Contents

List of Tables

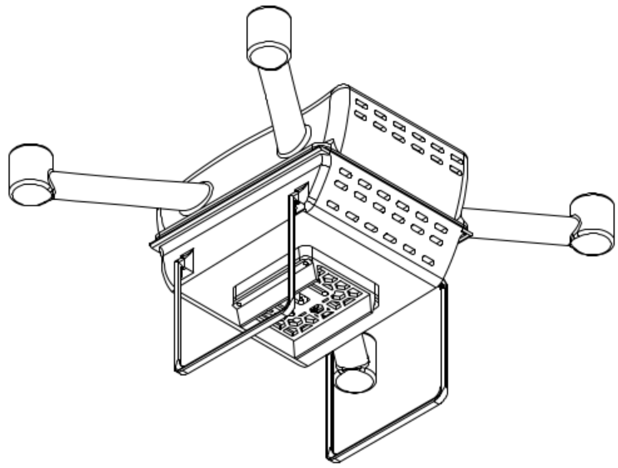
List of Figures

Introduction

1.1 Overview

In the face of natural and man-made disasters, rapid response and situational awareness are critical. Drones have emerged as powerful tools in disaster management, offering real-time aerial insights, access to hard-to-reach areas, and faster deployment compared to traditional methods. Whether locating survivors, assessing damage, or delivering essential supplies, drone technology enhances the efficiency and safety of relief operations. As disasters grow more complex and unpredictable, integrating drones into emergency response systems is no longer a luxury—it's a necessity.

Through Aerothon, Team UDSAV (*Uncrewed Disaster Surveillance Aerial Vehicle*) is not just competing—we are contributing to the evolution of drone-assisted disaster response, pushing the boundaries of what UAVs can achieve in life-saving missions.



1.2 Problem Statement and Mission Requirements

This year's AEROTHON is themed on ***Surveillance and Disaster Management.*** The problem statement is to build an ***Uncrewed Aircraft System (UAS)*** to be able to perform the mission requirements as per the rulebook. The mission requirements at a glance are as follows:

Mission - 1: *Advanced Obstacle Navigation & Fragile Payload Delivery with Precision Placement – Manual Operation*

Mission - 2: *Autonomous Object Classification, Disaster Situation Identification & Payload Drop – Autonomous Operation*

1.3 Scope of Report

The scope of this report is to provide a comprehensive understanding of the design rationale we have used while building this project. We have tried to provide the relevant calculations, figures, and analysis models to justify the materials/design/framework we've chosen to work with for our structural and system architectures.

Apart from that, this report is intended to also serve as an accessible guide catering to neophytes in UAV/UAS systems. We have tried our best to aim at providing clear context and insight that sort of demystifies drone development.

1.4 System Requirements & Design Objectives

1.4.1 Mission Profile

1. **Mission 1:** *Advanced Obstacle Navigation & Fragile Payload Delivery with Precision Placement*
   This is a ***Manual Operation.*** In this mission, the drone must transport a fragile payload through a challenging course filled with static obstacles such as walls, barriers, and narrow passages. The primary objective is to navigate these obstacles with high precision while ensuring the payload remains undamaged.

   Upon reaching the target zone, the drone must land carefully and place the fragile payload on the ground without causing any damage. After the successful placement, the drone must then return to the takeoff point or designated home base, ensuring safe and efficient navigation back through the course. The mission is complete once the payload is placed securely, and the drone successfully returns to the home base.

2. **Mission 2:** *Autonomous Object Classification, Disaster Situation Identification & Payload Drop*
   This is an ***Autonomous Operation.*** In this mission, the drone will autonomously scan, classify, and assess objects within a predefined area using onboard sensors and algorithms. The objects will vary in shape, size, color, and structure, and may be partially obscured, presenting challenges for detection and classification. Once the objects are classified, the drone will identify potential disaster scenarios, such as flooding, fire, or damaged infrastructure, within the same area.

1.4.2 Key Performance Indicators & Constraints

According to the above defined mission profiles, we have a few KPIs (*Key Performance Index*) to keep in mind.

1. Flight Endurance and Range
2. Payload Handling
3. Autonomous Capabilites
4. System Reliability
5. Design and Innovation

The design and development of the UAV is subjected to several constraints as per the guidelines mentioned in the rulebook AEROTHON 2025. These include dimensional constraints, payload restrictions and strict autonomy requirements. The drone must perform all missions bound by these constraints and we have taken great time and care to articulate them down to ensure nothing is amiss.

1. **Dimensional Constraints**
   - Maximum Wingspan: **1.5 metres** - the UAV must fit inside a ***1.5m x 1.5m x 1.5m bounding box*** in assembled condition.

- Maximum Takeoff Weight: $< 2kg$ including battery and payload.

2. **Payload Constraints**
   - Payload: One fragile payload cube of ***12cm x 7cm x 7cm*** weighing ***200g***.
   - Payload must be released within a ***3m x 3m*** target zone.

3. **Flight Environment Constraints**
   - Missions are conducted in ***open outdoor airspace***.
   - Expect wind speeds upto ***5m/s***

4. **Autonomy and Mission Constraints**
   - **Mission 1:** Manual flight only (no GPS or autopilot usage).
   - **Mission 2:** Fully autonomous flight (no pilot intervention or RC use).
   - All autonomous missions must avoid obstacles and make decisions based on ***onboard computation.***

5. **Power and Communication Constraints**
   - Must operate on battery only
   - No cellular or internet-based comms allowed
   - Only 2.4 GHz or 5.8 GHz RF modules permitted

6. **Safety and Compliance**
   - Must have a failsafe mode (e.g., return-to-home or emergency land)
   - Must pass technical inspection before flying
   - Compliance with DGCA drone guidelines (if relevant in test zones)

7. **Operational Constraints**
   - The team must complete the flight within a ***15-minute slot.***
   - Payload must be dropped in an area of ***3m x 3m.***

Conceptual Design Approach

2.1 Design Methodology

In design methodology, we followed structured top down system engineering approach. Our main mission is manual payload delivery and autonomous disaster surveillance. Once the mission is defined, we proceeded through the following design steps:

- Requirement Analysis: Key performance indicators (KPIs) such as payload stability, endurance and autonomy levels were associated to component-level specifications.
- Conceptual Design: Various design choices such as a multirotor configuration, frame, and AI-enabled onboard computation were evaluated.
- Component Selection: Each subsystem—propulsion, aerodynamics, structure, sensors—was chosen based on performance, power, weight, and cost.
- Iterative Prototyping: Using simulation and CAD modeling, we iteratively refine CG balance, and propulsion performance.
- Validation & Optimization: The design is validated using CFD/FEM tools for aerodynamic , deformation , stress and structural behavior.

2.2 Product Benchmark & Trade-off Analysis

For optimizing performance and ensuring mission reliability, a benchmarking exercise was conducted.

Benchmarked Categories:

- Frame type: H-frame, X-frame, and custom modular frames
- Motor: Low vs high KV ratings
- Material: Carbon fiber vs aluminum vs ABS composites
- Computational Units: Raspberry Pi vs Jetson Nano/Orin for onboard processing
- Flight Controllers: Pixhawk vs DJI N3 vs Navio2

Trade-off Analysis:

- Each option was assessed across several key criteria:
- Weight vs Strength (e.g., carbon fiber offers great stiffness at low weight)
- Cost vs Performance (e.g., Jetson Orin offers better AI capabilities than Pi but is costlier)
- Thrust vs Efficiency (e.g., higher KV motors offer more speed and power)

Detailed Design Breakdown

3.1 Preliminary Weight Estimation

Table 3.1: Detailed Weight Breakdown

Parameter	**Weight *(gms)***
NVIDIA Jetson Orin Nano	176
Pixhawk 2.4.8	39
Camera (*x2*)	20
SpeedyBee BL32 50A 4-in-1 ESC	90
DYS D2836-7 1120KV BLDC (*x4*)	280
Battery (Orange 5200mAh 11.1V 3S)	360
GPS – Neo M8N	23
Transmitter (RadioMaster)	370
Receiver	0.55 - 1
VTX	5.6
HDMI to AV Converter	40
Payload	200
Servo Motor	10
Propellor (9”)	40
Estimated Frame Weight	700
Total:	**2354.6**

Note: The transmitter is not a part of the UAS itself, so effective drone weight is **1984.6gms**.

3.2 Thrust Requirement & Propulsion System Selection

3.2.1 Thrust Requirement

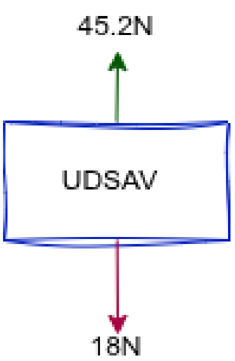


Figure 3.1: Thrust-to-Weight Diagram

To ensure stable and controlled flight, a multirotor drone must generate sufficient thrust to overcome its total weight. The drone in this design has a total takeoff weight of **1805 grams** (1.805 kg). For stable hovering, the combined thrust of all motors should ideally be at least equal to the total weight. However, to allow for effective maneuverability, rapid ascent, and compensation for wind or payload imbalance, a common design guideline is to target a thrust-to-weight ratio of at least **2:1**. This implies a minimum total thrust of approximately $2 \times 1.805 = 3.61$ kg. The selected propulsion system comprises four DYS D2836-7 1120KV brushless DC motors.

According to manufacturer test data, when paired with a suitable 10×4.7 propeller and a 3S (11.1V) LiPo battery, each motor can produce up to approximately **1130 grams** of thrust. Therefore, the total available thrust from all four motors is approximately **4.52 kg**, yielding a thrust-to-weight ratio of $4.52/1.805 \approx 2.5$. This satisfies the performance margin and confirms that the chosen motor-propeller combination is adequate for the drone's operational requirements.

3.2.2 Motor, ESC & Propellor

Motor: DYS D2836-7 1120KV Brushless Motor

The DYS D2836-7 1120KV Brushless Motor is our go-to motor for this project because of the following leverages it offers:

Figure 3.2: DYS D2836-7 1120kV BLDC

- ***KV Rating: 1120KV*** - KV generally means RPM per volt. In layman terms, in one volt, how many rotations does it make per minute = KV. In this case, 1120KV is mid-range, which means good thrust at moderate RPMs, and it works decently with larger propellors (*9" - 11"*) which improves lift and efficiency, *especially* at low speeds. This is perfect for surveillance drones that require loitering and stability. A lower KV would force us to use bulky propellors, and a higher KV would drain the battery faster. 1120KV is a sweet spot between the two.

- ***Power & Efficiency:*** - With a 3S or 4S LiPo, this motor produces 800g to 1100g of thrust, depending on the propeller used. It can pull 20–25A max, so it's efficient for mid-weight UAVs (*in our case, it is around **1.5 2kg AUW** (All Up Weight.)*), so it's ideal for our choice.

ESC: SpeedyBee BL32 50A 4-in-1 ESC

The SpeedyBee BL32 50A 4-in-1 ESC is a good choice for surveillance drones, and our use case for the following reasons:

Figure 3.3: SpeedyBee BL32 50A 4-in-1 ESC

- ***High Current Rating (50A per motor):*** Supports high-thrust motors and larger propellers. Useful for longer flight times, heavy payloads (cameras, sensors, gimbals), and stable cruising. Provides headroom — motors drawing 20–30A will run cooler and more reliably under a 50A ESC.

- ***BLHeli_32 Firmware:*** Smoother motor response, more efficient power delivery, and better low-end throttle control, which helps in steady hovering and slow maneuvering — perfect for surveillance.

- ***4-in-1 Design:*** Combines 4 ESCs into one board, and reduces weight and wiring complexity. Makes the stack cleaner, ideal for modular or compact drone frames. Fewer potential failure points (vs. 4 individual ESCs).

- ***Telemetry & Monitoring:*** Supports ESC telemetry (RPM, current, temperature) via BLHeli_32. This is important for diagnostics, health monitoring, and autonomous missions — ensuring no motor overheats or fails mid-flight.

- ***Built for 3–6S LiPo:*** Offers flexibility across drone designs. For surveillance, a 4S or 6S setup is common due to higher efficiency and flight duration. This ESC handles both without issue.
- ***Built-in TVS Protection:*** Has **Transient Voltage Suppression diodes** that protect against voltage spikes — vital for drone safety, especially in critical missions.

Propellor: HQProp Thin Electric Prop 9×5 (2CCW) Propeller

This propeller is perfect for our use case for the following reasons:

Figure 3.4: HQProp Thin Electric Prop 9×5 (2CCW) Propeller

- ***High Efficiency for Long Endurance Flights:*** It is Thin electric profile = low drag $\rightarrow$ reduces current draw. Designed for cruise efficiency over brute force thrust, it serves perfect for surveillance missions where hovering and slow, steady forward flight dominate.
- ***Optimized for Mid-Sized Motors (like D2836-7):*** The 9-inch diameter is a good disc area for smooth lift, and 5-inch pitch gives moderate speed per RPM (good forward motion without excess current). These features allows it to pair well with 1000–1200KV motors on 3S LiPo $\rightarrow$ ideal thrust-to-efficiency balance.
- ***Smooth Throttle Response:*** Thin blades create less turbulence and vibration. This is crucial for gimbal-mounted cameras or FPV systems, reducing jello and image blur.
- ***Expected Performance on 3S + DYS D2836-7:***
  1. Static Thrust 850–1000g
  2. Current @ full throttle 15–18A
  3. Thrust Efficiency $\approx$ 60–65 g/W

3.2.3 Propulsion Powertrain Efficiency

The total powertrain involves all the individual components that draw power from the battery, this includes things like the flight controller and flight computer. Here we are interested only in the propulsion powertrain. The propulsion powertrain typically includes:

Battery $\rightarrow$ ESC $\rightarrow$ Motors

The battery and ESC are suppliers, they supply on demand, and since all 4 motors won't derive the same amount of current (and hence power) at the same point of time– the real-life parameters will vary in time. Here we assume that all motors demand the same power at all times.

To quantify the overall efficiency of the UAV's propulsion system, we analyze losses in each powertrain component. That is mathematically given by,

$$\eta_{total} = \eta_{battery} \times \eta_{esc} \times \eta_{motor}$$

In order to calculate each of these components, we would need to calculate the power input and output at each stage. Since we don't currently have access to each component at the moment, we're going to use the parameters provided by the manufacturers for this calculation.

Battery Efficiency Derivation:

Given are the following from datasheets:

- Voltage (V): 11.1V
- Max Discharge Current: 208.0A (40C)
- Max Power Output ($P_{battery}$): V x I = 11.1 x 208 = **2308.8 W**

This output power from the battery shall be used as input to the ESC. Now, to calculate the efficiency of battery, we can define it as,

$$\eta_{battery} = \frac{P_{out}}{P_{stored}}$$

But in-flight, it's more feasible to model this using internal resistance. So,

$$\text{Power lost in battery} = I^2 \, R_{int}$$

$$\eta_{battery} = \frac{VI - I^2 R_{int}}{VI} = 1 - \frac{IR_{int}}{V}$$

Typically, for our battery, the internal resistance is $\boldsymbol{R_{int} = 0.015\Omega}$

$$\text{Power loss} = (208)^2 \times 0.015 = 648.96\,\text{W}$$

$$P_{out} = 11.1 \times 208 = 2308.8\,\text{W}$$

$$P_{stored} = 2308.8 + 648.96 = 2957.76\,\text{W}$$

$$\boldsymbol{\eta_{battery}} = \frac{2308.8}{2957.76} \approx 78.07\,\% = \mathbf{0.78}$$

ESC Efficiency Calculations:

The following data from the datasheets:

- Max Continuous Current: 50A (per channel)
- Voltage Range: 3–6S LiPo (up to 25.2V)
- Estimated Losses: 5–10% (heat dissipation)

Since this is a 4-in-1 ESC, it shares a single power input from the battery and distributes it internally to all 4 ESC channels. The output power is given by,

$$P_{out} = P_{in} - P_{loss}$$

$$\Rightarrow P_{out} = 2308.8 - \frac{5}{100} \times 2308.8$$

$$\therefore P_{out} \approx 2193.36\,W$$

$$\Rightarrow \boldsymbol{P_{in} = 2308.8W} \qquad \boldsymbol{P_{out} = 2193.36W}$$

$$\boldsymbol{\eta_{esc}} = \frac{P_{out}}{P_{in}} = \frac{2193.36}{2308.8} \approx \mathbf{0.95}$$

The total power output shared by all 4-channels of the ESC is **2193.36W**. A single channel is capable of supplying,

$$\mathrm{P_{in\_motor}} = \mathrm{P_{out\_ESC}} = \frac{2193.36}{4} = \mathbf{548.34}\,\text{W}$$

Motor Efficiency Derivation:

The following data is given in the official datasheet:

- KV Rating: 1120 RPM/V
- Max Power: 336 W
- Max Current: 23.2 A
- Voltage Range: 2–4S LiPo (7.4–14.8 V)
- Internal Resistance: 0.070 Ω
- Propeller: 9×5

The algorithm to derive the motor losses goes as follows: the efficiency is given as,

$$\eta = \frac{\boldsymbol{P_{out}}}{\boldsymbol{P_{in}}}$$

$$\text{Electrical input power:} \quad \boldsymbol{P_{in} = V \times I}$$

$$\text{Mechanical output power:} \quad \boldsymbol{P_{out} = T \times \omega}$$

where ***V*** is voltage at which thrust is rated, ***I*** is current drawn at that voltage; ***T*** is torque generated by the motor (in newton-meters), and $\boldsymbol{\omega}$ is angular velocity given by,

$$\omega = \frac{2\pi \times \text{RPM}}{60}$$

The RPM without any load will be $1120 \times 11.1V = 12432.0\,rpm$. But when we attach the propellers, some load will be acting agaisnt them, causing the RPM to drop by an amount. Let us assume the new RPM under load is $\boldsymbol{RPM_{load} = 12000\,rpm}$, then the angular velocity is

$$\omega = \frac{2\pi \times 12000}{60} \approx 1256.63\,rad/s$$

The theoretical torque can be calculated from the formula

$$\boldsymbol{\tau = K_t \,.\, I}$$

$$\text{where} \quad K_t = \frac{60}{2\pi\, K_v} \quad and \quad I \rightarrow \text{current in amps} \;=\; 23.2\text{A}$$

$$\text{now,} \qquad K_t = \frac{60}{2\pi \times 1120} \quad \approx \quad 0.00852$$

$$\therefore \tau = 0.00852 \times 23.2 \;=\; 0.19780\,\text{Nm}$$

$$\Rightarrow \boldsymbol{P_{out}} \quad \boldsymbol{= \tau \times \omega =} \quad \boldsymbol{248.57}\,\text{W}$$

Therefore, the motor efficiency is

$$\boldsymbol{\eta_{motor}} = \frac{P_{out}}{P_{in}} = \frac{248.57}{336} = 0.73979 \quad \approx \quad \mathbf{0.74}$$

which is a pretty reasonable efficiency in real world BLDC motors. Finally, the total propulsion powertrain efficiency is given as,

$$\eta_{total} = \eta_{battery} \times \eta_{esc} \times \eta_{motor}$$

$$\Rightarrow \boldsymbol{\eta_{total}} = 0.78 \times 0.95 \times 0.74 \quad = \quad \mathbf{0.55}$$

The final propulsion powertrain efficiency sums upto around 55%, which is a reasonable value considering that some of the parameters we're assumed. Real world values will obviously vary from this.

3.3 Aircraft Sizing

These are 2D schematics to ensure more precise depiction of the diagrams. 3D CAD schematics are provided in the **Appendix** section.

Propeller Clearance & Rotor Arm

Our propellers have a diameter of 9in. 1in = 25.4mm, so that means the diameter is 228.6mm. Each prop is placed 250mm away from the center (side-by-side adjacent distance), so the clearance is 250mm - 228.6mm = 21.4mm. **Rotor Arm** is approximately 120mm according to our schematics.

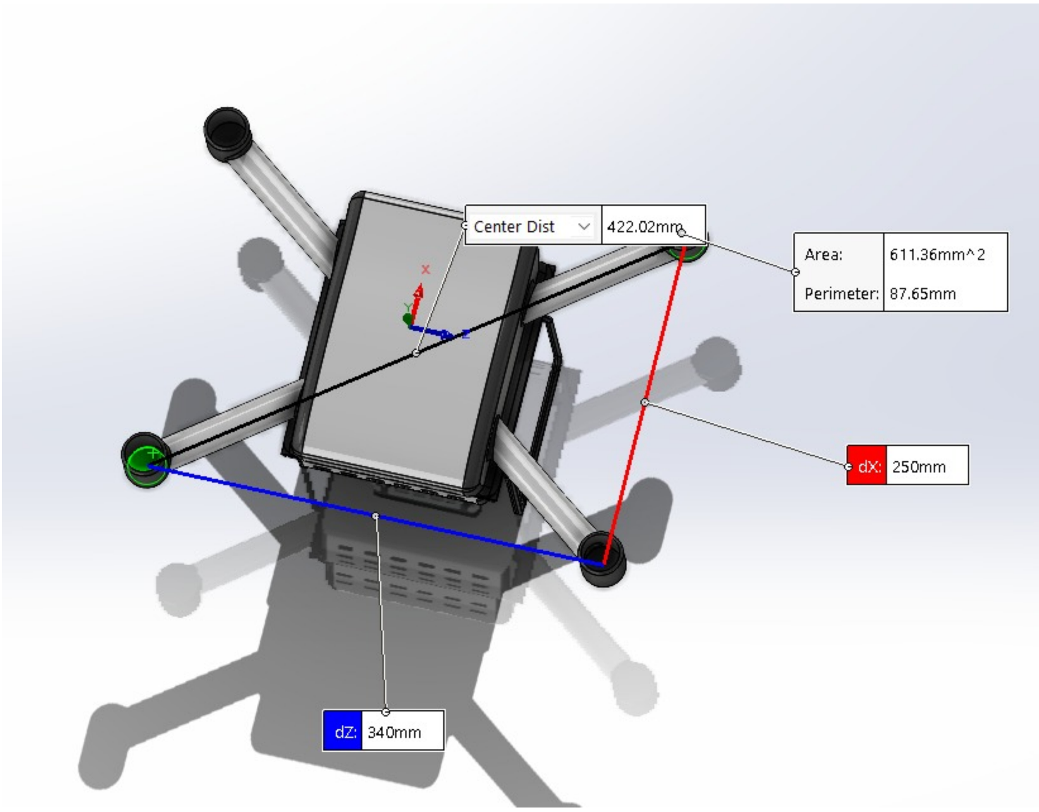


Figure 3.5: Propeller Clearance is around 21.4mm

Hub

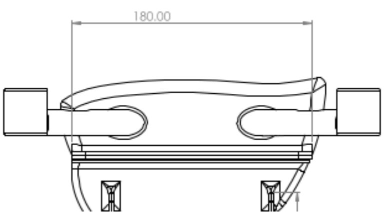


Figure 3.6: Hub Side

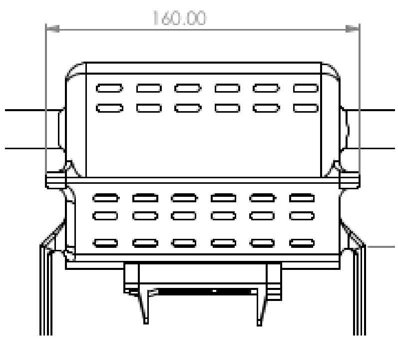


Figure 3.7: Hub Rear

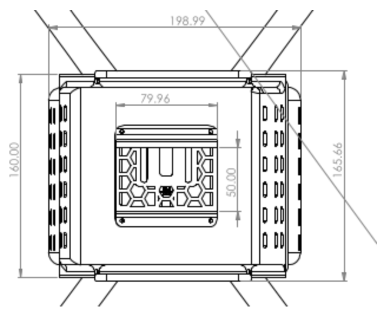


Figure 3.8: Hub Top

Landing Gear

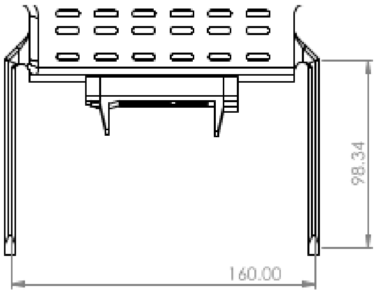


Figure 3.9: Rear View

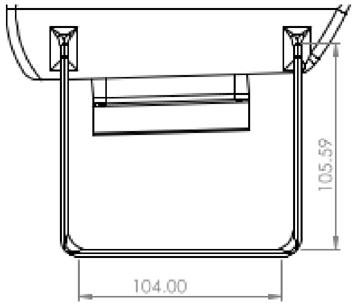


Figure 3.11: Side View

Wheelbase

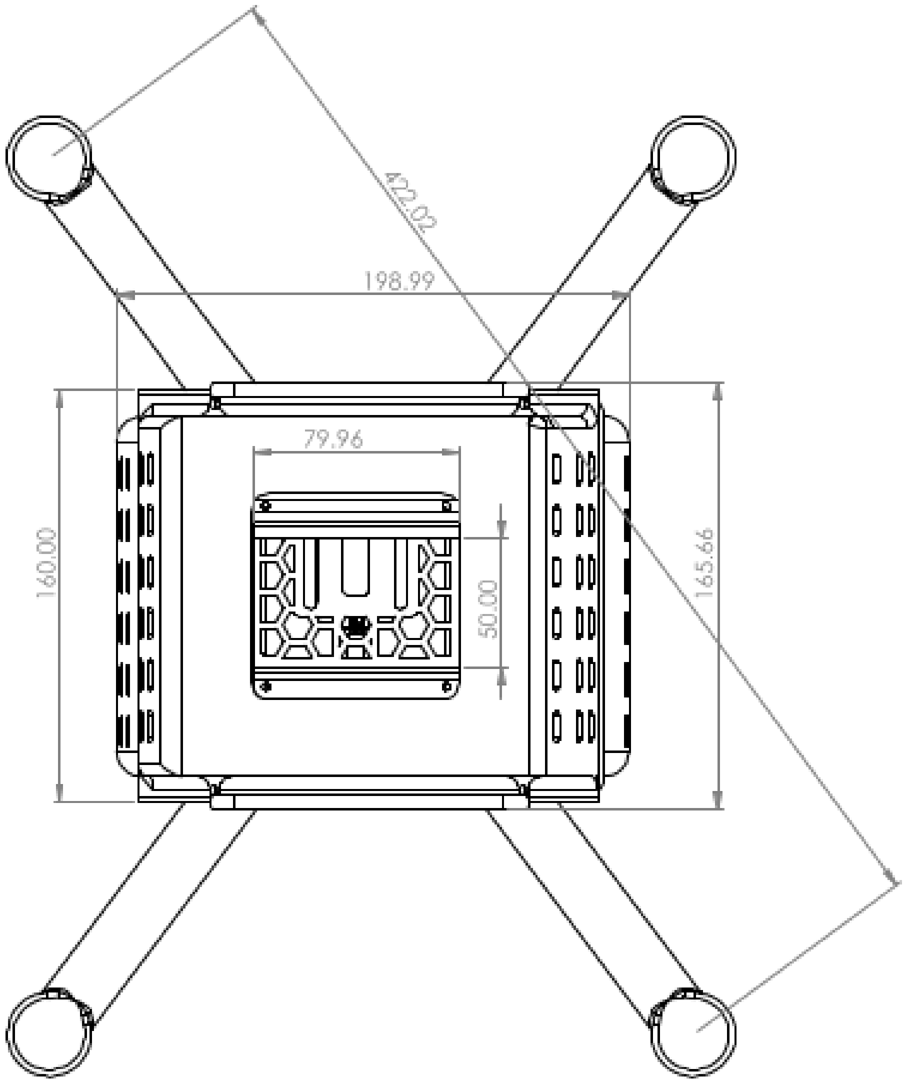


Figure 3.12: Wheelbase = 422.02mm

3.4 Aircraft Performance

3.4.1 Battery Selection and Endurance

Battery: Orange Pro-Range 11.1V 5200mAh (3S)

The Orange Pro-Range 11.1V 5200mAh battery is the best for our use case for the following reasons:

The 3S variant provides 11.1V, and has a **discharge-rate** of 40C. According to the official rated specifications, the maximum continuous discharge current is **208.0A** (*40C*). It also has a max. burst discharge of **416.0A** (*80C*). Let us assume that each motor draws 24A current at full-throttle, total current draw would be $24 \times 4 = 96A$ then

Figure 3.13: Orange 11.1V 5200mAh 3S

$$\text{Theoretical Flight Time (hrs)} = \frac{\text{Capacity (Ah)}}{\text{Current Draw (A)}} = \frac{5.2}{96} \approx 0.0542hrs = 3.25mins$$

But in real world applications, we dont use 100% of the battery, we use about 60%, so that would make the flight time around 5.2mins.

3.4.2 Total Power Budget Summary

This sub-section summarizes the total electrical power budget of the UAS, highlighting how power from the battery is allocated to propulsion and non-propulsion (avionics and payload) subsystems.

Table 3.2: Power Distribution Summary

Subsystem	**Included Components**	**Power Demand (W)**	**% of Total Power**
Propulsion	4 x Motors	1344 W	58.2%
Avionics	Flight Controller, GPS Module, Sensors (IMU, barometer)	6.9486 W	0.3009 %
Communication	RC Receiver, Wifi Modules	20.41 W	0.884 %
Payload	2 x Cameras	3.879 W	0.168 %
Onboard Computer	NVIDIA Jetson Orin Nano 8GB Module	6.997 W	0.303 %

Propulsion Demand

Based on the datasheet specification and text results, the propulsion subsystems (***4xMotors***) demands **1344W** during peak operation. This forms the largest share of the total Power requirement ($\sim$ 58.2 %).

Avionics Demand

The avionics system forms the central nervous system of any unmanned aerial vehicle (UAV), including those designed for **Aerothon-class missions**. Its architecture and power requirements play a decisive role in shaping the overall energy distribution and electrical resilience of the aircraft.

The avionics suite typically comprises a flight controller (e.g., Pixhawk), GPS module, telemetry and communication transceivers, RC receiver, and sensor systems such as IMUs and barometers.

In advanced UAV configurations—such as ours—it is further augmented by an onboard companion computer, namely the NVIDIA Jetson Orin Nano 8GB, which performs real-time perception and decision-making tasks.

Based on our subsystem-level analysis, the avionics block—comprising flight control, communication, and onboard computing—demands a cumulative peak power of approximately **34.36 W**. This includes:

- **6.948 W** for the flight controller and embedded sensor suite
- **20.41 W** for communication subsystems including RC receivers and WiFi modules,
- **6.997 W** for the Jetson Orin Nano, which manages AI workloads and perception.

While this represents only **1.49 %** of the total system power draw, the avionics demand is non-negotiable and continuous, requiring high uptime and precision. Power is supplied via a regulated DC bus derived from the main propulsion battery, with dedicated buck converters providing clean and stable 5V and 3.3V rails to sensitive electronics.

Margins, Safety Factors

Power Budget Margin:

1. **Purpose:** To ensure that the chosen power source (battery) and propulsion system can consistently provide sufficient power, even under demanding conditions or if components don't perform exactly to spec.
2. **Typical Range:** 15% - 30% of the calculated total power demand.
3. **Application:** After calculating the power required for propulsion (hover, cruise, max thrust) and avionics, an additional percentage is added.

   Total Required Power = (Propulsion Power + Avionics Power) * (1 + Power Margin)

4. **Considerations:**
   - Battery degradation over time.
   - Variation in motor/propeller efficiency.
   - Ambient temperature effects on battery performance.
   - Increased power draw due to wind or aggressive maneuvers.

Weight/Payload Margin:

1. **Purpose:** To allow for small increases in component weight during design evolution, manufacturing variations, or for potential future upgrades/additional payloads.
2. **Typical Range:** 10% - 20% of the estimated total weight (Empty Weight + Max Payload).
3. **Application:** When defining the Maximum Take-Off Weight (MTOW), a buffer is included. This also impacts thrust-to-weight ratio calculations.

   Max All-Up Weight (MAUW) = (Estimated Empty Weight + Max Payload) * (1 + Weight Margin)

4. **Considerations:**
    - Small design changes or additions.
    - Manufacturing tolerances.
    - Unforeseen weight of cabling, fasteners, etc.

Battery-Capacity/Flight-Time Margin:

1. **Purpose:** To ensure sufficient energy is available for the planned mission duration, plus a reserve for unexpected events (e.g., strong headwind, holding pattern, emergency landing).
2. **Typical Range:** 15% - 25% of the total mission energy requirement.
3. **Application:** After calculating the energy needed for the mission profile, additional capacity is added. Also, a ”return to home” or ”emergency landing” battery percentage is typically set (e.g., 20-30% remaining).

$$(\text{Battery-Capacity})_{\text{Total}} = \frac{\text{Mission Energy Requirement}}{\text{Battery Discharge Efficiency}} \times (1 * \text{Capacity Margin})$$

4. **Considerations:**
    - Battery performance degradation over cycles and temperature.
    - Unexpected mission deviations.
    - Wind conditions requiring higher power.
    - Maintaining a safe reserve for landing.

Structural/Safety Margin:

1. **Purpose:** To ensure that the drone’s airframe and structural components can withstand expected and unexpected loads without failure.
2. **Ultimate Factor of Safety (FoS):** 1.5 - 2.0 (or higher for critical components). This is the ratio of ultimate load capacity to the maximum expected operating load.
3. **Yield Factor of Safety:** 1.1 - 1.25. This is the ratio of yield strength to the maximum expected operating load, ensuring no permanent deformation.
4. **Application:** Applied to material strength calculations for frame arms, motor mounts, landing gear, etc.

Required Strength = Maximum Expected Load * Factor of Safety

5. **Considerations:**
    - Dynamic loads during flight (acceleration, turns).
    - Impact loads during hard landings or minor crashes.
    - Vibration fatigue.
    - Material imperfections and manufacturing variability.

3.5 Material Selection

ABS meets our requirements with its strength, lightweight, durability, and thermal stability. Its cost-effectiveness and printability make it ideal for reliable, efficient drone components. Thus, we chose ABS for our designs.

ABS Material Properties and Drone Application Relevance		
Property	**Material Description**	**Use Justification**
Mechanical Properties	Impact resistance: High Tensile strength: 30–45 MPa Flexibility: Moderate	Withstands crashes and hard landings. Can endure flight forces and minor impacts. Absorbs shocks effectively.
Density	Low density: $\sim$1.04 g/cm^3	Lightweight material improves flight efficiency and increases battery life.
Thermal Properties	Glass transition temperature: $\sim$105°C Stable under moderate heat	Resists deformation from motor heat or sunlight. Performs consistently in various environmental conditions.
Chemical Resistance	Resistant to oils, acids, and alkalis	Suitable for drones exposed to industrial or outdoor environments with potential chemical exposure.
Printability	Good layer adhesion Requires heated bed and controlled environment to avoid warping Easy post-processing (e.g., sanding, acetone smoothing)	Ensures durable, smooth, and aerodynamic drone parts. Customizable designs for prototyping and functional components.
Cost-Effectiveness	Relatively low cost compared to alternatives	Ideal for functional prototypes and small-scale production without compromising on quality.
Comparison with Alternatives	**PLA:** Brittle, less heat-resistant **Nylon:** Durable but expensive, harder to print **Carbon Fiber Composites:** Strong, lightweight, but costly and harder to process	ABS balances cost, durability, and ease of use compared to alternatives.

Table 3.3: Properties and Evaluation of ABS Material for Drone Applications

3.6 Avionics Subsystems Selection

3.6.1 Detailed Component Breakdown

Flight Controller (Pixhawk 2.4.8)

Pixhawk is widely egarded as one of the best flight controllers for drone and autonomous aircraft projects — especially in academic and research-grade prototypes — for several compelling reasons:

Figure 3.14: Pixhawk 2.4.8

- ***Open-Source and Flexible:*** It is built on open hardware and supported by powerful open-source firmware like PX4 or ArduPilot. This enables deep customization, ideal for research and control system testing. And for this reason also, we have **ample documentation backed by a strong collaborative community, forums, and tutorials**.

- ***Rich I/O capabilities:*** Multiple UART, I2C, CAN, and PWM ports for connecting sensors (GPS, IMU, barometer, etc.) and actuators (ESCs, servos). Ideal for integration with multiple onboard systems including companion computers (e.g., Jetson Nano).

- ***Compatible with autonomous and GPS-guided missions:*** Supports autonomous navigation, geofencing, waypoints, and RTL (Return to Launch).

- ***Built-in failsafes and safety features:*** Battery failsafes, signal loss handling, and software watchdogs protect the aircraft during unexpected conditions.

- ***Excellent simulation support:*** Compatible with **HITL (Hardware-in-the-loop)** and **SITL (Software-in-the-loop)** for control testing and simulation.

Flight Computer (NVIDIA Jetson Orin Nano 8GB)

The NVIDIA Jetson Orin Nano 8GB is a powerful, compact AI computing module designed for edge AI applications that demand both high performance and energy efficiency. In the context of our Aerothon UAV project, the Orin Nano plays a pivotal role in enabling advanced onboard computation, particularly for tasks such as **real-time image processing, autonomous navigation, and object detection.**

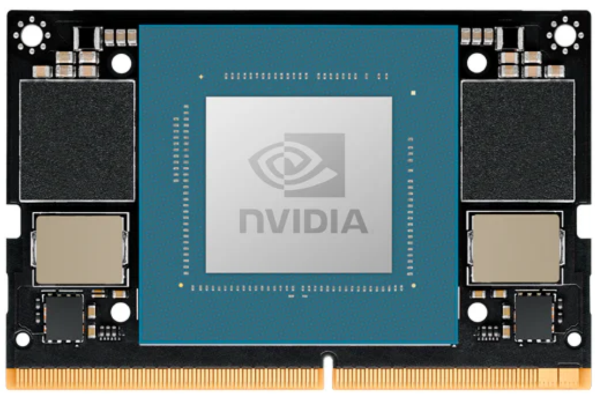


Figure 3.15: NVIDIA Jetson Orin Nano 8GB Module

We selected this module not only for its impressive up to **40 TOPS** of AI performance but also for its **low power footprint**, which makes it ideal for flight-based applications where every gram and watt matter. The 8GB RAM ensures sufficient memory for running heavy models, such as convolutional neural networks for visual recognition or SLAM algorithms for path planning.

The Jetson Orin Nano interfaces seamlessly with the Pixhawk flight controller via UART or serial USB connections, enabling a tight coupling between autonomous decision-making and low-level control. For example, live video feed from an **ESP32-CAM** or other camera modules is processed onboard the Jetson, where the output — such as target coordinates or navigation commands — is relayed to the Pixhawk for actuation.

This configuration allows the aircraft to function **autonomously even without constant ground station communication**, which is critical in GPS-denied or communication-constrained environments. By offloading high-level intelligence to the Jetson module, we achieve a modular and scalable architecture that separates perception and decision-making from flight stabilization, thereby improving system robustness and flexibility.

In essence, the NVIDIA Jetson Orin Nano 8GB empowers our drone with a true edge-AI brain — transforming it from a remotely controlled vehicle into a **fully autonomous aerial system** capable of intelligent flight and mission execution.

Cameras

We're using two Raspberry Pi Cam V2, one in the front for FPV purpose.. and another in the belly of the drone for obejct detection and payload drop purpose.
We have chosen this camera for the following reasons:

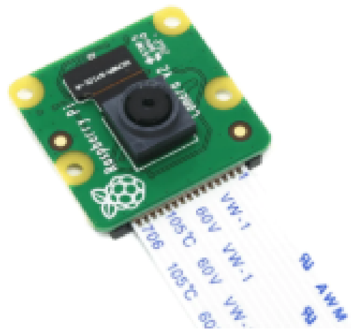


Figure 3.16: Raspberry Pi Cam V2

- ***8 Megapixel Sony IMX219 Sensor:*** Offers high-resolution image capture (3280 x 2464), which ensures that our object detection models get detailed input — especially helpful for detecting small or distant objects.
- ***1080p30 / 720p60 / 640x480p90 Video Modes:*** High frame rates allow for real-time object detection, essential in robotics, drones, and surveillance applications.
- ***Low Latency Capture:*** Minimal delay in capturing frames means your detection pipeline stays fast and reactive, especially when combined with hardware acceleration (like OpenCV + TensorRT on Jetson).
- ***Adjustable Focus (via lens mods or add-ons):*** Although the lens is fixed-focus out of the box, third-party adjustable lenses can be easily added for more precise depth-aware detection.
- ***CSI Interface for Direct Connection:*** Connects via the CSI-2 interface (Camera Serial Interface), which provides higher bandwidth and lower CPU usage compared to USB webcams — ideal for high-performance detection tasks.
- ***Excellent Linux + OpenCV + TensorFlow Compatibility:*** Fully supported in Raspberry Pi OS, Jetson Nano, and other edge-AI platforms, with drivers readily available for OpenCV and GStreamer pipelines.

Transmitter: RadioMaster Pocket Radio Controller (M2) ELRS

Reasons why we chose this transmitter:

- ***Multiprotocol Support (CC2500 Variant):*** It supports many popular protocols: FrSky D8/D16, Futaba SFHSS, Radiolink, etc.
- ***ELRS 2.4GHz: Ultra-Responsive & Long-Range:*** We're using ELRS protocol and this transmitter supports that. ELRS allows low-latency and long-range commuincation which is ideal for our use case.
- ***OpenTX/EdgeTX Firmware:*** Runs EdgeTX, a customizable, open-source firmware for transmitters. Lets us configure **custom mizes, telemetry screens, voice alerts, programmable switches.**
- ***Telemetry Support:*** When paired with the RP1 V2 or telemetry-capable CC2500 receivers, it supports real-time telemetry data: *battery voltage, GPS coordinates, RSSI, link quality etc.*

Figure 3.17: Radiomaster Pocket Radio CC2500/ELRS

Receiver: RP1 V2 ExpressLRS 2.4ghz Nano Receiver

Both the RP1 V2 receiver and our Radiomaster Pocket TX use ExpressLRS 2.4GHz, a modern, open-source RC link. That means no special configuration or cross-protocol hacks needed — they speak the same language out of the box. It also has Ultra-low latency (5–20ms) — ideal for fast, precise drone control.

Figure 3.18: RP1 V2 ExpressLRS 2.4ghz Nano Receiver

The RP1 V2 is nano-sized and weighs less than 2g, so it's perfect for your drone's weight budget. Great for aerodynamic efficiency and maintaining flight time. ELRS dynamically adjusts the packet rate and output power for optimal performance.

Video Transmitter: SpeedyBee TX800 5.8GHz VTX

The VTX (Video Transmitter) sends real-time video from the drone's onboard camera to the pilot's FPV goggles or ground station, enabling precise remote navigation and control.

Figure 3.19: SpeedyBee TX800 5.8GHz VTX

We're using the **SpeedyBee TX800 5.8GHz VTX** because it offers high power output (up to 800mW) for long-range, interference-resistant video transmission, multiple channel support for team use, and smartaudio control for easy configuration via Ardupilot (via MAVLink passthrough). Its **800mW output** ensures strong, stable FPV feed—critical for long-range missions.

NEO-M8N GPS Module

It supports multi-GNSS constellations (GPS, GLONASS, Galileo, BeiDou), which improves satellite lock speed and position accuracy, even in challenging environments. With an update rate of up to 10 Hz, it delivers real-time, high-refresh positional data to your flight controller—crucial for autonomous missions, waypoint navigation, and return-to-home functions.

It also integrates seamlessly with ArduPilot via UART, often including a compass module (like HMC5883L or QMC5883L), which helps stabilize heading and orientation. This makes the NEO-M8N a reliable, cost-effective, and flight-proven GPS solution for competitive UAV applications like Aerothon.

Figure 3.20: NEO-M8N GPS Module

HDMI-to-RCA Converter

We're usign this to convert the video output from Jetson Nano to analog via a HDMI cable output. This supports 480i/576i output (works with analog displays or FPV AV transmitters). Plug and play, no drivers or config needed.

Works with Raspberry Pi, laptops, or flight controllers that output HDMI (if needed for onboard video conversion).

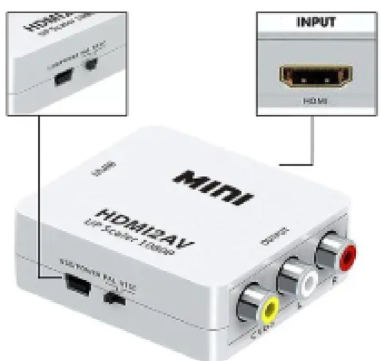


Figure 3.21: HDMI-to-RCA

3.6.2 Communication System Framework

RC Link

The Radio Control & Telemetry link being used is ELRS. ELRS stands for ExpressLRS. Some of the benefits of ELRS are -

- Compatibility and interoperability with Ardupilot GSC as well as ROS ecosystem.
- Long Range (compared to traditional 2.4GHz systems)
- Telemetry (over Mavlink)
- Versatile hardware and software support due to the open source nature.

The 2.4GHz ELRS communication link comprises of control and telemetry data, the packet rates and bandwidth of which can be adjusted according to the latency, range and signal strength required. The ELRS link is also fully compatible with the Ardupilot Flight Control software which provides open access to the Mission Planner Ground Control Software. This enables various smart features such as -

- Geofencing.
- Waypoint-based GPS navigation.
- Python language based control of the UAV during autonomous operation.
- Various failsafe triggers along with support for custom failsafe routines.
- Automation of the payload drop mechanism through the onboard flight computer via mavlink.

The Jetson Nano (flight computer) is the intermediary between the Digital cameras mounted on the UAV and the analog video transmitter unit. The schematic is as follows:

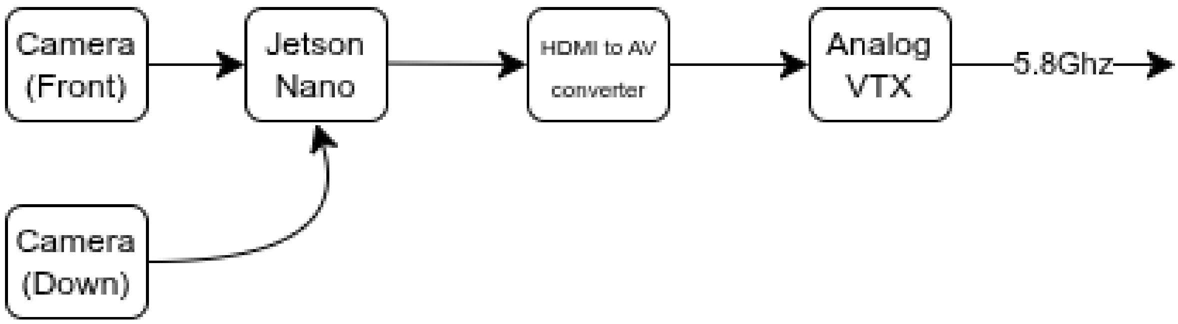


Figure 3.22: Schematic of Communication Framework

This setup allows for dynamic switching of both feeds as per need. The feed to be transmitted can be selected on GCS and the corresponding feed will be outputted over HDMI as an external display signal. This allows for onboard processing of the video feed on the UAV itself and subsequent transmission, avoiding latency issues encountered with offboard processing based systems.

3.7 Autonomous Navigation System

3.7.1 Hardware Setup

The hardware setup for autonomous navigation primarily consists of -

1. M8N GPS sensor with Magnetometer.
2. 8X 360 degree ultrasonic sensor array for proximity sensing (collision avoidance).
3. Front-facing camera sensor for scene detection (collision avoidance).
4. Down-facing camera sensor for identifying the landing pad.
5. Pixhawk 2.4.8 Flight Controller for UAV control after issuance from flight computer.
6. Jetson Nano (onboard flight computer) for mission execution and path planning.

3.7.2 Software Architechture

The software setup for autonomous navigation primarily consists of -

1. ROS2 for providing the inter-nodal communication during operation.
2. Ardupilot MissionPlanner GCS for geofencing and manual/autonomous operation control.
3. Tensorflow based custom trained Neural Network for disaster identification.
4. OpenCV for ”stiching the frames” from the downward feed to obtain a collective map of the area for more accurate counting.
5. Marker Tracker library for autonomous payload dropping.
6. Failsafe ROS2 node which stops the UAV in 3d space in order to a prevent collision by utilizing data from the Ultrasonic Sensor Array.

3.8 C.G. Calculation & Stability Analysis

Mass Properties of New UDSAV

Configuration: Default
Coordinate system: – default –

Mass: 734.38 grams
Volume: 686326.54 cubic millimeters
Surface area: 404467.91 square millimeters

Center of mass (millimeters):

- X = -42.86
- Y = 72.66
- Z = 18.98

Principal axes of inertia and principal moments of inertia (grams · mm^2) *Taken at the center of mass*

Axis	Principal Moment (g·mm^2)
$(0.000,\ -0.001,\ 1.000)$	$I_{px} = 4364311.10$
$(1.000,\ 0.005,\ 0.005)$	$I_{py} = 5070622.83$
$(0.005,\ 1.000,\ 0.001)$	$I_{pz} = 7387707.58$

Moments of inertia (grams · mm^2) *Taken at the center of mass and aligned with the output coordinate system*

$I_{xx} = 5077527.82$	$I_{xy} = -182965.60$	$I_{xz} = -2193.50$
$I_{yy} = 1236636.10$	$I_{yz} = 723658.63$	$I_{zz} = 4564390.02$

Moments of inertia (grams · mm^2) *Taken at the output coordinate system (using absolute accuracy)*

$I_{xx} = 9219433.51$	$I_{xy} = -2180617.87$	$I_{xz} = -599741.09$
$I_{yy} = 2610687.76$	$I_{yz} = 8990434.29$	$I_{zz} = 9591054.61$

Computational Analysis

4.1 CFD / FEM / MATLAB Simulations

CFD Analysis

The airflow around the drone was analyzed to optimize lift, minimize drag, and improve stability during low-speed operations like payload drop and hovering. Simulations focused on propeller wash interactions, pressure distribution, and vortex formation near the arms. This helped in validating the positioning of rotors and the overall aerodynamics of the UAV.

FEM Analysis

FEM simulations were analyzed on the frame and arm structures to assess stress concentration under typical flight loads and possible deformations. The goal was to verify that the ABS frame would withstand static and dynamic loads without deformation or failure. Stress, strain, and displacement plots were generated to ensure safety margins under thrust and payload weight.

Open-source and proprietary tools like ANSYS and SolidWorks Simulation were used. These analyses guided material thickness choices, arm length adjustments, and center of mass alignment to enhance both flight performance and structural durability.

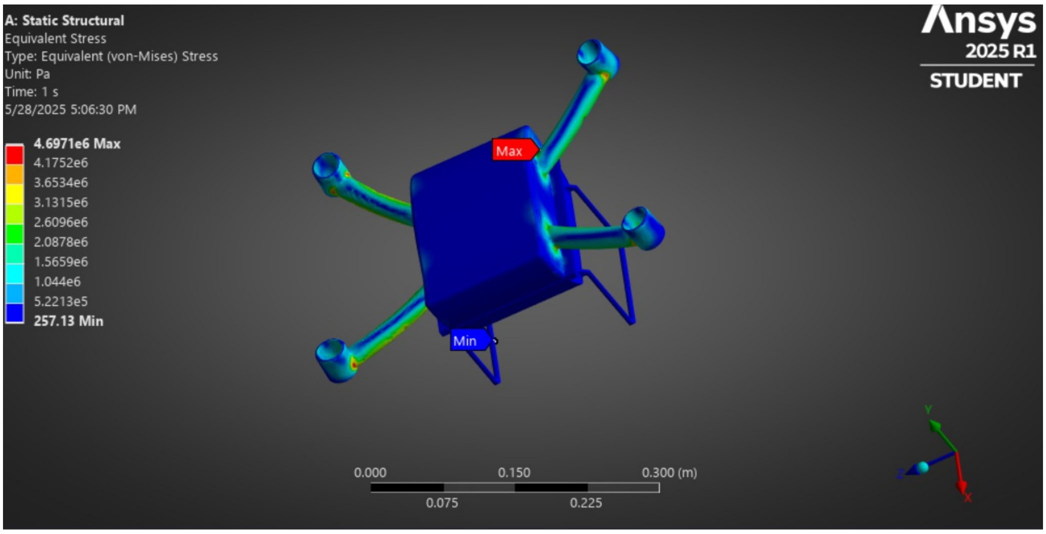


Figure 4.1: FEM Analysis

4.2 CAD Model and Performance Validation

A comprehensive 3D CAD model of the drone was created using SolidWorks to visualize and validate all mechanical subsystems.

1. **Spatial Layout:** The CAD model ensured that all components — motors, ESCs, battery, payload, flight controller, and camera — were properly mounted without interference. Special attention was given to propeller clearance, wiring space, and CG alignment.

2. **Mass and Balance Validation:** Using the CAD tool's mass properties feature, it is confirmed the total takeoff weight and ensured that the center of gravity (CG) fell within the desired range for stable flight.

3. **Design-to-Prototype Transition:** The finalized CAD was used to generate 2D technical drawings for 3D printing, and component fabrication. It served as the backbone for building a physically accurate prototype, ensuring minimal mismatch between design and build.

Safety & SORA Assessment

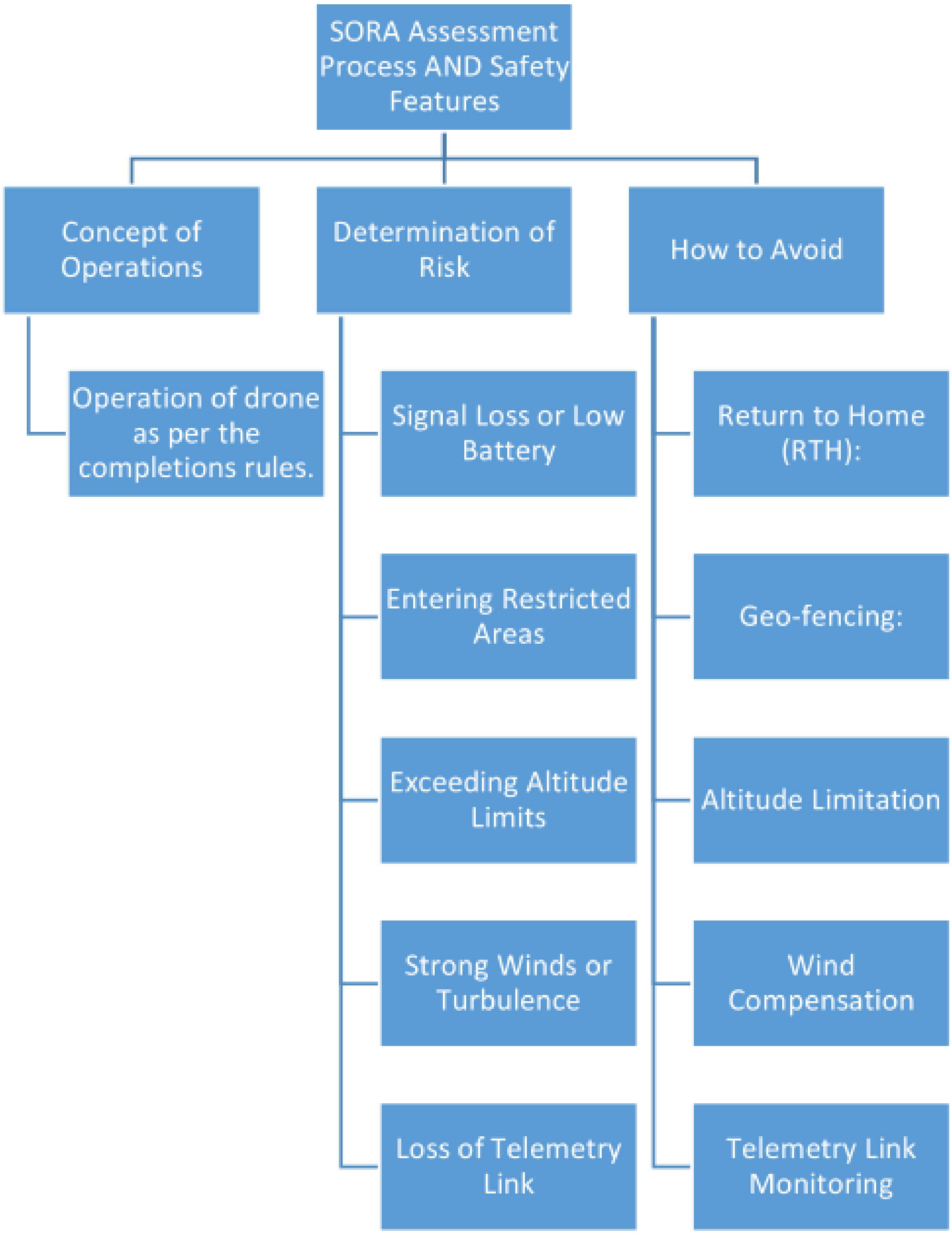


Figure 4.2

These features are present in the ardupilot software and can be configured to our needs. This helps us to ensure the safety of the drone and environment around it. Links to the saftey features in detailed will be shared in appendix

Methodology for Autonomous Operations

5.1 Flight Control Algorithm

The flight control algorithm primarily consists of a setup based upon the ROS2 architecture. Here, multiple operation node/addis will be executed as subroutines based upon their required use. The communication between the nodes will be based upon a subscriber-publisher relationship.

Under the various nodes, there is a primary node which decides upon the flight path. The flight path is generated using the geofence co-ordinates as the boundary then a maximum coverage algorithm is utilised to effectively scan the whole bound area for objects and disaster scenarios. The master node also runs the image recognition model which identifies objects in the frame. Then a counter slave node is called upon which inputs the incoming feed and outputs the number of objects identified under each class. This is a separate CNN meant for accurate counting. Another slave node is present for more classification of the disaster once a signal is received from the master node. The type of disaster is returned to the master node. The same feed is sent into a "headcount" which identifies the number for human bodies in the frame to look for potential survivors in the area, after notification from the master node.

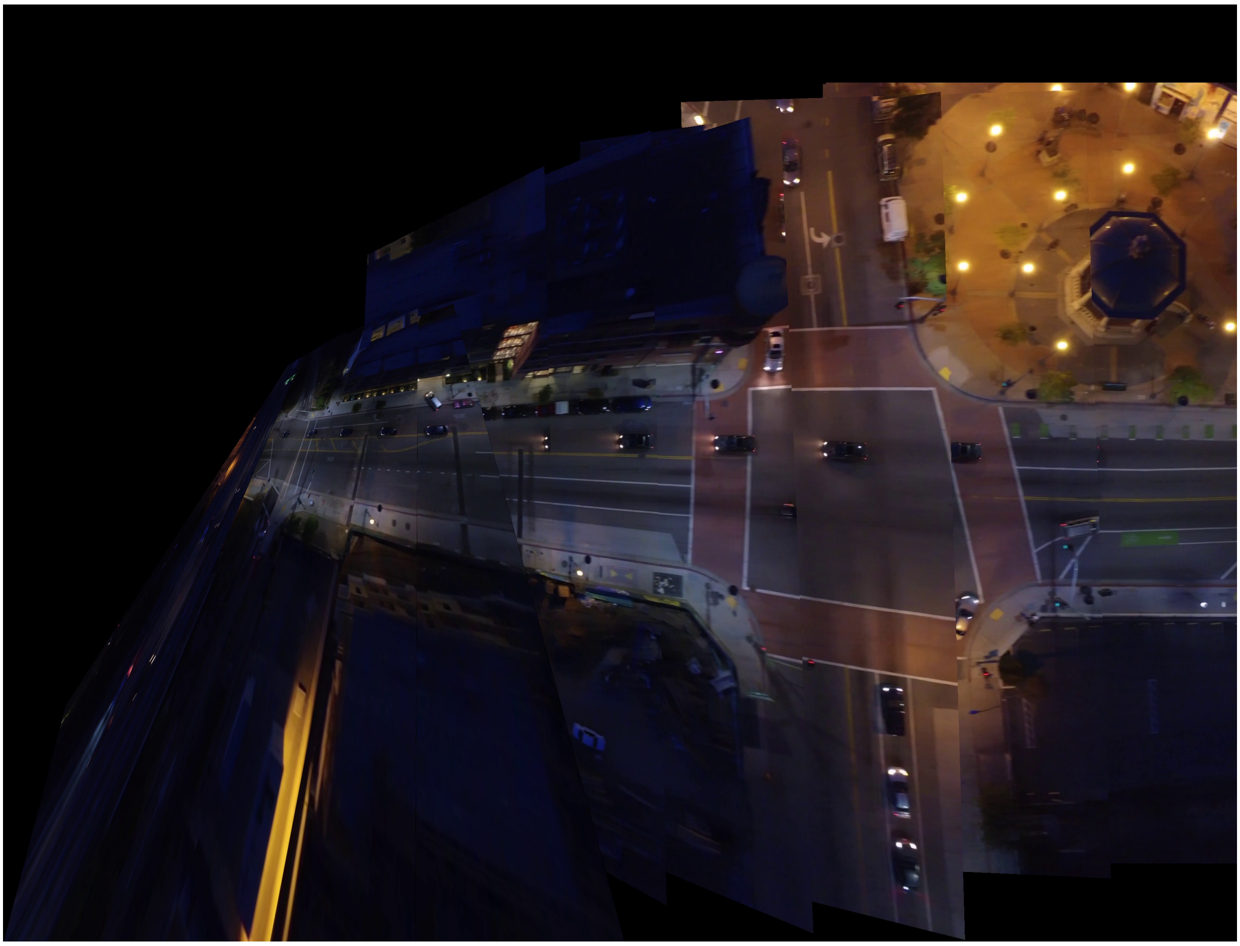


Figure 5.1: Stiching Using Panorama

As the mapping is completed, a "stiching" node inputs the video feed to turn the feed into a panorama map by utilizing the time series accelerometer data. This is then further fed into the object classification node for a final object count to ensure optimal accuracy. The following routine is executed during the Autonomous operation of the UAV. Our unique approach to cross checking with the stiched map ensures accuracy as it directly reduces the chances for one object getting counted multiple times if the UAV crosses an area twice. The divided processing approach based on the nodal architecture ensures optimal computation usage and saves on energy as well, which is an important factor for on-board processing systems.

5.2 Object Detection & Counting

Our model is built on deep learning frameworks like YOLOv8 trained extensively on real-world disaster scenarios:

- Human and Survivor Detection: Pinpoint stranded individuals quickly.
- Vehicle & Animal Identification: Understand evacuation conditions or trapped animals.
- Structural Feature Analysis: Detect broken rooftops, scattered debris, and partially collapsed infrastructure.

Beyond object recognition, the drone is equipped with environmental sensing intelligence:

- Flood Detection: Using color segmentation, motion tracking, and boundary anomaly mapping to recognize overflow zones.
- Fire Detection: Utilizing flame color profiles and motion-based flame dynamics; enhanced with thermal imaging when available.
- Damage Recognition: Edge detection and shape analysis identify collapsed structures and terrain changes that indicate danger zones.

Processing Pipeline

Each frame from the drone's onboard camera undergoes an accelerated data pipeline:

- Live Capture: Stabilized image stream from onboard camera.
- Fast Inference: Jetson device runs object detection using TensorRT-optimized deep learning models.
- Tracking & Validation: Deep SORT tracks objects persistently and filters false positives.
- Disaster Awareness Overlay: Visual tags like “Fire”, “Flooded Area”, or “Human Detected” are displayed with bounding boxes—turning raw footage into actionable insight.

Ground Impact

What good is vision if it doesn't translate into action?

- Real-time Alerts: Visual and audible notifications via onboard LEDs, buzzers, or wireless communication.
- Geo-Tagged Data Logs: Every detection is timestamped and GPS-tagged for immediate relay to rescue teams.
- Aerial Mapping: A bird's eye view of the hazard zones with object counts and threat labels—a living map to guide boots on the ground.

Figure 5.2

5.3 Autonomous Payload Drop Mechanism (Gripper)

In critical scenarios, delayed aid is denied aid. Our second module ensures that the drone does more than just report—it responds. Through a precisely-engineered servo-actuated gripper system, the drone can autonomously deliver essential supplies with centimeter-level precision—whether it's a medical kit, a water bottle, or communication tools.

Gripper Engineering

A durable, lightweight, two-finger gripper is mounted beneath the drone:

- Controlled by a high-precision PWM-driven servo, activated by the Jetson via GPIO.
- Payload capacity of 500g, with reinforced grip pads for flight stability.
- Designed for modular upgrades (magnet-based attachments, spring-loaded release, etc.).

Drop Intelligence

1. **Option A: GPS-Triggered Release**
    - Constantly monitors live GPS coordinates.
    - When drone enters target radius (e.g., 3–5m), drop logic initiates the release.
2. **Option B: Vision-Based Drop**
    - Uses onboard camera to detect high-contrast visual markers.
    - Verifies drone position and stability before executing release—a precise, visual-based delivery system.
3. **Built-In Redundancy**
    - Height and Position Checks: Confirms correct altitude before triggering drop.
    - Manual Override: Remote-based abort options via MAVLink.
    - Post-Drop Feedback: Sensor signal confirms payload release with optional auditory cue or digital flag.

Innovations and Future Scope

As a future enhancement, we envision integrating a stitching node that converts video feeds into a panoramic map using time-series accelerometer data. This map would feed into an advanced object classification node to ensure more accurate counts. By cross-referencing with the stitched map, we aim to reduce duplicate detections when the UAV revisits the same area. This nodal, divided-processing architecture not only boosts accuracy but also optimizes onboard computation and energy efficiency—key factors for real-time autonomous UAV operations.
Few images from our attempts at this innovation:

Figure 6.1: Image 1

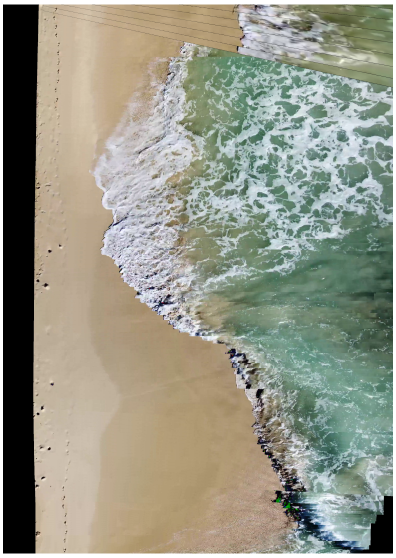


Figure 6.2: Image 2

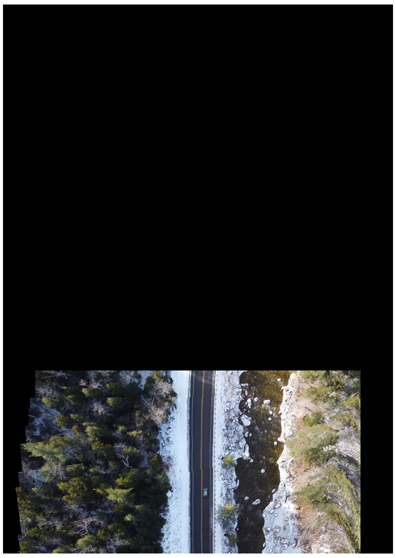


Figure 6.3: Image 3

The script used to iterate this process has been added in the appendix.

Hardware Enhancements

- Use carbon fiber composites to reduce weight further while increasing structural strength.
- Experiment with lightweight, impact-resistant polymers for additional durability.

Software Enhancements: Swarm Intelligence -¿ Enable multiple drones to collaborate for large-scale disaster assessments or coordinated payload delivery.

Mission Specific Enhancements: Disaster Response Toolkit

- Add modular attachments like water sprinklers for firefighting or chemical sensors for hazardous material detection.
- Equip drones with loudspeakers for communication during search-and-rescue missions.

Bill of Materials

Table 7.1: Bill of Materials

Component Name	Quantity	Unit Price INR
NVIDIA Jetson Orin Nano 8GB	1	36,499
Pixhawk 2.4.8 Flight Controller	1	11,179
Cameras	2	3,200
SpeedyBee BL32 50A 4-in-1 ESC	1	7139
DYS D2836-7 1120KV Brushless Motor	4	1,228
Orange Pro-Range 5200mah 11.1V	1	3,653
GPS – Neo M8N	1	1,500
HDMI-to-RCA	1	200
VTX	1	3000
SkyDroid Transmitter	1	12,075
Receiver	1	1,700
Servo	1	500
Frame	1	3,000
9-inch Propellers	4	200
Total	–	**90,357 /-**

Appendix