



Final Year Project Report

Design and Development of a Power Transmission Line Inspection Robot

Ismail Mamadayev, Ilyas Abutalip

A thesis submitted in part fulfilment of the degree of

BSc in Robotics and Mechatronics

Supervisor: Ahmad Alhassan

Department of Robotics and Mechatronics

Nazarbayev University

May 4, 2026

Table of Contents

Abstract	3
1 Introduction	5
1.1 Research Problem	5
1.2 Project Aims and Scope	5
1.3 Contributions	6
1.4 Report Structure	6
2 Related Works	7
3 Methodology	9
3.1 Agile Project Phases	9
3.2 Structural Optimisation	9
3.3 Control Architecture	10
3.4 Risk Management	10
4 System Design and Integration	11
4.1 Mechanical Design Evolution	11
4.2 Electronic Subsystem	14
4.3 Vision-Based Inspection System	15
5 Experiments and Results	20
5.1 Walking Subsystem — Mechanical Validation	20
5.2 Vision Pipeline — Line Detection	20
5.3 Vision Pipeline — Deep Learning Defect Detection	21
5.4 HSV Tracking and Telemetry	21
5.5 IMU Gyroscope Validation	22
6 Discussion	25
6.1 Mechanical Design	25
6.2 Vision Pipeline	25
6.3 System Integration	26

7	Conclusions and Future Work	27
7.1	Conclusions	27
7.2	Recommendation list	27
8	Learning Portfolio	29
8.1	Technical Knowledge Acquired	29
8.2	Professional Skills Development	29
8.3	Semester Progression	30
9	Ethics Portfolio	31
9.1	System Safety	31
9.2	Data Privacy and Intellectual Property	31
9.3	Societal and Environmental Impact	31
9.4	Broader Responsibilities and Transparency	32

Abstract

Manual crews or helicopters inspecting overhead power transmission lines are dangerous, expensive and inefficient. Existing robotic solutions such as UAVs and walking robots solve parts of the problem but are limited in endurance, payload or obstacle negotiation. This report outlines the design and development of a hybrid Power Transmission Line Inspection Robot (PTLIR) over two semesters of the Nazarbayev University Capstone program.

The first semester was devoted to the walking subsystem. Three mechanical iterations were realised, leading to a V-shape chassis with A2212 brushless motors, tested on laboratory conductor mock-ups. The second semester took the platform along two parallel tracks: mechanical refinement, which included the addition of a three point caster wheel support, a servo driven camera gimbal, and JGB37-520 geared DC motors (Iteration 4), and the development of an autonomous vision subsystem.

The vision pipeline is a composition of classical computer vision (Gaussian preprocessing, Canny edge detection, Probabilistic Hough Transform using OpenCV) for real-time line tracking and a YOLO11 deep-learning defect detector trained on the InsPLAD-det dataset. A custom To address the detection of small defects under resolution limits, a Manual Slicing inference strategy was developed, which achieved successful detection of defect classes on previously unseen images with a backbone precision of about 0.90. A Flask based MJPEG telemetry server supports remote supervision. Preliminary locomotion was confirmed on a metallic-tube test rig in April 2026. The results demonstrate the feasibility of a safe and inexpensive robotic inspection of high-voltage infrastructure.

Acknowledgments

The authors thank Supervisor Ahmad Alhassan for consistent guidance and weekly consultations throughout both semesters of the Capstone programme. The Department of Robotics and Mechatronics at Nazarbayev University is acknowledged for providing laboratory facilities and consumable resources essential to hardware prototyping and testing.

The open-source communities behind OpenCV, Ultralytics YOLO, and the InsPLAD dataset are gratefully acknowledged for making high-quality tools and data freely available to the research community.

Chapter 1: Introduction

Modern electrical infrastructure depends on overhead power transmission lines. But their inspection and maintenance is hazardous, resource intensive and technically challenging. Traditional methods involve manual crews climbing the towers or surveys by helicopter, both of which pose significant risks to workers, are high operational costs, and often do not deliver consistent or high-resolution data [1, 2]. These limitations have driven research into robotic platforms which can automate inspection tasks, thereby improving safety, reliability and efficiency. The long-term vision is not only inspection but also autonomous maintenance; current research such as hybrid locomotion designs [3] and mission-optimisation strategies [4] are focused on reliable data collection and endurance, but the final goal is a robotic platform that can also perform corrective actions on live infrastructure.

Research to date has focused on three major classes of robotic platforms. Flying robots (UAVs) are good at rapid deployment and flexible inspection, but they are limited by endurance, payload, and susceptibility to electromagnetic interference (EMI) near conductors [1]. Walking robots walk directly on the conductors, giving stability and allowing heavier sensors to be mounted, but have problems in passing over obstacles like clamps and spacers [2, 5]. Hybrid robots aim to exploit these advantages but introduce new challenges in mechanical design, control and safe aerial-to-walking transitions [6, 3].

1.1 Research Problem

Despite promising advances, existing robotic inspection solutions have not been widely deployed. UAVs have limitations in endurance and payload, while walking robots are limited by obstacles and limited mobility. Hybrid systems show promise, but are complex and under-explored in practice. The gap lies in developing a stable conductor locomotion and flexible deployment hybrid inspection platform that is feasible.

1.2 Project Aims and Scope

The objective of this project is to design, build and partially validate a Hybrid Power Transmission Line Inspection Robot (PTLIR) over two Capstone semesters. The project includes the design of a mechanical chassis, selection of a drive system, integration of electronics and development of an autonomous vision subsystem, which includes classical computer vision and deep-learning-based defect detection.

1.3 Contributions

This report makes the following principal contributions:

- A four-generation iterative mechanical design of a V-shape conductor-walking chassis with validated locomotion on laboratory mock-ups.
- A two-point caster-wheel stability enhancement.
- A dual-track vision pipeline combining classical Canny + Hough line tracking with YOLO11-based defect detection trained on InsPLAD-det.
- A custom Manual Slicing inference strategy that recovers recall on small defects without third-party tiling libraries.
- A Flask MJPEG telemetry server for remote supervision on resource-constrained hardware.

1.4 Report Structure

Chapter 2 reviews related work across UAV, walking, hybrid, and vision-based inspection systems. Chapter 3 describes the Agile methodology underpinning the project. Chapter 4 details system design and integration, covering mechanical iterations, electronics, and the full vision pipeline. Chapter 5 presents experimental results. Chapter 6 provides critical discussion. Chapter 7 states conclusions and future work. Chapters 8 and 9 contain the Learning Portfolio and Ethics Portfolio respectively.

Chapter 2: Related Works

Research on robotic systems for power transmission line inspection has been accelerated in the recent years, resulting into solutions that can be broadly classified in the following three categories: flying robots, hybrid robots and walking or snake-like robots. Each class offers unique strengths while facing distinct limitations. Current research trends focus on balancing agility, endurance, payload capacity and safety. In this chapter, we survey major contributions since 2021, both technical advances and persistent challenges.

Flying Robots (UAV-based Inspection)

Flying robots, in particular unmanned aerial vehicles (UAVs), are the most investigated approach for gearbox line inspection. Their agility and ease of deployment allow for rapid surveying of long spans, and onboard cameras capture high-resolution data for fault detection. Recently, deep learning techniques such as YOLOv5 and YOLOv7 have been employed for real-time recognition of components and defects, leading to a substantial improvement in detection accuracy compared to prior vision-based methods [7, 8]. UAVs have also been used for 3D reconstruction of transmission line corridors, providing better situational awareness and planning [9].

However, UAVs have some well-known limitations despite such advantages. Battery capacity limits endurance, and flight times are less than one hour in most cases. Payload limitations restrict the incorporation of heavier nondestructive testing (NDT) sensors, and susceptibility to electromagnetic interference in proximity to energised conductors poses safety concerns [1]. Therefore UAVs are good for quick visual inspection, but not for long term operation or repair work.

Hybrid Robots

Hybrid robots attempt to overcome UAV limitations by employing multiple locomotion modes, most commonly aerial access combined with perching or conductor traversing modes. Autonomous landing and perching strategies have been demonstrated, allowing UAVs to hop onto conductors to save energy and do long-term inspection [6]. Optimisation frameworks like multi-objective energy minimisation [4] have been used to extend mission duration and improve operational efficiency.

Recent hardware innovation is the over-actuated tilt-rotor UAVs that can fly and travel along conductors with constrained motion, to perform localised inspection without repeated takeoff and landing [10]. Methods for 3D reconstruction based on neural radiance fields (NeRF) have also been explored for aiding planning and autonomous decision-making [9]. These advancements open up hybrid platforms as promising candidates for long duration inspection and repair applications. However, there are still important challenges to be overcome in pro-

viding robust transition from flight to walking, increased mechanical and control complexity, and safe interaction with live high-voltage infrastructure.

Walking and Snake-like Robots

Walking and snake-like robots take a different approach, where the focus is on stability and payload capacity rather than agility. They can directly traverse conductors to carry heavier sensing equipment and perform close-range inspections including nondestructive testing. For example, recent works have presented detailed on-conductor inspection capabilities, and energy recovery methods during downhill locomotion that enhance endurance by regenerating power while descending inclined conductors [5].

The high flexibility of snake-like robots has been useful in getting through clamps, spacers and other obstacles that are difficult for traditional walking systems. However, their slower deployment makes them less practical for large scale or time-critical surveys. These robots are well suited for localised, detailed work where precision and payload capacity are more important than speed.

Vision-Based Defect Detection

Reliable line localisation on resource constrained embedded hardware has been demonstrated using classical pipelines based on Canny edge detection and the Probabilistic Hough Transform [11]. YOLO-family architectures are state-of-the-art on inspection benchmarks for defect detection. The InsPLAD dataset contains 10,607 high-resolution UAV images of 17 different power-line assets from an active 500,kV line in Brazil [12]. Strong precision results have been obtained with YOLO11 for insulator defect detection [13, 14], and image tiling (SAHI-style) has been proposed to overcome recall limitations for small defects [15].

A common approach for illumination-invariant robotic object localisation consists of HSV colour-space segmentation followed by filtering on contour area and aspect ratio [16, 17]. The widespread adoption of Flask-based MJPEG streaming over HTTP has led to its popularity for lightweight, browser-compatible video transmission from Raspberry Pi platforms [18].

Chapter 3: Methodology

This project applies an Agile development methodology, emphasising iterative design, incremental progress and flexibility to changing requirements. Agile is especially appropriate for hybrid robotic systems because of (1) the complexity of integrating different subsystems and (2) the need to adapt requirements as designs are validated by CAD modelling, simulation and prototyping. This process is performed in cyclic sprints, where the feedback from CAD evaluation, simulation, lab tests, and supervisor meetings is integrated into each subsequent iteration.

3.1 Agile Project Phases

Six Agile phases structured the two-semester effort, moving from initial research to integrated hardware validation.

Phase 1	Literature Review & Requirements (Sept. 2025). Review of UAV/walking systems; re-scoping from path-planning to a hardware-based hybrid robot.
Phase 2	Initial Subsystem Prototyping (Oct.–Nov. 2025). Three mechanical iterations in SolidWorks; 3D-printed prototypes with A2212 brushless motors.
Phase 3	Validation & Constraint Mapping (Nov.–Dec. 2025). Mock-up testing; identification of lateral instability and limited FOV as primary mechanical limitations.
Phase 4	Parallel Software Development (Jan.–Feb. 2026). Development of synthetic image generation and the classical Canny + Hough line-tracking pipeline.
Phase 5	Deep Learning & Perception (Feb.–Mar. 2026). YOLO11 training on InsPLAD-det; implementation of Manual Slicing strategy and Flask MJPEG telemetry server.
Phase 6	Mechanical Refinement & Integration (Apr.–May 2026). Assembly of Iteration 4 (widened chassis, JGB37-520 motors, caster wheels); final locomotion validation on metallic-tube rigs.

3.2 Structural Optimisation

The walking subsystem has been modelled in SolidWorks with care for structural stability, weight minimisation and obstacle crossing ability. Critical joints and links will be analysed by Finite Element Analysis (FEA). The optimisation strategy is a compromise between the

stiffness under the expected loads, the minimum weight for energy efficiency and the compatibility with the vision-subsystem payload.

3.3 Control Architecture

Locomotion control is implemented in Python on the Raspberry Pi 4B. An L298N H-bridge converts GPIO PWM signals to motor drive commands. A horizontal error signal

$$\varepsilon = c_{\text{centre}} - c_x \quad (3.1)$$

derived from the vision pipeline drives a dead-zone steering controller, where c_{centre} is the image centreline and c_x is the detected object centroid. A dead zone spanning pixels 540–740 in a 1280-pixel-wide frame suppresses high-frequency jitter during stable traversal.

3.4 Risk Management

The expected risks and mitigations are: structural failure (progressive load testing, material upgrades), control instability (incremental simulation-based tuning, tethered tests), component failure (spare motors and sensors), and power instability (LM2596 Buck Converter to prevent Raspberry Pi brownouts during high-torque events).

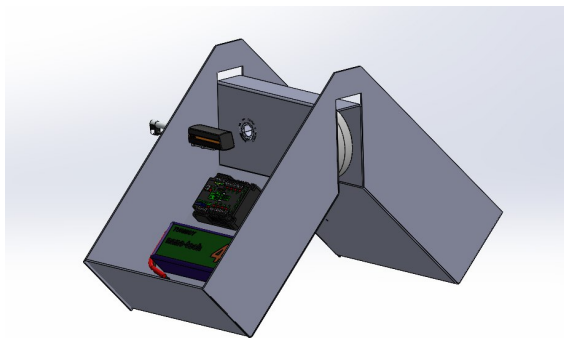
Chapter 4: System Design and Integration

4.1 Mechanical Design Evolution

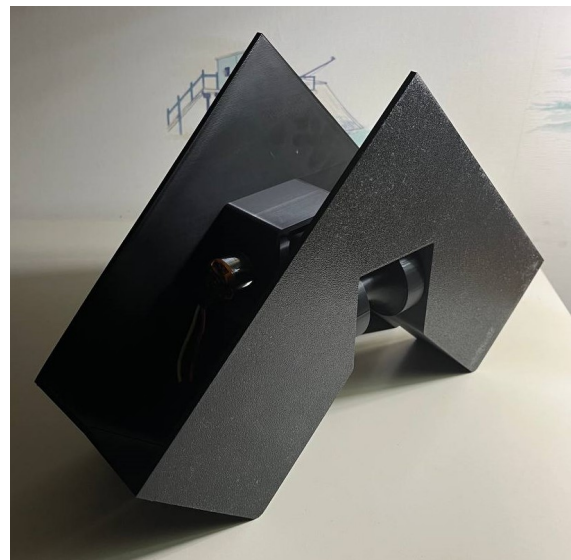
The PTLIR mechanical design went through four iterations across two semesters. Each iteration was informed by quantitative testing, CAD analysis, and feedback from the supervisor. Below, the sections detail each phase chronologically, illustrated by CAD renders and lab photos.

4.1.1 Iteration 1: Initial Concept (ROBT 491)

The first iteration in SolidWorks defined the general V-shape chassis geometry, and the basic design principal of two angled arms surrounding a conductor with a drive wheel at each apex. The earliest CAD models (Figure 4.1) have triangular side panels, a single central motor bay and rudimentary motor-shaft apertures. The design was intentionally minimal to validate the geometric concept before physical fabrication. By this time the drive units had been specified as brushless motors A2212 due to their availability and high power to weight ratio.



(a) CAD Model.



(b) 3D Printed Model.

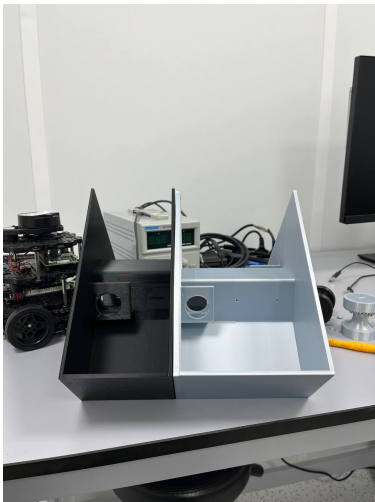
Figure 4.1: Iteration 1 SolidWorks CAD models establishing the V-shape chassis concept. The design defines the conductor-straddling geometry and initial motor-bay layout that all subsequent iterations refine.



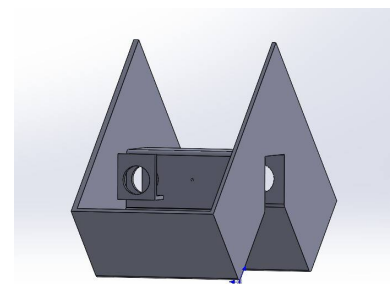
Figure 4.2: The A2212 brushless DC motor.

4.1.2 Iteration 2: Dual Motor Bays and Component Layout (ROBT 491)

Iteration 2 improved the chassis architecture with two independent motor housings and a wider central bridge with explicit shaft pass-through cutouts. The overall footprint of the chassis was also increased considerably to allow for a rear caster wheel, ensuring better static and dynamic balance of the robot. The structural envelope is extended to provide the required stability of the assembly and a sleek internal volume for future component integration.



(a) Comparison of 3D Printed Models of iteration 1 and iteration 2.



(b) CAD Model.

Figure 4.3: Iteration 2 CAD Model.

4.1.3 Iterations 3 and 3.5: Physical Assembly and Traction Optimization

Iterations 3 and 3.5 were focused on the physical assembly and testing of the 3D-printed PLA parts. The main goal of this phase was to increase the friction between the motor and the wheel. To overcome this, a new wheel design was developed that provided a dedicated connector for the motor shaft for a more secure mechanical interface. However, testing revealed that the motor lacked the power to effectively drive the entire assembly in this configuration.

Figure 4.4 shows a side-by-side comparison of the iterative refinement process. In particular, the motor-bay aperture alignment and arm-angle consistency improved between successive prints. A2212 brushless motors were mounted and the robot was tested on a laboratory conductor mock-up. Despite the mechanical improvements, two persistent limitations were identified through testing: (1) lateral tilt under disturbances due to the two-wheel configuration, and (2) a constrained forward-only camera field of view insufficient for comprehensive defect inspection.

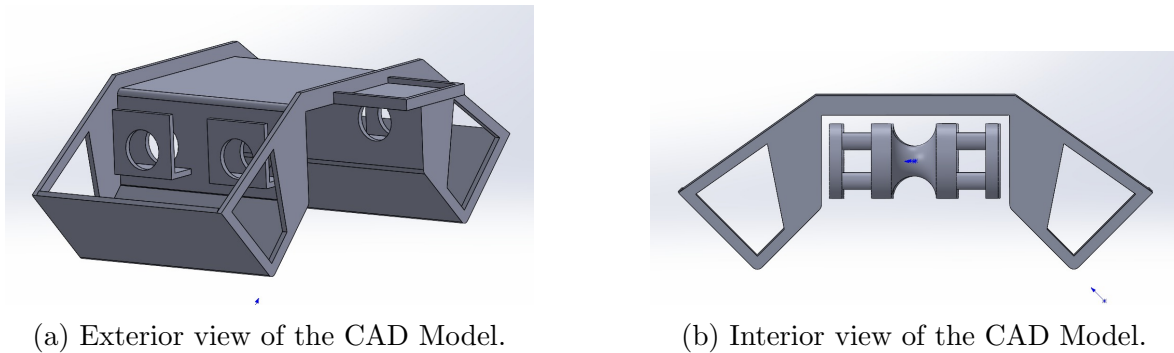


Figure 4.4: Iteration 3 CAD Model.

4.1.4 Iteration 4: Enhanced Stability and Bidirectional Inspection (ROBT 492)

Iteration 4 will have major mechanical upgrades to the locomotion assembly to resolve traction and stability issues found in previous tests.

Redesigned Drive Wheels and Chassis. The drive wheels were completely re-designed to improve grip of the conductor. The new wheel geometry has a profile optimised for more surface contact (traction) with the gearbox line. The main chassis was widened to accommodate these larger high-traction wheels. This expansion gives the required clearance for the new drive system and also reduces the center of gravity, which improves the overall stability of the robot during movement.

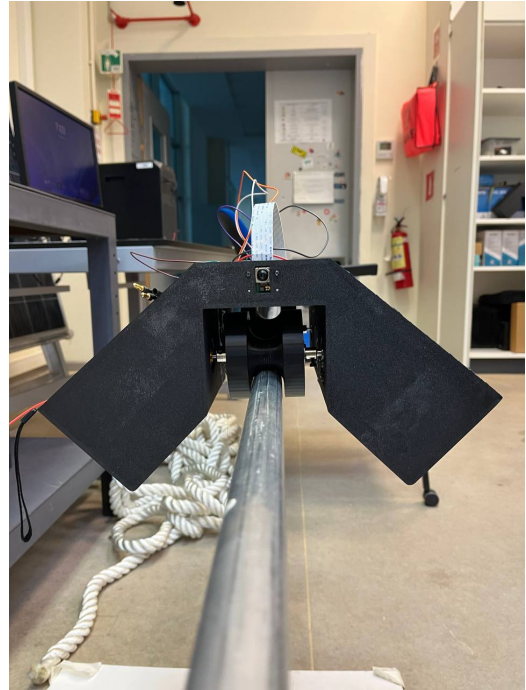
Motor Hub Integration. A major addition in this iteration is the implementation of custom motor hubs. These hubs are a tight and concentric fit between the JGB37-520 motor shafts and the wheels. The hubs eliminate the slippage and misalignment that typically occurs with simple press-fit or D-shaft couplings, ensuring that torque is transmitted consistently and preventing mechanical failure under the high-load conditions encountered during slope traversal.

Drive System Revision. The primary drive actuators were changed from A2212 brushless units to JGB37-520 geared DC motors to provide the higher low-RPM torque needed for slope traversal and conductor gripping. The A2212 motors of previous versions were used as passive guide rollers on the non-driven side, enabling symmetric contact force without active drive.

Figure 4.1.4 shows Iteration 4 assembled and mounted on the metallic-tube test rig used for preliminary locomotion validation in April 2026.



(a) Side view.



(b) Front view.

Figure 4.5: Iteration 4 assembled prototype on the metallic-tube test rig.

4.2 Electronic Subsystem

The electronic architecture was re-designed to meet higher power requirements and autonomous sensing. Table 4.1 shows the updated bill of materials.

Vision experiments during early development used a Defender USB camera, until the arrival of the Raspberry Pi Camera Module 3. We assume the use of Pi Camera Module 3 as the primary sensor for both classical CV and YOLO-based detection in the final system design. The USB camera was only used for preliminary testing and is not part of the final platform.

Table 4.1: Updated Electronic Subsystem Components

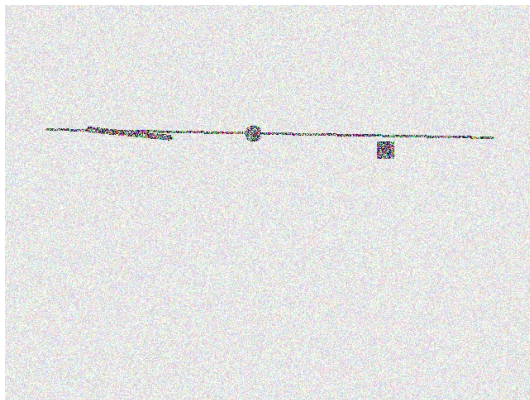
Component	Model	Primary Function
Microcontroller	Raspberry Pi 4B	Master control, image processing, telemetry
Camera	Pi Camera Module 3	Defect detection & line tracking
Drive Motors	JGB37-520 Geared DC	High-torque conductor traversal
Motor Driver	L298N Dual H-Bridge	PWM motor control
IMU	MPU-6050	Pitch/roll stabilisation
Power Regulation	LM2596 Buck Converter	12 V $\rightarrow$ 5 V step-down
User Interface	Logitech F710	2.4 GHz wireless override
Battery	LiPo Battery 3S	Power supply)

4.3 Vision-Based Inspection System

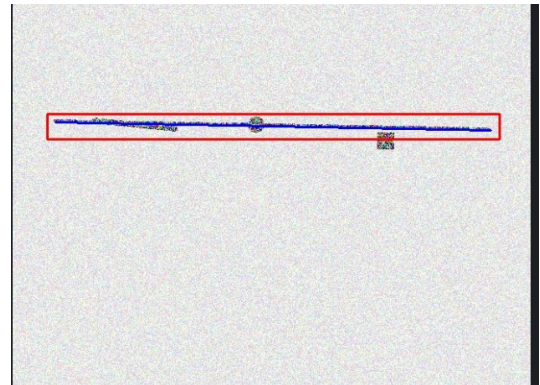
The vision subsystem runs on two complementary tracks: (a) classical computer vision, for real-time line tracking, and (b) deep learning, for defect identification. We created synthetic image datasets for algorithm development during hardware delivery delays.

4.3.1 Synthetic Data Generation

Python scripts generate realistic synthetic images for two purposes: (1) algorithm validation under controlled noise and lighting; (2) defect simulation, adding artificial anomalies (foreign objects, wire thickening, wire breaks) to evaluate detection feasibility. Figure 4.6 shows a synthetic pair of images. Left image shows a clean conductor with Gaussian noise added at a level corresponding to the properties of the Pi Camera Module 3 sensor. The right image introduces two intentional anomalies: a circular blob simulating a foreign object and a rectangular patch simulating a section of missing or damaged hardware. Both images are shown with the same noise profile to be able to attribute the defect signal to the anomaly and not to the noise differences.



(a) Clean synthetic conductor image with Gaussian noise applied to simulate real sensor characteristics.



(b) Same synthetic image with two simulated defects: a circular foreign-object anomaly and a rectangular missing-hardware patch.

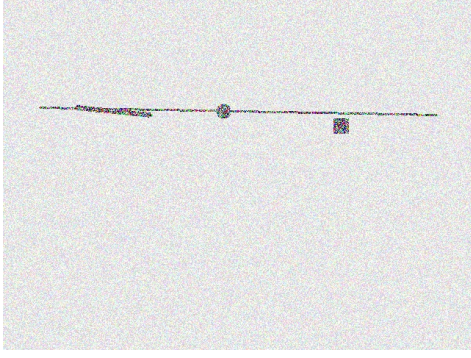
Figure 4.6: Synthetic images generated in Python for algorithm development and defect-simulation experiments.

4.3.2 Classical CV Pipeline for Line Tracking

The pipeline, implemented in OpenCV and optimised for the Raspberry Pi 4B, is composed of three sequential stages:

1. **Preprocessing:** Greyscale conversion and Gaussian blurring for reducing high-frequency noise and maintaining edge structure.
2. **Edge Detection:** The Canny algorithm calculates image gradients and applies double-threshold hysteresis. Canny is commonly used for power-line detection due to its high accuracy in complicated background environments [11].
3. **Line Identification:** Line segments are detected in the Canny edge map using the Probabilistic Hough Transform. The probabilistic version reduces the computational cost while maintaining the accuracy for fragmented line detection [11].

The pipeline was validated on synthetic images and real power line photos. The complete validation outputs are shown in Figure 4.7. On the synthetic dataset (Figures 4.7a and 4.7b) the Hough Transform successfully detected both conductor lines, with the red overlay tracing each identified segment along the full span of the wire. On a real UAV photograph acquired over a foggy rural corridor (Fig. 4.7c), the pipeline could extract the main conductors despite heavy clutter from trees, road markings and atmospheric haze (Fig. 4.7d). The Hough segments extend from the vanishing point at the tower, preserving the perspective geometry of the scene. Some spurious segments are found near the field edges, but the dominant conductor lines are clearly recovered, confirming the suitability for real-time conductor tracking on the Raspberry Pi 4B.



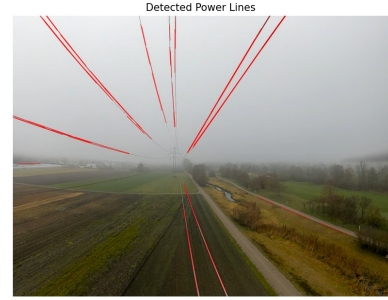
(a) Synthetic input image: single conductor with Gaussian noise.



(b) Hough output on synthetic image: both conductor lines correctly identified (red overlay).



(c) Real UAV photograph: foggy rural corridor with transmission tower and multiple conductors.



(d) Hough line segments on the real image. Main conductors are recovered despite background clutter from trees, road, and mist.

Figure 4.7: Classical computer vision pipeline validation results. Top row: synthetic dataset. Bottom row: real power-line photograph. Red overlays show Probabilistic Hough Transform line segments.

4.3.3 HSV Colour Segmentation for Object Tracking

Raw BGR frames are converted to HSV colour space to decouple chrominance from luminance, which allows for illumination-invariant segmentation [16, 17]. A blue cylinder, a laboratory proxy, was used to calibrate the system initially, and the thresholds were tuned to a specific range of hues. After deployment on Raspberry Pi 4B, the values were adjusted to target bright, achromatic surfaces, representative of the rope conductor mock-up, by collapsing the saturation channel to zero and raising the value floor:

$$L_{\text{HSV}} = [0, 0, 157], \quad (4.1)$$

$$U_{\text{HSV}} = [179, 0, 255]. \quad (4.2)$$

These bounds accept any hue ($H \in [0, 179]$) and fully desaturated pixels ($S = 0$) with brightness above 157, effectively targeting white and grey high-value surfaces. Two geometric constraints then filter false positives:

- **Area filter:** Contours with pixel area $A < 1000$ are discarded (Suzuki–Abe algorithm).
- **Largest-contour selection:** The largest remaining contour by area is selected and enclosed in a bounding rectangle for centroid computation.

An image based error signal with respect to the frame centre was calculated using the centroid extracted from the largest contour in initial stage experiments. This signal was used as an input to the dead-zone steering controller.

However, the evaluation (Section 5.4) revealed that the HSV-based segmentation is very sensitive to environmental conditions and produces pervasive false positives in realistic laboratory settings with multiple bright surfaces. Hence, HSV segmentation was not a suitable primary navigation signal.

In the final system architecture the classical Canny-Hough line detection pipeline is used as the main source of the steering error signal, while the HSV segmentation is kept only as an experimental or supplementary method under controlled conditions.

4.3.4 Wireless Telemetry Server

The processed frames are delivered to a remote browser interface by a Flask WSGI server through a `multipart/x-mixed-replace` MJPEG interface [18]. There is no client-side scripting involved and it is supported natively by all modern browsers. The streamed frame is overlayed on the detected centroid and bounding box, providing full situational awareness during inspection.

4.3.5 Deep Learning Defect Detection with YOLO11

To recognise fine-grained defects (cracked insulators, corrosion, broken caps, bird nests) more advanced than classical CV is necessary. The project employed the YOLO11 architecture [13, 14] trained with the InsPLAD-det dataset [12]. The dataset consists of 10,607 high-resolution UAV images of 17 different power-line assets captured from a live 500 kV transmission line in northeastern Brazil.

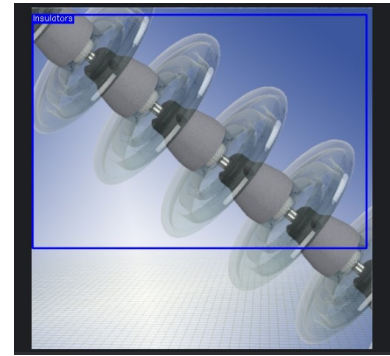
Baseline Inference with Pre-trained Model. Before local training, a detection baseline was built by utilising a pre-trained model from Roboflow. See figure 4.8 for some example outputs: the model correctly localises a tower structure and recognises an insulator string in two different types of images, showing that the general class vocabulary is recoverable even without domain-specific fine-tuning. These results prompted the following local training campaign on InsPLAD-det.

Training Progression. The training was performed in two steps. Phase 1 (5 epochs at 320 p resolution) achieved $mAP = 0.60$, precision ≈ 0.65 , and recall ≈ 0.55 , but generated false positives on high-clutter backgrounds. Phase 2 (10 more epochs) increased precision dramatically to 0.9044, showing high-confidence detections. Training at 320 p however caused small hardware components (nuts, washers, ceramic cracks) to be rendered as blurry blobs, capping recall – a phenomenon dubbed the *recall wall* in the small-object detection literature.

Manual Slicing Strategy. A custom Manual Slicing pipeline was implemented to overcome the resolution bottleneck, not using external tiling libraries (SAHI caused compatibility issues with the available Python and NumPy versions in the development environment). A single high-resolution image is divided into four overlapping quadrants and each quadrant is processed at an effective resolution of 640 p. This method conforms to the Slicing Aided Hyper Inference (SAHI) principle [15] and resulted in six verified defect detections on previously unseen images where global-view inference detected almost nothing. We achieve significant recall recovery while maintaining the 0.90 precision backbone.



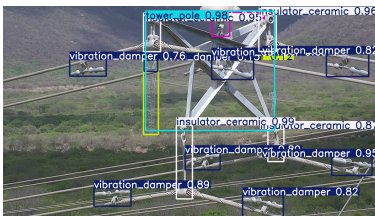
(a) Pre-trained baseline: **tower** class detected at high confidence on a real inspection photograph showing conductors and tower hardware.



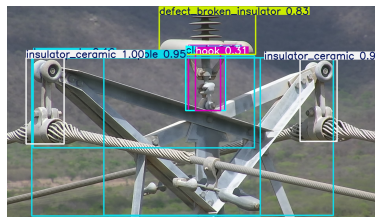
(b) Pre-trained baseline: **insulators** class detected on a rendered 3-D insulator model, confirming cross-domain class recognition.

Figure 4.8: Inference results from the Roboflow-hosted pre-trained model used as a detection baseline before local YOLO11 training. Detection boxes shown in blue.

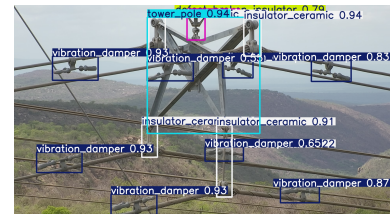
Figure 4.9 shows three example detections from the locally trained YOLO11 model tested on held-out InsPLAD-det images. In Figure 4.9a and Figure 4.9b, the model can always detect vibration dampers, ceramic insulators, and the tower pole with different viewing angles and conductor configurations, and most confidence scores are higher than 0.80. The most diagnostically significant result is in Figure 4.9c: a close-range view of a tower attachment point where the model flags a **defect_broken_insulator** at 0.83 confidence with high-confidence **insulator_ceramic** detections at 1.00 and 0.99. This exemplifies the generalisation power of the locally trained model, which ranges from simple component classification to the detection of defects on unseen imagery.



(a) Held-out image: vibration_damper (0.82–0.95), insulator_ceramic (0.87–0.99), and defect_broken_insulator (0.79).



(b) Second held-out image: High detection density near the tower cross-arm for insulators and dampers.



(c) Close-range attachment: confirmed defect_broken_insulator (0.83) alongside high-confidence insulators.

Figure 4.9: YOLO11 detections from the locally trained model (15 epochs, InsPLAD-det dataset) on three held-out UAV images. Colour coding: dark blue = `vibration_damper`; cyan = `insulator_ceramic`; magenta = `tower_pole`; yellow-green = `defect_broken_insulator`. Confidence scores are shown beside each label.

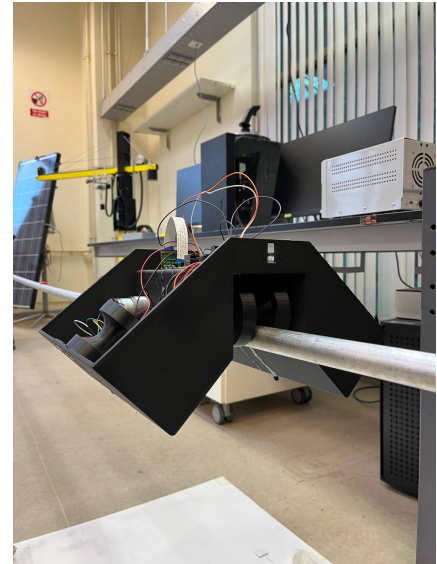
Chapter 5: Experiments and Results

5.1 Walking Subsystem — Mechanical Validation

The Iteration 3/3.5 prototype was tested on a laboratory scale conductor mock-up, first a metallic tube and then a 30 mm diameter rope to simulate the real conductor friction characteristics. The robot was able to traverse horizontal conductor sections successfully without detachment. The wheel geometry was V-shape with contact remaining under static loading. Lateral stability was confirmed as the primary limitation, motivating the Iteration 4 caster wheel. Preliminary iteration 4 locomotion on metallic-tube test rig confirmed motor torque sufficiency for JGB37-520.



(a) Validation setup for Iteration 3.5: Testing on a horizontal tube using a GameSir controller and DC power supply.



(b) Side view of Iteration 4 rig, validating JGB37-520 motor torque.

Figure 5.1: Laboratory validation of the walking subsystem iterations. The tests confirmed the geometric viability of the V-shape chassis and identified the need for enhanced lateral stability.

5.2 Vision Pipeline — Line Detection

The classical CV pipeline was validated on both synthetic and real power-line images. Both conductors were correctly detected on the synthetic data, as shown in Figure 4.7b. In a real photo with complex backgrounds (trees, sky, structures), the main conductors were successfully extracted despite the environmental clutter (Figure 4.7d), proving the suitability for real-time conductor tracking on Raspberry Pi 4B.

5.3 Vision Pipeline — Deep Learning Defect Detection

Table 5.1 displays the training progress and important results of YOLO11. The Manual Slicing approach found six defect detections on a high-resolution unseen image (two in the top-right quadrant and four in the bottom-right quadrant) while global-view inference found almost nothing.

Table 5.1: YOLO11 Training Metrics and Slicing Inference Results

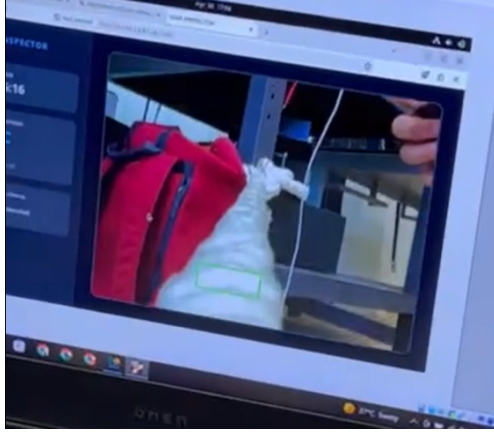
Phase	Epochs	mAP	Precision	Recall
Phase 1	5	0.60	≈ 0.65	≈ 0.55
Phase 2	15 (total)	—	0.9044	Limited (320 p)
Manual Slicing	15 (total)	—	0.90	6 detections

Phase 2 consisted of 10 additional epochs (15 total, including the initial 5 epochs of Phase 1).

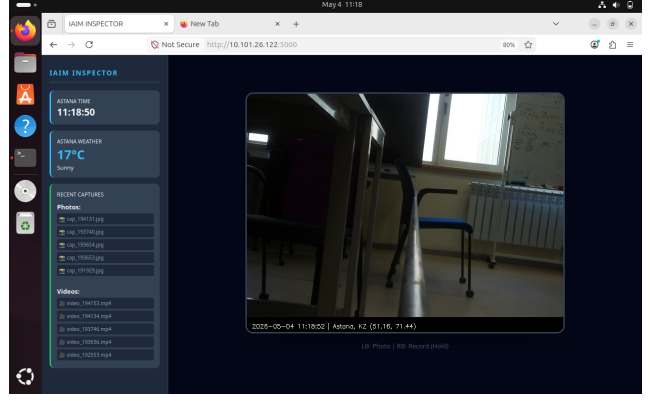
Figure 4.9 shows qualitative detection results on held-out InsPLAD-det images. The model identifies vibration dampers in all three test images with a confidence of ≥ 0.82 and ceramic insulators with a confidence of ≥ 0.87 . Importantly, the `defect_broken_insulator` class is verified at 0.79 (Figure 4.9a) and 0.83 (Figure 4.9c) showing that the model has learned discriminative features specific to defective vs. intact components and not just to asset class membership.

5.4 HSV Tracking and Telemetry

The HSV segmentation pipeline was validated in laboratory conditions and, subsequently, tested on the 30 mm rope conductor mock-up in late April 2026. In the laboratory the pipeline was able to detect a bright cylindrical target under controlled lighting conditions and the dead-zone controller was able to suppress oscillations during the stable tracking phase. However, rope testing revealed a core limitation of the revised HSV bounds: the thresholds are geared towards high-value, fully desaturated (white/grey) pixels, so any bright object in the scene — including the rope, the test-rig structure and ambient reflections — meets the segmentation criteria. The algorithm was only able to find the rope in near-ideal conditions (uniform dark background, no competing bright surfaces). In a typical laboratory environment with overhead fluorescent lighting and several reflective objects, false positives dominated the mask such that reliable centroid estimation was impossible. The Flask MJPEG server streamed annotated frames to a remote browser with acceptable latency for manual supervision.



(a) Tracking logic: Green bounding box identifies the conductor proxy.



(b) IAIM INSPECTOR dashboard: Live stream and capture repository.

Figure 5.2: PTLIR Inspection Interface: (a) illustrates the HSV tracking logic in the laboratory, while (b) shows the integrated web-based telemetry dashboard used for remote monitoring.

5.5 IMU Gyroscope Validation

5.5.1 Data Collection and Processing

Raw angular-velocity data were captured from the MPU-6050 inertial measurement unit mounted on the Iteration 4 chassis during a representative 10.64s traversal run on the metallic-tube test rig (April 2026). The IMU was sampled at about 9.8 Hz on the Raspberry Pi 4B I²C bus, and this resulted in 104 samples in three axes: X (pitch), Y (roll), and Z (yaw). All values are in units of $^{\circ}\text{s}^{-1}$.

Raw signals contain noise from high frequency vibration of the motor operation and chassis resonance. To recover the underlying attitude-rate trend, a moving-average filter with a window size of $w = 5$ samples was applied independently to each axis:

$$\hat{g}_k = \frac{1}{w} \sum_{i=0}^{w-1} g_{k-i}, \quad (5.1)$$

where g_k is the raw sample at index k and $\hat{g}_k$ is the filtered estimate.

5.5.2 Results

Figure 5.3 shows the raw and filtered signals for all three axes for the entire recording window. Table 5.2 summarises the descriptive statistics for each axis.

High-Fidelity Gyroscope Signal Analysis

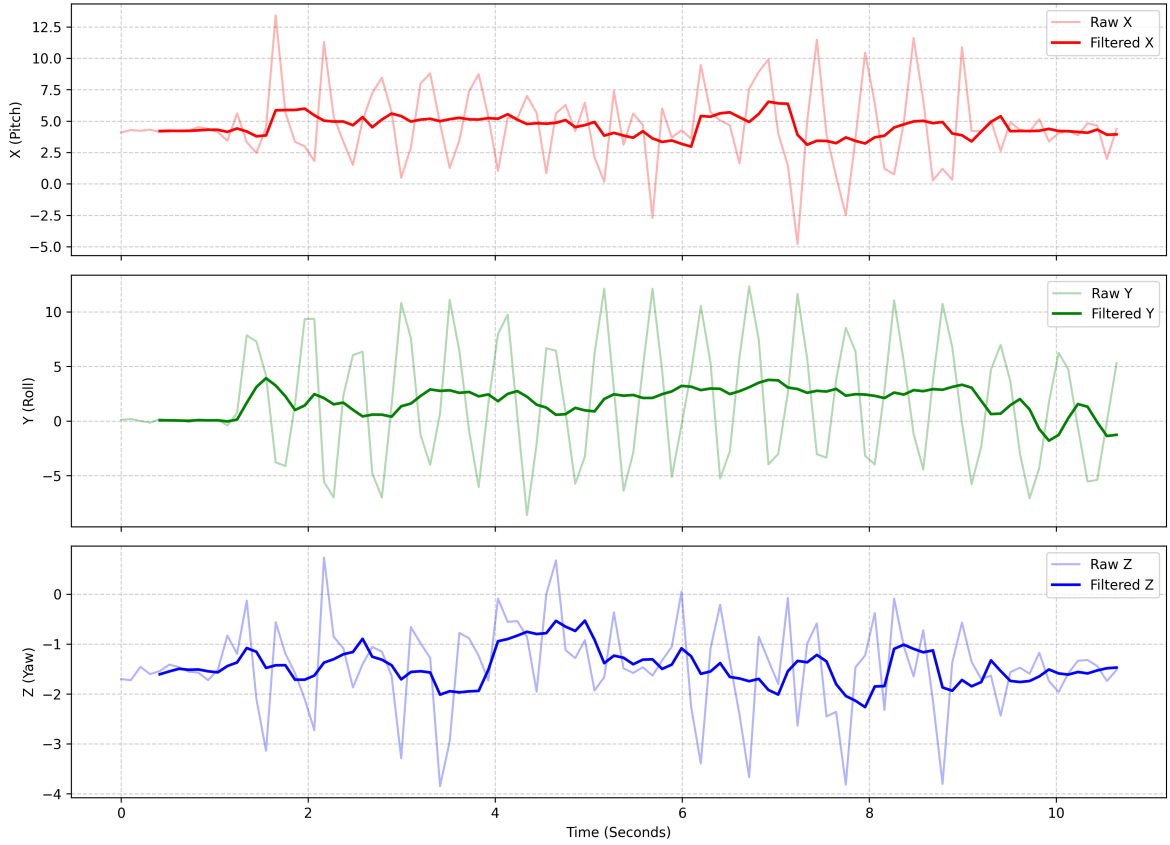


Figure 5.3: The filtered gyroscope signals provide only low-frequency trends due to the limited sampling rate and smoothing. While roll-rate oscillations are observable, accurate estimation of absolute tilt would require integration and sensor fusion, which is planned for future work. Data captured from the MPU-6050 IMU at ≈ 9.8 Hz on the Iteration 4 chassis (April 2026).

Table 5.2: MPU-6050 Gyroscope Descriptive Statistics (104 samples, 10.64 s)

Axis	Mean ($^{\circ} \text{s}^{-1}$)	Std Dev	Min	Max
X (Pitch)	4.56	3.00	-4.79	13.43
Y (Roll)	1.73	5.41	-8.63	12.34
Z (Yaw)	-1.46	0.88	-3.85	0.73

5.5.3 Discussion

After filtering (pitch), the X-axis signal has a mean value that remains relatively constant at $\approx 4.6^{\circ} \text{s}^{-1}$ during the whole test, which is a consistent positive pitch bias due to the small downward slope of the test rig. The largest variance is on the Y-axis (roll) ($\sigma = 5.41$), with oscillations above $\pm 10^{\circ} \text{s}^{-1}$ during the start-up of the motor (0–2 s). This behaviour is consistent with the lateral instability identified in the Iteration 3.5 testing; the three-point caster support introduced in Iteration 4 reduces but does not fully eliminate roll excitation under dynamic loading. The Z-axis (yaw) is bounded within $[-3.85, 0.73]^{\circ} \text{s}^{-1}$ with a small negative mean of -1.46 , which shows a slight leftward yaw drift during the traversal that is

within the correctable range of the dead-zone steering controller (Equation 3.1).

The moving-average filter (Equation 5.1) effectively attenuates high-frequency vibration while preserving attitude-rate trends that are critical for closed-loop control, showing promise as a lightweight on-device preprocessing step on the Raspberry Pi 4B. Future work will replace this filter with a Kalman estimator fusing IMU and wheel-encoder data for better state estimation during closed-loop traversal (see Section 7.2).

Chapter 6: Discussion

6.1 Mechanical Design

The progressive iteration strategy was effective. The V-shape chassis geometry was found to be mechanically sound and to provide an adequate contact force of the conductor across the tested tube and rope mock-ups. The key insight from the Iteration 3.5 testing — that a two-wheel contact configuration is inherently unstable for a passive line-traversal robot — straight to the three point caster wheel solution. By swapping out the A2212 brushless motors for JGB37-520 geared DC motors, the torque–endurance trade-off is resolved: brushless motors are faster, but have low low-RPM torque, whereas geared DC motors provide the steady gripping force needed to climb slopes and traverse obstacles. In Iteration 4, the A2212 units were maintained as passive guide rollers on the non-driven side, preserving their contribution to symmetric contact without the control overhead of a second driven axis.

One current limitation is that full Iteration 4 testing was constrained by consumable delivery timelines (late March–April 2026). FEA Structural analysis of the widened chassis with full payload is not yet complete, this is required prior to field deployment.

6.2 Vision Pipeline

The dual-track vision architecture is well-suited to the resource constraints of the Raspberry Pi 4B. The classical pipeline provides lightweight, reliable line-position estimates for the steering controller while YOLO11 runs as a slower asynchronous process for defect flagging — a sound separation of concerns. Figure 4.7 shows that the Hough Transform correctly identifies the prevailing conductor geometry on both synthetic and real images, but the presence of extraneous edge segments near the image borders suggests that further filtering (e.g., limiting the angular range to $\pm 30^\circ$ from the horizontal) would be beneficial for robustness in deployment.

The recall wall at 320p is a known challenge [15]; our in-house Manual Slicing solution achieves the same effect as SAHI without library dependency conflicts, demonstrating engineering adaptability. Figure 4.9 shows detection results on unseen data, confirming six class-level detections, and importantly, two `defect_broken_insulator` instances with operationally relevant confidence levels (≥ 0.79). The low-confidence `clhook` detection (0.31) in Figure 4.9c shows the trade-off between recall and false-positive rate: reducing the confidence threshold would recover more detections, but at the cost of precision. A confidence sweep on the validation set would determine the optimal operating point.

The HSV tracking pipeline performed well in controlled lab conditions but is expected to degrade outside with sky colour, reflective conductor surfaces and variable lighting overlapping with defined HSV bounds. The rope tests carried out in late April 2026 demonstrated this empirically, showing that the achromatic thresholds ($S = 0$, $V \geq 157$) selected for the bright rope mock-up were too permissive in a typical lab environment, producing many false posi-

tives in the presence of other bright surfaces. Reliable detection is needed under near-ideal isolation conditions, confirming that HSV segmentation alone is insufficient as a primary navigation signal for real deployment. Future work should consider an adaptive threshold calibration or suggest the Canny/Hough pipeline as the main navigation signal with HSV only as a secondary cue under constrained illumination.

6.3 System Integration

The remaining outstanding milestone is the physical integration of the full vision based control loop with the mechanical chassis. The Iteration 4 platform was mechanically assembled in mid April and preliminary locomotion was successfully demonstrated on the test rig, although testing was performed under manual control. The vision pipelines have been validated independently in software-in-the-loop configuration (classical CV and YOLO11). Complete closed-loop integration, where the vision system directly controls the locomotion in real-time, has not been achieved yet.

Chapter 7: Conclusions and Future Work

7.1 Conclusions

This report presented the design and development of a hybrid Power Transmission Line Inspection Robot over two Capstone semesters. Mechanical iterations led to a V-shape chassis with three-point caster support and a servo camera gimbal. Preliminary locomotion on a laboratory test rig confirmed the JGB37-520 motor torque was sufficient. The locally trained YOLO11 model was able to successfully detect defect-related classes on unseen images, including detection of the `defect_broken_insulator` class with confidence values above 0.79. These results demonstrate the model's capacity to generalise from simple component recognition to defect-level identification. A complete quantitative recall evaluation on a labelled test set is left as future work. Remote monitoring with minimal infrastructure using the Flask MJPEG telemetry server.

The project shows that a resource-constrained embedded platform (Raspberry Pi 4B) can achieve both real-time classical computer vision and asynchronous deep-learning inference, given that the computational pipeline is properly partitioned. The results offer a convincing proof-of-concept for safe and economical robotic inspection of energised high-voltage infrastructure.

7.2 Recommendation list

Kinematic and Dynamic Characterization: Provide a detailed performance analysis of the Iteration 4 architecture in terms of stochastic modelling of success rates and stability of locomotion over non-linear obstacle geometries.

Domain-Adaptive Perception Pipeline: Design a transfer learning method for YOLO11 to address covariate shift between standard datasets and images collected by the robot, allowing the model to acquire robust features under different field conditions.

Multi-Rate Sensor Fusion Architecture: Develop an Extended Kalman Filter (EKF) for high-frequency IMU and encoder-based odometry fusion for accurate state estimation and closed-loop control while traversing high-slip surfaces.

Illumination-Invariant Calibration Framework: Replace the heuristic-based HSV segmentation with an adaptive probabilistic perception model, able to perform real-time calibration under dynamic and extreme lighting conditions.

Heterogeneous System Orchestration: Design of a cooperative UAV-UGV subsystem for autonomous deployment with special focus on the synchronisation of aerial-to-walking handover protocols.

Sim-to-Real Transfer Validation: Validate handover tolerances and contact dynamics with high-fidelity physics engines: including aerodynamic disturbances and environmental

noise.

Electromagnetic Hardening and Reliability: Develop robust EMI mitigation techniques and redundant fail-safe state machines specifically rated for the extreme potential gradients of live high-voltage (HV) infrastructures.

Chapter 8: Learning Portfolio

8.1 Technical Knowledge Acquired

The project required rapid integration of mechanical design, embedded systems, and computer vision – three fields that had previously been studied in isolation. Iterative chassis development improved understanding of the weight–stiffness–geometry trade-offs in the constrained locomotion task, where excess weight directly hurts motor performance and battery endurance.

Deploying OpenCV on Raspberry Pi OS Bookworm required resolving environment-specific library conflicts. Moving to a full GUI environment was a hands-on lesson in Linux dependency management and cross-platform compatibility. The experience showed the difference between having the packages in the development environments versus the reality of deploying on embedded Linux.

The training journey for YOLO11 involved the full pipeline, from data curation and annotation, to hyper-parameter selection, training monitoring and inference optimisation. The experience with the recall wall phenomenon and the development of the Manual Slicing strategy illuminated the practical gap between benchmark performance and deployment reality, a challenge well-documented in the small object detection literature but not often faced in coursework. The qualitative outputs in Fig. 4.9 concretely demonstrate how the detection density and confidence distribution varies with scene content: tower cross-arms produce the highest density of vibration-damper detections, while the detection of defect class is triggered by close-range views of attachment hardware.

The HSV rope-tracking experiment delivered a concrete lesson in the distance between algorithm design and deployment reality: thresholds tuned in one environment can totally break down in another. Diagnosing the false-positive flood in the laboratory setting — and understanding why achromatic HSV bounds are inherently permissive in bright, cluttered scenes — deepened understanding of the conditions under which colour-based segmentation is and is not appropriate.

8.2 Professional Skills Development

Weekly team and supervisor meetings, GitHub version control, and agile sprint planning kept the project on schedule despite multiple weeks of consumable delivery delays that would have derailed a less structured workflow. The practice of committing to sprint goals and adjusting scope instead of dates was found useful in managing uncertainty.

Developed technical writing discipline, wrote two full interim reports and multiple deliverables reviewed by supervisor. Breaking down complex, multi-subsystem work into a cohesive narrative required learning to balance detail with readability—a skill directly translatable to industry and academic publication contexts.

Responsibility assignment – one member leading mechanical design, the other leading software development – with regular synchronisation meetings ensured subsystem compatibility without creating bottlenecks. This division of labour is reflected in industry team structures and required both members to develop enough cross-domain literacy to communicate effectively across subsystem boundaries.

8.3 Semester Progression

At the beginning of ROBT 492, the team had a validated mechanical baseline but no software capability. By the end of the semester, a full dual-track vision pipeline is implemented and validated, the mechanical design is significantly improved, and preliminary locomotion is confirmed. The development of a walking proof-of-concept into a sensor-driven inspection platform in one semester, despite hardware delays, shows a great deal of growth in individual technical competency and team execution capability.

Chapter 9: Ethics Portfolio

9.1 System Safety

All testing to date has been done in low-voltage, laboratory-scale environments, without energised conductors. Transition to field testing on live transmission lines requires a thorough risk assessment, EMI shielding qualification and fail-safe protocol design— tasks that are noted as future work. The choice of JGB37-520 geared motors was also driven by safety. The lower no-load speed decreases the risk of uncontrolled downhill acceleration and the higher torque increases the reliability of braking, decreasing the chance of the robot unexpectedly letting go of a conductor.

The current prototype is to be considered as only a research demonstrator. Before being deployed on live infrastructure, the system would need to be certified to the appropriate electrical safety standards (IEC 60900 or equivalent) and have independent review of fail-safe mechanisms including tether attachment points and emergency power-cut protocols.

9.2 Data Privacy and Intellectual Property

The InsPLAD-det dataset was publicly released by the Universidade Federal de Pernambuco under an open-access license and its use in this non-commercial academic project is in accordance with the terms of the original publication terms [12]. All third-party libraries (OpenCV, Ultralytics YOLO11, Flask, ROS) are used under their open-source licenses (Apache 2.0, AGPL-3.0, BSD) with attribution obligations met in this report and the project repository. No proprietary model weights were redistributed; Roboflow API usage was within free-tier ToS

9.3 Societal and Environmental Impact

Blackouts crippling hospitals, water-treatment facilities and millions of households are caused by transmission-line failures. Automation in inspection reduces the occurrence of undetected faults and the probability of fatality to line workers. Kazakhstan is especially ripe for robotic inspection, as outdoor infrastructure work is hazardous year-round due to extreme temperatures. Environmental considerations include battery longevity and the LM2596 Buck Converter was selected for its efficiency over linear regulators. Future iterations should investigate regenerative braking during downhill traversal [5] to improve energy autonomy.

9.4 Broader Responsibilities and Transparency

This report documents all limitations and failures, without omission: the recall limitation of Phase 1 training, SAHI compatibility conflicts, the outstanding Iteration 4 assembly milestone, and empirically confirmed limitations of HSV-based tracking on the rope test rig. All results are reported in the interest of responsible engineering practice to allow for reproducibility and transparency. The team acknowledges that the robot is not ready to be deployed on live infrastructure and that significant further validation is needed before the system can be recommended for operational use.

Bibliography

- [1] A. Suarez, R. Salmoral, P. J. Zarco-Perinan, and A. Ollero, “Experimental evaluation of aerial manipulation robot in contact with 15 kv power line: Shielded and long reach configurations,” *IEEE Access*, vol. 9, pp. 94573–94585, 2021.
- [2] X. Yue, H. Wang, Y. Feng, Y. Tian, and W. Wang, “Improving stability of line inspection robot during crossing jumper lines with a centroid adjustment adjusting mechanism,” *IEEE Access*, vol. 10, pp. 134571–134579, 2022.
- [3] Q. Fu, H. Zhu, and Y. Guan, “Enhanced mobility and hybrid locomotion in modular biped rolling–climbing robots for power transmission line inspection,” *IEEE/ASME Transactions on Mechatronics*, pp. 1–12, 2024.
- [4] Y. Wang, X. Qin, W. Jia, J. Lei, D. Wang, T. Feng, Y. Zeng, and J. Song, “Multiobjective energy consumption optimization of a flying–walking power transmission line inspection robot during flight missions using improved nsga-ii,” *Applied Sciences*, vol. 14, no. 4, p. 1637, 2024.
- [5] Z. Yang, X. Liu, C. Ning, L. Liu, W. Tian, H. Wang, D. Zhang, H. Li, D. Zou, and J. Kuang, “Research on speed control methods and energy-saving for high-voltage transmission line inspection robots along cable downhill,” *Actuators*, vol. 12, no. 9, p. 352, 2023.
- [6] Y. Zeng, X. Qin, B. Li, J. Lei, J. Zhang, Y. Wang, and T. Feng, “A novel autonomous landing method for flying–walking power line inspection robots based on prior structure data,” *Applied Sciences*, vol. 13, no. 17, p. 9544, 2023.
- [7] J. Song, X. Qin, J. Lei, J. Zhang, Y. Wang, and Y. Zeng, “A fault detection method for transmission line components based on synthetic dataset and improved yolov5,” *International Journal of Electrical Power & Energy Systems*, vol. 157, p. 109852, 2024.
- [8] N. Surantha, “Real-time overhead power line component detection on edge computing platforms,” *Computers*, vol. 14, no. 4, p. 134, 2025.
- [9] Y. Zeng, J. Lei, T. Feng, X. Qin, B. Li, Y. Wang, D. Wang, and J. Song, “Neural radiance fields-based 3d reconstruction of power transmission lines using progressive motion sequence images,” *Sensors*, vol. 23, no. 23, p. 9537, 2023.
- [10] V. Leal Lopes, L. Honório, M. Santos, A. Pancoti, M. Silva, L. Diniz, and P. Mercorelli, “Design of an over-actuated hexacopter tilt-rotor for landing and coupling in power transmission lines,” *Drones*, vol. 7, no. 6, p. 341, 2023.
- [11] H. Xu *et al.*, “Transmission line detection based on improved Hough transform.” arXiv:2402.02761, 2024.
- [12] A. L. B. Vieira e Silva *et al.*, “InsPLAD: A dataset and benchmark for power line asset inspection in UAV images,” *International Journal of Remote Sensing*, vol. 44, no. 23, pp. 7294–7320, 2023.
- [13] Y. Ji, D. Zhang, Y. He, J. Zhao, X. Duan, and T. Zhang, “Improved yolo11 algorithm for insulator defect detection in power distribution lines,” *Electronics*, vol. 14, p. 1201, 03 2025.

- [14] J. Wang, Y. Wang, X. Li, B. Yuan, and M. Wang, “Sci-yolo11: An improved defect detection algorithm for transmission line insulators based on yolo11,” *PLOS ONE*, vol. 20, pp. 1–22, 10 2025.
- [15] F. C. Akyon, S. O. Altinuc, and A. Temizel, “Slicing aided hyper inference and fine-tuning for small object detection,” in *2022 IEEE International Conference on Image Processing (ICIP)*, pp. 966–970, IEEE, 2022.
- [16] W. Zhang *et al.*, “HSV color-space-based automated object localization for robot grasping without prior knowledge,” *Applied Sciences*, vol. 11, no. 16, p. 7593, 2021.
- [17] M. S. Alam *et al.*, “Color-texture based object tracking using HSV color space and local binary pattern,” in *Proceedings of the International Conference on Computer and Communication Informatics*, 2015.
- [18] M. Grinberg, “Video streaming with Flask.” <https://blog.miguelgrinberg.com/post/video-streaming-with-flask>, 2014. [Online; accessed 2026].