Journal of Informatics and Web Engineering

Vol. 3 No. 2 (June 2024) eISSN: 2821-370X

Performance Evaluation of YOLO Models in Plant Disease Detection

Usman Ali[1], Maizatul Akmar Ismail[1*], Riyaz Ahamed Ariyaluran Habeeb[1], Syed Roshaan Ali Shah[2]

[1]Universiti Malaya, Malaysia

[2]University College London, UK

**Corresponding authors: (maizatul@um.edu.my, ORCiD: 0000-0003-1877-7128)*

Abstract - Plant diseases significantly impact global agriculture, leading to substantial production losses and economic consequences. Timely disease detection can enhance crop yield, optimize resource utilization, reduce costs, and mitigate environmental effects, ultimately ensuring high-quality food production. Deep learning, specifically computer vision-based techniques, have proven invaluable in tasks like image classification, segmentation, and object detection. Deep Learning techniques such as You Only Look Once (YOLO) models are state of the art neural network algorithms used for accurate object detection. In this study, YOLOv5, YOLOv7 and YOLOv8 models were trained on CCL'20 dataset for citrus disease detection. Data augmentation techniques such as image translation, image scaling, flip, mosaic augmentations were implemented to improve the models' performance during training phase. The model performance was evaluated using metric such as Mean Average Precision at 50% to 95% Intersection over Union score i.e. mAP@50-95. The results show that YOLOv8 model performs better than other variants and offers significant improvements over the benchmark performance from previous studies. The final hyper-parameter tuned model achieved 96.1% mAP@50-95 on testing data for citrus disease detection and mAP@50-95 of 95.3%, 96.0% and 97.0% for detection of Anthracnose, Melanose and Bacterial Brown Spot diseases, respectively. The trained model was able to detect single and multiple instances of same or different disease in an image showing the potential of recent YOLO models. The trained YOLOv8 model is deployed on Roboflow platform.

Keywords— Plant Disease Detection, Deep Learning, YOLOv5, YOLOv7, YOLOv8

Received: 20 March 2024; Accepted: 05 May 2024; Published: 16 June 2024

I. INTRODUCTION

Agriculture plays a pivotal role in bolstering a nation's economic growth, predominantly by increasing crop production. The early identification of plant diseases assumes an even more critical role in preserving global health [1]. Early detection facilitates timely intervention, mitigating the adverse effects that can be financially burdensome for farmers, entailing crop losses, increased pesticide consumption, and reduced yields. Beyond its significance in providing sustenance for human consumption, agriculture also serves as a primary source of livestock feed and raw materials for the food processing industry. Plants represent the quintessential energy source, offering both nutrition and therapeutic benefits. In the context of agriculture, the detection of plant diseases assumes paramount importance due to the potential for diseases to strike at any stage during crop cultivation, resulting in substantial losses in both crop productivity and financial value [2].

Plant diseases cause 10% to 16% global agriculture production losses which cost the global economy an estimated amount of $220 billion [3]. Food production must be increased up to 70 percent until 2050 to feed the entire world population [4]. The burden of crop production losses weighs most heavily on African countries, with estimates ranging

Journal of Informatics and Web Engineering

https://doi.org/10.33093/jiwe.2024.3.2.15

Published by MMU Press. URL: https://journals.mmupress.com/jiwe

from 20% to 40% [5]. Plant diseases, climate change and scarcity of land for agriculture are the main factors leading towards food security issues which has become a global concern.

Precision Agriculture, with its utilization of technology, data, and automation, strives to enhance crop yield production, optimize resource usage, minimize costs, and reduce environmental impacts. Furthermore, Precision Agriculture empowers farmers to conserve precious resources such as water, fertilizers, and pesticides by pinpointing specific field areas requiring attention at the right time. Plant disease detection at early and further growing stages can help the farmers to identify the infected fields and support efficient utilization of resources. Moreover, the integration of plant disease detection with Precision Agriculture Technologies (PATs) facilitates the use of Variable Rate Technology (VRT) among growers, enabling targeted resource allocation and maximizing crop yield production.

The traditional method of plant disease detection through physical field surveys is time-consuming and labor-intensive, rendering it nearly impossible for humans to comprehensively assess large field areas. The considerable diversity in leaf attributes, including shape, size, color, position, and lighting conditions, compounds the complexity of providing an effective and efficient automated solution. Conventional machine learning methods have fallen short in disease detection effectiveness [2]. Consequently, there is an exigent need to employ the latest and most effective deep learning techniques for disease detection, ideally capable of identifying multiple instances of similar or distinct diseases within a single image [1]. To address these challenges, modern deep learning (DL) techniques are introduced for plant disease detection, promising superior performance. This study is focused on performance evaluation of YOLO models including YOLOv5, YOLOv7 and YOLOv8, for plant disease detection accuracy. A high-quality dataset for plant disease detection and a benchmark study result for the same dataset will be used to compare the model's performance. The current study stressed on YOLO models i.e., YOLOv5, YOLOv7 and YOLOv8 to detect and locate multiple instances of the same or different diseases in one image, and to also work on different variations of leaf shape, size, color, position, and lighting conditions.

The study is organized into sections as follows: related work section addresses YOLO evolution and literature review, materials and method section describe about dataset, data preprocessing and modeling, results and discussion section is about models' training, hyperparameter fine tuning, models' comparison, benchmark study comparison and model deployment. Future work is also discussed in conclusion section.

II. LITERATURE REVIEW

Many studies have been conducted for early plant diseases and pests' detection to control the spread of disease on time. A CNN based model was developed for citrus fruit and leaf diseases detection with testing accuracy of 94.55% [6]. CNN based VGG16 and InceptionV3 architectures have been used to detect rice diseases and pests, it achieved mean validation accuracy of 97.12% [7]. ML and DL approaches are used in a variety of published literature for the automatic classification and detection of plant diseases [8]. DL techniques including two-stage and single-stage detection models have been utilized for detection of plant diseases [9] [10]. Comparison studies have concluded that in terms of plant disease detection, DL methods outperform ML methods [2] [9].

YOLO is a simple and fast single-stage object detection algorithm [11]. These algorithms have simplified the object detection problem by adopting a single-stage approach that regresses the bounding box coordinates and calculates label probabilities simultaneously, thus improving efficiency and accuracy. Therefore, it determines information about the context of classes and their appearance. It combines individual modules of object detection into a neural network and contains multiple benefits over conventional object detection techniques [12].

The original YOLO, YOLOv1, employed GoogleLeNet architecture and divided the input image into cells for object detection, with Non-Maximal Suppression (NMS) to refine and select the best bounding boxes. YOLO variants improved upon this by using Darknet-19 architecture, introducing anchor boxes for better object identification, clustering algorithms for bounding box generation, and multi-scale training to enhance small object detection. Further modifications utilized Darknet-53 architecture, anchor boxes with multiple aspect ratios and scales, and a multi-scaled prediction mechanism for improved object detection at various scales. The evolved YOLO models introduced CSPDarknet53 architecture, Cross Stage Partial (CSP) connections, and various modules like Spatial Pyramid Pooling (SPP), Path Aggregation Network (PANet), and Self-Adversarial Training (SAT) for superior performance and speed.

YOLOv5 used CBS modules, C3 modules, focus module and SPP module in the backbone network, top-down FPN, bottom-up PAN and detects multi-scale feature maps. It incorporates multiple data augmentation techniques for efficient training to improve the model's ability to generalize [13]. Further YOLO variant utilized the EfficientRep backbone, Rep-PAN, efficient decoupled head and Task Alignment Learning (TAL) for label assignment.

YOLOv7 introduced the CSPVoVNet architecture, extended ELAN, compound model scaling, and planned re-parameterization convolution to enhance accuracy and speed. It directly integrates batch normalization with convolution layers and utilizes EMA model for more decisive inference.

YOLOv8 used C2f modules, SPPF module, combination of FPN and PAN as multi-scale fusion module and adopted anchor-free detection approach [14]. It optimizes the loss function and employs mosaic augmentation during training while turning it off during the final training epochs to enhance overall performance.

Advancements in YOLO algorithms have significantly improved the efficiency and accuracy of object detection systems, making them invaluable in various applications, including plant disease detection and geographical information science. Several studies have previously used YOLO network models and its variations for plant disease detections. A modified YOLOv3 tiny model containing residual network structure with convolutional neural network (CNN) was used for early detection of main diseases of turmeric plant and it attained an F1 score of 83.4% [15]. A hybrid approach was used for automated citrus plant diseases detection in which YOLOv4 model was optimized for disease detection and EfficientNet model was used for classification and the study achieved F1 score and accuracy of 95.3% and 96.4% respectively [16]. An improved version of YOLOv5 model including squeeze and excitation module was used to detect tomato virus diseases, it accomplished accuracy and mean average precision of 91.07% and 94.10% respectively [17]. An automatic detection system was developed for citrus greening disease using the YOLOv5l model with Micro F1 score of 85.19% [18].

A hybrid model combining two-stage and single-stage detection networks, incorporating YOLO algorithm fused with Faster-RCNN, achieved a mean Average Precision (mAP) of 85.2% [10]. The single-stage detection models have achieved higher performance on even larger datasets compared to this hybrid model [19]. [20] used an enhanced YOLOv5s model, with improvements in various aspects, outperformed YOLOv4-tiny, YOLOX-s, and other models with a 93.1% mAP for detecting plant diseases.

The proposed model from [21] was compared to YOLO v3 and v4, RetinaNet, EfficientNet, M2det, SSD and CenterNet, it achieved mAP of 93.83% and F1 score of 95.4%. Yolov5-CAct model with mAP of 94.24% was proposed by [19] for disease detection of multiple crop species, which used repeated augmentation, focal loss, early stopping for model convergence and two customized techniques for performance enhancement. YOLO-JD model achieved detection accuracy mAP of 96.63% and F1 score of 95.83% for jute plant disease and pests' detection. The results show that mAP value of the proposed model is lower than the YOLOv5x model for some classes [22].

A lightweight YOLOv4 model based on MobileNet v3 backbone with three tailored modules and transfer learning technique was developed for detection of cucumber leaves. The model attained mAP of 97.21% for detection of cucumber leaves [23]. A YOLOv5s based Apple-YOLO model is proposed including three customized modules for detection of apple leaf diseases. It achieved mAP of 96.04% and inference speed of 34 Frames Per Second (FPS) for detection of apple plant diseases [24].

A study [9] compared two-stage and single-stage detection models including CenterNet, CenterNet2, Scaled YOLO v4 P5/P6/P7, Faster and Cascade RCNN, Deformable DETR, DetectoRS and FoveaBox for detection of citrus plant diseases. It determined that CenterNet2 achieved mAP of 91.4% for detection of citrus plant diseases. The results show that two models achieved high accuracy for detection of different classes, CenterNet2 achieved mAP of 94.8% for Melanose disease detection and Scaled YOLOv4 P7 achieved higher mAP of 92.8% and 93.9% for detection of Anthracnose and Bacterial brown spot diseases respectively [9].

The YOLOv7 model [25] was used for plant disease detection and the developed model attained 65% mAP, 59% precision and recall of 65%. A study proposed an EFDet i.e., efficient detection model based on EfficientNet backbone, SSD prediction module and feature fusion module, it achieved mAP of 85.52% when compared to EfficientDet-D1, YOLO V5(S), YOLO V4, YOLO V3-ASFF, YOLO V3, DSSD, RetinaNet, Faster RCNN and SSD. The study results show that the EfficientDet-D1 model achieved a higher mAP of 85.92% [26]. YOLOv5 and Mask RCNN models achieved average precision (AP) of 99% for detection of Alternaria alternata and Thrips citrus diseases [27]. YOLOv7 achieved mAP of 98.2% for tea leaf diseases in natural scene images which is better compared to CNN, Deep CNN, DNN, AX-Retina Net and improved DCNN [28]. Fusion Transformer (FTR) YOLO achieved mean Average Precision (mAP) of 90.67% for grape disease detection [29].

These research studies illustrate the significance of DL based object detection techniques for detection of plant diseases. The conducted studies have focused on various plant species and multiple diseases using diverse deep learning techniques. These have revealed promising results with practical implementations e.g., development of automated systems for early identification of diseases and pests.

There are many studies for plant disease detection using recent versions of YOLO models but with the improvements in YOLO models, there is still a research gap to evaluate the performance in the plant disease detection accuracy and compare it with the previous versions. The datasets used in many studies are small which may result in low performance of model training, have data quality issues affecting model accuracy, and lack proper validation data. To resolve these issues, a high-quality dataset of citrus plant disease which is properly annotated by the experts is used for this study [9].

III. RESEARCH METHODOLOGY

A. Data Collection

The image dataset of citrus plant leaves is obtained from Conghua Orchard, Conghua District, Guangzhou, China and can be used to identify and locate diseases [9]. The dataset, curated for the purpose of Citrus Plant Disease Detection is an unstructured dataset, comprising of a collection of 2,684 images which also includes 427 images of healthy leaves. There are three types of citrus diseases in the dataset including Disease A: Anthracnose (37%), Disease B: Melanose (35%) and Disease C: Bacterial Brown Spot. The dataset contains different variations of leaf shape, size, color, position, and lighting conditions and has multiple cases of the same or different diseases on one leaf. Dataset is labelled and correctly annotated with the help of experts to accurately identify the diseases. It contains 10,332 annotations of citrus plant diseases and these are saved in XML and JSON formats. This dataset is available for public usage [30] [31].

B. Data Pre-processing

The data preprocessing helps to improve the data quality for further data exploration and data modeling. Data preprocessing of unstructured data i.e., images, involves conversion of the image annotation files to a standard format, maintaining dataset in a proper hierarchical structure for learning algorithms and data augmentation techniques to prepare data for modeling phase.

The dataset contains RGB images of 1000 x 1000 resolution. Image annotations were converted from JSON to YOLO Darknet text format. Null annotations were added for the healthy images. Each line in the text file includes the object label, object center coordinates, height, and width. There is a numerical representation of the object label which starts from zero, at the start of each line in the text file. The annotations are normalized between the range of 0 and 1 so it can work even after image scaling. The training, validation and testing dataset, all have 'images' and 'labels' directories of respective data according to data split strategy. A configuration file 'data.yaml' has been created which contains information of training, validation and testing images path, number of classes and object labels array. The dataset is distributed into training, validation, and test segments of 70%, 20% and 10% respectively. The training dataset contains 1,878 RGB images of 1000 x 1000 resolution and a total of 7,095 annotations for 4 classes including 313 healthy leaf images.

Data augmentation techniques had already been applied on the dataset, so images are flipped horizontally and vertically, rotated at random angles, rotated at 900 clockwise and counterclockwise, randomly blurred, padded, cropped and color augmentations including brightness, contrast, saturation, and hue added. Figure 1 shows images of single and multiple instances of same disease on one leaf, it also shows single and multiple instances of different diseases on one leaf.

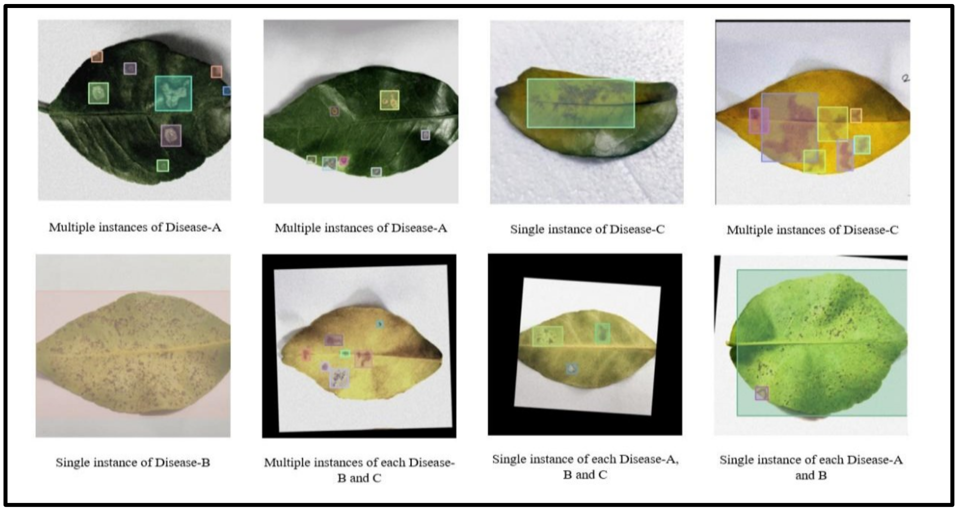

Figure 1. Dataset sample with disease bounding boxes

C. Modeling

The modeling phase includes training and testing of learning algorithms based on performance metric of Mean Average Precision at 50% to 95% Intersection over Union score i.e. mAP@50-90. The deep learning models including YOLOv5s, YOLOv7 and YOLOv8s were trained on the preprocessed dataset of infected and healthy leaves and evaluated on multiple samples with different conditions i.e., flipping, random rotation and resolution etc. to identify the plant disease and label the area on the image. Validation data was used to avoid model overfitting and to fine tune the model hyperparameters. The models were trained for 50, 100 and 200 epochs. There were multiple kinds of losses which are observed during training and validation phases. Box loss refers to the ability of an algorithm to locate the center of an object and coverage of an object by the predicted bounding box. Object loss or objectness loss refers to the probability of the existence of an object in the proposed area of interest. Class loss or classification loss refers to the ability of an algorithm to correctly predict the class of an object. The continuous box location distribution is Distributed Focal Loss (DFL) as a discretized probability distribution. The minimum score required for the model to consider a prediction to be accurate is known as the confidence threshold. The value of patience refers to the number of epochs to be observed with no significant improvement before stopping the training early and it helps to pick the best model. Figure 2 provides the steps of process flow including data preprocessing, modeling, evaluation and deployment.

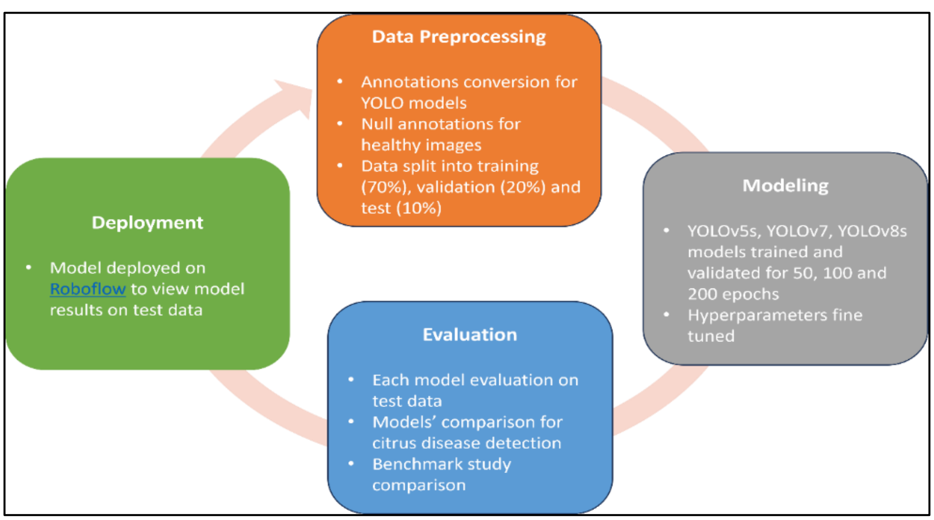

Figure 2. Process Flow Diagram

The performance of the developed models was evaluated based on mean Average Precision mAP@50-95. It is based on precision, recall and Intersection over Union (IoU) metrics. Precision is the ratio between correctly predicted plant disease as True and total predicted positive cases whereas recall is the ratio between correctly predicted plant diseases as True and total actual positive cases.

$$Precision = \frac{TP}{TP + FP} \qquad (Eq. 1)$$

$$Recall = \frac{TP}{TP + FN} \qquad (Eq. 2)$$

Intersection over Union (IoU) is used to calculate the overlap between two bounding boxes using Eq. 3. If the predicted bounding box and ground truth are the same, then IoU is 1. If the predicted bounding box and ground truth are completely missed, then IoU is 0. The degree of overlap will produce a value for IoU between 0 and 1 using Eq. 3.

$$IoU = \frac{Intersection\ Area\ of\ Predicted\ Bounding\ Box\ and\ Ground\ Truth}{Union\ Area\ of\ Predicted\ Bounding\ Box\ and\ Ground\ Truth} \qquad (Eq. 3)$$

Precision and recall should be higher for better performance of the model. Average Precision (AP) is described as an area under precision recall curve. A threshold value represents the degree of IoU e.g., 50%, and is used to evaluate a prediction as true positive. The IoU threshold value is changed over a range between 50% to 95% with a step of 5%, to reduce the bias. It will also change the precision and recall values and these value pairs can be plotted on a graph to view precision recall curve and average of these values is described as AP@50-95. It shows that the model is stable across different threshold values.

$$mAP = \frac{1}{n}\sum_{k=1}^{n} AP_k \qquad (Eq. 4)$$

Mean Average Precision (mAP) is the average AP of all classes, it is a performance measure for object detection and calculated using Eq. 4. The mAP at IoU of 50-95% i.e., mAP@50-95 is used for this study.

A study was conducted for citrus plant disease detection on the same dataset and compared the performance of multiple models for disease detection [9]. So, it will be used for a performance benchmark in this study. The results of the benchmark study showed that the CenterNet2 model achieved the highest mAP of 91.4% for an average of all diseases. Further, CenterNet2 model achieved AP of 94.8% for Melanose, Scaled YOLO v4 P7 model achieved AP of 92.8% and AP of 93.9% for Anthracnose and Bacterial brown spot, respectively. Table 1 shows the mean average precision of the models for all disease detection and the best performance of the models for detection of each disease.

Table 1. Performance Benchmark for disease detection

Model	**mAP (%)**	**Disease (AP (%))**
Scaled YOLO v4 P7	89.3	Anthracnose (**92.8**), Bacterial Brown Spot (**93.9**)
CenterNet2	91.4	Melanose (**94.8**)

In this study, each model i.e., YOLOv5s. YOLOv7 and YOLOv8s was trained for 50, 100, 200 epochs to determine the gradual increase in the performance. The performance benchmark study also used maximum 200 epochs for YOLO models[9].

Google Colab python environment with Tesla T4 GPU (15102 MiB), 2 CPUs and 12.7 GB RAM was used for the experiments. Python libraries such as PyTorch and Ultralytics were used.

IV. RESULTS AND DISCUSSIONS

A. Models' Training and Hyperparameter Fine Tuning

The arguments and hyperparameters for model training were initially configured as IoU threshold of 0.6, optimizer adam with initial learning rate of 0.001, patience value of 20, batch size of 16, image size of 512, number of workers to 8, weight decay of 0.001, box loss gain of 7.5, class loss gain of 0.5 and object loss gain of 0.7. The batch size, image size and number of workers are set to optimum values according to accessible computational resources. Image augmentation helps to improve the model performance, so these parameters are also configured for dataset training i.e., hsv-hue of 0.015, hsv-saturation of 0.7, hsv-value of 0.4, images translation of 0.1, image scaling of 0.5, flip right left of 0.5 and image mosaic probability of 1.0.

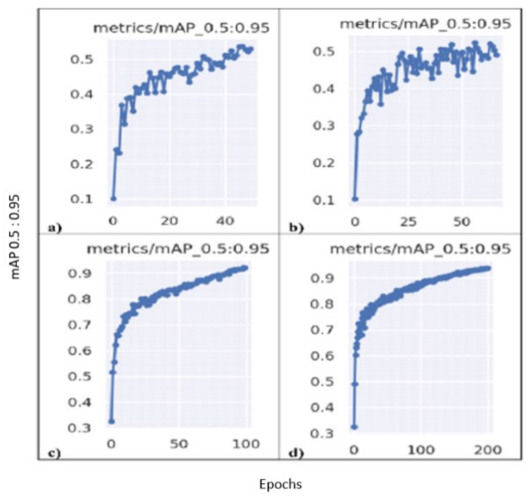

Figure 3. YOLOv5s training performance, mAP w.r.t epochs. a) for 50 Epochs, b) performance with early stopping at 67 Epochs, c) for 100 Epochs, and d) performance for 200 Epochs

Figure 3a shows performance metric of YOLOv5s for 50 epochs. The mAP@50-95 is 52.97% and the performance metric graph shows spikes in the initial epochs, but these are reducing with the number of epochs and the performance is gradually increasing. Training box loss, object loss and class loss were 2.18, 0.02 and 0.02, respectively. Validation box loss, object loss and class loss were 1.63, 0.01 and 0.02, respectively. Validation box and object losses were lower than training losses. Box and object losses of both training and validation are decreasing with the number of epochs. It is also observed that the class loss values are still decreasing with the number of epochs.

Figure 3b shows performance metrics of YOLOv5s for early stopping at 67 epochs. The mAP@50-95 is 49.01% and the performance metric graph shows more spikes and model performance has reduced with the number of epochs which indicates a problem in the model training. The results show that performance is decreasing with more epochs so several hyperparameters are fine-tuned to improve the model performance i.e., IoU threshold of 0.7, weight decay of 0.0005, box loss gain of 0.05, class loss gain of 0.5 and object loss gain of 1.0. The image augmentation parameters are not changed.

Figure 3c shows performance metrics of YOLOv5s for 100 epochs. The mAP@50-95 is 92.17% and the performance metric graph shows that mAP@50-95 is still increasing with the number of epochs so model performance is improving gradually. Whereas. Figure 3d shows performance metrics of YOLOv5s for 200 epochs. The mAP@50-95 is 93.82% and the performance metric graph has improved gradually compared to prior results.

The training and validation losses of YOLOv5s for 200 epochs are shown in Figure 4. Training box loss, object loss and class loss are 0.0115, 0.0089 and 0.0001, respectively. Validation box loss, object loss and class loss are 0.0088, 0.0047 and 0.0001, respectively. Validation losses are lower than training losses. The box loss and object loss for both training and validation are decreasing gradually and class loss for both training and validation is approaching a minimum.

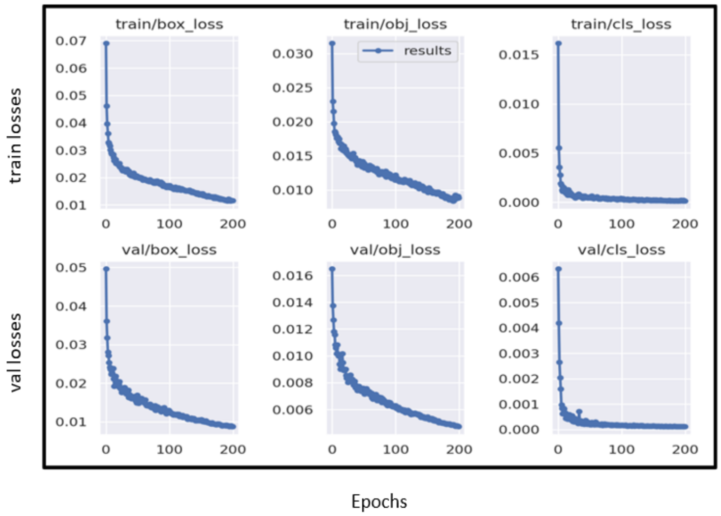

Figure 4. YOLOv5s Training and Validation Losses (200 Epochs)

YOLOv7 model was trained for 50, 100 and then 200 epochs using hyperparameters i.e., confidence threshold of 0.001, IoU threshold of 0.7, optimizer adam with initial learning rate of 0.001, batch size of 16, image size of 512, weight decay of 0.0005, box loss gain of 0.05, class loss gain of 0.3 and object loss gain of 0.7. YOLOv8s model was trained for 50, 100 and then 200 epochs using same hyperparameters except loss values i.e., box loss gain of 7.5, class loss gain of 0.5 and distributed focal loss of 1.5. Image augmentation parameters for both models were configured i.e., hsv-hue of 0.015, hsv-saturation of 0.7, hsv-value of 0.4, images translation of 0.1, image scaling of 0.5, flip right left of 0.5 and image mosaic probability of 1.0.

Models' performance improves with a greater number of epochs, whereas the jump in performance is gradual as shown in Figure 5. The performance of the YOLOv5s, YOLOv7 and YOLOv8s models for disease detection has an increasing trend with the number of epochs and mAP@50-95 has improved to 93.9% i.e., 5.1% from 50 epochs, 92.7% i.e., 4.6% from 50 epochs, 96.8% i.e., 3.0% from 50 epochs, respectively.

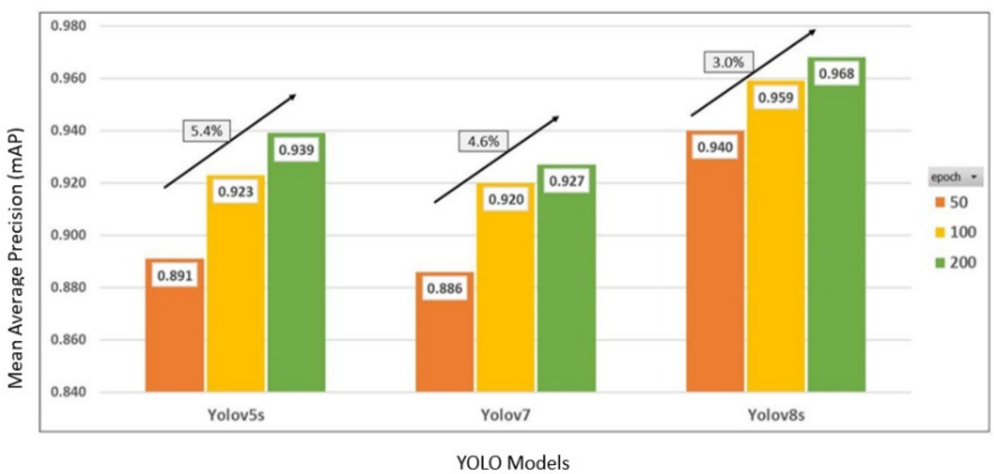

Figure 5. Validation mAP values for models w.r.t the number of epochs

B. Models' Comparison

Figure 3 and Figure 4 show YOLOv5s hyperparameter fine tuning to enhance model performance and model training w.r.t number of epochs. Figure 5 shows comparison of YOLOv5s, YOLOv7 and YOLOv8s models' training w.r.t number of epochs which helps to analyze models' performance on training and validation dataset. The results show that YOLOv8s has comparatively performed better in each number of epochs for citrus disease detection.

Mean average precision of YOLOv5s, YOLOv7 and YOLOv8s models' detection results of all diseases on train, test, and validation datasets after model training for 200 epochs are shown in Figure 6. YOLOv5s model has performed better on test data with mAP@50-95 of 94.2%. YOLOv7 has shown almost similar results for train, test, and validation with mAP@50-95 of 92.5%, 92.6% and 92.7% respectively. YOLOv8s model trained and validated for 200 epochs has performed better compared to YOLOv5s and YOLOv7 in training, testing and validation for detection of all citrus diseases i.e., mAP@50-95 of 96.8% on validation dataset.

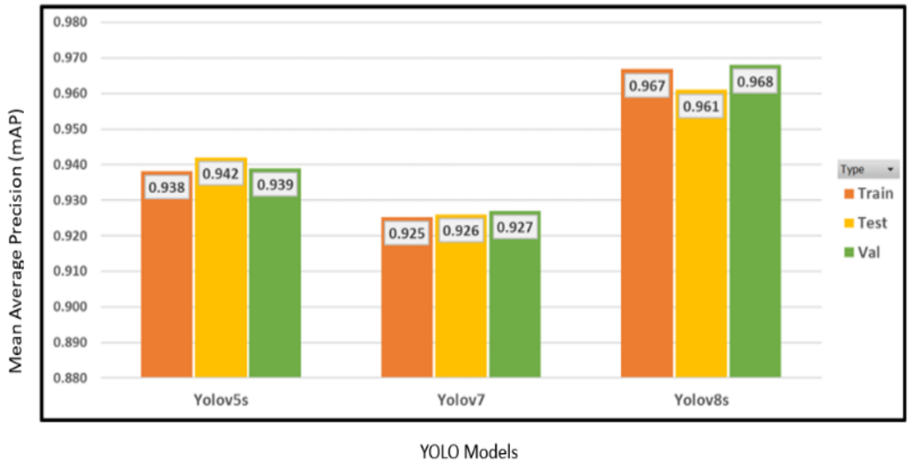

Figure 6. Overall mAP values for all diseases detection for train, test, and validation datasets of the 200 epochs trained models

The performance comparison of YOLOv5s, YOLOv7, YOLOv8s models for detection of each disease on the test dataset is shown in Figure 7. YOLOv8s has achieved the highest detection accuracy for each disease compared to YOLOv5s and YOLOv7. So, YOLOv8s model is the best model for each disease detection on current dataset with mAP@50-95 of 95.5%, 96.9% and 97.1% for Disease-A, Disease-B and Disease-C, respectively.

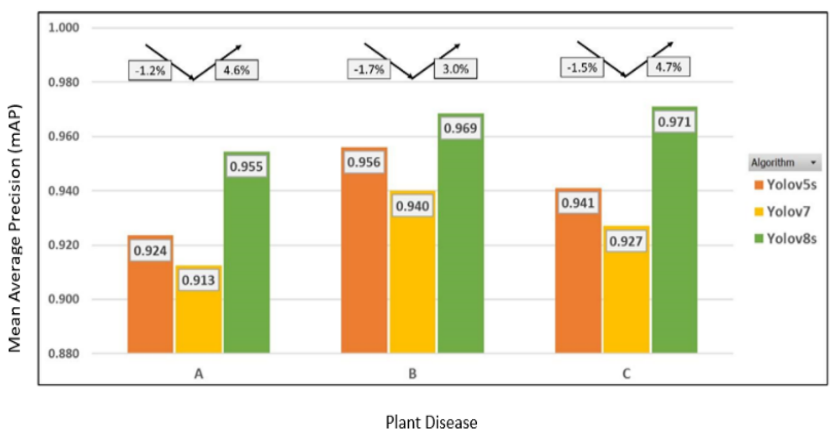

Figure 7. mAP Performance comparison of YOLOv5s, YOLOv7, YOLOv8s models for detection of each citrus disease on the test dataset

Mean average precision of YOLOv8s model trained for 200 epochs, on validation and test dataset for detection of each disease and all classes is shown in Figure 8.

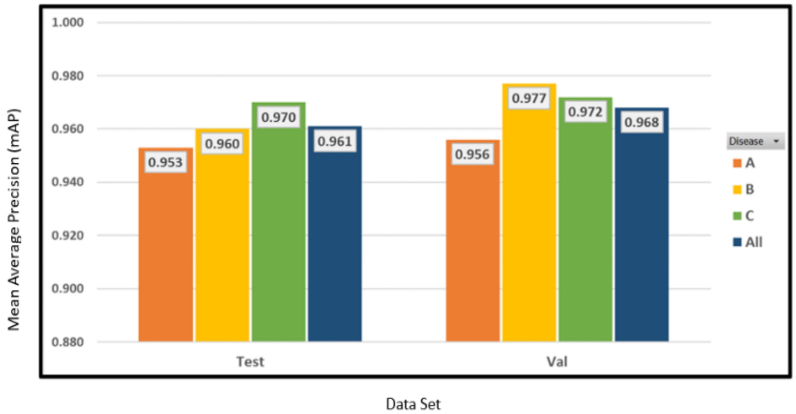

Figure 8. YOLOv8s final trained model mAP values for each disease on validation and test dataset

The trained YOLOv8s model trained for 200 epochs shows accurate predicted bounding boxes and labels with high confidence values as shown in Figure 9. YOLOv8s has correctly identified the healthy leaf, detected and labeled the diseases in the test dataset including single and multiple instances of same and different diseases in one image.

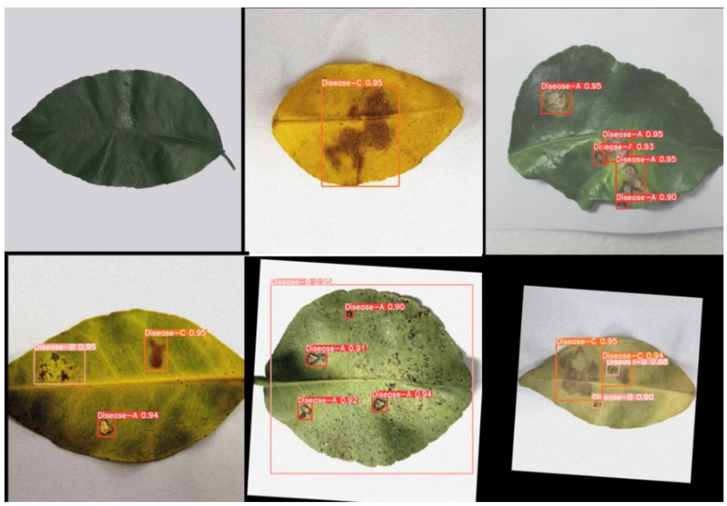

Figure 9. Predicted bounding boxes and labels with confidence value (YOLOv8s - Test Data)

It is determined from the experimental results that YOLOv5s, YOLOv7 and YOLOv8s models trained and validated for 200 epochs on the dataset have performed better than models trained for 50 and 100 epochs. YOLOv8s model trained for 200 epochs has achieved the highest performance compared to YOLOv5s and YOLOv7, for detection of each citrus disease and for mean average precision of all diseases.

C. Benchmark Study Comparison

The comparison between benchmark study models and YOLOv8s model of this study for single disease detection as shown in Figure 10 represent the conclusion that YOLOV8s has achieved higher performance than benchmark study results for citrus disease detection i.e., mAP of 95.3%, 96.0% and 97.0% for Disease-A, Disease-B and Disease-C, respectively.

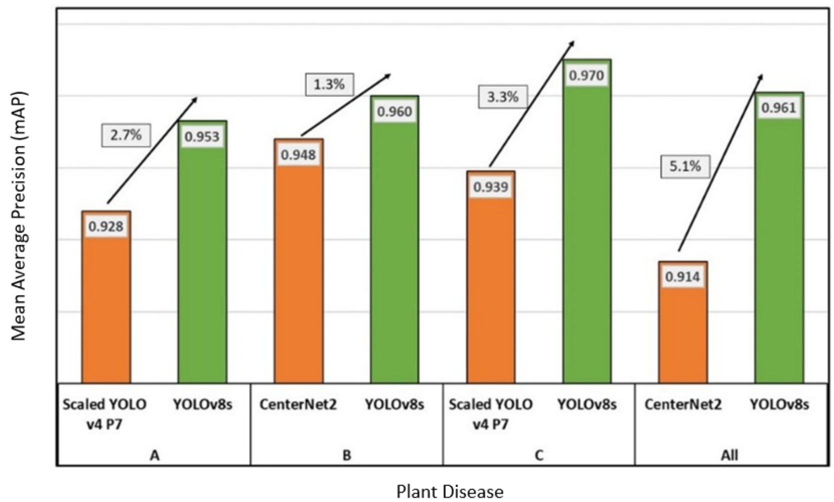

Figure 10. Benchmark Performance Comparison for Single and all Disease Detection

Figure 10 also shows performance comparison between benchmark study model CenterNet2 and YOLOv8s model for all citrus disease detection. YOLOv8s has achieved higher performance compared to benchmark study results i.e., mAP of 96.1% with a 5.1% improvement over CenterNet2 baseline. Based on the results, it can be stated that YOLOv8s model has achieved higher accuracy compared to benchmark study results for detection of each disease as well as mean average precision for detection of all citrus plant diseases on CCL'20 dataset.

D. Model Deployment

YOLOv8s model trained and validated for 200 epochs was the best model used for citrus plant disease detection and had been deployed on Roboflow platform [32]. Figure 11 shows the web interface to upload an image for detection of citrus plant disease on the test dataset.

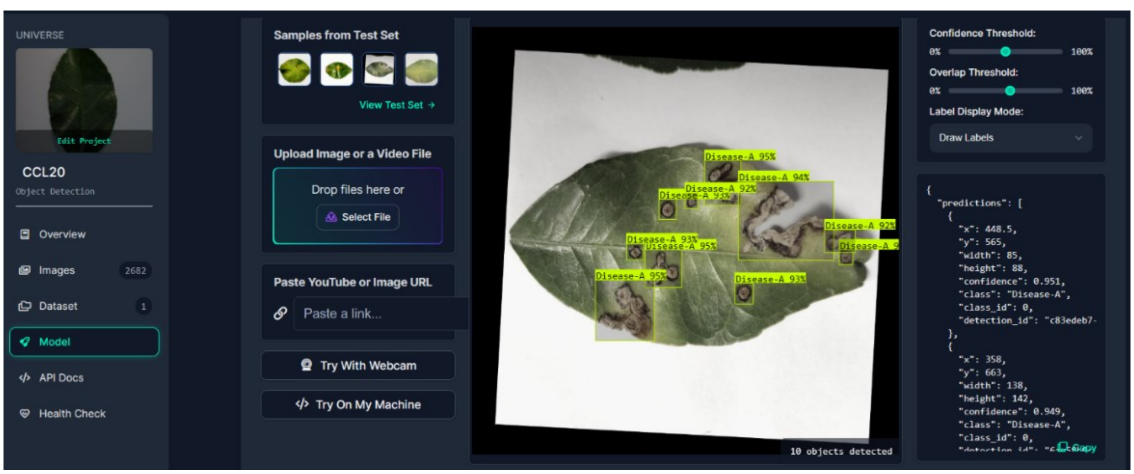

Figure 11. Web Interface of the deployed model

The user can register/sign in and select Model from the side menu to upload an image to test the model. Confidence and overlap threshold can be changed using sliders and label display mode can be changed from the dropdown menu. Samples from the test set are already available to test the model for citrus plant disease detection.

The main objectives effectively achieved in this study are, YOLOv5s, YOLOv7 and YOLOv8s models have been developed to detect and locate multiple cases of the same and different diseases in one image, it can also work on different variations of leaf shape, size, color, position and lighting conditions. Developed models have been evaluated based on performance metric of mean average precision mAP@50-95%. YOLOv8s model has been deployed on a web interface and can be used to view the detected citrus plant diseases on the test dataset.

V. CONCLUSION

The study focused on YOLOv5, YOLOv7 and YOLOv8 models for citrus plant disease detection. YOLOv8s has achieved 96.1% mAP@50-95% for citrus disease detection and mAP@50-95% of 95.3%, 96.0% and 97.0% for detection of Anthracnose, Melanose and Bacterial Brown Spot diseases, respectively. The developed model for detection of citrus diseases can significantly help the farmers to identify the disease and locate the infected region, improve crop yield production, efficient usage of resources, reduce cost and environmental effects. The model will also enable the potential for applications of Precision Agriculture Technologies (PATs).

The developed model is small and can be used for IoT devices e.g., precision agriculture technologies, however, larger models such as YOLOv8m, YOLOv8l and YOLOv8x can be custom trained for more datasets and complex scenarios. The model has better detection speed and accuracy in experimental results, but it needs to be further experimented for real-time detection in real-world scenarios.

For future work, the developed model should be trained and validated on multiple datasets for different plant species disease detection. The developed model can be integrated with PATs e.g., an aerial high resolution imagery crop survey drone and custom developed application based on Global Navigation Satellite System (GNSS) for field location marking, to detect the plant diseases and locate the infected field areas which can be helpful to plan and execute pesticide spray activity, control the disease, and increase crop productivity.

ACKNOWLEDGEMENT

The research for this paper was not funded.

AUTHOR CONTRIBUTIONS

Usman Ali: Conceptualization, Data Curation, Formal Analysis, Investigation, Methodology, Validation, Visualization, Writing – Original Draft Preparation;
Maizatul Akmar Ismail: Conceptualization, Project Supervision, Resources, Writing – Review & Editing;
Riyaz Ahamed: Supervision, Writing – Review & Editing
Syed Rosyan: Writing – Review & Editing

CONFLICT OF INTERESTS

No conflict of interests were disclosed.

REFERENCES

[1] W. Albattah, M. Nawaz, A. Javed, M. Masood, and S. Albahli, “A novel deep learning method for detection and classification of plant diseases,” Complex and Intelligent Systems, vol. 8, no. 1, pp. 507 – 524, 2022, doi: 10.1007/s40747-021-00536-1.

[2] R. Sujatha, J. M. Chatterjee, N. Jhanjhi, and S. N. Brohi, “Performance of deep learning vs machine learning in plant leaf disease detection,” Microprocess Microsyst, vol. 80, p. 103615, Feb. 2021, doi: 10.1016/j.micpro.2020.103615.

[3] S. K. Balasundram, K. Golhani, R. R. Shamshiri, and G. Vadamalai, “Precision Agriculture Technologies for Management of Plant Diseases,” in PLANT DISEASE MANAGEMENT STRATEGIES FOR SUSTAINABLE AGRICULTURE THROUGH TRADITIONAL AND MODERN APPROACHES, vol. 13, I. UlHaq and S. Ijaz, Eds., in Sustainability in Plant and Crop Protection, vol. 13. , 2020, pp. 259–278. doi: 10.1007/978-3-030-35955-3_13.

[4] I. El Ouadi, “Understanding the Climate-Water-Energy-Food Nexus and the Transition Towards a Circular Economy: The Case of Morocco,” in E3S Web of Conferences, 2023. doi: 10.1051/e3sconf/202336401006.

[5] D. Kumar and P. Kalita, “Reducing Postharvest Losses during Storage of Grain Crops to Strengthen Food Security in Developing Countries,” Foods, vol. 6, no. 1, 2017, doi: 10.3390/foods6010008.

[6] A. Khattak et al., “Automatic Detection of Citrus Fruit and Leaves Diseases Using Deep Neural Network Model,” IEEE Access, vol. 9, pp. 112942–112954, 2021, doi: 10.1109/ACCESS.2021.3096895.

[7] C. R. Rahman et al., “Identification and recognition of rice diseases and pests using convolutional neural networks,” Biosyst Eng, vol. 194, pp. 112–120, Jun. 2020, doi: 10.1016/j.biosystemseng.2020.03.020.

[8] S. Jain et al., “Automatic Rice Disease Detection and Assistance Framework Using Deep Learning and a Chatbot,” Electronics (Basel), vol. 11, no. 14, 2022, doi: 10.3390/electronics11142110.
[9] S. Dananjayan, Y. Tang, J. Zhuang, C. Hou, and S. Luo, “Assessment of state-of-the-art deep learning based citrus disease detection techniques using annotated optical leaf images,” Comput Electron Agric, vol. 193, p. 106658, Feb. 2022, doi: 10.1016/j.compag.2021.106658.
[10] M. Li et al., “High-Performance Plant Pest and Disease Detection Based on Model Ensemble with Inception Module and Cluster Algorithm,” Plants, vol. 12, no. 1, 2023, doi: 10.3390/plants12010200.
[11] J. Du, “Understanding of Object Detection Based on CNN Family and YOLO,” J Phys Conf Ser, vol. 1004, no. 1, p. 12029, Apr. 2018, doi: 10.1088/1742-6596/1004/1/012029.
[12] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779–788. doi: 10.1109/CVPR.2016.91.
[13] Y. Huang, Y. Qian, H. Wei, Y. Lu, B. Ling, and Y. Qin, “A survey of deep learning-based object detection methods in crop counting,” Comput Electron Agric, vol. 215, p. 108425, 2023, doi: https://doi.org/10.1016/j.compag.2023.108425.
[14] X. Zhai, Z. Huang, T. Li, H. Liu, and S. Wang, “YOLO-Drone: An Optimized YOLOv8 Network for Tiny UAV Object Detection,” Electronics (Basel), vol. 12, no. 17, 2023, doi: 10.3390/electronics12173664.
[15] V. Devisurya, R. Devi Priya, and N. Anitha, “Early detection of major diseases in turmeric plant using improved deep learning algorithm,” Bulletin of the Polish Academy of Sciences: Technical Sciences, vol. 70, no. 2, 2022, doi: 10.24425/bpasts.2022.140689.
[16] X. Zhang, Y. Xun, and Y. Chen, “Automated identification of citrus diseases in orchards using deep learning,” Biosyst Eng, vol. 223, pp. 249–258, Nov. 2022, doi: 10.1016/j.biosystemseng.2022.09.006.
[17] J. Qi et al., “An improved YOLOv5 model based on visual attention mechanism: Application to recognition of tomato virus disease,” Comput Electron Agric, vol. 194, p. 106780, Mar. 2022, doi: 10.1016/j.compag.2022.106780.
[18] R.-Z. Qiu et al., “An automatic identification system for citrus greening disease (Huanglongbing) using a YOLO convolutional neural network,” Front Plant Sci, vol. 13, 2022, doi: 10.3389/fpls.2022.1002606.
[19] G. Dai and J. Fan, “An Industrial-Grade Solution for Crop Disease Image Detection Tasks,” Front Plant Sci, vol. 13, 2022, doi: 10.3389/fpls.2022.921057.
[20] J. Li, Y. Qiao, S. Liu, J. Zhang, Z. Yang, and M. Wang, “An improved YOLOv5-based vegetable disease detection method,” Comput Electron Agric, vol. 202, p. 107345, Nov. 2022, doi: 10.1016/j.compag.2022.107345.
[21] W. Bao, T. Fan, G. Hu, D. Liang, and H. Li, “Detection and identification of tea leaf diseases based on AX-RetinaNet,” Sci Rep, vol. 12, no. 1, p. 2183, 2022, doi: 10.1038/s41598-022-06181-z.
[22] D. Li, F. Ahmed, N. Wu, and A. I. Sethi, “YOLO-JD: A Deep Learning Network for Jute Diseases and Pests Detection from Images,” Plants, vol. 11, no. 7, 2022, doi: 10.3390/plants11070937.
[23] X. Li, Z. Yue, J. Su, S. Wang, J. Hua, and F. Duan, “Application of Lightweight Object Detection Network in Cucumber Leaf Detection,” in 2022 IEEE International Conference on Mechatronics and Automation (ICMA), 2022, pp. 686–691. doi: 10.1109/ICMA54519.2022.9855974.
[24] J. Li, X. Zhu, R. Jia, B. Liu, and C. Yu, “Apple-YOLO: A Novel Mobile Terminal Detector Based on YOLOv5 for Early Apple Leaf Diseases,” in 2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC), 2022, pp. 352–361. doi: 10.1109/COMPSAC54236.2022.00056.
[25] P. Nayar, S. Chhibber, and A. K. Dubey, “An Efficient Algorithm for Plant Disease Detection Using Deep Convolutional Networks,” in 2022 14th International Conference on Computational Intelligence and Communication Networks (CICN), 2022, pp. 156–160. doi: 10.1109/CICN56167.2022.10008235.
[26] C. Liu, H. Zhu, W. Guo, X. Han, C. Chen, and H. Wu, “EFDet: An efficient detection method for cucumber disease under natural complex environments,” Comput Electron Agric, vol. 189, p. 106378, Oct. 2021, doi: 10.1016/j.compag.2021.106378.
[27] S. Uğuz, G. Şikaroğlu, and A. Yağız, "Disease detection and physical disorders classification for citrus fruit images using convolutional neural network," Food Measure, vol. 17, pp. 2353–2362, 2023. [Online]. Available: https://doi.org/10.1007/s11694-022-01795-3
[28] M. J. A. Soeb, M. F. Jubayer, T. A. Tarin et al., "Tea leaf disease detection and identification based on YOLOv7 (YOLO-T)," Sci Rep, vol. 13, 6078, 2023. [Online]. Available: https://doi.org/10.1038/s41598-023-33270-4
[29] Y. Liu, Q. Yu, and S. Geng, "Real-time and lightweight detection of grape diseases based on Fusion Transformer YOLO," Frontiers in Plant Science, vol. 15, p. 1269423, 2024.
[30] Researchzkhu, "Researchzkhu/CCL-20," GitHub. [Online]. Available: https://github.com/researchzkhu/CCL-20
[31] "CCL’20," Kaggle. [Online]. Available: https://www.kaggle.com/datasets/downloader007/ccl20
[32] "How to use the CCL20 object detection API," Roboflow. [Online]. Available: https://universe.roboflow.com/mds-h6qle/ccl20/model/1