People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research

UNIVERSITY MOHAMED KHIDER BISKRA
Faculty of Exact Sciences, Science of Nature and Life
Department of Computer Science

A thesis submitted in fulfillment of the requirements for the Master's degree in

COMPUTER SCIENCE

Speciality: Artificial Intelligence & Image and Artificial Life

Ordre N° : IVA StartUp 01/M2/2024

Biological Control of Greenhouses using AI Techniques

Authors: HARROUZ Moundher & DJOUDI Achraf Abdelmajid

Supervisor: Dr. BEN AISSA Yousra

Members of the Jury:

Fekraoui Farah	MCA	President
Babahenini Djihane	MCB	Examiner

Academic year : **2023 – 2024**

Declaration of Authorship

We, HARROUZ Moundher & DJOUDI Achraf Abdelmajid, declare that this thesis titled, Biological Control of Greenhouses using AI Techniques and the work presented in it are our own. We confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where we have consulted the published work of others, this is always clearly attributed.
- Where we have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely our own work.
- We have acknowledged all main sources of help.
- Where the thesis is based on work done jointly with others, we have made clear exactly what was done by us and what was contributed by others outside of our authorship.

Signed: HARROUZ Moundher & DJOUDI Achraf Abdelmajid

Date: 24-06-2024

Abstract

Greenhouse pest management is crucial for agricultural productivity, but traditional methods can be environmentally harmful and labor-intensive. This thesis presents a novel, integrated approach leveraging artificial intelligence (AI) and Internet of Things (IoT) technology to automate the identification and elimination of pests in greenhouse settings. A smart camera system enables real-time pest identification. Images captured by the camera are transmitted to a web server for real-time analysis using an AI model. The server then sends the detection results and corresponding instructions to a web application for visualization and analysis, as well as to a robotic response mechanism for targeted pest elimination. The developed prototype demonstrates the feasibility and efficacy of this autonomous system, offering a significant advancement towards sustainable and efficient greenhouse pest management. Future research will focus on system refinement and expanded pest detection capabilities.

Keywords: Artificial Intelligence (AI), Internet Of Things (IOT), Sustainable Agriculture, Pest Control

Acknowledgements

First of all, We would like to thank God Almighty for granting us the strength, perseverance, and guidance to complete this thesis together.

We are deeply grateful to our thesis advisor, Dr. Ben Aissa Youssra, for her unwavering support, invaluable insights, and meticulous guidance throughout this research journey. Her expertise and encouragement were instrumental in shaping this collaborative work.

We are sincerely thankful to all the individuals who contributed to this project, whether through their academic support, technical assistance, or personal encouragement. Your contributions have been priceless.

Finally, we would like to express our deepest gratitude to our families and friends for their unwavering belief in us and their constant support. Your love and encouragement have been our constant source of motivation.

Contents

List of Figures

List of Tables

General Introduction

General Context

The convergence of Artificial Intelligence (AI) and the Internet of Things (IoT) is revolutionizing the agricultural sector, promising to address long-standing challenges in crop management and boost productivity. This technological synergy, often referred to as "Smart Farming" or "Precision Agriculture," leverages a network of sensors, devices, and intelligent algorithms to collect, analyze, and act upon data from the field in real-time.

A prime example of this AI-IoT collaboration is in the realm of pest detection and management. Traditional pest control methods, often reactive and imprecise, have resulted in significant crop losses and excessive pesticide use. However, the integration of AI with IoT is transforming this landscape.

Modern pest monitoring systems deploy an array of IoT sensors, including cameras and environmental monitors, throughout the fields. The data captured by these sensors is transmitted via communication networks to a centralized AI platform. Here, sophisticated algorithms, often based on machine learning or deep learning techniques, analyze the data to rapidly identify and classify pest infestations. This early detection allows for timely and targeted interventions, minimizing crop damage and reducing the need for blanket pesticide applications.

The advantages of this AI-IoT approach are manifold. Continuous monitoring ensures early pest detection, enabling proactive measures. AI-driven analysis provides accuracy and speed in pest identification, far exceeding human capabilities. Furthermore, by automating the response through robotic systems or precision spraying, farmers can significantly reduce labor costs and pesticide usage, leading to more sustainable and environmentally friendly practices.

However, the successful implementation of such systems requires careful consideration of various factors. These include the selection and placement of sensors, data transmission protocols, the computational infrastructure for AI model training and inference, and the seamless integration of diverse components into a unified system.

The data collected by AI-IoT systems can be further utilized for various agricultural applications, such as soil analysis, irrigation optimization, and yield prediction, promoting a more data-driven and sustainable approach to farming.

By harnessing the power of AI and IoT, farmers are empowered to make informed decisions based on real-time, data-driven insights, ushering in a new era of precision agriculture that promises to transform the way we grow food.

Problem Statement

The development of an intelligent plant monitoring system, combining deep learning-based pest detection, imaging, targeted pest control mechanisms, and user-friendly desktop applications, holds immense potential for revolutionizing agricultural practices. By integrating cutting-edge technologies, this innovative solution aims to enhance crop protection by providing real-time pest identification, accurate infestation assessments, precise targeted spraying, and comprehensive monitoring tools. However, several challenges need to be addressed to ensure the successful implementation and usability of such an advanced system.

One critical challenge in pest management is accurately detecting and identifying various pest species. Traditional methods often rely on manual inspection or periodic scouting, which can be time-consuming, labor-intensive, and prone to human error. By incorporating deep learning models and imaging, the intelligent monitoring system can automatically detect and classify pests with high accuracy, enabling early intervention and targeted control measures.

Another problem to address is the efficient and targeted application of pesticides or control agents. Indiscriminate spraying can lead to environmental concerns, resistance development, and unnecessary chemical exposure. The monitoring system should leverage targeted spraying mechanisms that can precisely deliver control agents to infested areas, reducing waste and minimizing the overall environmental impact.

Furthermore, the development of a user-friendly desktop application for monitoring and control is crucial. This application should provide intuitive interfaces, real-time data visualization, and comprehensive reporting tools. Ensuring ease of use and seamless integration with existing agricultural practices will facilitate the adoption and engagement of farmers and agricultural professionals.

Contributions

To address the challenges outlined above, we divide our contribution into three aspects as follows:

- The first one, Integrated Imaging and Targeted Spraying System, consists on developing an advanced integrated system, comprising a camera unit and a targeted spraying mechanism, that can be mounted on a mobile robotic platform or an unmanned aerial vehicle (UAV).

- The second one, Communication, Data Processing, and Desktop Application, aims to propose a comprehensive system for processing the acquired plant images, detecting pests, and facilitating effective monitoring and control measures through a user-friendly desktop application.

- And the third one, Machine Learning Model, focuses on the development of a robust machine learning model for precise pest detection and classification. This part represents the core challenge of accurate pest identification, which is crucial for effective monitoring and targeted control measures.

The proposed contribution addresses critical aspects of the pest monitoring and control system, including integrated image acquisition and targeted spraying, data processing, a user-friendly desktop application, and a robust machine learning model for pest detection and classification. By combining these innovative solutions, this research aims to provide a comprehensive and effective approach to enhancing agricultural practices, minimizing crop losses, and promoting sustainable pest management strategies.

Chapter 1

Traditional vs Smart Greenhouse Agriculture

Contents

1.1 Introduction

This chapter delves into the critical issues faced by greenhouse farmers, emphasizing the persistent threat posed by pests to production and profitability. It examines the impact of these challenges on crop health and economic outcomes, highlighting the need for innovative solutions. Furthermore, the chapter explores the transformative role of AI and IoT in modern agriculture, showcasing their potential to revolutionize greenhouse farming practices. Finally, an overview of related works provides a comprehensive understanding of the current state of research and development in this field.

1.2 Greenhouse Pests: A Threat to Production and Profitability

In modern agriculture, greenhouses serve as controlled environments for crop production, providing optimal conditions for plant growth and protection from external elements. However, despite these controlled conditions, greenhouses are susceptible to infestations by a variety of insect pests.

1.2.1 Common Insect Pests in Greenhouse Environments

Greenhouse environments are susceptible to various insect pests that can significantly impact crop health and yield. Key pests include:

- **Leaf Miner(Tuta absoluta):** Small moths that primarily infest tomato plants, causing damage by feeding on leaves, stems, and fruits. This leads to reduced yield and plant death in severe infestations [5].
- **Aphids:** Soft-bodied insects that feed on a wide range of crops, causing stunted growth, leaf distortion, and honeydew secretion, which can lead to sooty mold growth [6].
- **Whiteflies:** Moth-like insects, they infest various crops, feeding on sap and causing yellowing, wilting, and leaf distortion. Their honeydew excretion also contributes to sooty mold growth [7].
- **Thrips:** Slender insects that damage plants by puncturing tissues and sucking out cell contents, leading to scarring, distorted growth, and potential virus transmission [8].
- **Mealybugs:** Waxy-coated, these insects infest a wide range of plants, causing damage by sucking sap and excreting honeydew, which can lead to sooty mold growth and attract ants [9].
- **Gryllus(Crickets):** While primarily outdoor insects, crickets can enter greenhouses and damage crops by consuming seedlings, young plants,

and foliage. They may also cause harm by tunneling into soil [10].

Understanding the characteristics, behavior, and damage patterns of these pests is crucial for developing effective pest management strategies in greenhouse environments.

1.2.2 Impact on Crop Health, Yield, and Economic Viability

Insect pests present in greenhouse environments can have profound effects on crop health and yield, leading to economic losses and reduced productivity. Understanding the impact of these pests is essential for implementing effective pest management strategies. Below are the key ways in which insect pests can affect crop health and yield:

Economic Losses

Insect pests can cause significant economic losses to greenhouse farmers through reduced crop yields, quality degradation, and increased production costs associated with pest control measures. Yield losses may result from direct feeding damage to plant tissues, which can inhibit growth, development and the production [11].

Additionally, the cost of implementing pest management strategies, such as purchasing and applying chemical pesticides or deploying biological control agents, can contribute to financial burdens for growers [11].

Reduced Yields

The feeding activities of insect pests can directly impact plant physiology and metabolism, leading to reduced photosynthesis, nutrient uptake, and water absorption.

Severe infestations may result in defoliation, wilting, and premature senescence of plants, further exacerbating yield losses.

Crop plants may also allocate resources towards defense mechanisms, such as producing secondary metabolites or structural barriers, rather than investing in growth and reproduction [12].

Quality Degradation

Insect pests can compromise the quality and marketability of greenhouse crops by causing physical damage, deformities, or contamination.

Feeding damage, such as leaf stippling, curling, or necrosis, can render affected plant parts unattractive or unsuitable for sale in fresh produce markets [13].

Transmission of Diseases

Some insect pests act as vectors for plant pathogens, including viruses, bacteria, and fungi, which can cause diseases in greenhouse crops.

Pests that feed on plant tissues may introduce pathogens directly into the plant's vascular system, leading to systemic infections and disease spread throughout the crop.

Disease transmission by insect vectors can exacerbate the impact of pest infestations on crop health and yield, potentially resulting in crop losses due to disease outbreaks [14].

1.3 Challenges Faced by Greenhouse Farmers

Greenhouse farming, while offering numerous advantages such as extended growing seasons, controlled environments, and higher crop yields, is not without its challenges. These challenges can significantly impact productivity, profitability, and sustainability. Among the most pressing issues faced by greenhouse farmers is pest and disease management, which encompasses two critical hurdles:

1.3.1 Pest Identification

Early and accurate pest identification is crucial for effective management, but it can be a complex and time-consuming process. Traditional methods often rely on visual inspection of plants and sending samples to laboratories for analysis, which can lead to delays in diagnosis and allow infestations to spread. Early detection of pests and diseases is crucial for effective management in greenhouse crops, but current diagnostic methods are often slow and labor-intensive. The development of rapid and accurate diagnostic tools, such as molecular techniques or biosensors, is an active area of research that aims to address this challenge [15].

1.3.2 Pest Elimination

Even with accurate identification, eliminating pests in a greenhouse environment presents its own set of challenges. Many farmers resort to broad-spectrum chemical pesticides, which can have negative impacts on beneficial insects, human health, and the environment. The intensive use of pesticides in greenhouses can lead to the development of pesticide resistance in pest populations, as well as negative effects on non-target organisms and human health. The development and implementation of integrated pest management (IPM) strategies, which combine biological, cultural, and chemical control methods, are essential for sustainable pest management in greenhouses [16].

By addressing these challenges in pest and disease management, greenhouse farmers can work towards more productive, profitable, and environmentally sustainable operations. Continued research and innovation in pest diagnostics and control methods are crucial for the future of greenhouse farming.

1.4 AI in Agriculture

Artificial Intelligence (AI) has emerged as a transformative force in agriculture, revolutionizing various aspects of farming practices. From precision farming to crop monitoring and yield prediction, AI technologies have significantly enhanced efficiency, productivity, and sustainability in the agricultural sector. In the context of biological control in greenhouses, AI techniques play a pivotal role in optimizing pest management strategies, minimizing environmental impact, and ensuring crop health and quality.

1.4.1 Applications of AI in agriculture

In recent years, the agricultural sector has increasingly turned to Artificial Intelligence (AI) to address various challenges hindering maximum yield. These challenges include inadequate soil treatment, disease and pest outbreaks, the need for handling large volumes of data, low productivity, and a gap in knowledge between farmers and technological advancements. AI offers promising solutions due to its adaptability, high performance, precision, and cost-effectiveness [17].

This overview will delve into AI applications in soil management, crop management, weed management, and disease management within agriculture. It will also explore the strengths and limitations of these applications, highlighting the role of expert systems in enhancing productivity .

Crop Yield Improvement: AI can help increase crop yield by up to 30% through precision farming techniques that optimize sowing, fertigation, and pest control. AI-powered systems can analyze weather patterns, soil conditions, and other factors to determine the optimal sowing date and depth for specific crops. They can also monitor crop growth and provide recommendations for fertilization and irrigation to maximize yield. Additionally, AI can help detect and manage pests and diseases, further improving crop health and yield [18].

Soil Analysis: AI can analyze soil samples to determine the optimal mix of nutrients for specific crops. By analyzing the chemical and physical properties of soil, AI-powered systems can provide recommendations for fertilizers, lime, and other soil amendments. This can help farmers improve soil health, reduce input costs, and increase crop yields [18].

Image-based Insight Generation: AI can analyze images from drones and other sources to monitor crop health, detect diseases, and identify the readiness of crops for harvest. By analyzing high-resolution images of crops, AI-powered systems can detect signs of stress, disease, or nutrient deficiencies. They can also identify the ripeness of fruits and vegetables, allowing farmers to optimize harvest times and reduce waste. [18]

Resource Optimization: AI can optimize the use of water, fertilizer, and pesticides to reduce waste and improve crop yields. By analyzing weather patterns, soil conditions, and other factors, AI-powered systems can provide recommendations for irrigation, fertilization, and pest control. This can help farmers reduce input costs, conserve resources, and improve crop health [18].

Crop Health Monitoring: AI can monitor crop health throughout the growing season and provide real-time alerts for any abnormalities. By analyzing data from sensors, drones, and other sources, AI-powered systems can detect signs of stress, disease, or nutrient deficiencies. They can also provide recommendations for corrective actions, such as adjusting irrigation or fertilization schedules [18].

Automation in Irrigation: AI can automate irrigation systems to conserve water and increase crop yields. By analyzing weather patterns, soil conditions, and other factors, AI-powered systems can optimize irrigation schedules and reduce water waste. They can also detect leaks and other issues in irrigation systems, allowing farmers to address them promptly [18].

Plant Stress Recognition: AI can recognize plant stress levels in various growth stages and provide recommendations for improving crop health. By analyzing data from sensors and other sources, AI-powered systems can detect signs of stress, such as changes in temperature, humidity, or light levels. They can also provide recommendations for addressing stress, such as adjusting irrigation [18].

1.5 IOT in Agriculture

The Internet of Things (IoT) is a transformative technology that is reshaping the agricultural sector. By seamlessly integrating physical devices with digital technologies, IoT is providing efficient and reliable solutions for modernizing farming practices. With IoT-based solutions, farms and greenhouses can now be automatically monitored and maintained with minimal human involvement. From precision farming to crop monitoring and yield prediction, IoT technologies are enhancing efficiency, productivity, and sustainability in the agricultural sector [19] .

1.5.1 IoT Architecture

Several technological layers make up the Internet of Things architecture, as listed below:

- Smart device/Perception layer: This is the foundational layer of the IoT architecture, tasked with monitoring changes in the physical state of connected objects. Sensors are the primary components of this layer, responsible for detecting and collecting data such as temperature, air quality, speed, humidity, pressure, flow, movement, voltage, and more. The collected data is then transferred to the cloud layer for storage and further processing.
- Communication/Transport Layer: Responsible for facilitating communication between different layers of the IoT architecture, the communication layer ensures seamless data transfer. Data collected by sensors is transmitted to the cloud or the service and application layer through routers, switches, and gateways. Various protocols are used to connect different IoT devices and enable the transmission of data to higher layers.
- Cloud/Processing Layer: Often referred to as the IoT system processing unit, the cloud layer receives data from sensors and devices. This layer is responsible for data processing, analysis, and storage. Typically, a data center serves as a central server to manage the data generated by the network edge.
- Management Layer: Operating and monitoring all other layers, the management layer utilizes cloud management tools for effective implementation and management of the IoT architecture.
- Services and Applications Layer: The topmost layer of the IoT architecture, this layer offers a wide range of services and applications including security, data collection, data analysis, and visualization. The services and applications provided are determined by specific use scenarios and the functionality desired by end-users [20].

1.5.2 IoT Characteristics

The Internet of Things (IoT) is a sophisticated network that brings together various real-world domains, each with its own unique characteristics. Here are the main ones:

- Distribution: IoT systems involve deploying numerous devices across various geographical locations.
- Computation Capability: IoT devices vary in computational power, ranging from small embedded sensors to powerful high-end servers.

- Large Quantities of Devices and Data: The rate of data produced by smart devices is exponentially rising due to widespread deployments and expansion of IoT applications.

- Heterogeneity: IoT systems consist of multiple types of devices with different hardware and software, following different standards and protocols.

- Dynamicity: IoT environments are very dynamic, with devices being added, terminated, connected, or disconnected from networks at any time.

- Mobility : Some devices, such as smartphones, have a high degree of mobility, implying that they can be under different domains of administration throughout their life cycle.

- Ubiquity of Services: IoT offers an unprecedented massive scale of service provisions which can be accessible across the globe. Many of them offer similar functionalities with different requirements and Quality of Service [21].

1.5.3 Agricultural IoT Structure

IoT-based smart farming consists of four major components:

- Physical Structure: The physical structure forms the backbone of IoT-based smart farming, ensuring precision agriculture by preventing unwanted occurrences. It controls sensors, actuators, and devices, ensuring seamless operation and management of the farm.

- Data Acquisition: Data acquisition is divided into two main components: IoT data acquisition and standard data acquisition. IoT data acquisition employs protocols like MQTT, Websocket, AMQP, CoAP, DDS, and HTTP. Standard data acquisition utilizes protocols such as ZigBee, WIFI, LoraWan, SigFox, and ISOBUS. These protocols facilitate the seamless gathering of data from various sensors and devices deployed across the farm.

- Data Processing: Data processing plays a crucial role in IoT-based smart farming. It involves various features such as image or video processing, data loading, decision support systems, and data mining. These processes enable the extraction of meaningful insights from the vast amount of data collected from the farm's sensors and devices.

- Data Analytics: Data analytics focuses on monitoring and controlling various aspects of smart farming. Monitoring includes applications like Livestock Monitoring, Field Monitoring, and Greenhouse Monitoring. Livestock Monitoring involves tracking parameters like temperature, heart rate, and digestion using sensors. Field Monitoring reports conditions such as soil richness, temperature, humidity, gas, pressure, and crop diseases. Greenhouse Monitoring automates climate parameter measurements according to plant requirements, minimizing manual intervention and optimizing crop growth [19].

1.5.4 Agricultural IoT Technologies

Numerous technologies are employed in IoT agricultural solutions, technologies that have significantly modernized IoT agricultural services such as:

- Cloud And Edge Computing: The integration of IoT and cloud computing in agriculture ensures widespread access to shared resources. Cloud computing plays a crucial role in meeting diverse agricultural needs and executing operations efficiently. Cloud-based software architecture has been proposed to enhance the accuracy of processing and retrieving agricultural information and tasks. In IoT, edge computing is seen as a solution for facilitating data processing at the source, such as sensors and actuators. Edge computing, also known as fog computing, serves as the backbone of cloud computing, deployed according to the features and requirements of smart farming [22].

- Big Data And Machine Learning: Big data, comprising vast amounts of essential data generated by agricultural sensors, enables efficient crop monitoring at various stages. A systematic review of big data analysis in agriculture has been conducted. Neural networks are popular for providing optimal solutions at high speeds. Advanced neural network principles and technology are used for intrusion detection. Moreover, neural networks offer a detection module and data training. Deep neural networks have been utilized to develop an IoT-based hydroponic system [23].

- Communication Networks And Protocols: IoT agricultural networks incorporate different long-range and short-range communication technologies. These technologies aid in designing crop or field monitoring sensors and devices. Communication protocols are essential for exchanging agricultural data and information across the network, forming the backbone of IoT agricultural systems and applications [24].

- Robotics: Several Agribots have been developed for smart farming, reducing the need for manual labor by increasing work speed through advanced techniques. These robots perform basic functions such as weeding, spraying, and sowing. IoT is used to control these robots, enhancing crop productivity and resource utilization. A multi-sensor robotics approach has been proposed for characterizing and mapping the ground [25].

1.5.5 Agricultural IoT Application Domains

The Internet of Things (IoT) is revolutionizing agriculture by providing farmers with innovative tools and solutions to enhance productivity, optimize resource management, and mitigate risks. In the context of pest control and precision pesticide application, the following IoT applications are particularly relevant:

- **Crop Monitoring and Pest Detection:** Precision farming helps farmers improve productivity by automating and optimizing various tasks. It involves using IoT sensors to measure soil quality, weather conditions, and moisture levels, and to plan harvesting techniques effectively [26].
- **Precision Pesticide Application:** IoT-enabled devices can analyze data from various sources to determine the optimal timing and dosage for pesticide application. This precision approach reduces pesticide wastage, minimizes environmental impact, and ensures effective pest control [27].
- **Robotics And Drone Technology:** Agricultural drones and robots equipped with sensors and imaging capabilities can be used to survey large fields, identify areas with pest problems, and apply pesticides with precision. They can also be used to monitor crop health and growth, providing valuable data for decision-making [25].
- **Data Analytics and Decision Support:** IoT platforms can collect and analyze data from various sensors and devices to provide farmers with insights into crop health, pest pressure, and environmental conditions. This data can be used to develop predictive models and decision-support tools that help farmers make informed decisions about pest management strategies [28].
- **Remote Farm Management:** IoT-enabled devices and platforms allow farmers to remotely monitor and control various aspects of their operations, including irrigation systems, pesticide applications, and crop monitoring systems. This remote access improves efficiency, reduces labor costs, and allows farmers to respond quickly to changing conditions [29].

By integrating these IoT applications into their operations, farmers can achieve more effective and sustainable pest control practices, reduce pesticide use, and improve crop yields.

1.6 Related Works

Several research studies and projects have focused on leveraging advanced technologies for pest detection and management in agricultural settings, including greenhouses. Here are some notable related works in the field

1.6.1 RobHortic: A Field Robot to Detect Pests and Diseases in Horticultural Crops by Proximal Sensing

Cubero et al, (2020) introduce RobHortic. RobHortic is a remotely-controlled field robot designed for the detection of pests and diseases in horticultural crops through proximal sensing. Developed primarily for use in carrot fields, the robot employs advanced imaging technologies to identify asymptomatic plants infected with 'Candidatus Liberibacter solanacearum' (CaLsol). RobHortic's construction features a frame with four wheels capable of absorbing terrain irregularities, with adjustable width between 100 and 200 cm to fit different crop row widths. The frame includes a closed structure that houses the cameras, sensors, and an industrial computer, along with four 100 W halogen spotlights and a tarp to counteract natural light variability. Two 24 V 250 W DC motors drive the robot, powered by a 24 V 10 Ah lithium battery recharged by a 2000 W inverter generator, and it is controlled via a wireless radio-controller that monitors parameters such as speed, distance traveled, and battery level [1].

The robot's sensors and imaging systems include a multispectral camera, three DSLR cameras (including modified versions for near-infrared imaging), a hyperspectral imaging system that acquires images in 133 bands between 410 and 1130 nm, along with a GNSS for geolocating images. RobHortic captures images approximately every 80 cm while moving at a speed of 1 m/s, with the GNSS recording location data at 25 Hz for precise geolocation. Custom software synchronizes image acquisition with the robot's movement, storing the images along with GNSS data for analysis. Using Partial Least Squares-Discriminant Analysis (PLS-DA), RobHortic achieved detection rates of 66.4% in the lab and 59.8% in the field, with other methods like Linear Discriminant Analysis (LDA) and Support Vector Machine (SVM) also tested [1].

Field tests demonstrated the robot's ability to capture necessary data across multiple campaigns, although vegetation index maps created from multispectral images were not effective for detecting CaLsol. Despite this, RobHortic's total material cost of under €5000 makes it an affordable solution for crop inspection, with a design adaptable to different crops and conditions and the ability to add or change sensors as needed [1].

FIGURE 1.1: A remotely-driven RobHortic operating in a carrot field. (a) the external appearance of the robot. (b): inside from the plant point of view [1]

1.6.2 Identification of fruit tree pests with deep learning on embedded drone to achieve accurate pesticide spraying

Chen et al. (2021) introduce an innovative approach to pest management in longan orchards, utilizing a dual-drone system and edge computing for real-time pest detection and precise pesticide application. The system employs a reconnaissance drone equipped with an embedded NVIDIA Jetson TX2 module to capture images of Tessaratoma papillosa, a significant pest in longan cultivation. The Tiny-YOLOv3 model, running on the TX2, processes these images in real-time, identifying the pest's life stage and location. This information is then used to plan an optimized flight path for a separate agricultural drone responsible for pesticide spraying [2].

The system's effectiveness is demonstrated through field experiments in sloped longan orchards, where it achieves a pest control rate of over 95%.

Notably, the approach reduces pesticide use by 70% and water consumption by 12.5% compared to traditional manual spraying methods. Additionally, the automated system significantly decreases labor requirements, making it a promising solution for addressing labor shortages in agriculture. The study's findings highlight the potential of integrating drone technology, edge computing, and deep learning for sustainable and efficient pest management practices [2].

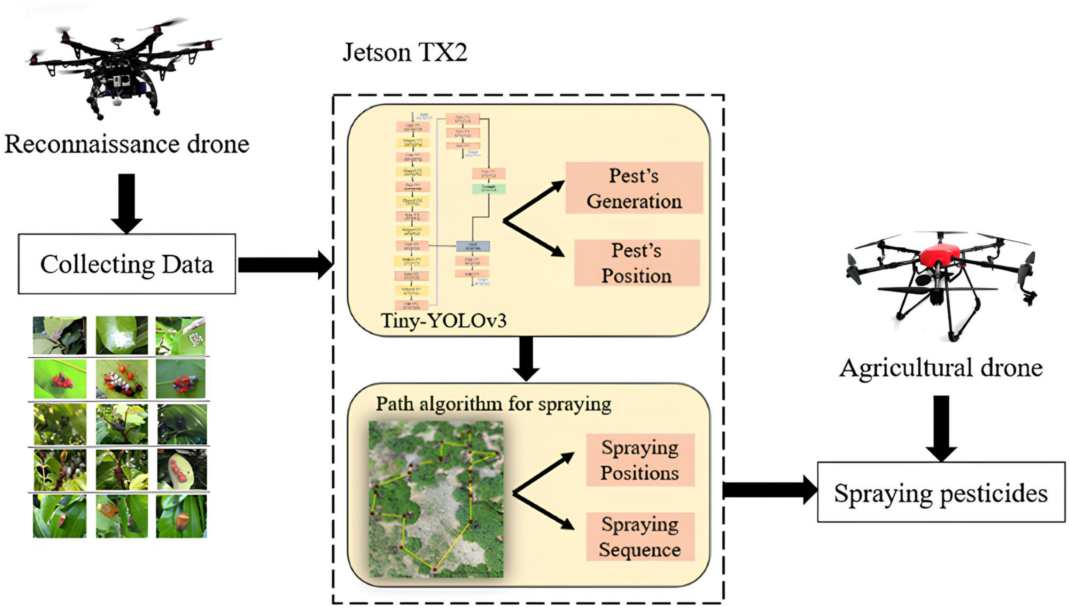


FIGURE 1.2: The system architecture flow chart [2]

1.6.3 Automatic pest identification system in the greenhouse based on deep learning and machine vision

The research focused on enhancing the YOLOv5 model to improve pest detection accuracy specifically in greenhouse environments. Greenhouses present unique challenges for pest detection due to the controlled yet varied conditions. The researchers aimed to refine the YOLOv5 model to address these challenges effectively.

The enhanced model demonstrated significant improvements in detecting various pests, including leaf miners, fruit flies, aphids, and houseflies. The detection accuracies achieved were remarkable, with 99% for leaf miners and fruit flies, 98% for aphids and houseflies, and 97% for whiteflies. Despite these high accuracy rates, the study also identified several limitations. The model struggled with detecting light-colored tobacco whiteflies and tiny thrips, which are particularly challenging due to their size and color. Additionally, there were instances of misidentification between houseflies and leaf miners, likely due to their similar coloration and shape. These findings highlight the need for further refinement of the model to address these specific detection challenges and ensure more reliable pest identification in greenhouse settings [3].

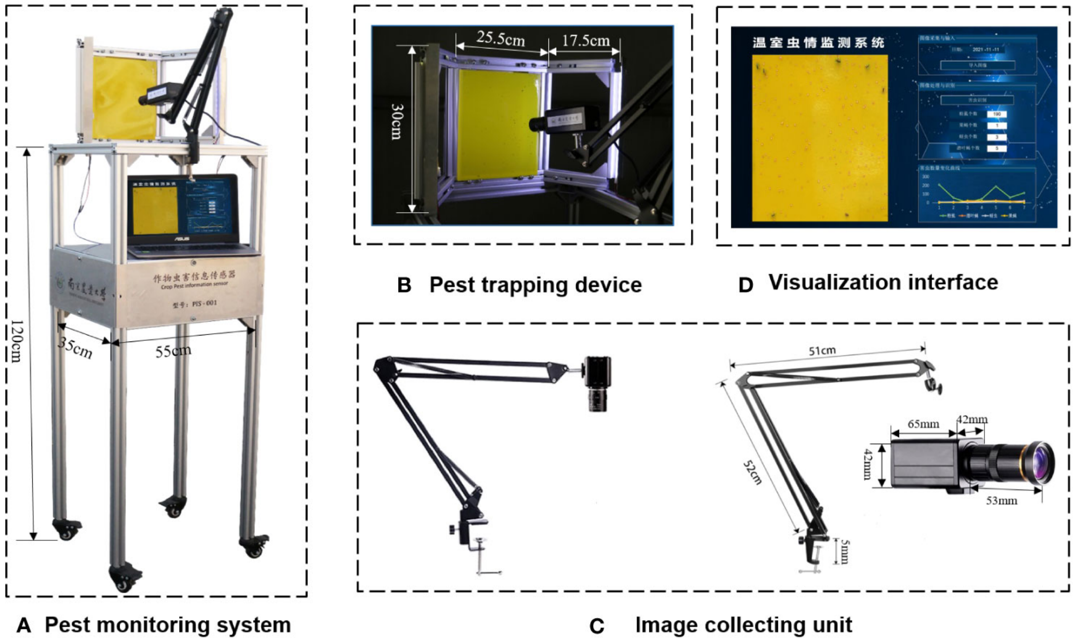


FIGURE 1.3: Automatic pest identification and monitoring system with LED trap lamp, sticky paper and image acquisition system [3]

1.7 Conclusion

In conclusion, moving from traditional to smart greenhouses has shown great promise in solving many problems that farmers face. With AI and IoT technologies, farmers can better manage pests, monitor crop health, increase yields, and improve their overall profitability. These advancements make greenhouse farming more efficient and sustainable.

Next, we'll explore how Artificial Intelligence, specifically Object Detection, can help even more. This technology can accurately and quickly identify pests, diseases, and the condition of crops, making greenhouse management even more precise and effective.

Chapter 2

Artificial Intelligence's Object Detection

Contents

2.1 Introduction

As the agricultural sector strives for sustainability and efficiency, ensuring effective pest and disease management within greenhouses is critical. This chapter delves into the transformative potential of artificial intelligence (AI), particularly machine learning (ML) and deep learning (DL) techniques, to revolutionize greenhouse crop protection strategies. By providing both theoretical and practical insights, this chapter aims to equip readers with the knowledge necessary to implement AI-driven solutions, ultimately fostering a more intelligent and responsive approach to greenhouse management.

2.1.1 Machine learning

Machine learning, a subset of artificial intelligence, involves training computer algorithms to learn from data without explicit programming. This capability enables computers to make predictions or take actions based on patterns in the data, with applications spanning various industries, including healthcare. In healthcare, machine learning algorithms are trained on extensive datasets comprising patient data, medical images, or electronic health records. These algorithms identify patterns and predict outcomes, thereby enhancing diagnosis accuracy, personalizing treatment plans, and forecasting patient outcomes. For instance, they've been instrumental in identifying patients at high risk for heart disease and predicting hospital readmissions [30].

The relationship between artificial intelligence (AI), machine learning (ML), and deep learning (DL) is illustrated in the figure below.

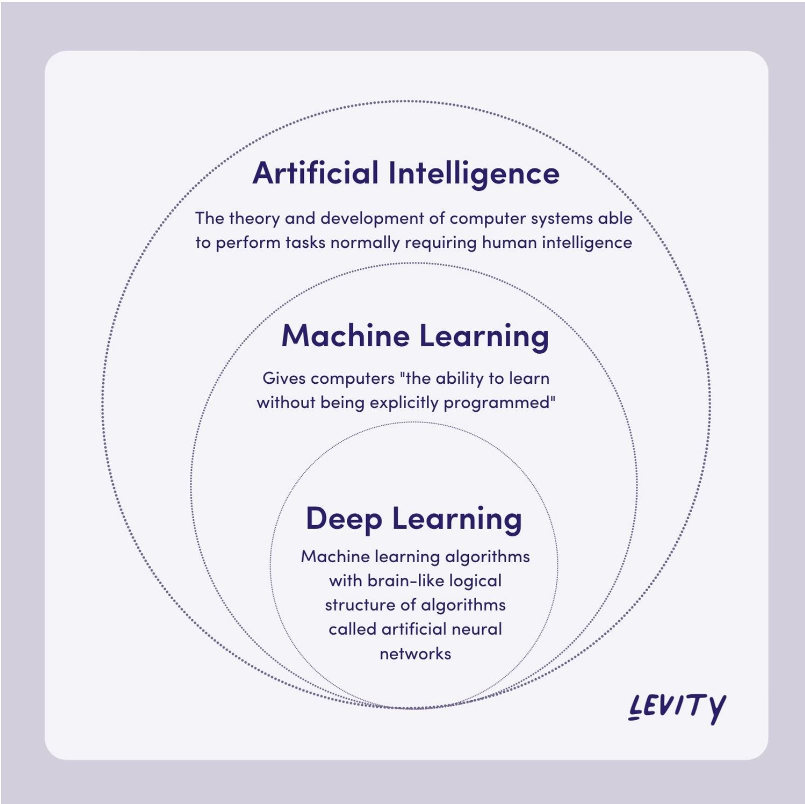


FIGURE 2.1: The relationship between(AI),(ML),and(DL)[1]

[1]Source: https://levity.ai/blog/difference-machine-learning-deep-learning

Types of Machine learning algorithms

There are three main types of machine learning:

- **Supervised Learning:** Supervised learning involves training a model on a "Labelled Dataset," which comprises both input and output parameters. In this type of learning, algorithms learn to establish correlations between inputs and corresponding outputs. Supervised learning encompasses both training and validation datasets, where each data point is labeled with the correct output [31].

  For example you're tasked with developing an image classifier to distinguish between elephants, cows, and camels. By feeding the algorithm labeled datasets containing images of these animals, the machine learns to differentiate between them. When presented with new images of elephants, cows, or camels, the trained algorithm applies its learned patterns to predict the correct classification. This exemplifies supervised learning, particularly in the context of image classification.

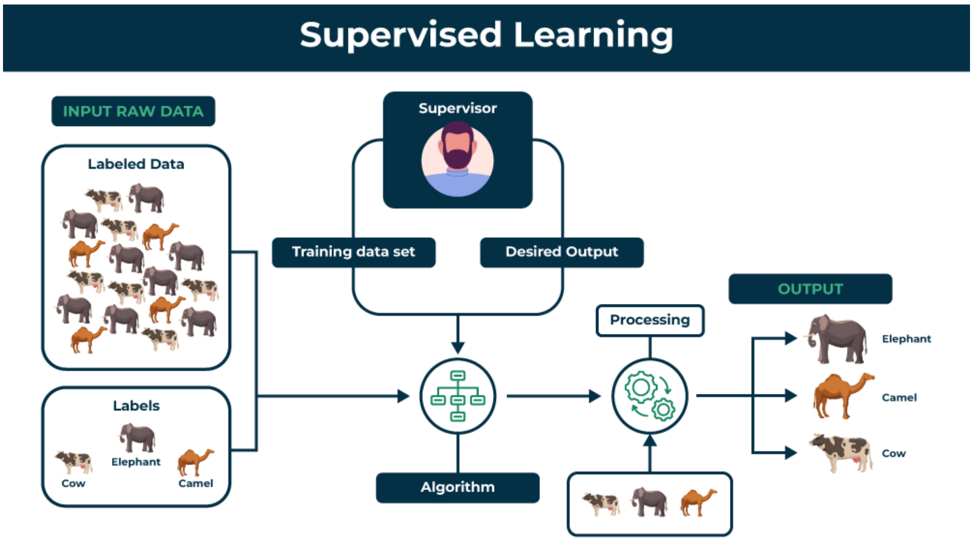


FIGURE 2.2: Supervised learning[2]

- **Unsupervised Learning:** Unsupervised learning is a machine learning approach where algorithms uncover patterns and relationships in unlabeled data. Unlike supervised learning, it doesn't rely on labeled target outputs. Instead, the primary objective of unsupervised learning is to unveil hidden patterns, similarities, or clusters within the data. These insights can be utilized for various purposes, including data exploration, visualization, dimensionality reduction, and more [31].

[2]Source: https://www.geeksforgeeks.org/types-of-machine-learning

For example Let's consider a scenario where a zookeeper collects data on the daily activities of various animals in their care, including elephants, cows, and camels. Using unsupervised learning techniques like clustering, the algorithm can identify patterns in the animals' behaviors without needing predefined labels. By clustering similar behavioral patterns, the zookeeper gains insights into the natural tendencies of each animal species without explicitly labeling the data. These insights can then inform decisions related to animal care, habitat design, and visitor experiences at the zoo .

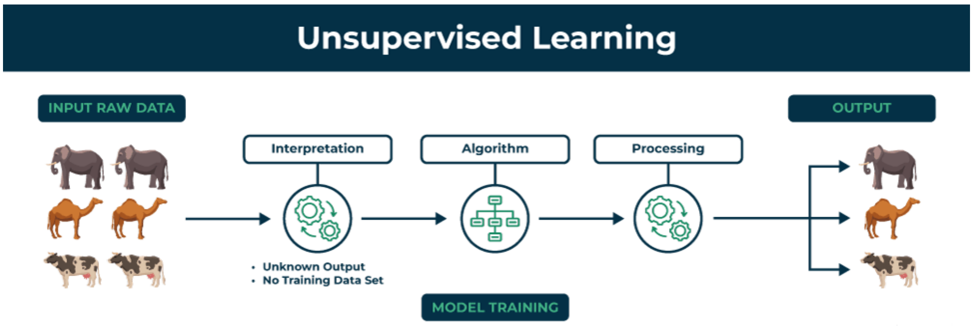


FIGURE 2.3: Unsupervised learning[3]

- **Semi-Supervised Learning:** Semi-supervised learning lies between supervised and unsupervised learning, utilizing both labeled and unlabeled data. It's particularly valuable when acquiring labeled data is costly or time-consuming. This approach is chosen when labeled data is scarce or requires specialized resources for training [32].

  It's used when dealing with partially labeled datasets, where a small portion is labeled and the majority is unlabeled. Unsupervised techniques can be employed to predict labels, which are then used to train supervised models. This technique finds significant application in scenarios like image datasets where not all images are labeled [32].

  For example we're developing a model to classify images of elephants, cows, and camels. Gathering labeled data for every image can be time-consuming and costly. Semi-supervised learning comes into play by utilizing both labeled and unlabeled images.

  For instance, the model can be trained on a mix of labeled images of elephants, cows, and camels, along with a larger set of unlabeled images. Using unsupervised techniques, the model can identify patterns in the unlabeled images and refine its understanding of distinguishing features for each animal category. This combined approach leads to improved classification accuracy, without the need for labeling every single image.

[3]Source: https://www.geeksforgeeks.org/types-of-machine-learning

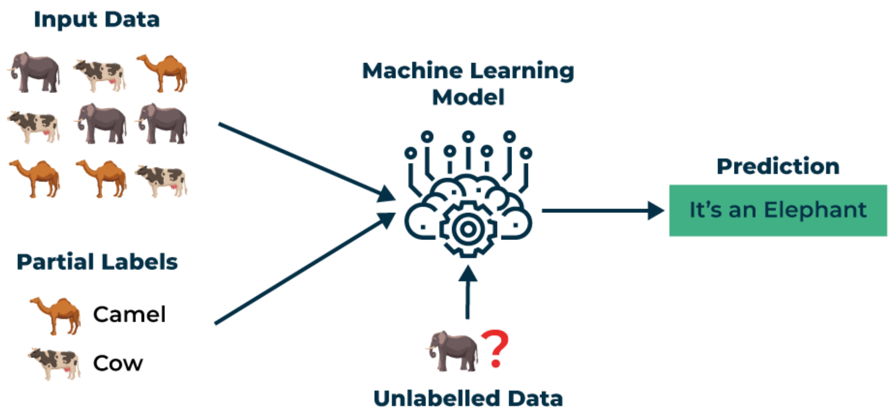


FIGURE 2.4: Semi supervised learning[4]

- **Reinforcement Machine Learning:** Reinforcement learning is a method where an algorithm interacts with its environment, taking actions and discovering errors to improve its performance. Trial, error, and delayed feedback are fundamental characteristics of this approach. The model learns by receiving reward feedback, adjusting its behavior or patterns accordingly [33].

  This technique is exemplified in applications like Google's Self-Driving car and AlphaGo, where the algorithm competes with humans or itself to enhance performance. With each interaction, the model learns and integrates new data into its knowledge base, thus improving its training and experience over time [33].

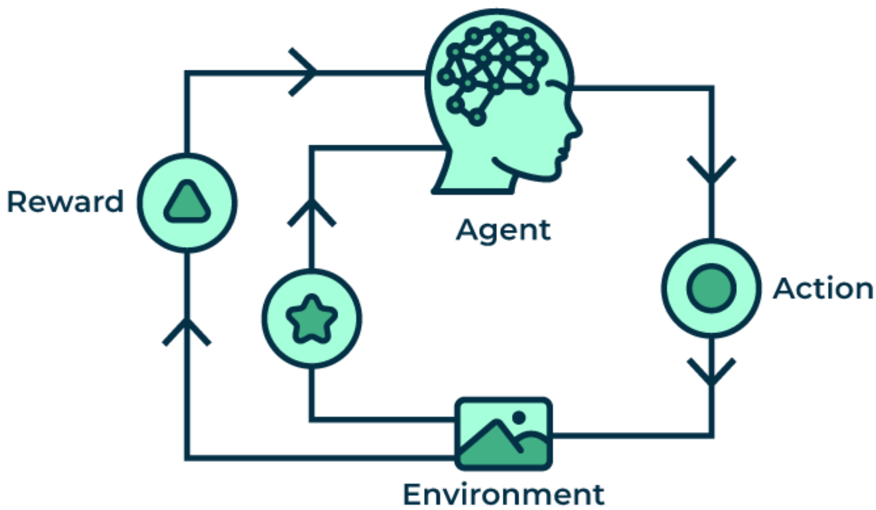


FIGURE 2.5: Reinforcement Machine Learning[5]

[4]Source: https://www.geeksforgeeks.org/types-of-machine-learning

2.1.2 Deep learning

Introduction

Deep learning, a subset of machine learning, is renowned for its ability to train neural networks. It stands out as a powerful technique in classification, finding applications across various disciplines utilizing machine learning.

In deep learning, neural network architectures comprise numerous hidden layers, hence the term "deep," indicating the depth of layers, typically exceeding two hidden layers. These networks are adept at processing and analyzing complex data types such as images, sounds, and text. The multiple layers within these neural networks enable them to progressively learn abstract features from the data, thereby enhancing their capability to make accurate predictions or classifications. [34]

Artificial Neural Networks

Artificial neural networks are inspired by the structure and function of biological neurons. They consist of interconnected nodes, or neurons, organized in layers. In a typical neural network, the input layer, the first layer, receives input from external sources and passes it to the hidden layer, the second layer. Each neuron in the hidden layer receives input from neurons in the previous layer, computes a weighted sum, and transfers it to neurons in the next layer.

These connections between neurons are weighted, meaning that the influence of inputs from the preceding layer is adjusted by assigning each input a unique weight. During the training process, these weights are iteratively adjusted to optimize the performance of the model. This allows the neural network to learn and adapt to the data it is trained on, ultimately improving its ability to make accurate predictions or classifications [35].

[5]Source: https://www.geeksforgeeks.org/types-of-machine-learning

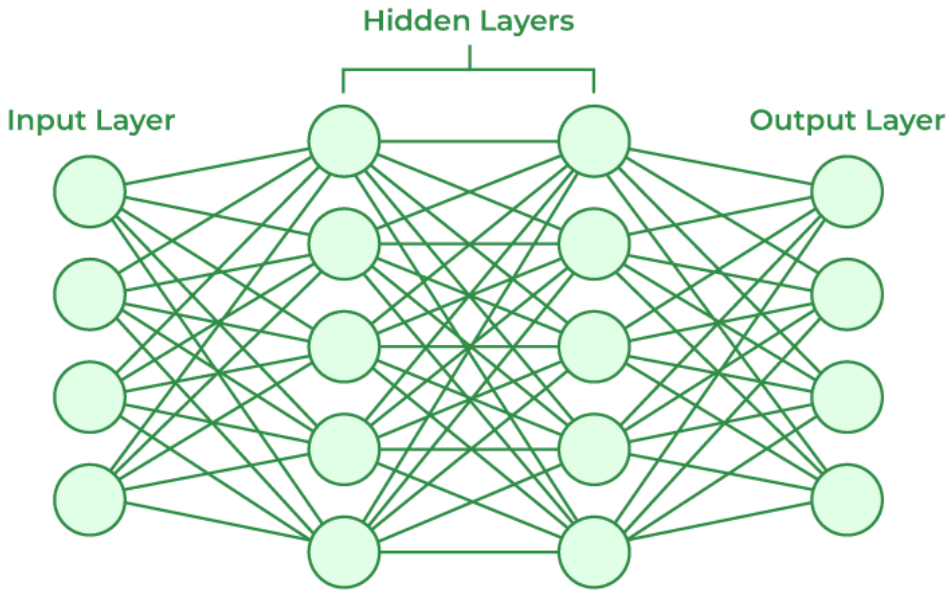


FIGURE 2.6: Fully Connected Artificial Neural Network [6]

Difference between Machine Learning (ML) and Deep Learning (DL)

- In machine learning, statistical algorithms are utilized to uncover hidden patterns and relationships within datasets, while deep learning employs artificial neural network architecture for the same purpose [35].
- Machine learning is effective with smaller datasets, whereas deep learning typically requires larger volumes of data for optimal performance.
- Machine learning is suited for low-label tasks, while deep learning excels in handling complex tasks like image processing and natural language processing.
- Training a machine learning model usually takes less time compared to deep learning, which often requires more time due to the complexity of the neural network architecture.
- In machine learning, models are created using manually extracted relevant features from data, whereas deep learning automatically extracts relevant features through an end-to-end learning process.

2.2 Introduction to YOLOv8

The YOLOv8 architecture operates by dividing the input image into a grid and simultaneously predicting bounding boxes and class probabilities

[6]Source: https://www.geeksforgeeks.org/neural-networks-using-the-r-nnet-package/

for objects within each grid cell. This grid-based approach enables YOLOv8 to detect multiple objects in a single pass through the network, making it exceptionally fast and efficient [4].

Key features of YOLOv8 include:

- Backbone Network: YOLOv8 typically employs a feature extraction backbone network, to extract hierarchical features from the input image [4].

- Detection Head: Following feature extraction, YOLOv8 utilizes a detection head comprising convolutional layers to predict bounding boxes and class probabilities for objects within each grid cell [4].

- Anchor Boxes: YOLOv8 utilizes anchor boxes, predefined bounding box shapes with varying aspect ratios, to facilitate accurate localization of objects of different sizes [4].

- Multi-Scale Prediction: YOLOv8 employs multi-scale prediction to detect objects at different resolutions within the input image, enhancing its ability to capture objects of varying sizes [4].

- Non-Maximum Suppression (NMS): is a pivotal technique utilized by YOLO to refine object detection predictions. After generating bounding box predictions, YOLO applies NMS to eliminate redundant detections and retain only the most confident predictions. This process involves identifying overlapping boxes, comparing confidence scores, and retaining the highest-scoring boxes while discarding lower-scoring ones [4].

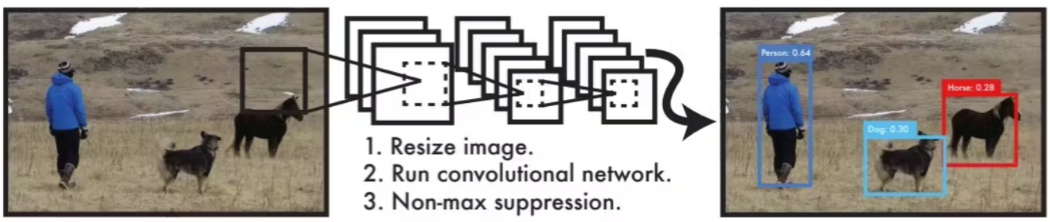


FIGURE 2.7: object Detection [4]

2.3 Object Detection Algorithm

Object detection is a critical task in computer vision that involves identifying and localizing objects within an image or video. Various algorithms have been developed to tackle this problem, each with its strengths and weaknesses. In this section, we will overview some of the most prominent object detection algorithms and discuss why we chose YOLOv8 (You Only Look Once version 8)

Below is a comparison of several key object detection algorithm

2.3.1 Faster R-CNN

Faster R-CNN (Region-Based Convolutional Neural Networks) integrates a Region Proposal Network (RPN) and Fast R-CNN into a single, unified model. The RPN generates region proposals, which are then refined by the Fast R-CNN. This architecture offers high accuracy, providing precise object detection by separating the proposal generation and classification stages. Additionally, it exhibits flexibility by being able to detect objects of varying sizes and aspect ratios due to the RPN. However, it is more computationally intensive and harder to implement.
Faster R-CNN is typically used in scenarios where detection accuracy is critical, such as detailed image analysis, security systems, and medical imaging [36].

2.3.2 SSD (Single Shot MultiBox Detector)

The architecture of SSD (Single Shot MultiBox Detector) eliminates the region proposal stage and directly predicts bounding boxes and class probabilities from feature maps at multiple scales. This approach offers several advantages: it is faster than R-CNN-based models due to its single-shot nature, and it is simpler to train and implement. However, the SSD has its disadvantages, as it is generally less accurate than Faster R-CNN, particularly for small objects.
SSD is suitable for real-time applications like surveillance and traffic monitoring [37].

2.3.3 YOLO

The architecture of YOLO frames object detection as a single regression problem, dividing the image into a grid and predicting bounding boxes and probabilities for each cell. This design has several advantages: it is extremely fast and capable of real-time detection, and its single-stage architecture makes it computationally efficient. However, early versions of YOLO had issues with small object detection and localization accuracy .but now With the newest version, YOLO has significantly improved its capabilities in small object detection and localization accuracy, addressing previous limitations. This enhancement further solidifies its position as a leading solution for real-time object detection tasks across various applications, including autonomous driving, video surveillance, and sports analytics [38].

2.3.4 RetinaNet

The architecture of RetinaNet introduces the Focal Loss to address class imbalance and uses a feature pyramid network (FPN) and anchors at multiple scales for object detection. This design has several advantages: it is effective for detecting small and infrequent objects due to the Focal Loss, and it strikes a good balance between speed and accuracy. However, RetinaNet is more computationally demanding compared to simpler models like SSD.
It is suitable for applications requiring high precision, such as retail analytics and medical imaging [39].

2.3.5 EfficientDet

The architecture of EfficientDet is built on the EfficientNet backbone and utilizes a weighted bi-directional feature pyramid network (BiFPN) for improved multiscale feature fusion. This model offers several advantages: it is optimized for both accuracy and resource usage, and it can be easily scaled for different accuracy and speed requirements. However, EfficientDet may not achieve the highest accuracy compared to more complex models.
It is particularly suitable for mobile and embedded applications due to its efficiency [40].

2.3.6 Comparison Table between Object Detection Algorithm

Algorithm	Architecture	Speed	Accuracy	Complexity	Best Use Cases
Faster R-CNN	Two-stage detector with RPN and Fast R-CNN	Moderate	Generally higher accuracy	High	Detailed image. analysis, security systems
SSD	Single-stage detector	High	Moderate	Low	Real-time surveillance, traffic monitoring
YOLO	Single-stage detector	Very High	Decent , especially for real-time applications	Moderate	Autonomous driving, video surveillance
RetinaNet	one-stage detectior	Moderate	High	High	Retail analytics, medical imaging
EfficientDet	Two-stage detector	High	Moderate to High	Moderate	Mobile applications, embedded systems

TABLE 2.1: Comparison Table between Object Detection Algorithm

2.3.7 Why YOLOv8 is Superior

While Fast RCNN is known for its superior accuracy in object detection tasks, YOLOv8 offers a significant advantage in terms of speed. This makes YOLOv8 more suitable for real-time detection applications, such as our pest management system, where quick and accurate identification of pests is essential to take immediate action. The balance between speed and accuracy provided by YOLOv8 ensures that we can maintain high detection performance while meeting the real-time operational requirements of our project.

YOLOv8,is from the latest version of YOLO, incorporates several advancements:

- **Speed:** Maintains the fast detection capabilities of previous YOLO versions, crucial for real-time applications.

- **Accuracy:** Enhanced architecture and training techniques improve accuracy, making it competitive with more complex models.

- **Efficiency:** Optimized for various hardware, from high-end servers to mobile devices.

- **Versatility:** Flexible and adaptable to different detection tasks, from large-scale environments to embedded systems.

- **Community Support:** A large, active community ensures continuous improvement and extensive support.

2.4 YOLOv8

YOLO, or You Only Look Once, has emerged as a leading solution for object detection tasks, celebrated for its remarkable speed and accuracy. Originally developed to detect and locate objects in images and videos, YOLO's primary focus lies in recognizing and categorizing objects based on their visual characteristics. This versatility extends beyond traditional applications, as YOLO has found utility in diverse domains.

In our project focusing on the detection of pest insects within greenhouse environments, we capitalize on YOLO's prowess in object detection to address the challenges of pest management. Similar to medical imaging, the identification and localization of pests in greenhouse imagery are pivotal for effective pest control strategies. By leveraging YOLO's ability to discern and classify objects in real-time, we aim to empower our system with the capability to swiftly identify and respond to pest infestations, safeguarding crop health and optimizing agricultural productivity.

here is deferent yolo version see Figure:

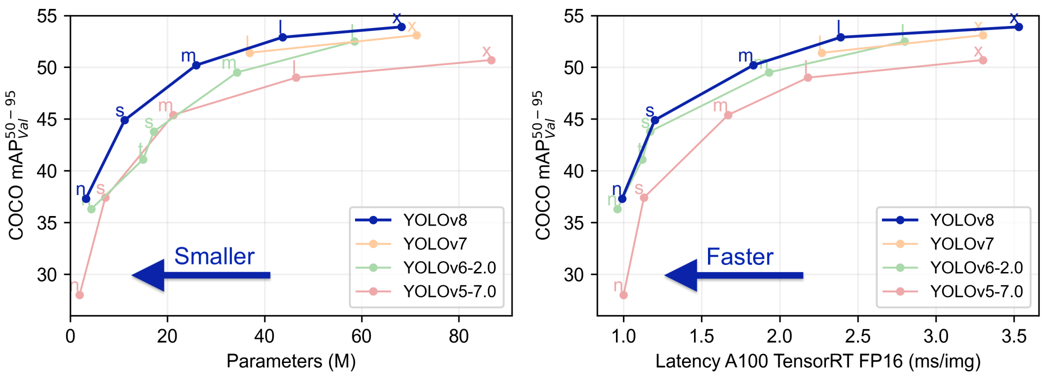


FIGURE 2.8: YOLO versions [4]

2.5 Conclusion

The integration of machine learning and deep learning, particularly through YOLOv8, demonstrates immense promise for the biological control of greenhouse pests and diseases. By automating and enhancing the accuracy of pest and disease detection, growers are empowered to make informed, timely decisions, leading to optimized crop protection strategies. The insights gained in this chapter lay the foundation for further research and development, encouraging the continued exploration of AI's potential to transform not only greenhouses but the broader agricultural landscape. Moving forward, collaboration between agricultural experts and AI specialists will be key to ensuring that these technologies are effectively harnessed to address the challenges of sustainable food production.

Chapter 3

System Design

Contents

3.1 Introduction

This chapter covers the overall design and implementation of our project, which focuses on using AI to detect pest insects in greenhouses. We start with the general design and diagrams to give a clear overview of the system. Next, we dive into the detailed design, explaining each component in depth. We then describe the dataset we used, how we prepared it, and the steps for data augmentation. Following this, we discuss how the dataset was split for training and testing. Finally, we explain how we utilized the YOLOv8 model for pest insect detection and the overall composition of the system.

3.2 General Design

The objective of our research endeavors to develop an AI-driven solution aimed at pest management within greenhouse environments using a robotic system equipped with a camera and spray mechanism. Our methodology entails several key steps, elucidated below:

- **Model Creation:** We begin by crafting by crafting a robust deep learning model tailored for pest detection within greenhouse settings. This involves selecting pertinent datasets containing images of various pest insects and undertaking preprocessing tasks to enhance data quality and relevance.
- **Training and Testing:** With the model architecture in place, we proceed to train it using the curated dataset, ensuring optimal performance in accurately identifying and classifying pest insects. Rigorous testing procedures are employed to validate the model's efficacy and robustness.
- **Deployment in Cloud:** Following successful model training and validation, we deploy our AI-powered solution in the cloud infrastructure. This facilitates seamless integration with our web application, enabling real-time interaction with users.
- **Web Application Development:** Concurrently, we develop a user-friendly web application designed to interface with our intelligent model. Through this application, users can see live streaming footage captured by the robotic system's camera for pest detection analysis. Additionally, we provide users with real-time notifications detailing the detected pest insects, along with relevant instructions on how to proceed, including recommendations for utilizing biological control products effectively. This comprehensive approach ensures that users have access to actionable insights, empowering them to make informed decisions in managing pest infestations within their greenhouse environments.
- **Real-time Detection and Response:** Upon receiving live streaming data from the robotic system, our deployed model swiftly analyzes the footage to detect the presence of pest insects. In the event of a positive detection, the system triggers an automated order to commence spraying with a designated biological control product.

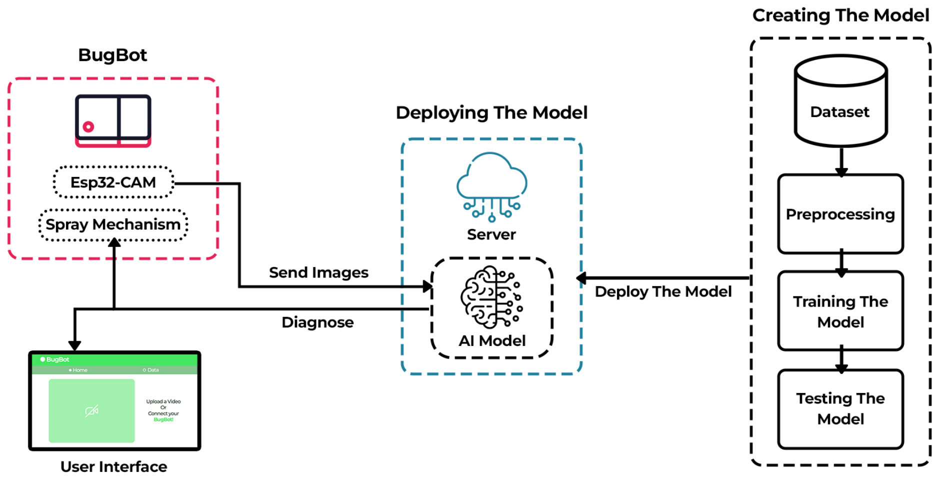


FIGURE 3.1: General design of our System

Illustrative Figure provides a visual depiction of our integrated framework, offering users a comprehensive understanding of our innovative approach to greenhouse pest management. Through the seamless integration of AI technology, robotic automation, and biological control strategies, we aim to revolutionize pest management practices, fostering sustainable agriculture and environmental stewardship.

3.2.1 Use Case Diagram

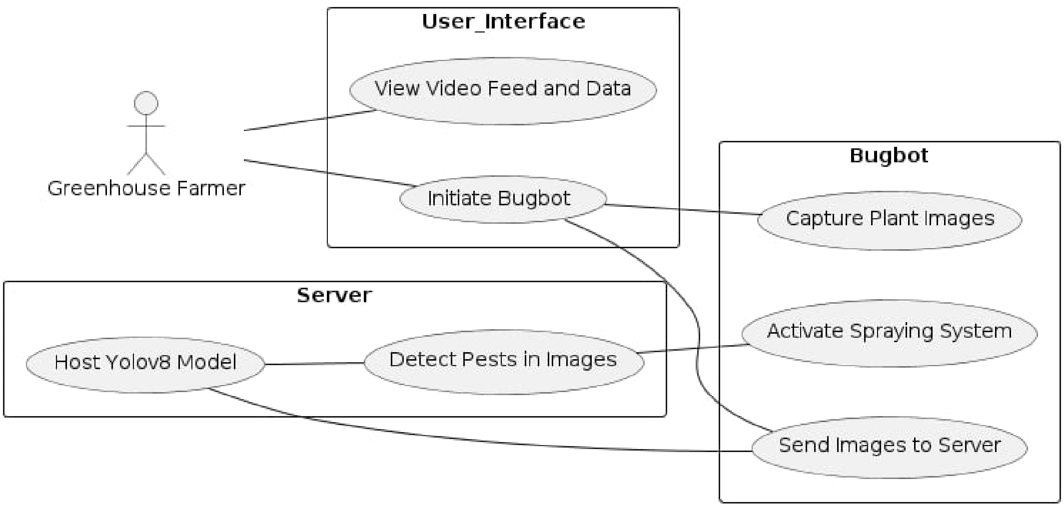


FIGURE 3.2: Use Case Diagram

This diagram is essential for understanding the system's scope and the roles of different actors, such as the user and the system itself. It depicts the main use cases, including initiating monitoring, capturing images, processing images, detecting pests, activating the sprayer, and viewing results. By illustrating these interactions, the use case diagram helps clarify the system's requirements and the user's interaction with the system.

3.2.2 Activity Diagram

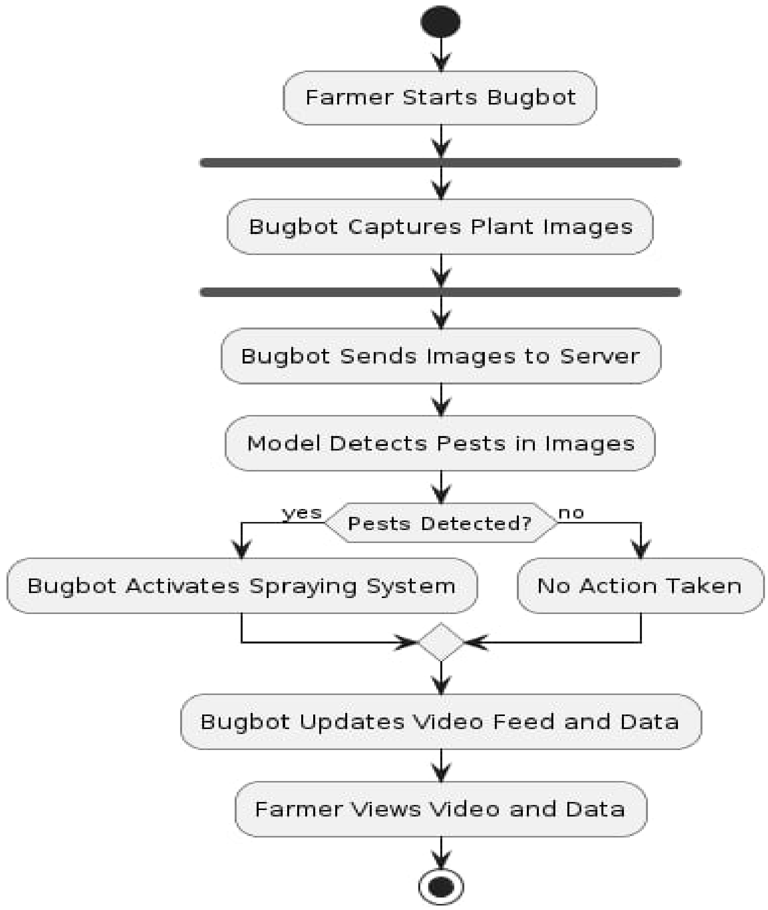


FIGURE 3.3: Activity Diagram

This diagram is crucial for understanding the workflow and the logical sequence of operations performed by the system. It showcases the steps involved in capturing images, processing those images, detecting pests, and activating the sprayer system. By illustrating these activities, the activity diagram helps clarify the process flow and decision points within the system, ensuring a comprehensive understanding of how the system operates.

3.2.3 Sequence Diagram

To further illustrate the interactions and workflows within our pest detection and management system, we present the sequence diagram. This diagram provides a detailed view of the communication between different components and the sequence of operations involved in detecting pests and triggering the appropriate responses.

The sequence diagram showcases the step-by-step process from capturing images through the ESP32 camera to processing these images, detecting pests using the YOLOv8 model, and ultimately activating the sprayer system to apply the biological control product. This visual representation helps to clarify the dynamic behavior of the system, ensuring a clear understanding of how each component collaborates to achieve the overall functionality.

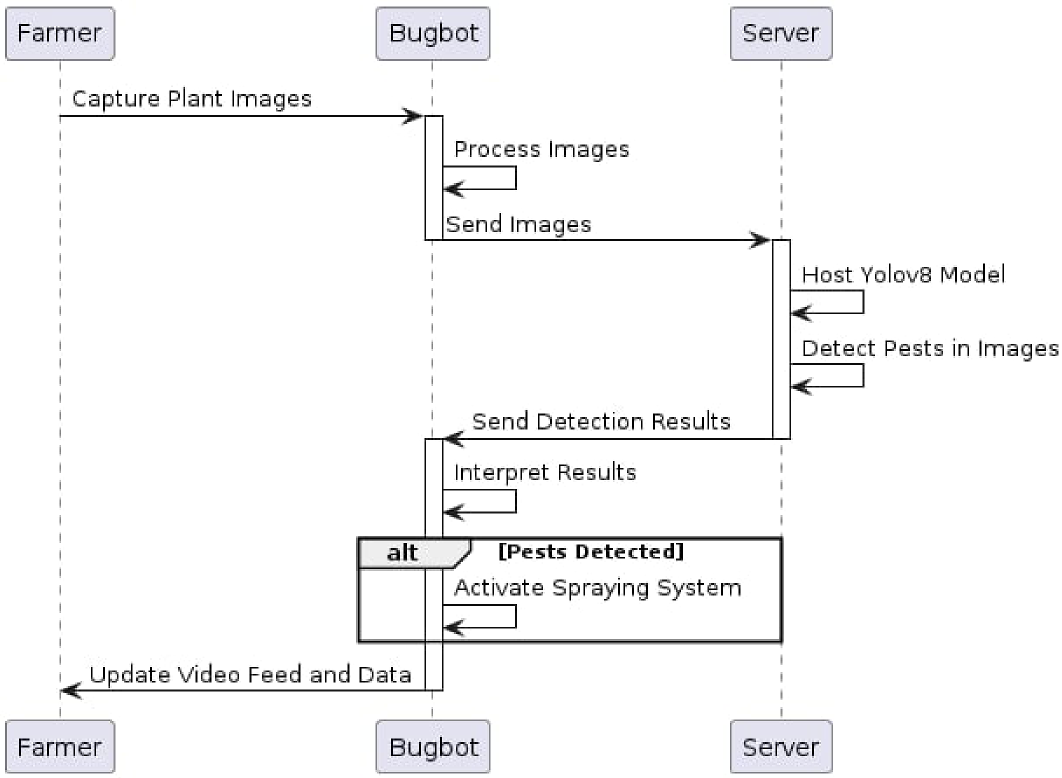


FIGURE 3.4: Sequence Diagram

3.3 Detailed Design

To construct our deep learning model for detecting and classifying pest insects, our system will adhere to a structured approach as outlined in the accompanying figure.

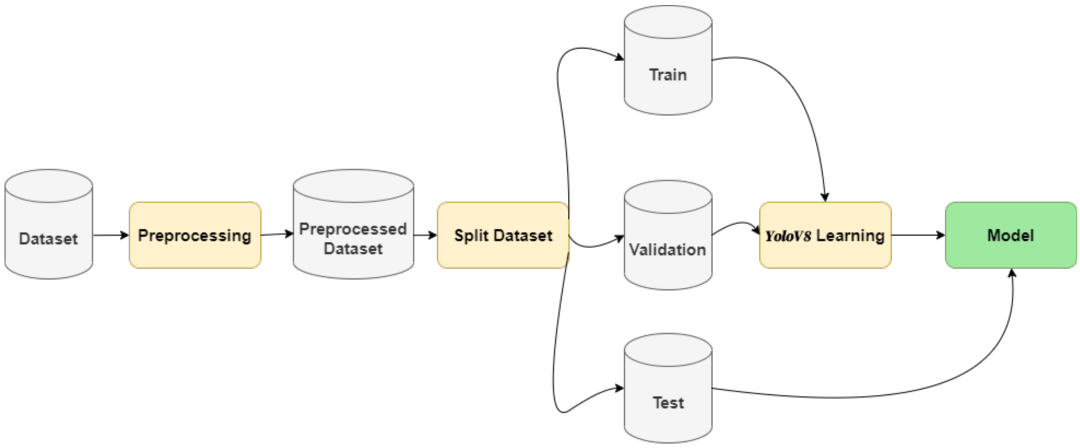


FIGURE 3.5: Detailed System Design

Initially, we commence by acquiring a comprehensive dataset comprising images of various pest insects. Subsequently, the dataset undergoes preprocessing to enhance its quality and suitability for model training. This includes tasks such as normalization, augmentation, and noise reduction to optimize the dataset for effective learning.

Following preprocessing, the dataset is partitioned into distinct subsets for training, validation, and testing purposes. This ensures robust model performance evaluation and validation against unseen data.

The core of our system lies in the utilization of the YOLOv8 architecture, a powerful deep learning model specifically designed for object detection tasks. The YOLOv8 model is fed with the partitioned dataset, undergoing iterative training epochs to learn and extract meaningful features from the input images.

Through this iterative training process, the YOLOv8 model gradually improves its ability to discern and classify different types of pest insects with high accuracy and reliability.

Upon completion of training, we obtain a fully trained YOLOv8 model capable of classifying new pest insect images with remarkable precision. This model serves as a potent tool for pest detection and classification, empowering users with actionable insights for effective pest management within greenhouse environments.

3.4 Dataset

To compile our dataset for pest insect detection and classification, we employed a meticulous process encompassing data collection, labeling, augmentation, and train/test split. Here's an overview of our methodology:

Data Collection: We meticulously curated our dataset by gathering a diverse array of images showcasing pest insects commonly found in greenhouse environments. These images were sourced from various platforms including **Kaggle**, **Roboflow**, and online repositories. Through this process, we compiled a comprehensive dataset that includes a wide range of pest insects along with detailed annotations

Labeling with Roboflow: To annotate our dataset with bounding boxes delineating the regions of interest (pest insects), we utilized Roboflow, an efficient labeling tool. Through Roboflow's intuitive interface, each image was meticulously annotated by marking the bounding box around the detected pest insect(s). This process ensured accurate labeling, essential for model training.

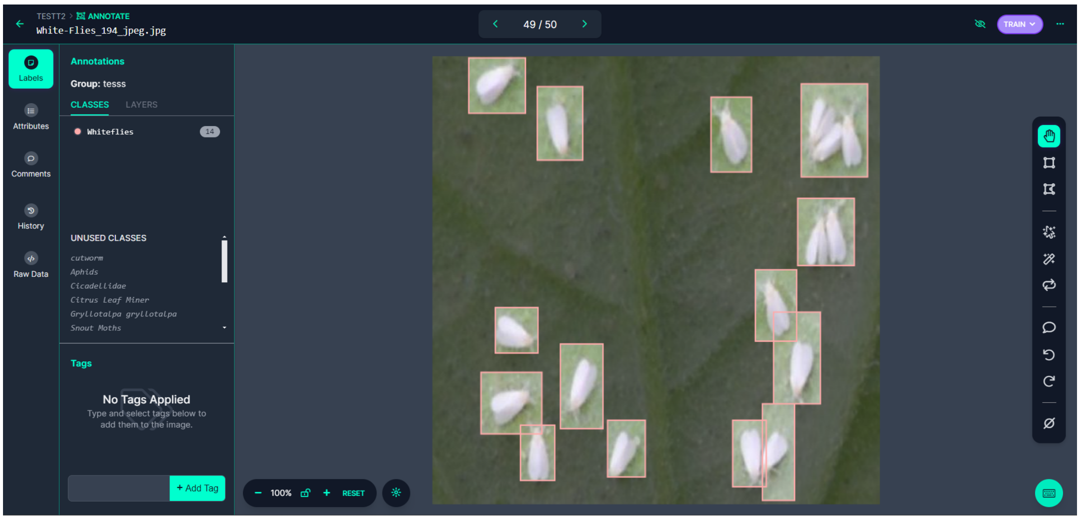


FIGURE 3.6: labeling images with roboflow

Augmentation: To increase the diversity and variability of the dataset, data augmentation techniques were applied. These techniques included random rotations, flips (both vertical and horizontal), and adjustments to brightness. Augmentation helps in preventing overfitting and improves the model's ability to generalize to unseen data.

Train/Test Split: The dataset was divided into three subsets: 80% for training, 10% for validation, and 10% for testing.

Data Conversion: The annotations generated during labeling were converted into YOLOv8 format, a popular framework for object detection tasks. Each image's bounding box annotations were encoded into corresponding .txt files, with each file containing the coordinates and class labels of the detected pest insects. This standardized format facilitates seamless integration with the model training pipeline.

By meticulously curating our dataset, annotating with precision, augmenting for diversity, and adhering to a structured train/test split, we ensured the robustness and effectiveness of our deep learning model for pest insect detection and classification within greenhouse environments.

3.5 Preprocessing

Before providing the data to the CNN model for training, we need to pass through the preprocessing phase. This is a crucial step where we make modifications to the data, such as resizing, applying filters, and removing noise, to ensure it is in the optimal format for training the model. In our project, we specifically used techniques like auto-adjust contrast, static crop, and resizing images to 640x640 pixels.

Preprocessing involves several essential steps to ensure that the data is in the optimal format for training the YOLOv8 model. This section will detail these steps, including data cleaning, augmentation, normalization, and formatting.

3.5.1 Adjust Contrast

Auto-adjust contrast is used to enhance the visibility of features in the images by adjusting the contrast levels. This helps in highlighting the details that are crucial for accurate pest detection.

3.5.2 Static Crop

Static cropping is used to focus on the central part of the image, which is often the region of interest. This reduces the amount of irrelevant background and ensures the model focuses on the relevant areas.

3.5.3 Resizing

The original dataset images were (1024,1024,3) We resize images to (640x640x3). The first number indicate the width of image and the second indicate Hight and the last number indicate image channels RGB (Red, Green, Blue)

Why resizing images ?
Resizing the images to a uniform size ensures consistency in the input data, which is crucial for efficient training and inference. The size 640x640 pixels was chosen to balance detail preservation and computational efficiency.

3.6 Data Augmentation

Image augmentations are modifications applied to images to increase the number of images by applying different transformations for example rotating, scaling, flipping, brightness, cropping or adding some noise, to create new samples that are similar to the original images. Data augmentation also helpful in improves model generalization, reduces overfitting, addresses class imbalance, and helps in scenarios with limited data availability.

In this work we increase the number of images by different transformations like rotation, scaling, flipping, shear, brightness adjustment, and color jittering, introduce variations to the images while preserving their semantic content.

- Rotation: Images were rotated between -15° and +15°.
- Shear: Shear transformations included ±14° horizontally and ±9° vertically.
- Brightness: The brightness of images was adjusted between -16% and +16%.
- Flip: Images were flipped both horizontally and vertically.
- Hue: Adjustments were made between -15° and +15°.
- Blur: Blurring was applied with a maximum radius of 2.1 pixels.
- Noise: Up to 1.21% of pixels were altered to add noise.

Initially, we had about 7000 images. After applying data augmentation, the dataset expanded to over 20000 images.

These augmentations introduce significant variations to the dataset while maintaining the integrity of the semantic content of the images, thus enabling the model to generalize better to unseen data.

3.7 Splitting Dataset

Following annotation and augmentation, the dataset was divided into three subsets: 80% for training, 10% for validation, and 10% for testing. This stratified split ensures that each subset contains a representative distribution of images across different pest insect classes, thus facilitating comprehensive model training and evaluation.

3.8 Utilization of YOLOv8 in Pest Insect Detection

In our pest insect detection system, we leverage the robust capabilities of YOLOv8 to detect and classify pest insects within greenhouse environments. The YOLOv8 architecture, with its real-time processing capabilities and high accuracy, is ideally suited for our application, where timely and accurate detection of pest insects is paramount.

To incorporate YOLOv8 into our system, we first preprocessed our annotated dataset to ensure compatibility with the YOLOv8 framework. This involved converting annotations into the YOLOv8 format, where each image is accompanied by a corresponding .txt file containing bounding box coordinates and class labels for detected pest insects.

Subsequently, we fine-tuned the pre-trained YOLOv8 model on our annotated dataset using transfer learning techniques. Transfer learning allows us to leverage the knowledge gained from training on a large dataset and adapt it to our specific task of pest insect detection. This process significantly accelerates model convergence and improves detection performance.

Once trained, our YOLOv8-based pest insect detection model is capable of processing live streaming footage from the robotic system's camera in real-time, accurately detecting and classifying pest insects within greenhouse environments.

By harnessing the power of YOLOv8, we equip our system with state-of-the-art object detection capabilities, enabling efficient and precise pest management within greenhouse settings.

3.9 BugBot Composition

This section outlines the implementation of the pest detection bot's hardware. The following materials are required:

Components Description

1. **ESP32-CAM Module:** A microcontroller board with integrated camera functionality is ideal for capturing images for pest detection. It includes Wi-Fi capabilities for data transmission and onboard flash memory. This module serves as the primary sensory and communication hub of the autonomous pest detection robot, capturing images of the greenhouse environment. The captured images are transmitted via Wi-Fi to a remote server for analysis, and the module receives instructions from the server based on the analysis results, controlling the relay and activating the water pump when necessary.

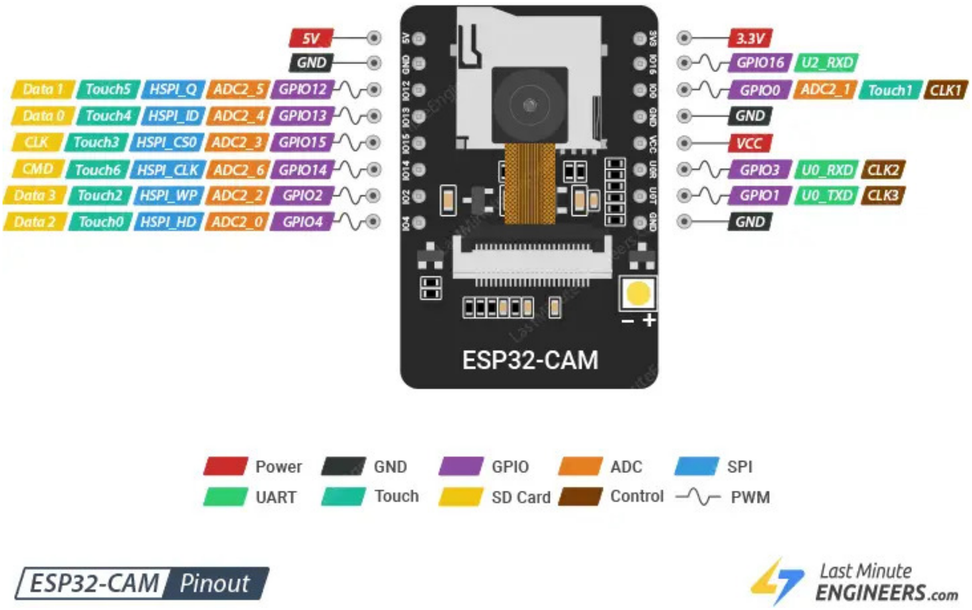


FIGURE 3.7: ESP32-CAM[1]

2. **Breadboard:** A solderless prototyping board, the breadboard facilitates the initial stages of circuit design and experimentation. Its matrix of interconnected sockets allows for easy component insertion and removal, promoting rapid prototyping and testing without the need for soldering.

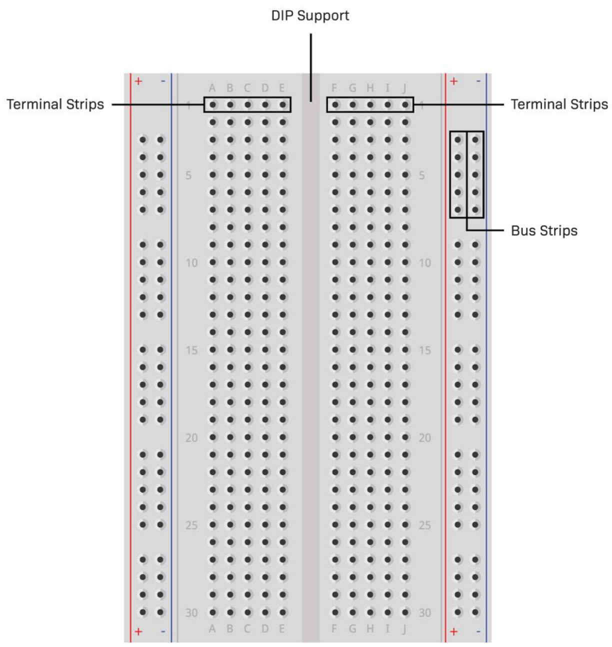


FIGURE 3.8: Breadboard[2]

[1]https://lastminuteengineers.com/getting-started-with-esp32-cam/
[2]https://diyodemag.com/education/fundamentals_breadboarding_basics

3. **Breadboard Power Supply Module (MB-102):**
During the development and testing phases, this module provides a versatile power source for the breadboard setup. It offers multiple output voltages and can be powered via USB, wall adapter, or battery, enhancing the flexibility of the prototyping environment.

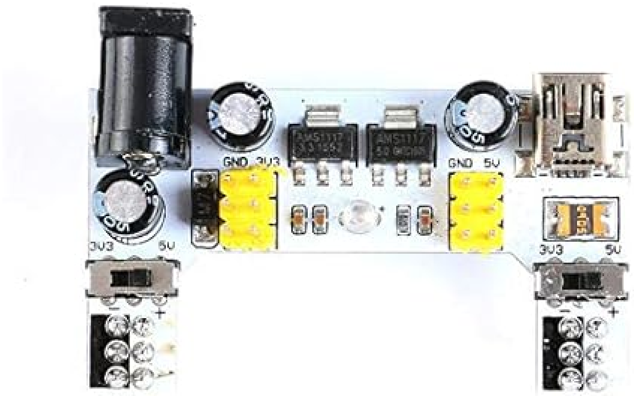


FIGURE 3.9: MB102 Power Supply[3]

4. **Jumper Wires:** These insulated wires, typically short in length and terminated with pin connectors, are instrumental in establishing electrical connections between various components. Their flexibility enables the creation of both temporary and permanent circuits, facilitating the development and assembly process.

[3]https://www.amazon.com/Interface-Breadboard-Supply-Regulator-Channel/dp/B07T8PL41N

FIGURE 3.10: Jumper Wires[4]

5. **Relay:** An electromagnetic switch, the relay acts as an intermediary between the low-voltage signals from the ESP32-CAM module and the higher power requirements of the water pump. Upon receiving a control signal from the ESP32-CAM (indicating the presence of pests), the relay closes its internal circuit, enabling the activation of the water pump.

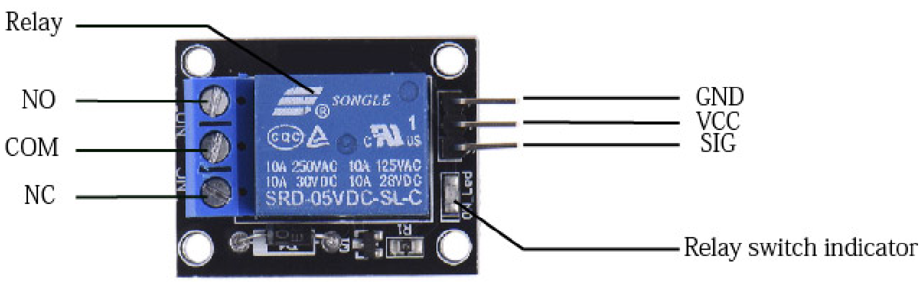


FIGURE 3.11: 5V Relay[5]

[4]https://www.amazon.com/dp/B0BDFML3XD?th=1
[5]https://www.aranacorp.com/en/using-a-relay-module-with-arduino/

6. **Water Pump:** This pump functions as the robot's effector mechanism. Upon receiving an activation signal from the relay, the pump initiates the dispensation of pesticides, as determined by the pest control strategy. The pump's specifications, such as flow rate and pressure, are selected based on the greenhouse dimensions and the specific pest mitigation requirements.

FIGURE 3.12: Water Pump[6]

3.10 Conclusion

In summary, this chapter provided a comprehensive look at the design and implementation of our AI-based pest insect detection system. We covered everything from the initial design and dataset preparation to the use of data augmentation and the YOLOv8 model. Each step was crucial in building an effective and accurate detection system. This detailed explanation sets the foundation for understanding how our approach works and its potential impact on improving greenhouse pest management.

[6]https://www.amazon.de/-/en/HALJIA-Horizontal-Submersible-Brushless-1-2-1-6L/dp/B07TTXTSL8

Chapter 4

Realization

Contents

4.1 Introduction

In this chapter, we delve into the realization of our innovative pest detection and management system tailored for greenhouse environments. Building upon the foundation laid in previous chapters, we transition from conceptualization to implementation, detailing the practical aspects of our solution's development and deployment.

4.2 Frameworks , Tools and Libraries

Firstly, the project is developed using a system powered by an AMD Ryzen 5 4600H CPU running at 3.00GHz, equipped with 16 GB of memory, and featuring an NVIDIA GeForce GTX 1650Ti with 4GB of dedicated GPU memory. The development environment is based on Windows 11/10.

The primary tools and programming languages utilized for the implementation of our system are the following:

- **Python:** Guido van Rossum introduced the high-level programming language Python in 1991 Python now is used extensively in various domains like web development, data analysis, machine learning, automation and scripting [1].

FIGURE 4.1: Python logo [1]

- **JavaScript:** JavaScript is a versatile programming language commonly used for web development. Initially designed to add interactivity to web pages, it has evolved into a powerful language for building dynamic, interactive web applications and server-side development through frameworks like Node.js [2].

FIGURE 4.2: JavaScript logo[2]

- **C++:** C++ serves as a versatile programming language utilized within the Arduino IDE, offering a powerful platform for developing software

[1]https://www.python.org/doc/essays/blurb/
[2]https://www.geeksforgeeks.org/introduction-to-javascript/

for Arduino microcontrollers. While retaining its object-oriented nature, C++ in the Arduino environment emphasizes creating and manipulating objects that represent various hardware components and functions. For instance, a user may define objects such as sensors, actuators, or communication modules, each possessing distinct attributes and behaviors. This object-oriented approach facilitates the creation of complex projects by encapsulating functionality within modular, reusable components [3].

FIGURE 4.3: C++ logo[3]

- **Google Colab:** Colab, or Google Colaboratory, is an online platform for coding in Python without the need for local installation. It provides a cloud-based environment with features like collaborative editing and access to powerful hardware resources [4].

FIGURE 4.4: Google Colab logo[4]

TABLE 4.1: Google Colab resources.

	GPU	**Runtime**	**RAM**	**Disk Capacity**
Google Colab	A100	4h	64GB	80GB

.

- **Kaggle:** Kaggle is a platform for data science competitions and collaboration where users can access and use real world datasets [5].

[3] https://www.arduino.cc/reference/en/
[4] https://colab.research.google.com/notebooks/intro.ipynb./

FIGURE 4.5: Kaggle logo[5]

- **OpenCv:** OpenCV or Open Source Computer Vision Library is a widely used open-source library for computer vision and image processing tasks. It offers a range of functions and algorithms for tasks like image manipulation, object detection and video analysis [6].

FIGURE 4.6: OpenCv logo[6]

- **VS Code:** Visual Studio Code combines the simplicity of a source code editor with powerful developer tooling, like IntelliSense code completion and debugging.

  First and foremost, it is an editor that gets out of your way. The delightfully frictionless edit-build-debug cycle means less time fiddling with your environment, and more time executing on your ideas . [7].

FIGURE 4.7: Visual Studio Code logo[7]

.

[5]https://www.kaggle.com/docs/notebooks/
[6]https://docs.opencv.org/4.x/d1/dfb/intro.html
[7]https://code.visualstudio.com/docs/editor/whyvscode

- **Arduino IDE:** The Arduino Integrated Development Environment (IDE) is a user-friendly software application that simplifies the process of writing, compiling, and uploading code to Arduino boards and Micro-controllers.

  The IDE provides a streamlined interface with a text editor for code, a message area for feedback, a console for serial communication, and a toolbar with essential functions. It seamlessly connects to hardware, enabling you to program and interact with your projects. [8].

FIGURE 4.8: Arduino IDE logo[8]

.

- **Flask:** Flask is a web framework, it's a Python module that lets you develop web applications easily. It's has a small and easy-to-extend core: it's a microframework that doesn't include an ORM (Object Relational Manager) or such features [9].

FIGURE 4.9: Flask logo[9]

.

[8] https://docs.arduino.cc/learn/starting-guide/the-arduino-software-ide/
[9] https://flask.palletsprojects.com/

- **Roboflow:** Roboflow is the universal conversion tool for computer vision datasets. We import any annotation format and export to any other, meaning you can spend more time experimenting and less time wrestling with one-off conversion scripts for your object detection datasets [10].

FIGURE 4.10: Roboflow logo[10]

.

4.3 Dataset Preparation and Preprocessing

4.3.1 Install YOLOv8

YOLOv8 can be installed in two ways from the source and via pip. This is because it is the first iteration of YOLO to have an official package.

```

!pip install ultralytics
```

LISTING 4.1: download the datasets

4.3.2 Download Dataset to Google Colab

First, we used the dataset API to download the Roboflow datasets to our Google Colab

```
!pip install roboflow

from roboflow import Roboflow
rf = Roboflow(api_key="**********************")
project = rf.workspace("agriculter").project("agriculer")
version = project.version(3)
dataset = version.download("yolov8")
```

LISTING 4.2: download the datasets

[10] https://roboflow.com/formats

4.3.3 Custom Training

After pasting the dataset download snippet into your YOLOv8 Colab notebook, we are ready to begin the training process.

```

!yolo task=detect mode=train model=yolov8s.pt data=/content/
   datasets/agriculter/data.yaml epochs=100 imgsz=640 plots=
   True
```

LISTING 4.3: Custom Training our model

4.4 Results

Here are the results of training a player detection model with YOLOv8:

4.4.1 Model Accuracy Measured on Validation Set

The accuracy of the model on the validation set indicates how well the model generalizes to unseen data. It is calculated as the ratio of correctly predicted instances to the total instances in the validation set

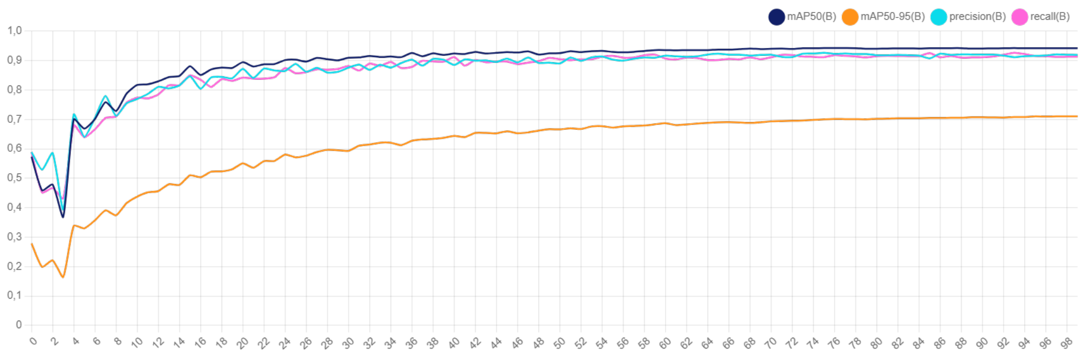


FIGURE 4.11: The Metrics

4.4.2 Confusion Matrix

The confusion matrix provides a summary of the prediction results, indicating true positives, false positives, true negatives, and false negatives for each class of pest insects.

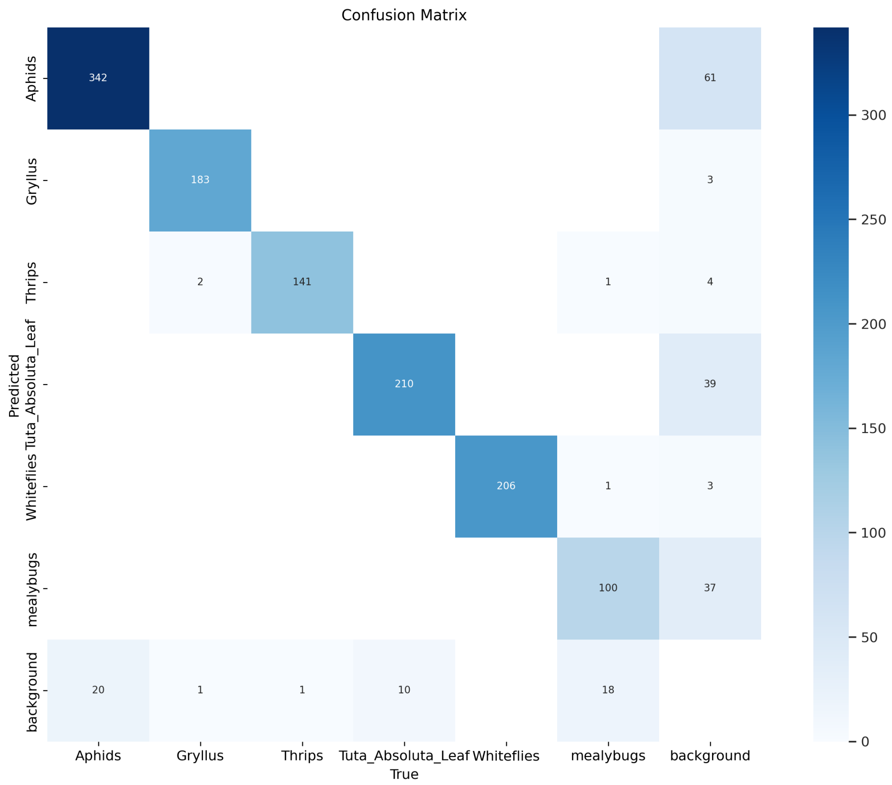


FIGURE 4.12: The confusion matrix returned after training

4.4.3 Class loss

Class loss measures the model's performance in classifying each instance into the correct category. Lower class loss values indicate better performance.

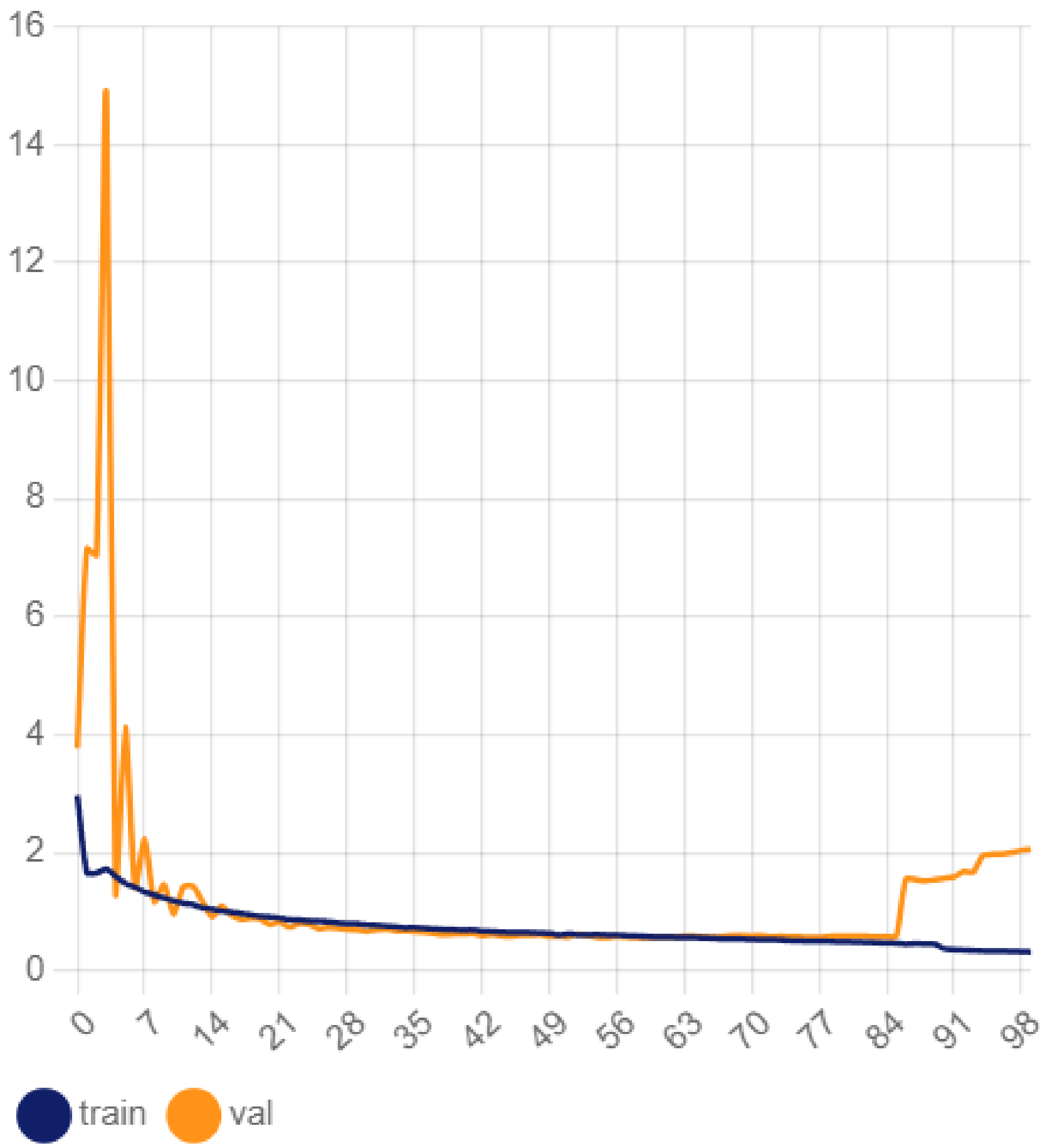


FIGURE 4.13: Class loss

4.5 Realization

4.5.1 Flask Web Application Framework

In our project, Flask serves as the backbone of our web application, facilitating communication between the ESP32-CAM detection sensor and other components of the system. It handles incoming image data from the sensor, processes it, and forwards it to the deep learning model for pest detection. Additionally, Flask provides a user-friendly interface for real-time monitoring and analysis of pest detection results.

By leveraging Flask's simplicity and flexibility, we streamline the development and deployment of our pest detection system, enabling efficient communication and interaction between its various components.

4.5.2 Detection Sensor (ESP32-CAM)

The ESP32-CAM serves as the detection sensor in our system, tasked with capturing live streaming footage and detecting pest insects within greenhouse environments. Equipped with a camera module and Wi-Fi connectivity, the ESP32-CAM captures real-time images, which are then processed for pest detection.

The ESP32-CAM communicates with a Flask server, a lightweight web application framework written in Python. The Flask server facilitates seamless integration between the ESP32-CAM and our deep learning model for pest insect detection. It serves as the intermediary between the sensor and the detection model, handling incoming image data from the ESP32-CAM and forwarding it to the model for analysis.

Upon receiving live streaming footage from the ESP32-CAM, the Flask server processes the images and feeds them into the trained YOLOv8 model for pest detection. The detection results are then relayed back to the server, which can be accessed by users through a web interface for real-time monitoring and analysis.

4.5.3 Reaction Mechanism

In response to the detection of pest insects, our system triggers a reaction mechanism involving a spray system to mitigate the infestation. The spray system is equipped with a biological control product designed to target and eliminate the detected pest insects while minimizing environmental impact.

Upon receiving confirmation of pest detection from the YOLOv8 model via the Flask server, the reaction mechanism is activated. The system sends a signal to the spray system, initiating the release of the biological control product within the greenhouse.

The spray system is strategically positioned to deliver targeted application of the biological control product, ensuring efficient pest management while minimizing collateral damage to non-target organisms.

By integrating detection sensor technology with a responsive reaction mechanism, our system enables proactive and precise pest management within greenhouse environments. This holistic approach facilitates timely detection and intervention, safeguarding crop health and optimizing agricultural productivity.

4.5.4 BugBot Prototype

This section details the assembly and initial configuration of the BugBot prototype, as outlined in Chapter 3/Sec. 3.9. The necessary electronic components were procured and interconnected on a breadboard as per the provided schematic. The MB-102 power supply module was utilized to power the breadboard and, consequently, all connected components. Jumper wires were employed to establish electrical connections between the various components on the breadboard. Notably, the ESP32-CAM module was connected to the breadboard both for power and to transmit signals to the relay's control pin. Similarly, the water pump was connected to the breadboard for power and to the relay's output, enabling activation upon receiving a signal from the ESP32-CAM module.

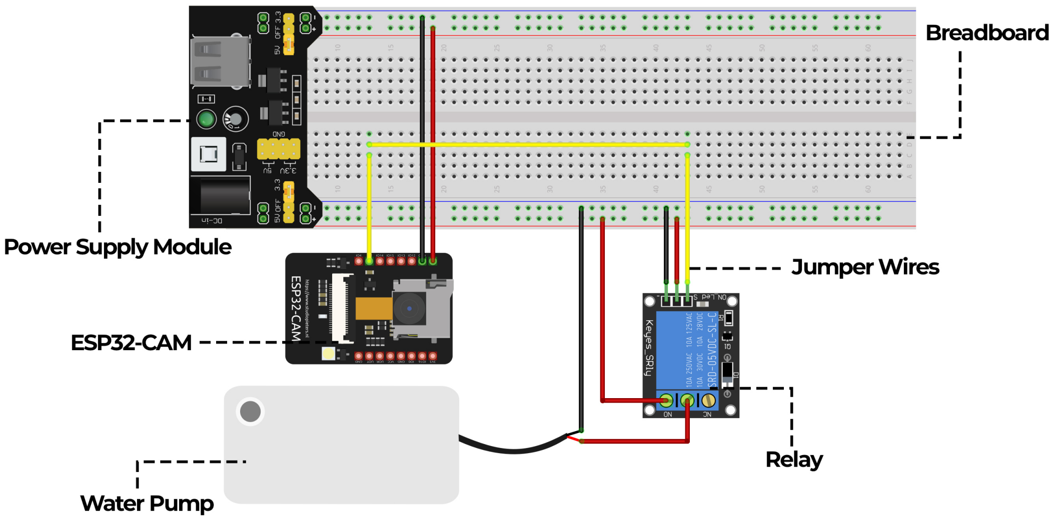


FIGURE 4.14: Interconnecting Prototype Components

Upon completion of the hardware assembly, in accordance with the provided instructions, the subsequent step involved programming and configuring the BugBot's operational environment. The Arduino IDE was employed as the development platform, and the programming process, which encompassed setting up the Wi-Fi connection, camera initialization, and communication with the remote server, is elaborated upon in the following section.

```
#include "WifiCam.hpp"
#include <WiFi.h>
#include <WebServer.h>
#include <esp32cam.h>

const char* WIFI_SSID = "Wifi_Name";
const char* WIFI_PASS = "Wifi_Password";

WebServer server(80);

```

```
static auto loRes = esp32cam::Resolution::find(320, 240);
static auto midRes = esp32cam::Resolution::find(480, 320);
static auto hiRes = esp32cam::Resolution::find(800, 600);

const int relayPin = 2;
```

LISTING 4.4: Part 1: Declarations and Variables

Part 1. This section includes the necessary header files for Wi-Fi connectivity, web server functionality, camera operation, and custom configurations. It also declares variables to store Wi-Fi credentials, camera resolutions, and the pin for controlling the relay.

```
void serveJpg() {
    // ... (code for capturing and serving JPEG images)
}

void handleJpgLo() {
    // ... (code for handling low-resolution image requests)
}

void handleJpgMid() {
    // ... (code for handling mid-resolution image requests)
}

void handleJpgHi() {
    // ... (code for handling high-resolution image requests)
}
```

LISTING 4.5: Part 2: Image Capture and Serving Functions

Part 2. These functions facilitate image capture from the ESP32-CAM module and their transmission as JPEG files via the web server. The functions handleJpgLo(), handleJpgMid(), and handleJpgHi() allow for capturing images at different resolutions to optimize bandwidth usage based on the requirements.

```
void controlRelay(bool on) {
    digitalWrite(relayPin, on ? LOW : HIGH);
}

void handlePump() {
    controlRelay(true);
    delay(5000);
    controlRelay(false);
    server.send(200, "text/plain", "Pump activated for 5 seconds
   ");
}
```

LISTING 4.6: Part 3: Relay and Pump Activation Function

Part 3. controlRelay(bool on) controls the state of the relay connected to the water pump, effectively turning it on or off. handlePump() is called when the BugBot receives a command from the server to activate the pump. It turns the pump on for a specified duration (5 seconds in this case) and then turns it off.

```
void setup() {
    // (Code for serial communication, camera configuration, Wi-Fi setup, and web server setup)
}

void loop() {
    server.handleClient();
}
```

LISTING 4.7: Part 4: Setup and Main Loop

Part 4. The setup() function initializes serial communication, configures the camera settings, connects to the specified Wi-Fi network, and sets up the web server routes to handle incoming requests. The loop() function continuously runs, listening for and responding to client requests. These requests can trigger image capture, resolution changes, or pump activation.

4.6 Application Interface

The application interface for our pest detection and management system is designed to provide users with a seamless and intuitive experience for monitoring and managing pest infestations within greenhouse environments. Leveraging modern web technologies, the interface offers real-time access to pest detection results, enabling users to make informed decisions and take timely action.

Key Features:

Live Streaming Feed: The interface displays a live streaming feed from the ESP32-CAM detection sensor, allowing users to observe real-time imagery of the greenhouse environment. This feed provides continuous monitoring and enables users to visually inspect for signs of pest activity.

Pest Detection Results: Detected pest insects are highlighted and labeled within the live streaming feed, providing users with immediate visibility into pest infestations. Each detected pest insect is annotated with relevant information.

Alert Notifications: In the event of significant pest detections or critical alerts, the interface generates real-time notifications to alert users. These notifications may include actionable insights or recommendations for pest management strategies.

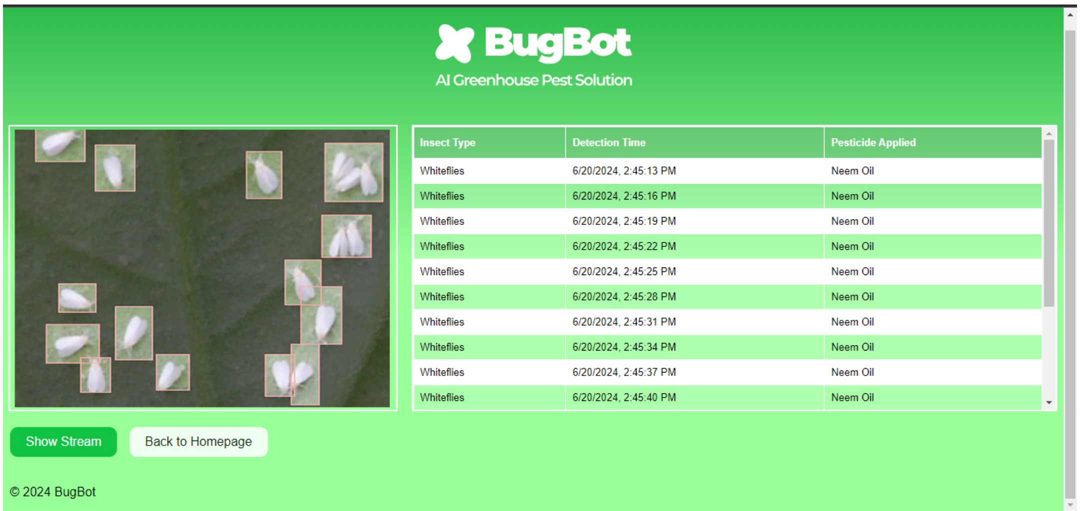


FIGURE 4.15: Application Interface

4.7 Comparison with Related Work Results

In this section, we compare our work with the related studies mentioned earlier. By examining the detection models, deployment environments, targeted pests, and achieved accuracies, we aim to highlight the strengths and unique contributions of our approach. Our system, which uses the YOLOv8 model for greenhouse pest detection, is evaluated against other innovative projects such as AIoT-based systems, dual-drone precision spraying, and improved YOLOv5 models. This comparative analysis underscores the effectiveness of our system and identifies areas for further improvement, contributing to the advancement of smart pest management solutions in agriculture.

Project	Model Used	Environment	Target Pests	Accuracy
Our Project	YOLOv8	Greenhouses	Aphids Gryllus Thrips Tuta_Absoluta Whiteflies mealybug	99% (Gryllus, Thrips,Whitefly,) 91% (aphids) 97% (Tuta_absuluta) 78%(Mealybugs)
RobHortic	PLS-DA	Carrot fields	'Candidatus Liberibacter solanacearum'	66.4% (lab), 59.8% (field)
Drone Based Pest Detection and Precision Pesticide Spraying in Longan Orchards	YOLOv3	Orchards	Tessaratoma papillosa	>95%
Improved YOLOv5 Model for Pest Detection in Greenhouses	YOLOv5	Greenhouses	Leaf miners, fruit flies, aphids, houseflies, tobacco whiteflies	99% (leaf miners) 98% (aphids, houseflies) 97% (whiteflies)

TABLE 4.2: Comparison Table with Related Work Results

4.7.1 Discussion

Now that we have compared our work with the related studies mentioned earlier, it is evident that our project offers several advantages over these existing solutions.

Strengths of Our Project

Our project demonstrates clear superiority in several key areas. While some existing algorithms rely on trapping insects and then using image detection to achieve high accuracy, our model excels at directly detecting pests on plants in real-time. By leveraging the YOLOv8 model, we achieve higher detection accuracy and efficiency in identifying pest insects compared to other models like YOLOv3 and YOLOv5. The integration of a real-time monitoring system with automated pest control measures further sets our project apart, providing a comprehensive and immediate response solution for greenhouse pest management. Additionally, our user-friendly web interface enhances accessibility and usability for farmers, allowing them to monitor and control the system effortlessly.

4.8 Conclusion

In this chapter, we have thoroughly examined the design, implementation, and effectiveness of our pest detection and management system for greenhouses. By leveraging the YOLOv8 model and integrating advanced technologies such as IoT devices and real-time monitoring, our project demonstrates superior performance in identifying and managing pest infestations compared to related works. The comprehensive design and seamless integration of various components ensure robust, accurate, and efficient pest control, offering significant advantages over existing solutions. Our project's innovative approach and potential for future enhancements highlight its contribution to advancing greenhouse pest management practices.

Conclusion and Perspectives

The agricultural industry has witnessed remarkable advancements in recent years, especially in the realm of pest management within controlled environments like greenhouses. Artificial intelligence (AI) and robotics have emerged as a promising solution for the early detection and precise elimination of pests, minimizing crop damage and reducing the need for broad-spectrum pesticides. This thesis explored the development and implementation of an AI model and robotic platform to address these challenges.

In traditional greenhouse settings, pest control often relies on manual scouting and scheduled pesticide applications, which can be labor-intensive, time-consuming, and environmentally impactful. This approach may result in delayed detection and insufficient targeting of pests, leading to unnecessary crop losses and increased chemical use. Therefore, a more proactive, automated, and precise pest management system is essential for sustainable and efficient greenhouse operations.

The proposed solution involves the deployment of an AI-powered robot equipped with a high-resolution camera and a targeted spraying mechanism. The AI model, trained on extensive datasets of pest images, accurately identifies and classifies various pest species within the greenhouse environment. Upon detection, the robot applies a precise dose of pesticide to the affected area, minimizing collateral damage to beneficial insects and the environment.

By leveraging AI and robotics, this system enables continuous monitoring, early pest detection, and targeted interventions, significantly improving pest control efficacy while reducing pesticide usage. The real-time data collected by the robot can also provide valuable insights into pest population dynamics, enabling proactive management strategies and further optimizing greenhouse operations.

In future work, we aim to enhance the capabilities of this robotic system significantly. We plan to implement autonomous movement, allowing the robot to navigate the greenhouse environment without human intervention. This will improve the system's responsiveness and efficiency in detecting and addressing pest outbreaks. Furthermore, we will focus on refining the spraying mechanism to ensure even more precise and targeted pesticide application, minimizing wastage and environmental impact. To enhance the accuracy and robustness of the AI model, we will expand the training dataset by incorporating images and data from various sources, capturing a wider range of pest species and environmental conditions. By continuously improving the hardware and software components of this system, we aim to revolutionize pest management in greenhouses and contribute to a more sustainable and productive agricultural industry.

BugBot Economical Part

الجمهورية الجزائرية الديمقراطية الشعبية
وزارة التعليم العالي والبحث العلمي
جامعة محمد خيضر -بسكرة

مشروع لنيل شهادة مؤسسة ناشئة في إطار القرار الوزاري 1275

Biological Control of Greenhouses Using AI Techniques

المكافحة البيولوجية للبيوت البلاستيكية باستخدام تقنيات الذكاء الاصطناعي

Company Name (الاسم التجاري) : BugBot - Entreprise

Company Logo (العلامة التجارية) :

BugBot

AI Greenhouse Pest Solution

1- فريق الإشراف :

الاستاذة المشرفة : بن عيسى يسرى
التخصص : اعلام الي

2- فريق العمل :

الطالب : جودي اشرف عبد المجيد
التخصص : ذكاء اصطناعي

الطالب : حروز منذر
التخصص : صورة و حياة اصطناعية


تقديم المشروع
الجوانب الإبتكارية
التحليل الإستراتيجي للسوق
خطة الإنتاج والتنظيم
الخطة المالية
النموذج الاولي التجريبي
فهرس المحتويات

مشروع ابتكار روبوت ذكي للكشف عن الآفات في البيوت الزراعية المحمية

المحور الأول: تقديم المشروع

1. الحل المقترح:

يهدف هذا المشروع إلى تطوير روبوت ذكي يعمل بالذكاء الاصطناعي، مصمم خصيصًا للكشف المبكر والدقيق عن الآفات الزراعية في البيوت المحمية. يسهم هذا الروبوت في تعزيز الإنتاجية الزراعية وضمان جودة المحاصيل وحماية البيئة من خلال تقليل الاعتماد على المبيدات الكيميائية.

2. القيم المقترحة:

- **الابتكار:** يقدم هذا المشروع حلاً مبتكرًا لمشكلة الآفات الزراعية، يجمع بين التكنولوجيا المتقدمة والذكاء الاصطناعي لتعزيز كفاءة الزراعة.
- **الدقة والكفاءة:** يعتمد الروبوت على تقنيات الذكاء الاصطناعي المتطورة لتحليل البيانات المرئية والبيئية بدقة عالية، مما يضمن الكشف المبكر والدقيق عن الآفات.
- **الاستدامة:** يساهم الروبوت في تحقيق الزراعة المستدامة من خلال تقليل استخدام المبيدات الكيميائية الضارة بالبيئة والصحة العامة.
- **سهولة الاستخدام:** يتميز الروبوت بتصميمه السهل الاستخدام وواجهة المستخدم البديهية، مما يجعله متاحًا للمزارعين من مختلف المستويات التقنية.
- **التكلفة الفعالة:** يوفر الروبوت حلاً فعالًا من حيث التكلفة لمكافحة الآفات، حيث يقلل من الخسائر الناجمة عن تلف المحاصيل و يوفر في تكاليف المبيدات.
- **الشراكة والتعاون:** نسعى إلى بناء شراكات مع المزارعين والخبراء الزراعيين لتطوير حلول تلبي احتياجاتهم وتسهم في تحقيق أهدافهم الزراعية.

3. فريق العمل:

- الطالب جودي اشرف عبدالمجيد ، تخصص الذكاء الاصطناعي.
- الطالب حروز منذر ، تخصص صورة وحياة اصطناعية.
- الأستاذة بن عيسى يسرى.

4. أهداف المشروع:

- **تطوير روبوت ذكي:** تصميم وتطوير روبوت متكامل قادر على التنقل داخل البيوت المحمية وجمع البيانات المرئية والبيئية اللازمة لتحليل الآفات.
- **تطبيق تقنيات الذكاء الاصطناعي:** استخدام خوارزميات التعلم الآلي والتعلم العميق لتمكين الروبوت من تحليل البيانات المرئية وتحديد أنواع الآفات بدقة عالية.
- **الكشف المبكر عن الآفات:** تمكين الروبوت من اكتشاف الإصابات المبكرة بالآفات قبل أن تنتشر وتتسبب في أضرار جسيمة للمحاصيل.
- **تقديم توصيات ذكية:** تزويد المزارعين بتوصيات مبنية على البيانات والتحليلات التي يقوم بها الروبوت، لمساعدتهم في اتخاذ قرارات فعالة لمكافحة الآفات.

- **تحسين الإنتاجية والجودة:** المساهمة في زيادة الإنتاجية الزراعية وتحسين جودة المحاصيل من خلال مكافحة الآفات بشكل فعال ومستدام.
- **تعزيز الاستدامة البيئية:** تقليل الاعتماد على المبيدات الكيميائية الضارة وتعزيز الممارسات الزراعية الصديقة للبيئة.

5. الفوائد الإضافية:

- **توفير بيانات مفصلة:** يوفر الروبوت للمزارعين بيانات مفصلة عن أنواع الآفات ومواقعها وشدتها، مما يساعدهم في اتخاذ قرارات مستنيرة بشأن إدارة الآفات.
- **توفير الوقت والجهد:** يقلل الروبوت من الجهد والوقت اللازمين لمراقبة الآفات يدويًا، مما يتيح للمزارعين التركيز على جوانب أخرى من إدارة مزارعهم.
- **تعزيز القدرة التنافسية:** يساعد الروبوت المزارعين على تحسين جودة منتجاتهم وتقليل تكاليف الإنتاج، مما يعزز قدرتهم التنافسية في السوق.

6. جدول زمني لتحقيق المشروع :

	الشهر	1	2	3	4	5	6	7	8
1	الدراسات الأولية		X						
2	تطوير قاعدة بيانات			X	X				
3	تطوير نموذج ذكاء الاصطناعي				X	X			
4	طلب التجهيزات و المعدات				X				
5	تركيب نموذج أولي					X	X		
6	برمجة واجهة تفاعلية						X		
7	تطوير نموذج أولي						X	X	
8	بداية إنتاج أول منتج								X

المحور الثاني: الجوانب الابتكارية

طبيعة الابتكارات:

يمتاز مشروعنا بالعديد من الابتكارات التي تتجلى في:

1- الابتكار التكنولوجي:

- **الذكاء الاصطناعي:** يعتمد الروبوت على تقنيات الذكاء الاصطناعي المتقدمة، مثل التعلم الآلي والتعلم العميق، لتحليل البيانات المرئية والبيئية بدقة عالية وتحديد أنواع الآفات بدقة غير مسبوقة.
- **الرؤية الحاسوبية:** يستخدم الروبوت كاميرات عالية الدقة وأجهزة استشعار متطورة لجمع البيانات المرئية وتحليلها، مما يمكنه من تحديد الآفات بدقة حتى في مراحل الإصابة المبكرة.

2- الابتكار في التصميم:

- **التصميم المدمج:** يتميز الروبوت بتصميمه المدمج وخفيف الوزن، مما يسهل عملية تنقله داخل البيوت المحمية دون الإضرار بالمحاصيل.
- **سهولة الاستخدام:** تم تصميم واجهة المستخدم الخاصة بالروبوت لتكون سهلة الاستخدام وبديهية، مما يجعله متاحًا للمزارعين من مختلف المستويات التقنية.

3- الابتكار في الخدمات:

- **تقديم توصيات ذكية:** يقدم الروبوت للمزارعين توصيات مبنية على البيانات والتحليلات التي يقوم بها، لمساعدتهم في اتخاذ قرارات فعالة لمكافحة الآفات.
- **التحديث المستمر:** يتم تحديث نظام الروبوت باستمرار بأحدث البيانات والمعلومات حول الآفات، مما يضمن كفاءته ودقته في الكشف عن الآفات الجديدة والمتطورة.

مجالات الابتكارات:

بالإضافة إلى الابتكارات المذكورة، يمكن للمشروع أن يستفيد من الابتكار في المجالات التالية:

1. **تحليل البيانات الضخمة:** يمكن استخدام تقنيات تحليل البيانات الضخمة لفهم الأنماط والعلاقات بين الآفات والظروف البيئية، مما يساهم في تطوير استراتيجيات مكافحة أكثر فعالية.
2. **إنترنت الأشياء (IoT):** يمكن دمج الروبوت مع شبكة من أجهزة الاستشعار والأجهزة المتصلة بالإنترنت لجمع بيانات بيئية إضافية وتحسين دقة التنبؤ بظهور الآفات.
3. **التعاون مع المزارعين:** يمكن إشراك المزارعين في عملية تطوير الروبوت وتقديم ملاحظاتهم واقتراحاتهم لتحسين أدائه وتلبية احتياجاتهم بشكل أفضل.

المحور الثالث: التحليل الإستراتيجي للسوق

القطاع السوقي:

يندرج مشروعنا ضمن قطاع التكنولوجيا الزراعية، وتحديدًا في مجال الزراعة الذكية (Smart Agriculture). يستهدف هذا القطاع المزارعين وأصحاب البيوت المحمية الذين يسعون إلى تحسين إنتاجيتهم وجودة محاصيلهم من خلال تبني التقنيات الحديثة.

الجمهور المستهدف:

- **المزارعون التقليديون:** الذين يبحثون عن حلول مبتكرة لمواجهة تحديات الآفات الزراعية وتحسين كفاءة عملياتهم الزراعية.
- **أصحاب المزارع العضوية:** الذين يفضلون استخدام أساليب مكافحة الآفات الصديقة للبيئة والحد من استخدام المبيدات الكيميائية.
- **شركات إنتاج المحاصيل الزراعية:** التي تسعى إلى زيادة إنتاجيتها وجودة محاصيلها وتقليل الخسائر الناجمة عن الآفات.
- **المهندسون الزراعيون والمستشارون:** الذين يمكنهم الاستفادة من البيانات والتحليلات التي يوفرها الروبوت لتقديم توصيات دقيقة للمزارعين.
- **الباحثون في مجال الزراعة:** الذين يمكنهم استخدام الروبوت كأداة بحثية لدراسة سلوك الآفات وتطوير استراتيجيات مكافحة جديدة.

مبررات اختيار السوق المستهدف:

- **الطلب المتزايد على الزراعة الذكية:** يشهد قطاع الزراعة الذكية نموًا متسارعًا، حيث يتزايد اعتماد المزارعين على التقنيات الحديثة لتحسين عملياتهم الزراعية وزيادة إنتاجيتهم.
- **التحديات المتزايدة في مكافحة الآفات:** تواجه الزراعة تحديات متزايدة بسبب تغير المناخ وانتشار الآفات الجديدة، مما يجعل من الضروري تطوير حلول مبتكرة لمكافحة الآفات بشكل فعال ومستدام.
- **الفجوة في السوق:** لا يزال هناك نقص في الحلول التكنولوجية المتكاملة للكشف عن الآفات في البيوت المحمية، مما يمثل فرصة كبيرة لمشروعنا لتلبية هذه الحاجة الملحة.
- **الوعي المتزايد بأهمية الاستدامة:** يتزايد وعي المزارعين بأهمية تبني ممارسات زراعية مستدامة وصديقة للبيئة، مما يجعل الروبوت الذي يقلل من استخدام المبيدات الكيميائية خيارًا جذابًا.
- **الدعم الحكومي للابتكار الزراعي:** تقدم العديد من الحكومات دعمًا ماليًا وتشجيعًا للابتكار في القطاع الزراعي، مما يوفر بيئة مواتية لنجاح مشروعنا.

أولا- تحليل المتغيرات الكلية(PESTEL):

شرح العوامل	العوامل	
• **الاستقرار السياسي:** يعتبر الاستقرار السياسي في الجزائر عاملًا إيجابيًا للمشروع، حيث يوفر بيئة آمنة ومستقرة للاستثمار والنمو. • **الدعم الحكومي للزراعة:** تقدم الحكومة الجزائرية دعمًا للقطاع الزراعي، بما في ذلك تقديم إعانات مالية وتسهيلات ائتمانية، مما يمكن أن يساهم في زيادة الطلب على الروبوت. • **اللوائح والقوانين:** قد تؤثر اللوائح والقوانين المتعلقة باستخدام التكنولوجيا في الزراعة على عمليات المشروع، مثل لوائح السلامة والأمان وحماية البيانات.	العوامل السياسية Political Factors	P
• **نمو القطاع الزراعي:** يشهد القطاع الزراعي في الجزائر نموًا مطردًا، مما يوفر فرصًا كبيرة للمشروع. • **الدخل المتاح للمزارعين:** يعتبر الدخل المتاح للمزارعين عاملًا مهمًا في تحديد قدرتهم على شراء الروبوت. • **التضخم وأسعار الفائدة:** يمكن أن يؤثر التضخم وارتفاع أسعار الفائدة على تكلفة المشروع و قدرة المزارعين على الحصول على التمويل.	العوامل الإقتصادية Economic Factors	E
• **الوعي بأهمية الزراعة المستدامة:** يتزايد الوعي بأهمية الزراعة المستدامة في الجزائر، مما يجعل الروبوت الذي يقلل من استخدام المبيدات الكيميائية خيارًا جذابًا للمزارعين. • **التحول نحو التكنولوجيا:** يزداد اعتماد المزارعين على التكنولوجيا في الجزائر، مما يوفر بيئة مواتية لتبني الروبوت. • **التغيرات الديموغرافية:** يمكن أن تؤثر التغيرات الديموغرافية، مثل زيادة عدد السكان، على الطلب على المنتجات الزراعية وبالتالي على الطلب على الروبوت.	العوامل الإجتماعية Social Factors	S
• **التطور السريع للتكنولوجيا:** يشهد مجال الذكاء الاصطناعي والروبوتات تطورًا سريعًا، مما يتطلب من الشركة مواكبة أحدث	العوامل التكنولوجية Technological Factors	T

		التطورات لتظل قادرة على المنافسة. • **تبني التكنولوجيا في الزراعة:** يزداد تبني التكنولوجيا في القطاع الزراعي في الجزائر، مما يوفر فرصًا كبيرة للمشروع. • **البنية التحتية التكنولوجية:** تعتبر جودة البنية التحتية التكنولوجية في الجزائر، مثل شبكات الإنترنت والاتصالات، عاملًا مهمًا في نجاح المشروع.
E	**العوامل البيئية** **Environmental Factors**	• **التغير المناخي:** يمكن أن يؤثر التغير المناخي على القطاع الزراعي في الجزائر، مما يزيد من أهمية الروبوت في الكشف عن الآفات والتكيف مع الظروف المناخية المتغيرة. • **الاستدامة البيئية:** يعتبر الروبوت الذي يقلل من استخدام المبيدات الكيميائية حلاً صديقًا للبيئة، مما يساهم في تعزيز الاستدامة البيئية في القطاع الزراعي. • **اللوائح البيئية:** قد تؤثر اللوائح البيئية المتعلقة باستخدام التكنولوجيا في الزراعة على عمليات المشروع.
L	**العوامل القانونية** **Legal Factors**	• **قوانين الملكية الفكرية:** يجب على الشركة حماية حقوق الملكية الفكرية المتعلقة بتقنيات الروبوت. • **قوانين حماية المستهلك:** يجب على الشركة الالتزام بقوانين حماية المستهلك فيما يتعلق بتسويق وبيع الروبوت. • **قوانين العمل:** يجب على الشركة الالتزام بقوانين العمل المتعلقة بتوظيف العمال وتوفير بيئة عمل آمنة وصحية.

ثانيا - تحليل القوى التنافسية (PORTER):

خطر دخول منافسين جدد	**متوسط إلى مرتفع:** يتطلب تطوير روبوتات الكشف عن الآفات استثمارات كبيرة في البحث والتطوير، وتقنيات متقدمة مثل الذكاء الاصطناعي و الرؤية الحاسوبية، مما يجعل حاجز الدخول مرتفعًا نسبيًا. ومع ذلك، قد يكون هناك شركات ناشئة أو شركات قائمة تسعى لدخول هذا السوق الواعد، خاصة مع تزايد الاهتمام بالزراعة الذكية.
قوة الموردين	**منخفضة إلى متوسطة:** تعتمد صناعة الروبوتات على مجموعة متنوعة من الموردين للمكونات الإلكترونية والمواد الخام. ومع ذلك، لا يوجد مورد واحد يسيطر على السوق بشكل كامل، مما يمنح الشركة مرونة في اختيار الموردين والحصول على أفضل الأسعار والشروط.
قوة المشترين	**متوسطة إلى مرتفعة:** يتمتع المزارعون وأصحاب البيوت الزراعية المحمية بقدرة تفاوضية جيدة، خاصة مع وجود بدائل أخرى لمكافحة الآفات مثل المبيدات الكيميائية والطرق التقليدية. ومع ذلك، يمكن للشركة تقليل قوة المشترين من خلال تقديم منتج عالي الجودة وخدمة عملاء ممتازة وبناء علاقات قوية مع المزارعين.
خطر المنتجات البديلة	**متوسطة:** توجد بدائل لمكافحة الآفات الزراعية، مثل المبيدات الكيميائية والطرق التقليدية مثل المكافحة اليدوية والفخاخ. ومع ذلك، يعاني استخدام المبيدات الكيميائية من مشاكل بيئية وصحية متزايدة، كما أن الطرق التقليدية قد تكون أقل فعالية وأكثر تكلفة على المدى الطويل. يوفر الروبوت بديلاً فعالًا ومستدامًا لهذه البدائل، مما يقلل من خطر المنتجات البديلة.
حدة التنافس الحالي	**منخفضة إلى متوسطة:** لا يزال سوق روبوتات الكشف عن الآفات في البيوت الزراعية المحمية في مرحلة النمو المبكرة، ولا يوجد عدد كبير من المنافسين المباشرين. ومع ذلك، من المتوقع أن تزداد حدة التنافس مع دخول المزيد من الشركات إلى هذا السوق. يمكن للشركة الحفاظ على ميزتها التنافسية من خلال الابتكار المستمر، وتقديم منتجات متطورة، وبناء علاقات قوية مع العملاء.

ثالثا - تحليل (SWOT):

S

نقاط القوة (Strengths)

التكنولوجيا المتقدمة: يعتمد الروبوت على تقنيات الذكاء الاصطناعي والرؤية الحاسوبية المتطورة، مما يمنحه قدرة فائقة على الكشف الدقيق للآفات في مراحل مبكرة.
دقة عالية في الكشف: يوفر الروبوت دقة عالية في تحديد أنواع الآفات وتقييم شدة الإصابة، مما يساعد المزارعين على اتخاذ قرارات مبنية على بيانات دقيقة.
توفير بيانات مفصلة: يجمع الروبوت بيانات قيمة حول أنواع الآفات ومواقعها وشدتها، مما يمكن المزارعين من تحليل هذه البيانات واتخاذ قرارات مستنيرة بشأن إدارة الآفات.
الاستدامة البيئية: يساهم الروبوت في تقليل استخدام المبيدات الكيميائية الضارة بالبيئة والصحة العامة، مما يعزز ممارسات الزراعة المستدامة.

W

نقاط الضعف (Weaknesses)

التكلفة الأولية: قد تكون تكلفة تطوير وإنتاج الروبوت مرتفعة في البداية، مما قد يمثل عائقًا أمام بعض المزارعين ذوي الميزانيات المحدودة.
الحاجة إلى صيانة دورية: يتطلب الروبوت صيانة دورية للتأكد من عمله بكفاءة، وقد يحتاج المزارعون إلى تدريب خاص للقيام بهذه الصيانة.

O

الفرص (Opportunities)

الطلب المتزايد على الزراعة الذكية: يشهد قطاع الزراعة الذكية نموًا متسارعًا، مما يوفر فرصة كبيرة لتسويق الروبوت وزيادة الطلب عليه.
الدعم الحكومي للابتكار الزراعي: تقدم العديد من الحكومات دعمًا ماليًا وتشجيعًا للابتكار في القطاع الزراعي، مما يمكن أن يساعد في تمويل المشروع وتسويقه.
التعاون مع الشركات الزراعية الكبرى: يمكن عقد شراكات مع شركات زراعية كبرى لتجربة الروبوت في مزارعهم وتسويقه على نطاق واسع.
التوسع في أسواق جديدة: يمكن استكشاف فرص التوسع في أسواق جديدة، مثل الدول التي تواجه تحديات كبيرة في مجال الآفات الزراعية..

T

التهديدات (Threats)

المنافسة من الشركات الأخرى: قد تواجه الشركة منافسة من شركات أخرى تعمل في مجال تطوير الروبوتات الزراعية.
التقلبات في أسعار المواد الأولية: قد تؤدي التقلبات في أسعار المواد الأولية إلى زيادة تكاليف الإنتاج وتقليل الربحية.
المخاطر التقنية: قد تواجه الشركة مخاطر تقنية، مثل الأعطال الفنية في الروبوت أو صعوبة في تحديث البرمجيات.

رابعا: المزيج التسويقي (4Ps):

- **المنتج (Product):**
  - **الخصائص والميزات:** روبوت ذكي متطور مزود بكاميرات عالية الدقة وأجهزة استشعار متقدمة، يعمل بتقنيات الذكاء الاصطناعي المتطورة للكشف المبكر والدقيق عن الآفات الزراعية في البيوت البلاستيكية.
  - **التصميم:** تصميم مدمج وخفيف الوزن، ليسهل تنقله داخل البيوت البلاستيكية دون الإضرار بالمحاصيل.
  - **الجودة:** ضمان أعلى معايير الجودة في تصنيع الروبوت ومكوناته، بما يضمن أداءً موثوقًا وعمرًا افتراضيًا طويلًا.
  - **العلامة التجارية:** بناء علامة تجارية قوية للروبوت تعكس الابتكار والجودة والموثوقية.
- **السعر (Price):**
  - **استراتيجية التسعير:** اعتماد استراتيجية تسعير تعتمد على القيمة (Value-based pricing)، حيث يتم تحديد السعر بناءً على القيمة التي يوفرها الروبوت للمزارعين من حيث زيادة الإنتاجية وتقليل الخسائر الناجمة عن الآفات.
  - **هيكل التسعير:** تقديم خيارات تسعير مختلفة لتلبية احتياجات المزارعين المختلفة، مثل البيع المباشر، والتأجير، والاشتراك الشهري أو السنوي.
- **التوزيع (Place):**
  - **قنوات التوزيع:** توزيع الروبوت من خلال قنوات متعددة، مثل:
    - البيع المباشر للمزارعين من خلال فريق المبيعات الخاص بالشركة.
    - التعاون مع شركات التوزيع الزراعي المتخصصة.
    - البيع عبر الإنترنت من خلال موقع الشركة الإلكتروني أو منصات التجارة الإلكترونية المتخصصة في المنتجات الزراعية.
- **الترويج (Promotion):**
  - **استراتيجية الترويج:**
    - **التسويق الرقمي:** استخدام وسائل التواصل الاجتماعي، والموقع الإلكتروني للشركة، و حملات البريد الإلكتروني للترويج للروبوت وميزاته وفوائده.
    - **العلاقات العامة:** بناء علاقات قوية مع وسائل الإعلام المتخصصة في الزراعة لنشر أخبار ومقالات عن الروبوت.
    - **المعارض والمؤتمرات الزراعية:** المشاركة في المعارض والمؤتمرات الزراعية لعرض الروبوت والتواصل مع المزارعين المحتملين.
    - **التسويق بالمحتوى:** إنشاء محتوى تعليمي و تثقيفي حول الآفات الزراعية وكيفية استخدام الروبوت لمكافحتها.

خامسا: الاستراتيجيات التسويقية:

- **استراتيجية التمايز (Differentiation Strategy):**
  - **التركيز على الميزات الفريدة:** تسليط الضوء على الميزات الفريدة للروبوت التي تميزه

عن الحلول الأخرى في السوق، مثل دقة الكشف العالية، والقدرة على التنقل الذاتي، وتقديم توصيات ذكية.

- **بناء علامة تجارية قوية:** بناء علامة تجارية قوية للروبوت تعكس الابتكار والجودة والموثوقية.
- **تقديم خدمة عملاء ممتازة:** توفير خدمة عملاء متميزة للمزارعين، بما في ذلك الدعم الفني والتدريب على استخدام الروبوت.

- **استراتيجية التركيز (Focus Strategy):**
  - **استهداف شريحة محددة:** التركيز على شريحة محددة من المزارعين، مثل أصحاب البيوت المحمية الذين يزرعون محاصيل ذات قيمة عالية.
  - **تلبية احتياجات الشريحة المستهدفة:** فهم احتياجات ومتطلبات الشريحة المستهدفة وتقديم حلول مخصصة تلبي هذه الاحتياجات بشكل فعال.

المحور الرابع: خطة الإنتاج والتنظيم

1. خطة الإنتاج:

- **اقتناء المواد الأولية:**
  - تحديد قائمة بجميع المواد الأولية والمكونات اللازمة لتصنيع الروبوت، بما في ذلك الكاميرات، وأجهزة الاستشعار، والمكونات الإلكترونية، والمواد المستخدمة في الهيكل الخارجي.
  - البحث عن موردين موثوقين يقدمون مواد عالية الجودة بأسعار تنافسية.
  - التفاوض مع الموردين للحصول على أفضل الشروط والعقود.
  - وضع خطة لإدارة المخزون وضمان توافر المواد الأولية في الوقت المناسب.
- **التصنيع:**
  - تصميم خط إنتاج فعال ومنظم لتصنيع الروبوت، مع تحديد المراحل والعمليات اللازمة.
  - توفير المعدات والأدوات اللازمة لعملية التصنيع، بما في ذلك آلات التجميع واللحام والاختبار.
  - توظيف فريق من الفنيين والمهندسين ذوي الخبرة في مجال تصنيع الروبوتات.
  - تطبيق إجراءات صارمة لمراقبة الجودة في جميع مراحل التصنيع.
- **التجميع والاختبار:**
  - تجميع المكونات المختلفة للروبوت وفقًا للمواصفات الفنية المحددة.
  - إجراء اختبارات شاملة للتأكد من عمل جميع مكونات الروبوت بشكل صحيح.
  - إجراء اختبارات ميدانية للروبوت في بيوت محمية حقيقية للتأكد من أدائه في الظروف الفعلية.
- **التعبئة والتغليف:**
  - تصميم عبوات مناسبة لحماية الروبوت أثناء النقل والتخزين.
  - توفير معلومات واضحة على العبوة حول كيفية استخدام الروبوت وصيانته.

2. خطة التنظيم:

- **هيكل تنظيمي:**
  - تحديد هيكل تنظيمي واضح للمشروع، مع تحديد الأدوار والمسؤوليات لكل عضو في الفريق.
  - تعيين مدير مشروع مسؤول عن الإشراف على جميع جوانب المشروع وضمان تنفيذه في الوقت المحدد وضمن الميزانية المحددة.
  - تشكيل فرق عمل متخصصة لكل مرحلة من مراحل الإنتاج، مثل فريق التصميم، وفريق التصنيع، وفريق الاختبار.
- **إدارة المشروع:**
  - وضع خطة عمل مفصلة تحدد المهام والجداول الزمنية والموارد اللازمة لكل مرحلة من مراحل المشروع.
  - استخدام أدوات إدارة المشاريع لتتبع التقدم المحرز وتحديد المخاطر المحتملة واتخاذ الإجراءات التصحيحية اللازمة.
  - التواصل المنتظم مع جميع أعضاء الفريق لتبادل المعلومات وتنسيق العمل.

3. التموين:

- **تحديد الموارد المالية اللازمة:**
  - تقدير التكاليف الإجمالية للمشروع، بما في ذلك تكاليف المواد الأولية، والمعدات، والعمالة، والتسويق.
  - البحث عن مصادر تمويل مختلفة، مثل الاستثمار، والقروض، والمنح الحكومية.
  - وضع خطة لإدارة الميزانية وضمان استخدام الموارد المالية بكفاءة.
- **إدارة سلسلة التوريد:**
  - بناء علاقات قوية مع الموردين لضمان توافر المواد الأولية والمكونات في الوقت المناسب وبأسعار تنافسية.
  - وضع خطط بديلة للتعامل مع أي مشاكل قد تحدث في سلسلة التوريد.

4. القوى العاملة:

- **تحديد الاحتياجات من العمالة:**
  - تحديد عدد الموظفين المطلوبين لكل مرحلة من مراحل الإنتاج، مع تحديد المهارات والمؤهلات اللازمة.
  - وضع خطة لتدريب الموظفين الجدد وتطوير مهاراتهم.
  - توفير بيئة عمل آمنة وصحية للموظفين.

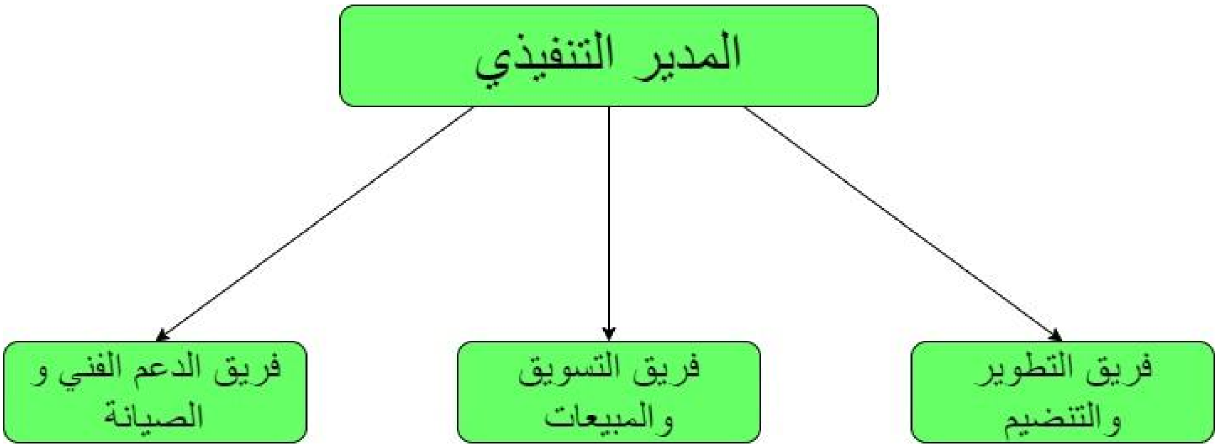
5. الشراكات الرئيسية:

- **الشراكات مع المزارعين:**
  - بناء علاقات قوية مع المزارعين وأصحاب البيوت المحمية للحصول على ملاحظاتهم واقتراحاتهم حول الروبوت.
  - إشراك المزارعين في عملية تطوير الروبوت وتجربته في مزارعهم.
- **الشراكات مع الشركات التقنية:**
  - التعاون مع شركات متخصصة في مجال الذكاء الاصطناعي و الرؤية الحاسوبية لتطوير وتحسين تقنيات الروبوت.
- **الشراكات مع مراكز الأبحاث الزراعية:**
  - التعاون مع مراكز الأبحاث الزراعية لتبادل المعلومات والخبرات حول الآفات الزراعية.

6. الجدول الزمني للإنتاج:

- **وضع جدول زمني مفصل لجميع مراحل الإنتاج، من اقتناء المواد الأولية إلى تسليم الروبوت للعملاء.**
- **تحديد المواعيد النهائية لكل مرحلة والتأكد من الالتزام بها.**

7. المخطط التنظيمي:



المحور الخامس: الخطة املالية FINANCIER PLAN :

تعتبر عنصرًا هاما في إدارة المشروع، وتشمل التكاليف والإيرادات وجدول حسابات الناتج المتوقع وخطة

الخزينة.

تكاليف المشروع واستهلاك الاستثمار.

تكاليف المشروع: تتمثل التكاليف الإجمالية للمشروع في التكاليف الاستثمارية والتكاليف التشغيلية

التكاليف الاستثمارية:

الأصول	التكلفة
الأثاث	10000
الآلات والمعدات	25000
رأس مال العامل	0
المجموع	35000

التكاليف التشغيلية:

الأصول	التكلفة
كراء المكتب	20000
مواد أولية	9000
الهاتف والانترنت	1000
أجور	0
المجموع	30000

1- الهيكل التمويلي: يتم تمويل مشروعنا بالاعتماد الكلي على الأموال الخاصة لصاحب المشروع وهذا ما يسمى بالتمويل الذاتي كما هو موضح في الجدول التالي:

البيان	النسبة	القيمة
أموال خاصة	100%	**100000 DZ**
القروض	0%	0
المجموع		**100000 DZ**

STARTUP : BugBot-Entreprise

	PRÉVISION					
Produit Destiné au Client	N	N+1	N+2	N+3	N+4	N+5
Quantité Produit a	25	50	75	100	150	250
Prix HT Produit a	19000					
Vente Produit a	8	20	50	80	120	240
Chiffre d'Affaires Global (DZ)	152000	380000	950000	1520000	2280000	4560000
	9842000					

نموذج العمل التجاري (BMC)

شرائح العملاء

- المزارعون التقليديون وأصحاب البيوت الزراعية المحمية.
- شركات إنتاج المحاصيل الزراعية.
- المهندسون الزراعيون والمستشارون الزراعيون.
- الباحثون في مجال الزراعة.

علاقات العملاء

- قديم خدمة عملاء ممتازة، بما في ذلك الدعم الفني والتدريب على استخدام الروبوت.
- بناء علاقات طويلة الأمد مع العملاء من خلال تقديم خدمات ما بعد البيع، مثل الصيانة والتحديثات.
- إنشاء مجتمع عبر الإنترنت للمزارعين لتبادل الخبرات والمعرفة حول استخدام الروبوت.

قنوات التوزيع

- البيع المباشر للمزارعين من خلال فريق المبيعات الخاص بالشركة.
- التعاون مع شركات التوزيع الزراعي.
- البيع عبر الإنترنت من خلال موقع الشركة الإلكتروني أو منصات التجارة الإلكترونية المتخصصة في المنتجات الزراعية.
- المشاركة في المعارض والمؤتمرات الزراعية.

قيمة العرض

- الكشف المبكر والدقيق عن الآفات الزراعية باستخدام تقنيات الذكاء الاصطناعي والرؤية الحاسوبية.
- توفير بيانات مفصلة عن أنواع الآفات ومواقعها وشدتها لمساعدة المزارعين في اتخاذ قرارات مبنية على البيانات.
- تقليل الاعتماد على المبيدات الكيميائية الضارة وتعزيز ممارسات الزراعة المستدامة.
- زيادة الإنتاجية الزراعية وتحسين جودة المحاصيل وتقليل الخسائر الناجمة عن الآفات.
- توفير الوقت والجهد في مراقبة الآفات وتقديم توصيات ذكية للمزارعين.

الأنشطة الرئيسية

- البحث والتطوير المستمر لتقنيات الذكاء الاصطناعي والرؤية الحاسوبية.
- تصميم وتصنيع الروبوت.
- تسويق الروبوت وتوزيعه.
- تقديم خدمات الدعم الفني والتدريب للعملاء.

الموارد الرئيسية

- فريق من الخبراء والمهندسين ذوي الكفاءات العالية في مجالات الذكاء الاصطناعي، والروبوتات، والزراعة.
- تقنيات الذكاء الاصطناعي والرؤية الحاسوبية المتقدمة.
- معدات التصنيع والاختبار.
- شبكة من الموردين والشركاء.

الشركاء الرئيسيون

- شركات التوزيع الزراعي المتخصصة.
- شركات التكنولوجيا المتخصصة في الذكاء الاصطناعي والرؤية الحاسوبية.
- مراكز الأبحاث الزراعية.
- المزارعون والمؤسسات الزراعية.

تدفقات الإيرادات

- بيع الروبوت.
- تقديم خدمات الصيانة والتحديثات.
- تقديم استشارات وخدمات تدريب للمزارعين.
- بيع البيانات والتحليلات التي يجمعها الروبوت لشركات الأبحاث الزراعية.

هيكل التكاليف

- تكاليف البحث والتطوير.
- تكاليف التصنيع والمواد الخام.
- تكاليف التسويق والتوزيع.
- تكاليف العمالة.
- تكاليف الصيانة والتحديثات.

Bibliography

[1] S. Cubero, E. Marco-Noales, N. Aleixos, S. Barbé, and J. Blasco, "Robhortic: A field robot to detect pests and diseases in horticultural crops by proximal sensing," *Agriculture*, vol. 10, no. 7, p. 276, 2020.

[2] C.-J. Chen, Y.-Y. Huang, Y.-S. Li, Y.-C. Chen, C.-Y. Chang, and Y.-M. Huang, "Identification of fruit tree pests with deep learning on embedded drone to achieve accurate pesticide spraying," *IEEE Access*, vol. 9, pp. 21 986–21 997, 2021.

[3] X. Zhang, J. Bu, X. Zhou, and X. Wang, "Automatic pest identification system in the greenhouse based on deep learning and machine vision," *Frontiers in Plant Science*, vol. 14, p. 1255719, 2023.

[4] Yolov8. [Online]. Available: https://docs.ultralytics.com/ar/models/yolov8/

[5] G. Tropea Garzia, G. Siscaro, A. Biondi, and L. Zappalà, "Tuta absoluta, a south american pest of tomato now in the eppo region: biology, distribution and damage," *EPPO bulletin*, vol. 42, no. 2, pp. 205–210, 2012.

[6] H. F. Van Emden and R. Harrington, *Aphids as crop pests.* Cabi, 2017.

[7] and others, *Whitefly of the world. A systematic catalogue of the Aleyrodidae (Homoptera) with host plant and natural enemy data.* John Wiley and Sons., 1978.

[8] J. G. Morse and M. S. Hoddle, "Invasion biology of thrips," *Annu. Rev. Entomol.*, vol. 51, pp. 67–89, 2006.

[9] D. J. Williams and M. C. Granara de Willink, *Mealybugs of Central and South America.*, 1992.

[10] R. G. Bland, "The orthoptera of michigan," *Michigan State University Extension, East Lansing*, 2003.

[11] H. Zhen, W. Gao, L. Jia, Y. Qiao, and X. Ju, "Environmental and economic life cycle assessment of alternative greenhouse vegetable production farms in peri-urban beijing, china," *Journal of cleaner production*, vol. 269, p. 122380, 2020.

[12] P. I. Kerchev, B. Fenton, C. H. Foyer, and R. D. Hancock, "Plant responses to insect herbivory: interactions between photosynthesis, reactive oxygen species and hormonal signalling pathways," *Plant, cell & environment*, vol. 35, no. 2, pp. 441–453, 2012.

[13] G. M. Gurr, S. D. Wratten, and J. M. Luna, "Multi-function agricultural biodiversity: pest management and other benefits," *Basic and Applied Ecology*, vol. 4, no. 2, pp. 107–116, 2003.

[14] K. Basavalingaiah, Y. Ramesha, V. Paramesh, G. Rajanna, S. L. Jat, S. Dhar Misra, A. Kumar Gaddi, H. Girisha, G. Yogesh, S. Raveesha *et al.*, "Energy budgeting, data envelopment analysis and greenhouse gas emission from rice production system: A case study from puddled transplanted rice and direct-seeded rice system of karnataka, india," *Sustainability*, vol. 12, no. 16, p. 6439, 2020.

[15] P. Boissard, V. Martin, and S. Moisan, "A cognitive vision approach to early pest detection in greenhouse crops," *Computers and Electronics in Agriculture*, vol. 62, no. 2, pp. 81–93, 2008. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0168169907002256

[16] A. Biondi, R. N. C. Guedes, F.-H. Wan, and N. Desneux, "Ecology, worldwide spread, and management of the invasive south american tomato pinworm, tuta absoluta: Past, present, and future," *Annual Review of Entomology*, vol. 63, no. Volume 63, 2018, pp. 239–258, 2018. [Online]. Available: https://www.annualreviews.org/content/journals/10.1146/annurev-ento-031616-034933

[17] V. Dharmaraj and C. Vijayanand, "Artificial intelligence (ai) in agriculture," *International Journal of Current Microbiology and Applied Sciences*, vol. 7, no. 12, pp. 2122–2128, 2018.

[18] N. C. Eli-Chukwu, "Applications of artificial intelligence in agriculture: A review." *Engineering, Technology & Applied Science Research*, vol. 9, no. 4, 2019.

[19] M. S. Farooq, S. Riaz, A. Abid, K. Abid, and M. A. Naeem, "A survey on the role of iot in agriculture for the implementation of smart farming," *IEEE Access*, vol. 7, pp. 156 237–156 271, 2019.

[20] M. Banu and S. C, "Iot architecture a comparative study," 11 2017.

[21] W. Viriyasitavat, T. Anuphaptrirong, and D. Hoonsopon, "When blockchain meets internet of things: Characteristics, challenges, and business opportunities," *Journal of Industrial Information Integration*, vol. 15, pp. 21–28, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2452414X19300202

[22] Y. Kalyani and R. Collier, "A systematic survey on the role of cloud, fog, and edge computing combination in smart agriculture," *Sensors*, vol. 21, no. 17, p. 5922, 2021.

[23] R. H. Ip, L.-M. Ang, K. P. Seng, J. Broster, and J. Pratley, "Big data and machine learning for crop protection," *Computers and Electronics in Agriculture*, vol. 151, pp. 376–383, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0168169917314588

[24] W. Tao, L. Zhao, G. Wang, and R. Liang, "Review of the internet of things communication technologies in smart agriculture and challenges," *Computers and Electronics in Agriculture*, vol. 189, p. 106352, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0168169921003690

[25] S. Pitla, S. Bajwa, S. Bhusal, T. Brumm, T. M. Brown-Brandl, D. R. Buckmaster, I. Condotta, J. Fulton, T. J. Janzen, M. Karkee *et al.*, "Ground and aerial robots for agricultural production: Opportunities and challenges," 2020.

[26] M. Chithambarathanu and M. Jeyakumar, "Survey on crop pest detection using deep learning and machine learning approaches," *Multimedia Tools and Applications*, vol. 82, no. 27, pp. 42 277–42 310, 2023.

[27] "Dropleaf: A precision farming smartphone tool for real-time quantification of pesticide application coverage," *Computers and Electronics in Agriculture*, vol. 180, p. 105906, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0168169920331112

[28] T. Balan, C. Dumitru, G. Dudnik, E. Alessi, S. Lesecq, M. Correvon, F. Passaniti, and A. Licciardello, "Smart multi-sensor platform for analytics and social decision support in agriculture," *Sensors*, vol. 20, no. 15, p. 4127, 2020.

[29] G. Messina and G. Modica, "The role of remote sensing in olive growing farm management: A research outlook from 2000 to the present in the framework of precision agriculture applications," *Remote Sensing*, vol. 14, no. 23, 2022. [Online]. Available: https://www.mdpi.com/2072-4292/14/23/5951

[30] I. El Naqa and M. J. Murphy, *What is machine learning?* Springer, 2015.

[31] E. Alpaydin, *Introduction to machine learning*. MIT press, 2020.

[32] J. E. Van Engelen and H. H. Hoos, "A survey on semi-supervised learning," *Machine learning*, vol. 109, no. 2, pp. 373–440, 2020.

[33] A. M. Andrew, "Reinforcement learning:: An introduction," *Kybernetes*, vol. 27, no. 9, pp. 1093–1096, 1998.

[34] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey," *Computers and electronics in agriculture*, vol. 147, pp. 70–90, 2018.

[35] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[36] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, 2015.

[37] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*. Springer, 2016, pp. 21–37.

[38] M. Hussain, "Yolo-v1 to yolo-v8, the rise of yolo and its complementary nature toward digital manufacturing and industrial defect detection," *Machines*, vol. 11, no. 7, p. 677, 2023.

[39] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[40] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," pp. 10 781–10 790, 2020.