IRoC-U

ISRO Robotics Challenge - URSC

Let's build a space robot

Proposal Report

For IRoC-U 2025

by

Purple Watermelon
11239

December 2024

IIT Dharwad

1. Description of ANAV

The autonomous navigation for an aerial vehicle (ANAV) is designed to autonomously navigate a foreign rough terrain and identify safe landing spots for a drone. It integrates advanced sensing, navigation and decision making systems.

1. **Flight Controller:**
    a. The Hex Pixhawk Cube orange flight controller that supports open source firmware like ArduPilot and PX4 is used for autonomous navigation, flight control and stabilization. It integrates onboard sensors such as IMU's and barometers to facilitate real time telemetry and communication with ground stations.
2. **Companion Computer:**
    a. The Jetson Nano Xavier NX serves as the primary computational unit, used for processing data from sensors and runn ing neural network models.
3. **Sensors:**
    a. A LiDAR Camera (Intel RealSense L515 LiDAR) is used to map the landing arena to capture depth information that is further used to create a Digital Elevation Model (DEM).
    b. Other sensors include IMU and Barometer onboard the flight controller.
    c. Telemetry Modules for real time data exchange.
4. **Safe landing sites detection algorithm:**
    a. Defining a safe landing site
    b. Development of DEM
    c. Deep Neural Network (DNN)
        i. The Digital Elevation Model generated using the Point cloud from the LIDAR is passed through a DNN which performs pixel by pixel classification of the arena to determine if a spot is safe for landing.

5. **Autonomous Navigation:**
    a. Sensor Integration
        i. LiDAR: Utilized for obstacle avoidance by creating a point cloud of the surroundings, ensuring the drone navigates safely around obstacles. Generates a Digital Elevation Model (DEM) by scanning the environment

to identify height variations in the terrain. Combines DEM data to identify safe landing spots by evaluating the gradient of the terrain.

    ii. IMU (Inertial Measurement Unit): Provides orientation, angular velocity, and acceleration data to estimate the drone's pose and improve navigation accuracy.
    iii. Barometer: Offers altitude data to stabilize the drone in varying environmental conditions.

b. Digital Elevation Model (DEM) Generation
    i. The camera and LiDAR data are combined to develop a DEM that provides a 3D representation of the environment. This is crucial for navigating uneven terrains and identifying safe landing spots with the least gradient.
c. OctoMap for 3D Mapping
    i. The system employs OctoMap, a probabilistic 3D mapping framework, to create an Occupancy Map of the environment.
    ii. The OctoMap combines sensor data (LiDAR and camera) to generate a 3D representation of the arena, which is used for path planning and navigation.
    iii. This mapping technique ensures robust navigation in GPS-denied regions by relying on visual-inertial odometry and the occupancy grid.
d. Visual Inertial Odometry (VIO)
    i. The combination of visual data (from cameras) and inertial data (from the IMU) enables precise localization and trajectory estimation.
    ii. The VIO system corrects for drift errors and ensures accurate pose estimation during autonomous navigation.
e. Deep Neural Network Framework
    i. A Bayesian SegNet Model is applied to process noisy DEM inputs.
    ii. It identifies navigable regions and avoids areas of uncertainty, which are determined using MC Dropout Inference.
    iii. This enhances safety by avoiding unpredictable areas and improving the reliability of navigation decisions.
f. Path Planning and Guidance
    i. The navigation subsystem calculates optimal paths to the target destination using inputs from the OctoMap, DEM, and VIO.
    ii. The flight controller, Pixhawk Cube+, executes these paths while ensuring real-time stability and responsiveness.
g. Redundancy for GPS-Denied Navigation
    i. The combined use of LiDAR, cameras, IMU, and OctoMap eliminates reliance on GPS for navigation.

ii. This approach ensures reliable and accurate operation in environments such as dense forests, urban canyons, or indoor spaces where GPS signals are weak or unavailable.

h. Collision Avoidance and Emergency Handling

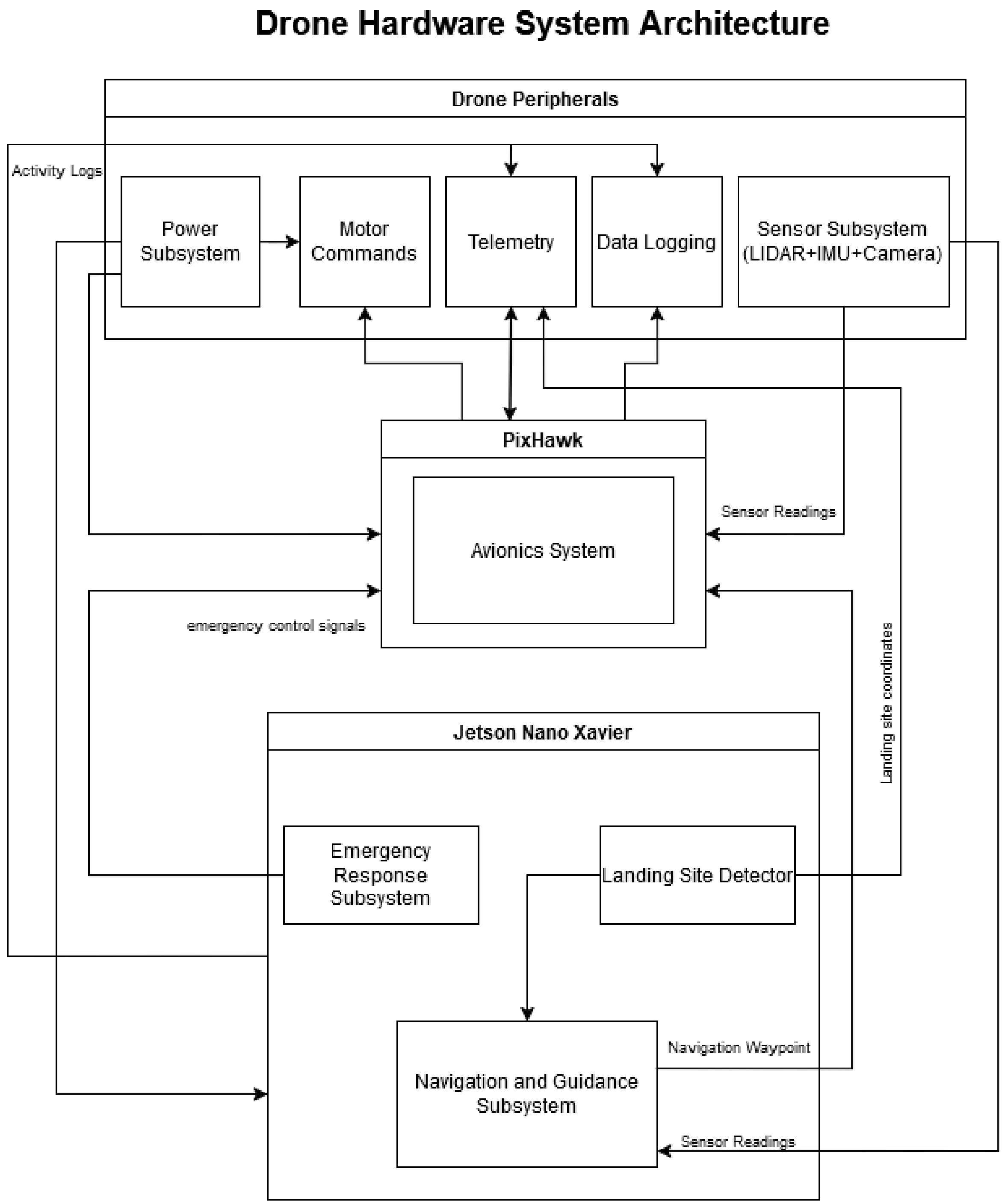
i. LiDAR and DEM data are continuously analyzed for obstacle detection and avoidance.

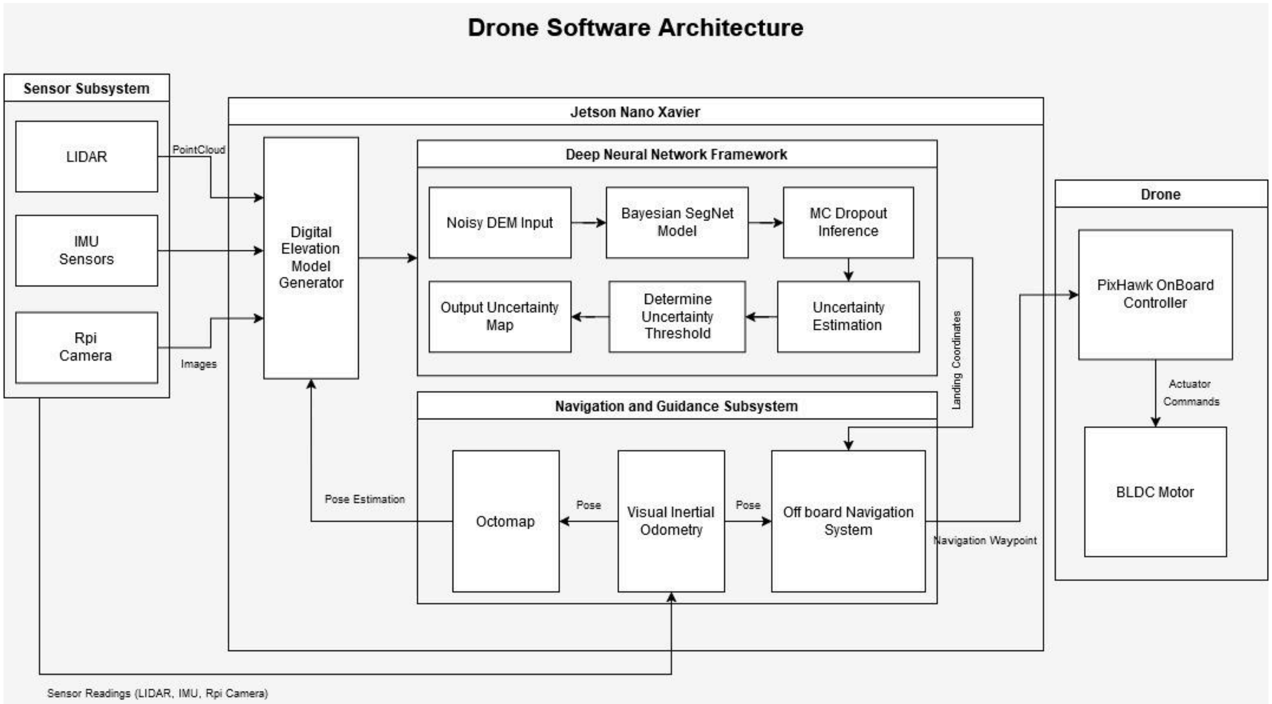
ii. The navigation system dynamically adjusts the flight path to prevent collisions while maintaining stability and trajectory.

6. Selection of a suitable Aerial Vehicle:

Aerial Vehicle	**Reasons for Accepting/Rejecting**
Quadcopter	This frame was selected for its lightweight design, structural stability, and ample space for component integration and ensuring efficient aerodynamics.
Hexacopter	This frame was *rejected* due to its excessive weight, larger size, and high-power consumption, which hinder efficient flight performance and maneuverability.
VTOL	This frame was *rejected* due to its complex flight mechanism, which complicates control and maintenance, and its limited payload capacity, making it unsuitable for integrating all mission-critical components.
Tricopter	This frame was *rejected* due to its inherent instability, compromising flight control and reliability, and its dependence on a servo mechanism for yaw control, increasing the risk of mechanical failure.

2. System Architecture





3. Identification of components with their specifications

3.1 Navigation Algorithm

Primary Sensors:

- **Intel RealSense L515 LiDAR:** Generates precise depth data for 3D mapping and pose estimation.
- **Raspberry Pi Camera V2:** Captures high-resolution imagery for visual odometry and surface analysis.
- **IMU (Inertial Measurement Unit):** Tracks drone orientation and motion for visual-inertial odometry (VIO).
- **TF Mini LiDARs:** Dedicated for real-time obstacle detection and avoidance.

Mapping and Localization:

- **Octomap Framework:** Creates a 3D voxel-based occupancy map from LiDAR and IMU data, identifying free, occupied, and unknown spaces.

- **Visual-Inertial Odometry (VIO):**
    - Combines IMU and camera data to estimate the drone's pose in real-time.
    - Eliminates drift errors in GPS-denied environments by fusing sensor data.

Systematic Coverage:

- **Lawnmower Motion Pattern:** Ensures complete arena coverage with minimal overlap, guided by Octomap to avoid redundant scans.
- **DEM (Digital Elevation Map):** A fused 3D map generated from LiDAR data to identify flat surfaces and safe landing zones.

Safe Landing Spot Detection:

- Uses DEM and gradient analysis from the LiDAR to find regions with the least undulation.
- The Raspberry Pi Camera V2 provides texture and surface details to avoid hazardous areas.

Obstacle Avoidance:

- TF Mini LiDARs scan the surroundings for real-time obstacle detection.
        - Octomap dynamically updates to include new obstacles.

```
// Initialization
Initialize Sensors (LiDAR, Stereo Camera, IMU, Barometer)
Initialize Flight Controller (Pixhawk Cube+)
Initialize Companion Computer (Jetson Nano Xavier NX)
Load Neural Network Model (Bayesian SegNet)
Initialize OctoMap for 3D mapping
Set Target Destination (waypoint or landing site)

// Main Loop for Autonomous Navigation
WHILE Drone is Operational:
    // Sensor Data Acquisition
    LiDAR_Data ← Capture_LiDAR_PointCloud()
    StereoCamera_Data ← Capture_Stereo_Images()
    IMU_Data ← Read_IMU_Sensors()
    Barometer_Data ← Read_Barometer()

    // Digital Elevation Model (DEM) Generation
    DEM ← Generate_DEM(StereoCamera_Data, LiDAR_Data)

    // Obstacle Detection and Mapping
    OctoMap ← Update_OctoMap(LiDAR_Data, DEM)

    // Pose Estimation using Visual-Inertial Odometry (VIO)
    Pose ← Estimate_Pose(IMU_Data, StereoCamera_Data)

    // Safe Landing Site Detection (if Landing is Required)
    IF Landing_Required:
        DEM_Uncertainty ← Run_Neural_Network(DEM, Bayesian_SegNet)
        Safe_Landing_Site ← Detect_Safe_Zones(DEM_Uncertainty)
        IF No Safe Site Found:
            Adjust Path and Search for Alternative Landing Zone
        ENDIF
    ENDIF

    // Path Planning and Navigation
    Path ← Plan_Path(OctoMap, Pose, Target_Destination)
    Navigation_Waypoints ← Generate_Waypoints(Path)

    // Collision Avoidance
    FOR Each Waypoint IN Navigation_Waypoints:
        Obstacle_Detected ← Check_For_Obstacles(OctoMap, Waypoint)
        IF Obstacle_Detected:
            Adjust_Path(Waypoint, Alternative_Route)
        ENDIF
    ENDFOR

    // Command Flight Controller
    Actuator_Commands ← Compute_Commands(Navigation_Waypoints, Pose)
    Send_Commands_To_FlightController(Actuator_Commands)

    // GPS-Denied Handling
    IF GPS_Lost OR GPS_Unreliable:
        Switch_To_VIO_Mode()
    ENDIF

    // Monitor IMU for Faults
    Fault_Detected ← Check_IMU_Faults(IMU_Data)
    IF Fault_Detected:
        Apply_Error_Correction(IMU_Data)
        Log_Fault_Event()
    ENDIF

    // Log Telemetry Data
    Log_Data(LiDAR_Data, DEM, OctoMap, Pose, Actuator_Commands)

    // Check Mission Completion
    IF Target_Destination_Reached:
        BREAK
    ENDIF
ENDWHILE

// Post-Flight Operations
IF Landing_Required:
    Execute_Safe_Landing(Safe_Landing_Site)
ENDIF
Shutdown_Sensors()
Shutdown_FlightController()
Log_Flight_Status()
```

snappify.com

Pseudocode for the 3D LiDAR

3.2 Safe landing site detection algorithm (Ref.)

Identifying Safe Landing Sites is crucial for planetary missions. The Landing site detection algorithm proposed is a robust, reliable and scalable solution that implements Bayesian Deep Learning (BDL) by combining safety predictions with uncertainty quantification. This method builds on deep learning techniques for semantic segmentation and incorporates measures to address challenges posed by sensor noise and prediction reliability.

Defining a Safe Landing Site:

An area of approximately 1.5 sq. m. which has the least undulation of surface is referred to as a safe spot.

Generating a DEM:

1. Collect LiDAR point clouds from different passes over the arena.
2. Filter point clouds to remove noise and extract ground points.
3. Align scans based on overlap and relative position using registration.
4. Merge the scans into a single scan using voxel filtering algorithms.
5. Post processing and refining to smoothen the DEM.

Approach:

1. Input Data and Processing:
    a. The algorithm utilizes a Digital Elevation Model (DEM) derived from the navigation of the arena with the help of L515 LiDAR (1024 x 768).
    b. Noisy DEM's are passed through a Bayesian SegNet which outputs a safety map, indicating whether each pixel is safe/unsafe for landing and an uncertainty map reflecting the prediction reliability.
2. Semantic Segmentation using Bayesian SegNet:
    a. Architecture: A convolutional encoder-decoder network with Monte Carlo Dropout during training and inference.
    b. Output: Multiple forward passes to generate a probabilistic safety map having pixel wise classification to determine safety positions.
3. Uncertainty estimation:
    a. The model combines aleatoric uncertainty and epistemic uncertainty using

predictive entropy.

   b. The uncertainty map indicates regions of less reliability of safety prediction, allowing to invalidate uncertain pixels.

4. Safety Map generation:
   a. An uncertainty threshold filters out unwanted pixels from contributing to the landing site identification, ensuring highly reliable predictions.

Advantages:

1. **Noise robustness**: The system performs effectively even when DEM's are highly noisy or data outside training distribution.
2. **Reliability**: Improving reliability of the model by using an uncertainty map.
3. **Flexibility**: The algorithm allows fine tuning of parameters to optimize trade-off between sensitivity and precision.

3.3 Sensors & Components Used:

Flight Controller

Product: Hex Pixhawk Cube+ Flight Controller Autopilot

(Product Link)

Purpose: Advanced flight control and navigation processing for autonomous operations, waypoint missions, and stabilized flight. Features triple redundant IMUs for reliable attitude estimation, isolated power supply, and vibration dampening. Supports multiple communication protocols including MAVLink and UAVCAN | Weight: 73g

Companion Computer

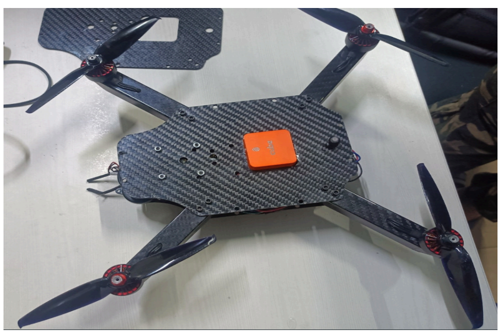
Product: NVIDIA Jetson Xavier NX Developer Kit (Product Link)

Purpose: High-performance AI computing and deep learning processing at the edge, featuring a 384-core NVIDIA Volta™ GPU with 48 Tensor Cores for accelerated AI workloads. Supports multiple neural networks running in parallel for computer vision, robotics, and autonomous systems. Includes comprehensive I/O including MIPI CSI camera ports and GPIO headers | Weight: 95g

Frame

a) Product: TBS 500 Frame (Product Link)

Purpose: Sturdy and spacious structure to house all components securely while maintaining stability and balance. Supports larger payloads and offers excellent cable management. Durable carbon fiber build | Weight: 400g



b) Product: Custom Carbon Fiber Frame

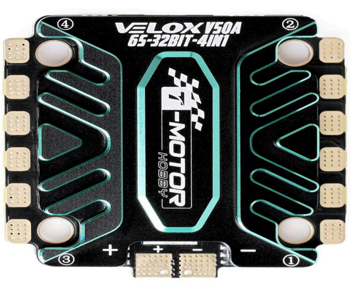
Purpose: Lightweight yet rigid frame construction using high-grade carbon fiber composite material for optimal strength-to-weight ratio. Features modular design for easy maintenance, integrated power distribution, and efficient propeller clearance for stable flight dynamics. Customizable mounting options for various payloads and components | Weight: 300g

Motors

Product: R2207 2207 2450KV Brushless Motor (x4)

(Product Link)

Purpose: powerful and efficient choice for drones, providing high RPM and excellent thrust for agile flight. | Total weight: 144g (36g each)



Electronic Speed Controllers (ESCs)

Product: T Motors Velox V50A 4in1 ESC (Product Link)

Purpose: It is designed to control brushless motors in drones with precision and efficiency. It supports up to 50A current per motor. |

Weight: 18.45g

Propellers

Product: 1045, 1085 or 6045 Carbon Fiber Propellers (4x) (Product Link)
Purpose: Generates thrust to achieve lift and ensures stable flight. Durable and lightweight construction. Optimized for smooth airflow | Total weight: ~40g (10g each)

Battery

Product: Orange 14.8V 5200mAh 40C 4S Lithium Polymer Battery Pack (Product Link)
Purpose: Provides sufficient power for prolonged flight and sensor operation. Lightweight with a high-discharge rate | Weight: 488g

Telemetry Module

Product: Holybro SiK Telemetry Radio V3 - 433 MHz (Product Link)
Purpose: Enables real-time data communication between the drone and ground station | Weight: 21g

Receiver Module

Product: Fly Sky FS IA6B 2.4GHz 6ChH PPM output with i-Bus port Receiver (Product Link)

Purpose:A 2.4GHz 6CH receiver with PPM and iBus output enables seamless communication between RC transmitters and flight controllers. It’s ideal for drones | Weight: 14.9g

Navigation Sensors

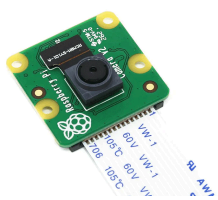
Option 1: Depth Camera

Product: Intel RealSense Depth Camera D455 (Product Link)
Purpose: Captures terrain depth information for landing and navigation.

Features: Depth Range: Up to 10 meters. Built-in IMU for enhanced SLAM accuracy | Weight: 120g

Option 2: Lidar camera
Product: Intel RealSense L515 Lidar Depth Camera

(Product Link)
Purpose: Combines AI and depth mapping for SLAM and real-time object detection.
Features:1920x1080 depth resolution, 30 fps frame rate, 0.25-10m range, and USB-C connectivity | Weight: 100g



Product: Raspberry Pi Camera V2 (Product Link)

Purpose: is utilized for safe landing by running AI algorithms to identify optimal landing spots and detect craters in real-time.It processes high-resolution imagery with machine learning models for terrain analysis and hazard avoidance | Weight: 3g

Product: TFMini-S Micro LiDAR Distance Sensor for Drones UAV UAS Robots (12m) UART (Product Link)

Purpose: provides precise distance measurement up to 12m for drones, enabling obstacle detection and altitude control. Its UART interface ensures efficient data communication for real-time navigation and collision avoidance | Weight: 5g

Emergency Shutdown

Product: Kill Switch Module

Purpose: Allows for immediate power cutoff during emergencies | Weight: 6g

Custom Landing Gear

Product: 3D printed landing gear
Purpose: Provides clearance for bottom-mounted sensors and ensures safe takeoff/landing.
Weight: 120g

Accessories

a) Vibration Dampening Mounts: For stabilizing sensitive components (e.g., Pixhawk, Jetson Nano). (Weight: ~20g)
b) Wires and Connectors: XT60 connectors and silicon wires for reliable power and signal connections. (Weight: ~50g)

Component Analysis and Selection Rationale

Flight Controllers

Considered Options:

a) **PIXHAWK 2.4.8**

Pros:

- Open-source architecture allowing custom firmware development
- Excellent community support
- Comprehensive sensor integration capabilities
- Cost-effective (₹16,000)
- Proven reliability in academic projects

Cons:

- Requires significant setup and configuration

b) **DJI N3**

Pros:

- Professional-grade performance
- Excellent stability
- Easy setup

Cons:

- Closed ecosystem limiting customization
- Higher cost (₹45,000+)

- Limited access to raw sensor data

c) **Orange Cube+ Flight Controller (*Selected*)**

Pros:

- Triple redundant IMUs provide excellent reliability
- Powerful STM32H7 processor for advanced flight modes
- Active community support and regular firmware updates
- High-quality vibration isolation for clean sensor data
- Built-in safety features like power redundancy

Cons:

- Premium price point (~₹60,000)
- Complex initial setup
- Larger size
- Requires carrier board purchase

Frame Selection

Considered Options:

a) **TBS 500 Frame**

Pros:

- Superior build quality with carbon fiber construction
- Excellent component spacing and layout flexibility
- Better vibration dampening characteristics
- Advanced cable management system
- Higher payload capacity with better weight distribution

Cons:

- Higher weight (400g)
- More expensive (₹8,000+)

b) **Custom Carbon Fiber Frame (*Selected*)**

Pros:

- Excellent strength-to-weight ratio
- Fully customizable design for specific needs

- High durability against crashes
- Clean power distribution integration options
- Professional appearance

Cons:

- High material cost
- Requires precision cutting/machining
- Limited repair options if damaged

Navigation Sensors

Considered Options:

a) **Intel RealSense L515 (*Selected*)**

Pros:

- Superior depth accuracy (up to 9m)
- High-resolution depth data (1920x1080)
- Compact and lightweight (100g)
- Excellent SDK support
- LiDAR-based technology for better performance in varying light
- Cost-effective compared to traditional LiDAR (₹45,000)

Cons:

- Shorter range compared to industrial LiDAR
- Requires additional processing power
- USB-C bandwidth requirements

b) **Livox MID-40 LIDAR**

Pros:

- Superior range (up to 260m)
- High precision
- Works in various lighting conditions

Cons:

- Heavy (590g) - exceeds our 2kg total weight limit
- Expensive (₹60,000-₹75,000)

- High power consumption
- Complex point cloud processing

c) **Luxonis OAK-D Pro**

Pros:

- Integrated AI capabilities
- Good depth range (12m)
- Lightweight (95g)

Cons:

- Higher cost than RealSense (₹30,000)
- Less established ecosystem
- Limited community support

Final Selection Rationale:

a) Flight Controller - Orange Cube+:

- Selected for its triple redundant IMUs and superior reliability in critical missions
- Powerful STM32H7 processor enables advanced autonomous features and custom algorithms
- Active firmware development and strong community support ensures long-term viability
- Excellent integration with ROS, PX4, and ArduPilot ecosystems
- Built-in safety features like power redundancy essential for professional operations

b) Navigation System - Multi-Sensor Approach:

- Primary: Intel RealSense L515 selected for:
    - High-precision LiDAR-based depth mapping
    - Excellent performance in varied lighting conditions
    - Strong SDK support and integration capabilities
- Secondary: Raspberry Pi Camera V2 for:
    - Visual navigation and feature tracking
    - Lightweight addition (3g) for redundancy
- Tertiary: TFMini-S LiDAR for:
    - Close-range obstacle detection
    - Reliable point measurement

c) Frame - Custom Carbon Fiber Frame

- Selected for its optimized strength-to-weight ratio, ensuring maximum flight efficiency while maintaining structural integrity
- Modular design allows for easy component integration and future upgrades
- Premium-grade carbon fiber construction provides superior durability and vibration dampening
- Customized mounting solutions for specialized payload requirements and sensor placement
- Professional-grade build quality ensures reliability during critical operations and extended flight times

Conclusion:

Our component selection process prioritized reliability, integration capability, and weight optimization while maintaining cost-effectiveness. The multi-sensor approach with L515, RPI Camera, and TFMini-S provides redundancy and comprehensive environmental awareness. This setup delivers the perfect balance of processing power, sensor accuracy, and structural integrity needed for successful autonomous navigation and landing operations in the environment.

4. Outline the plan for realization of ANAV

No.	Hardware Details	Procurement Source	Specifications/Realization Plan	Qty
1	Flight Controller	Market	Orange Cube+ Flight Controller Autopilot, triple redundant IMUs, STM32H7 processor	1
2	Companion Computer	Market	NVIDIA Jetson Xavier NX, 384-core NVIDIA Volta GPU, 48 Tensor Cores	1
3	Frame	Fabrication	Custom Carbon Fiber Frame, optimized for sensor mounting	1
4	Motors	Market	R2207 2207 2450KV Brushless Motor	4
5	Electronic Speed Controllers (ESCs)	Market	T Motors Velox V50A 4in1 ESC, 50A per motor	1
6	Propellers	Market	1045/1085 Carbon Fiber Propellers	4

7	Battery	Market	Orange 14.8V 5200mAh 40C 4S LiPo	1
8	Telemetry Module	Market	Holybro SiK Telemetry Radio V3 - 433 MHz	1
9	Receiver	Market	Fly Sky FS IA6B 2.4GHz 6CH PPM	1
10	Primary Camera	Market	Intel RealSense L515 LiDAR Camera	1
11	Secondary Camera	Market	Raspberry Pi Camera V2	1
12	Obstacle Detection	Market	TFMini-S Micro LiDAR Distance Sensor	1
13	Emergency Shutdown	Market	Kill Switch Module	1
14	Landing Gear	3D-Printed	Custom Landing Gear for sensor clearance	1
15	Accessories	Market	Vibration dampening mounts, wires, connectors	Multiple

Description:

a) Software Implementation Plan: The navigation and guidance software will integrate real-time image processing and AI-based terrain mapping, leveraging the powerful NVIDIA Jetson Xavier NX. It will employ visual SLAM algorithms using the RPI camera for localization, while the Intel L515 LiDAR camera provides precise depth mapping and terrain analysis. The TFMini-S LiDAR sensors handle close-range obstacle detection for enhanced safety. The Orange Cube+ flight controller, with its triple redundant IMUs, executes precise guidance commands and maintains stability using PX4/ArduPilot firmware. The system utilizes deep learning models for safe landing spot detection, processing sensor fusion data from multiple sources. Testing and iterative improvements will ensure robust communication between the flight controller, sensor array, and the Xavier NX companion computer.

b) Identified Hardware for Navigation and Guidance

Intel L515 LiDAR Camera: High-precision depth mapping and terrain analysis

Raspberry Pi Camera: Visual navigation and SLAM implementation

TFMini-S Micro LiDAR: Close-range obstacle detection and avoidance

Orange Cube+ Flight Controller: Advanced flight stability and autonomous control

NVIDIA Jetson Xavier NX: High-performance AI computations and real-time decision-making

Custom Carbon Fiber Frame: Optimized mounting for sensor array integration

Telemetry Module: Real-time communication with ground station for mission updates

5. Test Plan:

It is important to verify whether the realized system is performing to meet the intended requirements.

1. **Boundary Detection Test**
    a. **Objective**: Ensure the system accurately detects and stays within the yellow boundary.
    b. **Procedure**:
        i. Place the drone at various locations within and near the boundary.
        ii. Use the downward-facing camera to detect the yellow boundary using HSV filtering.
        iii. Simulate scenarios where the drone drifts toward or crosses the boundary.
    c. **Expected Outcome**:
        i. The drone detects the boundary with >95% accuracy.
        ii. The system corrects its trajectory to remain within the defined boundary.
2. **Pose Estimation and Visual Odometry Test**
    a. **Objective**: Validate the accuracy of pose estimation using the Intel RealSense L515 LiDAR and visual odometry.
    b. **Procedure**:
        i. Perform test flights in a controlled arena, logging pose data from VIO.
        ii. Compare the estimated pose with ground truth from a motion capture system.
        iii. Introduce environmental noise (e.g., vibrations, low lighting) to test robustness.
    c. **Expected Outcome**:
        i. Pose estimation error <10 cm in position and <2° in orientation.
        ii. Stable position tracking during motion.
3. **Systematic Motion Test**

   a. **Objective**: Verify that the drone performs complete arena coverage using systematic patterns.
   b. **Procedure**:
      i. Divide the arena into a virtual grid.
      ii. Program the drone to follow a lawnmower or spiral motion pattern.
      iii. Mark areas as scanned/unscanned in real-time.
   c. **Expected Outcome**:
      i. 100% coverage of the arena within the specified time.
      ii. Minimal overlap or redundancy in scanned areas.

4. **Obstacle Detection and Avoidance Test**
   a. **Objective**: Ensure the system detects obstacles and adjusts its trajectory in real-time.
   b. **Procedure**:
      i. Introduce static obstacles (e.g., boxes) and dynamic obstacles (e.g., moving objects).
      ii. Use the forward-facing camera or LiDAR to detect obstacles.
      iii. Observe the drone's reaction when an obstacle is detected.
   c. **Expected Outcome**:
      i. Obstacles detected with >90% accuracy.
      ii. Drone adjusts its trajectory smoothly and avoids collisions.
5. **Landing Execution Test**
   a. **Objective**: Verify precise and safe landing on selected spots.
   b. **Procedure**:
      i. Command the drone to land at detected safe spots.
      ii. Monitor landing accuracy and stability.
      iii. Repeat the test for different landing locations in the arena.
   c. **Expected Outcome**:
      i. Landing accuracy <10 cm deviation from the center of the selected spot.
      ii. Stable touchdown with no tilt or drift.
6. **Noise and Sensor Failure Test**
   a. **Objective**: Evaluate the system's resilience to noise, sensor errors, and failure scenarios.
   b. **Procedure**:
      i. Introduce simulated noise in camera and LiDAR data.
      ii. Simulate the failure of individual sensors (e.g., camera blackout or LiDAR failure).
      iii. Test the system's response and fallback mechanisms.
   c. **Expected Outcome**:
      i. Minimal degradation in navigation performance under noisy conditions.

ii. Failover mechanisms activate, and the system maintains functionality.

7. **Performance Under Disturbances Test**
    a. **Objective:** Test the drone's stability and performance under environmental disturbances.
    b. **Procedure:**
        i. Simulate environmental disturbances like wind or vibration.
        ii. Observe the drone's ability to maintain stable flight and accurate navigation.
    c. **Expected Outcome**:
        i. Stable flight with minimal deviation (<5 cm) from the desired trajectory.
        ii. Navigation and landing unaffected by disturbances.
8. **Octomap Integration Test**
    a. **Objective**: Evaluate the accuracy of the 3D occupancy map.
    b. **Procedure**:
        i. Fly the drone in a simulated arena and generate Octomap in real time.
        ii. Validate the map against a ground truth model of the arena.
        iii. Test dynamic updates by introducing new obstacles.
    c. **Expected Outcome**:
        i. Octomap accuracy >95%.
        ii. Dynamic obstacle integration within 1 second.
9. **IMU Fault Detection and Correction Test**
    a. **Objective**: Assess the system's ability to detect and correct IMU faults.
    b. **Procedure**:
        i. Simulate IMU errors by injecting noise or disabling axes (e.g., roll or pitch data).
        ii. Use redundancy from VIO to compare IMU readings with camera-based motion estimates.
        iii. Evaluate fallback mechanisms when IMU data becomes unreliable.
    c. **Expected Outcome**:
        i. Faults detected within 500 ms.
        ii. VIO compensates for faulty IMU data without significant drift (<2% positional drift

6. System specification

Sr. No.	Description	Specification
1.	Overall mass	1498.35g

2.	Overall dimensions	(350x350x150)mm
3.	Power requirements	14.8V, 4S battery
4.	Flight time per charge	15-20 minutes
5.	Number of propellers and their size	4 propellers and 6 inch
6.	Features	Equipped with Emergency Kill Switch, LiDAR, Cameras, VIO. Autonomous Mapping and Navigation. 3D- Reconstruction. Emergency Response system.
7.	Any other specifications	Nil

7. Overview of the Emergency Response System.

The proposed methods prioritize the safety of both the drone and its surroundings, ensuring effective recovery or controlled landings in critical situations.

1. **Low Battery Emergency**
    a. **Detection Method**:
        i. Use battery voltage and current sensors to monitor real-time energy levels.
        ii. A machine learning model predicts energy consumption based on historical flight data (e.g., flight path, speed, payload, and environmental conditions).
        iii. Estimate the energy required for returning to the home position using current pose data and flight dynamics.
    b. **Response**:
        i. If the remaining battery is less than the threshold required to return home:
            1. Abort the current task.
            2. Navigate to the nearest safe landing spot identified earlier.
        ii. If sufficient energy remains but is low:
            1. Issue a low-battery warning.
            2. Prioritize returning to the home position after completing critical tasks.
2. **Single Motor Failure**
    a. **Detection Method**:
        i. Monitor motor RPM and current draw using onboard sensors.
        ii. Compare RPM values across motors; a significant discrepancy indicates a motor failure.

b. **Response**:
   i. Activate a **thrust redistribution algorithm**:
      1. Reallocate power to the remaining motors to maintain stability.
      2. Reduce the payload or hover slowly toward a safe landing spot.
   ii. Use IMU data to dynamically adjust orientation and counteract torque imbalance caused by the failed motor.
   iii. Execute a controlled descent if continued flight is unsafe.

3. **Multiple Motor Failures (Catastrophic Failure)**
   a. **Detection Method**:
      i. If two or more motors fail, the drone becomes unstable and unable to generate sufficient lift.
      ii. Detected through sensor fusion: low RPM, erratic IMU readings, and loss of altitude.
   b. **Response**:
      i. Deploy an **emergency parachute** (spring-loaded or electronic deployment).
      ii. Slow the descent to avoid catastrophic damage.
      iii. Send an emergency alert with the drone's last known location to the base station.

4. **Battery Burst or Fire**
   a. **Detection Method**:
      i. Monitor battery temperature using thermal sensors.
      ii. Trigger an alert if the temperature exceeds a critical threshold (e.g., >60°C).
   b. **Response**:
      i. Shut down all non-essential systems to prevent further damage.
      ii. Trigger an emergency landing at the nearest location, prioritizing safety.
      iii. Deploy fire-resistant materials onboard to contain the fire and minimize damage.

5. **IMU Failure**
   a. **Detection**:
      i. Compare IMU data with visual motion estimates from VIO.
      ii. Sudden discrepancies (e.g., erratic roll/pitch values) trigger an IMU fault alert.
   b. **Response**:
      i. Switch to camera-only visual odometry.

ii. Recalibrate IMU in-flight if possible, or prioritize landing at the nearest safe spot.

6. **Collision or Mid-Air Impact**
   a. **Detection**:
      i. Monitor sudden altitude drops or changes in orientation using IMU and TF Mini LiDARs.
   b. **Response**:
      i. Hover and stabilize using IMU corrections.
      ii. Identify the nearest safe landing spot and initiate an emergency descent.
      iii. Incase of excessive damage, the drone faces total loss of control and an emergency alert with the drone's last known location is sent to the base station.

8. Project management

No.	Task	Main Responsibility	Deadline for Completion	Secondary Responsibility
1	Hardware Procurement	Karan Kumar	25-01-2025	Shreyas Reddy
2	Algorithms	Saipushkar Nagaraj	15-01-2025	Yash Halbhavi
3	Hardware Implementation	Karan Kumar	28-02-2025	Prajwal Koppad
4	Software Testing and Simulation	Aryan Bandaru	21-02-2025	-

To manage the schedule for the project, the following are the strategies we plan to implement to ensure a successful completion of the project:

1. **Create a Detailed Timeline**
   We plan to break the tasks into smaller milestones and sub-tasks within the deadlines provided.
2. **Daily and Weekly Progress Monitoring**
   We will conduct daily check-ins to monitor individual and team progress, addressing any challenges immediately along with weekly reviews to help us evaluate overall progress, resolve roadblocks, and ensure alignment across all tasks.
3. **Task Dependency Management**

We will identify and prioritize tasks that depend on others. By coordinating effectively, we'll minimize risks related to dependencies.

4. **Risk Management**

   We'll anticipate potential challenges, such as delays in hardware delivery or software bugs, and prepare contingency plans.

5. **Testing and Iteration**

   We will allocate sufficient time for testing and simulation to ensure the drone meets the competition performance standards.

9. Novelty in the overall proposal

1. **Semantic Segmentation using DNN accounting for Uncertainty:** Integrates Bayesian SegNet for pixel-level safety classification with confidence estimation.
2. **Improved Safety Margins for accurate predictions:** Adjusts uncertainty thresholds dynamically to enhance prediction validity while minimizing risks in safety-critical applications like planetary landing.
3. **Efficient Memory Use via an Octomaps:** Stores sparse 3D data compactly, reducing computational load while retaining accuracy.
4. **Collision Avoidance:** Directly integrates with path planning algorithms for obstacle avoidance and navigation.
5. **Advantages of Technologies:**
   a. DEM (Digital Elevation Model):
      i. **Terrain Analysis:** Provides precise elevation data for studying terrain features like slopes and roughness, which are essential for safe landing site detection.
      ii. **Noise Handling:** DEMs can integrate data from multiple scans and apply filters to reduce noise, offering reliable representations of terrain.
      iii. **Versatility:** Usable across different planetary surfaces (e.g., Mars, Moon) for mission-critical tasks like hazard detection and navigation.
   b. Bayesian Deep Learning:
      i. **Uncertainty Quantification:** Combines aleatoric (sensor noise) and epistemic (model reliability) uncertainty to improve prediction reliability.
      ii. **Robust Performance:** Handles noisy input data effectively, ensuring reliable outputs in scenarios with incomplete or degraded data.

10. Declaration format

Declaration

We hereby declare that the aerial vehicle (rotorcraft) built/procured by team Purple Watermelon, college/institute IIT Dharwad, PERMANENT CAMPUS, CHIKKAMALLIGAWAD, DHARWAD - 580 011, KARNATAKA, complies to Drone rules-2021, issued by Ministry of Civil Aviation as per the Gazette of India CG-DL-E-26082021-229221 or the latest version.

Team lead: Saipushkar Nagaraj

Mentor-1: Ameer Mulla

Mentor-2: Arvind Pandit