CEFET/RJ
DIRETORIA DE PESQUISA E PÓS-GRADUAÇÃO

Cooperative Aerial-Ground Robotic System for Solar Farms Inspection

Johann Steffanno Jerônimo da Costa Caniato Amorim

Defesa de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica, Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, CEFET/RJ, como parte dos requisitos necessários à obtenção do título de mestre.

Orientadores:
Milena Faria Pinto
Accacio Ferreira dos Santos Neto

Rio de Janeiro,
Janeiro, 2026

Cooperative Aerial-Ground Robotic System for Solar Farms Inspection

Defesa de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica, Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, CEFET/RJ, como parte dos requisitos necessários à obtenção do título de mestre.

Johann Steffanno Jerônimo da Costa Caniato Amorim

Aprovada por:

Presidente, Milena Faria Pinto (orientadora)

Accacio Ferreira dos Santos Neto, D.Sc. (coorientador)
CEFET/MG

Rafael da Silva Chaves, D.Sc.
CEFET/RJ

Luciana Faletti Almeida, D.Sc.
CEFET/RJ

Josiel Alves Gouvêa, D.Sc.
CEFET/RJ

Alexandre Santos Brandão, D.Sc.
UFV

Rio de Janeiro,
Janeiro, 2026

Ficha catalográfica elaborada pela Biblioteca Central do CEFET/RJ

B483 Amorim, Johann Steffanno Jerônimo da Costa Caniato
Cooperative aerial-ground robotic system for solar farms inspection / Johann Steffanno Jerônimo da Costa Caniato Amorim. — 2026.
xiii, 81f. : il. color. , enc.

Dissertação (Mestrado) Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, 2026.
Bibliografia : f. 76-81
Orientadora: Milena Faria Pinto
Coorientador: Accacio Ferreira dos Santos Neto

1. Drone. 2. Robôs móveis. 3. Coletores solares. 4. Redes neurais (Computação). I. Pinto, Milena Faria. (Orient.). II. Santos Neto, Accacio Ferreira dos. (Coorient.). III. Título.

CDD 623.7469

Elaborada pela bibliotecária Tania Mello – CRB/7 nº 5507/04

ACKNOWLEDGMENTS

The author would like to thank FAPERJ, CNPQ, CEFET/RJ, and CAPES. This work was carried out with the support of the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

I would like to sincerely thank my advisors, Dr. Milena Pinto and Dr. Accacio Neto, for their constant support, guidance, and dedication throughout this journey. I am also deeply grateful to my family for their understanding, patience, and continuous encouragement.

Johann Steffanno Jerônimo da Costa Caniato Amorim

You are worthy, Jehovah our God, to receive the glory and the honor and the power, because you created all things, and because of your will they came into existence and were created.

Revelation 4:11

RESUMO

Cooperative Aerial-Ground Robotic System for Solar Farms Inspection

Johann Steffanno Jerônimo da Costa Caniato Amorim

Orientadores:

Profa. Milena Faria Pinto

Prof. Accacio Ferreira dos Santos Neto

Defesa de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica, Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, CEFET/RJ, como parte dos requisitos necessários à obtenção do título de mestre em Engenharia Elétrica.

Esta dissertação propõe uma arquitetura colaborativa para a inspeção de parques solares, integrando Veículos Terrestres Não Tripulados (UGVs) e Veículos Aéreos Não Tripulados (UAVs). O objetivo é utilizar robôs autônomos em colaboração para inspeção de fazendas solares a fim de otimizar a inspeção de painéis, dos cabos e da estrutura. A partir disso, construindo essa cooperação, visa-se superar limitações operacionais comuns encontradas em sistemas isolados, como a vida útil limitada da bateria dos UAVs e a cobertura de campo restrita dos UGVs, que separados, tornariam a inspeção mais lenta e possivelmente menos eficaz. A solução desenvolvida emprega o modelo de rede neural convolucional You Only Look Once (YOLO) para a detecção e rastreamento de cabos elétricos pelo Unmanned Ground Vehicles (UGV), fornecendo informações visuais que apoiam a navegação e o posicionamento do Unmanned Aerial Vehicle (UAV) durante a missão. Crucialmente, o método Active Disturbance Reative Control (ADRC) é implementado para o controle de movimento robusto, garantindo rastreamento de alta precisão com Erro Máximo de Trajetória de 0,15 m com 82% de mellhoria em relação ao controle PID. A cooperação entre os robôs é orquestrada por meio de um mecanismo human-in-the-loop e de algoritmos de planejamento de caminho baseados no método $A^{\star}$, permitindo a tomada de decisão adaptativa e a coordenação em tempo real. A abordagem proposta foi validada através de simulações no Robot Operating System (ROS) e no ambiente Gazebo, demonstrando desempenho robusto em navegação autônoma, rastreamento de cabos e pouso seguro de UAV apoiado pelo UGV. Os resultados confirmaram a eficiência e viabilidade da arquitetura proposta para inspeções automatizadas em ambientes industriais, destacando o potencial da integração de visão computacional, ADRC, e cooperação multi-robô para aplicações em sistemas fotovoltaicos e infraestrutura crítica.

Palavras-chave: Colaboração UGV-UAV. Sistemas multi-robôs. Otimização de trajetória. Inspeção robótica. Rede neural convolucional.

ABSTRACT

Cooperative Aerial-Ground Robotic System for Solar Farms Inspection

Johann Steffanno Jerônimo da Costa Caniato Amorim

Advisors:

Profa. Milena Faria Pinto

Prof. Accacio Ferreira dos Santos Neto

Master's Thesis Defense submitted to the Electrical Engineering Graduate Program, Federal Center of Technological Education Celso Suckow da Fonseca, CEFET/RJ, as partial fulfillment of the requirements for the degree of Master's in Electrical Engineering.

This dissertation proposes a collaborative architecture for the inspection of solar parks, integrating Unmanned Ground Vehicles (UGVs) and Unmanned Aerial Vehicles (UAVs). The objective is to employ autonomous robots in collaboration for the inspection of solar farms in order to optimize the inspection of panels, cables, and structural components. By establishing this cooperation, the goal is to overcome common operational limitations found in isolated systems, such as the limited battery life of UAVs and the restricted field coverage of UGVs, which, if operating separately, would render inspections slower and potentially less effective. The developed solution employs the YOLO convolutional neural network model for the detection and tracking of electrical cables by the UGV, providing visual information that supports the navigation and positioning of the UAV during the mission. Crucially, the ADRC method is implemented for robust motion control, ensuring high-precision tracking with a Maximum Trajectory Error of 0.15 m, representing an 82% improvement over PID control. The cooperation between the robots is orchestrated through a human-in-the-loop mechanism and path planning algorithms based on the $A^{\star}$ method, allowing for adaptive decision-making and real-time coordination. The proposed approach was validated through simulations in ROS and the Gazebo environment, demonstrating robust performance in autonomous navigation, cable tracking, and UGV-supported safe UAV landing. The results confirmed the efficiency and viability of the proposed architecture for automated inspections in industrial environments, highlighting the potential of integrating computer vision, ADRC, and multi-robot cooperation for applications in photovoltaic systems and critical infrastructure.

Key-words: UGV-UAV collaboration. Multi-robot systems. Trajectory optimization. Robotic inspection. Convolutional neural network.

Contents

I Introduction **14**

I.1 Initial Considerations and Work Motivation 14

I.2 Main Contributions 16

I.3 Organization 17

II Background and Related Works **19**

II.1 Heterogeneous Robots Collaboration 19

II.2 Robots Applied to Solar Panel Farms 19

II.3 Computer Vision Based Path Following 21

II.4 ADRC Strategy 22

II.5 Reflections on Related Works 22

III Proposed Methodology **25**

III.1 Overall Idea 25

III.2 Proposed Solution 26

III.3 Solution Setup 27

III.4 Computer Vision-based Tracking and Monitoring 29

III.4.1 Application on a Digital Twin 31

III.5 Navigation and Cooperation 33

III.6 Robot kinematic 37

III.7 Robot dynamics 39

III.8 ADRC 40

IV Results of ADRC **44**

IV.1 Developed Robot Performance using ADRC compared to PID 44

IV.1.1 Testing Setup 44

IV.1.2 Active Disturbance Rejection Control (ADRC) 44

IV.1.3 Proportional-Integrative-Derivative (PID) Control 45

IV.1.4 Controller Parameters 45

V Results of Computer Vision and Digital Twin **52**

V.1 Computer Vision Results 52

V.1.1 Testing on Real Datasets 52

V.1.2 Testing on Simulated Synthetic Datasets 54

V.1.3 Testing on Mixed Datasets 56
V.1.4 Test Preformance metrics 58
V.2 Cable Follower and Path Planning Results 62

VI Results of Cooperation Task 67
VI.1 UAV-UGV Role Division 67
VI.2 Human-in-the-Loop Evaluation 68
VI.3 Comparison with Traditional Inspection Methods and Single-Robot Approaches 71

VII Conclusions and Future Work 73
VII.1 Conclusions 73
VII.2 Future Proposals and Limitations 74
VII.3 Scientific Production 75
Bibliographic References 76

List of Tables

II.1 A summary of the related works including robot types and collaboration strategies. . . . 24

III.1 Main hyperparameters used in YOLOv5m training 32
III.2 Dynamic model parameters 40

IV.1 System and Controller Parameters 45
IV.2 Comparative Performance Metrics: ADRC vs. PID 51

V.1 Results of training dataset tested on real images dataset 59
V.2 Results of training dataset tested on gazebo dataset 59

VI.1 UGV-UAV Cooperation Performance Metrics 68
VI.2 Comparison between UAV and UGV coverage capabilities 69
VI.3 Summary of HITL Scenarios 69
VI.4 Qualitative comparison of Photovoltaic inspection approaches 72

List of Figures

III.1 General Idea flowchart. Source: (AMORIM et al., 2025b) . . . 25
III.2 Cooperative framework. Based on a paper published by the authors in the Journal of Intelligent & Robotic Systems (AMORIM et al., 2025b) . . . 27
III.3 World created in the Gazebo Software. . . . 29
III.4 YOLO structure for cable following purposes. Source: (AMORIM et al., 2025b) . . . 30
III.5 Simulated world on software Gazebo. Source: (AMORIM et al., 2025b) . . . 32
III.6 Sample images of the dataset provided by (ZANELLA, 2023). . . . 33
III.7 Synthetic images incorporated in the dataset for training. Source: (AMORIM et al., 2025b) . . . 33
III.8 Images used for real-world test from III.6 . . . 33
III.9 UGV framework. Source: (AMORIM et al., 2025b) . . . 34
III.10 Decision-making process when the UAV requests to land. Source: (AMORIM et al., 2025b) . . . 35
III.11 Robot axis and Global axis representation. Source: Author . . . 37
III.12 ADRC diagram for navigation purposes. Adapted from (HERBST, 2013) . . . 43

IV.1 Comparison of Position Reference X System's output using different controllers. Source: Author . . . 47
IV.2 Comparison of control effort (wheel torques) generated by PID and ADRC strategies. Source: Author . . . 48
IV.3 Comparison between actual disturbances and ESO-estimated disturbances along the x, y, α axes, highlighting observer convergence and estimation accuracy. Source: Author 49
IV.4 Robot's trajectory using ADRC X Proporcional, Integral e Derivativo (PID) Control. Source: Author . . . 50

V.1 Performance metrics for the model (F1, PR, and Confusion Matrix) trained on synthetic images tested on real-world images. Source: Author . . . 53
V.2 Performance metrics for the model (F1, PR, and Confusion Matrix) trained on syntehtic images tested on simulated images. Source: Author . . . 54
V.3 Performance metrics for the model (F1, PR, and Confusion Matrix) trained on simulated images tested on simulated images. Source: Author . . . 55

V.4 Performance metrics for the model (F1, PR, and Confusion Matrix) trained on a mixed dataset (25% of real syntheic images + 75% of simulated images) tested on simulated images. Source: Author 56
V.5 Performance metrics for the model (F1, PR, and Confusion Matrix) trained on a mixed dataset (50% of real syntheic images + 50% of simulated images) tested on simulated images. Source: Author 57
V.6 Performance metrics for the model (F1, PR, and Confusion Matrix) trained on a mixed dataset (75% of real syntheic images + 25% of simulated images) tested on simulated images. Source: Author 58
V.7 Results of training step of dataset compound with 25% of simulated images. Source: Author 60
V.8 Detection in a Real scenario for 75% real synthetic images + 25% simulated images. Source: Author 60
V.9 Detection in a Real scenario for real synthetic images. Source: Author 61
V.10 Detection in an Ideal scenario for 50% real synthetic images + 50% simulated images. Source: Author 61
V.11 Detection in an Ideal scenario for simulated images. Source: Author 62
V.12 AGV recognizing the cable and performing the tracking. Source: (AMORIM et al., 2025b) 63
V.13 Path taken by the robot when following the cable. Source: (AMORIM et al., 2025b) . . 64
V.14 Position of robot for Cable Tracking. Source: Author 64
V.15 Data from the control approach used to follow the cable. Source: (AMORIM et al., 2025b) 65
V.16 Robot track to the UAV position requisition using $A^{\star}$ algorithm. Source: (AMORIM et al., 2025b) 65
V.17 Coordinated UAV and UGV inspection paths in a solar panel layout. The UAV follows a zigzag path covering the panel rows, while the UGV traverses between the rows and serves as a mobile charging station. A scale bar is included to visualize distances. Source: (AMORIM et al., 2025b) 66

VI.1 Interblock communication among the proposed architecture. Source: (AMORIM et al., 2025b) 68
VI.2 System-level HITL communication architecture. Colored arrows represent topic-based message flow from sensors to fallback actions through the operator node. Source: (AMORIM et al., 2025b) 70

List of Abbreviations

4IR Fourth Industrial Revolution 14

ADRC Active Disturbance Reative Control . . . . . v, vii, xi, 17, 40, 45, 46, 47, 48, 50, 51, 73

CNN Convolutional Neural Network 16, 25

ESO Extended State Observer 40, 41, 48, 50

GCS Ground Control Station 23, 25, 26, 28, 34

GPS Global Positioning System 28

IMU Inertial Measurement System 28

PID Proporcional, Integral E Derivativo . . . . . xi, 45, 46, 47, 50, 51

ROS Robot Operating System . . . . . v, vii

UAV Unmanned Aerial Vehicle . . . . . v, vii

UGV Unmanned Ground Vehicles . . . . . v, vii

YOLO You Only Look Once . . . . . v, vii, 16

Chapter I Introduction

I.1 Initial Considerations and Work Motivation

In the Fourth Industrial Revolution (4IR) era, robots are building innovative solutions to advanced technology for efficient and sustainable approaches to various fields and challenges (ALMALKI et al., 2021). The use of UAV (Unmanned Aerial Vehicle) is increasing due to their fast and cost-effective deployment, flexibility of flight, and adaptable use (RAMOS et al., 2022), (COELHO et al., 2024). The possibility of equipping UAVs with cameras and suitable sensors has allowed the use of these robots for surveys and inspections (PINTO et al., 2019), (CASTRO et al., 2023). However, UAVs are limited by battery life and payload capacity, making Unmanned Grounded Vehicle (UGV) more suitable for heavy and long-duration tasks like logistics or support (REA; OTTAVIANO, 2023). UGVs are well-suited for tasks that require transporting substantial payloads or sustaining operations for long durations without frequent battery changes or refueling. Unlike aerial systems, ground vehicles can manage much heavier cargo, such as sensors, tools, or other essential equipment (CASTRO et al., 2024).

Recent advancements in robotics and Artificial Intelligence (AI) have introduced several strategies to enhance perception and control in complex and constrained environments. Additionally, adaptive fault-tolerant control techniques for UAVs have been developed using fuzzy wavelet neural networks, ensuring robustness under actuator faults and irregular triggering conditions (SONG et al., 2024). These contributions reflect a growing trend toward making autonomous systems more resilient, responsive, and capable of operating in real-world conditions, which also motivates the development of mobile robotic platforms for field applications.

Wheeled mobile robots are the most prevalent type, known for their fast, reliable movement, low power usage, and relatively simple control systems. However, they struggle when faced with obstacles whose dimensions are larger than their wheel radius or with sizes comparable to them (REA; OTTAVIANO, 2023). Mobile robots represent an interesting field of research due to their vast potential and diverse applications, ranging from inspection tasks (HOFER et al., 2023), search and rescue operations (HABIBIAN et al., 2021), and manufacturing processes (DÖRFLER et al., 2022), to remote exploration (SHAPOVALOV; PEREIRA, 2020), agriculture (HAJJAJ; SAHARI, 2014), and more.

As stated by Wang and Ueda (WANG; UEDA, 2023), nowadays, there are plenty of researchers developing mobile robots equipped with advanced high-tech equipment to provide automated and high-efficiency data acquisition, bringing more flexibility to the inspection process. The literature presents different mobile robot-based platforms for damage inspection and various purposes, such as non-structural damage inspection (NITTA et al., 2013), pipeline inspection (BARANOWSKI; SIWEK, 2018), (KAKOGAWA; KOMURASAKI; MA, 2019), localization techniques of inspection-aimed mobile robots

(SZREK et al., 2020). Within this context, the use of mobile robots has also gained relevance in the inspection of large-scale infrastructure, including photovoltaic systems, where autonomous navigation and high-resolution sensing can significantly enhance monitoring efficiency and reduce operational costs.

Meribout et al. (MERIBOUT et al., 2023) emphasized that the depletion of nonrenewable global resources is accelerating the transition toward renewable energy, which is expected to dominate the electricity markets shortly (Mornsun Power, 2023). Despite the rapid expansion of large photovoltaic installations, some important challenges remain, such as efficient system localization, real-time fault detection, and diagnostic processes in operational settings.

These difficulties have inspired and motivated several research proposals (ABUBAKAR; ALMEIDA; GEMIGNANI, 2021), (YAHYA et al., 2022). For instance, UAVs are revolutionizing the solar energy industry by enabling cost-effective and efficient methods for detecting "hotspots" on solar panels (PRUTHVIRAJ et al., 2023). However, other types of robots have been developed for photovoltaic power station inspection. In the work of Felsch et al. (FELSCH et al., 2015), the authors developed and applied a climbing robot to inspect vertical structures within solar power plants. Venkatnikhil et al. (VENKATNIKHIL; RAVICHANDRAN; KUMAR, 2022) developed a rover robot capable of cleaning and monitoring solar panels. Additionally, Chellal et al. (CHELLAL et al., 2021) developed a robot that cleans and cools solar panels using camera-acquired images to enhance operational efficiency.

Another widely investigated alternative is the use of heterogeneous robotic systems that combine UAVs and UGVs to tackle challenges regarding the inspection and maintenance of photovoltaic systems. These systems integrate the complementary capabilities of aerial and ground robots to improve inspection efficiency, precision, and coverage, reducing downtime and optimizing resources (HALDER; AFSARI, 2023). However, integrating these platforms presents technical and operational challenges that warrant further investigation, particularly regarding autonomy, coordination, and real-time decision-making (CASTRO et al., 2024).

Despite these advances, most existing approaches still rely on conventional control strategies that struggle to cope with the high levels of uncertainty, external disturbances, and strong coupling effects observed in real solar farm environments. Factors such as uneven terrain, wind gusts affecting UAV stability, sensor noise, and interaction forces during close-range inspection introduce disturbances that degrade trajectory tracking quality and may lead to mission failure (HERNÁNDEZ; ALMEIDA; SOLÍS, 2025). This gap shows the need for more robust control frameworks that can maintain stability and performance under unpredictable operating conditions. In this context, *Active Disturbance Rejection Control* (ADRC) emerges as a promising solution, as it provides real-time estimation and compensation of internal and external disturbances without requiring an accurate dynamic model (ZACHI et al., 2025; GOUVÊA et al., 2023). The integration of ADRC on UGV for heterogeneous UAV-UGV inspection systems can significantly enhance autonomy, safety, and inspection consistency, offering a more resilient

alternative to classical PID or model-dependent controllers

I.2 Main Contributions

Building on these advancements, this work introduces a collaborative framework between UGVs and UAVs, specifically designed to inspect solar panels. In this approach, the UGV transports the UAV to a predefined location within the solar farm. Once in place, the UAV launches to start its inspection mission. Using the Convolutional Neural Network (CNN) model, YOLO version 5, the UGV is responsible for identifying and tracking the cables for inspection. This computer vision capability serves as a cable-follower system for the UAV, enabling accurate guidance during the mission. If the UAV's battery level falls below a critical threshold during inspections, it signals the UGV to reposition itself nearby, enabling a secure UAV landing. A path planning algorithm based on the $A^{\star}$ algorithm is then activated to compute the fastest route to the UAVs location, ensuring efficient coordination and minimizing downtime during the operation.To enhance robustness during motion in uncertain field conditions, such as uneven terrain, variable illumination, and wind disturbances affecting UGV stability, the framework incorporates the ADRC strategy to enable real-time estimation and compensation of internal and external disturbances, improving trajectory accuracy and system resilience without requiring precise dynamic modeling.

In this study, an omnidirectional robot was utilized to evaluate motion control across all three Degrees of Freedom (DOF). Although omnidirectional locomotion is generally not recommended for the irregular terrain typical of solar farms, the primary objective was to characterize its dynamics for application on the omnidirectional platform of Laboratório de Controle e Automação (LACEA). Furthermore, the use of this specific kinematic model does not limit the scope of this research; the proposed framework is modular and platform-agnostic, allowing it to be adapted to various robotic locomotion systems. This collaborative system guarantees continuous and efficient solar panel inspection with minimal human intervention. This proposed approach was assessed through simulations in the Robot Operating System (ROS) and the Gazebo platform (KOENIG; HOWARD, 2004). The main contributions of this research can be summarized as follows:

1. The proposed collaborative framework for UGVs and UAVs optimizes solar panel inspections by using their complementary capabilities. In this methodology, the UAV conducts aerial assessments to verify panel integrity, while the UGV inspects cabling and infrastructure beneath the photovoltaic system. Besides, it assists the UAV during a secure landing, i.e., a hot-swapping process.

2. The integration of advanced computer vision technology using the YOLOv5 model enables the UGV to accurately detect and track cables for inspection while guiding the UAV during its mission.

3. Introduction of the ADRC on UGV strategy to enhance robustness against disturbances during navigation, improving trajectory tracking performance in outdoor solar farm environments.

4. Assessment of the framework performance through simulations with a digital twin of the robots in Gazebo software through ROS environment, validating its effectiveness and practicality for real-world solar panel inspection scenarios.

Note that the current work method aims to be used for all gathered inspection data to be later processed and analyzed to detect potential issues, such as cable faults or damage to the solar panels. This automated system provides a comprehensive and efficient solution for maintaining solar farm facilities with minimal human intervention.

I.3 Organization

The remainder of this dissertation is organized as follows:

- **Chapter II**: This chapter reviews relevant literature on heterogeneous robot collaboration, robotic applications in solar panel farms, computer vision-based tracking, and the educational use of robotics. It provides a critical overview of existing solutions and justifies the research direction taken in this work.

- **Chapter III**: This chapter details the collaborative architecture developed in this work. It explains the kinematics and dynamics of an Omnidirectional Four-Wheeled Robot. It evaluates the implementation of ADRC considering the dynamics of the robot. It describes integrating the YOLOv5 model for cable tracking, the navigation and coordination strategies between UAV and UGV, and implementing a human-in-the-loop decision-making layer.

- **Chapter IV**: This chapter presents the results obtained from simulations and experiments. It evaluates the proposed controller, ADRC, focused on the Omnidirectional Four-Wheeled Robot's dynamics model.

- **Chapter V**: This chapter presents the results obtained from the training, validation, and testing of the proposed dataset. This chapter shows what happens if the dataset is mixed with simulated images from a digital twin and compares its performance in different scenarios. Furthermore, this chapter shows the performance of cable following and path–planning trajectory validated on the digital twin environment.

- **Chapter VI**: This chapter presents the results of this proposed framework and how the inspection task can be optimized when using heterogeneous robots.

- **Chapter VII**: The final chapter summarizes the findings and contributions of the research. It outlines the limitations of the current implementation and proposes directions for future development, including field testing, enhancement of autonomy, and broader applications in both industrial and educational contexts.

Chapter II Background and Related Works

II.1 Heterogeneous Robots Collaboration

The collaboration among heterogeneous robotic systems has emerged as a key paradigm for addressing complex missions that cannot be efficiently solved by a single type of robot. By combining platforms with distinct locomotion capabilities, sensing modalities, endurance profiles, and payload capacities, heterogeneous teams enable complementary strengths to be exploited while mitigating individual limitations. Various strategies have been developed to enable collaborative task execution among heterogeneous robots. For example, Roldán et al. (ROLDÁN et al., 2016) proposed a collaborative framework for monitoring environmental conditions within greenhouses. They employed ground and aerial robots to gather critical data, such as temperature, humidity, light intensity, and carbon dioxide levels, essential for maintaining ideal plant growth conditions. The ground robots provided durability and extended operational time, while the aerial robots offered faster mobility and access to hard-to-reach areas.

Berger et al. (BERGER et al., 2023) focused on a collaboration between UGV and UAV for inspection tasks. One of the key innovations in their work was the process that allowed the UAV to land on the UGV for a hot-swapping mechanism. This approach extended the mission duration without human intervention, highlighting how combining different robotic platforms can enhance operational efficiency and endurance. Across these studies, the UAV consistently plays a central role in facilitating collaborative efforts among the robots. However, UGVs offer unique advantages that complement the UAVs capabilities. Their robustness, longer operational endurance, and ability to carry heavier payloads make them particularly well-suited for tasks requiring sustained operations over extended periods. As demonstrated in the work of Berger et al. (BERGER et al., 2023), the UGV enabled UAV hot-swapping, extending mission duration without human intervention.

UGVs also provide a stable platform for sensors and tools that require proximity to the ground, offering more detailed inspections and data collection than UAVs can achieve at higher altitudes. For instance, Castro et al. (CASTRO et al., 2024) explored the cooperation between two UAVs and a UGV for conducting an inspection mission. Both types robots are responsible for inspection tasks. However, they navigate partially unknown terrains and deal with possible dynamic obstacles.

II.2 Robots Applied to Solar Panel Farms

Several studies have concentrated on detecting anomalies in solar panels, as these irregularities can lead to fluctuations in energy output and temperature (DÍAZ et al., 2020) hotspots, which indicate localized areas of overheating, can be classified into different failure levels depending on their shape and size (JAFFERY et al., 2017). For anomaly detection in solar panels, UAVs offer significant advantages

that make them highly effective. Their ability to quickly access large or hard-to-reach areas allows for efficient inspection of vast solar farms or rooftop installations. UAVs, equipped with thermal cameras, can capture detailed images from optimal angles, improving the detection of anomalies like hotspots or material fatigue, which may not be visible from ground-based perspectives (LIAO; LU, 2020), (ADDABBO et al., 2017).

Moreover, regarding detecting anomalies, such as hotspots or material damage, in addition to UAVs, other types of robots were reported in the context of photovoltaic systems. For instance, Felsch et al. (FELSCH et al., 2015) developed a climbing robot to inspect vertical structures within solar power plants. Chellal et al. (CHELLAL et al., 2021) developed a robot that cleans and cools solar panels using camera-acquired images to enhance operational efficiency. Furthermore, Mousavi et al. (MOUSAVI; FARAHANI, 2022) introduced a robot for cleaning photovoltaic systems. These innovations demonstrate the growing role of ground-based robotic systems in the automated inspection and maintenance of solar energy infrastructure, complementing aerial inspections and contributing to more comprehensive and efficient solar farm management.

While there is extensive research on the inspection and monitoring of photovoltaic systems, particularly focusing on solar panels and their performance, fewer studies have addressed the inspection of the broader photovoltaic infrastructure, including cables and supporting structures. For instance, research on the cable integrity of photovoltaic systems has highlighted the importance of regular inspections due to the critical role that cables play in energy transmission (HERRMANN et al., 2021), (ZHANG et al., 2018), (EKICI; KOPRU, 2017). To the best of the authors' knowledge, there is a gap in the literature regarding the application of robots to inspect these critical components.

Integrating heterogeneous robots-comprising both aerial and ground units-could significantly enhance the inspection processes of photovoltaic systems' infrastructure. UAVs equipped with advanced sensors can be applied to monitor and detect hotspots, insulation degradation, and other surface-level issues on solar panels, cables, and supporting structures from above. Simultaneously, ground robots can conduct more detailed inspections to assess cable integrity, structural elements, and hidden defects not visible to aerial systems. The use of heterogeneous robots can reduce the need for manual inspections, which are often time-consuming, expensive, and hazardous in large photovoltaic installations. Additionally, these robots can comprehensively assess the photovoltaic infrastructure by combining real-time data from various sensors. To achieve this detailed, ground-level assessment (such as cable following), robots must be equipped with robust sensors like cameras; however, processing this visual data effectively requires the use of computer vision to detect and track the cables.

II.3 Computer Vision Based Path Following

The rapid expansion of large-scale renewable energy infrastructure has created a critical demand for automated maintenance and monitoring solutions. To ensure operational efficiency and reduce human risk in expansive environments like solar farms, the development of autonomous robots capable of precise navigation is essential. In this context, computer vision has emerged as an important engineering and computer science field, with applications spanning diverse industries. For instance, Golightly and Jones (GOLIGHTLY; JONES, 2005) proposed a vision system to identify cables with Hough transform. Song and Li (SONG; LI, 2014) proposed a sequential detection framework to identify curved power cables. Bhola et al. (BHOLA et al., 2018) developed a spectral clustering approach for segmenting power cable pixels from aerial images. It is important to highlight that several efforts were made in the literature using traditional methods. However, in scenarios like real-world solar farms, these conventional methods can struggle with diverse backgrounds, varying lighting conditions, and noise or occlusions, leading to inconsistent and unreliable detection (DANTAS et al., 2020).

In recent years, advancements in deep learning have revolutionized computer vision tasks, including object detection and semantic segmentation (DAI et al., 2020). CNNs have demonstrated remarkable performance improvements in accuracy for challenging computer vision problems. Early models, such as R-CNN (GIRSHICK et al., 2014), introduced CNNs into object detection. Recently, YOLO (You Only Look Once) (REDMON, 2016), a pioneering deep learning-based framework, offered a streamlined and efficient pipeline for real-time object detection. Subsequent versions, including YOLOv3 (REDMON; FARHADI, 2018), optimized the model further, introducing a compact architecture capable of achieving high precision and computational efficiency.

Several versions of YOLO algorithm have been introduced through the foundation of YOLO architecture to improve performance and accuracy. Among these, YOLOv4 (BOCHKOVSKIY; WANG; LIAO, 2020), YOLOv5 (JOCHER, 2020), the version utilized in this study, YOLOv6 (LI et al., 2022), YOLOv7 (WANG; BOCHKOVSKIY; LIAO, 2023), and YOLOv8 (JOCHER; CHAURASIA; QIU, 2023). Each version focuses on advancing real-time object detection capabilities while achieving greater precision in object localization. The YOLOv5 has advanced the state-of-the-art in several ways. YOLOv5 is lightweight, versatile, and scalable, making it well-suited for real-time applications in dynamic and resource-constrained environments. It employs anchor-based detection with an optimized architecture integrating a CSPDarknet53 backbone for feature extraction, a Path Aggregation Network (PANet) for multi-scale feature fusion, and a detection head tailored for precise bounding box prediction and classification. These innovations allow YOLOv5 to perform exceptionally well even under challenging conditions, such as detecting cables amidst cluttered or highly textured backgrounds (JOCHER, 2020).

II.4 ADRC Strategy

The choice of ADRC for this work is justified by the challenges of solar farm environments, where unpredictable factors such as terrain irregularities, varying friction, and external disturbances, for example dusts, can compromise the stability of traditional controllers. Unlike PID controllers, which may require frequent retuning, or Model Predictive Control (MPC), which relies on highly accurate plant dynamics, ADRC employs an Extended State Observer (ESO) to estimate and compensate for both internal uncertainties and external perturbations in real-time. This provides the UGV with the necessary robustness to maintain precise path-following performance even when operating under non-ideal conditions, ensuring the reliability of the collaborative inspection mission.

Active Disturbance Rejection Control (ADRC) has gained significant relevance as a robust alternative to conventional control strategies, offering high performance with a simplified design (HERBST, 2013). The continuous development of this method is evident through several key contributions in recent literature.

The work by (HERBST, 2013) highlighted the appeal of the Linear ADRC (LADRC) variant for practitioners, emphasizing its simple tuning method and demonstrated robustness against general process parameter variations. Gouvêa et al. (2020) focused on refining the observer structure, proposing a new LADRC paradigm that utilizes only a single observer estimate in the control law, thus simplifying the control design while maintaining equivalent stability properties (GOUVÊA; VASCONCELLOS; ZACHI, 2020). Demonstrating the practical efficacy of the method, Khaled et al. (2021) successfully applied an ADRC scheme for dynamic path correction to significantly improve the path accuracy of an industrial robot using external sensor feedback (KHALED; AKHRIF; BONEV, 2020).

Collectively, these studies establish ADRC as a powerful and practically viable method for high-precision control in robotics and other uncertain systems.

II.5 Reflections on Related Works

In the literature, several works have focused on using UAVs to detect power cables (DAI et al., 2020), (JENSSEN; ROVERSO et al., 2018), (JENSSEN; ROVERSO et al., 2019), and Autonomous Underwater Vehicles (AUVs) to identify underwater cables (FENG et al., 2023), (LI; ZHANG; SHEN, 2022). Although these studies make significant contributions, they also highlight several limitations that the proposed solution aims to address. It is worth highlighting that many previous works often focused on isolated tasks, such as detecting anomalies in solar panels (DÍAZ et al., 2020) or identifying power cables using UAVs (BHOLA et al., 2018). However, in most studies, limited collaboration between ground and aerial robots restricts their ability to achieve comprehensive infrastructure coverage.

A major gap highlighted in recent literature is the limited adoption of heterogeneous robotic collaboration

(CASTRO et al., 2024). The absence of tightly coupled UGV-UAV systems constrains robustness, continuity of inspection, and environmental awareness, especially in large-scale solar farms where terrain variability and distributed assets demand complementary robotic capabilities. To the best of the authors knowledge, no prior research has proposed a unified framework combining (i) real-time cable tracking via advanced computer vision (YOLO-based), (ii) coordinated path planning through the $A^{\star}$ algorithm, and (iii) mutual interaction between aerial and ground robots using a Ground Control Station (GCS) to guarantee complete inspection coverage.

Furthermore, most existing approaches rely on classical control techniques that can be sensitive to disturbances such as uneven terrain, wind gusts, illumination variability, and sensor noiseconditions frequently encountered in outdoor solar fields. These challenges highlight the need for a robust control methodology capable of maintaining high tracking accuracy and stable operation under model uncertainties. In this context, ADRC emerges as a suitable alternative, compensating for internal and external disturbances in real time without requiring precise modeling of the system dynamics. By integrating ADRC with the proposed heterogeneous inspection pipeline, this work aims to increase autonomy, reliability, and overall operational resilience of solar farm inspection tasks.

To better understand the current state of the art and identify research gaps, Table **??** presents a comparative summary of relevant studies involving robotic systems for inspection tasks. This table highlights the main focus, technologies used, and whether the system includes heterogeneous robot collaboration. While several approaches utilize UAVs or UGVs independently, relatively few adopt a tightly integrated UAV-UGV architecture with shared decision-making and contextual awareness. The proposed solution builds upon these efforts by introducing a collaborative framework that incorporates advanced computer vision and adaptive coordination strategies for solar farm inspections.

Previous research has explored a wide range of robotic inspection tasks across agricultural and industrial settings. Early approaches, such as Roldán et al. (ROLDÁN et al., 2016), relied on traditional sensing for greenhouse monitoring-effective but limited in autonomy in some scenarios, such as having some flights carried out by manual control. Other works targeted specific inspection domains: Felsch et al. (FELSCH et al., 2015) for vertical structures and Bhola et al. (BHOLA et al., 2018) for aerial cable detection. Although these studies introduced specialized methods, they did not consider heterogeneous collaboration or broader operational adaptability. In the solar energy domain, Díaz et al. (DÍAZ et al., 2020) used UAV-based thermal imaging for hotspot detection, while Chellal et al. (CHELLAL et al., 2021) developed autonomous cleaning robots. Both solutions addressed relevant tasks but remained restricted to single-robot systems and lacked combined ground-air perspectives. More recent efforts involving heterogeneous teams, such as Berger et al. (BERGER et al., 2023) and Castro et al. (CASTRO et al., 2024), have demonstrated the advantages of UAV-UGV cooperation for agriculture and navigation in challenging terrain. Despite their contributions, these works did not focus on solar farm inspection

nor integrate cable tracking with panel analysis. The proposed solution aims to address these gaps by combining UGV-based cable tracking with UAV-based solar panel inspection. This heterogeneous strategy leverages task specialization and complementary sensing, providing a more comprehensive and efficient inspection framework than existing single-robot or domain-limited approaches.

Table II.1: A summary of the related works including robot types and collaboration strategies.

Year	**Work**	**Task**	**Robot Type**	**Focus**	**Technologies Used**	**Heterog. Robots**	**Collab.**
2016	Roldan et al.(ROLDÁN et al., 2016)	Environmental monitoring	UGVs and UAVs	Greenhouse environmental monitoring.	Traditional sensors for data collection.	Yes	Yes
2015	Felsch et al.(FELSCH et al., 2015)	Structural assessment	Climbing robots	Vertical structure inspection.	Climbing robots for physical inspection.	No	No
2018	Bhola et al.(BHOLA et al., 2018)	Infrastructure detection	UAVs	Aerial cable detection.	Spectral clustering for cable segmentation.	No	No
2020	Vega et al.(DÍAZ et al., 2020)	Thermal inspection	UAVs	Anomaly detection in solar panels.	UAVs with thermal imaging for hot spot detection.	No	No
2021	Chellal et al.(CHELLAL et al., 2021)	Maintenance and cleaning	Cleaning robots	Solar panel cleaning and cooling.	Autonomous cleaning robots.	No	No
2023	Berger et al.(BERGER et al., 2023)	Cooperative monitoring	UGV and UAV	Agriculture inspection with UGV-UAV	UAV-UGV collaboration with hot-swapping mechanism.	Yes	Yes
2024	Castro et al.(CASTRO et al., 2024)	Path planning and mapping	UGV and UAV	Navigation and inspection in unknown terrain.	UGV-UAV collaboration for obstacle navigation and inspection.	Yes	Yes
2025	**Proposed Solution**	Coordinated solar inspection	UGV and UAV	Solar farm inspection with UGV-UAV.	UGV for cable tracking and UAV for solar panel inspection.	Yes	Yes

Chapter III Proposed Methodology

III.1 Overall Idea

This work proposes a solution to the challenge of automating and intelligently inspecting cables in solar farm installations. The objective is to design a collaborative approach that harnesses the capabilities of UGVs and UAVs to perform inspections efficiently, synchronously, and with real-time data exchange (see Figure III.1). The methodology was published on Journal of Intelligent & Robotic Systems (AMORIM et al., 2025b) This problem involves solving the following subproblems:

1. **Transport and initial positioning of the UAV by the UGV**: Developing an integrated solution to transport the UAV to specific areas of the solar farm.
2. **Cable detection and tracking by the UGV using CNN (YOLO)**: Implementing robust visual detection to track cables and identify potential structural damages.
3. **Coordination and real-time data exchange between the UGV and UAV**: Ensuring synchronized operation of both robots to maximize inspection efficiency.
4. **UAV autonomy management**: The battery management procedure is triggered when the UAV's battery level falls to a critical threshold. Upon receiving a notification from the UAV, the GCS calculates a new route for the UAV to a safe point. Subsequently, the GCS employs the $A^{\star}$ algorithm to generate a path for the UGV to travel to the same destination to provide support.

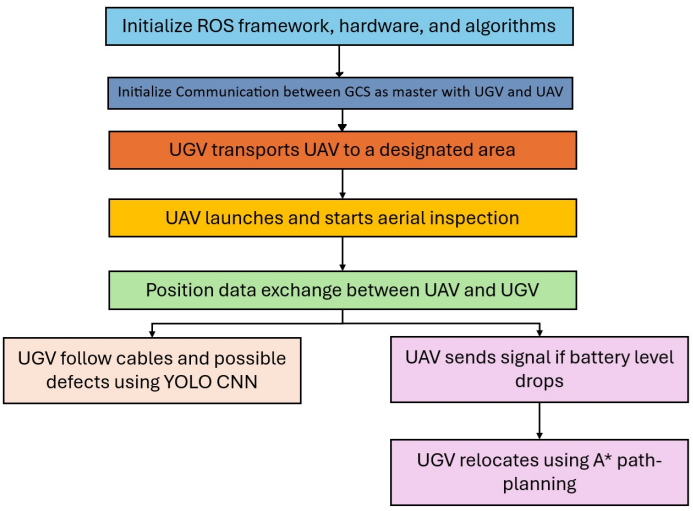


Figure III.1: General Idea flowchart. Source: (AMORIM et al., 2025b)

The objective is to overcome these challenges through control and planning algorithms, ensuring accurate and continuous inspection. To initialize the collaborative inspection procedure, the system requires several predefined parameters that influence the missions execution. These include: (i) the initial pose of the UGV, configured in the map frame based on the simulated environment layout; (ii) the predefined inspection route for the UGV, specified as a sequence of waypoints used by the ROS *move_base* package and the $A^\star$ path planning algorithm; (iii) the UAVs nominal take-off altitude, configured to ensure safe clearance from the ground and infrastructure (approximately 3 meters in the simulation environment); (iv) the battery level threshold of 20%, which triggers UAV landing support procedures by the UGV; and (v) the configurable confidence threshold of the YOLOv5 detector, which is adjusted to balance sensitivity and precision based on environmental conditions and inspection requirements. These parameters are defined during mission setup and loaded via the ROS parameter server. All experiments used the same initialization values to ensure consistency and repeatability.

III.2 Proposed Solution

This work aims to develop an automated solution for tracking cables in solar panel farm facilities by employing a collaborative approach between a UGV and a UAV. Fig. III.1 illustrates a detailed research flowchart of the proposed solution. The process begins with initializing the ROS framework, hardware components, and relevant algorithms. Following this, the UGV transports the UAV to a designated area within the solar farm. Once in position, the UAV launches and starts its aerial inspection. The UGV uses a CNN-based architecture to identify and follow the cables. This image is later used to identify possible structural damage. During the inspection, real-time data, such as battery level, mission status, error status, and global position, is exchanged between the GCS, the UAV, and the UGV. The GCS serves as the core of this communication; the UAV and UGV exclusively send and receive information to and from the GS. This real-time data exchange ensures synchronized operations. For example, if the UAV's battery level drops below a critical threshold, it sends a signal to the GCS requesting a landing location. The GCS then calculates a suitable landing position, plans a path for the UAV using the $A^\star$ method, and sends this route to the UAV. Simultaneously, the GCS sends these coordinates to the UGV, tasking it to navigate to the same point using the $A^\star$ method to recover the UAV. This coordinated procedure ensures minimal downtime and a seamless continuation of the inspection.

This work focuses on integrating a UGV with a UAV in a heterogeneous system for solar panel inspection. As the primary scope is the integration itself, this work will not specify the UAV's main tasks or decision-making logic.

The UGV employs a CNN-based architecture called YOLOv5 to detect and track cables. This computer vision capability supports ground-level inspections and provides accurate guidance for the UAV during its mission. By using the detected cable positions, the UGV can act as a dynamic anchor

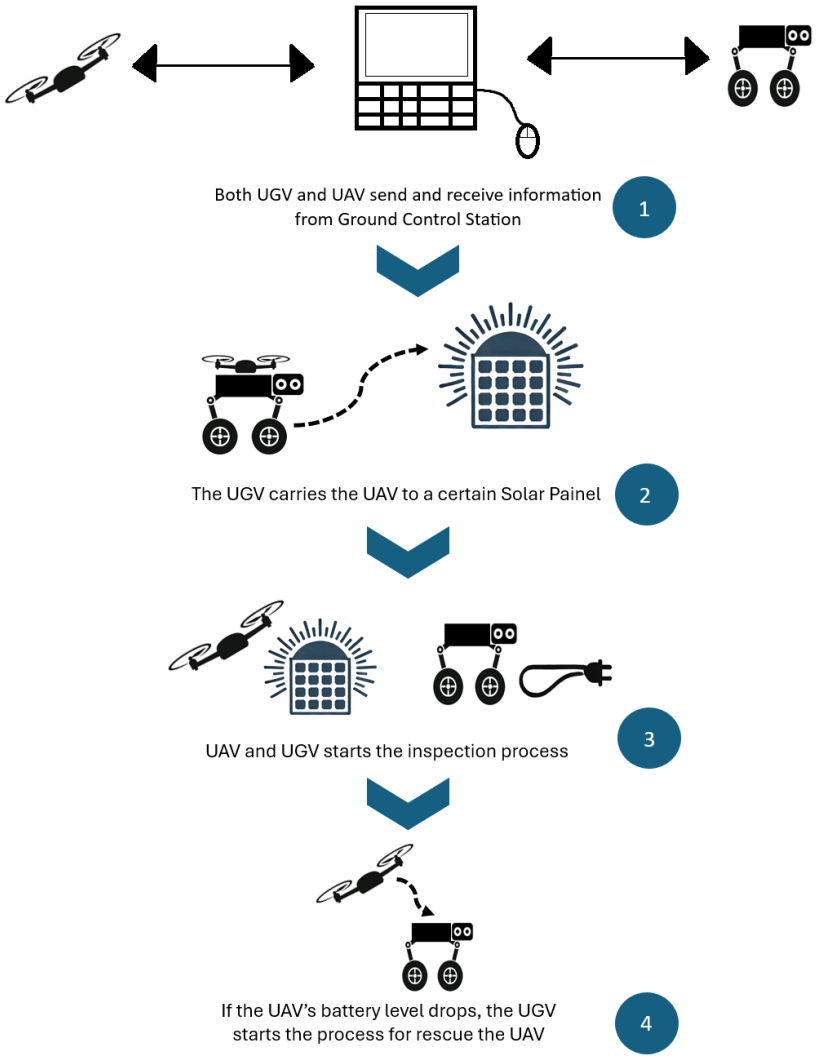


Figure III.2: Cooperative framework. Based on a paper published by the authors in the Journal of Intelligent & Robotic Systems (AMORIM et al., 2025b)

point, improving the UAVs operational efficiency.

As presented in Fig. III.2, the collaborative framework was designed to take advantage of the complementary strengths of UGVs and UAVs. The UGV serves as the base unit, responsible for transporting, rescuing the UAV, and conducting ground-level inspections, such as cable and infrastructure analysis. The UAV, in turn, is deployed for aerial inspections of solar panels, focusing on identifying surface-level issues like hotspots or cracks. In this sense, the inspection data from both platforms are collected and processed for further analysis, ensuring comprehensive coverage of the solar farm.

III.3 Solution Setup

The proposed collaborative framework between a UGV and a UAV was implemented using the ROS framework for system integration and Python for algorithm development and control logic. ROS served as the backbone of the solution setup. It provided essential tools and libraries for robot communication,

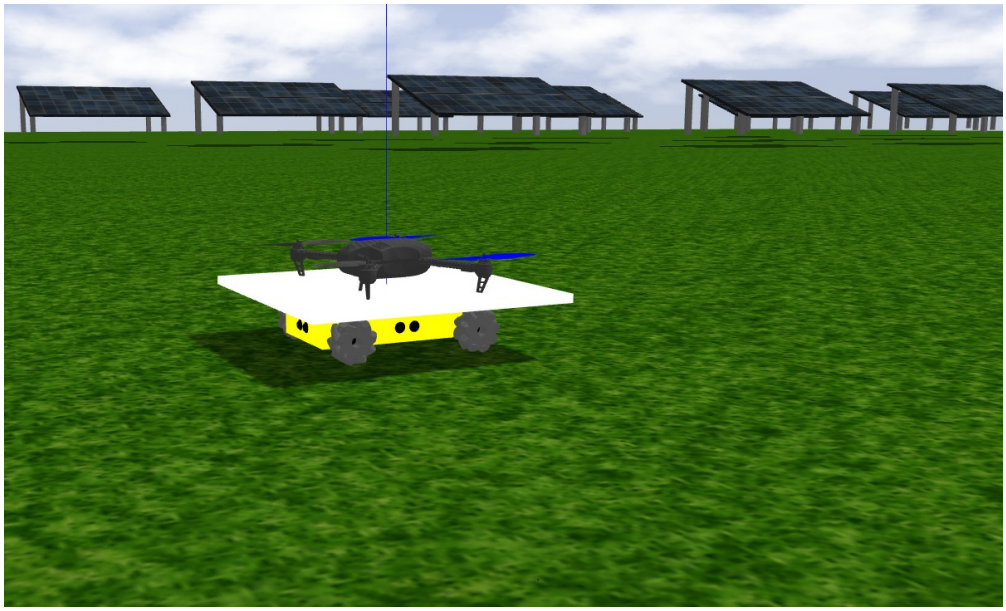
sensor data processing, and control (Stanford Artificial Intelligence Laboratory et al.,).

The GCS coordinates the UGV and the UAV operations in the proposed framework. The UAV used in this framework is the Iris model (3D Robotics, 2014), while the UGV is the Nexus robot. Communication between the GCS, the UAV, and the UGV is facilitated through a telemetry module on both the computer and the vehicles, utilizing the MAVLink communication protocol (MAVLink Contributors, 2022). This protocol ensures efficient and reliable data exchange between all system components. The PX4 software, which runs on the Pixhawk flight controller hardware, manages the UAV's flight operations. The PX4 processes data from the UAV's peripheral sensors, such as Global Positioning System Global Positioning System (GPS), Inertial Measurement System (IMU), and cameras, and adjusts actuator outputs for stable and precise flight. This setup enables the UAV to perform aerial inspections of solar panels while maintaining synchronization with the UGV.

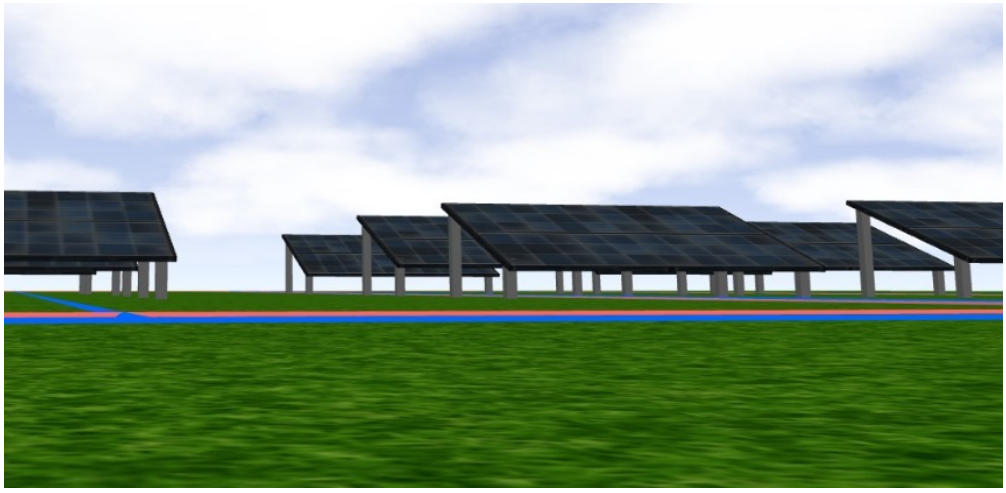
The Nexus UGV, acting as the ground unit, communicates with the GCS via ROS topics and messages. Equipped with cameras and sensors, the UGV uses the YOLOv5 model for cable detection and tracking. The GCS receives this processed information from the UGV and ensures it is shared with the UAV to guide its inspection mission. The GCS also manages real-time adjustments in the system, such as repositioning the UGV for UAV landing when the UAV's battery reaches a critical threshold.

The GCS runs an Intel i5-12450H processor, 32GB DDR4 memory, and an NVIDIA GeForce RTX 3050 Ti GPU with 4GB VRAM. The system runs on Ubuntu 20.04 and uses Gazebo to simulate the proposed operational environment and verify the approachs functionality as a proof of concept (KOENIG; HOWARD, 2004). The machine learning algorithms are implemented with PyTorch. Fig. III.3 shows the virtual environment created in Gazebo, where Fig. III.3(a) illustrates both robots, and Fig. III.3(b) depicts a solar farm with photovoltaic panels.

In the simulation environment, Gazebo generated camera images at a fixed rate of 30 Hz, and published them via ROS topics, replicating real-world inspection data. A Python script processed these images, using the CvBridge library to convert ROS messages into OpenCV-compatible arrays. These arrays were then fed into a YOLOv5 model pre-trained and fine-tuned to detect cables. The bounding boxes obtained frame-by-frame from the model were used by a separate script to guide the cable-following task in the solar farm environment. For the UGV's navigation, the *move_base* and *nav_msgs* ROS packages were utilized.



(a): Selected robots. Source: (AMORIM et al., 2025b)



(b): Solar panels. Source: (AMORIM et al., 2025b)

Figure III.3: World created in the Gazebo Software.

III.4 Computer Vision-based Tracking and Monitoring

This work utilizes YOLOv5 to improve the cable following strategy for the Nexus UGV, providing crucial support for the UAV in its collaborative inspection mission. YOLOv5 is a real-time object detection system renowned for its remarkable speed and accuracy, which makes it ideal for dynamic and challenging tasks (REDMON, 2016), (SILVA et al., 2022). The YOLOv5 architecture comprises a modular backbone for extracting image features, a neck for aggregating multi-scale features, and a detection head for predicting bounding boxes, object confidence scores, and class probabilities. It employs an anchor-based detection method, using predefined anchor boxes to guide the bounding-box regression process. This approach optimizes the detection of objects of varying sizes and aspect ratios, ensuring robust performance in diverse scenarios. The YOLOv5 pipeline is designed to provide efficient object

localization and detection, enabling real-time processing for the UGV during collaborative inspection tasks. The architecture of YOLOv5 is illustrated in Fig. III.4, highlighting the backbone, the neck, and the detection head.

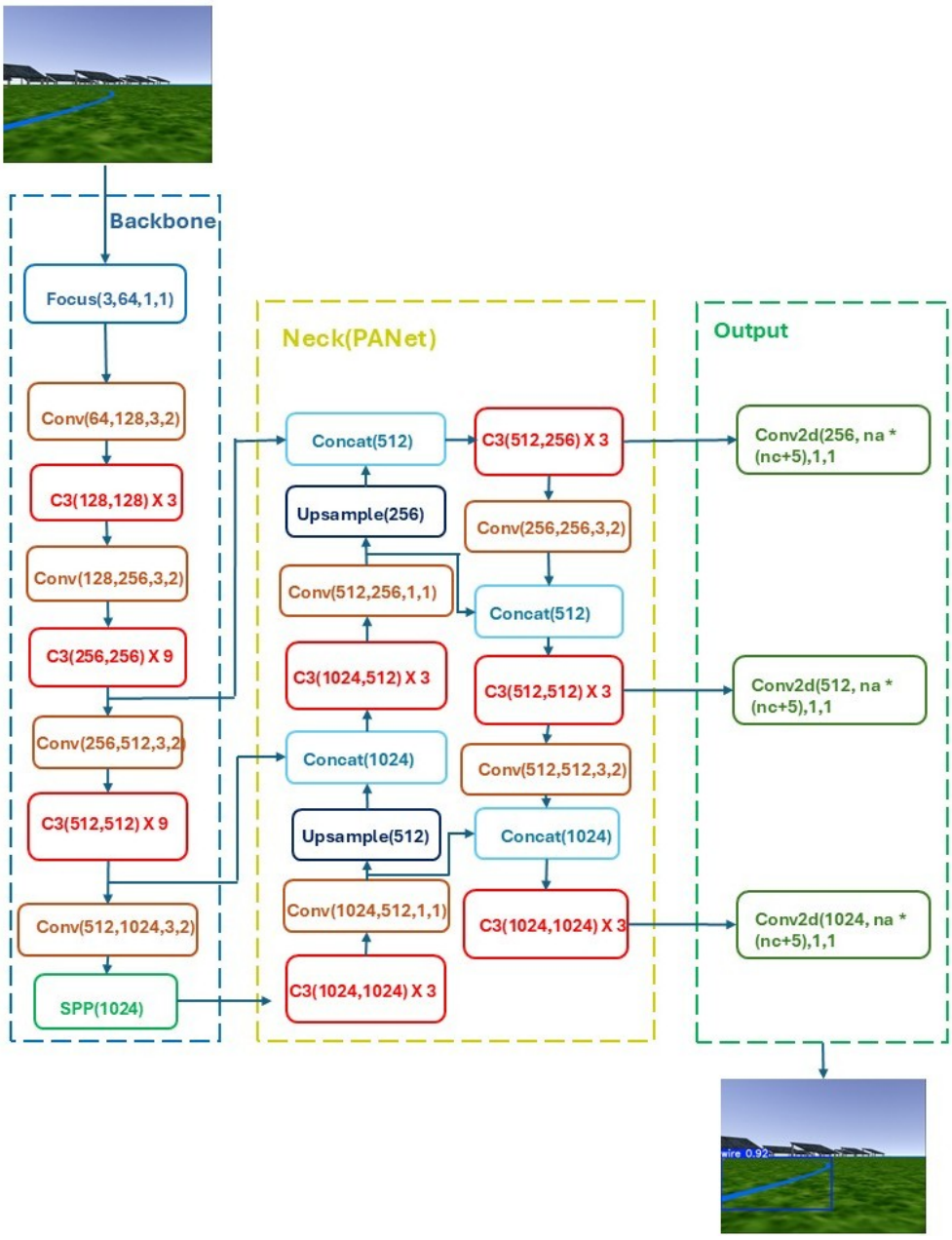


Figure III.4: YOLO structure for cable following purposes. Source: (AMORIM et al., 2025b)

The choice of YOLOv5 over other versions was driven by specific architectural and operational considerations aligned with the requirements of this research. YOLOv5 offers a modular and scalable architecture implemented natively in PyTorch, allowing for efficient integration with robotic frameworks like ROS. It supports model variants (s, m, l, x) that provide a trade-off between inference speed and accuracy, enabling deployment on devices with limited processing power, such as embedded systems and edge GPUs. Its backbone (CSPDarknet53), combined with PANet and SPP layers, ensures robust

feature extraction and multi-scale detection, which are crucial for tasks like cable detection in highly variable visual contexts.

The model was trained on a custom hybrid dataset comprising Real-Synthetic images, featuring real cables superimposed on artificial backgrounds generated via chroma keying (ZANELLA et al., 2021), and Simulated images rendered within the Gazebo environment. Both datasets can be seen on Figures III.6 and III.7. The dataset was partitioned into a training set (3,600 images: 2,700 real-synthetic, 900 simulated) and a validation set (400 images: 300 real-synthetic, 100 simulated). To improve model robustness, Simulated-Synthetic images were subjected to data augmentation, including flipping and affine transformations. The YOLOv5 architecture was initialized with pre-trained weights from coco and trained at a resolution of 480x480 pixels. To evaluate the generalization of the digital twin (Nexus 4WD) to real-world scenarios, two distinct test sets were utilized: 62 real-world images featuring authentic backgrounds (ZANELLA et al., 2021) on Figure III.8 and 30 Simulated-Synthetic images from Gazebo. A comprehensive analysis of the dataset is provided in V.1. In this implementation, the model facilitates UGV route correction by utilizing the detection bounding box to calculate a centroid, which serves as the feedback signal for autonomous navigation.

III.4.1 Application on a Digital Twin

Using a digital-twin introduces challenges in perfectly replicating real-world scenarios, creating a potential "reality gap". Therefore, it is necessary to evaluate the YOLOv5 model's performance under different training conditions. We must compare its effectiveness when trained only on the real-world images (which, in the chosen dataset, feature synthetic backgrounds) against its effectiveness when trained on a dataset with simulated images. This study will clarify the impact of simulated data on the model's cable detection capabilities in both real-world and simulated contexts.

This section details the tools, datasets, and neural network architecture employed throughout the experiments. The simulated synthetic datasets were generated using the Gazebo simulator in conjunction with the ROS platform. An omnidirectional mobile robot was programmed to follow a cable path inside a virtual environment resembling a solar panel farm. This setting enabled the generation of structured and repeatable visual data under controlled lighting and positional conditions, making it ideal for training and evaluation purposes. Figure III.5 presents the simulated world, built using the software Gazebo, along with the robot.

Training Methodology

In addition to the fully simulated images generated in Gazebo, a second dataset composed of real-synthetic images was employed. These images were created by compositing cropped target objects onto artificially generated backgrounds, producing photorealistic scenes that emulate real-world visual

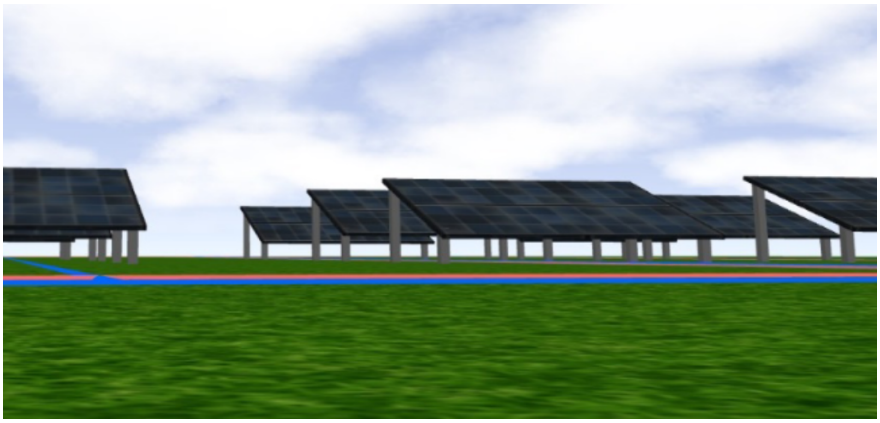


Figure III.5: Simulated world on software Gazebo. Source: (AMORIM et al., 2025b)

complexity while avoiding the need for manual annotation. To explore the influence of dataset composition on model performance, we created five training configurations: (i) 100% simulated data, (ii) 100% real-synthetic data, and three mixed configurations with 25%, 50%, and 75% real-synthetic data combined with the remaining portion of simulated images. All datasets were annotated consistently and used to train the YOLOv5m model under the same hyperparameters and training procedures, ensuring fair comparison across scenarios. Table III.1 summarizes the key hyperparameters adopted during training, including learning rate, batch size, optimizer settings, and augmentation techniques.

Table III.1: Main hyperparameters used in YOLOv5m training

Hyperparameter	**Value**
Model architecture	YOLOv5m
Number of epochs	200
Batch size	16
Learning rate	0.01
Momentum	0.937
Weight decay	0.0005
Optimizer	Stochastic Gradient Descent (SGD)
Input image size	480 $\times$ 480 pixels
Data augmentation	Flip, mosaic, blur, HSV shift
Learning rate scheduler	Cosine decay
Early stopping	Not applied
Number of classes	1 (cable existence)

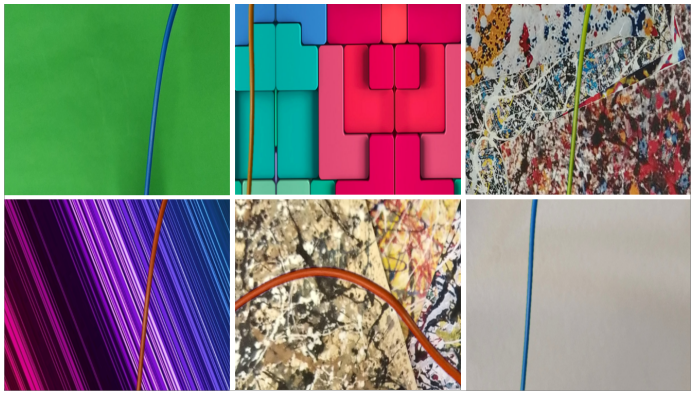


Figure III.6: Sample images of the dataset provided by (ZANELLA, 2023).

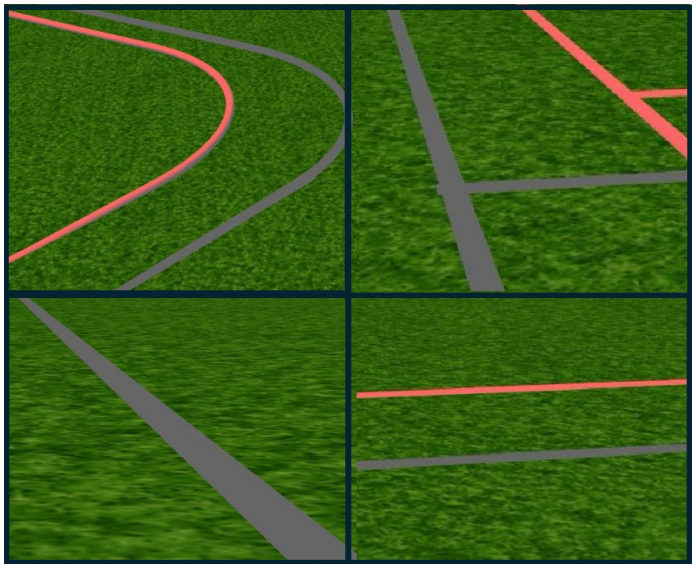


Figure III.7: Synthetic images incorporated in the dataset for training. Source: (AMORIM et al., 2025b)

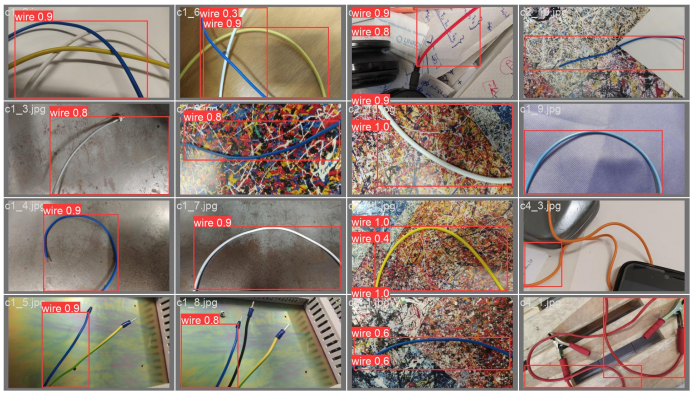


Figure III.8: Images used for real-world test from III.6

III.5 Navigation and Cooperation

The proposed framework for collaborative inspection between the Nexus UGV and the Iris UAV integrates advanced navigation and collaboration mechanisms to ensure efficiency and adaptability during operations. The framework depicted in Fig. III.9 represents the cognitive architecture implemented on

the UGV to facilitate autonomous decision-making, collaboration with the UAV, and interaction with the GCS through a Human-in-the-Loop (HITL) mechanism. This multi-layered cognitive structure ensures that the UGV can perform its tasks efficiently, adapt to dynamic environments, and communicate effectively with other system components, including human operators, when necessary.

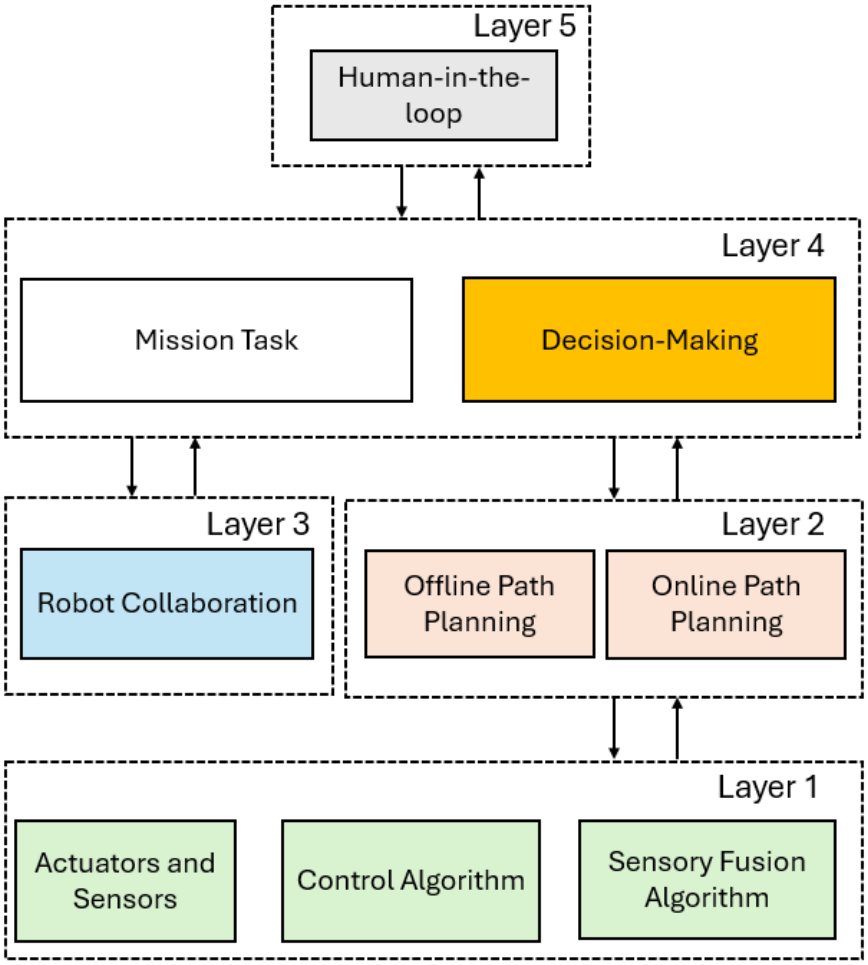


Figure III.9: UGV framework. Source: (AMORIM et al., 2025b)

The framework integrates sensory input and control capabilities to enable real-time perception and actuation. This layer incorporates sensors and actuators to collect environmental data and execute precise control commands. This is essential for tasks where real-time precision is paramount, such as cable tracking and navigation. Building upon the sensory and control foundation, the path-planning layer enables both pre-defined and dynamic navigation. Offline path planning establishes routes based on prior knowledge of the inspection environment, ensuring systematic coverage of the solar farm. Concurrently, online path planning adapts the UGVs trajectory in response to real-time changes in the environment, such as unexpected obstacles or requests from the UAV.

The framework also incorporates a HITL layer, which provides an interface for human operators to intervene when necessary. This layer facilitates bidirectional communication, allowing the UGV to relay status updates and critical information to the GCS while receiving commands or adjustments from

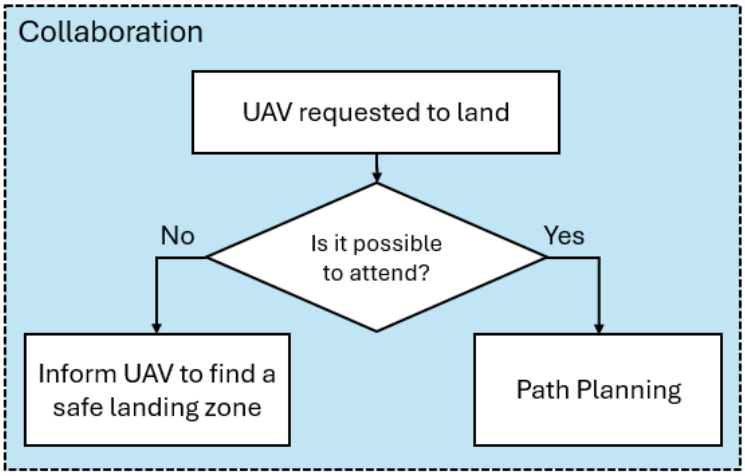


Figure III.10: Decision-making process when the UAV requests to land. Source: (AMORIM et al., 2025b)

human operators. At a higher level, the mission-level decision-making layer governs the overall execution of the task. This centralized module prioritizes tasks, evaluates system constraints, and makes decisions based on inputs from sensory data, collaborative needs, and potential human oversight. It ensures that the UGVs actions align with mission objectives while maintaining flexibility to adapt to changing conditions. For example, if the UAV signals a critical low-battery status, this layer can prioritize path planning for a safe landing zone.

The robot collaboration layer facilitates synchronized operations between the UGV and UAV. This layer manages shared tasks such as cable tracking and inspection, while addressing UAV-specific requirements, like assisting with safe landings. Collaboration is achieved through real-time communication, which is enabled by the MAVLink protocol. Fig. III.10 illustrates the decision-making process when the UAV requests land. The system evaluates whether the UGV can accommodate the request based on its current state and environmental constraints. If the request is feasible, the UGV initiates a path-planning module to determine the most efficient route to a safe landing zone via the GCS. The $A^\star$ algorithm (HART; NILSSON; RAPHAEL, 1968), employed for path planning, ensures collision-free and efficient navigation toward the UAV's location.

The proposed framework follows a layered architecture, ensuring modular and efficient decision-making. Algorithm 1 presents the integration of the computational techniques to enable autonomous navigation and task execution. At Layer 3, the UGV collaborates with the UAV, particularly when supporting landing requests. If a UAV requests landing assistance, the system evaluates the feasibility (i.e., Decision-Making block at Layer 4) and takes the UAV's position as a destination point (i.e., Mission Task block at Layer 4). At Layer 2, the UGV performs offline and online path planning, utilizing the $A^\star$ algorithm to determine optimal routes based on environmental constraints and mission objectives. Once the path is computed, the system transitions to Layer 1, where control algorithms and sensor fusion

Algorithm 1 Sequential Robot Behavior with Rescue Interrupt

Input: Robot initial position, mission targets, base station coordinates
Output: Mission completion or successful UAV rescue

begin
  Event: On UAV Rescue Request Received begin
    Interrupt current phase Verify if the path to the UAV coordinates is feasible **if** *Feasible* **then**
      Proceed to **Phase 2** (Navigate to landing point);
      Proceed to **Phase 3** (Execute rescue trajectory);
      Jump to Phase 5 (Direct return to base)
  Phase 1: Transporting UAV to Inspection Site;
  Calculate path using $A^\star$ to the UAV launch point;
  Execute trajectory until launch site is reached (distance < 0.1);
  Phase 2: Navigating to UGV Inspection Point;
  Read the score map and define the robot's inspection destination;
  Execute $A^\star$ algorithm to find the optimal path considering map objects;
  Simplify the obtained path;
  Phase 3: Robot Path Execution;
  Read all sensors to obtain world frame position and speed;
  Use the controller to stabilize the robot along the trajectory;
  while *distance to target* ≥ 0.1 **do**
    Maintain path stabilization via control loop;
  Stop the robot near the target region;
  Phase 4: Cable Following Task (YOLOv5);
  Start YOLOv5 and disable the standard trajectory controller;
  while *Cable detected AND no interruption* **do**
    Extract bounding box (BBox) size and central coordinates;
    if *BBox center is offset from camera center* **then**
      offset = gain $\times$ BBox_width;
      setpoint = pixel_center_of_image $-$ offset;
      PV = x-coordinate of BBox center;
      Use Proportional-Integrative (PI) controller to follow the cable at the offset distance;
  Phase 5: Return to Base;
  Define base station as the new destination;
  Execute $A^\star$ to calculate the return trajectory;
  Execute path stabilization until base is reached (distance < 0.1);

techniques guide real-time path execution, stabilizing the robot along the predefined trajectory. Finally, the algorithm incorporates a cable-following task using the YOLOv5 model within Layer 1, where sensory fusion and control algorithms operate. This enables the robot to detect and follow cables autonomously by processing real-time visual inputs and generating actuator commands.

III.6 Robot kinematic

An omnidirectional robot is a holonomic system (HENDZEL; RYKAŁA, 2017) defined by its ability to independently control all three planar degrees of freedom (x, y, α). The configuration addressed in this work achieves this capability through a four-wheel drive mechanism, as depicted in Figure III.11 (TAHERI; QIAO; GHAEMINEZHAD, 2015). S_X is the distance between the robot's center (O) and the wheel's axis. S_Y is half the distance between the wheel centers on a single axle. Each O_i $(i = 1, 2, 3, 4)$ denotes the center of each wheel, and θ is the angle of the wheel's rollers relative to its plane of rotation.

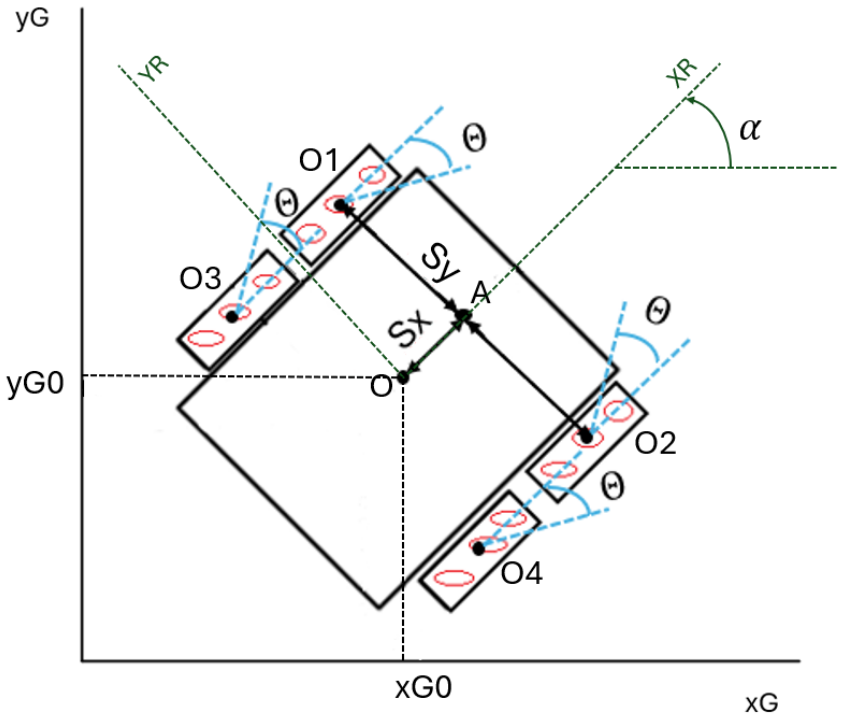


Figure III.11: Robot axis and Global axis representation. Source: Author

This holonomic definition is crucial for controller development, as it dictates the specific kinematic equations governing the robot's movement. Therefore, this section will detail the equations for this four-wheeled platform. Notably, this type of robot can use the behavior of a differential robot (i.e., rotating on its own axis and moving along its x-axis). However, by using omnidirectional wheels - in this work, Mecanum wheels were used - it also gains the ability to move laterally along its y-axis (TAHERI; QIAO; GHAEMINEZHAD, 2015). Due to this expanded mobility, the robot's movement can be described by Equations III.1.

$$\begin{bmatrix} V_X \\ V_Y \\ \dot{\alpha} \end{bmatrix} = \frac{R+r}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 \\ -\frac{1}{S_X+S_Y} & \frac{1}{S_X+S_Y} & -\frac{1}{S_X+S_Y} & \frac{1}{S_X+S_Y} \end{bmatrix} \begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \\ \dot{\phi}_4 \end{bmatrix} \tag{III.1}$$

V_x is the velocity on the robot's x-axis and V_y is the velocity on its y-axis. $\dot{\alpha}$ is the angular velocity, and $\dot{\phi}_i$ is the angular velocity of each wheel i. R is the wheel's radius and r is the roller's radius. The S_X and S_Y are the distances between the robot's center (O) and the center of the wheel axis (A) on the x- and y-axes. With these definitions, it is possible to obtain the desired wheel angular velocities from the velocities of these three axes. The inverse kinematics can be described in Equation III.2.

$$\begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \\ \dot{\phi}_4 \end{bmatrix} = \frac{1}{(R+r)} \begin{bmatrix} 1 & -1 & -(S_X+S_Y) \\ 1 & 1 & (S_X+S_Y) \\ 1 & 1 & -(S_X+S_Y) \\ 1 & -1 & (S_X+S_Y) \end{bmatrix} \begin{bmatrix} V_X \\ V_Y \\ \dot{\alpha} \end{bmatrix} \tag{III.2}$$

These equations can be related in the following way:

$$\dot{\boldsymbol{\phi}} = \boldsymbol{J} \cdot \boldsymbol{V} \tag{III.3}$$

where

$$\dot{\boldsymbol{\phi}} = \begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \\ \dot{\phi}_4 \end{bmatrix},$$

$$\boldsymbol{J} = \begin{bmatrix} \frac{1}{(R+r)} & \frac{-1}{(R+r)} & \frac{-(S_x+S_y)}{(R+r)} \\ \frac{1}{(R+r)} & \frac{1}{(R+r)} & \frac{(S_x+S_y)}{(R+r)} \\ \frac{1}{(R+r)} & \frac{1}{(R+r)} & \frac{-(S_x+S_y)}{(R+r)} \\ \frac{1}{(R+r)} & \frac{-1}{(R+r)} & \frac{(S_x+S_y)}{(R+r)} \end{bmatrix},$$

$$\boldsymbol{V} = \begin{bmatrix} V_x \\ V_Y \\ \dot{\alpha} \end{bmatrix}.$$

The kinematic equations in the global frame can be described below using the Rotation Matrix in Equations III.4 – III.6.

$$\begin{bmatrix} \dot{x}_G \\ \dot{y}_G \\ \dot{\alpha} \end{bmatrix} = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} V_X \\ V_Y \\ \dot{\alpha} \end{bmatrix} \tag{III.4}$$

$$\begin{bmatrix} \dot{x}_G \\ \dot{y}_G \\ \dot{\alpha} \end{bmatrix} = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \boldsymbol{J_o d} \cdot \dot{\boldsymbol{\phi}} \tag{III.5}$$

$$\begin{bmatrix} \dot{x}_G \\ \dot{y}_G \\ \dot{\alpha} \end{bmatrix} = \frac{(R+r)}{4} \cdot \begin{bmatrix} \cos(\alpha)+\sin(\alpha) & \cos(\alpha)-\sin(\alpha) & \cos(\alpha)-\sin(\alpha) & \cos(\alpha)+\sin(\alpha) \\ \sin(\alpha)-\cos(\alpha) & \sin(\alpha)+\cos(\alpha) & \sin(\alpha)+\cos(\alpha) & \sin(\alpha)-\cos(\alpha) \\ \frac{-1}{(S_X+S_Y)} & \frac{1}{(S_X+S_Y)} & \frac{-1}{(S_X+S_Y)} & \frac{1}{(S_X+S_Y)} \end{bmatrix} \cdot \begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \\ \dot{\phi}_4 \end{bmatrix} \tag{III.6}$$

III.7 Robot dynamics

This section demonstrates the robot's dynamics due to its holonomic behavior. These parameters are important in developing the position controller.

The dynamic model is based on (LIN; SHIH, 2013) that describes step by step how to obtain the dynamic model. The Equation III.7 describes the model of a holonomic robot, where $\mathbf{q} \in \mathbb{R}^{4\times 1}$ denotes the vector of generalized coordinates corresponding to the wheel angular positions.

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) \cdot sign(\dot{q}) + \tau_d = \frac{1}{R+r} \cdot B^T \tau \tag{III.7}$$

Where τ is the torque generated by mecanum wheels, $M(q) = \begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & (I_b + I_w) \end{bmatrix}$ is the inertia matrix, $C(q,\dot{q}) = diag(B_x, B_y, B_w)$ is the coefficient matrix of the viscous friction force, $G(q) = diag(C_x, C_y, C_w)$ is the Coulomb Force. $\boldsymbol{\tau_d}$ is the uncertain disturbance vector. Finally, B is the parameter matrx for the control signal vector described on Equation III.8 and q is the postion on the robot's axis (x, y, w). The parameters used in the dynamic model are described in Table III.2.

$$B = \begin{pmatrix} \frac{\sqrt{2}}{2r} & \frac{\sqrt{2}}{2r} & \frac{\sqrt{2}}{2r} & \frac{\sqrt{2}}{2r} \\ -\frac{\sqrt{2}}{2r} & \frac{\sqrt{2}}{2r} & \frac{\sqrt{2}}{2r} & -\frac{\sqrt{2}}{2r} \\ -\frac{l}{r} & \frac{l}{r} & -\frac{l}{r} & \frac{l}{r} \end{pmatrix} \tag{III.8}$$

With the equation described above and with all these parameters, a controller for navigation will be implemented to be able to adapt to disturbances. This method is well known as ADRC.

Table III.2: Dynamic model parameters

Parameter	Description
I_b	body Intertia (kg $\cdot m^2$)
I_w	Wheels Intertia (kg $\cdot m^2$)
l	diagonal distance between robot's center to wheels' center (m)

III.8 ADRC

This section is fully dedicated to describing Active Disturbance Rejection Control (ADRC), its importance, and the methodology for its development for a holonomic robot. ADRC presents a robust alternative paradigm to traditional control methods, which are often heavily dependent on the accuracy of the plant model (GAO; HUANG; HAN, 2001). As articulated by Gao, Huang, and Han, the fundamental philosophy of ADRC is to treat all system uncertainties, including external disturbances, unmodeled internal dynamics, and parameter variations, as a single, lumped "total disturbance." This total disturbance is then actively estimated in real-time using the framework's key innovation: the Extended State Observer (Extended State Observer (ESO)). The control law subsequently utilizes this real-time estimation to proactively cancel the disturbance's effect, effectively simplifying the complex plant to a canonical cascaded-integrator form. This approach yields a controller that is inherently robust to uncertainties and requires minimal *a priori* knowledge of the system, offering a practical solution for complex non-linear systems. Following this foundational description, the specific implementation for the holonomic platform will be detailed.

To implement this strategy, it is first necessary to define the control inputs and system outputs. In the context of navigation, the configuration vector q serves as the system output, while the torque vector τ, generated by each wheel, acts as the control input. Based on these definitions, Equation III.7 can be manipulated to isolate the acceleration term $\ddot{q}$, yielding Equation III.9 below.

$$\ddot{q} = M^{-1} \cdot \left(\frac{1}{R+r}\right) \cdot B^T \cdot \tau + M^{-1} \cdot [-(C \cdot \dot{q} + B^T \cdot S \cdot f)] \tag{III.9}$$

In the ADRC framework, the complexity of the dynamic model, such as parameter variations influenced by friction and other non-linearities, is treated as a generalized disturbance. This disturbance is estimated in real time by the ESO to simplify controller design. Consequently, the complex dynamic equation can be reformulated into the canonical form shown in Equation III.10, where $d(t)$ represents the total estimated disturbance (both internal and external) and b_0 represents the control gain matrix approximation derived from the term $M^{-1} \cdot (\frac{1}{R+r}) \cdot B^T$.

$$\ddot{q} = \boldsymbol{b_0} \cdot \boldsymbol{\tau} + \boldsymbol{d(t)} \tag{III.10}$$

Where $\boldsymbol{b_0}$ is a constant Matrix. The state space of this simplified equation is described in Equation

III.11. This Equation is necessary to implement the ESO on the robot system, which consists of a microcontroller, if it is a real robot, or a computer, if it is a simulated robot. After that, the ESO is obtained following these steps:

1. Define the Nominal State: Define the state vector $\mathbf{x}$ from the established dynamics (e.g., $\mathbf{x} = [q, \dot{q}]^T$).
2. Identify Extended State: Identify and define the total lumped uncertainty, $d(t)$, as a new, extended state variable, x_{n+1}.
3. Augment the Vector: Form the augmented state vector that combines the nominal states and the disturbance, $\mathbf{X} = [\mathbf{X}^T, x_{n+1}]^T$.
4. Formulate Extended Dynamics: Rewrite the system dynamics to obtain the complete extended state dynamics $\dot{\mathbf{X}} = A_e\mathbf{X} + B_e u + E_e d(t)$, introducing the assumption that the derivative of the disturbance, $\dot{x}_{n+1}$, is a bounded unknown function $d(t)$.
5. Define Observer Structure: Define the ESO structure based on the Luenberger observer form (ZEITZ, 1987), $\dot{\hat{\mathbf{X}}} = A_e\hat{\mathbf{X}} + B_e u + L(y - C\hat{\mathbf{X}})$, where L is the observer gain matrix used to estimate the extended states.

Following these steps, the estimated disturbance ($\hat{x}_{n+1}$) is ready to be utilized directly in the control law for active rejection. The Extended Dynamics model, defined by Equations III.11 and III.12, serves as the foundation for the observer design, explicitly coupling the control input $u(t)$ and the total disturbance $d(t)$ to the state derivative $\dot{\mathbf{X}}$.

The ESO structure is obtained by applying the Luenberger observer form to these Extended Dynamics. By introducing the observer gain matrix L and the estimation error feedback term $L(y(t) - C\hat{\mathbf{X}})$, the initial representation of the ESO is given by Equation III.13. Subsequently, this equation is algebraically simplified by fully incorporating the estimation error into the dynamic matrix. This is achieved through the operation $\mathbf{A}_{ESO} = (A_e - LC_e)$, which isolates the final implemented form of the observer shown in Equation III.14.

$$\left[\dot{X}\right] = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \cdot \left[X\right] + \begin{bmatrix} 0 \\ b_0 \\ 0 \end{bmatrix} \cdot u(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \cdot d(t) \tag{III.11}$$

$$y(t) = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \cdot \left[X\right] \tag{III.12}$$

$$\left[\dot{\hat{X}}\right] = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \cdot \left[\hat{X}\right] + \begin{bmatrix} 0 \\ b_0 \\ 0 \end{bmatrix} \cdot u(t) + \begin{bmatrix} L1 \\ L2 \\ L3 \end{bmatrix} \cdot (y(t) - \hat{X}) \tag{III.13}$$

$$\left[\dot{\hat{X}}\right] = \begin{bmatrix} -L1 & 1 & 0 \\ -L2 & 0 & 1 \\ -L3 & 0 & 0 \end{bmatrix} \cdot \left[\hat{X}\right] + \begin{bmatrix} 0 \\ b_0 \\ 0 \end{bmatrix} \cdot u(t) + \begin{bmatrix} L1 \\ L2 \\ L3 \end{bmatrix} \cdot y(t) \tag{III.14}$$

The next step after obtaining the ESO state space is to define the Control Law ($u(t)$), which executes the core function of ADRC: combining tracking performance with active disturbance cancellation.The control law is structured as a feedback loop around the ESO, performing two primary functions: Disturbance Cancellation and Tracking Error, as defined in Equations III.15 and III.16.

Equation III.15 performs Disturbance Cancellation, where the total estimated disturbance ($\hat{d}(t)$) is subtracted from the nominal control input ($u_0(t)$) and scaled by the approximate control gain (b_0). This term effectively linearizes the complex plant dynamics.

Equation III.16 defines the nominal input $u_0(t)$ using a Proportional-Derivative (PD) structure for Tracking Error, comparing the desired reference (q_r) with the estimated states ($\hat{y}$ and $\hat{\dot{y}}$).

The control gains K_p and K_D are typically tuned using the Control Bandwidth Method (or Bandwidth Parameterization). This method simplifies the tuning process by relating all necessary gains to a single parameter: the desired control bandwidth (ω_0 or ω_c).The standard relationship for tuning the PD control law in ADRC for a 2nd-order system is:

$$K_p = \omega_0^2 \quad \text{and} \quad K_D = 2 \cdot \omega_0$$

The control bandwidth (ω_0) is selected based on the required settling time (T_s). The common rule of thumb for estimating the required bandwidth is:

$$\omega_0 \approx \frac{4}{T_s}$$

$$\boldsymbol{u(t) = \frac{u_0(t) - \hat{d}(t)}{b_0}} \tag{III.15}$$

$$\boldsymbol{u_0(t) = K_p \cdot (q_r(t) - \hat{y}(t)) - K_D \cdot \hat{\dot{y}}(t)} \tag{III.16}$$

The final step in observer design is to determine the observer gain vector L by analyzing the

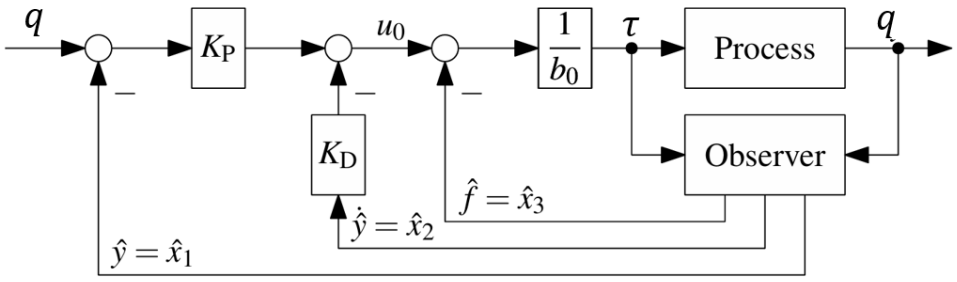


Figure III.12: ADRC diagram for navigation purposes. Adapted from (HERBST, 2013)

characteristic polynomial of the observer matrix $(A_e - LC_e)$. Similar to the control law, the observer poles must be placed at a stable location, $S^{ESO} = \omega_0^{ESO}$.

By selecting the desired characteristic polynomial to be critically damped and stable $\left((S + \omega_0^{ESO})^3\right)$, the resulting polynomial is matched against the algebraic determinant of the observer matrix, as shown in Equation III.8:

$$\det(\boldsymbol{S}\cdot\boldsymbol{I}-(\boldsymbol{A_e}-\boldsymbol{L}\cdot\boldsymbol{C_e})) = S^3+L_1S^2+L_2S+L_3 = (S+\omega_0^{ESO})^3 = S^3+3\cdot\omega_0^{ESO}S^2+3\cdot(\omega_0^{ESO})^2S+(\omega_0^{ESO})^3$$

By matching the coefficients of S in the expanded form of the stable characteristic polynomial, the observer gains are uniquely determined:

$$L_1 = 3 \cdot \omega_0^{ESO}, \quad L_2 = 3 \cdot (\omega_0^{ESO})^2, \quad L_3 = (\omega_0^{ESO})^3$$

This method ensures the observer is stable and that its bandwidth (ω_0^{ESO}) is often set higher than the control bandwidth (ω_0) to ensure timely disturbance estimation. Commonly ω_0 is between $3 \cdot \omega_0$ and $10 \cdot \omega_0$.

The overall control structure, integrating this PD controller with the disturbance estimation and cancellation features of the ESO, is visualized in the block diagram in Figure III.12.

Chapter IV Results of ADRC

IV.1 Developed Robot Performance using ADRC compared to PID

To evaluate the performance and robustness of the proposed control architectures, a numerical simulation was conducted using a four-wheeled omnidirectional mobile robot model. The simulation environment was designed to mimic the adverse conditions of a solar farm, incorporating time-varying external disturbances and model uncertainties.

IV.1.1 Testing Setup

The robot's mission consists of following a square trajectory defined by four waypoints: $(10, 0)$, $(10, 10)$, $(0, 10)$, and returning to the origin $(0, 0)$. Additionally, the robot undergoes orientation changes at specific nodes ($\alpha = [0, \pi/2, \pi, 0]$ rad).

To simulate real-world challenges such as wind gusts or uneven terrain, the following disturbances were injected into the x, y, and α coordinates:

- **Step Disturbances:** Sudden changes in force to test the recovery time.
- **Sinusoidal Noise:** Periodic perturbations to evaluate the steady-state error rejection.

The maximum torque for each wheel was saturated at 1.0 Nm to respect the physical limits of the actuators.

IV.1.2 Active Disturbance Rejection Control (ADRC)

The ADRC strategy is implemented by modeling each degree of freedom as a second-order system. The core of this approach is the Linear Extended State Observer (LESO), which treats all uncertainties and external forces as a "total disturbance" to be estimated and rejected in real-time.

The ESO is formulated as follows:

$$\begin{cases} \dot{\hat{x}}_1 = \hat{x}_2 + L_1(x_1 - \hat{x}_1) \\ \dot{\hat{x}}_2 = \hat{x}_3 + b_0 u + L_2(x_1 - \hat{x}_1) \\ \dot{\hat{x}}_3 = L_3(x_1 - \hat{x}_1) \end{cases} \tag{IV.1}$$

where $\hat{x}_1, \hat{x}_2$ are the estimated position and velocity, and $\hat{x}_3$ represents the estimated total disturbance. The control law is then defined as:

$$u = \frac{u_0 - \hat{x}_3}{b_0} \tag{IV.2}$$

with the nominal control input $u_0 = K_p(r - \hat{x}_1) - K_d\hat{x}_2$, where r is the reference trajectory.

IV.1.3 Proportional-Integrative-Derivative (PID) Control

As a baseline for comparison, a conventional PD controller was implemented. In the simulation, the integral term is ($K_i = 0.0001$), resulting in the following control law:

$$u_{PD} = K_{p_pi} \cdot e(t) + K_{d_pi} \cdot \dot{e}(t) + K_{i_pi} \cdot \int e(t)dt \tag{IV.3}$$

where $e(t)$ is the tracking error. For the orientation α, the error is normalized to the $[-\pi, \pi]$ range to ensure the shortest angular path.

IV.1.4 Controller Parameters

The parameters used for both controllers, derived from the simulation script, are summarized in Table IV.1.

Table IV.1: System and Controller Parameters

Parameter	Description	Value
r	Wheel radius	0.05 m
m	Total mass	2.0 kg
I_z	Rotational inertia	0.08 kg.m^2
u_{max}	Maximum torque per wheel	1.0 Nm
ω_0	Controller bandwidth	2.0 rad/s
K_p, K_d	ADRC proportional/derivative gains	4.0
L_1, L_2, L_3	ESO gains (Translational)	$37.92, 479.3, 2019.2$
$L_{1\theta}, L_{2\theta}, L_{3\theta}$	ESO gains (Rotational)	$36.0, 432.0, 1728.0$
K_{p_pi}	PID proportional gain	4.0
K_{d_pi}	PID integratve gain	0.0001
K_{d_pi}	PID derivative gain	4.0

The primary objective of applying ADRC was to mitigate the effects of external disturbances inherent in external environments. This section presents a comparative analysis with a conventional PID controller and evaluates key performance metrics. The comparative analysis was performed using a Python simulation script that incorporates a simplified state-space representation of the robot's nonlinear dynamic model. The specific physical parameters of the robot, as defined in Table IV.1, were essential inputs for this dynamic model. Both the ADRC and PID controllers were implemented within this simulated plant environment.

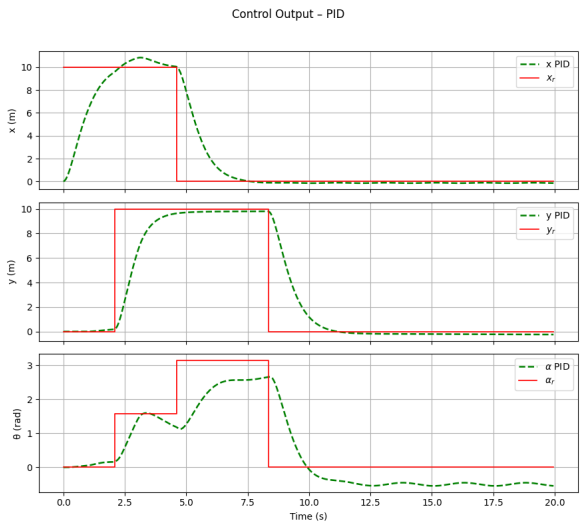
Figure IV.1(a) illustrates the system's output under PID control. It is important to emphasize that the K_p, K_D, and K_i gains for the PID controller were intentionally kept the same as the gains derived for the ADRC control law to ensure a comparative basis. Figure IV.1(b) shows the system's tracking performance using the ADRC methodology. A direct visual comparison of these two figures immediately reveals differences in performance between ADRC and PID, particularly during aggressive transitions and under the influence of disturbance. Although the same mission profile is considered, the reference

signals shown in the figures differ because axis-dependent reference trajectories are used. Each degree of freedom (x, y, α) is commanded independently, with distinct activation times and amplitudes reflecting the sequential execution of the maneuver.

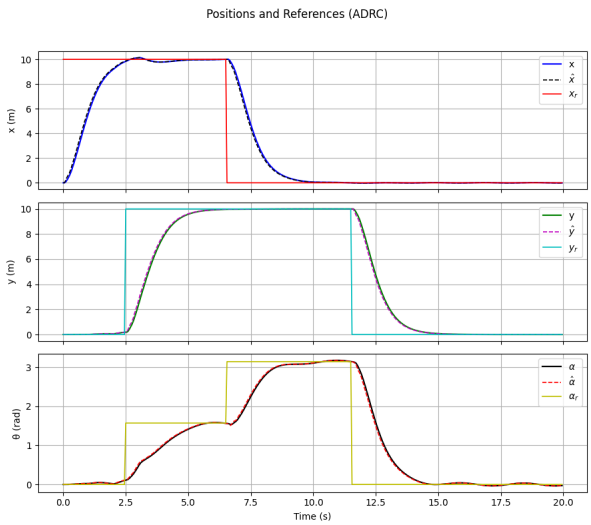
1. **Tracking Fidelity and Overshoot (ADRC Superiority):** The PID controller exhibits significant drawbacks in its transient response. For the initial steps (e.g., x_r and y_r transitions around $t = 2.5$ s), the PID experiences substantial overshoot (e.g., reaching 10.5 m for a 10.0 m reference) and a slower convergence time. In sharp contrast, the ADRC methodology achieves highly accurate and fast tracking of all references (x_r, y_r, α_r) with negligible overshoot and rapid convergence to the steady state.

2. **Rotational Control (α Response):** The most critical difference is observed in the rotational control. The PID controller shows a highly oscillatory and erratic transient response for α between $t = 2.5$ s and $t = 5$ s, indicating difficulty in managing coupled dynamics and external disturbances. After the final step-down (around $t = 9.5$ s), the PID exhibits a sustained offset and low-frequency oscillation in the steady state, failing to return precisely to 0 rad. Conversely, the ADRC achieves smooth, highly damped tracking of the α reference throughout all transitions, settling quickly and accurately to the zero position.

3. **Disturbance Mitigation:** The ADRC's intrinsic ability to proactively mitigate external and internal disturbances, estimated by the $\hat{x}$, $\hat{y}$, and $\hat{\alpha}$ states, ensures reliable tracking even during aggressive transitions. This result clearly confirms that the ADRC design effectively linearizes the perturbed, nonlinear plant, whereas the PID controller's reliance solely on error feedback leads to pronounced performance degradation.



(a): Output with PID control.



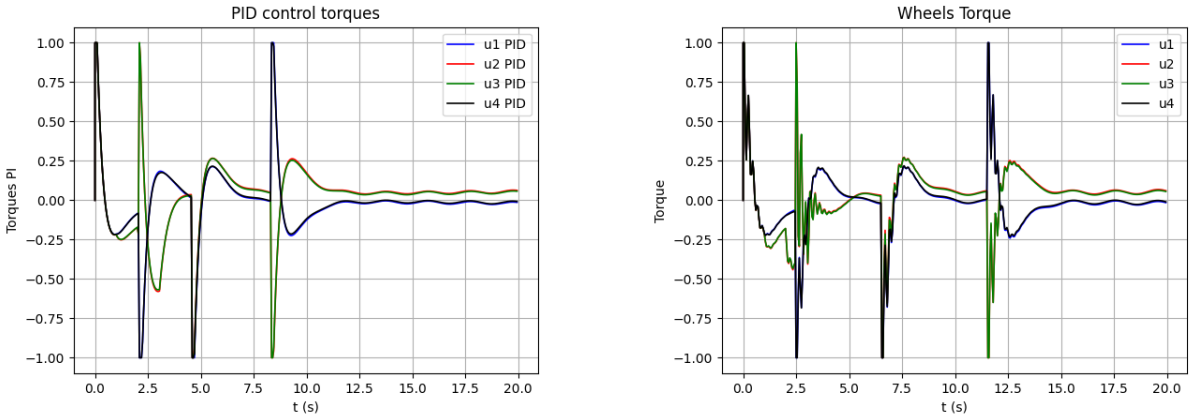
(b): Output with ADRC.

Figure IV.1: Comparison of Position Reference X System's output using different controllers. Source: Author

The comparison of control torques highlights a crucial distinction in efficiency, despite superficial similarities in magnitude. Figure IV.2(a) illustrates the control torques (u_1 to u_4) generated by the PID controller. These signals exhibit large initial spikes (saturating near ± 1.0) during aggressive maneuvers ($t < 10$ s). While the PID torque settles quickly after $t = 10$ s, this high control effort is directly responsible for the PID's significant overshoot, prolonged settling time, and pronounced oscillatory behavior observed in the positional outputs (Figure IV.1(a)), particularly in the rotational state (α).

In contrast, the ADRC torque outputs, shown in Figure IV.2(b), display a similar initial profile (also saturating near ± 1.0) and rapid settling time. However, the crucial difference lies in the outcome: the ADRC utilizes this transient control effort to achieve near-perfect tracking with minimal overshoot and immediate damping (Figure IV.1(b)). This observation confirms that, while both controllers demand comparable peak torque, the ADRC's core strength lies in its active compensation feature. By effectively canceling the estimated disturbance ($\hat{d}$), the ADRC translates the commanded torque (τ) into highly precise motion, achieving superior tracking quality and reduced transient error, a performance level unattainable by the passive, error-dependent PID controller under the same disturbed conditions.

As observed in Figure IV.2(b), the response obtained with the ADRC presents a slower transient behavior and increased oscillations when compared to the PID controller. This behavior is mainly due to the higher effective damping and disturbance-compensation mechanisms embedded in the ADRC structure. Unlike the PID controller, which prioritizes fast convergence to the reference, the ADRC actively estimates and compensates for total disturbances through the ESO. As a result, the control action becomes more conservative, leading to a longer settling time before the system proceeds to the next reference point. Note that the observed oscillations are associated with the continuous compensation of measurement noise and external disturbances by the ESO. This dynamic adjustment



(a): Control effort (torques) for PID.

(b): Control effort (torques) for ADRC.

Figure IV.2: Comparison of control effort (wheel torques) generated by PID and ADRC strategies. Source: Author

enhances robustness and tracking accuracy, particularly in position and orientation, but introduces small oscillatory behavior during the transient phase.

Figure IV.3 presents the validation of the Extended State Observer (ESO) by comparing the actual total disturbances (d_x, d_y, d_α) with their corresponding estimates ($\hat{d}_x$, $\hat{d}_y$, $\hat{d}_\alpha$). During the initial aggressive transient period (approximately $t < 4$ s), the ESO demonstrates rapid convergence, successfully tracking the large-amplitude spikes (reaching up to 5 m/s^2 on the d_x axis and 12 m/s^2 on the d_y axis) induced by the fast maneuvering and the applied external disturbance. Crucially, in the steady-state phase ($t > 5$ s), the estimated states maintain high fidelity, accurately following the small, sustained, high-frequency oscillations and the slow drift of the actual disturbance. The near-perfect overlap observed in all three axes, particularly on the rotational axis (d_α) and its estimate ($\hat{d}_\alpha$), confirms that the ESO is effectively lumping together and estimating the total uncertainty of the nonlinear system in real-time, thus successfully fulfilling its primary role in the ADRC framework.

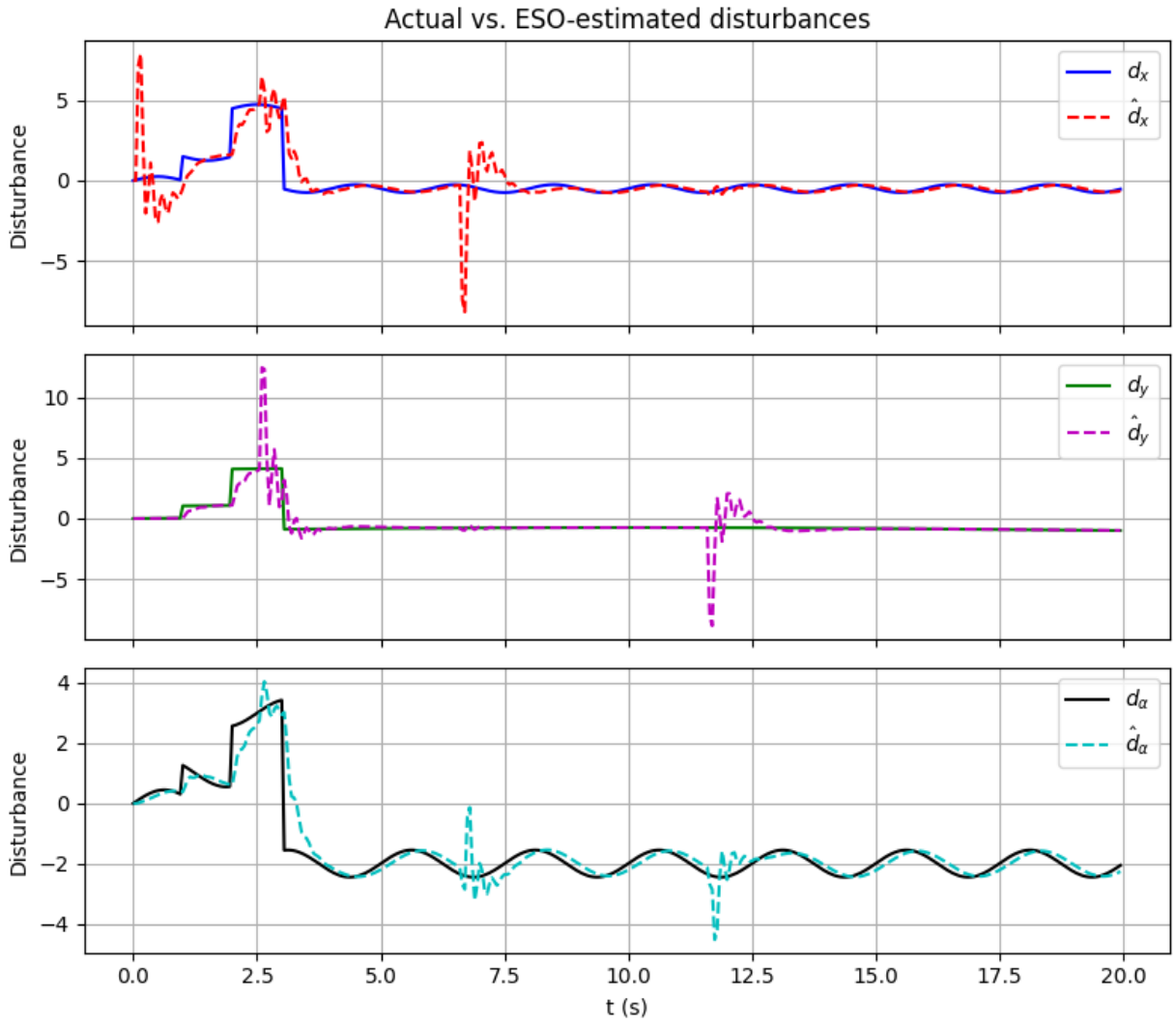


Figure IV.3: Comparison between actual disturbances and ESO-estimated disturbances along the x, y, α axes, highlighting observer convergence and estimation accuracy. Source: Author

The comparison of the robot's overall path demonstrates the critical difference in tracking fidelity between the passive PID and the active ADRC methodology. Figure IV.4 illustrates the path executed by the PID controller (dashed green line), clearly showing significant deviations from the ideal Reference Path (red dotted line). Specifically, the PID trajectory exhibits pronounced lag and overshoot at the corners, requiring much wider and slower turns to compensate for the applied disturbance and system inertia. This is particularly evident in the turn away from the waypoint at $(10, 0)$, where the robot significantly misses the corner before correcting its course, and in the overshoot into the final straight segment at $y = 10$ m.

In sharp contrast, ADRC control (blue solid line), shows the trajectory almost perfectly overlapping the ideal Reference Path. The ADRC system successfully mitigates the disturbance effects in real-time, resulting in precise cornering and negligible deviation along the straight segments. This near-perfect alignment confirms that the Extended State Observer (ESO) effectively neutralized the internal and external uncertainties. This allowed the ADRC-controlled robot to follow the geometric path with high accuracy and minimal lag, achieving a level of performance unattainable by the conventional PID controller under the same disturbed conditions.

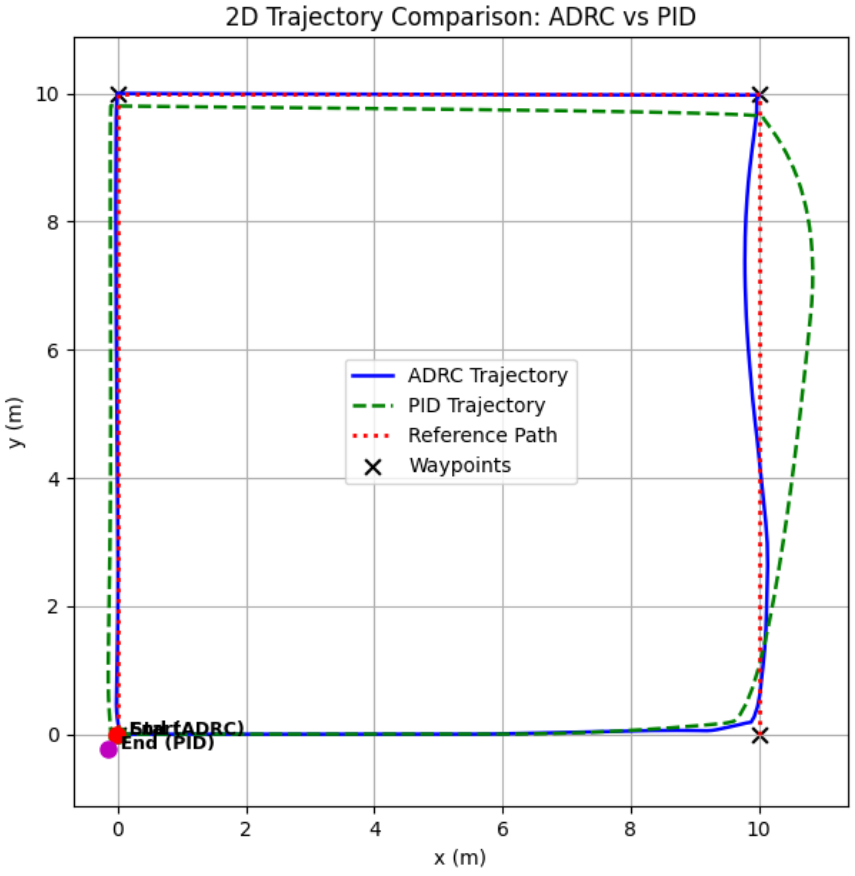


Figure IV.4: Robot's trajectory using ADRC X PID Control. Source: Author

Quantitative Performance Summary

The performance advantage of the ADRC approach over the conventional PID controller is definitively quantified by the metrics presented in Table IV.2. The Integral Absolute Error (IAE), which represents the accumulated tracking error over the entire mission, shows that the ADRC achieved superior performance in both translational distance (D_{Euc}) and rotational error (α), recording lower values of 29.7934 and 7.5644, respectively, compared to the PID's 32.7014 and 11.6031. This lower IAE confirms the ADRC's ability to maintain higher tracking fidelity throughout the aggressive maneuvers and under continuous disturbance. Regarding the Maximum Error (MAX), the ADRC demonstrated slightly superior performance in translational error (4.4798 vs. 4.5631), indicating a better peak suppression during transient phases. Crucially, while the PID showed a lower maximum recorded rotational error (MAX(α) of 2.6549 vs. the ADRC's 3.1402), this is directly attributable to the higher error tolerance window imposed on the PID controller to enable way-point completion. This restricted operational window prevented the PID from exhibiting its full oscillatory range during the severe transient phase. Despite this limitation on maximum deviation, the PID's significantly higher IAE(α) confirms that its rotational control was characterized by smaller, but more frequent and prolonged oscillations. Collectively, the data unequivocally demonstrates that the ADRC strategy provides a substantial quantitative improvement in overall tracking accuracy and disturbance rejection, validating its selection for the robust control of the holonomic robot.

Table IV.2: Comparative Performance Metrics: ADRC vs. PID

Controller	**IAE**		**MAX**	
	$\mathbf{D}_{\text{Euc}}$	α	$\mathbf{D}_{\text{Euc}}$	α
ADRC	29.7934	7.5644	4.4798	3.1402
PID	32.7014	11.6031	4.5631	2.6549

Chapter V Results of Computer Vision and Digital Twin

V.1 Computer Vision Results

The objective of this experimental evaluation is to systematically investigate the impact of training data composition on the performance and generalization capability of a deep learningbased cable detection model. Specifically, the experiments aim to analyze how different proportions of simulated images generated in Gazebo and photorealistic real-synthetic images influence detection accuracy, robustness, and domain transfer. By evaluating models trained under purely simulated, purely real-synthetic, and mixed-data configurations across both real and simulated test scenarios, this study seeks to identify effective training strategies that mitigate the sim-to-real gap while maintaining reliable performance in controlled environments. The analysis is conducted using standard detection metrics, including F1-score, precisionrecall (PR) curves, confusion matrices, and mean average precision, providing a comprehensive assessment of the trade-offs associated with each dataset configuration.

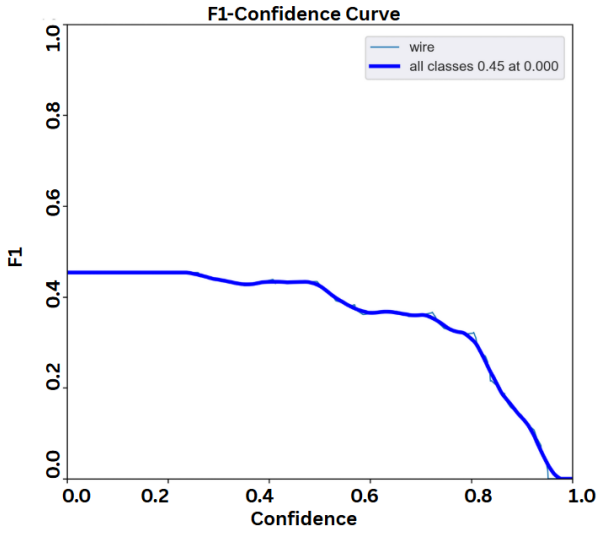
To validate the performance of the proposed detection models under various data conditions, a comprehensive evaluation was conducted using real-synthetic and simulated datasets. The object detection task is formulated as a single-class problem, where the only target class corresponds to electrical cables. The results are visually summarized through a series of figures, including F1-score curves, PR curves, and confusion matrices. These visualizations are crucial to understanding the models behavior in different test scenarios.

In this work, we distinguish between simulated images, generated directly in the Gazebo environment, and real-synthetic images, created by compositing real cable samples onto artificially generated backgrounds. And the datasets for test consist of one database with real world images, with a real background, and another test dataset with synthetic-simulated images.

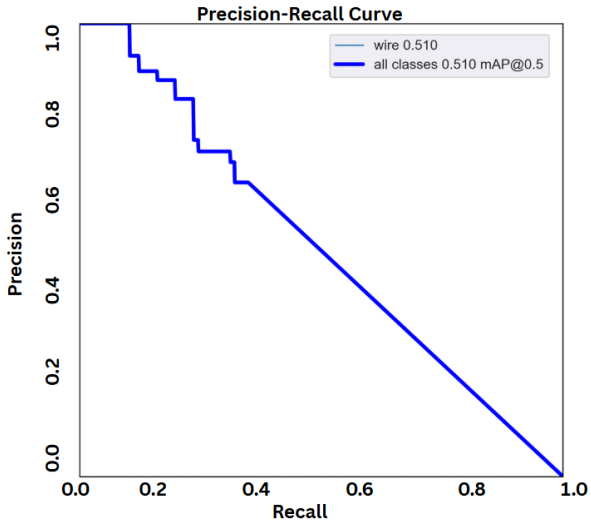
V.1.1 Testing on Real Datasets

Figures V.1(a) to V.1(c) present evaluation results on real-world synthetic data sets, showing how well models trained on synthetic data generalize to real-world conditions. Figures V.2(a) to V.2(c) illustrate the model's performance when tested on fully simulated datasets, providing insight into its ability to handle synthetic environments.

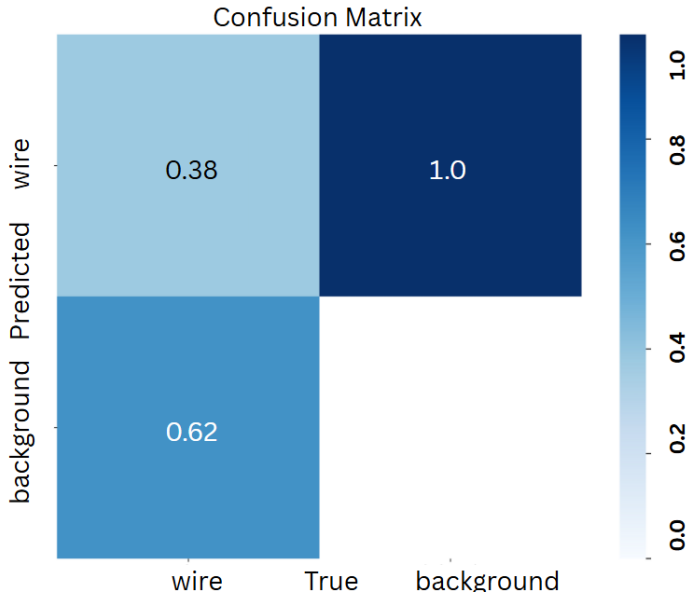
Note that since the problem is formulated as a single-class detection problem, the matrix represents the classification between the target class (wire) and the background. The confusion matrix is normalized with respect to the true class labels, such that each column sums to one. The results indicate that while the detector achieves perfect background rejection, it misses a significant portion of wire instances.



(a): F1-curve



(b): Precision-Recall curve

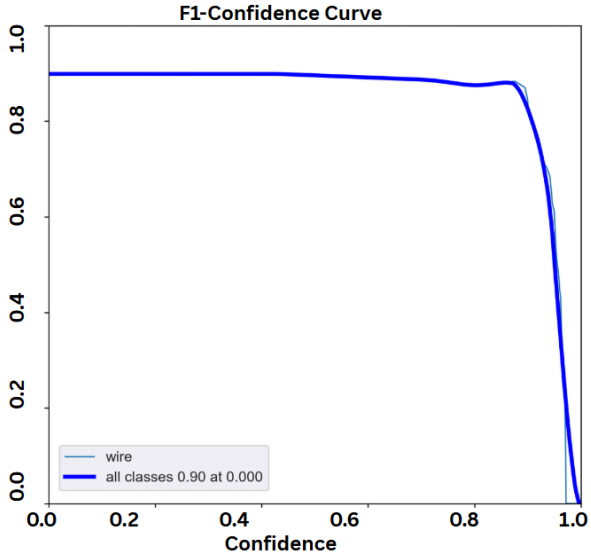


(c): Confusion Matrix

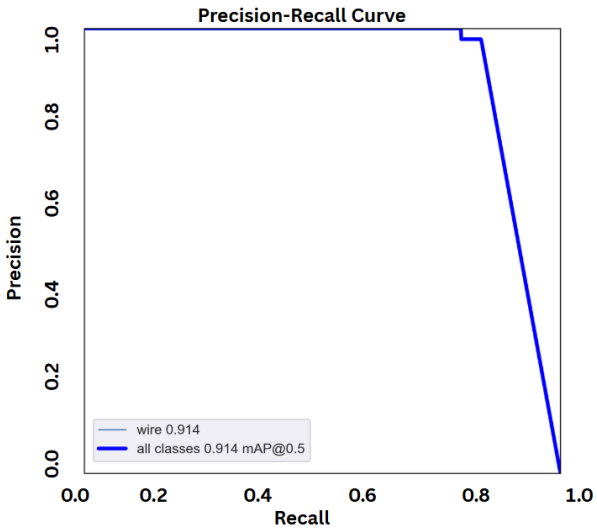
Figure V.1: Performance metrics for the model (F1, PR, and Confusion Matrix) trained on synthetic images tested on real-world images. Source: Author

V.1.2 Testing on Simulated Synthetic Datasets

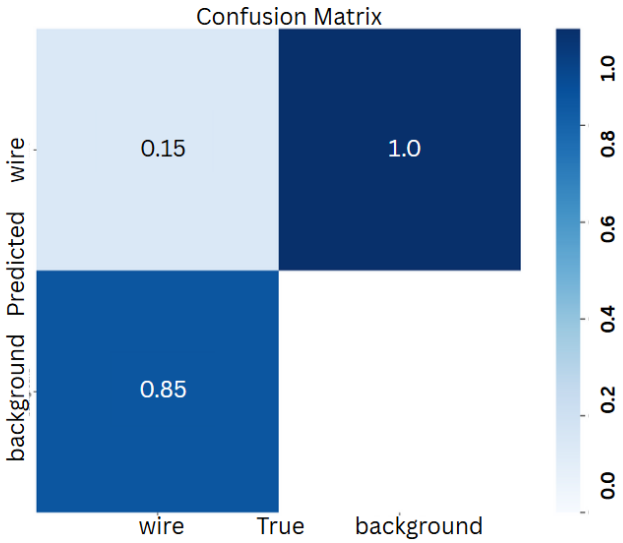
In this subsection, we analyze the models performance when tested exclusively on synthetic images generated in simulation environments. This evaluation is crucial to assess the model's accuracy in fully controlled scenarios, such as those produced in Gazebo. Figures V.3(a), V.3(b), and V.3(c) illustrate the corresponding F1-score curve, precision-recall curve, and confusion matrix, respectively. These results help to quantify the models detection capability under ideal, noise-free conditions, and serve as a baseline for comparison with evaluations on real synthetic and mixed datasets (i.e., real synthetic and simulated synthetic ones).



(a): F1-curve

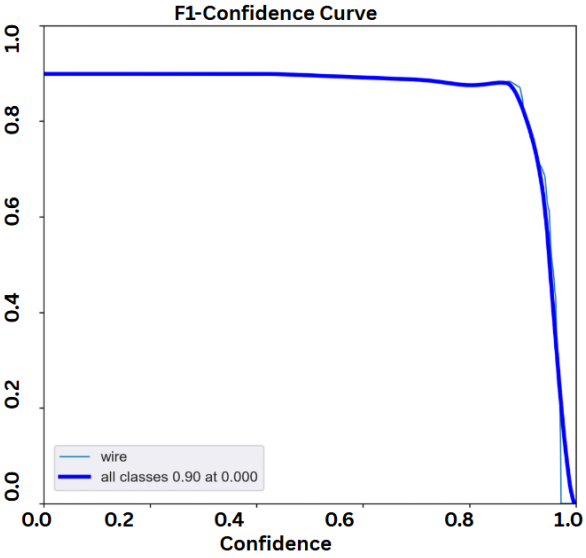


(b): Precision-Recall

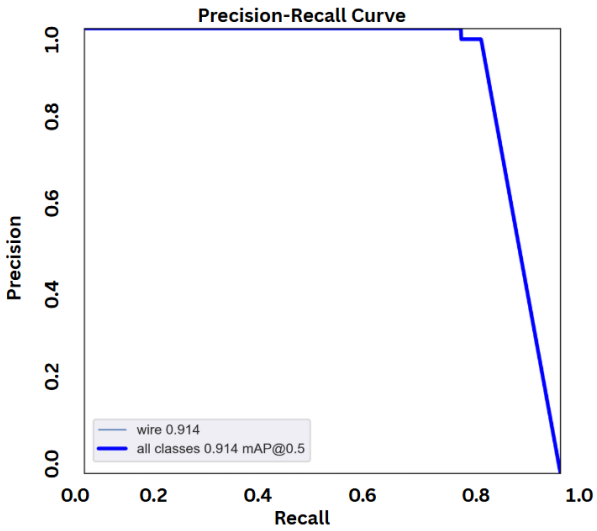


(c): Confusion Matrix

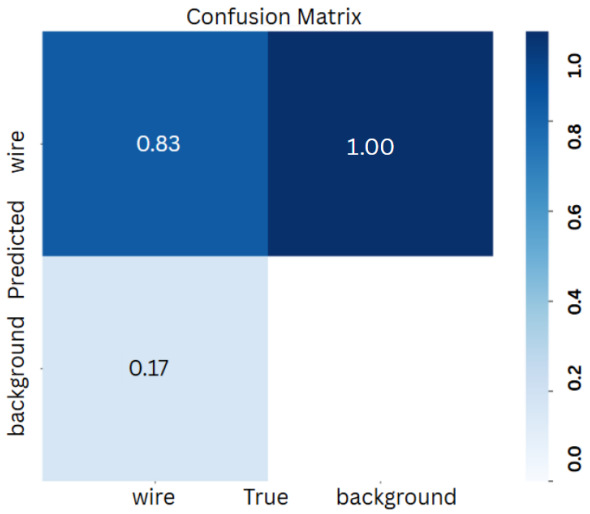
Figure V.2: Performance metrics for the model (F1, PR, and Confusion Matrix) trained on syntehtic images tested on simulated images. Source: Author



(a): F1-curve



(b): Precision-Recall



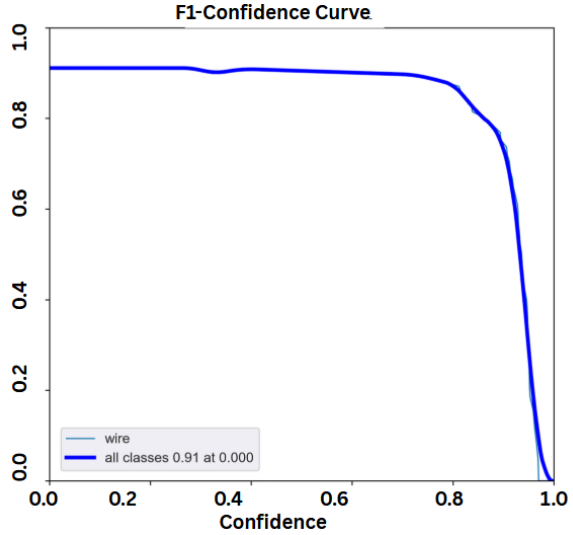
(c): Confusion Matrix

Figure V.3: Performance metrics for the model (F1, PR, and Confusion Matrix) trained on simulated images tested on simulated images. Source: Author

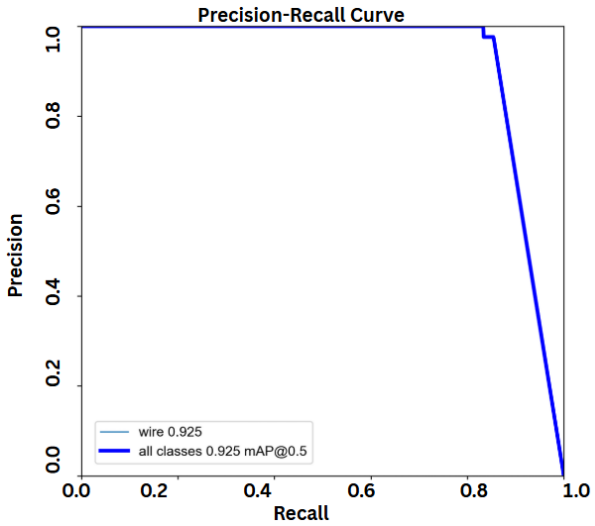
V.1.3 Testing on Mixed Datasets

Testing on Mixed Datasets (25% real synthetic images + 75% simulated images)

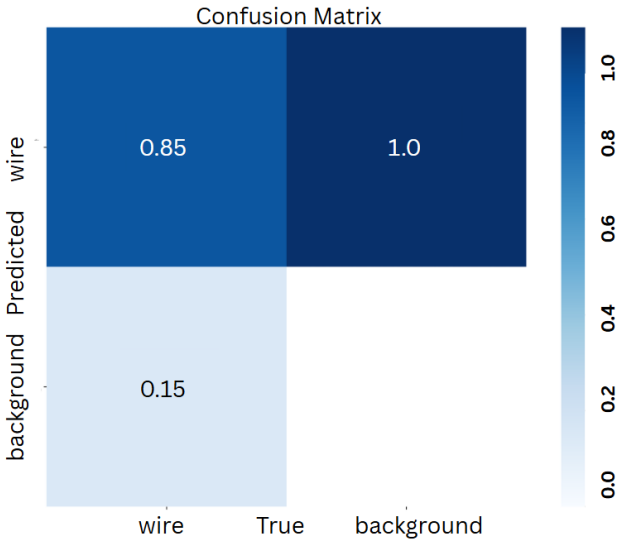
In this experiment, the model was evaluated using a mixed dataset composed of 25% real synthetic images and 75% simulated images. This configuration aims to investigate the impact of introducing a small proportion of photorealistic synthetic data into a predominantly simulated training environment. Figures V.4(a), V.4(b), and V.4(c) display the F1-score curve, precision-recall curve, and confusion matrix, respectively. These visualizations provide the necessary information in how a modest inclusion of realistic features affects the models ability to generalize and accurately detect targets across domains on an ideal scenario.



(a): F1-curve



(b): Precision-Recall curve

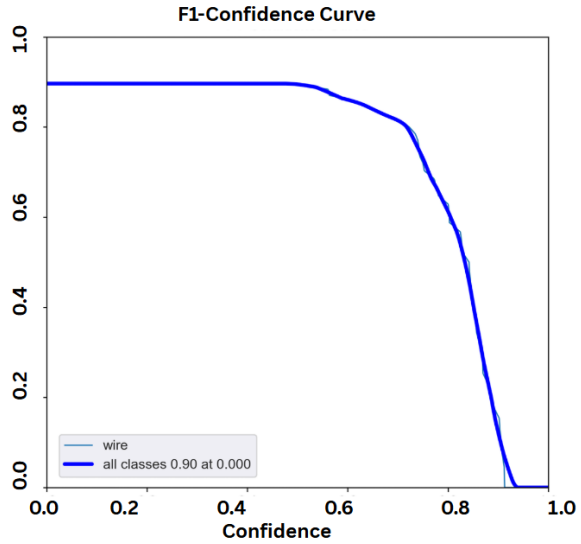


(c): Confusion Matrix

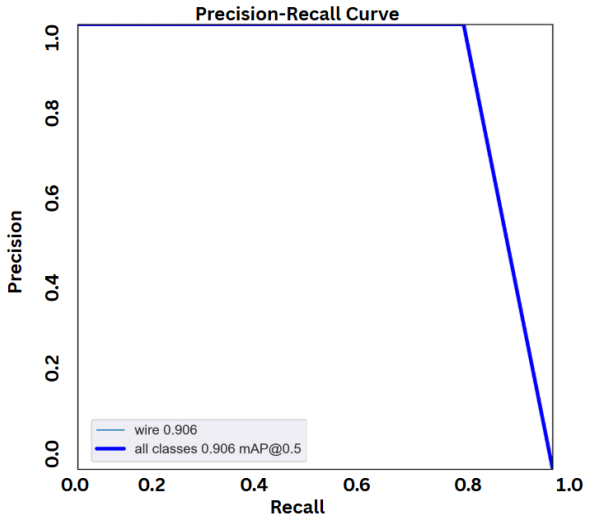
Figure V.4: Performance metrics for the model (F1, PR, and Confusion Matrix) trained on a mixed dataset (25% of real syntheic images + 75% of simulated images) tested on simulated images. Source: Author

Testing on Mixed Datasets (50% real synthetic images + 50% simulated images)

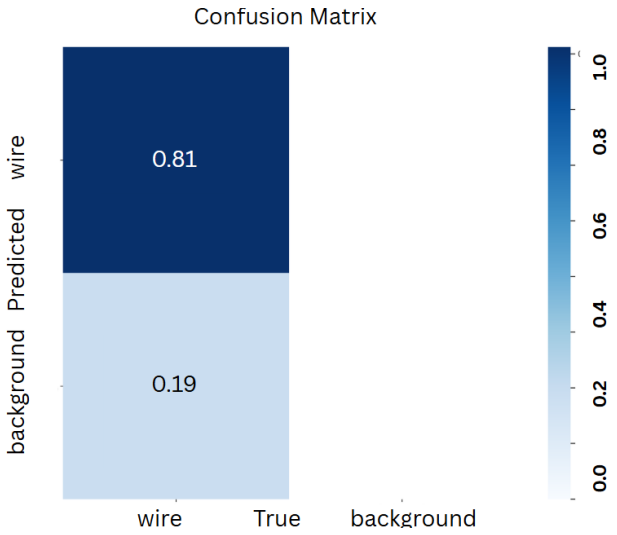
In this configuration, an equal balance between real synthetic and simulated images was used for training. The objective is to assess how a more even distribution of realistic and simulated features influences the models ability to generalize to unseen data. Figures V.5(a), V.5(b), and V.5(c) present the F1-score curve, precision-recall curve, and confusion matrix, respectively.



(a): F1-curve



(b): Precision-Recall curve



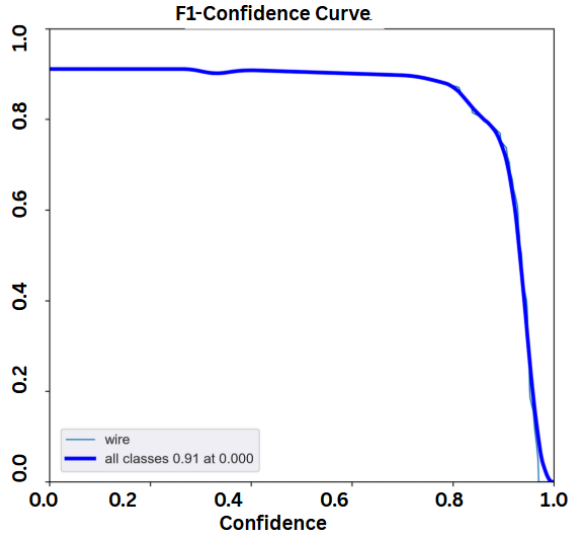
(c): Confusion Matrix

Figure V.5: Performance metrics for the model (F1, PR, and Confusion Matrix) trained on a mixed dataset (50% of real syntheic images + 50% of simulated images) tested on simulated images. Source: Author

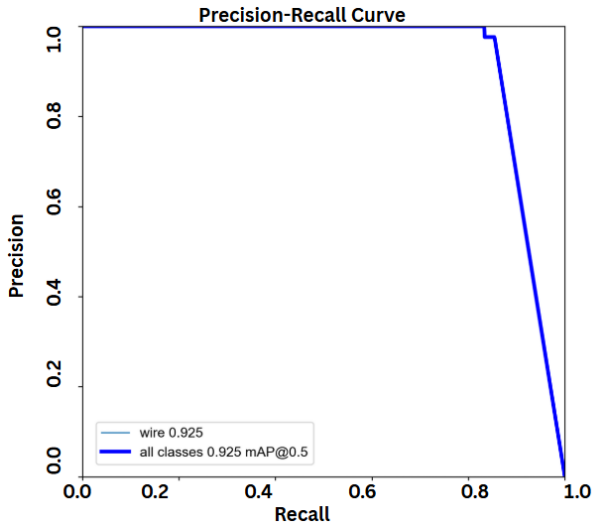
Testing on Mixed Datasets (75% real synthetic images + 25% simulated images)

In this final mixed-data experiment, the training set consisted predominantly of real synthetic images (75%) combined with a smaller portion of simulated data (25%). This setup aims to investigate

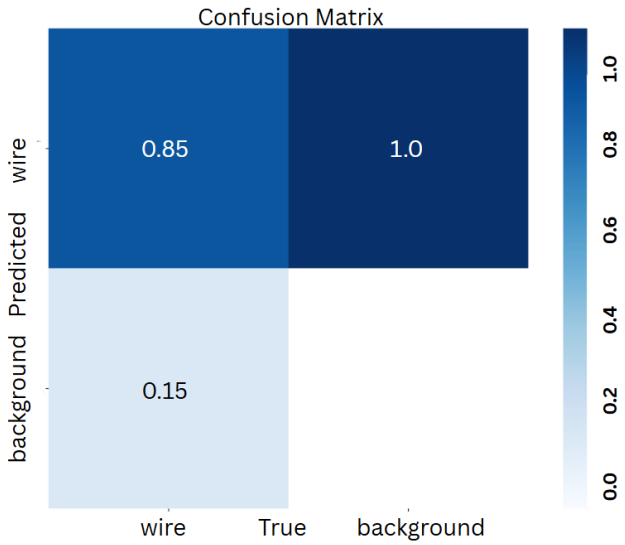
the extent to which a higher presence of photorealistic content enhances model performance and generalization. Figures V.6(a), V.6(b), and V.6(c) display the F1-score curve, precision-recall curve, and confusion matrix, respectively.



(a): F1-curve



(b): Precision-Recall curve



(c): Confusion Matrix

Figure V.6: Performance metrics for the model (F1, PR, and Confusion Matrix) trained on a mixed dataset (75% of real syntheic images + 25% of simulated images) tested on simulated images. Source: Author

V.1.4 Test Preformance metrics

The results are presented in Table V.1 and Table V.2. They demonstrate that a dataset augmented with 25% simulated (Gazebo) images significantly improves the model's detection performance in real-world scenarios compared to a model trained only on real-synthetic images. Similarly, this same mixed dataset notably improves detection in simulated environments when compared to using exclusively

simulated images.

Models trained exclusively on synthetic-simulated data, although useful for baseline detection, showed inferior performance in complex, real-world scenarios, as observed through lower F1-scores and precision-recall curves. This reinforces the hypothesis that purely simulated environments, while beneficial for early-stage testing, may not capture the rich variability found in real-world textures, lighting, and object-background interactions.

Dataset	Precision	Recall	mAP50	MAP50-95
Gazebo dataset	0.531	0.384	0.431	0.230
75% Gazebo 25% synthetics	0.525	0.406	0.436	0.232
50% Gazebo 50% synthetics	0.566	0.307	0.328	0.186
25% Gazebo 75% synthetics	0.522	0.453	0.480	0.269
Synthetics	0.638	0.377	0.429	0.205

Table V.1: Results of training dataset tested on real images dataset

Dataset	Precision	Recall	mAP50	MAP50-95
Gazebo dataset	0.976	0.853	0.944	0.829
75% Gazebo 25% synthetics	0.977	0.87	0.899	0.781
50% Gazebo 50% synthetics	0.997	0.812	0.895	0.727
25% Gazebo 75% synthetics	0.931	0.848	0.926	0.77
Synthetics	0.445	0.188	0.229	0.145

Table V.2: Results of training dataset tested on gazebo dataset

The results demonstrate that combining synthetic and simulated data does not lead to a strictly linear improvement in detection performance. In the evaluation with real images (Table V.1), the configuration using 25% Gazebo and 75% synthetic data achieved the best trade-off between precision (0.522) and recall (0.453), resulting in the highest mAP50 (0.480) and mAP50-95 (0.269). This indicates that a greater proportion of synthetic-real data enhances the model's generalization to real-world scenarios. In contrast, in the evaluation with the Gazebo dataset (Table V.2), although the 50%-50% split produced the highest precision (0.997), the best overall performance (mAP50 of 0.944 and mAP50-95 of 0.829) was obtained by the model trained exclusively on simulated data. These findings suggest that while training solely on simulated data is effective for synthetic environments, it is less suitable for generalization to real-world conditions. As shown in Figure V.7, the model's performance during the training phase converged to favorable values for each metric, indicating that the training and validation were successful and that the model was ready for evaluation on the test set. Figure V.8 illustrates the detection performance in a real-world scenario using the 25% Gazebo and 75% synthetic configuration, in comparison with Figure V.9, which shows results using only synthetic

images. Meanwhile, Figures V.10 and V.11 display detection outcomes in simulated scenarios using the best-performing configurations.

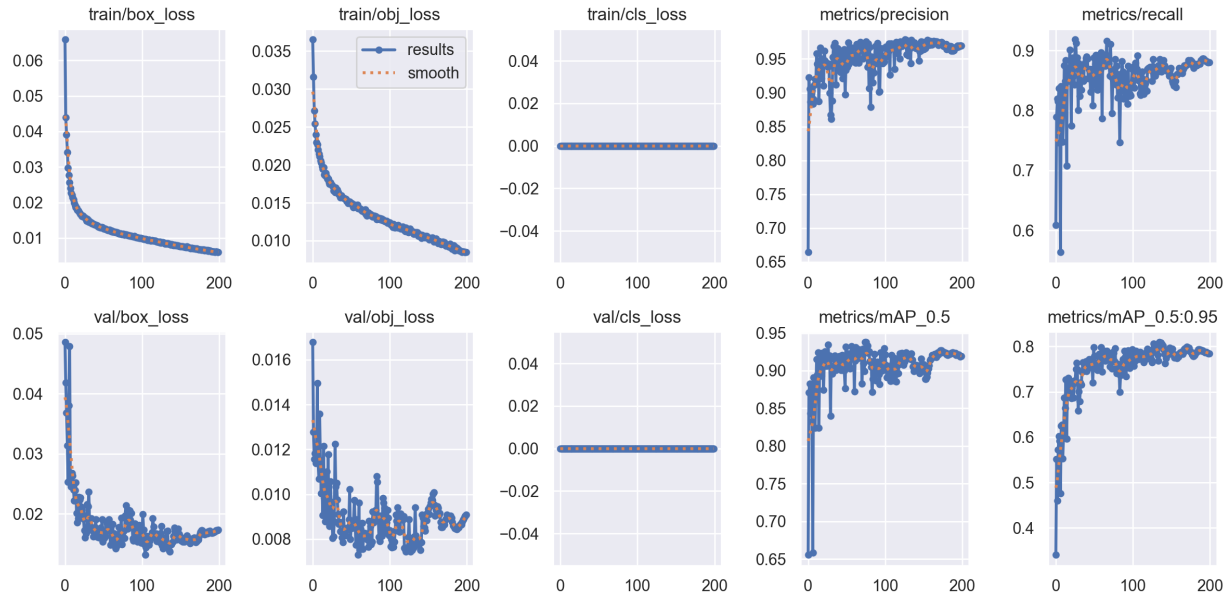


Figure V.7: Results of training step of dataset compound with 25% of simulated images. Source: Author

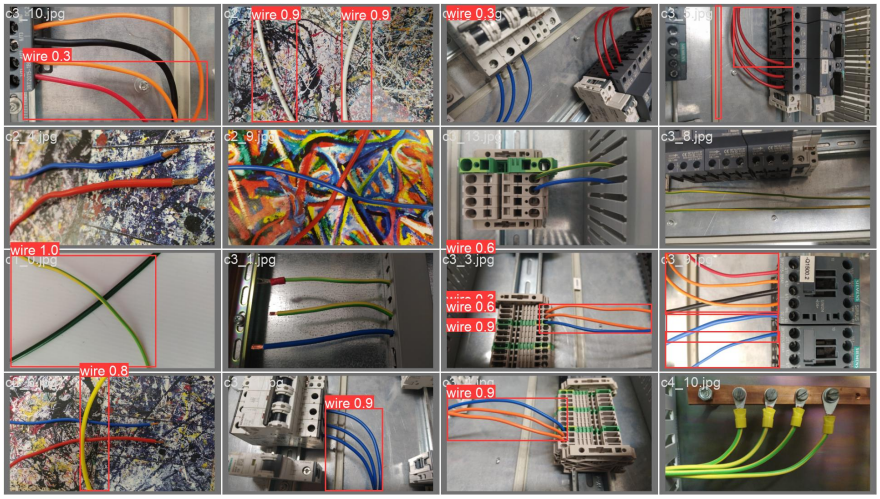


Figure V.8: Detection in a Real scenario for 75% real synthetic images + 25% simulated images. Source: Author

In this sense, the analysis validates the importance of considering both the quality and realism of training data in robotic perception tasks. It also highlights the potential of combining simulation-based data generation tools, such as Gazebo, with photorealistic rendering techniques to achieve scalable and effective training pipelines for deep learning-based inspection systems. Furthermore, the results indicate that a dataset composed exclusively of real-synthetic cable images is effective for training detection models in fully simulated environments. However, such datasets exhibit limited generalization capability when deployed in real-world scenarios, highlighting the importance of incorporating photorealistic data to mitigate the sim-to-real gap.

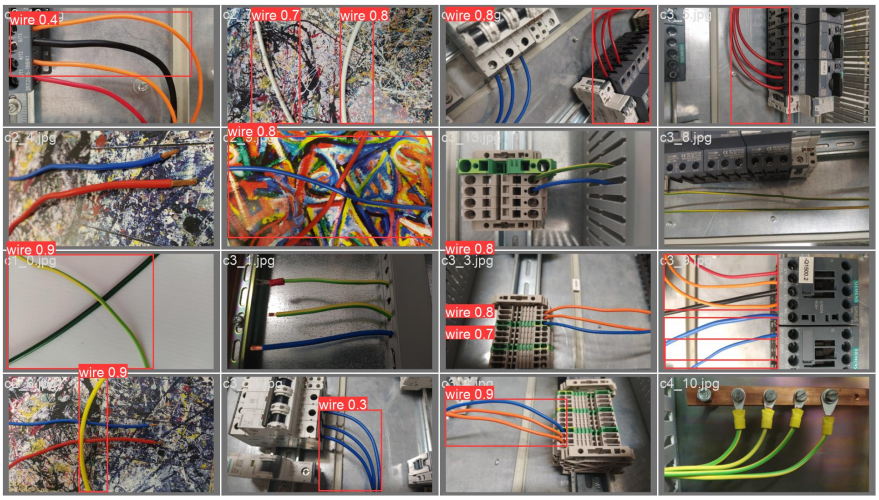


Figure V.9: Detection in a Real scenario for real synthetic images. Source: Author

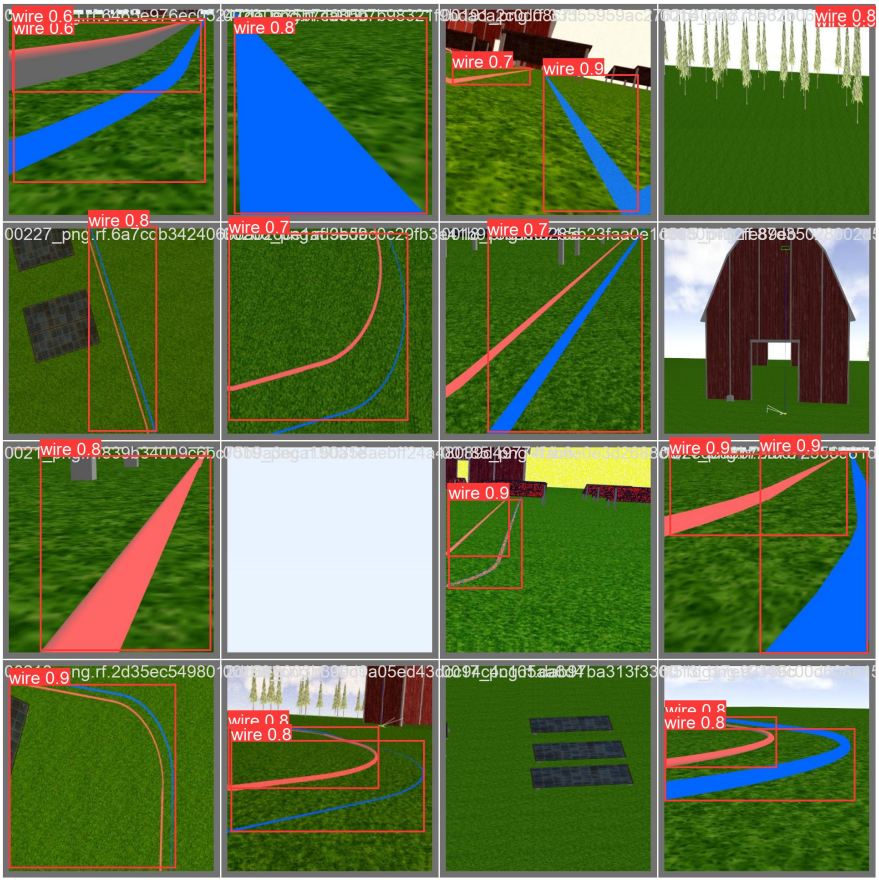


Figure V.10: Detection in an Ideal scenario for 50% real synthetic images + 50% simulated images. Source: Author

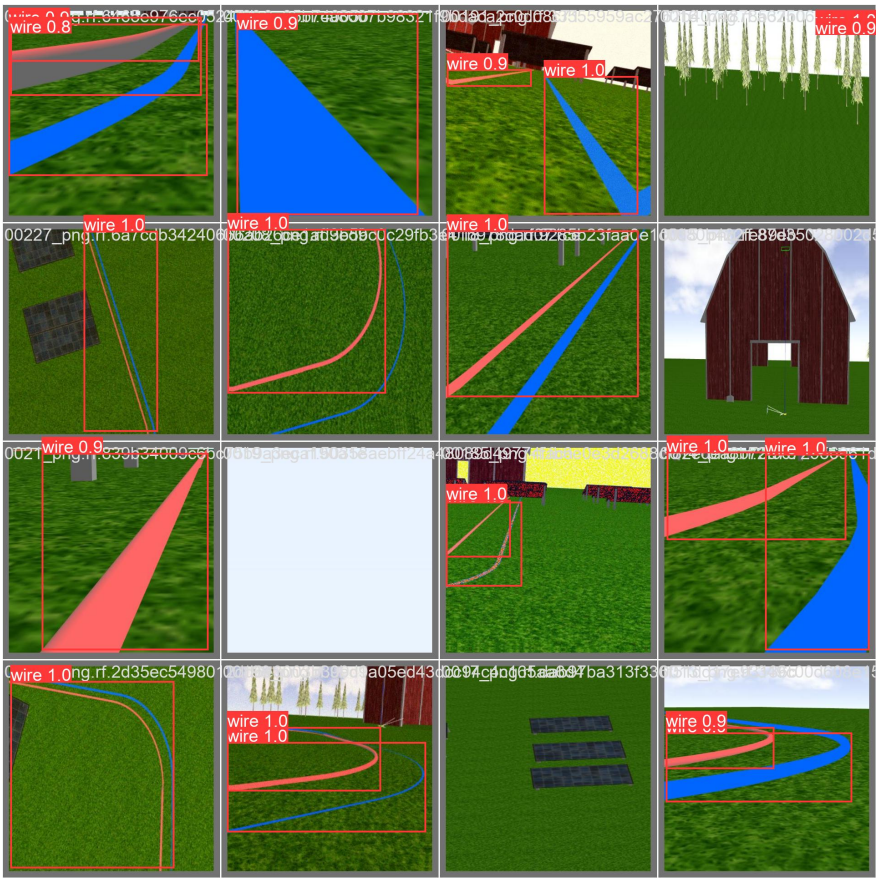


Figure V.11: Detection in an Ideal scenario for simulated images. Source: Author

V.2 Cable Follower and Path Planning Results

This section is fully dedicated to describing the validation of the UGV's two fundamental operational pillars: Cable Following for precise low-level inspection, and Path Planning for high-level maneuverability and coordination. Successful cable detection and tracking are essential for executing the ground inspection mission, while robust path planning is required to ensure efficient and safe movement across the field, particularly for coordinating resource-exchange operations with the UAV. The results presented herein demonstrate the system's accuracy and robustness under challenging simulated conditions, confirming the UGV's capability to perform both its primary inspection duties and its secondary support roles effectively.

Fig. V.12 presents the proposed approach to detecting the power cables in the simulated solar farm environment. The simulated environment introduces challenges that mimic real-world conditions, such as lighting variation and multiple objects that could visually interfere with detection. Despite these complexities, the system maintains accurate performance, demonstrating its robustness and ability to generalize across varied scenarios. However, the system successfully identifies the cable. This detection capability is an important part of the systems cable-tracking mechanism, once the detected cable position serves as input for the UGV's controller.

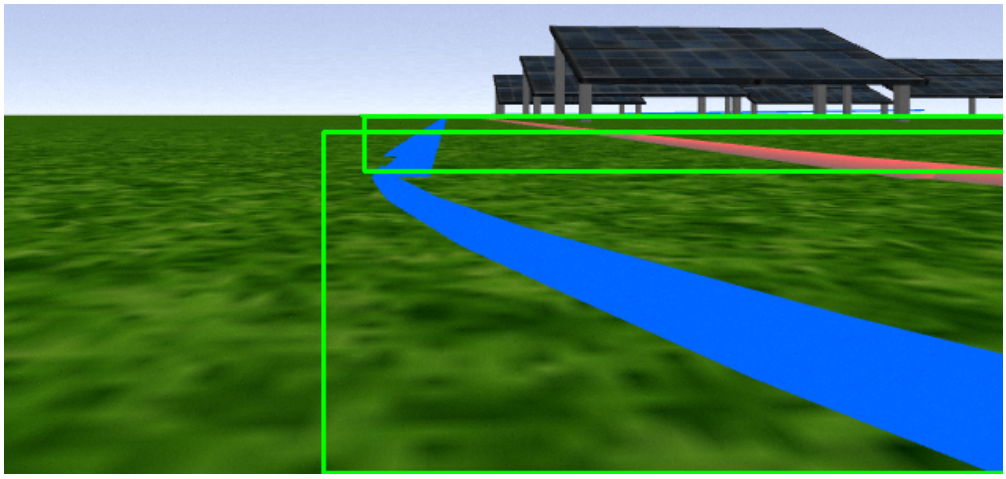


Figure V.12: AGV recognizing the cable and performing the tracking. Source: (AMORIM et al., 2025b)

This detection feeds into the internal robot controller, which dynamically adjusts the UGV's position relative to the cable based on real-time error signals. Fig. V.15 presents the data from the PID control system utilized by the UGV to follow the power plant cables. The graph showcases the system's error, setpoint, and PV over time. Initially, the error value is high as the UGV adjusts its position relative to the cable, albeit the error decreases as the system stabilizes and the UGV aligns itself with the cable. The setpoint remains constant throughout, indicating the desired position the system aims to maintain, while the PV converges toward the setpoint over time, demonstrating effective tracking. Note that the decision-making process (i.e., Layer 4 of Fig. III.9) ensures that the UGV continuously aligns with the cable despite environmental variations or initial positioning errors.

When the UAV requests landing assistance, the UGV uses the $A^\star$ algorithm to compute the optimal path to the UAVs location. This process involves evaluating potential paths and selecting the one with the lowest cost in terms of time and distance. Fig. V.16 illustrates the UGV's path as it responds to the UAV's landing request. It can be observed that the $A^\star$ algorithm performs effectively in generating an optimal path. However, there is a minor deviation between the real path and the reference track. Fig. V.13 shows the robot's actual trajectory (in blue) relative to the cable's position (in red) when performing the cable following for inspection purposes. Note that the robot adjusts its position using its control system, maintaining proximity to the cable. Note that during the cable-following task, the UGV is positioned laterally relative to the cable according to Fig. V.14, maintaining a fixed offset that keeps the cable within the field of view of the forward-facing onboard camera. The perception system continuously estimates the lateral deviation between the detected cable centroid and the image reference axis, which serves as the controller's error signal.

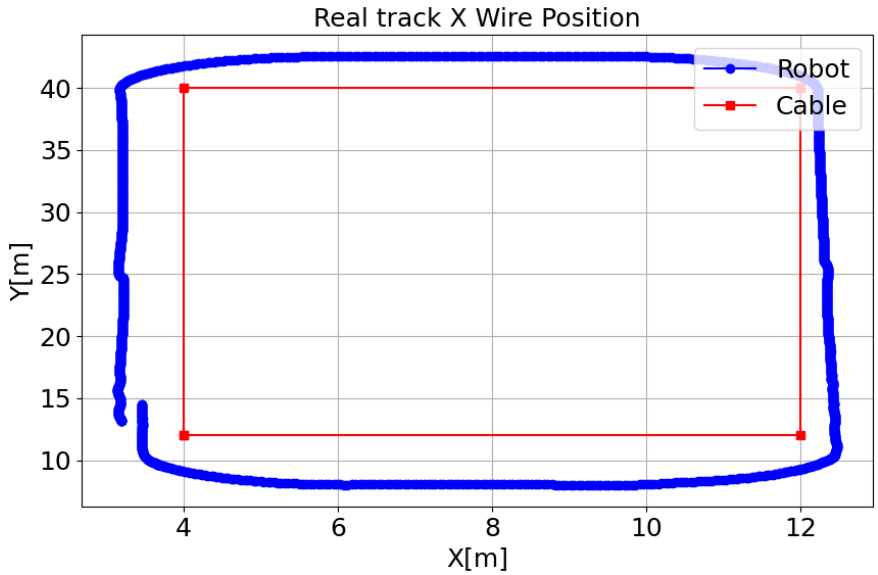


Figure V.13: Path taken by the robot when following the cable. Source: (AMORIM et al., 2025b)

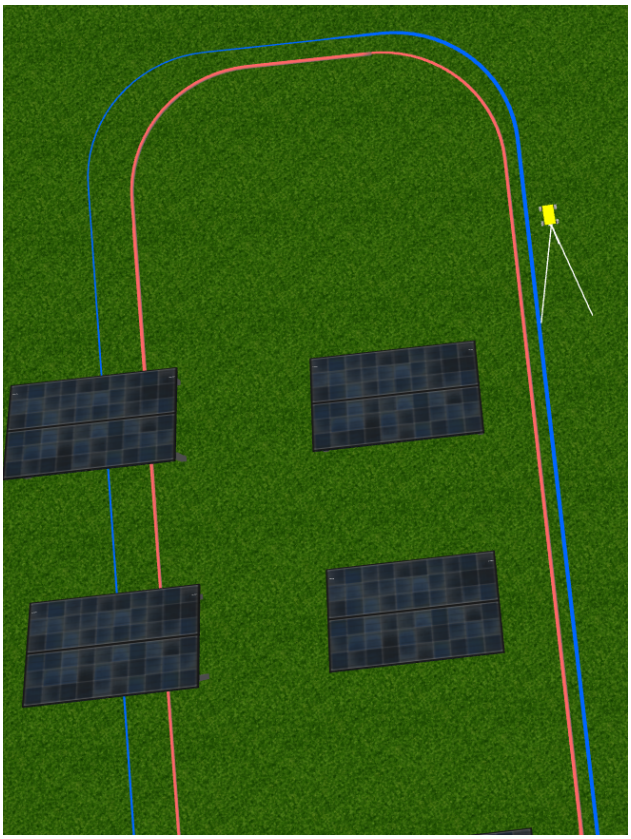


Figure V.14: Position of robot for Cable Tracking. Source: Author

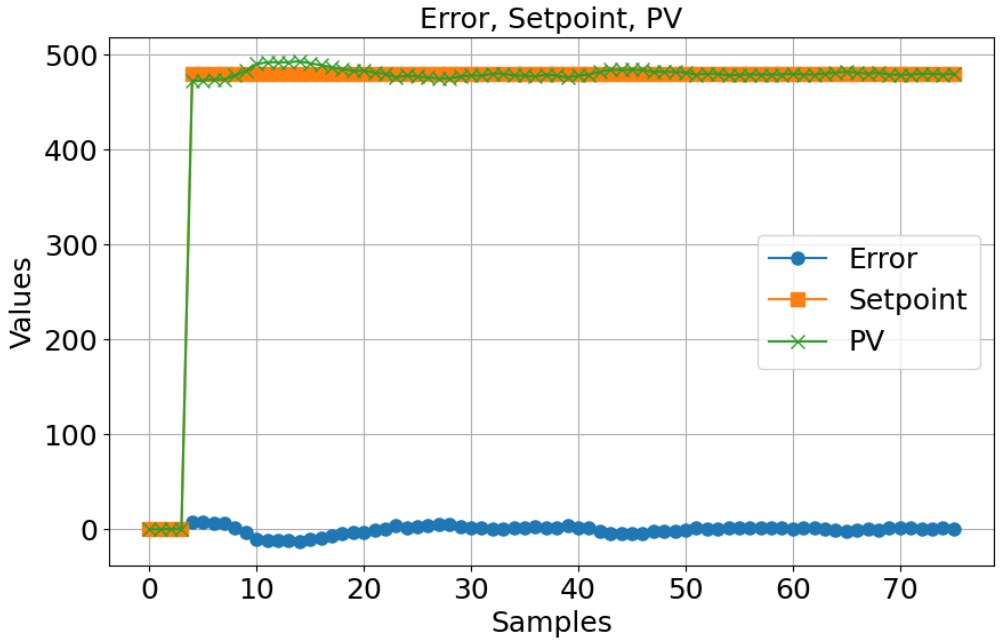


Figure V.15: Data from the control approach used to follow the cable. Source: (AMORIM et al., 2025b)

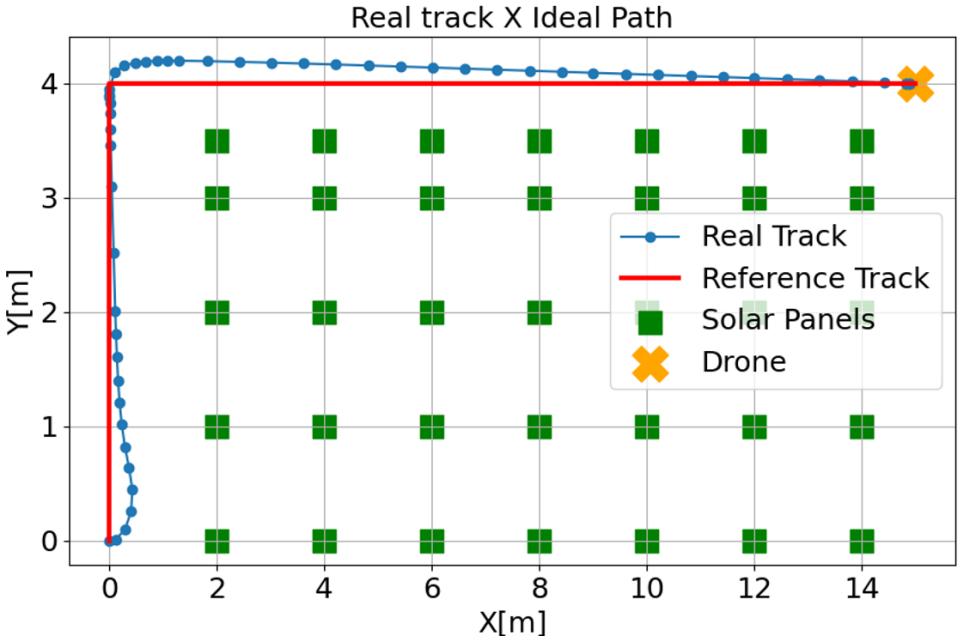


Figure V.16: Robot track to the UAV position requisition using $A^{\star}$ algorithm. Source: (AMORIM et al., 2025b)

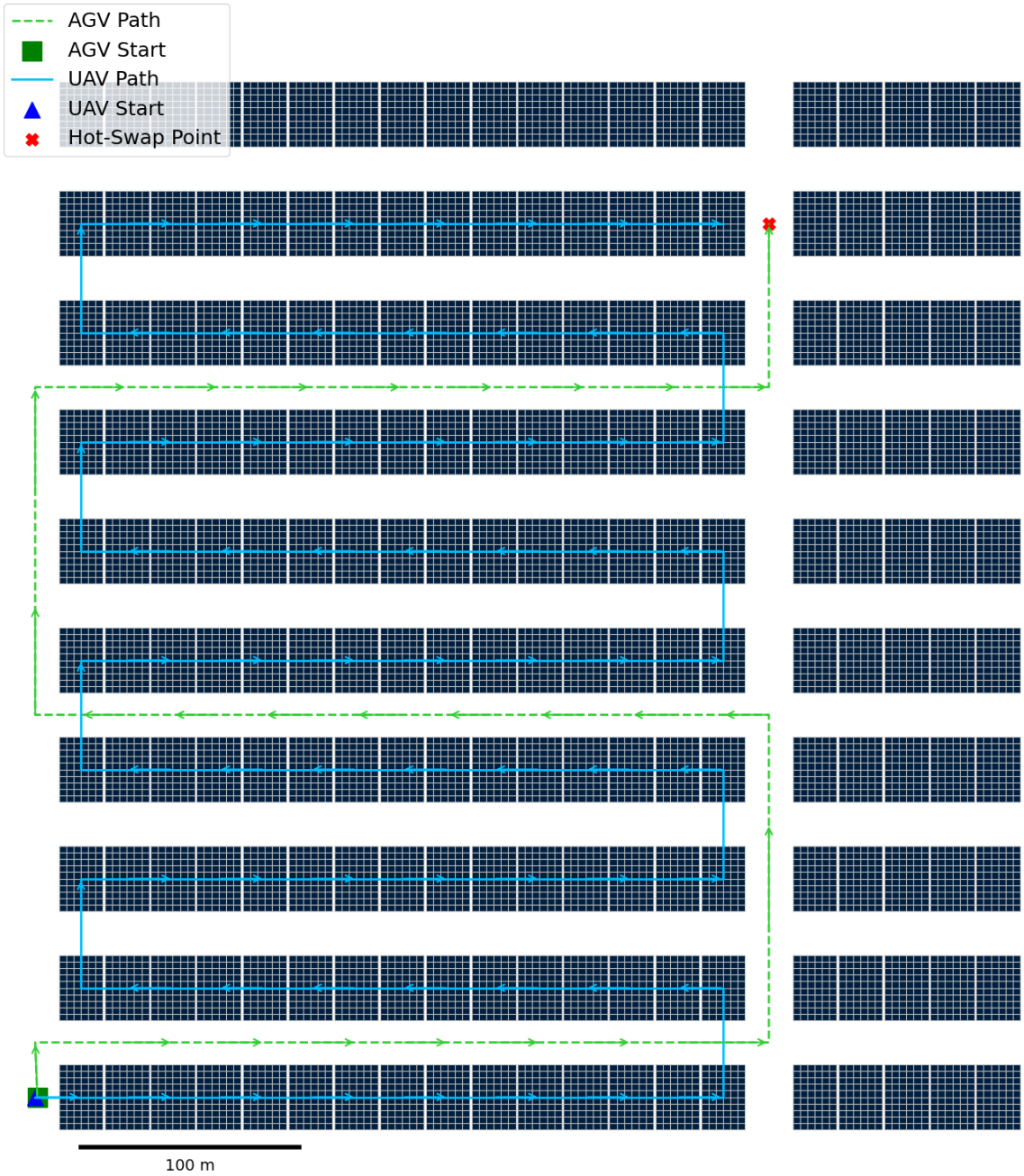


Figure V.17: Coordinated UAV and UGV inspection paths in a solar panel layout. The UAV follows a zigzag path covering the panel rows, while the UGV traverses between the rows and serves as a mobile charging station. A scale bar is included to visualize distances. Source: (AMORIM et al., 2025b)

Chapter VI Results of Cooperation Task

VI.1 UAV-UGV Role Division

A recurring challenge in collaborative multi-robot systems is achieving an effective division of labor while accounting for the different sensing, mobility, and endurance characteristics of aerial and ground platforms. In the proposed framework, the task allocation is designed to leverage each robot's complementary capabilities rather than enforce a uniform distribution of responsibilities. The UAV is primarily employed for inspecting the surface of solar panels, benefiting from its aerial perspective and rapid coverage, whereas the UGV is assigned to ground-level inspection tasks, such as cable and support structure monitoring, where proximity, stability, and payload capacity are advantageous. Additionally, the UGV can provide operational support functions, including battery hot-swapping for the UAV. This division of roles aims to enhance overall mission efficiency by exploiting platform-specific strengths.

Figure V.17 illustrates this coordinated inspection scenario. The UAV executes a high-speed zigzag trajectory over the panel rows, traveling at an average speed of 3.0 m/s with a battery autonomy of approximately 15 minutes, enabling up to 2700 m of coverage per mission. In contrast, the UGV moves at 1.0 m/s and can operate for significantly longer periods. However, physical constraints and the need to navigate between panel supports restrict its effective coverage radius to approximately 900 m. Consequently, it is infeasible for the UGV to match the UAVs aerial coverage, underscoring the need for a complementary mission design.

Table VI.2 provides a comparative overview of the UAV and UGV capabilities and their mission roles, highlighting how each contributes uniquely to overall inspection performance. Total mission coverage is therefore evaluated not through the range of individual robots but through the effectiveness of their joint and complementary tasks. The UAV performs fast, broad-area inspections, while the UGV ensures detailed ground analysis and logistical support.

Figure VI.1 presents the interblock communication between the agents, where mission data, resource usage, and inspection status are dynamically exchanged. This interaction is monitored by the HITL mechanism integrated into the GCS, enabling operator oversight and intervention when necessary.

Table VI.1 summarizes the key performance metrics of the cooperative UGV-UAV system during solar farm inspection. While the architecture operates autonomously for the majority of tasks, HITL supervision, described in Layer 5 of Fig. III.9,ensures operational safety in unpredictable scenarios. The landing procedure follows the methodology described by Berger et al. (BERGER et al., 2023), supporting reliable UAV-UGV interaction, which is outside the main scope of this study. An additional benefit arises from the parallelism of aerial and ground inspections, which significantly increases total coverage and reduces inspection time for large-scale solar farms.

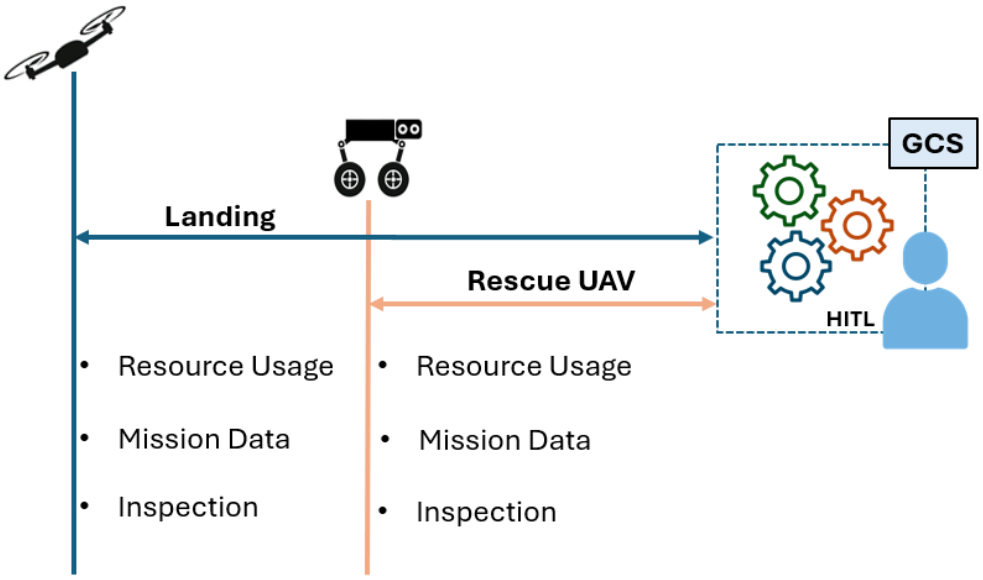


Figure VI.1: Interblock communication among the proposed architecture. Source: (AMORIM et al., 2025b)

Table VI.1: UGV-UAV Cooperation Performance Metrics

Metric	Value
Path Planning Time	5.5 seconds
Path Deviation	1.34%
Battery Threshold	Trigger at 20%
Inspection Time	Reduced compared to manual inspection
Autonomy Level	High

Another important aspect is the low path deviation of the UGV, ensuring that it remains closely aligned with the planned route and maintains high inspection efficiency. When the UAV battery level reaches the 20% threshold, the $A^{\star}$ algorithm computes an optimal repositioning path for the UGV in 5.5 s to facilitate a safe landing. In situations where obstacles or environmental constraints prevent the UGV from reaching the UAV, the UAV autonomously lands in a predefined safe area, strategically selected to avoid hazards such as cables or support structures. This entire process is continuously monitored by the GCS through the HITL system.

All experiments were initialized using predefined inspection paths, take-off altitudes, and detection thresholds. These conditions ensured fairness across multiple simulation runs and improved reproducibility. To support further development and validation, the project scripts are publicly available at: `https://github.com/Johanns123/UGV_Wire_Follower`.

VI.2 Human-in-the-Loop Evaluation

In autonomous robotic systems, especially those deployed in industrial inspection tasks, human supervision often complements full autonomy to handle edge cases. To emulate such conditions, this

Table VI.2: Comparison between UAV and UGV coverage capabilities

Metric	**UAV (Iris)**	**UGV (Nexus)**
Velocity	3.0 m/s	1.0 m/s
Mission Duration (max)	∼15 min (battery limit)	∼30+ min (extended)
Maximum Coverage Distance	2700 m	900 m
Inspection Focus	Surface (panels)	Cables & ground
Autonomy Constraint	Battery-based	Less restrictive
Navigation Area	Over panel rows	Between rows

work uses a ROS-Gazebo simulation integrated with MAVROS and ArduPilot for UAV control, and a differential drive robot using the ROS navigation stack for UGV. A suite of Python scripts was developed to simulate unexpected failures and log the response behavior of the operator node, mimicking human intervention. Six representative HITL scenarios were designed (see Table VI.3), each involving a distinct trigger and a defined fallback behavior. The operator response was simulated with latency, logging detection, and action times.

Table VI.3: Summary of HITL Scenarios

ID	**Description**	**Agent**	**Trigger**	**Operator Action**
S1	GPS signal loss in UAV	UAV	Sensor dropout	Switch to *ALT_HOLD* and manually reposition
S2	Unexpected obstacle on path	UGV	Perception node alert	Reroute via RViz interface or override stop
S3	UAV-UGV meeting point mismatch	Both	Arrival time desync	Modify meeting location or delay UGV
S4	Ambiguous route decision in planner	UGV	Tied cost paths	Manual route selection based on context
S5	Sudden wind instability	UAV	High IMU variance	Hold position and await operator clearance
S6	Sensor anomaly in UGV	UGV	Temperature spike	Enter *IDLE* mode and pause inspection

To operationalize HITL behavior, a modular control structure was designed based on ROS communication primitives. The system continuously monitors critical topics for anomaly detection and propagates fallback requests through a dedicated control channel. A supervisory node, written in Python, subscribes to telemetry and sensor topics, i.e., */mavros/global_position/raw/fix*, */mavros/imu/data*, and UGV-specific perception alerts. Upon detecting an abnormal event (e.g., GPS dropout, obstacle proximity,

time desynchronization), it publishes a string alert to the topic */hitl/override_mode*, encoding the type of fallback requested (e.g., *GPS_LOST*, *OBSTACLE_UGV*, etc.).

Figure VI.2 illustrates the internal communication flow of the HITL system, including the sensor pipelines, the supervisor node responsible for failure detection, the operator node emulating human intervention, and the final actuation commands issued to UAV and UGV agents.

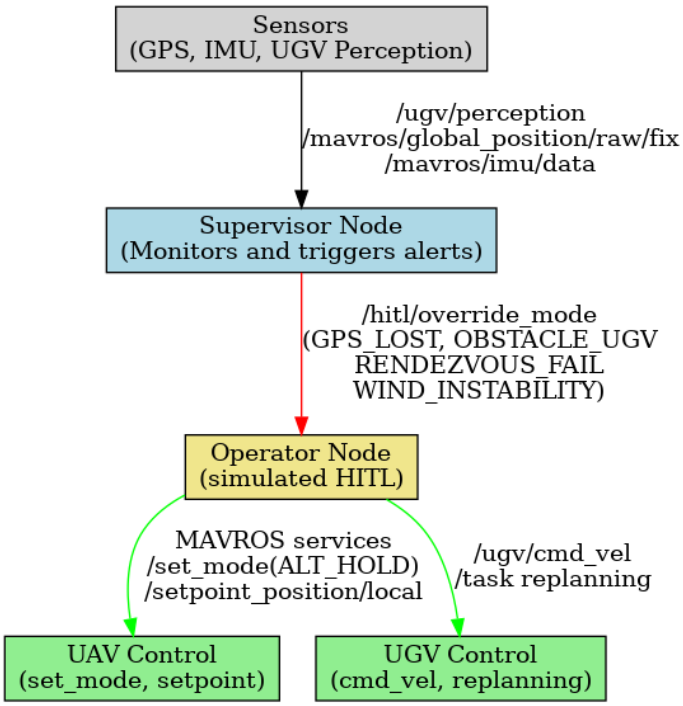


Figure VI.2: System-level HITL communication architecture. Colored arrows represent topic-based message flow from sensors to fallback actions through the operator node. Source: (AMORIM et al., 2025b)

An operator simulation node, acting as a proxy for human intervention, subscribes to */hitl/override_mode*, initiates a latency timer to simulate reaction delay, and then executes the appropriate fallback procedure. For the UAV, this involves calling the MAVROS service */mavros/set_mode* to switch to a safe mode, which is *ALT_HOLD*, and optionally publishing a position setpoint to */mavros/setpoint_position/local*. For the UGV, custom ROS messages are published to */cmd_vel* or task replanning services, depending on the scenario.

For example, in Scenario S4, tied-cost paths generated by the UGV planner trigger a manual override, where the operator selects the preferred route using contextual knowledge, triggering a replan via a ROS service. In the same way, in Scenario S6, thermal anomalies lead to immediate inspection suspension by issuing a zero-velocity command and entering *IDLE* mode.

The sequence of detection, alert publication, operator override, and acknowledgment logging is timestamped using ROS clock synchronization to compute precise response latencies. Logs confirm that system-level acknowledgment (e.g., mode switch) occurs within 0.2 seconds of service invocation, validating the responsiveness of MAVROS and the low-overhead nature of ROS intra-process communication.

Human reaction latency is critical in HITL robotic systems, particularly in scenarios requiring rapid perception and control. Pfeiffer and Scaramuzza (PFEIFFER; SCARAMUZZA, 2021) demonstrated that experienced drone pilots exhibit an average visual-motor response delay of approximately 220 milliseconds when navigating high-speed drone racing tasks. This value serves as a lower-bound reference for HITL systems, where contextual decision-making may increase total response time to several seconds depending on the situation's complexity. The latencies observed in our simulated scenarios, which ranged from 0.3 to 2.1 seconds, are thus within a realistic and operationally acceptable range.

The MAVLink protocol is designed for low-latency communication and typically exhibits transmission delays below 10 milliseconds in local simulations (MEIER et al., n.d.). Similarly, ROS communication in intra-machine setups results in sub-millisecond latency. Thus, system response is largely governed by operator reaction time rather than infrastructure delay.

The UAV system was configured to support automated fallback modes in response to failure events. For instance, upon GPS signal loss (Scenario S1), the system triggers a mode change request to *ALT_HOLD* and publishes an alert on the */hitl/override_mode* topic. The operator node then confirms the transition using the MAVROS service */mavros/set_mode*. According to logs, mode change acknowledgments were received within 0.2 seconds, demonstrating the effectiveness of the protocol and its low overhead.

The integration of HITL scenarios and latency-aware mechanisms significantly strengthens the realism and reliability of the proposed inspection framework. By explicitly modeling the timing and conditions under which human operators intervene and defining robust fallback behaviors, the system becomes more resilient to unexpected events. This design philosophy aligns with recent advances in HITL robotic control, emphasizing safety guarantees through intention-aware adaptation, trajectory shaping, and shared autonomy protocols (DANI et al., 2024). These techniques are particularly relevant in critical applications such as aerial inspection and human-robot collaborative environments, where safety and responsiveness are paramount.

VI.3 Comparison with Traditional Inspection Methods and Single-Robot Approaches

Traditional Photovoltaic inspection methods, such as manual thermography and visual walkthroughs, are time-consuming, labor-intensive, and prone to inconsistency and human error. Recent literature and industrial applications highlight the growing use of autonomous robots, either aerial or ground-based, to overcome these issues (SITECAPTURE, 2025), (FLYGUYS, 2024). However, many of these solutions rely solely on either UAVs or UGVs. UAVs can quickly cover large panel surfaces with thermal or RGB cameras but suffer from limited flight autonomy, payload constraints, and difficulty accessing lower structures like cable trenches. On the other hand, UGVs are well-suited for inspecting low-lying components and enabling close-up evaluations, but are considerably slower and less efficient for large-

area coverage. The proposed UAV-UGV cooperation brings the best of both worlds:

- While UAVs prioritize fast aerial scanning of wide panel fields, UGVs handle critical base-level elements such as cable trays, electrical boxes, and thermal anomalies near the ground.
- The UGV acts as a mobile hot-swap station, enabling energy-aware planning for UAVs and extending their operational time without requiring fixed docking stations.
- Simultaneous ground and aerial perspectives reduce false positives and improve precision in anomaly triangulation.
- In case of UAV failure, the UGV can still perform partial inspection, and vice versa, improving system resilience.

Table VI.4 summarizes the qualitative differences between traditional, single-robot, and our collaborative inspection approaches.

Table VI.4: Qualitative comparison of Photovoltaic inspection approaches

Criterion	**Manual**	**UAV/UGV only**	**UAV + UGV**
Inspection time	High	Moderate	Low
Coverage range	Limited	Moderate High	High
Access to low structures	Yes	UGV only	Yes
Energy autonomy	N/A	UAV-limited	Balanced (UGV support)
Fault redundancy	Low	Medium	High
Safety	Low	High	High
Adaptability to failures	Low	Medium	High
Operational cost	High	Medium	Medium

Recent industry reports reinforce the relevance of both modalities. For example, Clearpath Robotics presents the *Husky Observer* as a ground solution for solar field inspection, highlighting its robustness and sensor payload flexibility (ROBOTICS, 2024). Conversely, companies like FlyGuys (FLYGUYS, 2024) emphasize the scalability of drone-based inspection. However, neither fully addresses the need for redundancy, recharging logistics, or multi-layered inspection, all enabled by our hybrid approach.

Chapter VII Conclusions and Future Work

VII.1 Conclusions

The proposed framework successfully integrates a collaborative system between the Nexus UGV and the Iris UAV to optimize solar panel inspections. While the UGV focuses on ground-level tasks such as cable detection and navigation, the UAV performs aerial inspections to identify issues such as cracks, hotspots, and surface anomalies. This study demonstrates the effectiveness of multi-robot collaboration in large-scale and complex environments, where complementary strengths significantly improve overall inspection performance. The inclusion of a cognitive framework and a human-in-the-loop mechanism further enhances system robustness, ensuring adaptability to dynamic scenarios while providing a fail-safe layer for critical decisions.

Integrating the YOLOv5 object detection model substantially improved the UGVs ability to detect and track cables with high precision, and the $A^\star$ algorithm ensured efficient path planning to support the UAV during cooperative missions. These components were validated in a simulated ROS–Gazebo environment, showing strong performance, adaptability, and potential for real-world deployment.

The Active Disturbance Rejection Control ADRC strategy also played a key role in improving the holonomic UGVs mobility performance. By actively estimating and compensating for internal and external disturbances, the ADRC-based controller enabled smoother trajectory tracking, reduced steady-state error, and improved robustness against model uncertainties and irregularities in the simulated terrain. This resulted in more stable and reliable motion behavior, which is essential for precise cable following and coordinated maneuvers with the UAV.

VII.2 Future Proposals and Limitations

Although the proposed framework demonstrated promising results in simulation, several limitations must be acknowledged to contextualize its current performance and guide future developments. One of the primary constraints lies in the YOLOv5 detection model. The network was trained using a dataset composed largely of synthetic backgrounds that do not accurately represent real inspection environments. This variety of realistic textures, lighting conditions, and physical cable variations reduced detection accuracy during testing, particularly when identifying cables under challenging visual conditions. Consequently, the models generalization capability remains limited when transitioning from simulation to real-world scenarios.

Another limitation concerns the evaluation of the ADRC strategy. While the controller achieved robust performance in simulation-successfully compensating for modeled disturbances and ensuring smooth trajectory tracking, its behavior under real environmental perturbations remains unverified. Factors such as uneven terrain, wheel slippage, sensor noise, and dynamic external forces were not fully replicated in the simulation environment, leaving an important gap in assessing the true robustness of the holonomic UGV under real operating conditions.

Furthermore, the collaborative framework itself has to be tested on physical robotic platforms. The coordinated operation between the omnidirectional UGV and the UAV, including cooperative inspection, communication exchange, and landing support procedures, was validated only in simulation. Real-world tests are essential to evaluate communication delays, mission latency, battery constraints, and the reliability of inter-robot cooperation under environmental and operational uncertainties.

As future developments, we intend to deploy the complete collaborative system on real robotic platforms, integrating the omnidirectional UGV with a UAV in an actual solar farm inspection scenario. Additionally, a new training cycle of the YOLOv5 model will be carried out using a dataset composed of real inspection backgrounds and samples containing broken cables, missing segments, and other integrity-related anomalies. This improved dataset is expected to enhance detection accuracy and robustness significantly. Ultimately, these efforts aim to validate the full framework under real operational conditions and strengthen the reliability of cable-integrity assessment in photovoltaic installations.

VII.3 Scientific Production

To date, this masters research has yielded one journal article and two conference papers, all of which list the author of this dissertation as the first author:

- Amorim, Johann SJC, et al. "Collaborative Inspection of Solar Panel Farms Using YOLOv5-Based Computer Vision and UGV-UAV Integration." Journal of Intelligent & Robotic Systems 111.2 (2025): 66.
- Amorim, Johann SJCC, et al. "A Practical Approach to Teaching Differential Robot Control Using a Goal-to-Goal Interface." 2024 Brazilian Symposium on Robotics (SBR), and 2024 Workshop on Robotics in Education (WRE). IEEE, 2024.
- Amorim, Johann SJCC, et al. "Design of a Mobile Robot for Educational and Research Purposes." Congresso Brasileiro de Automática-CBA. Vol. 4. No. 1. 2024.
- Amorim, Johann SJCC, et al. "Optimized Kalman Gain for Enhanced Localization in Differential Drive Robots." 2025 Brazilian Symposium on Robotics (SBR) and 2025 Workshop on Robotics in Education (WRE) (AMORIM et al., 2025a)
- Amorim, Johann SJCC, et al. "A ROS2-Based Robotics Course for Undergraduate Program." 2025 Brazilian Symposium on Robotics (SBR) and 2025 Workshop on Robotics in Education (WRE) (PINTO et al., 2025)

Bibliography

3D Robotics. *IRIS Drone*. 2014. https://3dr.com/support/products/iris/. Accessed: November 24, 2024. Citado na página 28.

ABUBAKAR, A.; ALMEIDA, C. F. M.; GEMIGNANI, M. Review of artificial intelligence-based failure detection and diagnosis methods for solar photovoltaic systems. *Machines*, MDPI, v. 9, n. 12, p. 328, 2021. Citado na página 15.

ADDABBO, P. et al. A uav infrared measurement approach for defect detection in photovoltaic plants. In: IEEE. *2017 IEEE International Workshop on Metrology for AeroSpace (MetroAeroSpace)*. [S.l.], 2017. p. 345–350. Citado na página 20.

ALMALKI, F. A. et al. A low-cost platform for environmental smart farming monitoring system based on iot and uavs. *Sustainability*, MDPI, v. 13, n. 11, p. 5908, 2021. Citado na página 14.

AMORIM, J. G. J. C. et al. Optimized kalman gain for enhanced localization in differential drive robots. In: *2025 Brazilian Symposium on Robotics (SBR) and 2025 Workshop on Robotics in Education (WRE)*. [S.l.: s.n.], 2025. p. 102–107. Citado na página 75.

AMORIM, J. S. et al. Collaborative inspection of solar panel farms using yolov5-based computer vision and ugv-uav integration. *Journal of Intelligent & Robotic Systems*, Springer, v. 111, n. 2, p. 66, 2025. Citado 16 vezes nas páginas xi, xii, 25, 27, 29, 30, 32, 33, 34, 35, 63, 64, 65, 66, 68 e 70.

BARANOWSKI, L.; SIWEK, M. Use of 3d simulation to design theoretical and real pipe inspection mobile robot model. *Acta mechanica et automatica*, v. 12, n. 3, p. 232–236, 2018. Citado na página 14.

BERGER, G. S. et al. Cooperative heterogeneous robots for autonomous insects trap monitoring system in a precision agriculture scenario. *Agriculture*, MDPI, v. 13, n. 2, p. 239, 2023. Citado 4 vezes nas páginas 19, 23, 24 e 67.

BHOLA, R. et al. Detection of the power lines in uav remote sensed images using spectral-spatial methods. *Journal of environmental management*, Elsevier, v. 206, p. 1233–1242, 2018. Citado 4 vezes nas páginas 21, 22, 23 e 24.

BOCHKOVSKIY, A.; WANG, C.-Y.; LIAO, H.-Y. M. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*, 2020. Citado na página 21.

CASTRO, G. G. de et al. Dynamic path planning based on neural networks for aerial inspection. *Journal of Control, Automation and Electrical Systems*, Springer, v. 34, n. 1, p. 85–105, 2023. Citado na página 14.

CASTRO, G. G. de et al. Heterogeneous multi-robot collaboration for coverage path planning in partially known dynamic environments. *Machines*, MDPI, v. 12, n. 3, p. 200, 2024. Citado 5 vezes nas páginas 14, 15, 19, 23 e 24.

CHELLAL, A. A. et al. Innovative robot design for cleaning solar panels. In: *11th International Conference on Simulation and Modeling Methodologies, Technologies and Applications, SIMULTECH 2021*. [S.l.: s.n.], 2021. p. 264–270. Citado 4 vezes nas páginas 15, 20, 23 e 24.

COELHO, F. O. et al. Autonomous uav exploration and mapping in uncharted terrain through boundary-driven strategy. *IEEE Access*, v. 12, p. 92464–92483, 2024. Citado na página 14.

DAI, Z. et al. Fast and accurate cable detection using cnn. *Applied Intelligence*, Springer, v. 50, p. 4688–4707, 2020. Citado 2 vezes nas páginas 21 e 22.

DANI, A. P. et al. Human-in-the-loop robot control for human-robot collaboration. *IEEE Control Systems Magazine*, 2024. Tutorial on human intention estimation and safe trajectory control in collaborative robotics. Citado na página 71.

DANTAS, G. M. et al. Dust detection in solar panel using image processing techniques: A review. *Research, Society and Development*, v. 9, n. 8, p. e321985107–e321985107, 2020. Citado na página 21.

DÍAZ, J. J. V. et al. Solar panel detection within complex backgrounds using thermal images acquired by uavs. *Sensors*, MDPI, v. 20, n. 21, p. 6219, 2020. Citado 4 vezes nas páginas 19, 22, 23 e 24.

DÖRFLER, K. et al. Additive manufacturing using mobile robots: Opportunities and challenges for building construction. *Cement and concrete research*, Elsevier, v. 158, p. 106772, 2022. Citado na página 14.

EKICI, S.; KOPRU, M. A. Investigation of pv system cable losses. *International Journal of Renewable Energy Research (IJRER)*, v. 7, n. 2, p. 807–815, 2017. Citado na página 20.

FELSCH, T. et al. Robotized inspection of vertical structures of a solar power plant using ndt techniques. *Robotics*, MDPI, v. 4, n. 2, p. 103–119, 2015. Citado 4 vezes nas páginas 15, 20, 23 e 24.

FENG, H. et al. Automatic tracking method for submarine cables and pipelines of auv based on side scan sonar. *Ocean Engineering*, Elsevier, v. 280, p. 114689, 2023. Citado na página 22.

FLYGUYS. *Drone Inspections for Solar Panels: A Smarter Way to Inspect*. 2024. https://flyguys.com/solar-farm-inspections-using-drones-for-data-collection/. Accessed: April 12 2025. Citado 2 vezes nas páginas 71 e 72.

GAO, Z.; HUANG, Y.; HAN, J. An alternative paradigm for control system design. In: IEEE. *Proceedings of the 40th IEEE conference on decision and control (Cat. No. 01CH37228)*. [S.l.], 2001. v. 5, p. 4578–4585. Citado na página 40.

GIRSHICK, R. et al. Rich feature hierarchies for accurate object detection and semantic segmentation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2014. p. 580–587. Citado na página 21.

GOLIGHTLY, I.; JONES, D. Visual control of an unmanned aerial vehicle for power line inspection. In: IEEE. *ICAR'05. Proceedings., 12th International Conference on Advanced Robotics, 2005.* [S.l.], 2005. p. 288–295. Citado na página 21.

GOUVÊA, J. A. et al. Attitude control of ornithopter wing by using a mimo active disturbance rejection strategy. *Sensors*, MDPI, v. 23, n. 14, p. 6602, 2023. Citado na página 15.

GOUVÊA, J. A.; VASCONCELLOS, M. C. de; ZACHI, A. R. Controle de sistemas incertos via método adrc-um novo paradigma. In: *Congresso Brasileiro de Automática-CBA*. [S.l.: s.n.], 2020. v. 2, n. 1. Citado na página 22.

HABIBIAN, S. et al. Design and implementation of a maxi-sized mobile robot (karo) for rescue missions. *Robomech Journal*, Springer, v. 8, n. 1, p. 1, 2021. Citado na página 14.

HAJJAJ, S. S. H.; SAHARI, K. S. M. Review of research in the area of agriculture mobile robots. In: SPRINGER. *The 8th International Conference on Robotic, Vision, Signal Processing & Power Applications: Innovation Excellence Towards Humanistic Technology*. [S.l.], 2014. p. 107–117. Citado na página 14.

HALDER, S.; AFSARI, K. Robots in inspection and monitoring of buildings and infrastructure: A systematic review. *Applied Sciences*, MDPI, v. 13, n. 4, p. 2304, 2023. Citado na página 15.

HART, P. E.; NILSSON, N. J.; RAPHAEL, B. A formal basis for the heuristic determination of minimum cost paths. *IEEE transactions on Systems Science and Cybernetics*, IEEE, v. 4, n. 2, p. 100–107, 1968. Citado na página 35.

HENDZEL, Z.; RYKAŁA, Ł. Modelling of dynamics of a wheeled mobile robot with mecanum wheels with the use of lagrange equations of the second kind. Zielona Góra: Uniwersytet Zielonogórski, 2017. Citado na página 37.

HERBST, G. A simulative study on active disturbance rejection control (adrc) as a control tool for practitioners. *Electronics*, Multidisciplinary Digital Publishing Institute, v. 2, n. 3, p. 246–279, 2013. Citado 3 vezes nas páginas xi, 22 e 43.

HERNÁNDEZ, J. C. O.; ALMEIDA, D. I. R.; SOLÍS, E. V. G. Overcoming inclined surface challenges in mecanum-wheeled robots using active disturbance rejection control. *Control Theory and Technology*, Springer, p. 1–17, 2025. Citado na página 15.

HERRMANN, W. et al. Qualification of photovoltaic (pv) power plants using mobile test equipment. *IEA-PVPS T13-24: 2021*, 2021. Citado na página 20.

HOFER, W. et al. A robotic tunnel drainage inspection rover. *Transportation Research Procedia*, Elsevier, v. 72, p. 2724–2731, 2023. Citado na página 14.

JAFFERY, Z. A. et al. Scheme for predictive fault diagnosis in photo-voltaic modules using thermal imaging. *Infrared Physics & Technology*, Elsevier, v. 83, p. 182–187, 2017. Citado na página 19.

JENSSEN, R.; ROVERSO, D. et al. Automatic autonomous vision-based power line inspection: A review of current status and the potential role of deep learning. *International Journal of Electrical Power & Energy Systems*, Elsevier, v. 99, p. 107–120, 2018. Citado na página 22.

JENSSEN, R.; ROVERSO, D. et al. Intelligent monitoring and inspection of power line components powered by uavs and deep learning. *IEEE Power and energy technology systems journal*, IEEE, v. 6, n. 1, p. 11–21, 2019. Citado na página 22.

JOCHER, G. *YOLOv5: 1000 FPS on a single GPU*. 2020. https://github.com/ultralytics/yolov5. Accessed: February 24, 2024. Citado na página 21.

JOCHER, G.; CHAURASIA, A.; QIU, J. *Ultralytics YOLOv8*. 2023. https://github.com/ultralytics/ultralytics. Accessed: November 1st, 2024. Citado na página 21.

KAKOGAWA, A.; KOMURASAKI, Y.; MA, S. Shadow-based operation assistant for a pipeline-inspection robot using a variance value of the image histogram. *Journal of Robotics and Mechatronics*, Fuji Technology Press Ltd., v. 31, n. 6, p. 772–780, 2019. Citado na página 14.

KHALED, T. A.; AKHRIF, O.; BONEV, I. A. Dynamic path correction of an industrial robot using a distance sensor and an adrc controller. *IEEE/ASME Transactions on Mechatronics*, IEEE, v. 26, n. 3, p. 1646–1656, 2020. Citado na página 22.

KOENIG, N.; HOWARD, A. Design and use paradigms for gazebo, an open-source multi-robot simulator. In: IEEE. *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*. [S.l.], 2004. v. 3, p. 2149–2154. Citado 2 vezes nas páginas 16 e 28.

LI, C. et al. Yolov6: A single-stage object detection framework for industrial applications. *arXiv preprint arXiv:2209.02976*, 2022. Citado na página 21.

LI, Y.; ZHANG, X.; SHEN, Z. Yolo-submarine cable: an improved yolo-v3 network for object detection on submarine cable images. *Journal of Marine Science and Engineering*, MDPI, v. 10, n. 8, p. 1143, 2022. Citado na página 22.

LIAO, K.; LU, J. Using matlab real-time image analysis for solar panel fault detection with uav. *Journal of Physics: Conference Series*, v. 1509, n. 1, p. 012010, 2020. Citado na página 20.

LIN, L.-C.; SHIH, H.-Y. Modeling and adaptive control of an omni-mecanum-wheeled robot. *Intelligent Control and Automation*, Scientific Research Publishing, v. 4, n. 2, p. 166–179, 2013. Citado na página 39.

MAVLink Contributors. *MAVLink Protocol*. 2022. https://mavlink.io/en/. Accessed: November 24, 2024. Citado na página 28.

MEIER, L. et al. *MAVLink Protocol Specifications*. n.d. https://mavlink.io. Accessed: April 12, 2025. Citado na página 71.

MERIBOUT, M. et al. Solar panel inspection techniques and prospects. *Measurement*, Elsevier, v. 209, p. 112466, 2023. Citado na página 15.

Mornsun Power. *Exploring the 3.3V to 5V Conversion: Methods and Solutions*. 2023. https://www.mornsun-power.com/html/news-detail/blog-posts/213.html. Accessed: 2024-09-29. Citado na página 15.

MOUSAVI, S.; FARAHANI, G. Introducing a new method of automatic cleaning of the pv array surface using a suction robot. *Mechatronics*, Elsevier, v. 85, p. 102845, 2022. Citado na página 20.

NITTA, Y. et al. Damage assessment methodology for nonstructural components with inspection robot. *Key Engineering Materials*, Trans Tech Publ, v. 558, p. 297–304, 2013. Citado na página 14.

PFEIFFER, C.; SCARAMUZZA, D. Human-piloted drone racing: Visual processing and control. *IEEE Robotics and Automation Letters*, v. 6, n. 2, p. 3188–3195, 2021. Citado na página 71.

PINTO, M. F. et al. A ros2-based robotics course for undergraduate program. In: *2025 Brazilian Symposium on Robotics (SBR) and 2025 Workshop on Robotics in Education (WRE)*. [S.l.: s.n.], 2025. p. 308–313. Citado na página 75.

PINTO, M. F. et al. A framework for analyzing fog-cloud computing cooperation applied to information processing of uavs. *Wireless Communications and Mobile Computing*, v. 2019, n. 1, p. 7497924, 2019. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1155/2019-/7497924>https://onlinelibrary.wiley.com/doi/abs/10.1155/2019/7497924. Citado na página 14.

PRUTHVIRAJ, U. et al. Solar photovoltaic hotspot inspection using unmanned aerial vehicle thermal images at a solar field in south india. *Remote Sensing*, MDPI, v. 15, n. 7, p. 1914, 2023. Citado na página 15.

RAMOS, G. S. et al. Hybrid methodology based on computational vision and sensor fusion for assisting autonomous uav on offshore messenger cable transfer operation. *Robotica*, Cambridge University Press, v. 40, n. 8, p. 27862814, 2022. Citado na página 14.

REA, P.; OTTAVIANO, E. Hybrid inspection robot for indoor and outdoor surveys. *Actuators*, v. 12, n. 3, p. 108, 2023. Citado na página 14.

REDMON, J. You only look once: Unified, real-time object detection. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2016. p. 779–788. Citado 2 vezes nas páginas 21 e 29.

REDMON, J.; FARHADI, A. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018. Citado na página 21.

ROBOTICS, C. *Autonomous Solar Farm Inspection with Husky Observer*. 2024. https://clearpathrobotics.com/blog/2024/08/autonomous-solar-farm-inspection-with-husky-observer/. Accessed: April 12 2025. Citado na página 72.

ROLDÁN, J. J. et al. Heterogeneous multi-robot system for mapping environmental variables of greenhouses. *Sensors*, MDPI, v. 16, n. 7, p. 1018, 2016. Citado 3 vezes nas páginas 19, 23 e 24.

SHAPOVALOV, D.; PEREIRA, G. A. Tangle-free exploration with a tethered mobile robot. *Remote Sensing*, MDPI, v. 12, n. 23, p. 3858, 2020. Citado na página 14.

SILVA, Y. M. da et al. Computer vision based path following for autonomous unmanned aerial systems in unburied pipeline onshore inspection. *Drones*, MDPI, v. 6, n. 12, p. 410, 2022. Citado na página 29.

SITECAPTURE. *How Drone Inspections are Replacing Traditional Solar Inspections*. 2025. https://sitecapture.com/essential-solar-site-survey-tools/. Accessed: April 12 2025. Citado na página 71.

SONG, B.; LI, X. Power line detection from optical images. *Neurocomputing*, Elsevier, v. 129, p. 350–361, 2014. Citado na página 21.

SONG, X. et al. Fuzzy wavelet neural adaptive finite-time self-triggered fault-tolerant control for a quadrotor unmanned aerial vehicle with scheduled performance. *Engineering Applications of Artificial Intelligence*, Elsevier, v. 131, p. 107832, 2024. Citado na página 14.

Stanford Artificial Intelligence Laboratory et al. *Robotic Operating System*. https://www.ros.org. Citado na página 28.

SZREK, J. et al. Accuracy evaluation of selected mobile inspection robot localization techniques in a gnss-denied environment. *Sensors*, MDPI, v. 21, n. 1, p. 141, 2020. Citado na página 15.

TAHERI, H.; QIAO, B.; GHAEMINEZHAD, N. Kinematic model of a four mecanum wheeled mobile robot. *International journal of computer applications*, Foundation of Computer Science, v. 113, n. 3, 2015. Citado na página 37.

VENKATNIKHIL, A.; RAVICHANDRAN, S.; KUMAR, N. Rover robot for solar panel cleaning and monitoring. In: IEEE. *2022 IEEE North Karnataka Subsection Flagship International Conference (NKCon)*. [S.l.], 2022. p. 1–5. Citado na página 15.

WANG, C.-Y.; BOCHKOVSKIY, A.; LIAO, H.-Y. M. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. [S.l.: s.n.], 2023. p. 7464–7475. Citado na página 21.

WANG, J.; UEDA, T. A review study on unmanned aerial vehicle and mobile robot technologies on damage inspection of reinforced concrete structures. *Structural Concrete*, Wiley Online Library, v. 24, n. 1, p. 536–562, 2023. Citado na página 14.

YAHYA, Z. et al. Applied imagery pattern recognition for photovoltaic modules inspection: A review on methods, challenges and future development. *Sustainable Energy Technologies and Assessments*, Elsevier, v. 52, p. 102071, 2022. Citado na página 15.

ZACHI, A. R. et al. Analysis and control of ball displacement in air levitation experiment by using a disturbance rejection-based technique. *IEEE Access*, IEEE, 2025. Citado na página 15.

ZANELLA, R. *Electric Wires Image Segmentation*. Kaggle, 2023. Accessed: 2024-11-20. Disponível em: <https://www.kaggle.com/datasets/zanellar/electric-wires-image-segmentation>https://www.kaggle.com/datasets/zanellar/electric-wires-image-segmentation. Citado 2 vezes nas páginas xi e 33.

ZANELLA, R. et al. Auto-generated wires dataset for semantic segmentation with domain-independence. In: IEEE. *2021 International Conference on Computer, Control and Robotics (ICCCR)*. [S.l.], 2021. p. 292–298. Citado na página 31.

ZEITZ, M. The extended luenberger observer for nonlinear systems. *Systems & Control Letters*, Elsevier, v. 9, n. 2, p. 149–156, 1987. Citado na página 41.

ZHANG, Z. et al. The fault analysis of pv cable fault in dc microgrids. *IEEE Transactions on Energy Conversion*, IEEE, v. 34, n. 1, p. 486–496, 2018. Citado na página 20.