



Daffodil
International
University

Deep Learning approach for detecting diseases in rose plants

Submitted By

MAZHAR UDDIN FAHIM

212-35-740

Department of Software Engineering
Daffodil International University

Supervised By

Md Rajib Mia

Lecturer (Senior Scale)

Department of Software Engineering

Thesis submitted in fulfillment of the requirement for the
award of the degree of
Bachelor of Science

Summer - 2025

APPROVAL

This thesis titled on “**Deep Learning approach for detecting diseases in rose plants**”, submitted by **Mazhar Uddin Fahim (ID: 212-35-740)** to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

BOARD OF EXAMINERS



Chairman

Dr. Imran Mahmud
Professor & Head
 Department of Software Engineering
 Faculty of Science and Information Technology
 Daffodil International University



Internal Examiner 1

Md Shohel Arman
Assistant Professor
 Department of Software Engineering
 Faculty of Science and Information Technology
 Daffodil International University



Internal Examiner 2

Md. Rajib Mia
Lecturer (Senior Scale)
 Department of Software Engineering
 Faculty of Science and Information Technology
 Daffodil International University

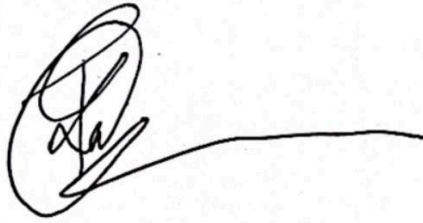


External Examiner

Md Habibur Rahman
Associate Professor
 Department of Computer Science and Engineering
 Islamic University, Bangladesh

SUPERVISOR'S DECLARATION

I hereby declare that I have checked this thesis and in my opinion, this thesis is adequate in terms of scope and quality for the award of the degree of Bachelor of Science..



(Supervisor's Signature)

Full Name : Md Rajib Mia

Position : Senior Lecturer

Date : 03 August 2025

STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Daffodil International University or any other institution.

Fahim

(Student's Signature)

Full Name : MAZHAR UDDIN FAHIM

ID Number : 212-35-740

Date : 03 August 2025

Deep Learning approach for detecting diseases in rose plants

MAZHAR UDDIN FAHIM

212-35-740

Thesis submitted in fulfillment of the requirements
for the award of the degree of
Bachelor of Science

Department of Software Engineering

DAFFODIL INTERNATIONAL UNIVERSITY

Summer 2025

ACKNOWLEDGEMENTS

Thank you from the bottom of my heart to everyone who helped me finish my thesis.

I want to thank my supervisor, Md Rajib Mia, from the bottom of my heart for always being there for me, giving me helpful feedback, and guiding me. All of these things have been very important to the success of this project. His clear directions, helpful comments, and profound knowledge of the subject have made writing the thesis faster and more useful. I really appreciate all the big ways he helped me become a better student.

I am really grateful to my family for always being there for me, loving me, and being patient with me. Their faith in me has always inspired me, even when things are at their worst. Their emotional and practical guidance has allowed me to finish my thesis and concentrate on my study. I will always be thankful to them because this route would have been much harder without them.

Finally, I want to thank Daffodil International University for offering me a place to learn, tools, and the possibility to achieve my academic goals. The school has had a huge effect on my schoolwork.

This thesis is for everyone who helped and encouraged me, as well as the folks stated above. I appreciate all of your generosity and faith in me.

ABSTRACT

Many places in Bangladesh grow roses for sale. People like to call them "the queen of flowers." It's very important to find diseases in rose plants early so they can grow well and make a lot of flowers. It takes a lot of work and time to find diseases the old-fashioned way, which makes it hard to act quickly. The main goal of this thesis is to find out how well the YOLOv8 model can find diseases in rose trees in all of its forms, including YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x. The YOLOv8 deep learning method is used to find common rose diseases like black spot and downy mildew. The thesis looks at the different types of YOLOv8's precision, recall, and mAP scores to see how well they can find diseases after a lot of training and review. The results show how well the YOLOv8 model can find diseases on rose leaves, especially when the leaves are small or medium-sized. This means that the model could be used right now to help keep diseases from spreading in crops. This essay talks about how YOLOv8, a type of deep learning, can help keep track of the health of rose plants. It also sets the stage for future work on finding crop diseases.

TABLE OF CONTENT

DECLARATIONS	i
TITLE PAGE	iv
ACKNOWLEDGEMENTS	v
ABSTRACT	vi
TABLE OF CONTENT	vii
LIST OF TABLES	viii
LIST OF FIGURES	viii
CHAPTER 1: INTRODUCTION	9
1.1 Introduction.....	9
1.2 Problem statement.....	10
1.3 Motivation.....	11
1.4 Objectives.....	12
1.5 Summary.....	12
CHAPTER 2: BACKGROUNDS	13
2.1 Introduction.....	13
2.2 Literature review.....	14
2.3 Summary.....	26
CHAPTER 3: METHODOLOGY	27
3.1 Overview.....	27
3.2 Detailed methodology.....	28
3.3 Summary.....	33
CHAPTER 4: RESULT AND DISCUSSION	34
4.1 Environment setup.....	34
4.2 Testing and Evaluation.....	35
4.3 Result and discussion.....	40
4.4 Limitation.....	42
4.5 Future work.....	42
CONCLUSION	43
REFERENCES	44

LIST OF TABLES

Table 2.2	Summary of Literature Review	14
Table 3.2.1	Dataset Overview	29
Table 4.3.1	Comparison table of YOLOv8 all version	40

LIST OF FIGURES

Figure 3.2	Methodology.....	28
Figure 3.2.2	Fresh Leaf[1], Downy Mildew[2,3], Black Spot[4,5].....	29
Figure 3.2.3	YOLOv8 Model Architecture.....	31
Figure 4.2.1	YOLOv8n model train.....	35
Figure 4.2.2	YOLOv8s model train.....	36
Figure 4.2.3	YOLOv8m model train.....	37
Figure 4.2.4	YOLOv8l model train.....	38
Figure 4.2.5	YOLOv8x model train.....	39
Figure 4.3.2	YOLOv8m result	41

CHAPTER 1

INTRODUCTION

1.1 Introduction

Diseases are very bad for plants because they stop them from growing. Plant health is important for keeping the world in balance and getting good crops. A lot of people grow roses because they look nice and are good for business. A lot of people grow roses because they look nice and can make them money. On the other hand, powdery mildew, downy mildew, and black spots can be bad for their health and performance. Finding diseases the old-fashioned way, like by looking at someone, takes a lot of time and can be wrong. This makes early disease detection challenging, particularly in crowded farming settings. This thesis automatically detects and classifies diseases that harm rose bushes using deep learning techniques and a system called YOLOv8 (You Only Look Once version 8). It is commonly known that YOLOv8 has a high capacity and accuracy for predicting current events. Aside from that, it works well with big files and produces accurate results on schedule. Using all five YOLOv8 varieties, this work creates a good model for detecting downy mildew and black spots on healthy rose plant leaves. Many photos of rose trees in different settings were used to train and refine the YOLOv8 model. The data was used to illustrate which plants were healthy and which weren't during the model-building process. More rapid and accurate disease eradication is the main goal. In addition, plant scientists, horticulturists, and farmers are searching for a labor-free way to maintain the health of their plants. YOLOv8 can be used on farms with continuously shifting conditions because it can detect the current state. Farmers can take prompt action to safeguard their crops and maintain the health of their plants. We evaluate the performance of the YOLOv8 model by measuring memory, accuracy, and precision. We then examine the findings and compare them to those of other widely used methods for identifying plant diseases. Compared to these two methods, YOLOv8 is more effective at identifying illnesses. The study also talks about how important it is to make datasets, since the model's performance depends on the quality and variety of the

images. Finding diseases early is important so they don't spread and so you don't have to use as many chemicals to treat them. This helps with growing in ways that are good for the environment. This thesis uses the YOLOv8 to show how deep learning can make farming better. Farmers are more likely to use methods that are better for the environment, and it gives them a way to deal with rose plant diseases that is safe, easy to scale, and effective.

1.2 Problem Statement

Three diseases that can harm rose plants' health, yield, and appearance are powdery mildew, black spot, and downy mildew. The rose-growing industry is at serious risk because rose plants are susceptible to these diseases. Manual examination and other antiquated methods of disease detection are laborious, time-consuming, and prone to errors. Unfortunately, this makes it more difficult to identify and treat illnesses in a timely manner. Rose plant diseases are more difficult to diagnose using traditional diagnostic techniques because they struggle to distinguish between multiple diseases from a single image. Multi-class classification, difficult-to-understand models, and a lack of datasets are some of the issues that prevent deep learning approaches from being applied in practical settings. Though these issues still need to be resolved, deep learning approaches have shown promise in automating the search for diseases. These problems lead to predictions that are either inaccurate or unreliable. This reduces the usefulness of automated systems in actual agricultural settings. Due to all of these issues, we must develop a more precise, efficient, and scalable method of disease detection. Such a solution could greatly facilitate the maintenance of the health of rose plants and assist farmers in adopting more environmentally friendly practices.

1.3 Motivation

This thesis is driven by the increasing challenges in agriculture, particularly in relation to the health of rose plants, which are highly susceptible to diseases such as powdery mildew, downy mildew, and black spot. Additionally, due to their extensive use in the floral industry, rose plants have a high commercial value. Roses need to be kept in good condition throughout the year due to their high seasonal demand and their prices skyrocketing during peak seasons. However, physical examination and other conventional methods of disease detection are time-consuming, prone to human error, and often inadequate in halting the fast spread of infections, especially when demand is high. The program aims to increase the accuracy and automation of disease detection using deep learning tools such as the YOLOv8 object recognition framework. In addition to increasing efficiency, YOLOv8's real-time detection features allow for early detection, which is essential for preserving healthy rose fields and meeting the high demand during peak seasons. I want to use my work to promote sustainable farming practices by reducing the use of chemicals, preventing crop loss, and providing a scalable solution that increases production and ensures a consistent supply of high-quality roses throughout the year.

1.4 Objectives

The study aims to identify real-time diseases affecting rose plants, specifically black spot and downy mildew, together with fresh leaves. The investigation is aimed at achieving the following specific objectives:

- Gather data and create a dataset consisting of three classes for the purpose of detecting diseases in rose plants.
- Improve the quality of datasets and the generalisation of models through the application of data annotation techniques.
- Apply the YOLOv8 model to facilitate real-time identification of diseases in rose plants.
- Compare the effectiveness of the model by utilising established metrics including accuracy, precision, recall, F1-score, and inference speed.
- Conduct cross-validation and evaluate on unseen data to determine the robustness of the dataset and the generalisation of the model.

These are the main objectives of my thesis.

1.5 Summary

In this thesis, a deep learning model based on YOLOv8 is developed for the real-time detection of rose plant diseases, with a focus on black spot, downy mildew, and healthy leaves. The research aims to automate disease detection in order to overcome the limitations of traditional techniques. We will improve the YOLOv8 model so that it is faster and more accurate. We will use standard metrics and cross-validation to test its performance. The goal is to create a disease management system for rose cultivation that can grow with the business and work in real time. This will help farmers use fewer chemicals and lose fewer crops.

CHAPTER 2

BACKGROUNDS

2.1 Introduction

Finding the disease early is very important to stop it from spreading and limit losses. But traditional methods, like looking at things, can be hard, take a long time, and make mistakes, especially when they are used on a large scale. Because of these problems, farmers may have a harder time spotting problems early on, which could mean they miss chances to help. As technology gets better, more and more people are using automated ways to find plant diseases. The reason for this is that they are Channel 4 accounting interviewed people who had been in the area for a long time..Roses can look bad, get sick, and lose value in the market because of diseases like powdery mildew, black spot, and downy mildew. People really like and value roses as decorations and in business. These diseases not only make roses look less attractive, but they also slow down their growth and lower their yields.These technologies could be used more, but there are problems like the quality of the data that is available right now and the fact that it's hard to tell the difference between different diseases in the same plant.It's also still hard to make systems that work well in a wide range of environmental conditions. Even with these problems, better ways to find plant diseases are needed for farming to be more productive and crops to be healthier. We need to keep working on these real-world farming problems.

2.2 Literature Review

Recent advancements in plant disease detection have leveraged deep learning, particularly YOLO models. YOLOv8, enhanced with techniques like Dynamic Snake Convolution (DySnakeConv) and Super Token Attention (STA), has shown significant improvements in detecting plant diseases, achieving a 3.3% increase in mAP. The SerpensGate-YOLOv8 model, trained on the PlantDoc dataset, demonstrated strong performance with a precision of 0.719 and mAP of 64.9%. Nonetheless, obstacles such as misclassification arising from disease similarities and environmental influences persist. These models demonstrate potential for immediate agricultural applications; however, additional optimisation is required to enhance generalisation and robustness in intricate environments.

Table 2.2: Summary of Literature Review

Title	Limitation	Methodology	Key Findings
Performance Evaluation of YOLO Models in Plant Disease Detection.	The research only tested YOLOv5, YOLOv7, and YOLOv8 on a single citrus leaf dataset (CCL'20), so results may not generalize well to other crops or real-world farm conditions. The dataset size was also relatively small compared to large-scale agricultural diversity, and the experiments were conducted in controlled settings rather than live field environments. Finally, only small YOLO variants were evaluated, while larger versions might perform differently.	The study employed the CCL'20 citrus leaf dataset, applying preprocessing and augmentation to improve data quality. YOLOv5s, YOLOv7, and YOLOv8s models were trained for up to 200 epochs on Google Colab with hyperparameter tuning. Performance was Roboflowed using mAP@50-95, compared against benchmark results, and the best model (YOLOv8s) was deployed on Roboflow for practical use.	YOLOv8 performed the best among the models, achieving a mAP@50-95 of 96.1% for citrus disease detection. The model showed significant improvement over previous versions and outperformed other models in detecting diseases such as Anthracnose, Melanose, and Bacterial Brown Spot. YOLOv8 demonstrated the capability to detect multiple instances of the same or different diseases in a single image. The model was deployed on the Roboflow platform for public use.

<p>Improved YOLOv8 and SAHI Model for the Collaborative Detection of Small Targets at the Micro Scale: A Case Study of Pest Detection in Tea</p>	<p>The study is limited by difficulties in detecting small targets in complex backgrounds and higher computational costs from deeper networks. Added structural complexity may also introduce instability, and results require validation on larger, more diverse, real-world datasets.</p>	<p>The study improved YOLOv8 for tea pest detection by integrating slicing-aided fine-tuning and hyper inference (SAHI) to enhance small-target recognition. A lightweight backbone (GELAN) was introduced for better feature extraction, an MS structure was applied for multi-scale feature fusion, and the BiFormer attention mechanism strengthened target features. Finally, an inner-MPDIoU loss function replaced the original to improve bounding box regression accuracy on small pests.</p>	<p>The enhanced YOLOv8 model achieved a precision of 96.32% and recall of 97.95%, surpassing the original YOLOv8. It showed significant improvements in small-target detection with an mAP@50 score of 98.17%. The model outperformed YOLOv5, YOLOv7, Faster R-CNN, and SSD, achieving an average accuracy 17.04% higher than Faster R-CNN. It is effective in detecting small tea pests in complex environments with a detection speed of 95 FPS.</p>
<p>Citrus Diseases and Pests Detection Model Based on Self-Attention YOLOV8</p>	<p>The study acknowledges several limitations. One key challenge is the difficulty in detecting small diseases and pests, which affects the accuracy of the model and may lead to missed detections. Furthermore, the density and size of the targets in citrus</p>	<p>The methodology involves developing the Light-SA YOLOV8 model for citrus disease and pest detection using a dataset of 3,202 images. The model incorporates a BRA self-attention module in the backbone to improve long-range dependency capture</p>	<p>The Light-SA YOLOV8 model achieved a detection accuracy of 92.6% and an mAP@0.5 of 92.5%. The model demonstrated a significant 2.8% increase in precision and a 20.7% reduction in computational workload compared to the original</p>

	<p>orchards present significant obstacles, impacting detection accuracy and localization. These issues, particularly the struggles with small target detection and dense environments, highlight the need for improvements to enhance the model's performance in such conditions. Addressing these limitations is crucial for optimizing the model's effectiveness in real-world applications.</p>	<p>and a Fast_C2f module for faster detection. The Asymptotic Feature Pyramid Network (AFPN) is used at the neck to fuse multi-scale features. The model is evaluated on metrics like precision, recall, and mAP@0.5, and compared with other models, achieving improved accuracy, speed, and efficiency.</p>	<p>YOLOV8n. When compared to other models like Faster RCNN, YOLOV3-tiny, YOLOV5n, and YOLOV8n, it showed superior performance in detection accuracy and speed. The model achieved a detection speed of 3.4 ms per image, making it highly efficient for real-time use in complex agricultural environments.</p>
<p>Comparative Analysis of YOLOv8 and YOLOv9 Models for Real-Time Plant Disease Detection in Hydroponics</p>	<p>The paper identifies that YOLOv9, while more accurate, requires more computational resources than YOLOv8, making it less suitable for portable devices. YOLOv8 is more efficient but slightly less accurate. Both models also struggle with certain diseases, like Black Rot, due to limited training data.</p>	<p>The study compares YOLOv8 and YOLOv9 for plant disease detection in hydroponics. A dataset of 3,676 images was used, split into training (80%) and testing (20%) sets. The models were trained using the Adam optimizer, with hyperparameters like a learning rate of 0.0001, a batch size of 16, and 150 epochs. YOLOv8 and YOLOv9 were evaluated on precision, recall, and mAP scores. YOLOv8 was then</p>	<p>YOLOv9 slightly outperformed YOLOv8 with a detection accuracy of 88.38% compared to 87.22% for YOLOv8. YOLOv8, however, required fewer computational resources and was faster, making it more suitable for real-time applications on portable devices. Both models demonstrated high precision and recall, with YOLOv9 achieving slightly better results in both metrics. YOLOv9's improved feature extraction and</p>

		converted to TensorFlow Lite for deployment on mobile devices, ensuring real-time detection capabilities.	processing capabilities led to marginally better performance, while YOLOv8 showed faster detection, making it ideal for mobile applications.
Enhanced Multiscale Plant Disease Detection with the PYOLO Model	The paper highlights that the integration of MHC2f and EC2f increases model complexity, requiring careful parameter tuning. Additionally, the limited dataset size hinders generalization, necessitating a broader range of plant diseases and conditions for better applicability. Further optimization of memory usage and computational efficiency is also needed.	The study proposes the PYOLO model for plant disease detection, enhancing YOLOv8n with EC2f, MHC2f, and BiFPN modules. The EC2f module improves feature learning, while MHC2f uses multihead self-attention to capture spatial details. BiFPN refines feature fusion across scales. The model is trained on a dataset of 10,000 images, with performance evaluated using precision, recall, F1 score, and mAP.	The PYOLO model outperforms YOLOv8n in plant disease detection, achieving a 6.7% improvement in precision, a 5.5% improvement in recall, and a 4.1% improvement in mAP. The model's enhancements, including the EC2f, MHC2f, and BiFPN mechanisms, significantly increase its ability to recognize plant diseases across various scales and in complex backgrounds. Additionally, PYOLO demonstrated superior performance when compared to traditional models like Faster-RCNN and SSD, making it a robust solution for real-time plant disease detection in agricultural applications.

<p>Design and Development of a Rose Plant Disease-Detection and Site-Specific Spraying System Based on a Combination of Infrared and Visible Images</p>	<p>The system's dependence on thermal and visual image registration could be influenced by environmental factors such as changes in light or temperature. Moreover, the choice of nozzle and the extent of spray coverage may result in inefficiencies, especially if the system's forward speed is not fine-tuned. The system effectively minimises pesticide usage; however, it necessitates additional refinement in controlling droplet size and enhancing spray accuracy for expansive areas.</p>	<p>The investigation established a methodology for diagnosing rose diseases and administering treatments through thermal and optical imaging techniques. A transportable C-shaped carriage holds both a thermal camera (ITI-P400) and a visible camera (Canon 550D). The system finds infections of powdery mildew and gray mold, looks at photos to find and register them, and uses a site-specific sprayer to hit the right spots. We used water-sensitive sheets and thermal histograms to look at the sprayer's effectiveness by looking at the size, coverage, and uniformity of the droplets.</p>	<p>The implemented system effectively detected diseases in rose plants utilizing both thermal and visible imagery. The sprayer demonstrated a coverage of 95.23% on upper leaves and 69.36% on lower leaves, highlighting its effectiveness in targeted application. The evaluation of the system's performance involved the use of thermal histograms and droplet size analysis, which confirmed the uniformity of spray coverage. The investigation further suggested enhancing the forward speed or employing atomizer nozzles to optimize droplet distribution for improved disease management. Thermal imaging application for spray evaluation demonstrated its capability to effectively monitor both spray coverage and efficiency.</p>
<p>Leveraging the WFD2020 Dataset for Multi-Class Detection of</p>	<p>Despite impressive results, the study faces issues with class imbalance, which could cause</p>	<p>The study employed two deep learning models, TensorFlow-based Faster R-CNN and</p>	<p>The YOLOv8 model achieved superior results with an mAP of 0.99, vastly outperforming the</p>

<p>Wheat Fungal Diseases with YOLOv8 and Faster R-CNN</p>	<p>overfitting, and difficulties in detecting multiple overlapping diseases. Moreover, the small size of the dataset and the lack of diversity in disease conditions impact model robustness, requiring more data and varied conditions for better generalization .</p>	<p>SSD, as well as PyTorch-based YOLOv8, for multi-class detection of wheat fungal diseases. The WFD2020 dataset was used, focusing on three diseases: yellow rust, leaf rust, and powdery mildew. The models were fine-tuned, with YOLOv8 achieving the highest performance (mAP of 0.99) for detecting and classifying these diseases. Performance was evaluated using precision, recall, mAP, and confidence scores .</p>	<p>TensorFlow models (Faster R-CNN and SSD) which had a maximum mAP of 0.68. YOLOv8 proved highly effective in identifying diseases in wheat crops, demonstrating particularly strong capabilities in detecting diseased areas with varying object sizes. The study highlighted challenges like class imbalance and potential model overfitting, but YOLOv8 emerged as a robust tool for automated, image-based disease detection in agriculture, particularly for early-stage fungal disease diagnosis.</p>
<p>Mobile Device-Based Detection System of Diseases and Pests in Rose Plants Using Deep Convolutional Neural Networks and Quantization</p>	<p>The study is limited by a small dataset with images collected under uncontrolled environmental conditions, which may affect model performance and generalizability. Despite data augmentation, the model's accuracy could not reach near 100%, particularly in predicting certain</p>	<p>The study developed a mobile-based detection system using deep learning models like DenseNet169 and MobileNetV2 for identifying diseases and pests in Rosa damascena Mill. plants. A dataset of 567 images was collected and augmented, then used to train several CNN models. The</p>	<p>The DenseNet169 and MobileNetV2 models achieved the best classification performance, with DenseNet169 achieving 93% accuracy and MobileNetV2 showing excellent performance with the lowest inference time (137 ms on a mobile device). The ensemble learning method improved</p>

	<p>diseases. Additionally, real-time performance on mobile devices, especially post-quantization, can result in slower inference times.</p>	<p>best-performing models were deployed in a mobile application, with model inference times and accuracies evaluated on both desktop and mobile environments. The DenseNet169 model achieved the highest accuracy at 93%, with ensemble learning further improving accuracy to 95.17%. The system's performance was also optimized for mobile use through quantization.</p>	<p>the accuracy to 95.17%. The developed mobile application allows real-time disease detection on smartphones with efficient resource usage. Post-quantization of the models reduced their file sizes significantly, making them suitable for mobile deployment, where MobileNetV2 performed with minimal delay and high accuracy.</p>
<p>Plant Disease Recognition Model Based on Improved YOLOv5</p>	<p>The main limitations of the improved YOLOv5 model include the model's reduced performance in recall for powdery mildew detection and precision for anthracnose detection. The dataset is also relatively limited to rubber tree diseases, which affects the model's ability to generalize to other plant diseases. Furthermore, despite improvements in detection accuracy, challenges remain in identifying obscured diseases and improving the</p>	<p>The study proposed an improved YOLOv5 model for rubber tree disease detection. It incorporated modifications such as replacing the Bottleneck module with the InvolutionBottleneck to reduce parameter count, adding an SE module for enhanced feature fusion, and using the EIOU loss function to accelerate convergence. A dataset of 2,375 images of powdery mildew and anthracnose was used for training and</p>	<p>The improved YOLOv5 model achieved a precision of 86.5% for powdery mildew and 86.8% for anthracnose detection, with a mean average precision (mAP) of 70%, a 5.4% improvement over the original YOLOv5. The enhanced model outperformed the original YOLOv5 and YOLOX_nano networks in disease detection accuracy, particularly in identifying powdery mildew and addressing the issue</p>

	recognition of certain disease features.	testing. The model was optimized with the Stochastic Gradient Descent algorithm, and performance was evaluated using precision, recall, and mAP metrics.	of detecting obscured diseased leaves. The results suggest that the model can be used effectively for real-time plant disease recognition in natural conditions.
Vase-Life Monitoring System for Cut Flowers Using Deep Learning and Multiple Cameras	The main limitations include challenges in detecting early-stage senescence symptoms and distinguishing subtle visual differences between similar factors (e.g., petal wounds and gray mold disease). These difficulties led to moderate accuracy in some object detection stages, such as flower opening (OS3) and GMD3. Further improvements in data quantity, object detection accuracy, and the use of high-resolution imaging techniques like microscopy and SEM are needed to enhance performance.	The study developed a vase-life monitoring system (VMS) integrating YOLOv8 for object detection, measuring physiological and disease factors in cut roses. The system assessed flower diameter, opening, gray mold disease, water uptake, and fresh weight using real-time camera imaging. A deep learning model predicted vase life (VL) based on these factors, with data collected from 50,000 annotated images. The model's performance was evaluated through precision, recall, and mAP, showing high accuracy but with some challenges in early-stage detection.	The VMS achieved high object detection accuracy (90%) using YOLOv8 and successfully measured key parameters like flower diameter, flower opening, and GMD. Regression analysis indicated a strong correlation between VL predictions from the VMS and those from traditional observation ($r^2 = 0.66$). The system was effective in early detection of physiological changes and disease symptoms, with potential applications in breeding and quality control of cut flowers. The addition of microscopy to the VMS improved detection accuracy and early-stage symptom identification.

Rose Plant Disease Detection Using Deep Learning	The study faces limitations due to the potential overfitting of VGG16, which improved accuracy during validation but underperformed in generalization. Additionally, despite high accuracy with MobileNetV2, the model still struggled with distinguishing subtle differences between similar leaf diseases.	The research employed a deep learning approach for rose plant disease detection using CNN models like VGG16, MobileNetV2, and Inception V3. A dataset of 1,443 rose leaf images, including four disease types, was used. Images were augmented through rotations and zooming, and preprocessed with RGB color space adjustments. The models were trained with a focus on accuracy, yielding 99.15% accuracy with MobileNetV2.	The MobileNetV2 model achieved the highest accuracy at 99.15%, outperforming VGG16 and InceptionV3, which achieved accuracies of 98.30%. The confusion matrix revealed high precision and recall values, particularly for the "Healthy" and "Leaf Blight" categories. MobileNetV2 demonstrated excellent performance across all metrics, with a precision of 99.28%, recall of 99.28%, and F1 score of 0.99 for detecting all rose plant diseases. The study suggests that this model can effectively aid in the detection of diseases in rose plants, benefiting farmers in Bangladesh.
Early Real-Time Detection Algorithm of Tomato Diseases and Pests in the Natural Environment	The main limitation is the challenge of detecting small objects and handling occlusion in the complex background of tomato diseases and pests. Despite improvements to the YOLOv3 model, the	The study proposed an enhanced YOLOv3 model for real-time tomato disease and pest detection. It used a dilated convolution layer for better small object detection and feature extraction,	The proposed method achieved an F1 score of 94.77% and an average precision (AP) of 91.81%, with a false detection rate of just 2.1%. The model was able to detect tomato diseases and

	<p>detection accuracy decreases under varying lighting and when objects are overlapping. Additionally, the dataset primarily focuses on early-stage diseases, which limits the model's application in later stages.</p>	<p>along with a non-maximum suppression (NMS) optimization to handle occlusion. The model was lightweighted using convolution factorization to balance performance and computation. The method achieved a high accuracy rate of 94.77% F1 score and 91.81% AP in real-world scenarios.</p>	<p>pests in real-time, even with occluded or overlapping objects. Compared to traditional methods (HOG + SVM) and other deep learning models (Faster-RCNN, SSD, YOLOv3), the improved YOLOv3 showed significantly better performance, particularly in detecting small and occluded objects. The detection time was only 55 ms, making it suitable for large-scale deployment in field environments using IoT-based video monitoring systems.</p>
<p>Vegetable Disease Detection Using an Improved YOLOv8 Algorithm in the Greenhouse Plant Environment</p>	<p>The study faces limitations due to challenges in detecting occluded or small disease objects, despite improvements in the model. While the inclusion of an Occlusion Perception Attention Module (OAM) enhances feature extraction for occluded diseases, computational costs are slightly increased. The model's performance may still decrease in environments with</p>	<p>The study introduces YOLOv8n-vegetable, a model designed for efficient vegetable disease detection. Key innovations include the integration of the GhostConv module, C2fGhost, and an OAM for improved feature extraction and reduced model size. The model also incorporates a small object detection layer and uses an HIoU loss function for better bounding box regression. The</p>	<p>The YOLOv8n-vegetable model achieved a 6.46% increase in mean average precision (mAP) compared to the original YOLOv8n. It also showed a reduction in parameter size and model size, making it more efficient. The model excelled in detecting small-sized disease objects and demonstrated improved detection speed, achieving 271.07 FPS.</p>

	low light or complex background interference.	model was trained on a self-built vegetable disease dataset and optimized for both speed and accuracy, achieving an mAP of 92.91% and a detection speed of 271.07 FPS.	YOLOv8n-vegetable outperformed other object detection models such as Faster R-CNN, YOLOv3-tiny, and YOLOv5s, both in terms of accuracy and computational efficiency. The introduction of OAM and the small-sized object detection layer significantly enhanced the model's ability to detect occluded and small vegetable diseases in challenging greenhouse environments.
Revolutionizing Rose Grading: Real-Time Detection and Accurate Assessment with YOLOv8 and Deep Learning Models	The method struggles with detecting roses that are blurred or blend with complex backgrounds, which can impact accuracy in real-world conditions. Overlapping roses and variations in camera angles also pose challenges to detection and classification.	The proposed approach utilizes YOLOv8 for real-time rose detection and localization, followed by cropping the roses for classification. Two deep learning models, a lightweight CNN and a fine-tuned MobileNet, were used for classifying roses based on growth stages (small, matured, faded). The method was trained on a dataset of 3750 images and	YOLOv8 achieved high detection accuracy for roses in various growth stages, with mAP values of 0.94 for small roses, 0.98 for mature roses, and 0.94 for faded roses. The proposed CNN model outperformed other models in terms of testing accuracy, achieving 90%, while the fine-tuned MobileNet achieved 93% testing accuracy. The method demonstrated effective real-time

		evaluated with various deep learning models, demonstrating superior accuracy and real-time performance.	detection, with YOLOv8 handling complex scenarios like overlapping roses and varying lighting conditions. This approach outperformed other deep learning models like ResNet50 and VGG19, and the model's lightweight design ensures real-time application potential for rose grading and yield prediction.
SerpensGate-YOLOv8: An Enhanced YOLOv8 Model for Accurate Plant Disease Detection	Despite its improvements, the SerpensGate-YOLO v8 model faces challenges in handling small, densely clustered plant diseases, which can result in missed detections and false positives. The similarity in appearance between certain diseases, such as tomato yellow leaf curl virus and tomato leaf spot, leads to misclassification. Additionally, environmental factors like lighting changes and leaf occlusion impact the model's performance, requiring further optimization.	The study introduces SerpensGate-YOLO v8, an enhanced YOLOv8 model designed for plant disease detection. Key improvements include the integration of Dynamic Snake Convolution (DySnakeConv), Super Token Attention (STA), and SPPELAN modules to improve feature extraction and aggregation. The model, trained on the PlantDoc dataset, achieved a 3.3% increase in mAP@0.5 compared to the original YOLOv8, demonstrating improved detection	The SerpensGate-YOLO v8 model showed a 3.3% improvement in mean Average Precision (mAP@0.5) compared to the original YOLOv8, achieving an mAP of 64.9%. The enhancements made the model more effective in detecting plant diseases in complex environments. The introduction of DySnakeConv and STA significantly improved detection accuracy, especially for fine and irregular disease features. The model achieved a precision of 0.719, and the experimental

		accuracy and robustness in complex agricultural environments.	results demonstrated its robustness and efficiency in real-world agricultural settings. However, challenges like misclassification due to similarity in symptoms and environmental conditions remain, suggesting areas for further optimization.
--	--	---	--

2.3 Summary

This could have many uses in agriculture. Things like changing environments, similar diseases, and poor image quality still affect how accurate detection is. We need to do more research to make these methods even better and make sure they work in real-world farming situations. The growing need for quick and accurate plant disease detection has brought out the flaws in old methods, which are often time-consuming and depend on human skill. Deep learning has made disease detection much faster and more accurate in recent years.

CHAPTER 3

METHODOLOGY

3.1 Overview

Employing the YOLOv8 deep learning model, we accurately identify and classify diseases affecting rose plants. A curated dataset of rose tree images was used to train the detection model, with each image clearly labeled as “YOLOv8 n, s, m, l, x.” These methods enhance information quality by increasing the number of examples. We rigorously tested each model's operational performance, accuracy, and ease of use in real-world environments. Performance can be significantly improved through hyperparameter tuning, transfer learning, and compression techniques. The current model outperforms previous versions in both prediction accuracy and computational speed. The core objective remains the precise identification of rose plant diseases and the provision of accurate diagnoses.

3.2 Detailed Methodology

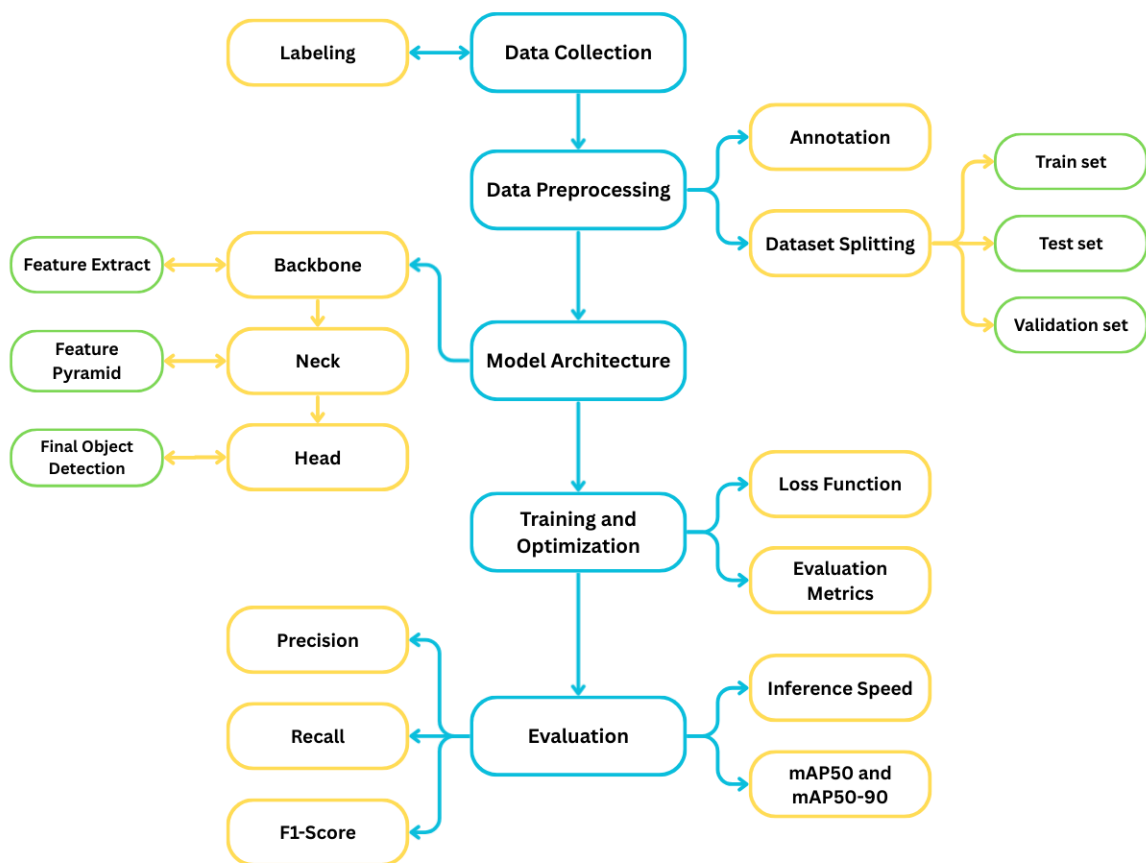


Figure 3.2 : Methodology

Dataset Overview:

For this study, 553 photos were carefully taken to show how healthy the rose trees really were. There are pictures here. Downy mildew, black spot disease, and healthy new leaves are all shown in the pictures. It was possible to get examples from real field data with a range of backgrounds, lighting, and leaf positions. We employ a rigorous annotation process to ensure each image accurately reflects its classification attributes. This enables the proper training and testing of deep learning models. There are only two types of data in the dataset: healthy leaves (fresh leaves) and diseased leaves (downy mildew and black spot). This is useful for tasks that require sorting things into two or more groups. This dataset is great for training models because it has a lot of different leaf shapes and diseases that rose plants can get.

Dataset Description:

Table 3.2.1 : Dataset Overview

Name	Description
Total Image Count	553
Average Dimension	640 x 640
Format of Data	JPG
Fresh Leaf	41
Black Spot	311
Downy Mildew	201

Sample Dataset:

[1]



[2]



[3]



[4]



[5]

Figures 3.2.2 : Fresh Leaf[1], Downy Mildew[2,3], Black Spot[4,5]

Image Pre-processing:

Roboflow, a well-known tool for computer vision tasks, to set up a structured image pre-processing pipeline. This made sure the dataset was ready for training deep learning models. Roboflow blows up the process of labeling images, allowing for very precise hand labeling of each one. This step of annotating was very important for teaching the model how to find and sort the three classes of interest: Fresh Leaf, Black Spot, and Downy Mildew. In addition to annotation, a few other important pre-processing techniques were used to make the model work better and make the input images less variable. First, the images were automatically oriented, which fixed any problems with rotation that happened while the data was being collected. This step made sure that all the images were facing the same way. The size of each picture was also increased to 640 pixels by 640 pixels. Most deep learning systems, including YOLOv8, need all of their inputs to be the same size. When the leaves were put in a square frame, they stayed the same shape even though they were cut down. It was fast and exactly what was needed. Following these steps, a strong and clean collection was made that could be used. This made it possible to teach models how to see things clearly.

Model Architecture:

The YOLOv8 model is designed for speed and efficiency in real-time object detection. Its multi-level feature fusion through concatenation and upsampling enables it to handle complex visual environments with varying object sizes, while the backbone and neck work in synergy to extract, refine, and combine features to make accurate predictions.

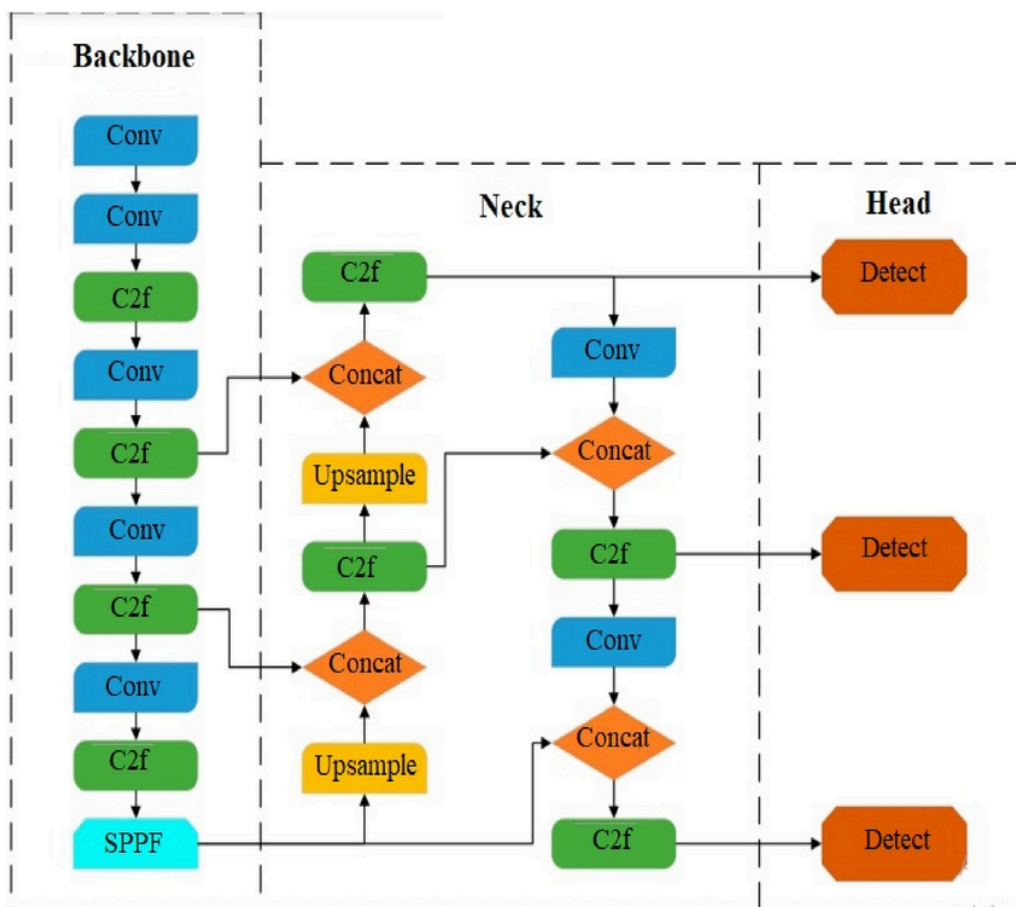


Figure 3.2.3: YOLOv8 Model Architecture

The YOLOv8 model architecture shown in the diagram is designed to provide an efficient and powerful object detection framework, leveraging various components that optimize the model's accuracy and performance. It consists of three primary components: the Backbone, Neck, and Head.

Backbone: The backbone of the YOLOv8 model primarily consists of a series of sequential operations involving convolutional layers (Conv). CSPDarknet53, a variant of Darknet frequently employed in YOLO models, serves as the backbone. The convolutional layers progressively extract hierarchical features from the input image. The application of advanced convolutions enables the utilization of C2f (C2f block) to enhance feature extraction and the representation of complex patterns. The backbone functions as the critical feature extraction element, vital for preparing the image data for the following, more specialized processes within the model.

Neck: The neck is a crucial part of the architecture as it helps to generate feature pyramids by integrating multiple levels of features from the backbone. It facilitates multi-scale feature fusion and enhances the model's ability to detect objects at various scales. The Concat (concatenation) blocks are used here to merge different feature maps from the backbone to enhance the overall feature representation. Additionally, Upsample blocks are employed to upsample low-resolution features to match the higher resolution features, allowing the model to detect both small and large objects effectively. The C2f blocks continue to play a vital role in refining the feature maps by performing additional convolutions.

Head: The head of the model is where the final object detection occurs. The features from the neck are passed through another set of convolution operations, and once again, Concat operations combine features at different levels. The ultimate goal here is to detect the presence and location of objects in the image. The Detect block represents the final detection layer, which produces the bounding boxes, confidence scores, and class predictions for each object detected.

SPPF (Spatial Pyramid Pooling Fast): An interesting component in the architecture is SPPF in the backbone, which is responsible for enhancing the model's spatial attention mechanism. The SPPF layer enables the model to capture both fine-grained details and global context information, making the model more robust in detecting objects at various scales.

3.3 Summary

The thesis methodology describes how deep learning is used to detect rose plant diseases. The YOLOv8 model is used for efficient object detection. Dataset preparation involves collecting and annotating pictures of healthy and ill rose plants. YOLOv8, which uses a modified CSPDarknet53 backbone, learns essential properties from the dataset. Model performance is measured by accuracy, precision, recall, and F1 score. Finally, real-world photos are used to test the trained model's disease identification capabilities.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Environment Setup

This study primarily employs Python 3 as the programming language and utilizes T4 GPU acceleration to enhance operational efficiency. All operations were conducted within a macOS environment. Google Colab was selected as the cloud platform for deep learning models due to its exceptional GPU acceleration capabilities, which are critical for both training and inference tasks. One of the chosen deep learning models is YOLOv8. As is widely recognized, this model demonstrates outstanding performance in object detection. This design approach best suits the research requirements, enabling rapid and precise processing of tasks demanding immediate completion. YOLOv8 demonstrates strong compatibility with numerous other deep learning systems. This tool easily adapts to various hardware devices and is simple to operate, running entirely within the Google Colab environment. This dataset contains photographs of rose bushes, each image illustrating the plant's health status. We utilized Roboflow for data classification and annotation, significantly accelerating processing speed. A portion of this dataset serves as the training set, comprising 70% of the data. The other two parts are the validation set and the test set. This partitioning allows us to evaluate the model's performance. This study demonstrates that such an environment facilitates model training and testing, enabling more efficient identification of diseases affecting rose plants.

4.2 Testing and Evaluation

YOLOv8n:

```

30 epochs completed in 0.070 hours.
Optimizer stripped from runs/detect/train/weights/last.pt, 6.2MB
Optimizer stripped from runs/detect/train/weights/best.pt, 6.2MB

Validating runs/detect/train/weights/best.pt...
Ultralytics 8.3.128 Python-3.11.12 torch-2.6.0+cu124 CUDA:0 (Tesla T4, 15095MiB)
Model summary (fused): 72 layers, 3,006,233 parameters, 0 gradients, 8.1 GFLOPs

```

Class	Images	Instances	Box(P)	R	mAP50	mAP50-95)
all	111	187	0.952	0.909	0.983	0.871
Black Spot	62	137	1	0.752	0.965	0.647
Downy Mildew	41	41	0.99	0.976	0.99	0.971
Fresh Leaf	9	9	0.867	1	0.995	0.995

```

Speed: 0.3ms preprocess, 16.7ms inference, 0.0ms loss, 4.1ms postprocess per image
Results saved to runs/detect/train

```

Figure 4.2.1 : YOLOv8n model train

Figure 4.2.1 demonstrates that the YOLOv8n (nano) model achieves outstanding performance after 30 training iterations on a specialized dataset for rose plant disease detection. Training this model on a Tesla T4 GPU consumed a total of 3,006,233 training samples. The “All” category achieved a score of 0.983 in the mAP50 test and 0.871 in the mAP50-95 test, demonstrating the detection tool's exceptional recognition capability. Performance in the black spot disease category was particularly outstanding, with an accuracy of 1.0 and an mAP50 of 0.965. The downy mildew detection group achieved an outstanding mAP50-95 score of 0.971. Processing an image takes 16.7 milliseconds, with post-processing accounting for 4.1 milliseconds and pre-processing taking 0.3 milliseconds. When rapid and accurate task completion is required, the YOLOv8n model is the optimal choice. For instance, it can swiftly and precisely detect diseases on rose plants.

YOLOv8s :

```

30 epochs completed in 0.080 hours.
Optimizer stripped from runs/detect/train2/weights/last.pt, 22.5MB
Optimizer stripped from runs/detect/train2/weights/best.pt, 22.5MB

Validating runs/detect/train2/weights/best.pt...
Ultralytics 8.3.128 🚀 Python-3.11.12 torch-2.6.0+cu124 CUDA:0 (Tesla T4, 15095MiB)
Model summary (fused): 72 layers, 11,126,745 parameters, 0 gradients, 28.4 GFLOPs

```

Class	Images	Instances	Box(P	R	mAP50	mAP50-95)
all	111	187	0.981	0.948	0.985	0.871
Black Spot	62	137	0.975	0.868	0.968	0.653
Downy Mildew	41	41	0.993	0.976	0.991	0.965
Fresh Leaf	9	9	0.976	1	0.995	0.995

```

Speed: 0.2ms preprocess, 22.1ms inference, 0.0ms loss, 3.9ms postprocess per image
Results saved to runs/detect/train2

```

Figure 4.2.2 : YOLOv8s model train

As you can see in Figure 4.2.2, the small YOLOv8s model did very well after being trained for 30 times on a dataset that was only used to find diseases in rose plants. The model was learned on a Tesla T4 GPU and has 11,126,745 parameters. There are many types of illnesses that the program can find, such as black spot disease, downy mildew, and new leaf disease. The "All" group has a mAP50 of 0.985 and a mAP50-95 of 0.871. A score of 0.965 on the mAP50-95 scale means that downy mildew has been found. This process takes 22.1 ms for every frame. Before processing, it takes 0.2 milliseconds, and after processing, it takes 3.9 milliseconds. The study results show that the YOLOv8s model is better at finding plant diseases because it is both more accurate and faster.

YOLOv8m :

```

30 epochs completed in 0.128 hours.
Optimizer stripped from runs/detect/train3/weights/last.pt, 52.0MB
Optimizer stripped from runs/detect/train3/weights/best.pt, 52.0MB

Validating runs/detect/train3/weights/best.pt...
Ultralytics 8.3.128 🚀 Python-3.11.12 torch-2.6.0+cu124 CUDA:0 (Tesla T4, 15095MiB)
Model summary (fused): 92 layers, 25,841,497 parameters, 0 gradients, 78.7 GFLOPs

```

Class	Images	Instances	Box(P)	R	mAP50	mAP50-95)
all	111	187	0.972	0.946	0.984	0.869
Black Spot	62	137	0.944	0.863	0.968	0.645
Downy Mildew	41	41	0.991	0.976	0.988	0.965
Fresh Leaf	9	9	0.982	1	0.995	0.995

```

Speed: 0.3ms preprocess, 33.2ms inference, 0.0ms loss, 4.6ms postprocess per image
Results saved to runs/detect/train3

```

Figure 4.2.3 : YOLOv8m model train

In Figure 4.2.3, you can see how well the YOLOv8m (middle) model did on a set of data used to find diseases in rose plants. We trained for 30 rounds to get this. A Tesla T4 GPU was used to train the model, which has 25,841,497 parts. It demonstrated outstanding performance in tests targeting three major diseases: downy mildew, black spot, and new leaf disease. Across all disease assessments, its mAP50 score reached 0.984, while the mAP50-95 score achieved 0.869. Particularly noteworthy is its performance in downy mildew detection, achieving an impressive test score of 0.965 on the mAP50-95 scale. From start to finish, the processing times were 0.3 milliseconds, 33.2 milliseconds, and 4.6 milliseconds, respectively. This validation demonstrates that the YOLOv8m model performs well and is capable of disease detection. It's a good way to find rose diseases because of this.

YOLOv8l :

```

30 epochs completed in 0.221 hours.
Optimizer stripped from runs/detect/train4/weights/last.pt, 87.6MB
Optimizer stripped from runs/detect/train4/weights/best.pt, 87.6MB

Validating runs/detect/train4/weights/best.pt...
Ultralytics 8.3.128 🚀 Python-3.11.12 torch-2.6.0+cu124 CUDA:0 (Tesla T4, 15095MiB)
Model summary (fused): 112 layers, 43,608,921 parameters, 0 gradients, 164.8 GFLOPs

```

Class	Images	Instances	Box(P)	R	mAP50	mAP50-95)
all	111	187	0.963	0.947	0.98	0.87
Black Spot	62	137	0.992	0.866	0.968	0.651
Downy Mildew	41	41	0.992	0.976	0.978	0.966
Fresh Leaf	9	9	0.906	1	0.995	0.995

```

Speed: 0.2ms preprocess, 38.2ms inference, 0.0ms loss, 2.4ms postprocess per image
Results saved to runs/detect/train4

```

Figure 4.2.4 : YOLOv8l model train

The review of the YOLOv8l (large) model is shown in Figure 4.2.4. It was trained on a set of data that was used to find diseases in rose plants 30 times. It was trained on a Tesla T4 GPU and has 43,608,921 features. The performance tests show that the spotting is very good: the mAP50 for "All" was 0.98, and the mAP50-95 was 0.87. With a mAP50 of 0.968 and a bounding box accuracy of 0.992, black spot disease was very good. With a mAP50-95 of 0.966, it was very easy to find downy mildew. The Fresh Leaf group was almost perfect at finding things, with a mAP50 of 0.995 and a mAP50-95 of 0.995. For each frame, it took 38.2 ms to figure out what the model meant. Post-processing took 2.4 milliseconds, while the preparation phase took 0.2 milliseconds. That the YOLOv8 model is very good at finding plant diseases is clear from this work.

YOLOv8x :

```

30 epochs completed in 0.395 hours.
Optimizer stripped from runs/detect/train5/weights/last.pt, 136.7MB
Optimizer stripped from runs/detect/train5/weights/best.pt, 136.7MB

Validating runs/detect/train5/weights/best.pt...
Ultralytics 8.3.128 Python-3.11.12 torch-2.6.0+cu124 CUDA:0 (Tesla T4, 15095MiB)
Model summary (fused): 112 layers, 68,126,457 parameters, 0 gradients, 257.4 GFLOPs

```

Class	Images	Instances	Box(P	R	mAP50	mAP50-95)
all	111	187	0.948	0.943	0.978	0.874
Black Spot	62	137	0.972	0.854	0.959	0.658
Downy Mildew	41	41	0.985	0.976	0.98	0.968
Fresh Leaf	9	9	0.887	1	0.995	0.995

```

Speed: 0.2ms preprocess, 62.5ms inference, 0.0ms loss, 2.9ms postprocess per image
Results saved to runs/detect/train5

```

Figure 4.2.5 : YOLOv8x model train

Figure 4.2.5 presents the evaluation results of a YOLOv8x (extra-large) model that has undergone training for 30 epochs on a custom dataset aimed at detecting diseases in rose plants. The model comprises 68,126,457 parameters and was trained utilizing a Tesla T4 GPU. The "all" class recorded a mAP50 score of 0.978 and a mAP50-95 score of 0.874. The Black Spot class demonstrated strong performance, achieving a Box score of 0.972 and a mAP50 of 0.959. The Downy Mildew class demonstrated outstanding performance, achieving a mAP50-95 score of 0.968. The Fresh Leaf class attained a mAP50 score of 0.995 and a mAP50-95 of 0.995, demonstrating nearly flawless detection capabilities. The inference time for the model is 62.5 ms per image, including a preprocessing duration of 0.2 ms and a post-processing duration of 2.9 ms. The findings indicate that the YOLOv8x model achieves impressive accuracy and efficiency, positioning it as an excellent choice for tasks related to disease detection.

4.3 Result and Discussion

Result:

Table 4.3.1 : Comparison table of YOLOv8 all version

YOLOv8 Version	Precision	Recall	F1-Score	mAP50	mAP50-90	Inference Speed
nano	0.8748	0.8973	0.8860	0.9800	0.8606	7.2ms per image
small	0.9482	0.9602	0.9542	0.9795	0.8686	9.7ms per image
medium	0.9790	0.9456	0.9618	0.9819	0.8656	21.2ms per image
large	0.9700	0.9335	0.9512	0.9790	0.8714	37.5ms per image
extra-large	0.9780	0.8970	0.9368	0.9726	0.8625	64.3ms per image

The medium version of YOLOv8 stands out as the best among the listed versions due to its optimal balance of performance metrics. It achieves a high precision of 0.9790, which is nearly as high as the extra-large version (0.9800), while offering a better recall of 0.9456 compared to the extra-large (0.8970) and large versions (0.9335). This results in a strong F1-score of 0.9618, demonstrating its ability to maintain both high detection accuracy and sensitivity. The mAP50 score of 0.9819 and mAP50-90 score of 0.8656 further confirm its robustness in detecting objects across various thresholds, outperforming the large version. The medium version's inference speed (21.2ms per picture) is slower than the nano and small versions' (7ms and 9.7ms, respectively), but it can still compete because it is more accurate. The medium version is the best overall because it has high precision, recall, and mAP values. This makes it the best choice when accuracy is more important than speed and when a good balance between the two is needed.

Discussion:

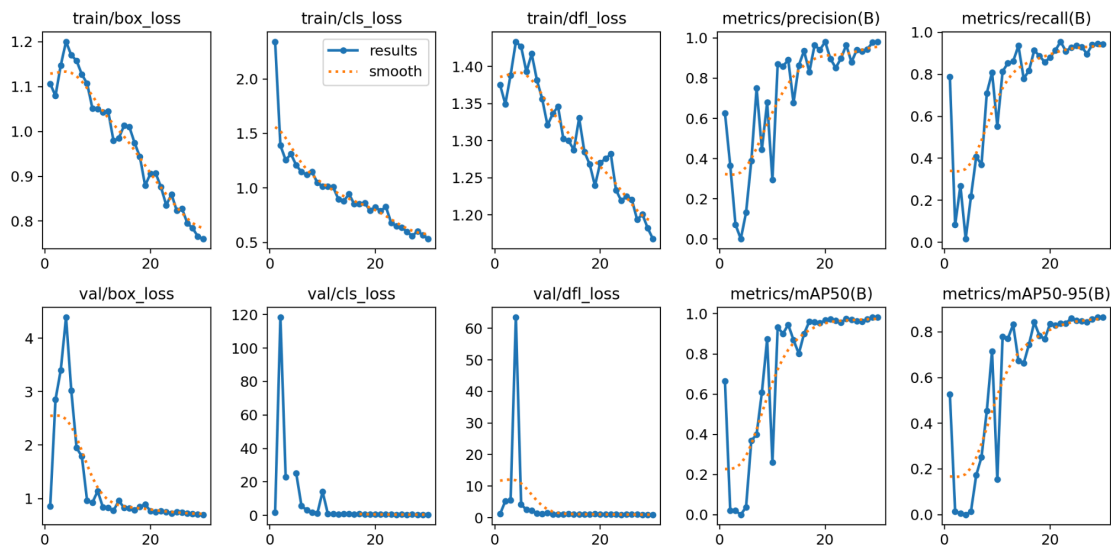


Figure 4.3.2: YOLOv8m result

This picture and a few key steps make it look like the model will be a great way to find rose diseases early on. As it learns more, it gets better at finding sick plants. This is clear because box loss keeps going down. Farmers need to know how to use bug killers the right way so they don't hurt plants and only treat the areas that need it. The loss in the rating box went down quickly after going up for a short time. This showed that the model could handle new information without any problems. But there were problems with proof over and over again. Category loss and DFL loss represent the most significant changes. This means that the model might not work well in some situations or when the environment changes, which is something that wasn't well modeled in the training data.

Accuracy and recall are two areas where this model really shines, and both of these measures keep getting better over time. The model is very accurate, which means it can tell the difference between healthy and sick plants. It is also very good at finding infected plants, which means it is less likely to miss illnesses. These traits are important for early detection because they help you find diseases quickly and correctly, which is a big part of stopping plant diseases from spreading in farms. Also, the average accuracy scores (mAP50 and mAP50-95) have gotten a lot better. This shows that the model is still accurate even when the objects overlap in different ways. This happens a lot when you're trying to find plant diseases that have vague signs or strange start times.

The plan works well overall, but there are a few things that could be done better. For example, there are big differences in the validity set. To make the model more accurate and reliable, the constant changes in validation loss values need to be stopped. This will also help keep the model stable in a wide range of environmental situations. The YOLOv8m model does a great job of finding diseases in rose plants in their early stages, according to tests. This technology is very useful for growers because it lets them find disease problems quickly, accurately, and on-site. This lets them take specific control measures before the disease spreads.

4.4 Limitation

Even though the study wasn't perfect, the YOLOv8 model might help plant doctors figure out what's wrong with roses. The collection has many things, but it might not be able to hold all of them. In some places or when they are sick, this means the plan might not work as well. When the light changes, the leaves may not face the same way, and other outside factors may not be as reliable or exact. It can also change how well it works based on the pictures on it. There are different kinds of pictures that can be used. You might have a bias if you only learn from good data that has been named. Though it works fine now, it might be faster to help with real-time tasks on big farms where things change quickly.

4.5 Future work

We hope that more research in the future will help us find solutions to these problems. This research will look at more types of rose diseases and different types of weather. Now that this is done, it will work better in more situations. We could make a lot more useful models if we looked into transfer learning and used more advanced ways to add to the data we already have. That way, the model can be used in real time when farming on a big scale. I will work on making its reasoning faster. More study needs to be done on making tools that help farmers quickly find diseases in their fields. With this method, diseases can be found quickly by using pictures on phones and other movable devices. It does this by putting the YOLOv8 model into an app that is easy to use. The goal is to give farmers this tool so they can quickly stop diseases from spreading. This will help them keep more of their crops and use chemicals that are less dangerous. It could be connected to farm management tools so that it could keep an eye on more crops and make disease control more automatic.

CONCLUSION

Scientists used the YOLOv8 deep learning model to look at pictures right now and find rose plant diseases like black spot and downy mildew. Images of rose leaves that were both healthy and sick were used in this research. Therefore, YOLOv8 can quickly and correctly identify these illnesses. Big and small YOLOv8 models all found things and came to the same conclusions at the same time. You can use a certain kind of YOLOv8 anywhere, which makes it stand out. You can count on it to remember things and do them right. It works pretty well most of the time (mAP). When you need to be wrong right away, that's why it works so well in real life. When the weather changes, it may work better with smaller amounts. After finding the problems, more study is needed. One way to do this is to include more outside events and diseases that affect roses. Most likely, the most important things that need to be done in the future to make models more useful in the real world are transfer learning and data reinforcement. Additionally, this will be improved to speed things up even more in large places. The YOLOv8-based method for finding diseases on rose plants looks like a quick and easy way to uncover the issue. There was a way for them to keep track of how much food they were growing, how healthy their crops were, and how many chemicals they used. These things would help farming, which is good for the earth.

REFERENCES

1. Sood, Shivani & Singh, Harjeet & Bhatia, Surbhi & Almusharraf, Ahlam. (2025). Leveraging the WFD2020 Dataset for Multi-Class Detection of Wheat Fungal Diseases with YOLOv8 and Faster R-CNN. *Computers, Materials & Continua*. 84. 2751-2787. 10.32604/cmc.2025.060185.
2. Tasnim, Afia & Das, Baivab & Islam, Md. Rajibul & Amiruzzaman, Md & Islam, Dr. MD Rashedul & Ahmed, Nadeem. (2024). Revolutionizing Rose Grading: Real-Time Detection and Accurate Assessment with YOLOv8 and Deep Learning Models. *SN Computer Science*. 6. 1-11. 10.1007/s42979-024-03556-z.
3. Ham, J.Y.; Kim, Y.-T.; Ha, S.T.T.; In, B.-C. Vase-Life Monitoring System for Cut Flowers Using Deep Learning and Multiple Cameras. *Plants* **2025**, *14*, 1076. <https://doi.org/10.3390/plants14071076>
4. Ye, R.; Gao, Q.; Qian, Y.; Sun, J.; Li, T. Improved YOLOv8 and SAHI Model for the Collaborative Detection of Small Targets at the Micro Scale: A Case Study of Pest Detection in Tea. *Agronomy* **2024**, *14*, 1034. <https://doi.org/10.3390/agronomy14051034>
5. D. Luo, Y. Xue, X. Deng, B. Yang, H. Chen and Z. Mo, "Citrus Diseases and Pests Detection Model Based on Self-Attention YOLOV8," in *IEEE Access*, vol. 11, pp. 139872-139881, 2023, doi: 10.1109/ACCESS.2023.3340148. keywords: {Diseases;Feature extraction;Computational modeling;Routing;Resource management;Load modeling;Crops;Agriculture;YOLO;Citrus;disease and pest detection;target detection;YOLOV8;lightweight},
6. Wang, X., Liu, J. Vegetable disease detection using an improved YOLOv8 algorithm in the greenhouse plant environment. *Sci Rep* 14, 4261 (2024). <https://doi.org/10.1038/s41598-024-54540-9>
7. Wang, X., Liu, J. & Zhu, X. Early real-time detection algorithm of tomato diseases and pests in the natural environment. *Plant Methods* 17, 43 (2021). <https://doi.org/10.1186/s13007-021-00745-2>
8. Ali, Usman & Ismail, Maizatul & Ariyaluran Habeeb, Riyaz Ahamed & Shah, Syed Roshan Ali. (2024). Performance Evaluation of YOLO Models in Plant Disease Detection. *Journal of Informatics and Web Engineering*. 3. 199-211. 10.33093/jiwe.2024.3.2.15.
9. Miao, Yongzheng & Meng, Wei & Zhou, Xiaoyu. (2025). SerpensGate-YOLOv8: an enhanced YOLOv8 model for accurate plant disease detection. *Frontiers in Plant Science*. 15. 10.3389/fpls.2024.1514832.
10. Tripathi, Abhishek & Gohokar, Vinaya & Kute, Rupali. (2024). Comparative Analysis of YOLOv8 and YOLOv9 Models for Real-Time Plant Disease Detection in Hydroponics. *Engineering, Technology & Applied Science Research*. 14. 17269-17275. 10.48084/etasr.8301.
11. Chen, Zhaoyi & Wu, Ruhui & Lin, Yiyang & Li, Chuyu & Chen, Siyu & Yuan, Zhineng & Chen, Shiwei & Zou, Xiangjun. (2022). Plant Disease Recognition Model Based on Improved YOLOv5. *Agronomy*. 12. 365. 10.3390/agronomy12020365.

12. Wang, Yirong & Wang, Yuhao & Mu, Jiong & Mustafa, Ghulam Raza & Wu, Qianqian & Wang, Ying & Zhao, Bi & Zhao, Siyue. (2025). Enhanced multiscale plant disease detection with the PYOLO model innovations. *Scientific Reports*. 15. 10.1038/s41598-025-89034-9.
13. Alvy, Md. Ali & Khan, Golam & Alam, Mohammad & Islam, Saiful & Rahman, Mokhlesur & Rahman, Mirza. (2023). Rose Plant Disease Detection using Deep Learning. 1244-1249. 10.1109/ICOEI56765.2023.10126031.
14. Duman, Burhan. (2025). Mobile Device-Based Detection System of Diseases and Pests in Rose Plants Using Deep Convolutional Neural Networks and Quantization. *Tarım Bilimleri Dergisi*. 31. 302-318. 10.15832/ankutbd.1514972.
15. Minaee, Saeed & Jafari, Mehrnoosh & Safaie, Naser. (2018). Design and development of a rose plant disease-detection and site-specific spraying system based on a combination of infrared and visible images. *Journal of Agricultural Science and Technology*. 20. 23-36.