfire was extensively tested for more than a year, after more than 50 flights we are confident that manual radio link will not lose connection up to a range of 5 miles line of sight in optimal scenarios.

The Telemetry radio subsystem uses 433Mhz UHF 500mW radios, these radios are compact and reliable. They were tested during 10 flights for distances unto 3 miles line of sight. The telemetry radio subsystem is a bidirectional radio link and connects to the USB port of the ground station. Micro Air Vehicle Link (MAVLink) packets are transmitted to and from between the UAV and the GCS.

The data radio subsystem uses a 5Ghz AC Wi-Fi system (TP-Link Pharos CPE 510). The onboard computer system (OCS) connects to the access point on the ground with USB Wi-Fi Adapter equipped with a high gain antenna. The Wi-Fi access point has a bidirectional 13dBi 2x2 dual-polarized directional MIMO antenna which has a narrow angle of transmission, to tackle this an antenna tracking system is being used where an antenna operator always maintains line of sight with the UAV.

Table 2.6.2.1 Metrics of Communication

Subsystem	Frequency	Protocol	RP Power
Telemetry	433 MHz UHF	MAVLink	500mW
Data	5GHz AC Wi-Fi	TCP	200mW
Manual Radio	915 MHz UHF	SBUS	10mW-2000mW

2.6.2 Onboard Intercomponent Communication

The onboard intercomponent communication is shown in the diagram below. The main system components include the autopilot, onboard computer system (OCS), the camera, radio receivers and the power distribution module. Onboard components connect via different protocols as stated in the diagram (2.6.2.1).

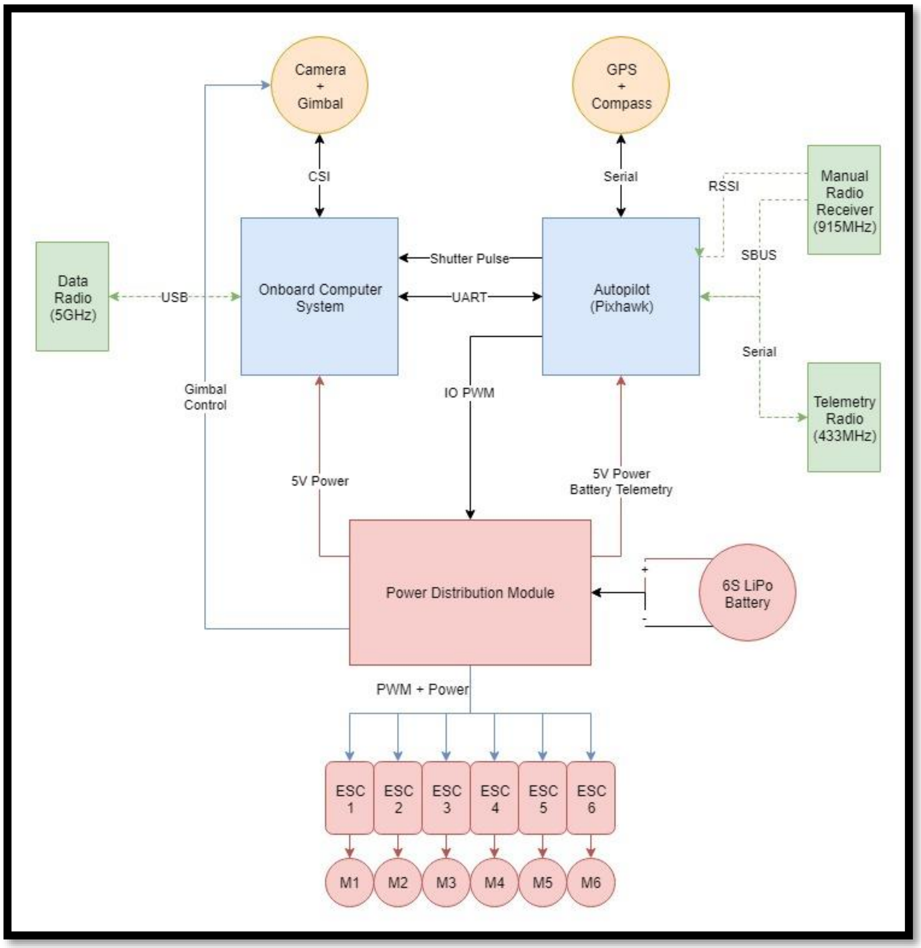


Figure 2.6.2.1 Onboard Intercomponent Communication

2.7 Air Drop

While testing the dropping mechanism using a rudimentary array of servos and a parachute, it was discovered that the added weight of the dropping mechanism and additional batteries needed to perform this task did not justify the performance decrease in other mission tasks given the grading scheme.

2.8 Cyber Security

- Telemetry Radio - The Micro Air Vehicle Communication Protocol (MAVLink Protocol) is a point-to-point communication protocol that allows two entities to exchange information. It is used for bidirectional communications between the UAV and the GCS.
    - The MAVLink packet header consists of a SystemID field which is unique for each vehicle or ground station. The only way to intercept communication is through the program which is bound to the communicating serial port.
    - The protocol also implements two integrity checks. The first check is on the integrity of the packet during transmission using the X.25 checksum. This only ensures that the data has not been altered on the link. The second integrity check is on the data description to ensure that two messages with the same ID are indeed containing the same information. To achieve this, the data definition itself is run through CRC-16-CCITT and the resulting value is used to seed the packet CRC.

- Data Radio – The data radio uses WPA-PSK secured Wi-Fi connection.

- Manual Radio – The manual radio system uses a frequency hopping, self-healing link which prevents intentional jamming or accidental interference.

- GPS - The GPS connected to the Autopilot is secure against GPS spoofing and jamming due to specific hardware used for tracking inconsistencies in incoming GPS data.

- On-Board Computer and GCS - Both are password protected with the keys that are only shared amongst the team members.

2.9 Safety, Risks & Mitigation

Safety is paramount when undertaking such an endeavor as building an autonomous UAV. Strict protocols were implemented at every phase (fabrication, propulsion testing, flight testing, etc.) so that the safety of the development team and the observers in the field was never compromised.

The following tables identifies the risks that the team believes are possible and discusses how these risks are mitigated so that safety is never compromised. The table ranks the chance of occurrence of the risk on a scale of 1-3 with 3 being the maximum probability. Severity is ranked on a scale of 1-5 with 4, 5 being extremely severe and can threaten the safety of human life if not mitigated correctly while 1, 2, 3 being dangerous to the UAV itself and not to any form of human life.

Table 2.9.1 Developmental Risks

Developmental Risks	Chance of occurrence	Severity	Mitigation
Personal injury caused during fabrication	1	5	Every fabrication procedure has a safety check-list that must be passed before initiation. E.g. Appropriate safety gear must be worn
Personal injury caused during testing component	**2**	**5**	Every testing procedure has a safety check-list that must be passed before initiation. E.g. While testing electrical components the operator must be grounded
Unexpected anomalies in flight testing	**1**	**5**	All mission personnel are clear of the UAV before starting the mission

Table 2.9.2 Mission Risks

Mission Risks	Chance of Occurrence	Severity	Mitigation
Failure of GPS	**1**	**3**	In case manual control is available the UAV enters altitude mode or stabilized mode depending on the availability of height estimation In case manual control is not available the UAV enters altitude, mode or executes a flight termination procedure (kills power to all motors) depending on the availability of data.
Motor failure	**1**	**3**	Using a Hex copter gives the team the luxury of being able to lose / two diagonally opposing motors as the autopilot can stabilize the UAV as if it were a quad copter in either case
Loss of telemetry or data link	**2**	**2**	For loss of either links, the UAV waits for a period of 120 seconds. If the link is not established, it Lands at the current position
Battery voltage Management		**3**	The Power Distribution Module monitors and distributes power, it also sends battery telemetry to the autopilot making it possible for the autopilot to take necessary actions in case of low battery or extreme voltage sag.
Autopilot configuration error	**1**	**4**	The auto-stabilization is tested by jerking the UAV (with propellers off) in the roll, pitch and yaw axes respectively and then checking if the appropriate motors speedup/slowdown

3. Conclusion

Team Mavericks UAS has rigorously worked to clearly understand and abide with all the required competition specifications and safety regulations. The team has worked diligently for the past 18 months researching and developing UAV technologies while testing them on the Hex wife for the AUVSI SUAS 2019 competition.