# Revolutionizing Plant Disease Detection in Agriculture: a Comparative Study of Yolov5 and Yolov8 Deep Learning Models

**Vaibhav Hawaldar**
vaibhav.hawaldar197@nmims.edu.in

Narsee Monjee Institute of Management Studies

**Rishabh Jain**
Narsee Monjee Institute of Management Studies

**Mihir Mengde**
Narsee Monjee Institute of Management Studies

**Supriya Agrawal**
Narsee Monjee Institute of Management Studies

**Research Article**

**Keywords:** Object detection, Deep-learning, YOLO, YOLOv5, YOLOv8, Comparative study

**Posted Date:** March 5th, 2024

**DOI:** https://doi.org/10.21203/rs.3.rs-3936780/v1



**Additional Declarations:** No competing interests reported.

# REVOLUTIONIZING PLANT DISEASE DETECTION IN AGRICULTURE: A COMPARATIVE STUDY OF YOLOv5 AND YOLOv8 DEEP LEARNING MODELS

Vaibhav Hawaldar[1*], Rishabh Jain[2†], Mihir Mengde[3†], Supriya Agrawal[4†]

[1]Computer Engineering, SVKM's Mukesh Patel School of Technology Management and Engineering, Mumbai, 400056, Maharashtra, India.
[2]Computer Engineering, SVKM's Mukesh Patel School of Technology Management and Engineering, Mumbai, 400056, Maharashtra, India.
[3]Computer Engineering, SVKM's Mukesh Patel School of Technology Management and Engineering, Mumbai, 400056, Maharashtra, India.
[4]Computer Engineering, SVKM's Mukesh Patel School of Technology Management and Engineering, Mumbai, 400056, Maharashtra, India.

Contributing authors: vaibhav.hawaldar197@nmims.edu.in; rishabh.jain117@nmims.edu.in; mihir.mengde055@nmims.edu.in; Supriya.agrawal@nmims.edu;
†These authors contributed equally to this work.

**Abstract**

Object detection stands as a pivotal task within computer vision, finding extensive use across various domains. Recent years have witnessed a transformative shift in object detection thanks to deep learning methodologies, with You Only Look Once(YOLO) emerging as a prominent algorithm in this field. In this research paper, our focus lies in conducting an in-depth comparative analysis between two advanced deep learning models, You Only Look Once Version 5(YOLOv5) and You Only Look Once Version 8 (YOLOv8), to assess their applicability in the context of plant leaf disease detection within the agricultural sector. Our results unequivocally establish YOLOv8 as the superior performer, exhibiting exceptional precision, recall, and class differentiation, and notably, outperforming YOLOv5 by approximately 3% in mean average precision (mAP). This study demonstrates the prowess of YOLOv8 as a state-of-the-art object detection algorithm, offering implications for diverse applications beyond agriculture.

**Keywords:** Object detection, Deep-learning, YOLO, YOLOv5, YOLOv8, Comparative study

# 1 Introduction

Agriculture occupies a pivotal position in the global economy, with its contribution extending to both economic growth and livelihood sustenance. According to World Bank, globally, this sector constitutes approximately 4% of the gross domestic product (GDP), emphasizing its substantial role in economic activities. It is worth noting that in certain least-developed countries, agriculture assumes an even more pronounced significance, accounting for over 25% of their GDP. The significant variation highlights agriculture's diverse impact on economies and its vital role in addressing economic, social, and environmental challenges. Tailored strategies are crucial for recognizing and utilizing agriculture's full importance in sustainable development.

However, this crucial sector could have made a more substantial contribution if it could effectively tackle various challenges, such as nematodes, insect pests, diseases, and weeds that harm plants, reduce crop quality and yield, and consequently result in significant economic losses for the agricultural industry and farming community [1][2]. To prevent further crop damage and maintain overall plant health, early detection of diseases is vital [3]. This practice helps reduce the spread of diseases and allows for effective agricultural management. Traditional methods of disease detection, primarily relying on visual scouting, are time-consuming and can lead to lower crop yields, diminishing profits[4]. Therefore, to meet the increasing demands of our agriculture sector and efficiently resolve these issues, modernizing agriculture has become more critical than ever [5].

Deep learning has emerged as a game-changing solution by integrating object detection methods with its capabilities, particularly in identifying crop diseases. This technology significantly enhances decisions related to crop management. Advanced deep-learning models like YOLO have different versions like YOLOv5 and YOLOv8 which accurately detect diseases and assess the overall health of plant leaves. These models save time and effort while providing more accurate results in identifying crop diseases, ultimately maximizing profits and yields [6] [7][8].

This paper focuses on implementing various deep-learning models, with a prominent emphasis on YOLOv5 and YOLOv8. Our first objective involves comprehensively training these models and conducting a meticulous comparative analysis of their performance. Our study's second contribution is introducing a unique, custom-annotated dataset. This dataset, previously unavailable in the context of YOLO-based plant disease detection, offers an invaluable resource for comparing YOLO models, setting benchmarks, and tackling agricultural challenges.

Throughout this research, our primary focus remains on conducting a rigorous comparative analysis of YOLOv5 and YOLOv8 to shed light on the most effective approach for advancing agricultural outcomes.

## 2 Literature Review

In the early stages of plant disease detection, researchers relied solely on visual aids to determine whether a plant was diseased or healthy. As technology progressed, they turned to image analysis, primarily centered around color-based distinctions and machine learning techniques used after extracting features from the images, to enhance their accuracy.

Zhang, Y., Yin, X., Xu, T., Zhao, J.(2009) employ PCA (Principle Component Analysis) and LDA (Linear Discriminant Analysis) for cherry tomato maturity determination. Using three images from different angles per tomato, PCA and LDA differentiate ripe states (immature, half-ripe, full-ripe), achieving 94.9% accuracy. This methodology captures the color and quality features of tomatoes. This study showcases the efficacy of machine vision for real-time sorting and maturity assessment of cherry tomatoes [9]. Meenu Dadwal, V.K.Banga(2012) integrates RGB (Red-Green-Blue) color analysis and fuzzy logic to estimate fruit ripeness. After color segmentation (of color image) and mean value calculation (for red, blue, and green layers), fuzzy logic classification assigns ripeness levels, addressing the lighting variation of images due to changes in the environment. This approach yields accurate fruit ripeness assessment by accounting for uncertainty in color changes [10].

Van Huy Pham and Byung Ryong Lee(2014) introduce a two-step fruit defect detection methodology. Initial k-means clustering segments images, followed by a graph-based algorithm that refines results by enhancing accuracy. This balancing method improves accuracy and slightly increases the consumption time. The combination of k-means clustering and graph-based refinement proves effective for robust fruit defect detection [11].

Traditional methods for plant disease detection encountered limitations due to occlusion, complex patterns, and variable lighting conditions, rendering their results unreliable. To address these challenges, researchers turned to deep learning models, harnessing their capabilities to overcome occlusion and intricate disease patterns, ultimately enhancing the effectiveness of plant disease detection.

Murk Chohan, Adil Khan, Rozina Chohan, Saif Hassan Katpar, Muhammad Saleem Mahar(2020) introduces a deep learning model for plant disease detection. With over 95% accuracy on a real-world dataset (PlantVillage Dataset), the model displays promising potential for on-site plant disease identification. The study utilizes a Convolutional Neural Network (CNN) architecture, trained on a publicly available dataset.

Despite challenges in varying conditions, the model demonstrates accurate disease classification, raising prospects for integration into live disease identification systems [12]. Mahmoud Bakr, Sayed Abdel-Gaber, Mona Nasr, and Maryam Hazman(2022) highlight a comprehensive methodology that leverages DenseNet201 architecture and preprocessing techniques to surpass existing methods on PlantVillage plant datasets.

This proposed model achieves an impressive average accuracy of 98.23 percent by integrating preprocessing steps like k-means clustering.

In comparison research, DenseNet201 outperforms VGG16, Inception V3, and ResNet152V2, even with transfer learning. It serves as the feature extraction phase in the suggested model, followed by a CNN classifier, ultimately demonstrating the highest accuracy among the models considered. Thus, the proposed DenseNet201-based model emerges as the most effective choice for accurate plant disease detection [13].

While these models showed promise in mitigating occlusion and lighting problems, they struggled when confronted with the intricacies of complex patterns inherent in plant diseases. This dilemma led to the exploration of more advanced deep learning models, such as Faster R-CNN (Faster Region-Convolutional Neural Network) and the YOLO family, in the quest for improved plant disease detection.

Bari BS, Islam MN, Rashid M, Hasan MJ, Razman MAM, Musa RM, Ab Nasir AF, P.P. Abdul Majeed A.(2021) presents a real-time solution for rice leaf disease diagnosis, leveraging the Faster R-CNN framework with deep learning. Methodologically robust, the study encompasses data augmentation, annotation, and multi-phase training. By integrating deep learning and Faster R-CNN, the approach holds promise for revolutionizing real-time disease diagnosis in agriculture. The process involves CNN feature extraction, RPN-based (Region Proposal Network) candidate region generation, classification, and regression. Multi-phase training refines accuracy through iterative iterations. With the potential to mitigate crop loss, this approach amalgamates deep learning and innovation to enable precise real-time rice leaf disease diagnosis [14].

One-stage methods, exemplified by the YOLO series, directly pinpoint the target without generating numerous candidate boxes as required by two-stage methods like Faster R-CNN. In practical plant disease detection applications, one-stage methods exhibit higher effectiveness. For instance, despite the potential of Faster R-CNN, its two-stage approach posed limitations on overall efficacy, reinforcing the superiority of one-stage YOLO family models, particularly in the context of plant disease detection. One-stage models streamline the process, leading to improved efficiency and accuracy in identifying plant diseases.

Zhaoyi Chen, Ruhui Wu, Yiyan Lin, Chuyu Li, Siyu Chen, Zhineng Yuan, Shiwei Chen, Xiangjun Zou(2022) focused on enhancing the performance of the YOLOv5 model for the critical task of detecting rubber tree diseases in visible light images. To achieve this goal, they made significant modifications to the model architecture. Key improvements included replacing the conventional Bottleneck module with the more efficient InvolutionBottleneck module within the backbone network. Additionally, they integrated the SE (Squeeze-and-Excitation) module into the final layer of the backbone, facilitating more effective feature fusion. Further boosting accuracy, the authors switched from the Generalized Intersection over Union (GIOU) loss function to the Efficient Intersection over Union (EIOU) loss function, which accounted for variations in target frame dimensions and confidence levels. These meticulous refinements resulted in remarkable precision rates, with the enhanced YOLOv5 achieving 86.5 percent precision for powdery mildew detection and 86.8% precision for anthracnose detection [15].

Sajitha P, Alwin John, V L Devika, Gayathri S V, Nafla Sakhir(2023) introduces a comprehensive solution for plant disease detection, combining YOLO v7 and GPT-3. YOLO v7 accurately identifies leaf diseases with 96% accuracy. The integration of GPT-3, with its natural language processing prowess, generates actionable recommendations encompassing cultural practices, chemical treatments, and biological controls. This dual model approach not only detects diseases but also offers effective remedies, enhancing the system's utility. Outperforming traditional models, this amalgamation provides a holistic disease management solution, contributing to efficient crop health management [16].

Ping Li, Jishu Zheng, Peiyuan Li, Hanwei Long, Mai Li, and Lihong Gao (2023) showcase the transformative potential of MHSA- YOLOv8 in tomato assessment. Through meticulous image annotation, the study categorizes datasets into maturity grading and counting segments. Maturity grading involves labels like immature (IM), semi-mature (SM), and mature (M). In contrast, the counting segment simply uses "tomato" as the label. Central to the paper is the MHSA-YOLOv8 model, ingeniously integrating Multi-Head Self-Attention within its architecture. The paper highlights the model's dual applications: accurately grading tomato maturity and efficiently counting tomatoes while handling complexities. The validation of these models reaffirms their effectiveness in real-world applications, reflecting a significant advancement in agricultural innovation [17].

Within the YOLO family, several variants, including YOLOv5, YOLOv6, and YOLOv7, were put to the test in various studies. However, as technology evolved, YOLOv8 emerged as the newer and more advanced contender in the field. Recent research findings have highlighted the superiority of YOLOv8 over not only previous YOLO models but also other plant disease detection methods in general.

Hence, by applying Yolo:

- It improves object detection, can identify complex patterns, and can adapt to varying lighting.
- It offers accurate object localization and real-time performance.
- Its adaptability enhances accuracy and reliability, making it superior to traditional methods and other conventional deep learning models.

# 3 YOLOv5

## 3.1 Model Overview

YOLOv5, a cutting-edge one-stage target recognition algorithm leveraging Convolutional Neural Networks (CNNs), stands out for its exceptional speed and accuracy in object detection[18][19]. Originating in 2015 as YOLO under Joseph Redmon, the series evolved through versions 1 to 3, incorporating innovations like anchor boxes and feature pyramids. Glenn Jocher at Ultralytics transitioned YOLOv3 to PyTorch, resulting in the development of YOLOv5, marked by a flexible Pythonic structure and collaborative enhancements[20][21].A standout feature in YOLOv5 is the introduction of auto-learning bounding box anchors. This mechanism adapts anchor box dimensions based on the dataset's bounding box distribution, enhancing the model's adaptability. Key equations guide this process, contributing to YOLOv5's exceptional

**Table 1** Comparison of Models and Accuracy

| Models Used | Dataset | Accuracy |
|---|---|---|
| PCA and LDA [9] | The samples were hand-harvested on 23rd November 2007 from the experimental orchard in "Jin Rui" Institute of Agricultural, Zhenjiang | 94.9% |
| RGB Color Analysis Fuzzy Logic [10] | - | - |
| K-means Clustering, Graph-based Algorithms [11] | - | - |
| CNN [12] | PlantVillage dataset | 95% |
| ImageNet, DenseNet201, VGG16, Inception V3, Resnet152V2 [13] | PlantVillage dataset | 98.23% |
| Faster R-CNN [14] | Kaggle database and a collected dataset created by capturing diseased rice leaf images in the laboratory, collected by the authors from actual rice fields | 98.88% |
| Improved Yolo v5 [15] | The images were collected from a rubber plantation in Shengli State Farm, Maoming City, China | 86.5% precision for powdery mildew detection and 86.8% precision for anthracnose detection |
| Yolo v7 and GPT3 [16] | The dataset named PlantLeaf was collected in real-life scenarios by our team under the supervision of plant pathologists | 96% |
| Yolo v8, MHSA (Multi-Head Self-Attention) [17] | The dataset was collected from Shouguang Smart Agricultural Science and Technology Park in Shandong Province, China, using RGB cameras | - |

performance in handling diverse datasets.YOLOv5 combines the efficiency of one-stage target recognition with the evolutionary principles of YOLO. Originating in the Darknet framework and transitioning to PyTorch, its collaborative development under Ultralytics underscores its state-of-the-art status. The incorporation of auto-learning bounding box anchors further solidifies YOLOv5 as a formidable choice for precise and adaptable object detection across various datasets.[22][23][24][8]

## 3.2 An Overview of YOLO Training Procedures

YOLOv5 employs innovative training procedures to optimize model performance. Two key aspects are:

1)**Data Augmentation:** YOLOv5 uses sophisticated data augmentation techniques, including scaling, color space adjustments, and mosaic augmentation. Mosaic

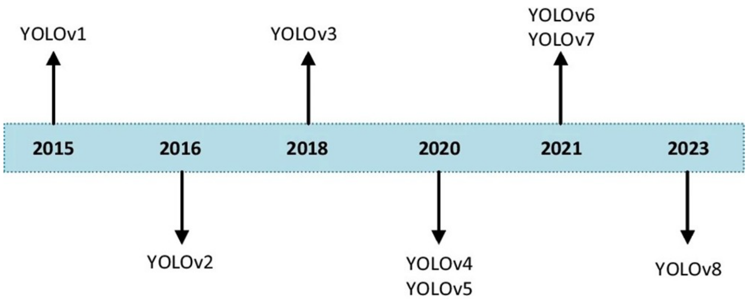


**Fig. 1** Evolution of Yolo Models

augmentation, in particular, combines multiple images into composite tiles, addressing the "small object problem" and enhancing model robustness[25][24].

2) **Loss Calculations and Loss Function:** YOLOv5 utilizes a carefully crafted loss function that unifies several critical elements, including Generalized Intersection over Union (GIoU), objectness (obj), and class losses (cls). This mathematically defined loss function is instrumental in optimizing the model during the training process, as described by the equations (1) to (4) in our paper [25].

The loss function is mathematically expressed as:

$$\text{loss} = l_{\text{box}} + l_{\text{cls}} + l_{\text{obj}} \tag{1}$$

$$\begin{aligned} l_{\text{box}} = \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{i,j}^{\text{obj}} \cdot B_j (2 - w_i \cdot h_i) \\ \times \left[ (x_i - x\Lambda_i^j)^2 + (y_i - y\Lambda_i^j)^2 + (w_i - w\Lambda_i^j)^2 + (h_i - h\Lambda_i^j)^2 \right] \end{aligned} \tag{2}$$

$$l_{\text{cls}} = \lambda_{\text{class}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{i,j}^{\text{obj}} \sum_{c \in \text{classes}} p_i(c) \log \left( p_i^{\wedge}(c) \right) \tag{3}$$

$$\begin{aligned} l_{\text{obj}} = \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{i,j}^{\text{noobj}} (c_i - c\Lambda_i)^2 \\ + \lambda_{\text{obj}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{i,j}^{\text{obj}} (c_i - c\Lambda_i)^2 \end{aligned} \tag{4}$$

Where:

$l_{\text{box}}$ represents the localization loss,

which accounts for the spatial accuracy of predicted bounding boxes.

$l_{\text{cls}}$ represents the classification loss,

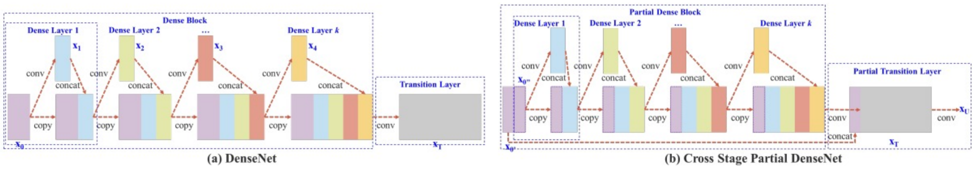


**Fig. 2** (a) Illustrates the structure of DenseNet. (b) demonstrates the proposed Cross Stage Partial DenseNet (CSPDenseNet) concept. In CSPDenseNet, the base layer's feature map is divided into two segments. One segment undergoes processing through a dense block and a transition layer, while the other segment is merged with the transmitted feature map before proceeding to the next stage.[27]

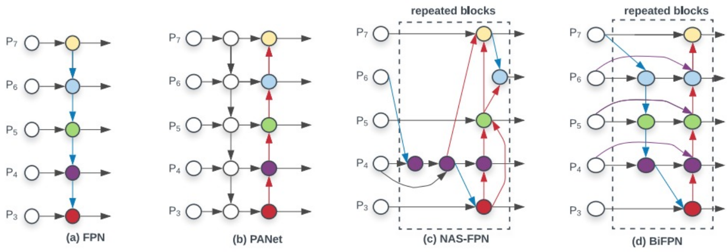


**Fig. 3** In feature network design: (a) FPN incorporates a top-down pathway for merging multi-scale features from levels 3 to 7 (P3 - P7). (b) PANet enhances FPN by adding an extra bottom-up pathway. (c) NAS-FPN employs neural architecture search to discover an unconventional feature network structure and applies it repeatedly. (d) Our approach, BiFPN, achieves improved accuracy and efficiency trade-offs compared to the others.[28]

which measures the accuracy of predicted class labels.

$l_{\text{obj}}$ is the objectness loss,

which evaluates the confidence of object predictions.

## 3.3 Architectural Details

YOLOv5 utilizes a Convolutional Neural Network (CNN) architecture comprising three essential components: the Backbone, Neck, and Head[21][24].

### 3.3.1 Backbone

The Backbone plays a pivotal role as the initial feature aggregator, meticulously extracting image features at varying granularities. YOLOv5 adopts the Cross-Stage Partial (CSP) network architecture, particularly the New CSP- Darknet53[26] structure. This architecture enhancement is further complemented by the incorporation of the spatial pyramid pooling fast (SPPF) module into the CSPDarknet53 structure[21][24]. This integration empowers YOLOv5 to excel in extracting global information crucial for accurate object detection.

### 3.3.2 Neck

The Neck component acts as a bridge between the Backbone and the prediction layer, facilitating precise object identification. YOLOv5 further optimizes this critical function by introducing the PA-Net (Path Aggregation Network) structure within the Neck. The PA-Net is responsible for intricate feature fusion and multi-scale prediction across distinct layers, which significantly strengthens the propagation of semantic features and positional information within the model[25][29].

### 3.3.3 Head

The Head component synthesizes the refined features from the Neck to produce the final output, including bounding boxes and class predictions. YOLOv5 offers four primary versions: small (s), medium (m), large (l), and extra-large (x), each progressively enhancing accuracy[30]. The model's evolution includes architectural improvements, such as the adoption of the New CSP-Darknet53 structure in the Backbone, the integration of SPPF and New CSP-PAN structures in the Neck, and the replacement of the Focus structure with a 6x6 Conv2d structure for increased efficiency. Additionally, YOLOv5 replaces the SPP structure with SPPF, effectively doubling processing speed[25][21].This model offers superior accuracy and efficiency, making it a promising tool for researchers and practitioners alike[25][21].

# 4 YOLOv8

## 4.1 Introduction

YOLOv8, a cutting-edge algorithm developed by Ultralytics, is designed for real-time crop disease detection, integrating object detection, image classification, and instance segmentation. As an evolution from the influential YOLO series, especially YOLOv5, YOLOv8 achieves a notable leap in both accuracy and efficiency within computer vision. Its versatility spans tasks like with a user-friendly Python package and command-line interface, supported by a robust expert community[31][22].Notably, YOLOv8 adopts an anchor-free detection strategy, predicting object centers directly for improved accuracy and efficiency. The inclusion of mosaic augmentation during training further underscores its commitment to innovation, exposing the model to diverse scenarios for comprehensive learning[32][18][19].YOLOv8 stands at the forefront of real-time crop disease detection, offering a powerful combination of accuracy, efficiency, and versatility, fueled by its evolutionary lineage and innovative approaches to object detection.

## 4.2 Enhanced Loss Function for Precision

YOLOv8's success lies in its meticulously crafted loss function, a critical component for fine-tuning network parameters, thus achieving superior object detection performance. This customized loss function seamlessly integrates three essential components:

• **Classification Loss:** YOLOv8 employs VFL(Varifocal Loss), an asymmetric weighting scheme that effectively balances positive and negative samples to enhance classification accuracy.

Formula:

$$VFL(p,q) = \begin{cases} -q(q\log(p) + (1-q)\log(1-p)), & \text{if } q > 0; \\ -p^{\log(1-p)}, & \text{if } q = 0. \end{cases} \tag{5}$$

• **Localization Loss:** DFL(Distribution Focal Loss) transforms single-value coordinate regression into a probability distribution centered around the target, optimizing localization accuracy.

Formula:

$$DFL(S_i, S_i + 1) = -\left((y_i + 1 - y)\log(S_i) + (y - y_i)\log(S_i + 1)\right) \tag{6}$$

• **Confidence Loss:** Complete Intersection over union (CIoU) builds upon Distance Intersection over Union loss (DIoU), incorporating an additional influence factor to enhance object localization precision.

Formula:

$$CIoULoss = 1 - CIoU = 1 - IoU + \frac{d^2}{c^2 + \varepsilon} + v^2(1 - IoU) \tag{7}$$

These integrated components collectively contribute to YOLOv8's precision, enabling it to focus on target areas and achieve more accurate and reliable object detection, a crucial aspect for research and practical applications[19][22].

## 4.3 Architectural Details

In this section, we delve into the architectural refinements that underscore YOLOv8's efficacy in object detection. These strategic changes are meticulously designed to enhance gradient flow, improve computational efficiency, and elevate overall model performance while staying true to the model's foundational strengths[23].

### 4.3.1 Backbone Optimization

One noteworthy enhancement within the backbone involves the replacement of a 6x6 convolution in the stem with a more efficient 3x3 convolution. This adaptation aligns YOLOv8 with contemporary computational efficiency paradigms, streamlining its feature extraction process[32][18].

### 4.3.2 C2f Enhancement

YOLOv8 introduces a novel approach in its architecture, termed C2f, as a replacement for its predecessor, C3. Within C2f, outputs from two 3x3 convolutions with residual connections are concatenated. This innovative alteration responds to evolving research findings and effectively optimizes feature representation[32][18].

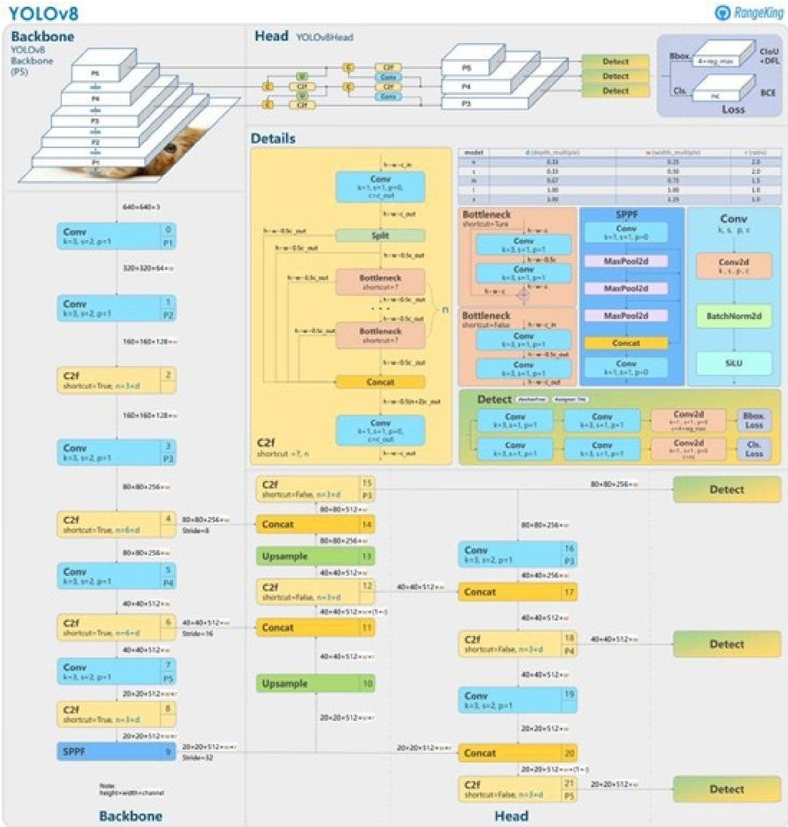


**Fig. 4** Architecture of Yolov8[32]

### 4.3.3 Detection Head Transformation

While YOLOv8 retains the foundational Bottleneck architecture from YOLOv5, it implements a pivotal change by modifying the first convolution's kernel size from 1x1 to 3x3. This strategic adjustment aligns YOLOv8 with the ResNet block design from 2015, showcasing its adaptability and responsiveness to architectural evolution[32][18].

## 4.4 Classification and Detection Separation

YOLOv8 introduces a structural refinement by separating the classification and detection heads. This architectural leap enhances the model's sophistication and positions it as a formidable asset for object detection, delivering both precision and efficiency[18].

These architectural refinements play a pivotal role in YOLOv8's success, making it invaluable for research and practical applications.

Model Versions: YOLOv8 is available in four primary versions: small (s), medium (m), large (l), and extra-large (x), each progressively enhancing accuracy. YOLOv8 is a remarkable milestone in the field of object detection and computer vision, especially for plant leaf disease detection.

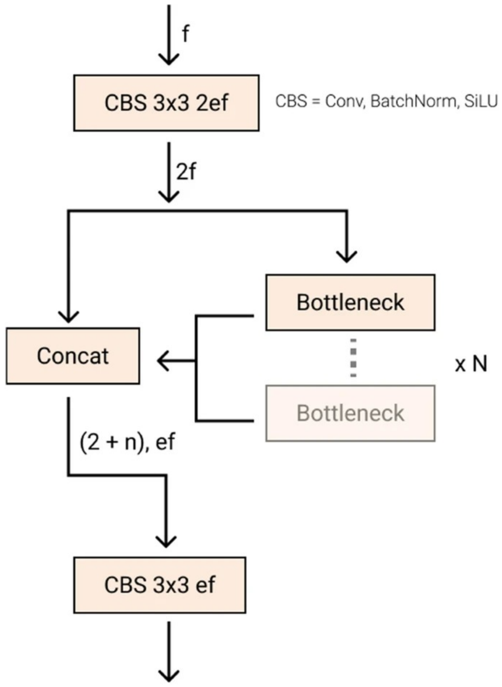


**Fig. 5** YOLOv8 C2f module[32]

# 5 Material and Methods

In this section, we provide a comprehensive account of our implementation process for training and evaluating YOLOv5 and YOLOv8 models for plant disease recognition. Our methodology encompasses the following steps:

## 5.1 Dataset Acquisition

We sourced our dataset from Kaggle, known as the "Plant disease recognition dataset." This dataset consists of 1530 images, each meticulously labeled with one of three distinct conditions: "Healthy," "Powdery," and "Rust." To elucidate these conditions, "Healthy" signifies plant leaves without any disease symptoms, while "Powdery" denotes leaves affected by powdery mildew, characterized by a white, powdery growth on the leaf surface, and "Rust" indicates leaves afflicted by rust disease, typically manifesting as reddish-brown or orange pustules on the leaf surface. Subsequently, we partitioned the dataset into three sets: the training set (comprising 60 percent of the total images), the validation set (30 percent), and the test set (10 percent).

## 5.2 Custom Annotation for Disease Spots

To refine the granularity of our dataset and prepare it for effective training with YOLOv5 and YOLOv8, we implemented custom annotation. We employed the Roboflow custom annotation software for this purpose. The primary objective was to pinpoint the precise regions of interest (ROIs) on plant leaves where diseases such as ”Powdery” and ”Rust” were present. In addition to manual annotation, we harnessed Roboflow Label Assist, a tool that harnesses model checkpoints from a previous version of the model to recommend annotations, ensuring that our dataset was labeled with high precision.

### 5.3 Model Training

Armed with our meticulously annotated dataset, we embarked on the training of both YOLOv5 and YOLOv8 models. This training was conducted within Google Colab notebooks, leveraging the platform's Graphics processing unit(GPU) capabilities for swift and efficient model training with carefully configuring the models, and specifying hyperparameters, network architectures, and training schedules. This encompassed the selection of appropriate loss functions, optimizers, and learning rates. The enriched dataset, fortified with custom annotations, was then partitioned into training, validation, and test subsets, adhering to the predetermined ratios. The models underwent iterative training on the annotated dataset, with weights fine-tuned to optimize performance. The training process spanned multiple epochs to ensure convergence.

### 5.4 Experimental Setup

In our research, we conducted model training on Google Colab, harnessing the provided GPU and Central Processing unit(CPU) resources. Google Colab typically offers access to a high-performance GPU, particularly the NVIDIA Tesla T4, for accelerated model training. The YOLOv5 model was trained with 100 epochs, a batch size of 16, and an image size of 416 pixels. Similarly, for the YOLOv8 model, we utilized the same GPU, with 20 epochs, a batch size of 16, and an image size of 100 pixels. Furthermore, it's important to note that our Google Colab notebook was executed on a MacBook Pro M1 with 512GB storage and 8GB RAM, providing a stable and resourceful environment for our experiments.

## 6 Results

This section presents a comprehensive evaluation of YOLOv5 and YOLOv8 models for plant leaf disease detection. Our investigation begins with training YOLOv5 and YOLOV8 models on our custom annotated dataset. YOLOv5 was trained on 100 epochs, while YOLOv8 was trained on 20 epochs. We then evaluate the model performance in terms of various metrics, such as confusion matrices, F1-confidence curves, precision-recall curves, accuracy metrics, mAP, and losses in the bounding box.

### 6.1 Image Analysis

To provide a visual context for our results, we have conducted image analysis, generating visual representations of YOLOv5 and YOLOv8's detection capabilities. These images showcase bounding boxes that denote the location and classification of crop diseases. The top-left corner of each bounding box contains the label for the detected

**Fig. 6** YOLOv5 Detects Healthy Object with Precise Bounding Box


**Fig. 7** YOLOv8 Accurately Combines Class Images with Bounding Boxes

class. In Figure 6 and 7, we present these images to help readers better understand the models performance.

## 6.2 Confusion Matrix

A confusion matrix is a table used in machine learning and classification tasks to evaluate the performance of a classification model. It helps us understand how well the model is performing by showing the number of true positives, true negatives, false positives, and false negatives for each class in the dataset. Let's break down the confusion matrices for YOLOv5 and YOLOv8 and compare their performance for the different classes: Healthy, Powdery, Rust, and Background.

1) **Healthy:**

• YOLOv8 has a higher true positive rate (0.71) compared to YOLOv5 (0.62), indicating that it is better at correctly identifying healthy instances.

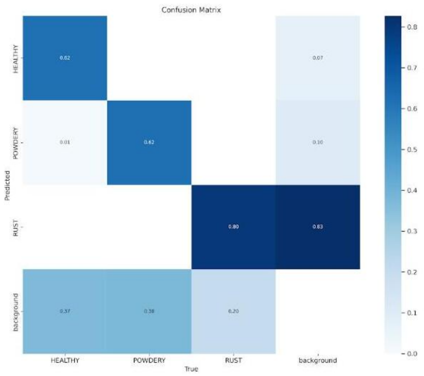


**Fig. 8** YOLOv5 confusion matrix

• YOLOv5 has a lower false positive rate (0.01) compared to YOLOv8 (0.24), indicating that it makes fewer incorrect predictions for the healthy class.

2) **Powdery:**

• YOLOv8 has a higher true positive rate (0.70) compared to YOLOv5 (0.62), indicating that it is better at correctly identifying powdery instances.

• YOLOv5 has a slightly lower false positive rate (0.04) compared to YOLOv8 (0.21) for the Powdery class.

3) **Rust:**

• YOLOv5 has a higher true positive rate for the Rust class (0.80) compared to YOLOv8 (0.74), indicating that it is better at correctly identifying instances of rust.

• YOLOv5 also has a higher false positive rate for the Rust class (0.94) compared to YOLOv8 (0.55), indicating that it makes more incorrect predictions for this class.

4) **Background:**

• YOLOv8 has a higher true positive rate (0.55) for the Background class compared to YOLOv5 (0.21), indicating that it is better at identifying the background class.

• YOLOv5 has a lower false positive rate (0.02) compared to YOLOv8 (0.29) for the Background class.

YOLOv8 performs better than YOLOv5 in correctly identifying instances of the Healthy and Background classes. YOLOv5 performs slightly better in the Rust class in terms of true positives but has a higher false positive rate. The performance for the Powdery class is quite similar between the two models. Ultimately, the choice between the models depends on the specific priorities of your application and whether you value precision (fewer false positives) or recall (higher true positives) for different classes.

In Fig 8 and 9, we present the confusion matrices for both YOLOv5 and YOLOv8. These matrices provide a detailed breakdown of the models' predictions across different disease classes, namely Healthy, Powdery, Rust, and the Background class. These visual representations allow readers to observe how the models perform in distinguishing between different crop diseases, further enhancing their understanding of the models' strengths and areas for improvement.

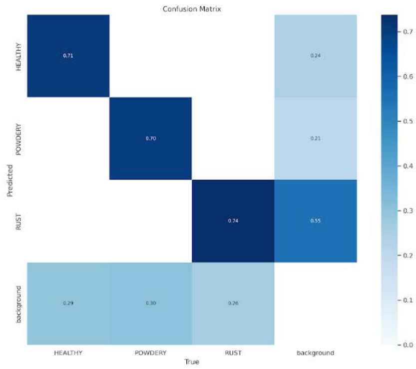


**Fig. 9** YOLOv8 confusion matrix

## 6.3 Precision and Recall Curve

In our pursuit to assess the effectiveness of YOLOv5 and YOLOv8 for crop disease detection, it is essential to consider key evaluation metrics that provide a comprehensive view of their performance. This section discusses the Precision-Recall (PR) Curve highlighting its significance in evaluating detection capabilities.

Precision and Recall are fundamental metrics in object detection. Precision quantifies the ratio of true positives (instances correctly identified) to the sum of true positives and false positives (instances incorrectly identified). Recall measures the ratio of true positives to the sum of true positives and false negatives (instances of the class not detected). The formulas for Precision and Recall are as follows:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{8}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{9}$$

Precision-Recall Curve, presented in Figure 10 and 11, provides a visual representation of the trade-off between precision and recall as the confidence threshold varies. It enables a detailed analysis of how different confidence thresholds impact the models' precision-recall dynamics. The area under this curve for each class, known as Average Precision (AP), quantifies the precision-recall trade-off for that class.

## 6.4 Mean Average Precision (mAP)

It takes the evaluation a step further. It is calculated as the average of the AP values computed for each class. The formula for calculating mAP is as follows:

$$mAP = \frac{1}{n}\sum_{i=1}^{n} AP_i \tag{10}$$

Here, 'n' represents the number of classes, and denotes the Average Precision for class 'i'.

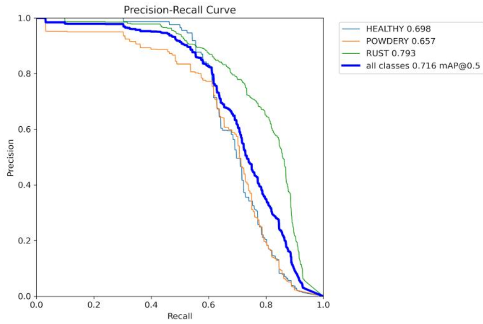


**Fig. 10** YOLOv5 Precision-Recall Curve

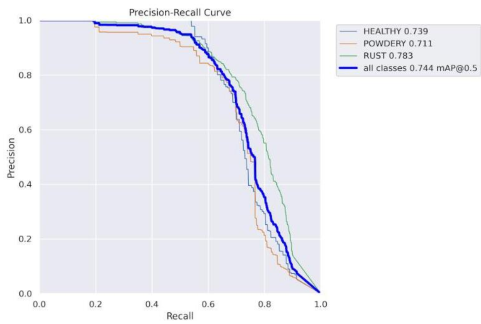


**Fig. 11** YOLOv8 Precision-Recall Curve

The significance of mAP lies in its ability to offer a consolidated measure of the models' detection capabilities across all classes and confidence thresholds. It provides a comprehensive assessment of the overall performance. For instance, YOLOv5 achieved a mean average precision of 0.716 at a confidence threshold of 0.5, indicating strong overall detection performance. In comparison, YOLOv8 surpassed it with a mean average precision of 0.744 at the same threshold, reflecting even better detection capabilities.

By examining the PR curve and calculating mAP, we gain valuable insights into the precision-recall dynamics of our models, enabling us to assess their ability to detect and classify crop diseases accurately. These metrics are pivotal for understanding the trade-offs between precision and recall, ultimately contributing to the optimization of model performance for practical applications.

## 6.5 F1-Confidence Curve

In our comparative study of YOLOv5 and YOLOv8 for object detection, we evaluated model performance using the F1 score. The F1 score is a metric that combines precision and recall, calculated as the harmonic mean of these two measures. Precision represents the ratio of true positive predictions to all positive predictions, while recall represents the ratio of true positive predictions to all actual positive instances. The F1 score

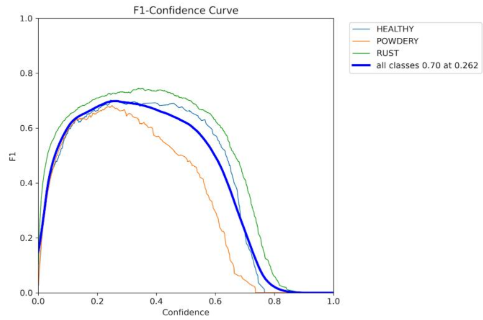


**Fig. 12** YOLOv5 F1-Confidence Curve

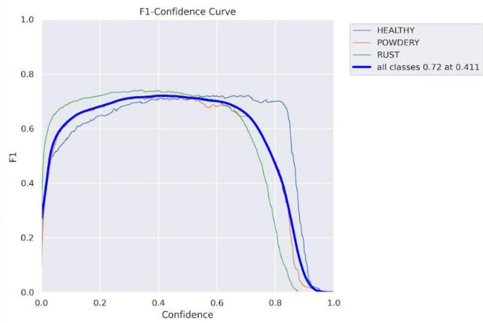


**Fig. 13** YOLOv8 F1-Confidence Curve

provides a balanced assessment of classification performance, taking into account both false positives and false negatives

The F1 score, a balanced metric considering both precision and recall, is calculated using the formula:

$$F1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{11}$$

To elucidate the relationship between confidence thresholds and F1 scores, we present F1-confidence curves in Figure 12,13. These curves illustrate how F1 scores change as the confidence threshold varies. For YOLOv5, all classes achieve an F1 score of 0.70 when the confidence threshold is set at 0.262.

Similarly, for YOLOv8, all classes attain an F1 score of 0.72 at a confidence threshold of 0.411. These curves help readers comprehend how different confidence thresholds impact the models' precision and recall.

## 6.6 Training Progress and Loss Analysis

In our comparative research evaluating the performance of YOLOv5 and YOLOv8 for crop disease detection, a pivotal aspect of our investigation involved a meticulous examination of the training progress and loss trends exhibited by these models. The insights garnered from this analysis shed light on the learning dynamics and convergence rates of the two models, providing valuable information for the effective deployment of crop disease detection systems.

### 6.6.1 Loss Trends

We scrutinized several key loss values throughout the training process, each of which played a distinctive role in assessing the models' capabilities. The loss components included:

•**Bounding Box and Segmentation Loss:** We monitored the Training Box Loss (train/box loss) and Validation Box Loss (val/box loss), which quantify the alignment of predicted bounding boxes with ground truth coordinates.

•**Classification Loss:** The Training Class Loss (train/cls loss) and Validation Class Loss (val/cls loss) were pivotal in determining the models classification capabilities. These metrics indicated how effectively each model predicted the correct class for a given image, providing insights into their classification accuracy.

•**Objectness Loss:** The Objectness Loss (Obj loss)assesses how accurately the model predicts whether an object is present or absent within a given bounding box. This loss penalizes the model for misclassifying the presence or absence of an object in a box, helping to refine the model's ability to determine object locations and improve overall object detection performance.

### 6.6.2 Interpreting Loss Trends

The evolution of these loss metrics over the training epochs unveiled compelling insights into the convergence dynamics of YOLOv5 and YOLOv8. Notably, the Training Box Loss exhibited a consistent downward trend, signifying that both models adeptly optimized their ability to predict crop disease locations with increasing training. This trend underscored the models' proficiency in aligning predicted bounding box coordinates with ground truth coordinates.

Similarly, the Validation Box Loss followed distinct trajectories. This loss trend analysis was pivotal in understanding the pace at which Validation Box Loss decreased. Rapid descent indicated swift model convergence, suggesting that the models quickly adapted to the data. In contrast, a gradual decline implied a more cautious learning process, potentially necessitating adjustments to hyperparameters or model architectures.Images related to the analysis are available for reference in Figure 14 and 15.

A visual examination of the indicators used to assess our YOlOv5 and YOLOv8 model performances during training, covering model iterations from 0 to 100 for YOlOv5 and 0 to 20 for YOLOv8.

In conclusion, our in-depth analysis of the training progress and loss trends of YOLOv5 and YOLOv8 serves as a cornerstone in understanding the capabilities of these models for crop disease detection. These findings enable data-driven decisions in model selection and refinement, ensuring the effective deployment of crop disease detection systems in agricultural applications. This systematic evaluation of loss trends underscores the importance of considering multiple loss components to gain comprehensive insights into model performance during training.

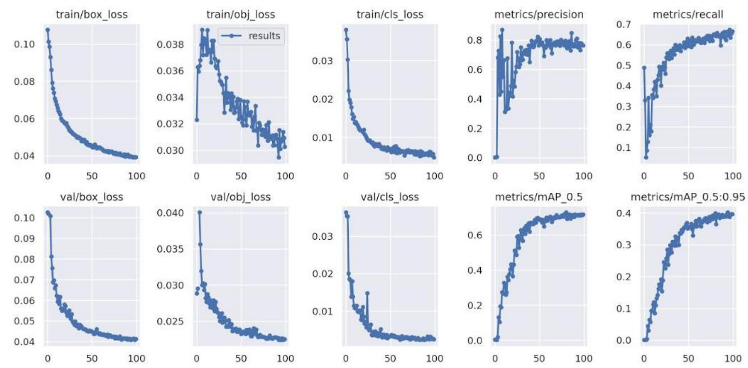


**Fig. 14** YOLOv5 overall analysis

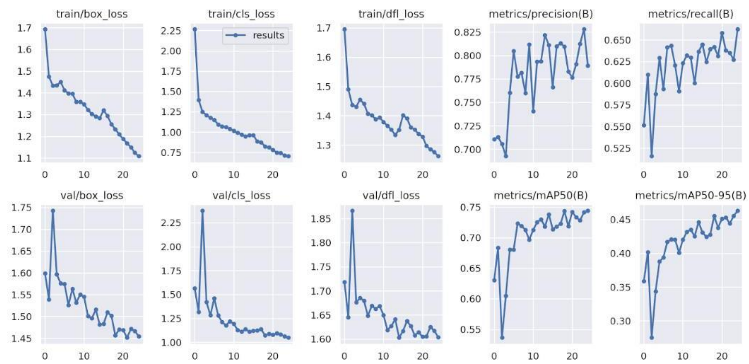


**Fig. 15** YOLOv8 overall analysis

# 7 Discussion

## 7.1 Model Comparison and Analysis

Our comparative analysis of YOLOv5 and YOLOv8 reveals the strong detection capabilities of both models. Notably, YOLOv8 exhibits better performance in distinguishing between various crop diseases, as indicated by the confusion matrices and higher F1 scores. Due to the adoption of an anchor-free detection and newly optimized architecture approach in YOLOv8, it took fewer epochs to reach a higher mAP which as compared to YOLOv5 though YOLOv5 took 8 hours to train for 100 epochs and YOLOv8 took 4 hours for just 25 epochs YOLOv8 training took more time though YOLOv8 less time than it would due to switching of mosaic augmentation at last 10 epochs. However, it's essential to acknowledge some instances of misclassification in both models, indicating opportunities for further refinement. Finally, the adoption of an anchor-free detection approach in YOLOv8 further enhances its robustness and computational efficiency, making it a compelling choice for real-world deployment. A proposed model with a more specific dataset containing more categories of plants and diseases is needed to make the plant disease detector more accurate and better.

# 8 Conclusion

Our paper delves into the remarkable potential of YOLOv5 and YOLOv8 to revolutionize crop disease detection within the agricultural sector, leveraging a custom annotated dataset. Among these models, YOLOv8 emerges as the standout performer, boasting exceptional accuracy, precision, recall, and class differentiation. Its real-time processing capabilities, anchor-free approach, customized loss functions, and architectural refinements position it as the leading choice for object detection in this domain and its related applications.

Through rigorous experimentation, we consistently demonstrate that both YOLOv8 and YOLOv5 achieve an F1 score of 0.70 when the confidence threshold is set at 0.262, utilizing our meticulously curated custom annotated dataset. Furthermore, for YOLOv8, all classes attain an F1 score of 0.72 at a confidence threshold of 0.411, reinforcing the effectiveness of our approach. These performance curves offer valuable insights into how different confidence thresholds impact the models' precision and recall. This robust performance underscores their pivotal role in addressing the urgent challenge of crop disease detection, providing invaluable tools for farmers and researchers alike.

Our paper makes a significant contribution to the ever-expanding field of deep learning-based agricultural applications, offering novel insights and laying the groundwork for further exploration and advancement in this critical area. Beyond academia, the implications of our research are profound, as it has the potential to transform agricultural technology, promote sustainable crop management practices, and bolster global food security. In essence, our work represents a pivotal step forward in harnessing cutting-edge technology, alongside our custom annotated dataset, for the betterment of agriculture and, by extension, society as a whole. Statements and Declarations

# 9 Statements and Declarations:

## 9.1 Competing Interests:

The authors declare that there are no conflicts of interest, financial or non-financial, that could be perceived as influencing the research or that could potentially bias its outcomes. This includes, but is not limited to, employment, consultancies, funding, honoraria, patents, or shared affiliations that may have an impact on the study's integrity or the interpretation of results. All authors have no known or potential competing interests associated with this research. This transparency ensures that the research remains impartial and unbiased in its pursuit of knowledge. Funding Information: This research was conducted without any external funding or financial support. No grants, scholarships, or financial contributions were received for this study.

# References

[1] Morbekar, A., Parihar, A., Jadhav, R.: Crop disease detection using yolo. In: 2020 International Conference for Emerging Technology (INCET), pp. 1–5 (2020).

https://doi.org/10.1109/INCET49848.2020.9153986

[2] Alirezazadeh, P., Schirrmann, M., Stolzenburg, F.: A comparative analysis of deep learning methods for weed classification of high-resolution uav images. Journal of Plant Diseases and Protection (2023) https://doi.org/10.1007/s41348-023-00814-9

[3] Martinelli, F., Scalenghe, R., Davino, S., *et al.*: Advanced methods of plant disease detection. a review. Agronomy for Sustainable Development **35**, 1–25 (2015) https://doi.org/10.1007/s13593-014-0246-1

[4] Nagasubramanian, K., Jones, S., Singh, A.K., *et al.*: Plant disease identification using explainable 3d deep learning on hyperspectral images. Plant Methods **15**, 98 (2019) https://doi.org/10.1186/s13007-019-0479-8

[5] Fang, Y., Ramasamy, R.P.: Current and prospective methods for plant disease detection. Biosensors **5**, 537–561 (2015) https://doi.org/10.3390/bios5030537

[6] Slimani, H., Mhamdi, J.E., Jilbab, A.: Artificial intelligence-based detection of fava bean rust disease in agricultural settings: An innovative approach. International Journal of Advanced Computer Science and Applications **14**(6) (2023) https://doi.org/10.14569/IJACSA.2023.0140614

[7] Orchi, H., Sadik, M., Khaldoun, M., Sabir, E.: Real-time detection of crop leaf diseases using enhanced yolov8 algorithm. In: 2023 International Wireless Communications and Mobile Computing (IWCMC), pp. 1690–1696 (2023). https://doi.org/10.1109/IWCMC58020.2023.10182573

[8] Wang, Q., Liu, C., Xia, X., Guo, Y., Men, H.: Classification and identification of crop disease based on depthwise separable group convolution and feature fusion. Journal of Plant Diseases and Protection (2023) https://doi.org/10.1007/s41348-023-00826-5

[9] Zhang, Y., Yin, X., Xu, T., Zhao, J.: On-line sorting maturity of cherry tomato by machine vision. In: Zhao, C., Li, D. (eds.) Computer and Computing Technologies in Agriculture II, Volume 3, pp. 2223–2229. Springer, Boston, MA (2009)

[10] Dadwal, M.: Estimate Ripeness Level of Fruits Using RGB Color Space and Fuzzy Logic Technique, (2012). https://api.semanticscholar.org/CorpusID:14799723

[11] Pham, H., Lee, B.-R.: An image segmentation approach for fruit defect detection using k-means clustering and graph-based algorithm. Vietnam Journal of Computer Science **2**, 25–33 (2014) https://doi.org/10.1007/s40595-014-0028-3

[12] Chohan, M., Khan, A., Chohan, R., Katper, S., Mahar, M.: Plant disease detection using deep learning. International Journal of Recent Technology and Engineering **9**, 909–914 (2020) https://doi.org/10.35940/ijrte.A2139.059120

[13] Bakr, M., Abdel-Gaber, S., Nasr, M., Hazman, M.: Densenet based model for plant diseases diagnosis. European Journal of Electrical Engineering and Computer Science **6**(5), 1–9 (2022) https://doi.org/10.24018/ejece.2022.6.5.458 . https://www.ejece.org/index.php/ejece/article/download/458/280

[14] Bari, B.S., Islam, N., Rashid, M., Hasan, J., Razman, M.A.M., Musa, R.M., Nasir, A.F.A., Majeed, A.P.P.A.: A real-time approach of diagnosing rice leaf disease using deep learning-based faster r-cnn framework. PeerJ Computer Science **7**, 432 (2021) https://doi.org/10.7717/peerj-cs.432 . https://doi.org/10.7717/peerj-cs.432

[15] Chen, Z., Wu, R., Lin, Y., Li, C., Chen, S., Yuan, Z., Chen, S., Zou, X.: Plant disease recognition model based on improved yolov5. Agronomy **12**(2) (2022) https://doi.org/10.3390/agronomy12020365

[16] P, S., John, A., Devika, V.L., V, G.S., Sakhir, N.: Leaf disease detection & correction using yolo v7 with gpt3 integratedccccc. INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) **12** (2023)

[17] Li, P., Zheng, J., Li, P., Long, H., Li, M., Gao, L.: Tomato maturity detection and counting model based on mhsa-yolov8. Sensors **23**(15) (2023) https://doi.org/10.3390/s23156701

[18] Tamang, S., Sen, B., Pradhan, A., Sharma, K., Singh, V.K.: Enhancing covid-19 safety: Exploring yolov8 object detection for accurate face mask classification. International Journal of Intelligent Systems and Applications in Engineering **11**(2), 892–897 (2023)

[19] Soylu, E., Soylu, T.: A performance comparison of yolov8 models for traffic sign detection in the robotaxi-full scale autonomous vehicle competition. Multimedia Tools and Applications (2023) https://doi.org/10.1007/s11042-023-16451-1

[20] Xiang, Y., Yao, J., Yang, Y., Yao, K., Wu, C., Yue, X., Li, Z., Ma, M., Zhang, J., Gong, G.: Real-time detection algorithm for kiwifruit canker based on a lightweight and efficient generative adversarial network. Plants **12**, 3053 (2023) https://doi.org/10.3390/plants12173053

[21] Dai, G., Fan, J.: An industrial-grade solution for crop disease image detection tasks. Frontiers in Plant Science **13**, 921057 (2022) https://doi.org/10.3389/fpls.2022.921057 . https://www.frontiersin.org/articles/10.3389/fpls.2022.921057/pdf

[22] Xiao, B., Nguyen, M., Yan, W.Q.: Fruit ripeness identification using yolov8 model. Multimedia Tools and Applications (2023) https://doi.org/10.1007/s11042-023-16570-9

[23] Aboah, A., Wang, B., Bagci, U., Adu-Gyamfi, Y.: Real-time multi-class helmet violation detection using few-shot data sampling technique and yolov8. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5349–5357 (2023)

[24] Zhao, Q., Liu, B., Lyu, S., Wang, C., Zhang, H.: Tph-yolov5++: Boosting object detection on drone-captured scenarios with cross-layer asymmetric transformer. Remote Sensing **15**, 1687 (2023) https://doi.org/10.3390/rs15061687

[25] Chen, R., Chen, Y.: Improved convolutional neural network yolov5 for underwater target detection based on autonomous underwater helicopter. Journal of Marine Science and Engineering **11**, 989 (2023) https://doi.org/10.3390/jmse11050989

[26] Redmon, J.: Darknet: Open source neural networks in c (2013)

[27] Wang, C.-Y., Liao, H.-Y.M., Wu, Y.-H., Chen, P.-Y., Hsieh, J.-W., Yeh, I.-H.: Cspnet: A new backbone that can enhance learning capability of cnn. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 390–391 (2020)

[28] Tan, M., Pang, R., Le, Q.V.: Efficientdet: Scalable and efficient object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10781–10790 (2020)

[29] Liu, H., Sun, F., Gu, J., Deng, L.: Sf-yolov5: A lightweight small object detection algorithm based on improved feature fusion mode. Sensors **22**, 5817 (2022) https://doi.org/10.3390/s22155817

[30] Chen, Z., Wu, R., Lin, Y., Li, C., Chen, S., Yuan, Z., Chen, S., Zou, X.: Plant disease recognition model based on improved yolov5. Agronomy **12**(2), 365 (2022) https://doi.org/10.3390/agronomy12020365

[31] Qadri, S.A.A., Huang, N.-F., Wani, T.M., Bhat, S.A.: Plant disease detection and segmentation using end-to-end yolov8: A comprehensive approach. In: 2023 IEEE 13th International Conference on Control System, Computing and Engineering (ICCSCE), pp. 155–160 (2023). https://doi.org/10.1109/ICCSCE58721.2023.10237169

[32] Talaat, F.M., ZainEldin, H.: An improved fire detection approach based on yolo-v8 for smart cities. Neural Computing & Applications **35**, 20939–20954 (2023) https://doi.org/10.1007/s00521-023-08809-1