UNIVERSITY OF BRISTOL

MSc in Aerial Robotics

Company Project Report

Autonomous Aerial System for Wildlife Survey and Targeted Payload Delivery

Team Name:
AeroZeno (Company 2)

Team Members:
Thejas Thomson
Sai Pavan
Ming Ye
Nitin Kesarla Rajendrakumar
Swapnil Roy

Submission Date:
March 27, 2025

Contents

List of Figures

List of Tables

1 Introduction

1.1 Context and Background

Effective wildlife preservation is primarily reliant on reliable monitoring and non-intrusive interventions. Traditional approaches such as physical tracking, tagging, and sensor placement can disrupt animal habitats and endanger both species and conservation staff. To address these issues, autonomous aerial robots have emerged as a vital tool, allowing for distant sensing, population surveys, and targeted payload delivery without requiring direct human intervention.

This project tackles these important conservation responsibilities by creating an **Autonomous Aerial System for Wildlife Survey and Targeted Payload Delivery in Conservation Zones**. The system is intended to autonomously survey a certain region, precisely detect and record zebra numbers, and strategically deliver payloads near recognised elephants inside a preset operational zone at the University of Bristol's aerial robotics laboratory, Fenswood Farm.

Figure 1: Designated Survey Area and Sensitive Zone within Fenswood Farm test site, as defined by the project brief.

1.2 Project Overview

The University of Bristol Flight Lab WildDrone team presented us with a project brief that clearly specified our major goals. Key criteria included:

- An autonomous census of zebras using aerial imagery.
- Deploy a sensor within 20 meters of a detected elephant.
- Meet operational safety standards, including geofencing and altitude limits.

To achieve these objectives, our solution combines cutting-edge machine learning-based animal recognition technologies with reliable aerial hardware and software platforms. We initially planned for onboard inference capabilities, but due to practical limits, we modified our strategy to use offboard processing on a ground station for increased dependability and simplicity of deployment.

1.3 Company Description

Our team worked as a specialised UAV solution provider, bringing together different engineering knowledge. While the majority of participants had backgrounds in aerospace engineering, one had a degree in energy, power, and automation, which provided a distinct viewpoint. This multidisciplinary basis boosted our expertise in aerial robotics, software development, hardware integration, and operational logistics. The team's structure encouraged collaborative, adaptive decision-making, allowing us to successfully respond to technical issues using a well-rounded, solutions-focused approach.

1.4 Team Member Roles and Responsibilities

Member	Responsibilities
Sai Pavan	Handled mission planning and waypoint scripting using Mission Planner, autonomous flight operations, initial onboard inference testing with Raspberry Pi, and managed ground station operations as Ground Station Lead.
Thejas Thomson	Developed and led the computer vision pipeline, including dataset generation, image annotation using LabelImg, YOLOv8n model training, video frame extraction, and system evaluation. Implemented the off-board ground station inference pipeline and acted as the project's Safety Pilot.
Ming Ye	Constituted the Hardware and Flight Operations Team, overseeing drone assembly, payload integration, and hardware maintenance.Collaborated on mission planning by setting waypoints and configuring geofencing. Supported the setup of pymavlink for drone control.
Nitin K R	Constituted the Hardware and Flight Operations Team, overseeing drone assembly, payload integration, and hardware maintenance. Designed and 3D printed custom mounts for securely attaching the Raspberry Pi and camera modules. Participated in field flight testing and on-site hardware adaptations.
Swapnil Roy	Constituted the Hardware and Flight Operations Team, responsible for drone assembly, payload integration, drone maintenance, and operational flight testing.

Table 1: Team Contributions

2 Design Rationale

This section describes and justifies the important design decisions made during the project's life cycle. Each choice is explored in terms of practical restrictions, technological requirements, and stakeholder demands, as well as a thorough STEEPLE analysis that demonstrates holistic thought.

2.1 Key Design Decisions

2.1.1 Object Detection Model Comparison and Justification

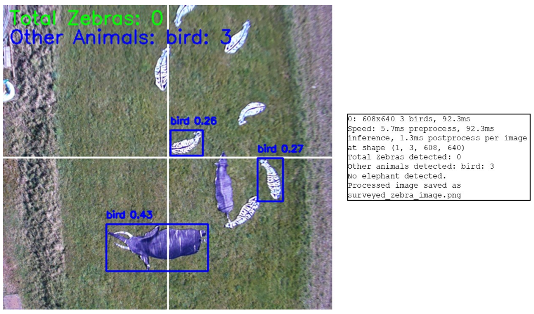
During the development of our vision pipeline, we assessed many common object detection models based on inference speed, detection accuracy, model size, and deployment simplicity. The table below summarises this comparison, focussing on the real-time detection of animals from aerial images on low-power platforms.

Model	Speed (FPS)	Accuracy (mAP)	Model Size	Ease of Deployment	Comments
YOLOv5n	Very High	Moderate	Small	Easy	Fast and lightweight; however, YOLOv5 has limited future development focus.
YOLOv7-tiny	High	High	Medium	Moderate	Good performance but lacks native Ultralytics integration.
YOLOv8n	Very High	High	Small	Very Easy	Latest Ultralytics model; lightweight and easy to train and deploy.
SSD MobileNet	Moderate	Moderate	Small	Easy	Efficient on edge devices but lower accuracy on aerial views.
Faster R-CNN	Low	Very High	Large	Complex	High accuracy but unsuitable for real-time or onboard use.

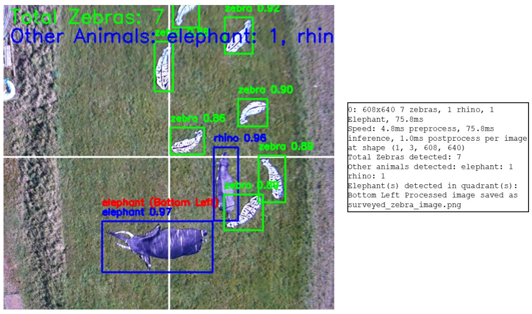
Table 2: Comparison of Object Detection Models

We eventually chose YOLOv8n, the smallest variation of Ultralytics' current YOLOv8 model, for its greater speed, accuracy, and deployment ease. Unlike YOLOv7 and Faster R-CNN, which need more complicated integration or considerable processing resources [1, 2], YOLOv8n delivers real-time performance with excellent accuracy, even on tiny datasets and edge devices [3]. It also has a well-maintained and beginner-friendly Python API, which allows for quick iteration and debugging during the dataset training and detection procedure [4].

Furthermore, YOLOv8n's model architecture is modular and completely interoperable with Ultralytics' training pipeline, simplifying the training and validation process. Given the time restrictions and the requirement for dependability in a field-tested system, YOLOv8n provided the most effective option.



(a) Detection using pre-trained YOLOv8n model



(b) Detection after training with custom dataset

Figure 2: Comparison of YOLOv8n detection before and after training on custom aerial dataset.

2.1.2 Onboard vs. Offboard Processing

The first project concept called for onboard processing using the Raspberry Pi 5 and a global shutter camera for real-time object identification and decision-making. However, integration challenges and stability concerns occurred during the development and testing phases. To assure mission success and decrease risk, it was decided to move inference operations to a more robust ground station laptop, streamlining the mission pipeline and using more powerful hardware resources to reliably handle real-time data processing.

Feature	**Original System**	**Fallback System (Implemented)**
Processing location	Onboard Raspberry Pi	Offboard Ground Station Laptop
Real-time Processing	Yes	No, post-flight analysis
Quadrant Determination	Immediate (real-time onboard)	Post-flight analysis
Mission Continuity	Seamless onboard decision making	Requires drone landing & data transfer
Reliability	Lower (hardware limitations encountered)	Higher (uses ground station)
Payload Deployment	Autonomous (onboard inference)	Manually triggered post-analysis

Table 3: Onboard vs. Offboard Processing Trade-offs

2.1.3 Mission Autonomy and Waypoint Planning

The crew used a waypoint-based navigation method guided by Mission Planner. Two-step autonomous missions were implemented:

- **Step 1:** Conduct a specified waypoint mission to scan the chosen region and capture photographs at 5-second intervals.
- **Step 2:** Ground station analyses photos to identify the quadrant with the target elephant. Following that, one of four pre-programmed waypoint missions was selected and uploaded autonomously to ensure that the payload was delivered precisely.

This modular and gradual autonomy provided for explicit milestones, reducing mission complexity while increasing flexibility and reliability.

2.2 Risk Assessment

A comprehensive risk assessment was performed to identify possible concerns and develop appropriate mitigation strategies.

		Pre Mitigation				Post Mitigation		
No.	Risk Description	Likelihood	Impact	Risk Score	Mitigation Strategy	Likelihood	Impact	Risk Score
1	Hardware failure (Raspberry Pi, ESC, etc.)	3	4	12	Maintain redundancy (spare parts) Implement offboard processing fallback	2	3	6
2	Loss of GPS signal	2	4	8	Use Return-to-Home (RTH) failsafe Geofencing & thorough preflight checks	1	4	4
3	Insufficient onboard computing power (e.g., Pi constraints)	4	3	12	Shift inference to ground station Optimize YOLO model	2	3	6
4	Poor detection accuracy (YOLO model)	3	4	12	Expand & refine training dataset Validate detection in real-world scenarios	2	3	6
5	Payload deployment inaccuracies	3	4	12	Quadrant-based waypoint missions Manual release post-landing	2	3	6
6	Adverse weather conditions	3	3	9	Monitor forecasts closely Contingency plans for bad weather	2	3	6
7	Data loss or corruption during flights	2	3	6	Systematic backups & logging Validate data integrity in real time	1	3	3
8	Skill/Expertise Gaps (Human Risk)	3	2	6	Assign tasks based on individual skill strengths Assign 2 or more members per team	2	2	4
9	Timeline Constraints/Delays	4	3	12	Develop a detailed Gantt chart with clear milestones Conduct frequent progress reviews	2	3	6

Figure 3: Risk Assessment Matrix

Risk Scoring: Likelihood × Impact *1–4: Low risk* *5–9: Medium risk* *10–16: High risk*

2.3 STEEPLE Analysis

Factor	Detailed Consideration and Project Impact
Social	- Reducing human-wildlife conflicts: Drone surveys and payload delivery greatly reduced direct human intervention, which improved both safety and conservation efforts. - Educational impact: The project's execution proved the potential of autonomous technology for conservation education and community participation.
Technological	- Edge AI optimisation: YOLOv8n was chosen for its effective mix of precision and computing needs, which allows for deployment on low-power platforms. - Robust system design: The ground station fallback offered technological robustness, allowing operational success despite hardware integration problems with the Raspberry Pi.
Economic	- Cost-effective: Used existing university resources (Hexsoon EDU-450, Cube Orange, Raspberry Pi) and free open-source software (YOLO, ArduPilot, LabelImg) to drastically reduce overall project expenditures. - Future scalability: The modular system architecture encourages reuse and adaption in future conservation applications, hence increasing long-term economic feasibility.
Environmental	- Low ecological footprint: Aerial surveys reduced the ecological disruptions caused by hand surveys, tagging, and tracking. - Emission-free operation: Electric propulsion and meticulous cargo deployment methods minimised environmental effect during flights.
Political	- Alignment with conservation policies: The system directly supports biodiversity and wildlife management initiatives promoted by governmental and international organisations (e.g., EU Horizon, WildDrone.eu). - Improved accessibility: Remote operating capabilities enabled usage in restricted or politically sensitive conservation areas without human intervention.
Legal	- Compliance: All flying operations strictly followed CAA laws, university drone safety processes, and established flight rules (geofencing, altitude limitations, RTH failsafes). - Operational oversight: A dedicated safety pilot monitored legal compliance and operational safety throughout the project's implementation.
Ethical	- Non-invasive methodology: The training used fake cut-outs rather than live animals, ensuring that there were no ethical concerns about animal suffering. - Passive monitoring: Payload deployments were carefully planned to prevent direct encounter with wildlife, emphasising ethical responsibility in conservation activity.

Table 4: STEEPLE Analysis of the Project

2.4 Project Planning and Gantt Chart

Effective project management was crucial for timely delivery and adaptation to challenges. A detailed Gantt chart was utilized throughout the project lifecycle to track progress, assign tasks, and manage resources efficiently.

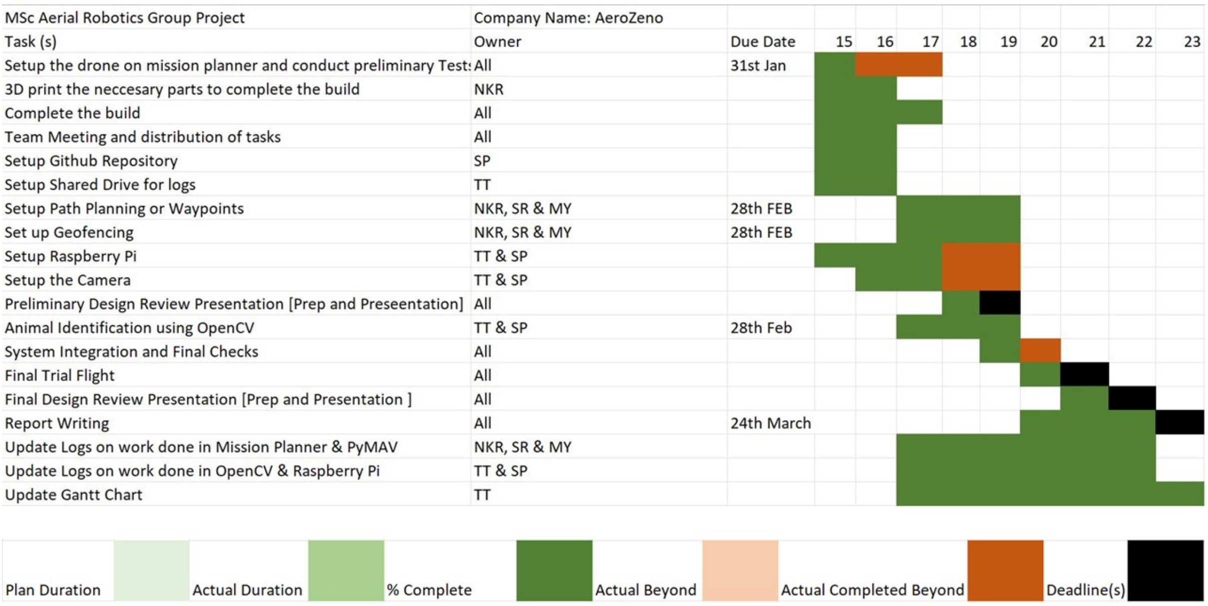


Figure 4: Project Gantt Chart

3 System Description

The Autonomous Aerial System, developed for wildlife scanning and targeted payload distribution, is comprised of multiple interconnected subsystems, each having a key role in accomplishing project objectives. Due to practical issues, the original system was transformed into a backup solution that maintained modularity, operational stability, and ease of deployment and maintenance.

3.1 Hardware Overview

The Hexsoon EDU-450 drone was selected as the aerial platform due to its durable and modular architecture, which makes it well-suited for aerial robotics applications. The drone is powered by four HS2216 KV920 motors, each driven by its own Electronic Speed Controller (ESC). A Hexsoon Power Distribution Board (PDB) is used to manage power delivery, ensuring consistent voltage supply to essential flight components.

Key Hardware Components:

- **Hexsoon EDU-450 Drone Platform** – Provides a stable, modular frame for integration of all components.
- **Cube Orange Flight Controller** – Enables autonomous flight, GPS waypoint navigation, and safety features such as geofencing and Return-To-Launch (RTL).
- **Here 3+ GPS Module** – Delivers precise global positioning data essential for navigation and accurate payload delivery.
- **Raspberry Pi 5** – Initially intended for onboard inference, now used primarily for aerial image capture and sensor interfacing.
- **Raspberry Pi Global Shutter Camera** – Captures high-resolution aerial images to ensure consistent and blur-free animal detection.

- **Tarot Payload Release Mechanism** – Facilitates reliable release of sensor payloads over the identified drop zone.
- **Ground Station Laptop** – Handles mission planning, autonomous script uploads, and post-flight image processing using YOLOv8n.

3.2 System Overview

Software components include ArduPilot firmware running on the Cube Orange, which enables autonomous flying, waypoint-based navigation, geofencing, and safety fail-safe features. Additional software components include:

- **Mission Planner:** Ground station software for designing, executing, and monitoring waypoint-based missions.
- **YOLOv8n Object Detection Model:** Trained on aerial images and video frames to identify zebras and elephants.
- **Python Scripts:**
  - Image and video processing for animal detection and counting using YOLOv8n.
  - Frame extraction from video clips to enhance training datasets.
  - A mission script initially designed for onboard autonomy, later adapted for offboard processing. (See Appendix for full script.)

Figure 5: Drone surveying the designated test area at Fenswood Farm with simulated wildlife cut-outs placed for aerial detection.

3.3 Operational Process and Mission Execution

3.3.1 Original (Intended) Operating Flow

- Drone and Raspberry Pi onboard system initialization.
- Autonomous waypoint mission to survey the region and capture images at fixed intervals.

- Real-time onboard image processing using YOLOv8n model to detect elephants and identify quadrants.
- Autonomous payload deployment based on onboard detection.

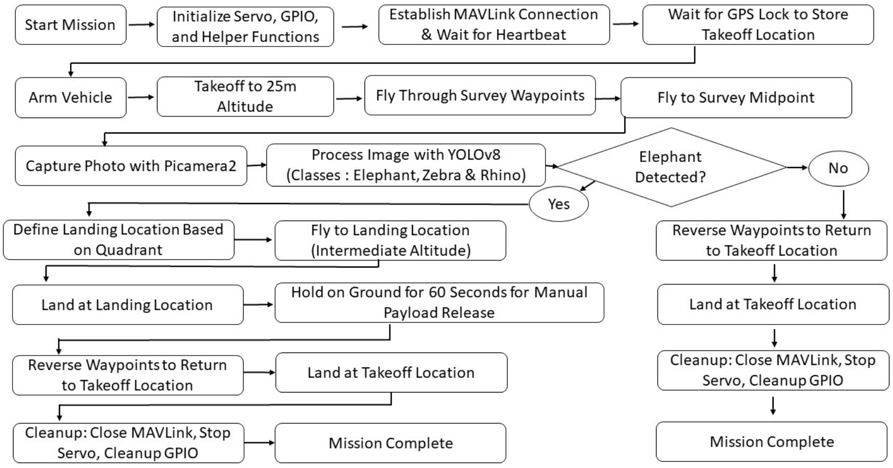


Figure 6: Flowchart of Original (Intended) Mission

3.3.2 Fallback (Implemented) Operating Flow

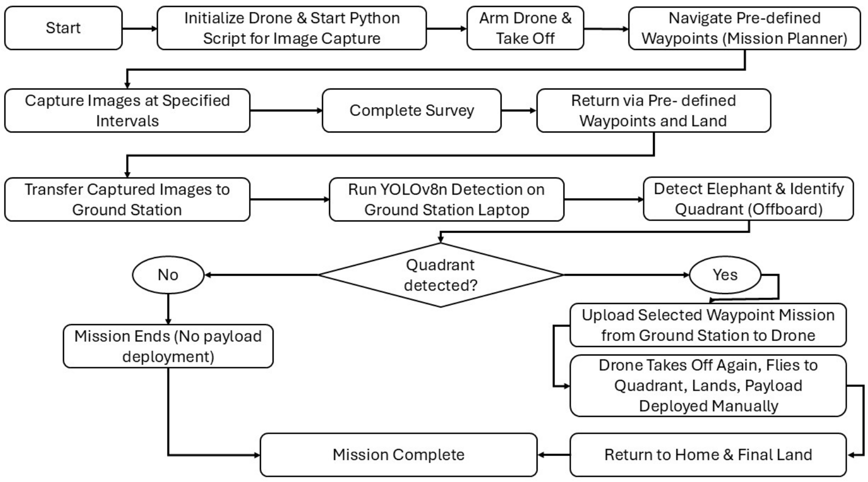


Figure 7: Flowchart of Fallback (Implemented) Mission

- **Initialization and Survey:** Autonomous flying using Mission Planner to capture aerial images at 5-second intervals.

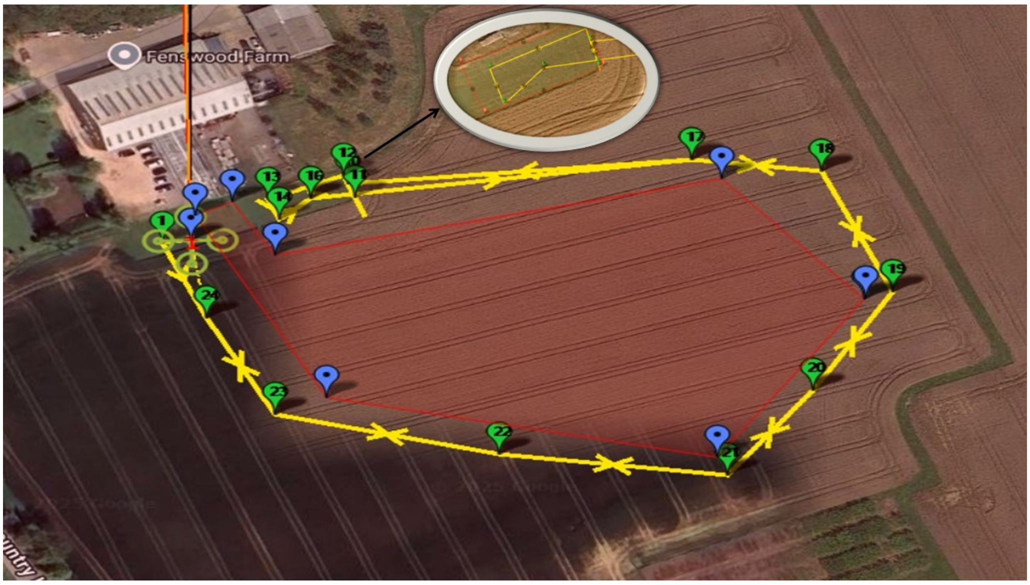


Figure 8: Survey waypoint plan covering all quadrants.

- **Post-flight Data Transfer:** Manual transfer of images to ground station.
- **Inference and Quadrant Analysis:** Ground station runs YOLOv8n to detect elephants and determine their quadrant.

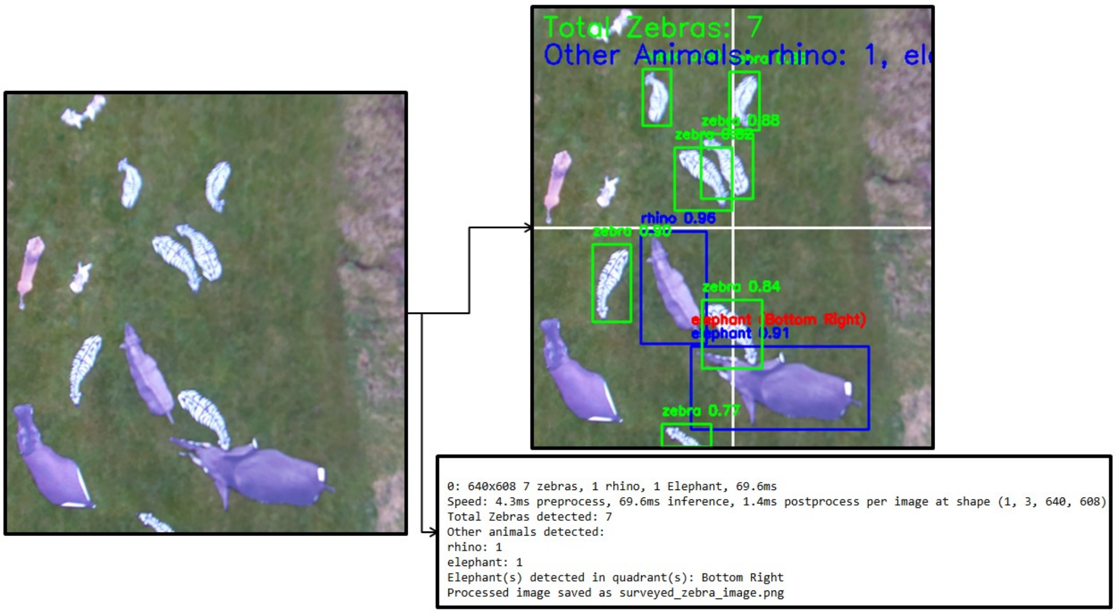


Figure 9: Post-survey inference showing detected animals and quadrant-based elephant localization. Elephant detected in bottom-right quadrant.

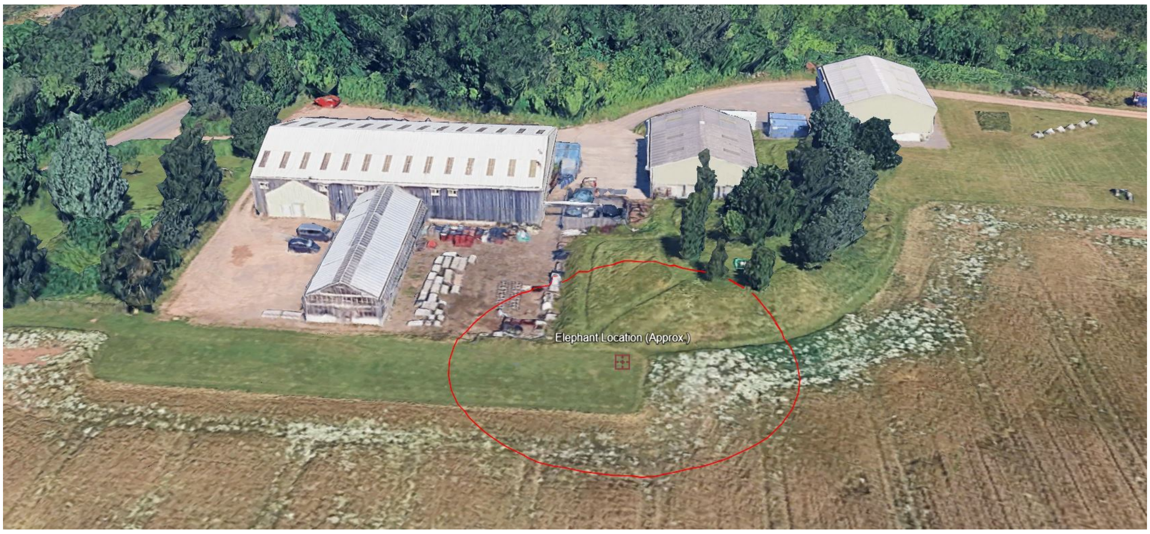


Figure 10: Elephant location and 20m drop zone visualised in Google Earth for accuracy validation.

- **Payload Deployment:** Based on analysis, a matching waypoint mission is uploaded to the drone.

Figure 11: Quadrant-specific waypoint missions for payload deployment.

- **Manual Deployment:** Drone flies to the quadrant, lands, and payload is deployed manually.

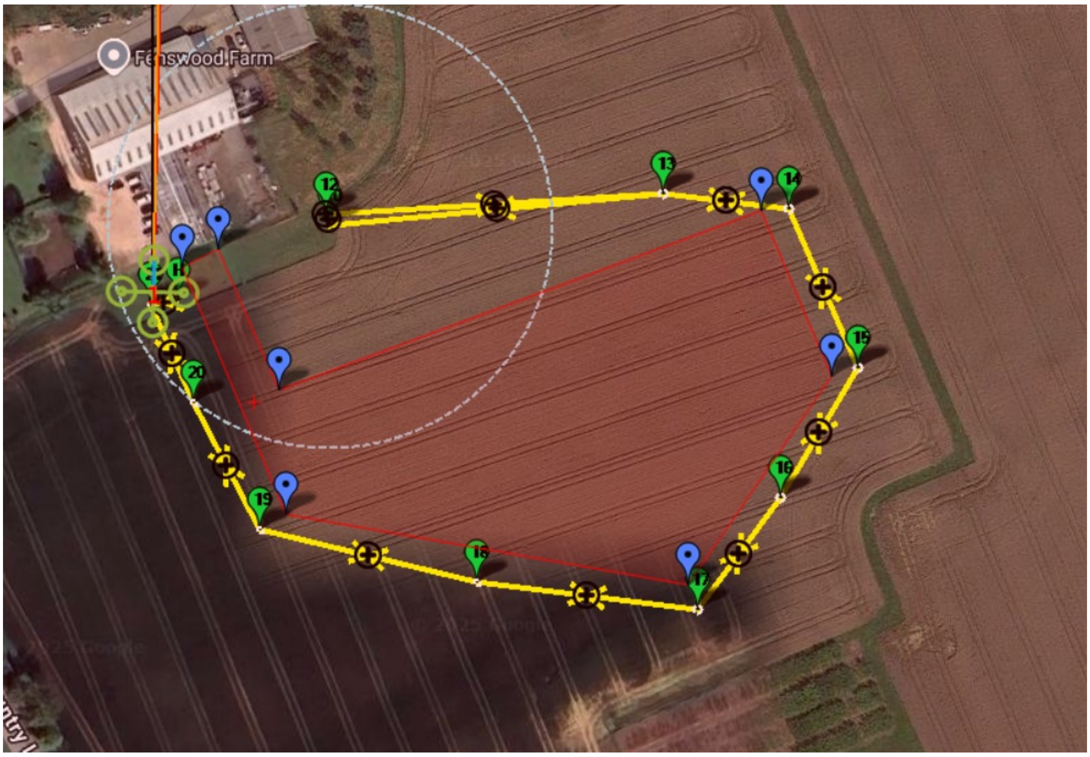


Figure 12: Final selected waypoint mission for payload deployment based on quadrant analysis. The target location lies within the 20-meter drop zone.

3.4 System Feature Comparison

Feature	Original System	Fallback System (Implemented)
Processing Location	Onboard Raspberry Pi	Offboard Ground Station Laptop
Real-Time Processing	Yes	No, post-flight analysis
Quadrant Determination	Immediate (real-time onboard)	Post-flight analysis
Mission Continuity	Seamless onboard decision making	Requires drone landing and data transmission
Reliability	Lower (hardware limitations)	Higher (stronger ground station capabilities)
Payload Deployment	Fully autonomous	Manually activated after analysis

Table 5: Comparison of Original and Fallback System Features

4 Requirements Verification

4.1 Verification of Functional and Safety Requirements

The autonomous aerial system was developed and tested to meet all stated operational and safety requirements. The following table outlines each requirement, its corresponding verification method, and current status:

Requirement	Verification Method	Status
Flight Area Compliance	Geofencing; telemetry data	Verified (Appendix E)
Altitude Restriction (50m max)	Firmware limit; telemetry monitoring	Verified (Appendix A)
Zebra Survey Accuracy	Image capture; YOLOv8n model detection	Verified (Figure 9 and Appendix F)
Payload Deployment Accuracy	Quadrant analysis; waypoint mission execution	Verified (Appendix B)
Safety Feature Implementation	Controlled flight testing and simulated emergency scenarios during Preflight Tests	Verified (Appendix C)

Table 6: Requirements Verification Summary

4.2 Technology Readiness Level (TRL) Assessment

The development and testing of the autonomous aerial system for wildlife survey and payload delivery were evaluated using the Standardised Technology Readiness Level (TRL) methodology. The project successfully reached **TRL 6**, with all system components demonstrated in a relevant environment and working in unison.

TRL	Level Name	Description
1	**Basic Principles Observed**	Fundamental ideas for wildlife monitoring using drones have been identified. Object detection and UAV viability were investigated theoretically.
2	**Technology Concept Formulated**	Concepts for vision-based object detection (YOLO), autonomous navigation, and payload delivery have been defined.
3	**Proof of Concept**	Three proof-of-concept models were developed and evaluated for YOLO detection using sample photos and pre-trained models. Drone simulations or early field testing with cutouts are accomplished.
4	**Lab Validation of Key Components**	YOLOv8n was trained using a proprietary dataset. Drone hardware (Hexsoon EDU-450, Raspberry Pi, payload mechanism) is constructed and configured.
5	**Validation in Relevant Environment**	The system was tested in field-like conditions (Fenswood Farm), with all key components incorporated (detection, waypoint navigation, and manual payload).
6	**System Shown in Relevant Environment**	A fully integrated system (UAV + object detection + payload delivery logic) is presented in a controlled yet realistic environment with test animals.
7	**Prototype for Operational Use**	Real-time autonomous decision-making, onboard processing, and automated payload distribution in conservation zones.
8	**System Qualified by Testing**	Extended operational testing under a variety of climatic conditions, with full autonomy, dependability, and robustness confirmed.
9	**Operational System Proven**	Successfully implemented in a real-world conservation program with minimal human intervention.

Table 7: Technology Readiness Level (TRL) Assessment

TRL 6: System Demonstrated in a Relevant Environment

The whole system, including the Hexsoon EDU-450 drone, Cube Orange flight controller, GPS navigation, YOLOv8n-based object detection, waypoint-based navigation, and quadrant-based payload delivery, was integrated and tested at the University of Bristol's Fenswood Farm.

The system met all its functional objectives: image collection, offboard animal detection, and location-specific payload delivery near the target (simulated elephant cut-out). While onboard processing could not be achieved due to technical limitations, the fallback architecture using a ground station was successfully implemented. Safety mechanisms such as geofencing, return-to-home (RTH), and altitude restriction were consistently validated under field conditions.

5 Evaluation

5.1 Technical Performance Evaluation

Flight Performance
The Hexsoon EDU-450 drone with Cube Orange flight controller displayed exceptional flying stability, accurate waypoint navigation, and dependable autonomous behaviour. All operational aircraft successfully followed geofencing limits, altitude constraints (maximum 50 meters), and maintained stable GPS connectivity.

Accurate payload delivery within a 20-meter radius was achieved using a quadrant-based approach. Manually activating the Tarot payload mechanism after landing guaranteed dependability despite increased operating complexity.

Object Detection and Processing
The YOLOv8n model reliably and accurately identified zebras and elephants in aerial images. Despite offboard processing, post-flight picture analysis enabled exact quadrant identification and informed payload deployment decisions.

5.2 Plus/Delta Analysis

Plus (Strengths)	Delta (Areas of Improvement)
Reliable autonomous waypoint navigation.	**Due to technical issues, onboard processing could not be completed.**
Effective fallback system (off-board processing).	**Real-time detection capabilities are limited owing to offboard data processing.**
High detection accuracy with the YOLOv8n model.	**The payload release mechanism had to be initiated manually; there was no autonomous payload actuation.**
Robust safety features (RTH, geofencing, altitude limitations).	**The reliance on manual interventions increased operational complexity and mission duration.**
Good flexibility to hardware integration difficulties.	**Integration and processing restrictions with Raspberry Pi slowed real-time autonomy.**

Table 8: Plus/Delta Evaluation

5.3 Recommendations for Future Work

Based on the project outcomes and identified areas for improvement, the following recommendations are made for future iterations:

1. **Automated Payload Mechanism:** Create an automated payload delivery mechanism triggered by real-time detections.

2. **Real-Time Autonomous Decision Making:** Optimize onboard software for real-time object identification, analysis, and mission planning, reducing dependency on ground-station post-processing.

3. **Extend Field Testing:** Enhance operational robustness by expanding test scenarios under different environmental circumstances, such as weather and terrain.

4. **Integration with Toroidal Propellers:** For operations involving sensitive animals, such as elephants or zebras, reducing noise and rotor turbulence is essential. Future aerial system versions may include toroidal propellers, which have been found to minimise acoustic signatures and enhance aerodynamic efficiency due to their closed-loop blade design.

   These propellers might enable quieter, smoother operations, allowing drones to approach animals more discreetly while also enhancing detection quality and payload distribution precision. This is especially useful in conservation settings where non-invasiveness is essential [5].

References

[1] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," 06 2015.

[2] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," *arXiv:2207.02696 [cs]*, 07 2022.

[3] G. Jocher, "ultralytics/yolov5," 08 2020.

[4] G. Jocher, A. Chaurasia, and J. Qiu, "Yolov8 by ultralytics," 01 2023.

[5] W. Wei, Y. Ma, S. Wei, D. Wang, M. Guo, and Q. Yan, "Analysis and evaluation of aerodynamic noise characteristics of toroidal propeller," *Drones*, vol. 8, pp. 753–753, 12 2024.

[6] "Getting started with image classification — label studio," 2022.

[7] "A friendly guide to labelimg [+open datasets, models, alternative tools]."

[8] "Humansignal/label-studio," 10 2023.

[9] R. P. Ltd, "Raspberry pi connect beta - access your raspberry pi from anywhere – raspberry pi," 2025.

Appendices

A Waypoint Altitude Compliance

	Command	P1	P2	P3	P4	Lat	Lon	Alt	Frame	Delete	Grad %	Angle	Dist	AZ
1	TAKEOFF	0	0	0	0	51.423406	-2.671561	10	Relative	X	132.6	53.0	12.5	261
2	WAYPOINT	0	0	0	0	51.423169	-2.671393	25	Relative	X	52.1	27.5	32.5	156
3	WAYPOINT	0	0	0	0	51.422861	-2.671139	25	Relative	X	0.0	0.0	38.5	153
4	WAYPOINT	0	0	0	0	51.422734	-2.670292	25	Relative	X	0.0	0.0	60.4	104
5	WAYPOINT	0	0	0	0	51.422666	-2.669438	25	Relative	X	0.0	0.0	59.7	97
6	WAYPOINT	0	0	0	0	51.422938	-2.669116	25	Relative	X	0.0	0.0	37.6	36
7	WAYPOINT	0	0	0	0	51.423254	-2.668816	25	Relative	X	0.0	0.0	40.8	31
8	WAYPOINT	0	0	0	0	51.423639	-2.669082	25	Relative	X	0.0	0.0	46.6	337
9	WAYPOINT	0	0	0	0	51.423675	-2.669568	25	Relative	X	0.0	0.0	33.9	277
10	LOITER_TIME	10	0	0	0	51.423599	-2.670862	25	Relative	X	0.0	0.0	90.1	265
11	LOITER_TIME	10	0	0	0	51.423553	-2.670839	25	Relative	X	0.0	0.0	5.4	163
12	LOITER_TIME	10	0	0	0	51.423623	-2.670878	25	Relative	X	0.0	0.0	8.2	341
13	LOITER_TIME	10	0	0	0	51.423548	-2.671165	25	Relative	X	0.0	0.0	21.6	247
14	LOITER_TIME	10	0	0	0	51.423482	-2.671122	25	Relative	X	0.0	0.0	7.9	158
15	LOITER_TIME	10	0	0	0	51.4235501	-2.6710034	20	Relative	X	-44.7	-24.1	12.2	47
16	LOITER_TIME	5	0	0	0	51.4235501	-2.6710034	35	Relative	X	-	90.0	15.0	180
17	WAYPOINT	0	0	0	0	51.423675	-2.669568	25	Relative	X	-10.0	-5.7	101.0	82
18	WAYPOINT	0	0	0	0	51.423639	-2.669082	25	Relative	X	0.0	0.0	33.9	97
19	WAYPOINT	0	0	0	0	51.423254	-2.668816	25	Relative	X	0.0	0.0	46.6	157
20	WAYPOINT	0	0	0	0	51.422938	-2.669116	25	Relative	X	0.0	0.0	40.8	211
21	WAYPOINT	0	0	0	0	51.422666	-2.669438	25	Relative	X	0.0	0.0	37.6	216
22	WAYPOINT	0	0	0	0	51.422734	-2.670292	25	Relative	X	0.0	0.0	59.7	277
23	WAYPOINT	0	0	0	0	51.422861	-2.671139	25	Relative	X	0.0	0.0	60.4	284
24	WAYPOINT	0	0	0	0	51.423169	-2.671393	25	Relative	X	0.0	0.0	38.5	333
25	RETURN_TO_LAUNCH	0	0	0	0	0	0	10	Relative	X	0	0	0	0

Figure 13: Waypoint parameters confirming all flight altitudes remain within the 50-meter legal limit.

B Flight Log Snapshots

(a) Flight Log 1: Successful area survey.

(b) Flight Log 2: Payload drop after quadrant detection.

Figure 14: Flight mission records showing (a) survey phase and (b) targeted payload delivery.

C RTL Simulation in Mission Planner

Figure 15: RTL simulated in Mission Planner.

D LabelImg Annotation Tool

Figure 16: LabelImg interface used for manual annotation of zebras, rhinos, and elephants in aerial images. Annotations were saved in YOLO format to train the object detection model.

E Geofencing and Sensitive Area Compliance

Figure 17: Sensitive zone from the brief (left) and corresponding geofenced mission setup in Mission Planner (right).

F YOLOv8n Inference with Quadrant Mapping

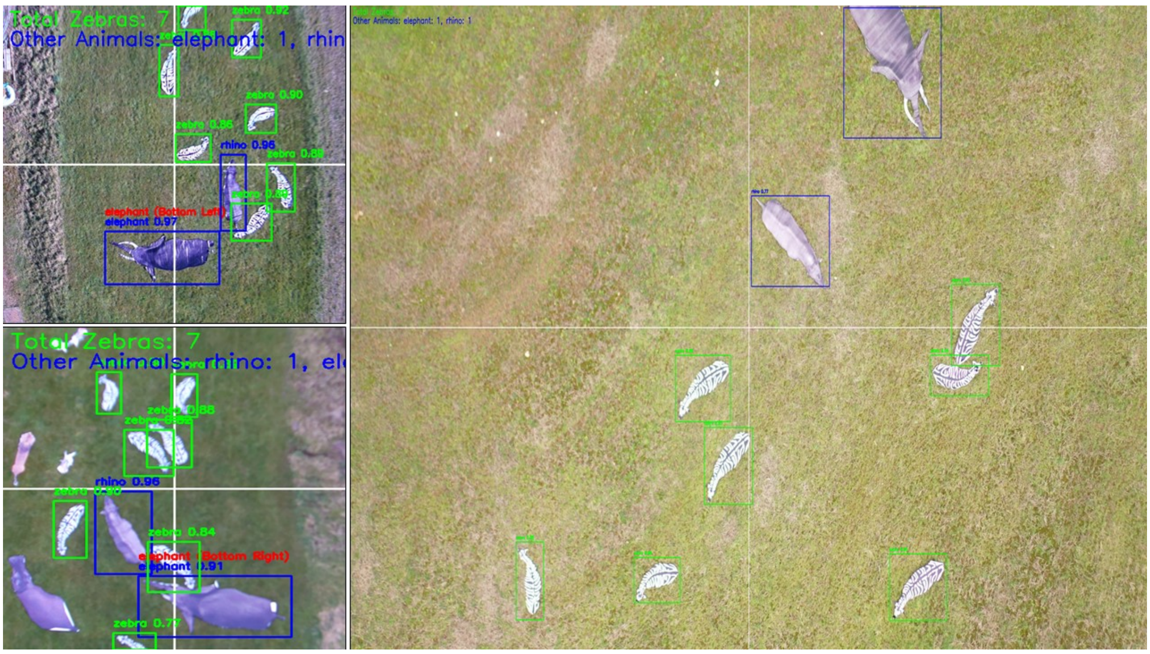


Figure 18: Inference results from the trained YOLOv8n model showing accurate detection of zebras, elephants, and rhinos with class labels and confidence scores. Grid overlay was used to map animals to corresponding quadrants for payload planning.

G GitHub Repository

The full project codebase and documentation are available at:

`https://github.com/ThejasThomson10/AeroZeno`

This repository includes:

- YOLOv8 training scripts
- Offboard image processing pipeline
- Mission Planner '.waypoints' files