International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

# Smart Plant Disease Management: Integrating Deep Learning and IoT for Rapid Diagnosis and Precision Treatment

**Prameeta Pai[1], Shubhan S Bhat[2], Lavanya G[3], Shabeena A[4], Rakshitha G[5]**

**Submitted**: 27/01/2024 **Revised**: 05/03/2024 **Accepted**: 13/03/2024

**Abstract:** Agriculture plays a crucial role in sustaining global food security, yet crop diseases pose significant threats to agricultural productivity. Traditional diagnostic methods often prove inefficient, prompting the exploration of innovative technologies like deep learning and the Internet of Things (IoT) for revolutionizing plant disease management. Deep learning algorithms offer the capacity to analyse extensive datasets of plant images, distinguishing between healthy and diseased with remarkable accuracy. Concurrently, IoT devices facilitate real-time data collection on crop health and environmental conditions, enabling early disease detection. As agricultural demands surge, enhancing crop resilience and yield becomes imperative, driving the integration of deep learning and IoT technologies.

***Keywords:*** *Deep learning, IOT, Disease Detection, Disease Management*

## 1. Introduction

Crop diseases are a Major concern for global food security as they can lead to significant economic losses and reduced crop yield[1]. Automated image classification systems are becoming increasingly essential in plant disease detection due to the Significant impact of diseases on agricultural productivity. With traditional methods proving to be inadequate for large-scale crops, researchers are focusing on image feature extraction and classification using deep learning approaches[2]. Deep learning models, particularly CNNs, have potential in accurately identifying plant diseases across different crops. Researchers have explored methods to optimize feature extraction and improve classification accuracy[3][4].

As the world population continues to grow, the importance of agriculture in providing food and sustenance cannot Exaggerated. However, the increasing demand for food has made it essential to adopt sustainable and efficient crop management strategies to ensure both adequate food supply and long-term agricultural sustainability. One of the most significant challenges to agricultural productivity is plant diseases, which can lead to massive economic losses and reduced crop yields. Traditional diagnostic methods have proven inadequate in addressing this issue, necessitating the exploration of innovative technologies such as Deep Learning and the Internet of Things (IoT) to revolutionize plant disease management.

Smart Plant Disease Management is an emerging field that leverages the power of deep learning and IoT technologies to analyze vast amounts of plant image datasets, enabling early disease Detection, and precise treatment interventions. This emerging field of Smart Plant Disease Management represents a paradigm shift in agricultural practices, empowering farmers with data-driven decision-making capabilities and enhancing overall crop health management.

[1] *BMS.College of Engineering, Bangalore, India*
[2] *BMS.College of Engineering, Bangalore, India*
[3] *BMS.College of Engineering, Bangalore, India*
[4] *BMS.College of Engineering, Bangalore, India*
[5] *BMS.College of Engineering, Bangalore, India*

The development of an autonomous wheeler robot using deep learning for plant disease detection and treatment. The envisioned system aims to accurately identify diseases through object detection, classify them, and administer targeted medication via automated spraying. By seamlessly integrating cutting-edge technologies, this system seeks to revolutionize precision agriculture, offering real-time disease management solutions and contributing to the sustainable enhancement of crop health.

The challenge of plant diseases in agriculture, employing a multifaceted approach integrating deep learning, IoT, and precision treatment mechanisms. The model utilized in this endeavour encompasses a range of deep learning architectures, including Convolutional Neural Networks (CNNs), YOLOv8, and ResNet models, chosen for their proven Effectiveness in image classification tasks

**Fig.1** Healthy Agricultural Field

## 2. Problem Statement

The main problem faced by the farmers is with their crop yield. Our focus is to address this problem and find out the factors which affect the crop yield. Our main aim is to Develop an autonomous wheeler robot using deep learning for plant disease detection and immediate treatment mechanism using IOT. The system should accurately identify diseases through object detection, classify them, and administer targeted medication via automated spraying, contributing to precision agriculture and crop health management.

**Fig 2 . Pesticide Spraying Mechanism**

### 3. Objectives

Our main objectives are: -

**A. To Develop a Deep Learning Model for Disease Identification**

To identify the disease in a plant we have leveraged the power of Object detection. We will train a deep learning model which is fine tuned to detect diseases from various parts of the plant.

There are several deep learning models which can be trained and fine-tuned for this purpose but we will be choosing the best model which has the highest accuracy

**Fig 3.1 Disease Identification**

**B. To Provide Precision Treatment Recommendations**

After successful identification the next step is to take measures to tackle that disease. Hence, we have also a developed a system which will give accurate treatment recommendations to the identified disease.

**Fig 3.2** Treatment Recommendations

**C. To Provide immediate Treatment to the disease**

Disease identification is a crucial part. But providing immediate treatment to the disease infected plant is also an important step. We have leveraged the power of IOT and combined it with the Deep learning model to dispense medicine as soon as the disease is detected.

**Fig 3.3** Spraying Medicine

### 4. Related Work

[5] The paper introduces an intelligent precision farming system that makes use of cognitive vision drones equipped with advanced deep learning models to autonomously identify and handle plant diseases. These drones come with high-resolution cameras for capturing images of plants in agricultural fields. [2] The paper presents a methodology for the classification of plant diseases in precision agriculture using Convolutional Neural Networks (CNNs) for texture feature extraction. It utilizes publicly available datasets containing plant leaf images and employs pre-trained CNN models, such as AlexNet, Vgg16, and ResNet, to extract texture features from different layers of these models.

[6] In this paper, the authors presented a new wood defect detection method called STC-YOLOv5, which improves the model's ability to detect defects of different types and sizes, including small and dense defects. The authors used reinforcement techniques to create a large dataset containing different types of wood defects.

[4] The authors created a dataset of plant diseases in five important horticultural crops in New Zealand. They then optimized a deep learning model called RFCN to detect plant disease on this dataset. They evaluated different data augmentation techniques, image resizers, weight initializers, batch normalization methods, and deep learning optimizers to improve the performance of the RFCN model.

[7] Presents an innovative approach to crop disease recognition in the agricultural industry using deep learning and IoT technology. The proposed system combines three stages of recognition and a compensation layer to achieve high accuracy in identifying crop diseases. It distinguishes between different disease levels and offers a valuable tool for crop management, especially in determining appropriate treatment protocols.

[3] The proposed method enhances the generalization ability and improves categorization accuracy compared to traditional classification networks. Furthermore, it consumes less memory, making it suitable for running on low performance terminals.

[1] This paper involves the development of a restructured dense residual network(RDN) for identification of tomato leaf disease. The RDN combines the advantages of deep residual networks and dense networks to improve the accuracy and efficiency.

[8] The study utilized two primary datasets: PlantVillage and PlantDoc for plant disease detection. This method enhances plant disease detection in agricultural settings with a dataset of 5,170 real-world plant disease images and state-of-the-art deep learning models, outperforming previous datasets.

[9] The method involves building a diverse dataset, training a deep learning model using transfer learning, and creating a user-friendly app for real-time plant disease and pest detection, with plans for data management, continuous improvement, and future enhancements.

[10] The authors propose a cheaper, autonomous, and easier to maintain robot that uses a combination of AlexNet to detect fire and ImageNet for detecting the type of fire.


**Fig 4.1 Cognitive Vision Drone**

## 5. Proposed System

Our proposed system contains of two main parts or subsystems. First part is to develop the object detection model for Realtime disease detection and the second part is an IOT system which after successful identification of the disease dispenses medicine. Let us discuss each subsystem in detail

### 5.1. Part one

In this phase we train and fine tune an object detection model to detect specific disease in plants. The accuracy of the object detection model depends on few factors:-

1.Quality and Quantity of the data.

2.Annotation Quality

3.Model Architecture

4.Hyper parameters and Training configuration

5.Testing the trained model on a new dataset

6.Validating the model

We are sure to achieve a well performing object detection model if we address all the above mentioned factors

We trained multiple object detection model to classify and detect diseases in plants. For our specific use case we have trained the model to detect diseases in tomato plant. Detailed implementation and training steps will be covered further. After the object detection model is ready we are moving forward to the next part of our project that is to give immediate treatment by utilizing the power of raspberry pi

### 5.2 Part two

In this part we use the trained object detection model to detect diseases in Realtime. We load the trained model on the raspberry pi 4 then use the raspberry pi camera to start the live feed. The trained model detects the disease with confidence. In yolo we have something called as the bounding box algorithm which is used in Realtime where it encloses the detected object in a box with certain confidence level. We make use of this. We have written an algorithm to activate the GPIO pins of the raspberry pi after reaching a certain level of confidence . The GPIO in turn activates the dispensing unit which dispenses medicine on the affected plant. This is how we provide immediate treatment to the plants.

Raberry pi's main advantage is that its compactness. It can be loaded on a wheeler module or a drone module and this whole system can be given mobility.

**Fig 5.1 Raspberry pi Model 4B**

## 6. Diagrammatic representation

The figure 6 shows the hardware circuit diagram

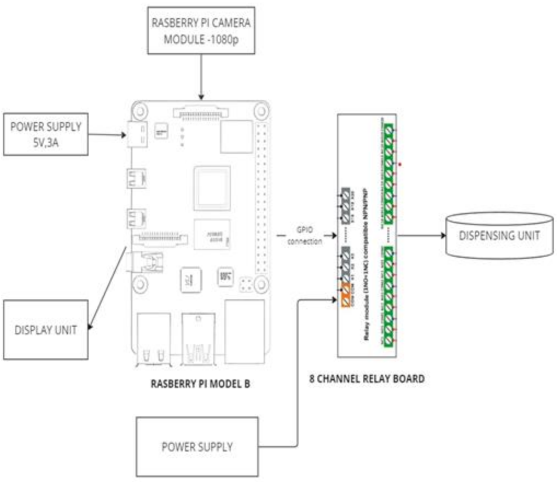


**Fig 6.1. Raspberry pi Controller Circuit**

In the above diagram we have used raspberry pi 4 module as our main component. A constant power supply is to be provided to the raspberry pi module. The yolo model is loaded on to the raspberry pi and then the object detection program is ran using the OpenCV and raspberry pi camera. A display is connected to the raspberry pi module . The 8 channel relay board is connected to a 12v power supply and to the GPIO pins of raspberry pi. The reason to do this is because the GPIO pins are not capable of controlling the motor on its own. As soon as the camera detects the disease the GPIO is activated. The Dispensing unit which consists of the water pump which dispenses the medicine.

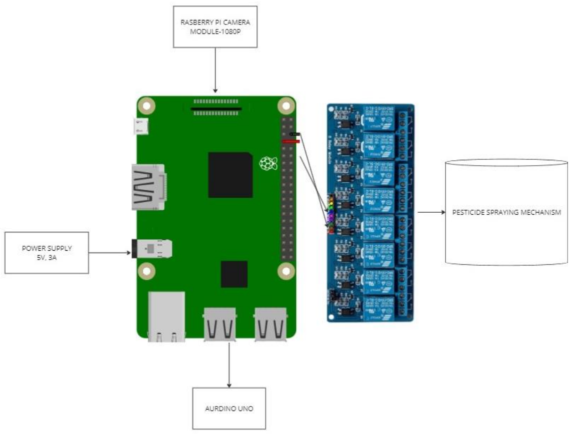


**Fig 6.2: Raspberry pi Connection unit**

## 7. Flowchart

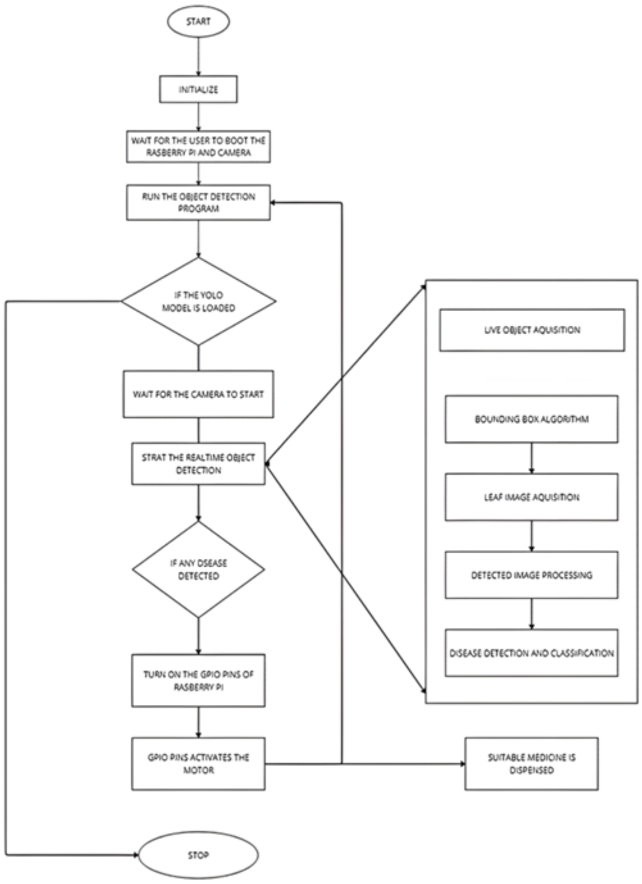


**Fig 7. ASM of the project**

## 8. Tools Used

8.1 Tools used for Software implementation

1. Python
2. Tensorflow
3. Pytorch

4. OpenCV

5. Anaconda

6. Jupyter Notebook

7. Visual Studio Code

8.2 Tools used for Hardware implementation

1. Raspberry Pi 4 Model B

2. Raspberry Pi Camera Module

3. 8 channel Relay Board

4. 5v Power supply

5. 12V Power supply

6. 5v water pump

7. Connecting wires

8. Jumper Wires

## 9. Implementation and testing

### 9.1 Software Implementation

#### 9.1.1. Collecting dataset

We collected a large no of tomato leaf disease images specifically belonging to these classes

a .Tomato_Early_Blight

b. Tomato_Late_Blight

c. Tomato_Yellow_Curl_leaf_disease

d. Tomato_Mossaic_virus

We collected these from Google, Kaggle, and some manually taken photos. Totally we got 3000 images After collection its time for annotation

#### 9.1.2. Annotation and Augmentation

Annotation is a crucial step when training an object detection model. The more in the quality of annotation the more is the model's accuracy.

Hence, we used a tool called Roboflow to annotate and augment our dataset.

Roboflow is a modern dataset and tool which is better than label img and helps you to manage, assign, annotate and gives various technologies to perform preprocessing on your data.

We uploaded the image in roboflow and manually annotated each image according to it classes. Then we exported it as dataset. Roboflow also gives you options to manually split the dataset into train, test and validation set. We split 80% of data into train, 10% into test, 10 % into validation

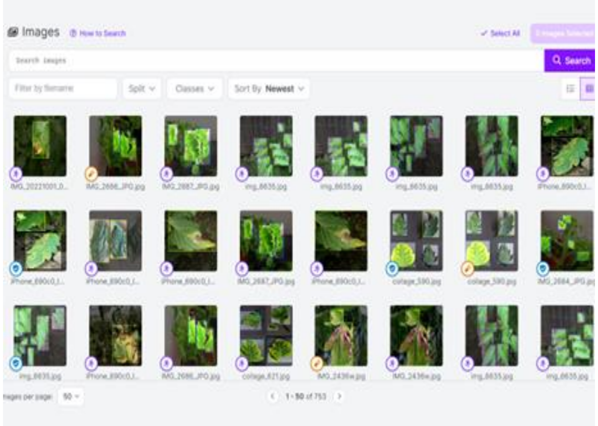


**Fig 9.1.1** Dataset Preparation in roboflow

After the split there is an option to export it. In this step there are lot of preprocessing steps which can be applied automatically like augmentation, rotation, resizing, Gray scaling etc. We only used augmentation and resizing it 654x654. After this step is done the dataset is exported as code which will be used in the Google Colab notebook for training the model

#### 9.1.3 Training the YoloV8 model

We trained the Yolo model in a colab environment. The main reason for doing this is that colab notebooks can be shared easily among people so contributions will be easy and it also provides a free GPU called T4 GPU which can be used to train the model. The detailed training steps are discussed below:-

**a. First we have to install the following requirements**

a. Ultralytics

b. Roboflow

b. Then clone the robflow dataset using the code which you get after exporting the dataset

c. Set the Yolov8 model size, number of epochs you want to train your model on. As this is a training step we are using train mode.

We have used Yolov8s model and trained it on 100 epochs

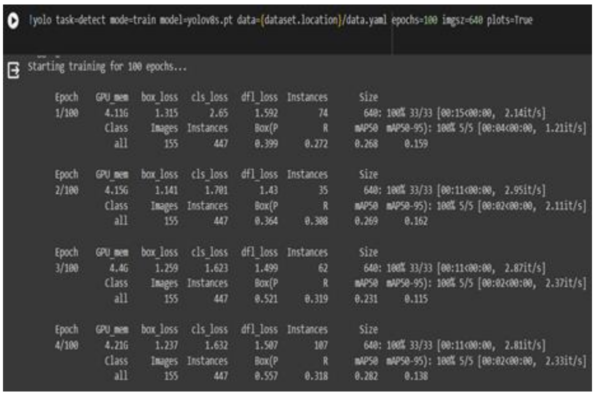


**Fig 9.1.2 Yolov8 Training**

d. From fig 9.1.2 we can note that as the no of epochs increase train loss decreases and the train accuracy increases which is a good sign. Totally after training the model for 100 epochs the mAP50 score of the model was found to be 0.986 or 98% which is a good score

e. The trained model is saved as best.pt which is a pytorch model. This trained yolo model will be used for further testing on the test dataset

f. The best.pt model was tested to see whether it's able to detect new images and classify according to the classes.


**Fig 9.1.3 Prediction Results**

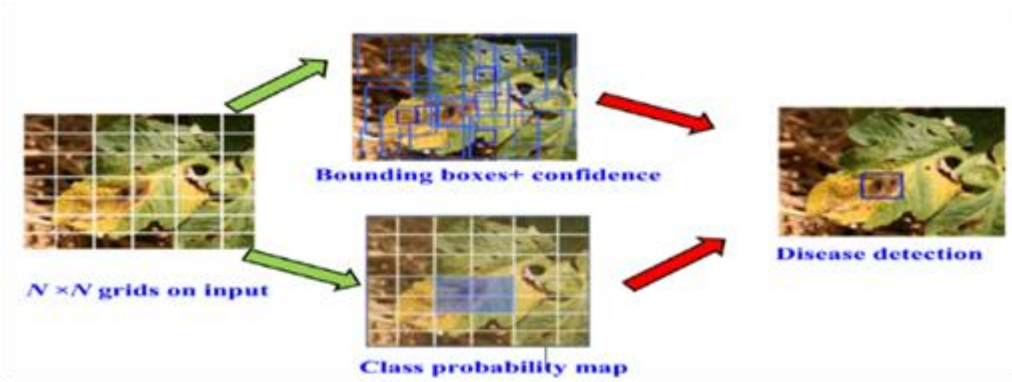


**Fig 9.1.4. Yolov8 Disease detection Flow**

After the Yolov8 model is properly trained and tested the models weights are saved. These weights are downloaded as .pt file which will be used for our application. Leveraging Opencv we used an algorithm which detects disease in realtime. When the program runs the webcam opens up and the yolo model starts receiving the live feed. As soon as it spots the disease, bounding box is made over that area as you can see in fig 9.1.5 and after reaching a certain level of confidence the image of the detected disease is saved.


**Fig 9.1.5 Detected disease**

**9.2 Hardware implementation**

**1. Installing the raspberry pi OS**

The first step before using the developed program on the raspberry pi is to install a strong os which can handle all the process

For this purpose, first we installed the raspberry pi imager on our system used a sd card then loaded the arch46 ubuntu os on the sd card

The sd card will be used as a Hard disk for the raspberry pi.

**2. Installing the required software on the Raspberry pi**

We installed all the following software on the raspberry pi

a. Vs code

b. Python

c. Anaconda

d. OpenCV

e. Ultralytics

f. GPIO

We create our own conda environment and install all the requirements.

**3. Connecting the raspberry pi to all the required hardware**

Connect the 8 channel Relay board to the GPIO pins of the raspberry pi board. Connect the medicine dispensing module to the relay board.

The relay board is givne a 12v power supply. This is done because the GPIO pins of the rabserry pi board can only output 5v which is not sufficient to power the water pump(Dispensing unit). Hence an extra power is supplied to via the Relay board. The full figure of connection is shown in Fig 9.2.1**.**

**4. Realtime plant disease detection by using raspberry pi4 and raspberry pi camera**

This system utilizes the trained yolov8 model and the computer vision(OpenCV) algorithm to analyze the images in real-time. This developed system detects the disease in real-time and captures the image if any disease is detected. This developed system is highly efficient and cost effective. Fig 17 shows how the developed system looks like

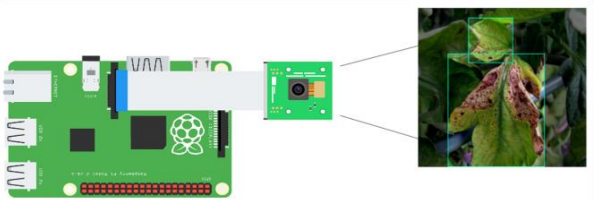

**Fig 9.2.1. Realtime disease detection using Raspberry pi 4**

**5. Medicine dispensing Mechanism using raspberry pi and raspberry Relay board**

First the disease is detected using the raspberry pi real-time detection system. Then an algorithm is developed to activate the GPIO pins of the raspberry pi. The GPIO sends signal to the relay board which is activates the water pump which dispenses the medicine. This helps in taking action immediately after a certain disease is detected. This actually reduces the wastage of medicine and practices the environmentally friendly practice in agriculture. Fig 9.2.1 shows the connection of the raspberry pi with medicine spraying mechanism

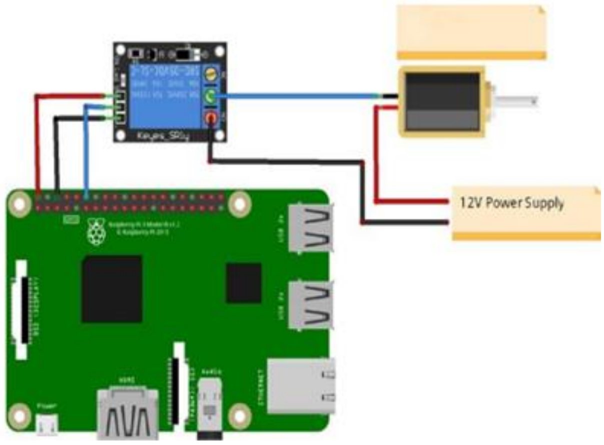


**Fig 9.2.2 Raspberry pi medicine spraying mechanism**

## 10. Model Comparison

In this implementation we have also evaluated different deep learning models for the same dataset. The models assessed include MobileNet, CNN and YOLOv8.

### 10.1 Experimental Setup

1. **Dataset**

   Different classes of tomato leaf diseases where annotated

2. **Evaluation Metrics**

   The models where evaluated on the following metrics

   1. Precision
   2. Recall
   3. F1 score

### 10.2 Results

Table 1 summarizes the performance metrics of each model

| Model | Precision | Recall | F1 Score |
|---|---|---|---|
| MobileNet | 0.994958 | 0.9949989 | 0.9949743 |
| CNN | 0.9327122 | 0.921627 | 0.9252525 |
| YOLO | 0.993 | 0.869 | 0.9268711 |

**Table 1. Comparison table of the model**

### 10.3 Observations

The precision of an object detection model can be used to evaluate the ability of the model to correctly identify positive samples from the total predicted positive instances. It quantifies the accuracy of positive detections made by the model.

Precision measures the proportion of correctly predicted positive instances (true positives) out of all instances predicted as positive by the model (true positives + false positives).

It reflects the model's ability to avoid false positives

Precision can be calculated by:

$$Precision = \frac{TruePositives}{FalsePositives + TruePositives}$$

where:

**True Positives (TP)** are the instances that are actually positive (e.g., disease present) and are correctly predicted as positive by the model.

**False Positives (FP)** are the instances that are actually negative (e.g., disease absent) but are incorrectly predicted as positive by the model.

From the results we can infer that Mobile net achieved the highest precision followed by YOLOv8 and then CNN.

A high precision value indicates that when the model predicts a positive result, it is likely to be correct.

Recall, also known as sensitivity or true positive rate, is a metric used to evaluate the completeness or coverage of a classification model in identifying all relevant instances (positives) from a dataset

Recall measures the proportion of correctly predicted positive instances (true positives) out of all actual positive instances in the dataset (true positives + false negatives). It quantifies the model's ability to identify all relevant instances of a particular class

Recall is calculated using the following formula:

$$Recall = \frac{TruePositives}{FalseNegatives + TruePositives}$$

where:

**True Positives (TP)** are the instances that are actually positive (e.g., disease present) and are correctly predicted as positive by the model.

**False Negatives (FN**) are the instances that are actually positive (e.g., disease present) but are incorrectly predicted as negative (e.g., disease absent) by the model.

The Mobilenet model has the highest Recall followed by CNN and then the YOLOv8

A high recall value indicates that the model is good at capturing most of the positive instances in the dataset

The F1 score for an object detection model is a measure that combines precision and recall, specifically adapted for the task of detecting objects within images or video frames. Object detection involves not only classifying objects but also localizing them with bounding boxes

To calculate the F1 score for an object detection model, we first need to calculate the Precision and Recall of an object detection model

F1 score is calculated using the following formula:

$$2 \times \frac{Precision + Recall}{Precision \times Recall}$$

In object detection tasks, true positives (TP) are defined as correctly detected objects with correct localization (i.e., predicted bounding boxes that sufficiently overlap with ground truth bounding boxes). False positives (FP) occur when the model predicts an object that does not exist or when the localization is inaccurate. False negatives (FN) occur when a ground truth object is not detected by the model

As precision and recall is higher in Mobile net the F1 score of the Mobile Net model is the highest followed by CNN and then YOLOv8 model.

Although mobilenet model had the highest accuracy we went ahead with YOLOv8 because of its compatibility and easiness in Realtime object detection. We improved the model accuracy by training it for more no of epochs.

### 11. Conclusion

The amalgamation of deep learning and Internet of Things (IoT) technologies has ushered in a new era in agricultural practices, particularly in the domain of smart plant disease management. This innovative synergy promises to revolutionize the way we approach crop health, offering swift and accurate diagnosis coupled with precise treatment strategies. As we delve into the intricate details of this integration, it becomes evident that its implications extend far beyond conventional farming methodologies.

At the core of this transformative approach lies deep learning, a subset of artificial intelligence that excels at pattern recognition and complex data analysis. By leveraging deep learning algorithms, agricultural experts can swiftly and accurately identify signs of plant diseases, often before visible symptoms manifest. This early detection is crucial in preventing the spread of diseases and minimizing crop losses. The ability of deep learning models to continuously learn and adapt enhances their diagnostic accuracy over time, making them indispensable tools for modern plant health management. Complementing the prowess of deep learning is the integration of IoT devices into agricultural landscapes. These devices, ranging from sensors to drones, facilitate the collection of real-time data from the field. This influx of data provides a dynamic and comprehensive view of the environmental conditions affecting crops. IoT-enabled devices can monitor factors such as soil moisture, temperature, humidity, and even detect subtle changes in plant physiology. This constant stream of information equips farmers with a nuanced understanding of their crops' health, enabling proactive decision-making.

The synergy between deep learning and IoT not only aids in early disease detection but also allows for precision treatment strategies. Armed with a wealth of data, farmers can tailor interventions with a level of specificity previously unimaginable.

**Table 1.** Example of full page table

| ***Fungal Symptom*** | ***Causal organism*** | ***Family*** | ***Order*** |
|---|---|---|---|
| Mango Anthracnose | Glomerella cingulata | Glomerellaceae | Incertaesedis |
| Mango Powdery mildew | Oidium mangiferae | Erysiphaceae | Erysiphales |
| Pomegranate Anthracnose | Glomerella cingulata | Glomerellaceae | Melanconiales |
| Grape Anthracnose | Elsinoë ampelina | Elsinoaceae | Incertaesedis |

## References


[1] CHANGJIAN ZHOU , SIHAN ZHOU , JINGE XING , AND JIA SONG, "Tomato Leaf Disease Identification by Restructured Deep Residual Dense Network," in IEEE Access, vol. 9, Feb 2021, doi: 10.1109/ACCESS.2021.3058947

[2] S. Barburiceanu, S. Meza, B. Orza, R. Malutan and R. Terebes, "Convolutional Neural Networks for Texture Feature Extraction. Applications to Leaf Disease Classification in Precision Agriculture," in IEEE Access, vol. 9, pp. 160085-160103, 2021, doi: 10.1109/ACCESS.2021.3131002.

[3] YANG WU, XIAN FENG1, AND GUOJUN CHEN, "Plant Leaf Diseases Fine-Grained Categorization Using Convolutional Neural Networks," in IEEE Access, vol. 10, Apr 2022, doi: 10.1109/ACCESS.2022.3167513.

[4] Muhammad Hammad Saleem, Johan Potgieter, Khalid Mahmood Arif, "A Performance-Optimized Deep Learning-Based Plant Disease Detection Approach for Horticultural Crops of New Zealand," in IEEE Access, vol. 10, pp. 89798-89822, 2022, doi: 10.1109/ACCESS.2022.3201104.

[5] Latif, Ghazanfar et al. 'Deep Learning Based Intelligence Cognitive Vision Drone for Automatic Plant Diseases Identification and Spraying'. 1 Jan. 2020 : 8103 – 8114.

[6] S. Han, X. Jiang and Z. Wu, "An Improved YOLOv5 Algorithm for Wood Defect Detection Based on Attention," in IEEE Access, vol. 11, pp. 71800-71810, 2023, doi: 10.1109/ACCESS.2023.3293864

[7] W. -J. Hu, J. Fan, Y. -X. Du, B. -S. Li, N. Xiong and E. Bekkering, "MDFC–ResNet: An Agricultural IoT System to Accurately Recognize Crop Diseases," in IEEE Access, vol. 8, pp. 115287-115298, 2020, doi: 10.1109/ACCESS.2020.3001237.

[8] E. Moupojou et al., "FieldPlant: A Dataset of Field Plant Images for Plant Disease Detection and Classification With Deep Learning," in IEEE Access, vol. 11, pp. 35398-35410, 2023, doi: 10.1109/ACCESS.2023.3263042.

[9] L. Li, S. Zhang and B. Wang, "Plant Disease Detection and Classification by Deep Learning— A Review," in IEEE Access, vol. 9, pp. 56683-56698, 2021, doi: 10.1109/ACCESS.2021.3069646.

[10] Dhiman, Amit and Shah, Neel and Adhikari, Pranali and Kumbhar, Sayli and Dhanjal, Inderjit Singh and Dhanjal, Inderjit Singh and Mehendale, Ninad, Fire Fighter Robot with Deep Learning and Machine Vision (June 23, 2020). Available at SSRN: https://ssrn.com/abstract=3633609 or http://dx.doi.org/10.2139/ssrn.3633609