



OPEN ACCESS

EDITED BY

Lei Shu,
Nanjing Agricultural University, China

REVIEWED BY

Sangeetha T. Sri Krishna College of
Technology,
India
Matthew Oladipupo,
University of Salford, United Kingdom

*CORRESPONDENCE

Suresh Gawande

✉ Suresh.Gawande@icar.gov.in

RECEIVED 26 December 2024

ACCEPTED 18 April 2025

PUBLISHED 22 May 2025

CITATION

Raj A, Dawale M, Wayal S, Khandagale K,
Bhangare I, Banerjee S, Gajarushi A,
Velmurugan R, Baghini MS and Gawande S
(2025) YOLO-ODD: an improved YOLOv8s
model for onion foliar disease detection.
Front. Plant Sci. 16:1551794.
doi: 10.3389/fpls.2025.1551794

COPYRIGHT

© 2025 Raj, Dawale, Wayal, Khandagale,
Bhangare, Banerjee, Gajarushi, Velmurugan,
Baghini and Gawande. This is an open-access
article distributed under the terms of the
[Creative Commons Attribution License \(CC BY\)](#).
The use, distribution or reproduction in other
forums is permitted, provided the original
author(s) and the copyright owner(s) are
credited and that the original publication in
this journal is cited, in accordance with
accepted academic practice. No use,
distribution or reproduction is permitted
which does not comply with these terms.

YOLO-ODD: an improved YOLOv8s model for onion foliar disease detection

Anusha Raj¹, Mukund Dawale¹, Sagar Wayal¹,
Kiran Khandagale¹, Indira Bhangare¹, Susmita Banerjee²,
Ashwini Gajarushi², Rajbabu Velmurugan³,
Maryam Shojaei Baghini³ and Suresh Gawande^{1*}

¹Indian Council of Agricultural Research (ICAR)-Directorate of Onion and Garlic Research, Pune, India,

²TIH Foundation for Technology Innovation Hub, Internet of Things and Internet of Everything (IoT & IoE), Mumbai, India, ³Indian Institute of Technology, Bombay, India

Onion crops are affected by many diseases at different stages of growth, resulting in significant yield loss. The early detection of diseases helps in the timely incorporation of management practices, thereby reducing yield losses. However, the manual identification of plant diseases requires considerable effort and is prone to mistakes. Thus, adopting cutting-edge technologies such as machine learning (ML) and deep learning (DL) can help overcome these difficulties by enabling the early detection of plant diseases. This study presents a cross layer integration of YOLOv8 architecture for detection of onion leaf diseases viz. anthracnose, Stemphylium blight, purple blotch (PB), and Twister disease. The experimental results demonstrate that customized YOLOv8 model YOLO-ODD integrated with CABM and DTAH attentions outperform YOLOv5 and YOLO v8 base models in most disease categories, particularly in detecting Anthracnose, Purple Blotch, and Twister disease. Proposed YOLOv8 model achieved the highest overall 77.30% accuracy, 81.50% precision and Recall of 72.10% and thus YOLOv8-based deep learning approach will detect and classify major onion foliar diseases while optimizing for accuracy, real-time application, and adaptability in diverse field conditions.

KEYWORDS

artificial intelligence, disease detection, deep learning, YOLOv8, image annotation, onion

1 Introduction

Onion (*Allium cepa* L.) is an important vegetable crop consumed. Onion crop suffer from multiple diseases throughout its life cycle. The incidence of various diseases significantly affects the yield and quality of onion crop. Pests and diseases collectively cause 30-50% bulb yield losses (Larentzaki et al., 2007; Nault and Shelton, 2010). Timely and accurate identification of plant diseases is of great significance for protecting crop

safety and controlling the spread of diseases (Iqbal et al., 2018). The present method of disease identification is based on visual inspection of symptoms by plant protection experts. The accuracy of this identification is based on the experience and skill levels of the expert. To increase the accuracy of disease detection, several researchers have built disease detection models to classify and identify plant diseases using image processing techniques, machine learning, and deep learning algorithms.

Previous reports on plant disease detection in various cropping scenarios are briefed here, Fuentes et al. (2017) used “deep learning meta-architectures” to identify nine different kinds of pests and diseases in tomato plants using pictures taken in camera at different resolutions. In another study, Cassava leaf diseases were detected using a Convolutional Neural Networks (CNN) model, trained with field image dataset (Sambasivam and Opiyo, 2021). Pre-trained transfer learning techniques such as AlexNet and GoogleNet were applied to identify soybean diseases (Jadhav et al., 2021). The CNN model was created to classify purple blotch disease in onion crop by employing a pre-trained InceptionV3 model (Zaki et al., 2021). A deep neural network model proposed to automatically detect onion downy mildew using pictures taken periodically by a field-monitoring system of onion fields (Kim et al., 2020).

When evaluated on the tomato leaf diseases dataset, the YOLOv5 model exhibited an impressive accuracy rate of 93% (Rajamohanan and Latha, 2023). YOLOv8 has been used to increase the efficiency and precision of rust disease classification in fava bean fields (Slimani et al., 2023). The application of the YOLO model, specifically YOLOv8 and YOLOv9, for the identification of plant diseases in a hydroponic setting, was examined. The findings demonstrated that YOLOv9 outperformed YOLOv8 by a small margin in terms of detection accuracy, with 88.38% and 87.22%, respectively. For real-time plant disease detection, YOLOv8 uses less time and processing resources than YOLOv9 (Tripathi et al., 2024). A combination of YOLOv5 and YOLOv8 models was more effective for the same diseases, achieving a detection accuracy between 86.6% and 94.3%, according to Ahmed and Abd-Elkawy (2024). Further, the diseases tomato splitting, sun-scaled rot, and blossom end rot were identified with a 93.6% accuracy using YOLOv8 and YOLOv9 models, as noted by Zayani et al. (2024). Blossom end rot, splitting, and sun-scaled rot were detected using the YOLOv8 model with an accuracy of 66.67%, as reported by Iren (2024).

Deep Learning (DL) has opened new avenues for automatic plant disease identification through object detection and image classification. There are two major categories of object detection models which differs in architecture, time needed, dataset needed and accuracy. 1) Two stage Models- Fast R-CNN, Faster R-CNN, Mask R-CNN and 2) one stage includes, YOLO (You Only Look Once), Retina Net etc.

In this study, we propose cross layer integration of YOLOv8 model for improved accuracy in disease detection. YOLO is a real-time object detection method that uses a neural network to process a picture in a single forward pass. YOLO completes bounding box regression and object recognition in a single step, in contrast to

conventional object detection techniques, which require several processing steps. Multiple versions of YOLO have been created, from YOLOv1 to the recently developed YOLOv11. Every new version builds upon previous versions, offering more features, including increased precision, quicker processing, and better object handling. YOLOv8 is a cutting-edge model for object recognition that evaluates photographs in the range of 40–155 frames per second (FPS) depending on the configuration. This novel method of image analysis divides an image into several grid cells and predicts the bounding box coordinates and class probabilities using a single neural network. This results in a faster and more precise disease identification process by enabling more efficient and accurate image assessment (Orchi et al., 2023). Pretrained object detection models are available in different repositories these models are trained on large datasets however training from scratch on custom dataset has added advantage of avoiding possible learning bias due to objective function difference and limited design space on networks (Shen et al., 2025).

Grad-CAM visualization technique used to highlight important regions in image which enhances model trust and accuracy. AI-driven deep learning models have been widely used for diagnosing plant diseases in major crops, including rice, wheat, maize, tomato, banana, apple, grapes, citrus, mango, tea, cucumber, cassava, ginger, sugarcane, papaya, common bean, and pearl millet. Many of these studies rely on the PlantVillage dataset, which includes only a limited selection of diseases from specific crops. However, onion diseases are not covered in the PlantVillage database. Onion pathogens and symptoms are unique, and only a few studies have explored image-based detection, typically focusing on just one or two diseases. While Jahan M. et al. (2024) conducted a comprehensive study on onion leaf disease classification and hierarchical image feature extraction, no credible image-based detection model currently exists for onion disease identification.

Currently, onion disease diagnosis depends on human expertise, with accuracy varying based on the specialist's experience and skill level. To address this gap, we aim to develop an automated image-based object detection model for onion disease diagnosis, designed for deployment in a mobile camera-based application. This study aims to authenticate a robust onion disease detection model by analyzing the effectiveness of YOLOv5, YOLOv8, and customized YOLOv8 models using different attention modules in detecting onion diseases and evaluate their performance across different disease categories.

This paper addresses the need of a robust digital guide for realtime assistance to onion growers upon integration of this model into smartphone app (iSARATHI) (<https://play.google.com/store/apps/category/FAMILY?hl=en-US>). The model was optimized by integrating CBAM attention after C2f in neck and backbone network in addition dynamic task align head was replaced with detection head in YOLOv8 architecture which enhances feature extraction especially for smaller disease spot detection. The findings of this study will contribute to advancements in precision agriculture and smart disease management systems for onions.

1.1 Brief overview of related research

Leaf disease recognition has been extensively studied using various computer vision techniques, including traditional machine learning, deep learning-based CNNs and recent advancements in YOLO architecture. Most research efforts focus on image classification and bounding box-based detection using existing datasets. However, complex leaf diseases often have irregular shapes that bounding boxes cannot effectively capture. To address this, we integrated bounding box-based annotations utilizing real-time field data collected directly from agriculture research field plots which were then trained with customized YOLOv8, making it a more reliable and practical solution for farmers and agronomists. An illustration of previous modalities utilized for plant disease recognition enabled by object detection algorithms is described in [Table 1](#).

This table explains both parallels and contrasts among a subset of previously built plant disease detection models. This manifests the importance of this research work completed using a dataset generated in natural settings.

2 Materials and methods

2.1 Study area

The photographs were captured at ICAR-Directorate of Onion and Garlic Research (DOGR), Pune, Maharashtra, India, centered at latitude 18°50'27.99"N, longitude 73°53'12.88"E EPSG:4326 WGS 84/UTM zone 43°N over the span of 4 months from September to December 2023.

TABLE 1 Comparative interpretation of different deep learning models previously used in object detection.

Model	Dataset	Task	Utility	Reference
YOLO-ACT (YOLOv8s)	Open-source dataset of apple leaf diseases- PlantDoc and AppleLeaf9	Disease detection	Uses cross-layer integration to improve detection accuracy	Zhang et al., 2024 .
YOLOv8	Roboflow dataset (Leaf diseases)	Disease detection	GOGR EELAN module for enhanced feature extraction and WSIoU loss function to improve convergence speed and localization accuracy	Wen et al., 2025 .
YOLOv8	Greenhouse vegetable disease dataset built by collecting online videos	Disease detection	Occlusion perception attention module to diminish background and HIoU loss function to address regression loss	Wang et al., 2024 .
MSAM-YOLO (YOLOv8)	PlantVillage and other network datasets	Disease identification	Multiscale attention mechanism to capture subtle features of disease	Xie and Ding, 2025
TSBA-YOLO (YOLOv5)	Custom dataset on Tea leaf disease created using drone and mobile photography	Disease detection	Feature fusion network and spatial feature fusion to resist background inference	Lin et al., 2023
YOLOv8	Mixed dataset containing images collected from the field and the internet	Detection and categorization of tomato fruits	Depthwise separation convolution to reduce computational complexity and Feature enhancement module to avoid occlusion due to overlaps in the images	Yang et al., 2023 .
YOLOv8n	Open source apple leaf disease dataset -AppleLeaf9	Disease detection	C3 Ghost convolution to reduce parameter count and a global attention mechanism with a bidirectional feature pyramid network for small lesion detection	Gao et al., 2024 .
Pyramid-YOLOv8 (YOLOv8x)	Rice blast disease dataset created using smartphone	Disease detection	Multi attention feature fusion network to enhance model precision	Cao et al., 2024
CAM-YOLO (YOLOv5)	Lobaro tomato dataset	Detection and classification	Convolutional block attention module and Distance oversection over union to detect overlapped and small objects	Appe et al., 2023
YOLOv8	Sourced from Kaggle	Classification	Anchor boxes were added to align with size and shape of disease patterns	Ghafar et al., 2024
EggplantDet (YOLOv8)	Custom dataset consisting images collected by inspection robots and smartphones	Disease detection	FasterNet for efficient feature extraction and Tripple Attention Mechanism for small object detection	Liu and Wang, 2025
Alpha-ElOU-YOLOv8 (YOLOv8n)	Custom dataset of rice leaf disease images	Disease detection	Uses EloU and alpha loss function instead of box loss function to enhance model performance	Trinh et al., 2024
YOLOv8	Custom dataset of tomato leaf disease images	Disease detection	Uses Grouped Depthwise Convolutions and Squeeze and Excitation blocks to improve model accuracy	Shen et al., 2025

2.2 Image collection

The photographs were taken in a natural environment using a Nikon D7500 DSLR camera with 4000 x 6000 pixels resolution and a Moto g72 smartphone with an image resolution of 3000 x 4000 pixels. The camera was positioned 0.3 to 0.5 m away from the plants. Images of both healthy and diseased plants were obtained. Onion plants were photographed in multiple orientations so that the main features of the disease, such as texture, coloration, shape and morphology depended on the extent of the damage. Four major diseases of onion, such as Anthracnose, Twister disease, Stempylium blight and Purple blotch were the focus of model development. **Figure 1** depicts images of the above-mentioned diseases taken from the symptomatic onion plants.

2.3 Data preprocessing

From the captured images, 1000 images were chosen to generate a dataset for this study. Among the 1000 images, 800 images were equal number of diseased plants infected by Anthracnose, Stempylium blight, Purple blotch and Twister disease; and the remaining 200 images were from healthy onion plants. A computer vision platform 'Roboflow' was used to manually label the disease symptoms. Roboflow supports various annotation formats (bounding boxes, polygons, masks, keypoints) for diverse computer vision tasks (<https://roboflow.com/formats>). Real-time collaboration ensures consistent annotations, while pre-labeling with existing models greatly reduce human effort. The images were uploaded batch-wise and image annotation was done using such as bounding box feature. **Figure 2** shows the manually annotated images that were resized to a resolution of 640 x 640. Data augmentation techniques, such as flipping, rotation and scaling, were utilized to enhance the dataset and mitigate overfitting. The images were augmented with a 50% probability of horizontal flip, and salt-and-pepper noise was applied to 0.1% of the pixels. After augmentation 1391 images were used for training the model, 189 for validation and 100 for test.

2.4 Model training and architecture of YOLO

The YOLOv8 and YOLOv5 network are similar, consisting mainly of the backbone, neck and head. YOLO is an object detection algorithm that can identify and detect objects in an image. By applying the selected YOLO models to the preprocessed dataset, these YOLO models will train the data to detect object. YOLO models were employed for training because of its advanced capabilities for real-time object detection and superior performance in terms of speed and accuracy. YOLOv8 is a state-of-the-art object detection, segmentation, and classification algorithm developed by Ultralytics. It builds on previous YOLO versions with improved accuracy, speed, and flexibility. Key features include anchor-free detection, a dynamic backbone, and a unified framework for multiple vision tasks. YOLOv8 is optimized for real-time applications and supports deployment on edge devices. The training process involved the data pre-processing methods and the data.yaml file, a configuration file integral to the YOLOv8 framework, which is programmatically generated to encapsulate essential metadata about the dataset, including class labels, directory paths for training, validation, and testing datasets, as well as the number of instances per class. The file served as a critical input for the YOLOv8 pipeline, enabling streamlined preprocessing and facilitating efficient integration of the model with the dataset during the training process. The models were fine-tuned over 50 epochs with a batch size of 16, utilizing the Adam optimizer and a learning rate scheduler for optimal convergence. Here, Adam optimizer was used since it enhances YOLOv8 for onion disease detection by handling small, variable symptoms like browning and lesions. Its adaptive learning rates help the model adjust to image variations, while its robustness ensures reliable training even with noisy or incomplete data. Adam's fast convergence makes it ideal for real-time disease detection, enabling timely agricultural decisions (Ghafar et al., 2024). **Figure 3** shows a block diagram of the training and testing frameworks of the proposed model. The architecture of a basic object detection model consists of a

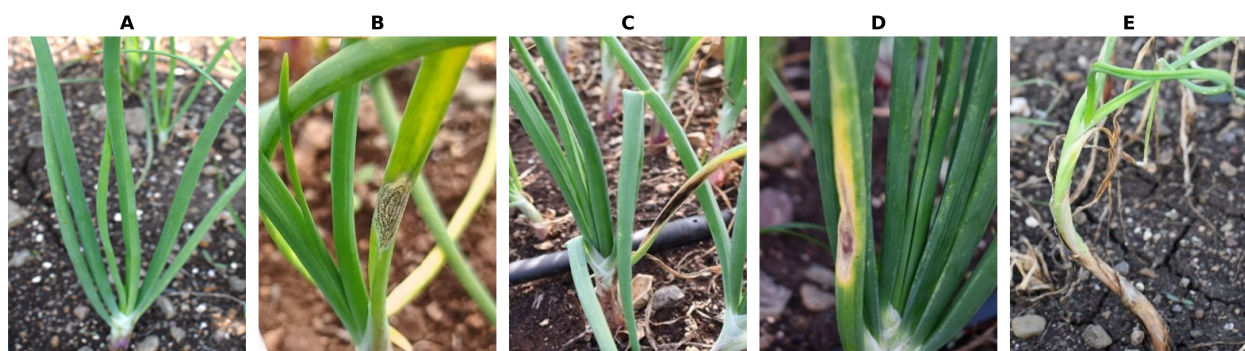


FIGURE 1
Images of diseased and healthy onion plants (A) Healthy plant (B) Anthracnose (C) Stempylium (D) Purple blotch (E) Twister disease.

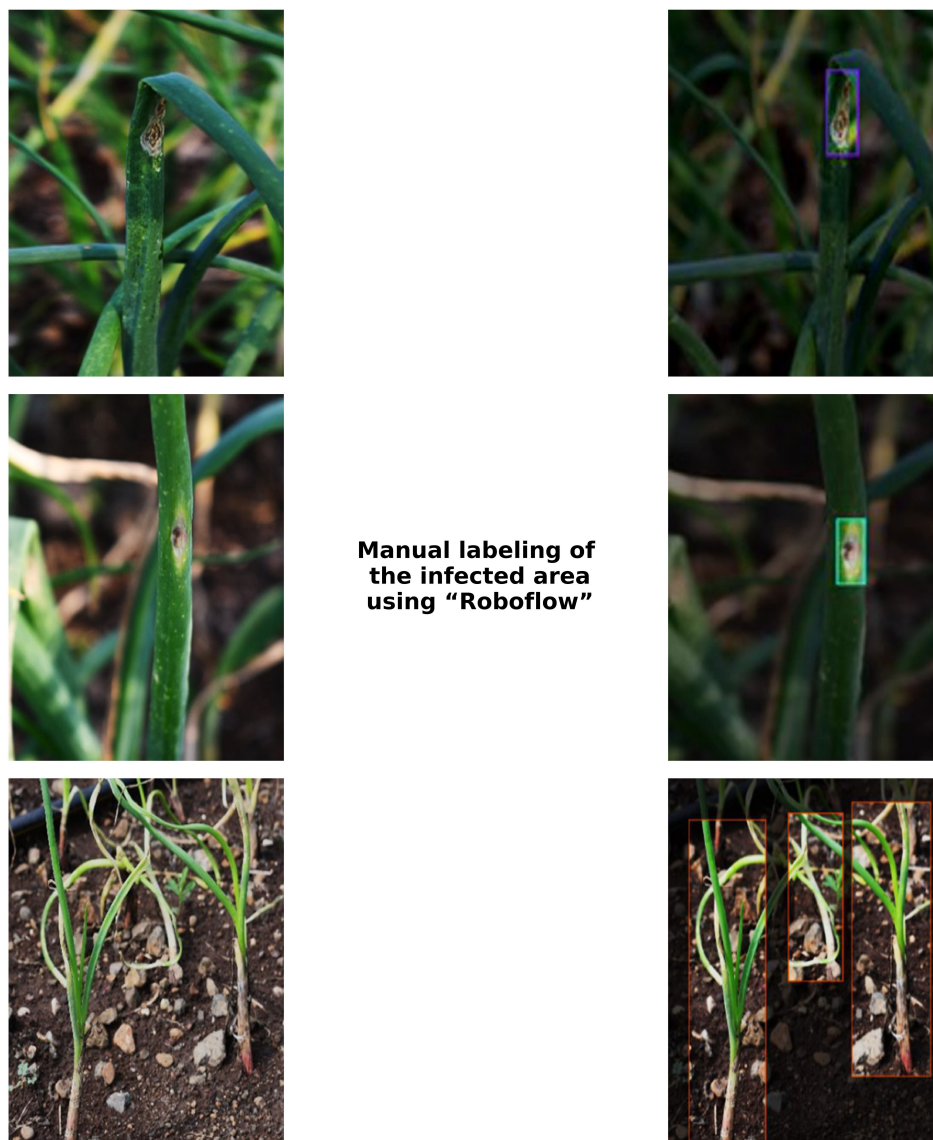


FIGURE 2
Annotated images.

backbone, a neck and a head (Figure 4). Meaningful feature extraction from the input is the responsibility of the backbone, which is also referred to as a feature extractor. The neck operates as a bridge between the backbone and the head, executing feature fusion processes and integrating contextual information. The head was responsible for producing outputs, bounding boxes, class predictions, and confidence scores for detected objects. The primary features of YOLOv5 and YOLOv8 include anchor-free detection, mosaic data augmentation, C2f module, decoupled head, and a modified loss function (Sandhya and Kashyap, 2024). Mathematical equations of YOLOv8 are given below:

1. Object detection problem formulation:

Object detection is treated as a regression problem, where the goal is to predict both the class of an object and its bounding box from a given input image. Each bounding box represented by four coordinates

x, y – coordinates of center of bounding box

w, h – width and height of bounding box, relative the entire image

Class prediction: for each bounding box YOLO predict the probability distribution over the class labels.

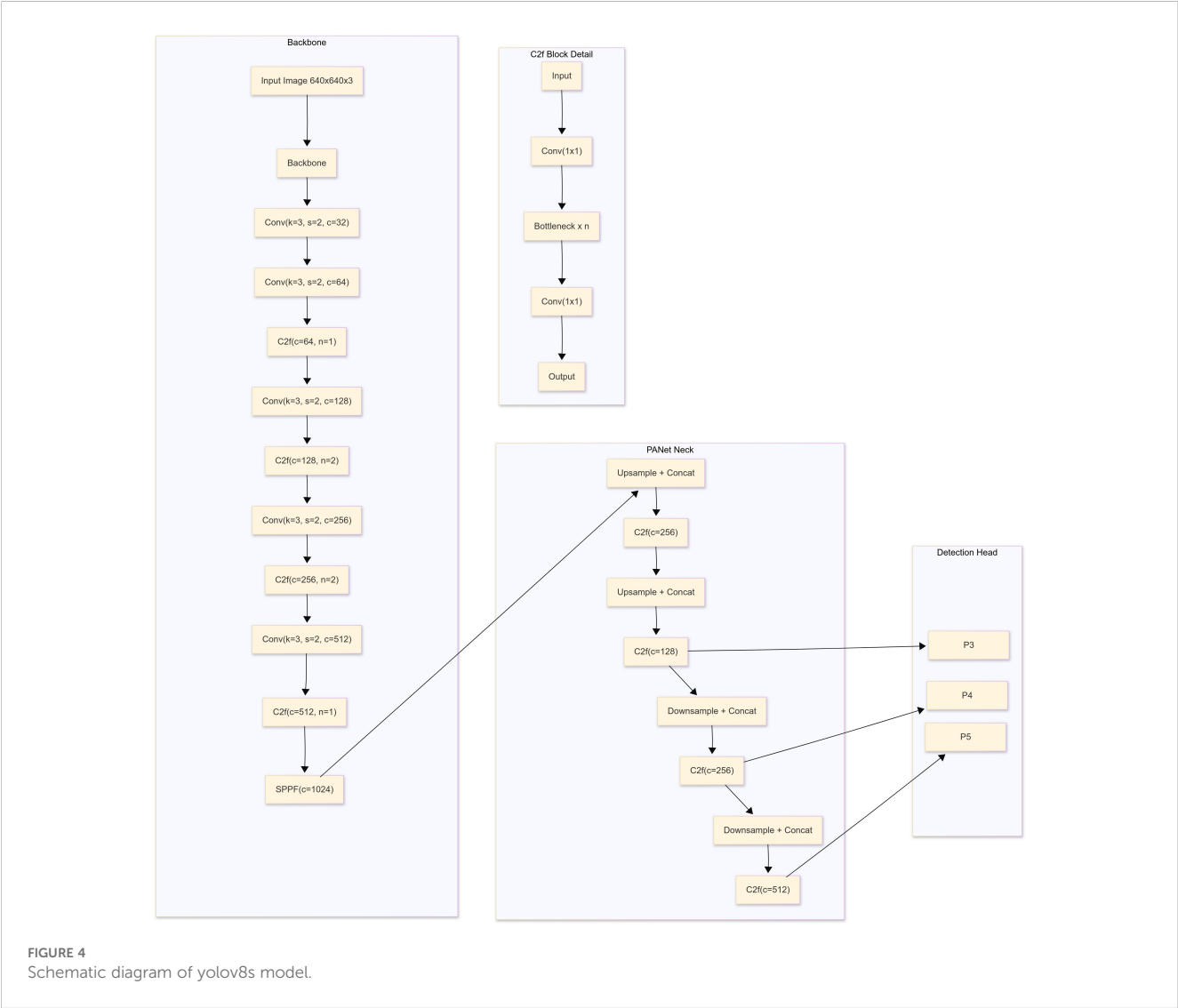
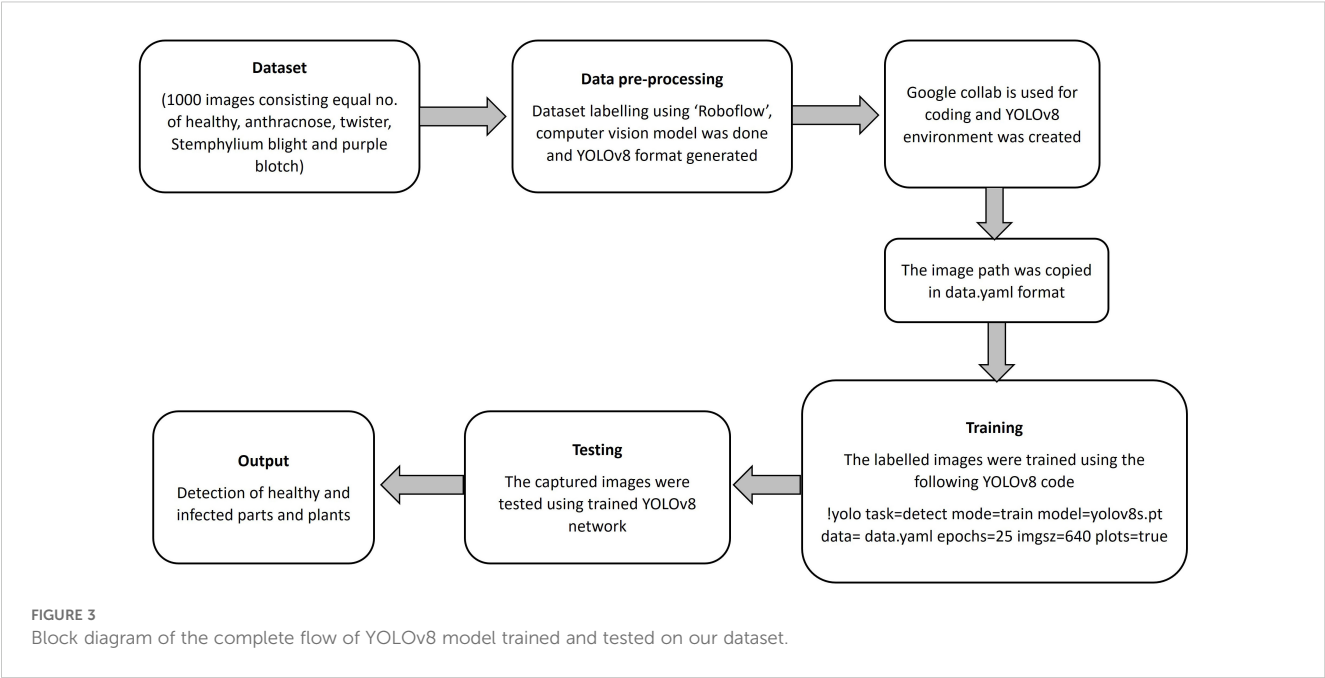
C – no of classes,

S – no of cell grid in image,

B – no of bounding boxes per cell.

2. YOLOv8 Architecture:

Backbone: The backbone extracts features from the input image using a convolutional neural network (CNN). YOLOv8 employs the CSPDarknet architecture with ELAN (Efficient Layer Aggregation Networks), which enhances gradient flow and efficiency



$$F = \text{Backbone}(I) \quad (1)$$

Where F is a feature map, and I is the input image (Equation 1).

Neck: The neck helps in multi-scale feature aggregation and uses Path Aggregation Network (PAN) which combines features from different scales for better object detection across various sizes.

$$F_{\text{neck}} = \text{Neck}(F) \quad (2)$$

Where F_{neck} is the feature map passed to the detection head (Equation 2).

Detection Head: The head predicts object bounding boxes, class scores, and confidence scores using anchor-free detection approach. YOLOv8 divides the image into a grid and predicts bounding boxes and their corresponding class probabilities for each cell.

The output of each cell is vector $[x, y, w, h, p_1, p_2, p_3, \dots, p_c]$, Where, x, y, w, h are the bounding box coordinates, p_i is the class probability for class i , C is the number of object classes.

3. Mathematical formulation for YOLOv8

Both YOLOv5 and YOLOv8 follows the regression approach to predict bounding boxes and class probabilities. The loss function use to training is a combination of several components:

Bounding Box Loss: The difference between the predicted and ground truth bounding box coordinates is measure using the IoU (intersection over Union) loss, typically combined with a smooth L1 loss.

$b_{\text{pred}} = (x, y, w, h)$ predicted bounding box and

$b_{\text{gt}} = (x_{\text{gt}}, y_{\text{gt}}, w_{\text{gt}}, h_{\text{gt}})$ corresponding ground truth.

The loss can be written as: during network training, the loss function is a tool used to represent the difference between predicted and actual values. It plays crucial role in training of disease detection models. In YOLOv5s and YOLOv8s multiple loss functions are combined for training bounding box regression, classification, and confidence. The loss function use are as follows:

$$\begin{aligned} \text{Loss}_{\text{box}} = & \lambda_{\text{coord}} \sum_i I_{\text{obj}} [(x_i - x_{i,\text{gt}})^2 + (y_i - y_{i,\text{gt}})^2] \\ & + \lambda_{\text{wh}} \sum_i I_{\text{obj}} [(w_i - w_{i,\text{gt}})^2 + (h_i - h_{i,\text{gt}})^2] \end{aligned} \quad (3)$$

Where, I_{obj} is the indicator function (equal to 1 if an object is present, and 0 otherwise), and λ_{coord} , λ_{wh} are the weights that balance different components of the loss (Equation 3).

Classification Loss: YOLOv8 also predict the class probabilities for each bounding box. The loss for class prediction is computed using binary cross-entropy

$$\text{Loss}_{\text{cls}} = - \sum_c y_{\text{cls}} \log(p_{\text{cls}}) \quad (4)$$

where y_{cls} is the true class label and p_{cls} is the predicted probability for the class (Equation 4).

As shown in Equation 5, the total loss is a combination of classification and box regression losses.

$$\text{Loss}_{\text{conf}} = \text{Loss}_{\text{cls}} + \text{Loss}_{\text{box}} \quad (5)$$

4. CIoU (intersection over Union):

The CIoU is a crucial part of the bounding box prediction, and it measures the overlap between predicted and ground truth boxes.

Its used to determine whether a predicted bounding box is considered a correct detection or not.

$$\text{CIoU} = 1 - \text{IoU} + \frac{\rho^2(b_{\text{center}}, b_{\text{center}}^{\text{gt}})}{C^2} + \alpha \cdot v \quad (6)$$

Where represents the intersection over Union, is the distance between the center point of the predicted box and the ground truth box, c is the diagonal length of the smallest enclosing box covering both the predicted and ground truth box, v is the aspect ratio consistency term and α is the probability coefficient (Equation 6).

$$\text{dfl} = - \sum_{i=1}^n w_i \log(p_i) \quad (7)$$

Where w_i is the weight usually adjusted according to the position of the true bounding box. p_i is the probability of each class in the predicted probability distribution (Equation 7).

Convolutional Block Attention Module

Convolutional Block Attention Module (CBAM) is an attention module for convolutional neural networks. CBAM is simple yet effective attention module for feed-Forward convolutional neural networks (Figure 5). Given an intermediate feature map, the module sequentially infers attention maps along two separate dimensions, channel and spatial, then the attention maps are multiplied to the input feature map for adaptive feature refinement. CBAM combine channel and spatial attention mechanism, effectively identified the key features in the images while suppressing irrelevant noise. This dual attention mechanism notably enhances the accuracy and efficiency of detection (Woo et al., 2018).

Channel Attention (CAM) identifies “what” features are important by analyzing channel-wise relationships. It applies both average and max pooling to capture global context, processes them through a shared MLP with bottleneck reduction (e.g., reducing channels by ratio $r=16$), and generates a channel attention mask via sigmoid activation. This mask highlights informative channels while suppressing less relevant ones.

Average Pooling:

$$F_{\text{avg}}^c = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W F^c(i, j) \quad (8)$$

Output: F_{avg} (Equation 8)

Max Pooling as shown in Equation 9:

$$F_{\text{max}}^c = \max F^c(i, j) \quad (9)$$

Shared MLP(Bottleneck): with reduction ratio r :

$$\text{MLP}(F_{\text{pool}}) = W_1 (\text{ReLU}(W_0 F_{\text{pool}})) \quad (10)$$

Where: $W_0 \in \mathbb{R}^{(C/r) \times C}$ weight for dimension reduction,

$W_1 \in \mathbb{R}^{(C/r) \times C}$ weight for dimension restoration (Equation 10).

Combine and Activate:

Sum the MLP outputs and apply sigmoid(σ) to generate the channel attention mask M_{c} (Equation 11):

$$M_{\text{c}}(F) = \sigma (\text{MLP}(F_{\text{avg}}) + (\text{MLP}(F_{\text{max}})) \quad (11)$$

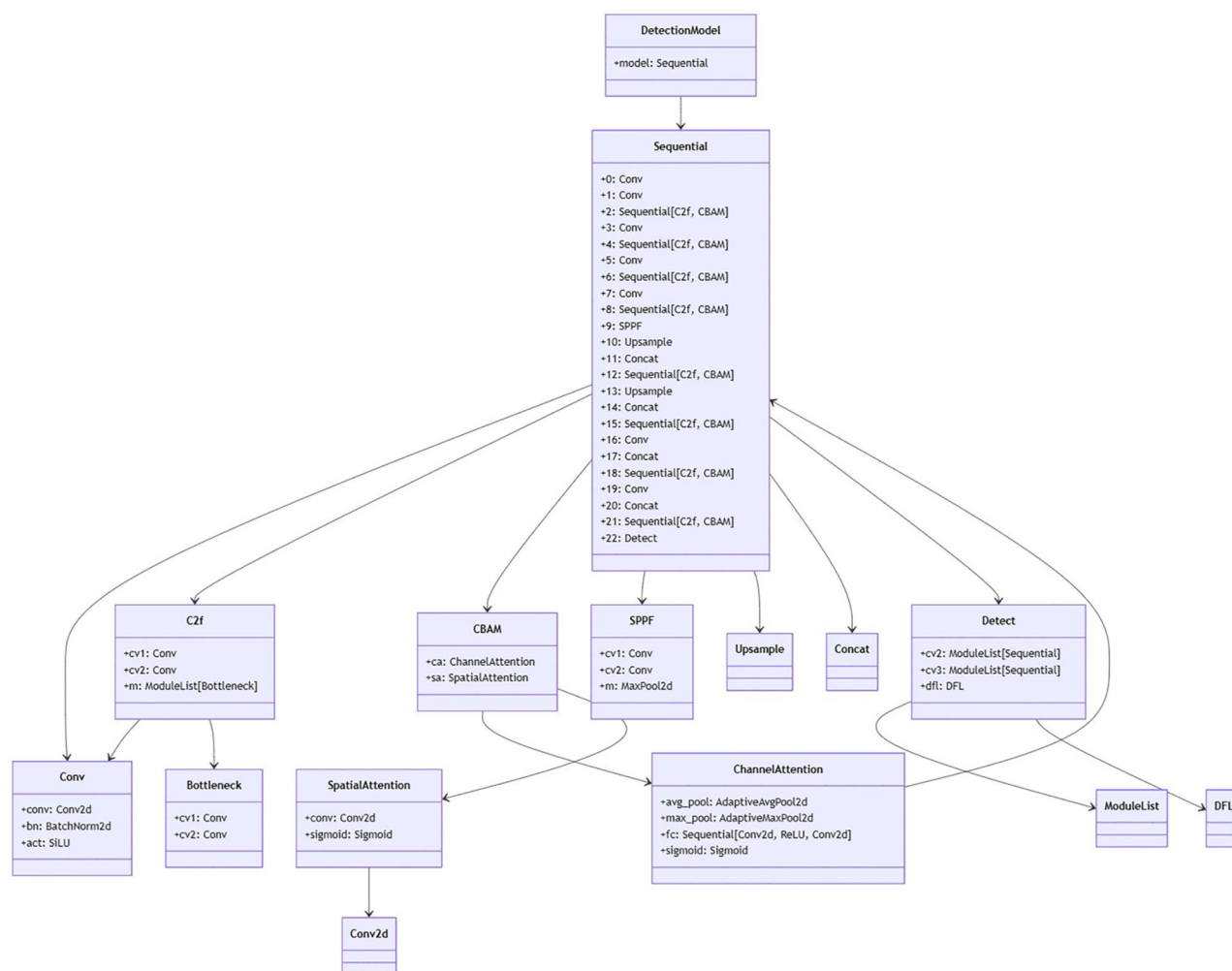


FIGURE 5
Schematic diagram of Yolov8+CBAM.

Apply to input features:

$$F' = M_c (F) \otimes F \quad (12)$$

Where: denote $\otimes$ the element wise multiplication (Equation 12).

Average pooling capture global context, while max pooling preserves salient features combine both improves robustness. Bottleneck ($r=16$) balances efficiency and effectiveness by reducing MLP parameters. Sigmoid normalizes the attention weights to $[0,1]$, acting as a soft feature selection.

Spatial Attention module (SAM) Formulas: spatial attention module is comprised of a three-fold sequential operation. The first part of it is called the channel pool, where the input tensor of dimensions ($C \times H \times W$) is decomposed to 2 channels, i.e. ($2 \times H \times W$), where each of the two channels represents max pooling and average pooling across the channels. This serve as the input to the convolutional layer which output a 1-channel feature map, i.e. the dimension of output is ($1 \times H \times W$).

Given the channel-refined feature map $F' \in \mathbb{R}^{C \times H \times W}$ from CAM: the spatial attention mechanism follows these steps

Compute average-pooled features across channels as shown in Equation 13:

$$F'_{avg} = AvgPool_{channel}(F') \in \mathbb{R}^{1 \times H \times W} \quad (13)$$

Equation 14 shows, where each spatial position (i,j) is the mean of all channels

$$F'_{avg}(i,j) = \frac{1}{C} \sum_{c=1}^C F'_{c,i,j} \quad (14)$$

Equation 15 shows compute max-pooled features across channels:

$$F'_{max} = maxPool_{channel}(F') \in \mathbb{R}^{1 \times H \times W} \quad (15)$$

Where each spatial position (i,j) is the maximum across channels (Equation 16)

$$F'_{max}(i,j) = max_{c=1}^C F'_{c,i,j} \quad (16)$$

Concatenate pooled features

In Equation 17 stack F'_{avg} and F'_{max} along the channel dimension:

$$F'_{concat} = \text{Concat}(F'_{avg}, F'_{max}) \in \mathbb{R}^{2 \times H \times W} \quad (17)$$

Apply convolution to generate spatial attention pass the concatenated features through a 7×7 convolutional layer followed by sigmoid activation

$$M_s(F') = \sigma(f_{7 \times 7}(F'_{concat})) \in \mathbb{R}^{1 \times H \times W} \quad (18)$$

In Equation 18 Where: $f_{7 \times 7}$ is a convolution with single output channel, σ is the sigmoid function, normalizing attention weight to [0,1]

Apply attention to the refined feature map multiply the spatial attention map M_s with channel-refined feature map F'

$$F'' = M_s(F') \otimes F' \quad (19)$$

Where denote $\otimes$ the element wise multiplication (Equation 19).

Dynamic Task-Aligned Head (DTAH)

The Dynamic Task-Aligned Head (DTAH) is designed to address the inherent misalignment between classification and localization task in object detection (Feng et. al., 2021). Traditional decoupled heads process these tasks in parallel, leading to discrepancies in feature learning where classification focuses on discriminative local features, while localization requires global spatial context for precise bounding box regression. In domain disease detection, the accuracy of both task is indispensable. DTAH mitigates this issue through three key innovation:

1. Task interaction module: A shared feature extractor using grouped convolutions to explicitly model interactions between classification and localization tasks. This module generates task aware features by using multi-level spatial and semantic information, ensuring that both tasks operate on aligned feature representations.

$$F_{interact} = \sigma(W_2 * \text{ReLU}(W_1 * F_{in})) \circ F_{in} \quad (20)$$

Where in Equation 20:

$F_{in} \in \mathbb{R}^{H \times W \times C}$ Input backbone features.

W_1, W_2 : Grouped convolutional kernels

σ : Sigmoid activation for attention gating.

$\circ$: Element-wise multiplication.

2. Deformable Localization branch: Incorporates deformable convolutional (DCN) in the localization pathway to dynamically adjust receptive fields based on object geometry. This allows the model to adapt to irregular disease pattern by predicting per pixel sampling offsets, enhancing boundary precision.

For adaptive spatial sampling, deformable convolutions predict offsets Δp_k for each kernel position p_k

$$F_{deform} = \sum_{k=1}^k w_k \cdot F_{interact}(p + p_k + \Delta p_k) \quad (21)$$

Where in Equation 21: $\Delta p_k = W_{offset} * F_{interact}$ (offset prediction)

w_k : Learnable kernel weights.

3. Unified optimization:

A joint loss function balances classification accuracy and localization precession with additional alignment loss term that panelizes spatial mismatches between task specific features.

The total loss combines task specific objectives with an alignment penalty as shown in Equation 22.

$$L_{DTAH} = \lambda_1 L_{cls} + \lambda_2 L_{reg} + \lambda_3 ||M_{cls} - M_{reg}||_2 \quad (22)$$

Where:

L_{cls} : Focal loss classification.

L_{reg} : Distribution Focal Loss (DFL) for bounding box regression.

M_{cls}, M_{reg} : spatial attention maps from each task.

λ_i : Balancing weights.

The DTAH architecture is implemented as a lightweight yet powerful replacement for conventional decoupled heads in Figure 6. Supplementary Figure S1 is visualization of overall architecture of proposed onion foliar disease detection model based on yolov8 integrated with CBAM (convolutional block attention module) and DTAH (Dynamic Task-Aligned Head). The diagram shows CBAM is embedded within the C2f blocks for enhanced feature extraction, and DTAH is added at the end of improve detection accuracy with deformable convolutional.

2.5 Model evaluation parameters

F1 Score: To evaluate the detection performance, metrics like precision, recall, and F1 score are used. For each class:

Precision: The fraction of true positive prediction among all positive predictions (Equation 23).

$$\text{Precision} = \frac{TP}{TP + FP} \quad (23)$$

Recall: The fraction of true positive predictions among all actual positives (Equation 24).

$$\text{Recall} = \frac{TP}{TP + FN} \quad (24)$$

F1 Score: The harmonic means of precision and recall (Equation 25):

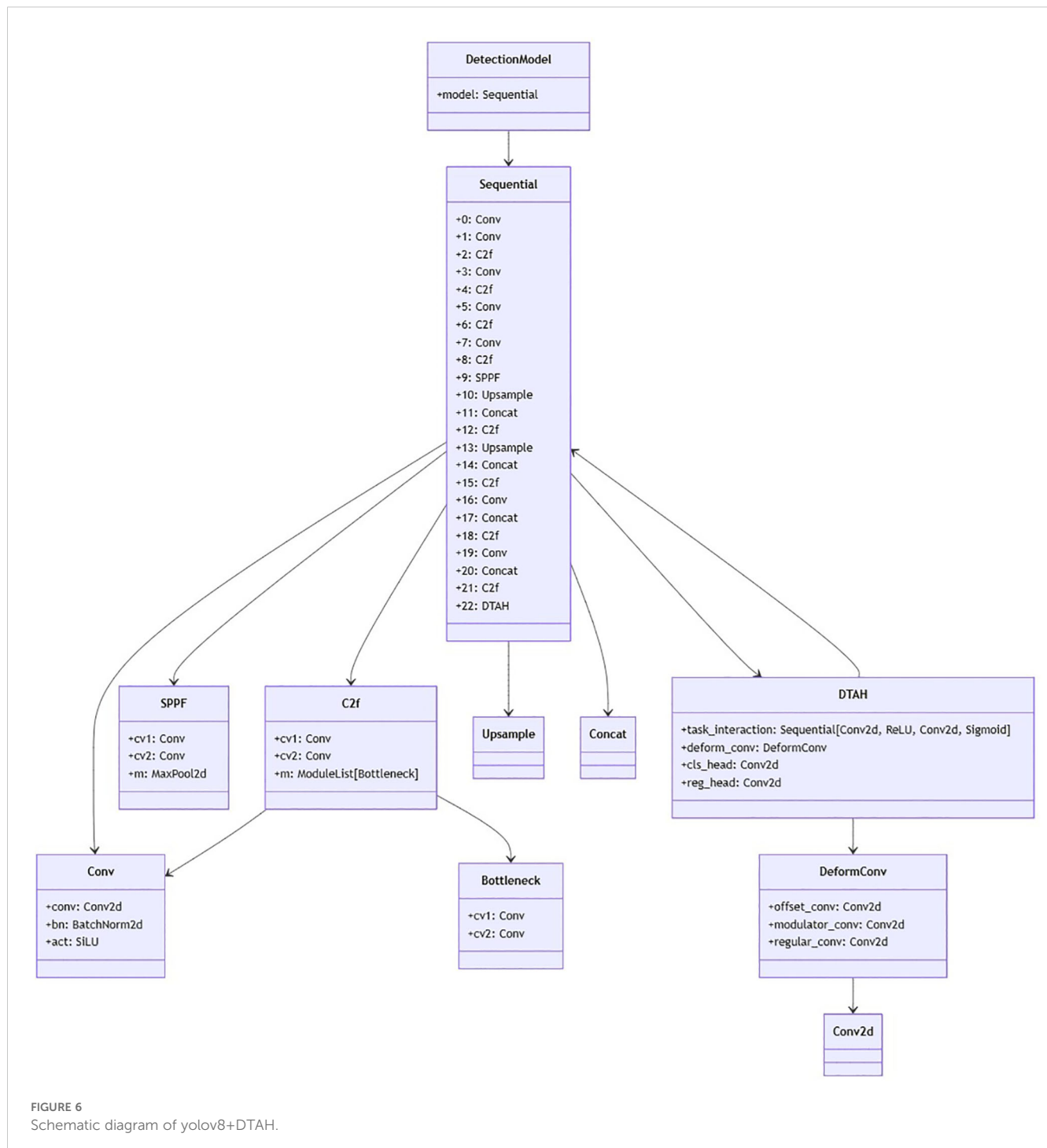
$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (25)$$

TP (True Positive) refers to the number of correctly identified positive samples.

TN (True Negative) refer to the number of correctly identify negative samples.

FP (False positive) represent the number of negative samples incorrectly identified as positive.

FN (False Negative) refer to the number of positive sample incorrectly identified as negative.



3 Experimental results and analysis

3.1 Dataset distribution

The distribution of the dataset and number of instances were calculated and the results shown in Figure 7. Figure 7A shows a bar chart of instances per class. The classes represented were healthy, Anthracnose, Purple blotch (PB), Stemphylium blight and Twister disease. The training dataset is well balanced with each class having comparable number of instances Anthracnose (421), healthy (420), PB

(379), Stemphylium blight (420) and twister (416) this balanced class distribution ensure that no single class dominates the dataset, reducing the risk of model bias. It enhances the model ability to generalize across different disease categories, leading to improve training and prediction accuracy. Figure 7B shows the bounding-box distribution. Distinct colors were used to depict boxes, which may indicate distinct classes. In some images, items were frequently found in the center, as indicated by the overlap and concentration of boxes in the center which could lead to a central bias in the model. Figures 4C and 7D show a heatmap of bounding box centers (x, y) and a heatmap of bounding box

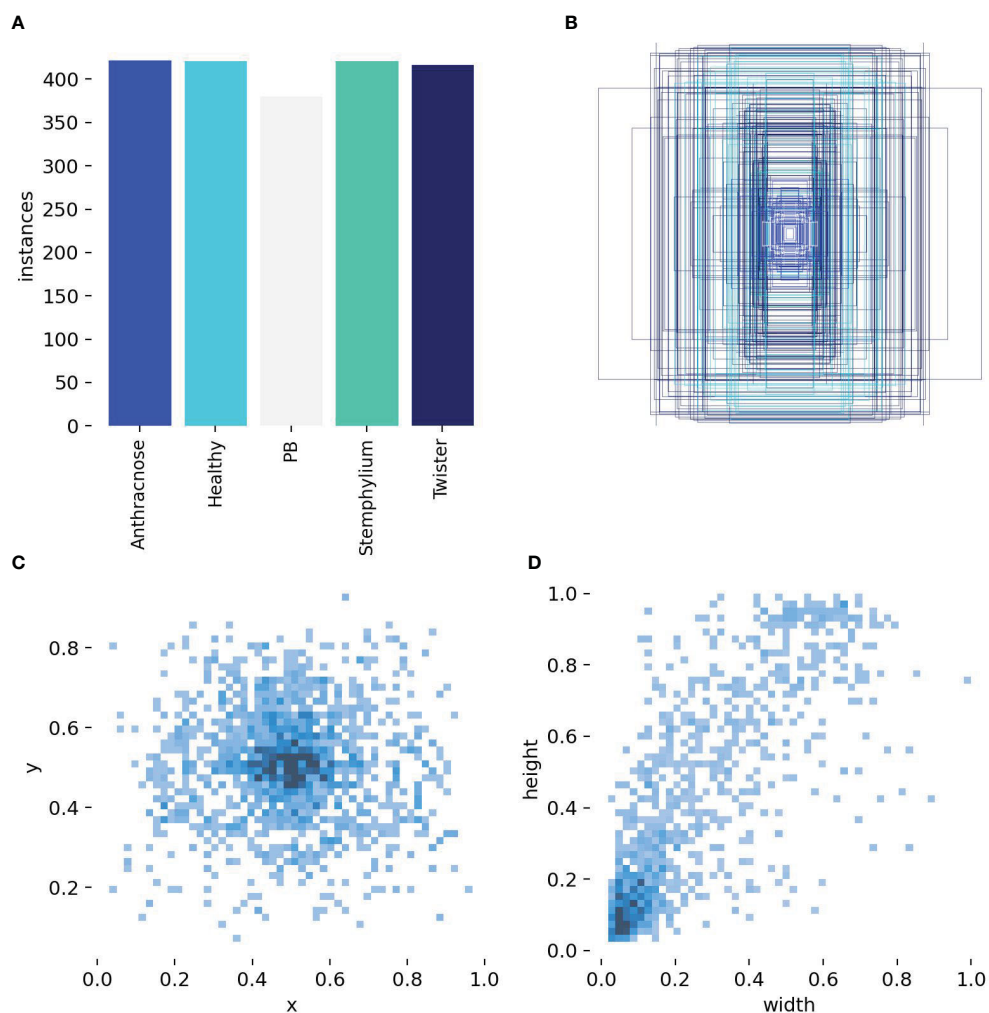


FIGURE 7
Labels and label distribution, (A) Bar Chart of instances per class, (B) Bounding box distribution, (C) Heatmap of bounding box centers, (D) Heatmap of bounding box dimensions.

dimensions (width and height) respectively. The heat map shows that the majority of the objects are situated close to the center of the image, and the majority of items in the dataset are comparatively small, as evidenced by the large concentration of bounding boxes with smaller width and height values. Uniformity in object dimensions could lead to quicker model convergence but may not generalize well to real-world data with varied object sizes. The correlogram of the custom dataset (Supplementary Figure S2) highlights the model's detection confidence across image regions. This analysis could help us adjust the parameters of the object identification model, such as adjusting anchor sizes, balancing the dataset, or employing data augmentation to lessen the central bias.

3.2 Training and validation performance metrics

The performance metrics of the model are shown in Figure 8. With all training losses such as box, classification, and DFL

(Distribution Focal Loss) gradually declining, the improved YOLOv8s model exhibits learning to predict more precise bounding box details, suggesting enhanced object localization and classification on training data, indicating that the model is generally effective on unseen data. CABM and DTAH attention mechanisms were applied to improve overall precision and accuracy in disease detection by localization of smaller disease spots in the image. Grad cam heatmaps were adapted for comparative visualization of detection accuracy (Figure 9). The overall performance demonstrates that the model is capable of accurate detection while maintaining sensitivity to true positives. Overall, the model showed good convergence and encouraging outcomes.

3.3 Model performance measure

The comparative evaluation of various YOLO architectures on the onion leaf disease detection dataset reveals significant performance variations across different model configurations

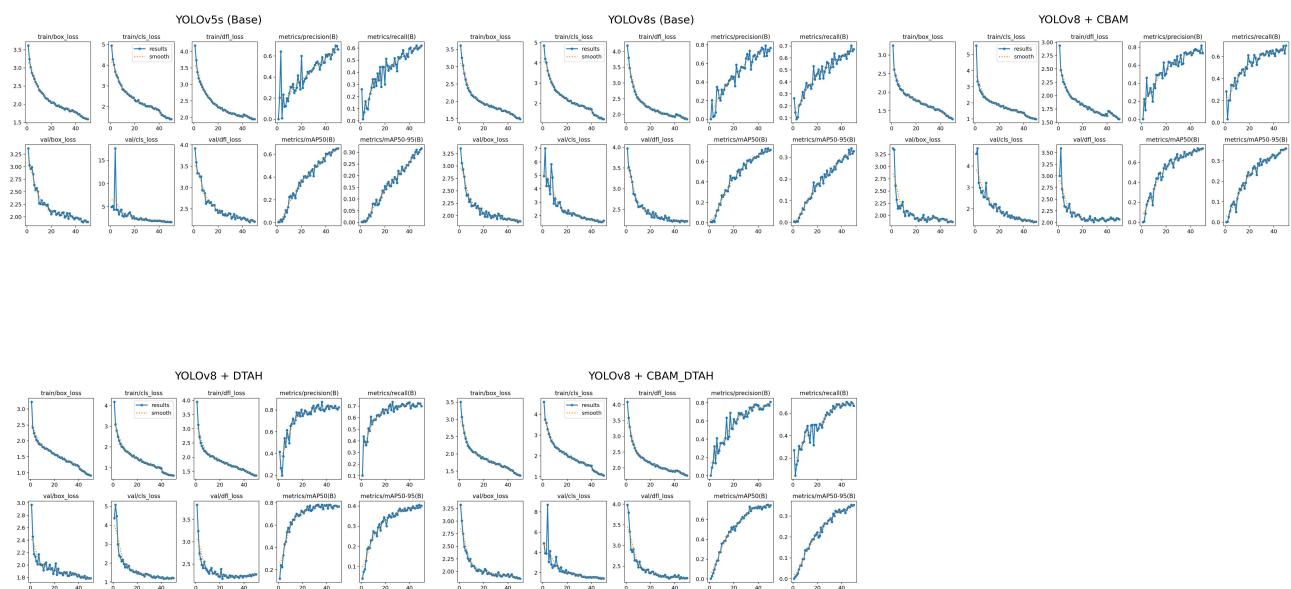


FIGURE 8

Visual analysis of model evaluation indicators (Precision, recall, and mAP@0.5) during training.

(Figure 10). The dataset comprises five classes (Anthracnose, Healthy, PB, Stemphylium, Twister) with 189 validation images containing 257 total instances. The results demonstrate that YOLOv8 with DTAH achieved the best balance of performance metrics, attaining the highest mAP50 (0.773) while maintaining competitive inference speed (8.1ms). However, base YOLOv8s model shows significant improvement over YOLOv5s (+7.5% in mAP50), and the DTAH modification provides an additional 4% boost in mAP50. Notably, the DTAH variant achieves the highest recall (0.727) among all models, indicating better detection capability for difficult cases (Table 2).

The class-wise mAP50 performance (Table 3), Precision recall curve (Supplementary Figure S3) and normalized confusion matrix (Figure 11) reveals several important insights about how each model variant performed across different onion leaf disease categories. Starting with Anthracnose detection, we observed a clear progression in performance from the baseline YOLOv5s (0.620 mAP50) to the enhanced YOLOv8 variants, with the DTAH modification showing particularly impressive results (0.786 mAP50) comparatively showing improvement of 17% over YOLOv5s and 9% over base YOLOv8s. This substantial boost suggests the deformable attention mechanism in DTAH is exceptionally well-suited for detecting the irregular lesion patterns characteristic of Anthracnose. The Healthy class shows more consistent performance across all models (ranging from 0.716 to 0.784 mAP50), indicating that all architectures can reliably identify healthy plants, with the CBAM+DTAH attention integrated model achieving the highest score (0.784).

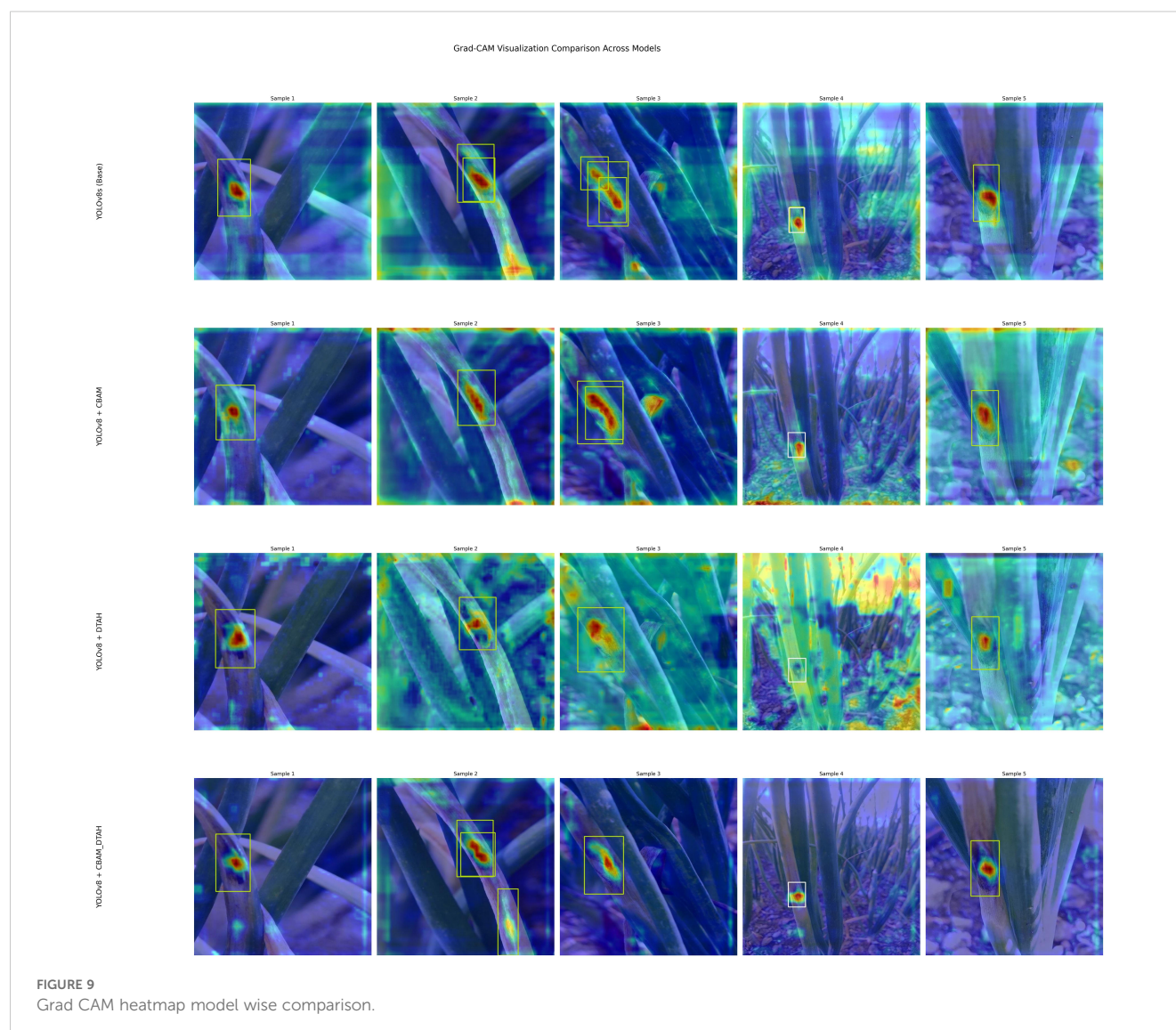
For PB detection, the base YOLOv8s showed improvement over YOLOv5s (0.717 vs 0.680), with DTAH providing a modest additional gain (0.79). Interestingly, the CBAM model showed slightly reduced performance for this class, suggesting the CBAM attention may be less

effective for small spot compared to disease patterns. Stemphylium detection follows a similar pattern, where CBAM provides the best performance (0.742), outperforming even DTAH (0.665).

In contrast, our YOLOv8+DTAH (Dynamic class align head) modification exhibited superior performance across all metrics, achieving 77.3% mAP50 with 4.85–6.56% absolute improvement over the Detectron2 models - while using only 11.1M parameters (69% fewer than RetinaNet, 75% fewer than Faster R-CNN) (Table 4). Remarkably, this accuracy gain comes with substantially faster inference speed (123 FPS), making it 1.9–2.8X faster than the comparative methods. The results demonstrate that the YOLOv8 architecture with DTAH attention mechanisms provides better feature representation for onion leaf disease detection compared to traditional two-stage approaches, while maintaining the computational efficiency crucial for real-world agricultural applications. The performance advantage is particularly notable given the model's compact size, suggesting that the DTAH modification effectively enhances the network's capability to recognize varied disease patterns without requiring larger backbone networks.

4 Discussion

Microclimate alterations across the crop growth cycle greatly influences productivity and biotic interference in onion crop. Onion production in Maharashtra has been hampered by diseases such as Twister-anthracnose complex, Stemphylium blight and Purple blotch. The twister-anthracnose complex is a devastating disease in nurseries and bulb crops of onion, which shrink onion yield up to 80–100% (Patil et al., 2018). Purple blotch along with Stemphylium, affects the leaves and bulbs, culminating in losses up to 97% (Kareem et al., 2012). Stemphylium causes severe leaf blighting



and incomplete bulb development which results in reduced bulb size alongside yield loss up to 20% (Correa et al., 2023). Sustainable technologies such as artificial intelligence, robotics, machine learning, deep learning etc., can be leveraged to address these issues effectively by deploying decision support system for timely implementation of countermeasures. Digital image processing and unmanned aerial vehicles (UAVs) have been utilized to monitor the crop and spot trends in biotic and abiotic stress response under varying scenarios including humid areas, weed growth, deficiency of vegetation, and poor harvest efficiency in onion farming.

The YOLOv8s Architecture integrated with DTAH attention mechanism is an Advanced method for object recognition that evaluates photographs at high speed compared to other deep learning models. This novel method of image analysis divides an image into several grid cells and predicts the bounding box coordinates even for small instances and class probabilities using a single neural network with better localization of diseased area on leaf. This results in a faster and more precise disease identification process by enabling a more efficient and accurate image assessment

(Zhang et al., 2024). This model manifested highest performance among five analogous variants of yolo showcasing precision, recall and mAP50 values of 82%, 72.7% and 77.3% respectively. In similar context, Lin et al., 2023. Used integrated multiple attention mechanisms in TSBA-YOLO - an tea leaf disease detection model-built on base architecture of yolov5 which detected tea leaf diseases with 85.35% accuracy. In another study, YOLOv7 (YOLO-T) model build using CBS attention integrated with YOLOv7 resulted in 97.3% detection accuracy while detecting tea leaf diseases (Soeb et al., 2023). Incorporation of attention mechanisms has significantly improved precision of leaf disease detection while reducing possibility of error occurrence because of variation in disease spot size. (Li et al., 2022; Sun et al., 2023; Wang and Liu, 2024). In present study, among the classes, Twister exhibited the highest performance (Precision: 0.88, Recall: 0.841, mAP50: 0.848), demonstrating the model's strong ability to identify diseased plants. Similarly, PB and Anthracnose showed high classification accuracy (mAP50: 0.837 and 0.824, respectively), aligning with findings from prior studies that deep learning

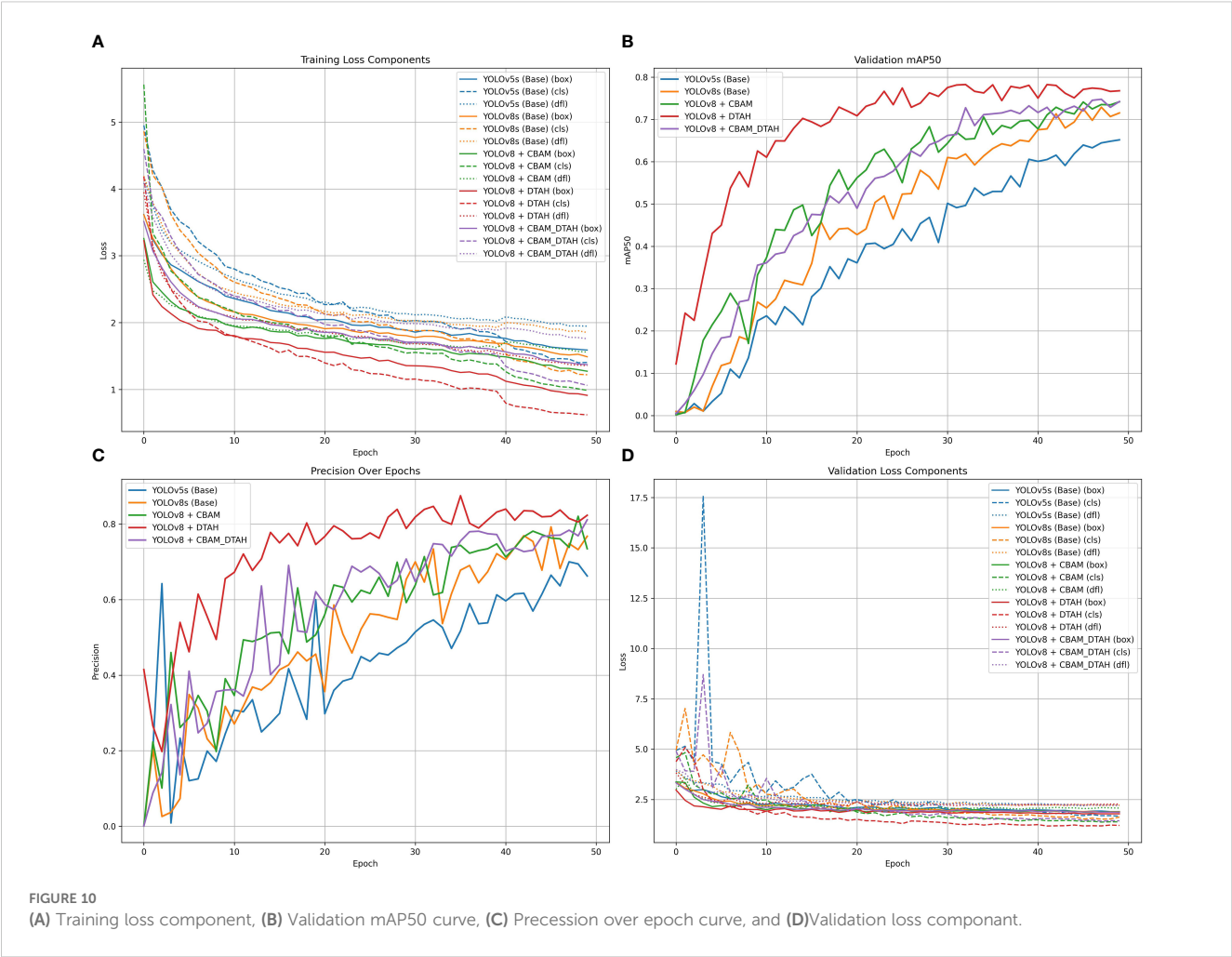


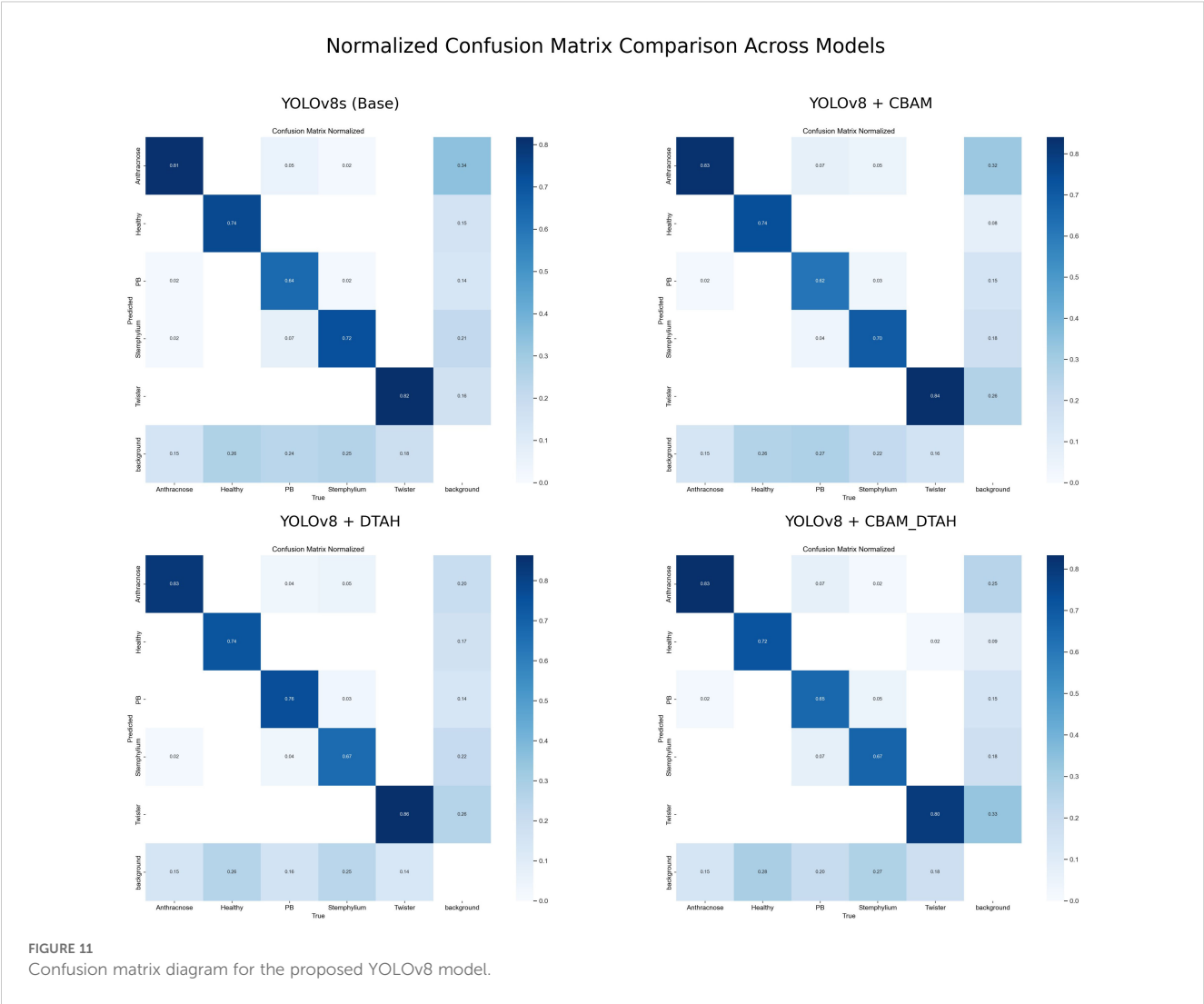
FIGURE 10 (A) Training loss component, (B) Validation mAP50 curve, (C) Precession over epoch curve, and (D) Validation loss component.

TABLE 2 Comparison of accuracy, precession and recall among yolo models.

Model	mAP50	Precision	Recall	Inference Speed (ms)
YOLOv5s (Base)	0.654	0.665	0.626	9.1
YOLOv8s (Base)	0.729	0.749	0.708	8.4
YOLOv8 + CABM	0.74	0.73	0.7	8.5
YOLOv8 + DTAH	0.772	0.815	0.721	8.1
YOLOv8 + CABM_DTAH	0.75	0.784	0.698	8.9

TABLE 3 Comparative class wise accuracy.

Class	YOLOv5s	YOLOv8s	YOLOv8+CABM	YOLOv8+DTAH	YOLOv8+CABM_DTAH
Anthravnose	0.62	0.694	0.74	0.786	0.748
Healthy	0.716	0.741	0.764	0.768	0.784
PB	0.68	0.717	0.649	0.792	0.709
Stemphylium	0.573	0.693	0.74	0.661	0.725
Twister	0.678	0.801	0.811	0.855	0.776



models perform well in distinct symptom recognition (Mohanty et al., 2016; Too et al., 2019). However, Stemphylium had the weakest performance (Precision: 0.854, Recall: 0.585, mAP50: 0.746), with significant misclassification as Purple blotch and background. This issue could stem from visual similarities among foliar diseases, as observed in prior research on plant disease classification (Ferentinos, 2018).

The proposed algorithm effectively detects and identifies onion diseases by generating an optimal bounding box around the affected areas. Figure 12. illustrates the results of visualizing the four major categories of onion diseases, with the bounding box tightly drawn around the relevant regions. This approach ensures that the training algorithm learns features exclusively from the affected area within

the bounding box, enhancing its ability to detect significantly smaller disease spots accurately. An added advantage of this method is its high image resolution, which contributes to precision. An image input size of 640 × 640 pixels achieves the highest level of accuracy, as larger input dimensions provide more detailed information (Chen et al., 2023).

In onion disease detection, image-based approaches have primarily focused on image classification using deep learning techniques, including convolutional neural networks (CNNs) and transfer learning models (references). For example, a recent study by Jahan M. et al. (2024) explored the use of various pre-trained CNN architectures, including VGG16, VGG19, ResNet50, InceptionV3, InceptionResNetV2, and MobileNetV2, for classifying leaf diseases. Their approach involved extracting hierarchical image features using convolutional and max-pooling operations, followed by classification layers. The study also incorporated soft attention and LSTM layers to improve feature selection and sequential learning.

The primary focus of this study, however, was on the development of an onion disease object detection model for mobile camera-based applications aimed at farmers. The proposed YOLOv8-based model

TABLE 4 Results of other advanced algorithms.

Method	mAP50 (%)	Params (M)	FPS
RetinaNet R-50	70.34	36	65
Faster R-CNN R-101	72.05	44	44
YOLOv8+DTAH	77.3	11.1	123

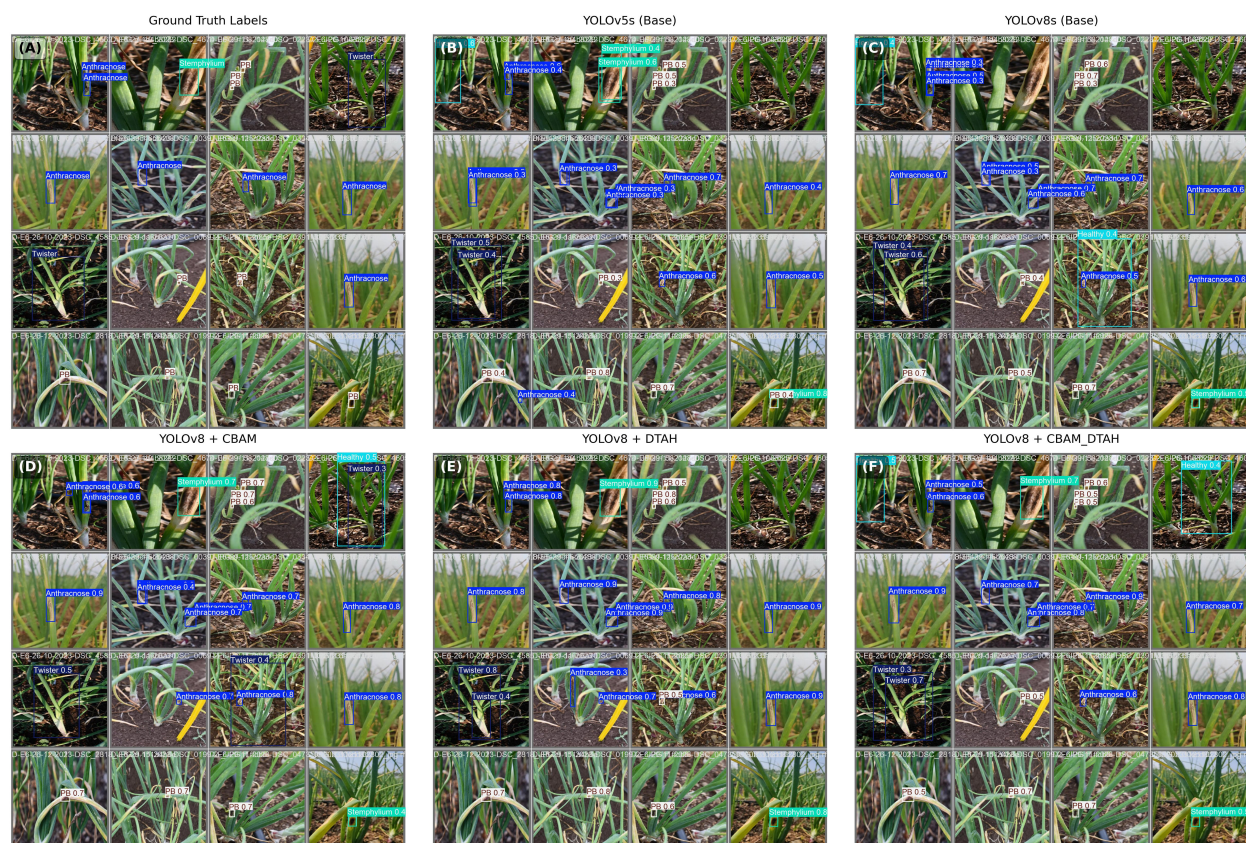


FIGURE 12
Prediction results from an object detection model.

(YOLOv8s+DTAH) can be seamlessly integrated into both Android and iOS platforms using cross-platform frameworks like TensorFlow Lite or PyTorch Mobile. These frameworks enable efficient deployment of deep learning models on mobile devices. Similar applications have already been developed for disease detection in strawberries (Van Tran et al., 2024) and tomatoes (Zeng et al., 2023) by leveraging model quantization.

In a related study, Chen et al. (2023) proposed a system that deployed a CNN-based object detection model on mobile devices using the Keras platform. This system successfully identified and localized three types of scale pests in images. Our team has already deployed YOLOv5 within the iSARATHI mobile application for onion disease detection, providing farmers with actionable disease management advisories. The YOLOv8 model in this study builds upon these advancements, forming a key component of a digital decision support system.

Beyond disease detection, deep learning (DL) and machine learning (ML) are also being applied to other areas of precision agriculture, such as weed control, soil health management, robotic harvesting, and weather analysis. The growing use of DL in agriculture has gained traction among researchers, with notable applications including YOLOv7 for weed detection using Unmanned Aerial Vehicle (UAV)-collected data (Gallo et al., 2023) and YOLOv8n and Mask R-CNN models for automatic identification and digital phenotyping in rapeseed (Wang et al.,

2023). The proposed onion disease detection model offers farmers a valuable tool for identifying affected areas, enabling more efficient disease control and better crop management.

5 Conclusion and future perspective

It is essential to monitor and manage the diseases at the right time and place for yield optimization in onion crop. A smart disease detection and identification system could facilitate trimming the yield gap, in addition to boosting the affluent lifestyle of farmers and extension workers. This study delivers YOLO-ODD an improved YOLOv8s model deployable into a smartphone app and detecting various onion diseases could be a digital guide for farmers. This model is able to distinguish between healthy and diseased onion classes and automatically detects four different kinds of diseases in onions. The approach works with real-world applications and Internet of Things (IoT) devices. The suggested algorithm can be used in a mobile application to help farmers getting help for their crops whenever they need it. This framework is adaptable to other plants and can be significantly adjusted to cater for additional crop diseases. More study initiatives could concentrate on gathering data on temperature and humidity, pathogenic inoculum, soil, and environmental parameters via various sensors, merging data from multiple sources, and developing an early warning model for onion plant diseases.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Author contributions

AR: Formal analysis, Methodology, Software, Writing – original draft, Data curation. MD: Data curation, Formal analysis, Methodology, Software, Writing – original draft, Validation, Visualization. SW: Formal analysis, Methodology, Software, Visualization, Writing – original draft, Investigation, Writing – review & editing. KK: Formal analysis, Investigation, Methodology, Writing – original draft, Writing – review & editing, Supervision, Validation. IB: Writing – original draft, Data curation, Visualization. SB: Investigation, Software, Writing – review & editing. AG: Investigation, Writing – review & editing, Supervision. RV: Writing – review & editing. MS: Writing – review & editing. SG: Writing – review & editing, Conceptualization, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Supervision, Validation, Writing – original draft.

Funding

The author(s) declare that financial support was received for the research and/or publication of this article. This work was supported by TIH foundation of IoT and IoE.

References

- Ahmed, R., and Abd-Elkawy, E. H. (2024). Improved tomato disease detection with Yolov5 and yolov8. *Eng. Technol. Appl. Sci. Res.* 14, 13922–13928. doi: 10.48084/etasr.7262
- Appe, S. N., Arulselvi, G., and Balaji, G. N. (2023). CAM-YOLO: tomato detection and classification based on improved YOLOv5 using combining attention mechanism. *PeerJ Comput. Sci.* 9, e1463. doi: 10.7717/peerj-cs.1463
- Cao, Q., Zhao, D., Li, J., Li, J., Li, G., Feng, S., et al. (2024). Pyramid-YOLOv8: a detection algorithm for precise detection of rice leaf blast. *Plant Methods* 20, 149. doi: 10.1186/s13007-024-01275-3
- Chen, J., Bai, S., Wan, G., and Li, Y. (2023). Research on YOLOv7-based defect detection method for automotive running lights. *Syst. Sci. Control Eng.* 11, 2185916. doi: 10.1080/21642583.2023.2185916
- Correa, D., Rodríguez-Reséndiz, J., Díaz-Flórez, G., Olvera-Olvera, C., and Álvarez-Alvarado, J. M. (2023). Identifying growth patterns in arid zone onion crops (*Allium cepa*) using digital image processing. *J. Technol.* 11 (3), 67. doi: 10.20944/preprints202305.0624.v1
- Feng, C., Zhong, Y., Gao, Y., Scott, M. R., and Huang, W. (2021). “Tood: Task-aligned one-stage object detection,” in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)* (Los Alamitos, CA, USA: IEEE Computer Society), 3490–3499.
- Ferentinos, K. P. (2018). Deep learning models for plant disease detection and diagnosis. *Comput. Electron. Agric.* 145, 311–318. doi: 10.1016/j.compag.2018.01.009
- Fuentes, A., Yoon, S., Kim, S. C., and Park, D. S. (2017). A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors* 17, 2022. doi: 10.3390/s17092022
- Gallo, I., Rehman, A. U., Dehkordi, R. H., Landro, N., La Grassa, R., and Boschetti, M. (2023). Deep object detection of crop weeds: Performance of YOLOv7 on a real case dataset from UAV images. *Remote Sens.* 15, 539. doi: 10.3390/rs15020539
- Gao, L., Zhao, X., Yue, X., Yue, Y., Wang, X., Wu, H., et al. (2024). A lightweight YOLOv8 model for apple leaf disease detection. *Appl. Sci.* 14, 6710. doi: 10.3390/app14156710
- Ghafar, A., Chen, C., Atif Ali Shah, S., Ur Rehman, Z., and Rahman, G. (2024). Visualizing plant disease distribution and evaluating model performance for deep learning classification with YOLOv8. *Pathogens* 13, 1032. doi: 10.3390/pathogens13121032
- Iqbal, Z., Khan, M. A., Sharif, M., Shah, J. H., ur Rehman, M. H., and Javed, K. (2018). An automated detection and classification of citrus plant diseases using image processing techniques: A review. *Comput. Electron. Agric.* 153, 12–32. doi: 10.1016/j.compag.2018.07.032
- Iren, E. (2024). Comparison of yolov5 and yolov6 models for plant leaf disease detection. *Eng. Technol. Appl. Sci. Res.* 14, 13714–13719. doi: 10.48084/etasr.7033
- Jadhav, S. B., Udipi, V. R., and Patil, S. B. (2021). Identification of plant diseases using convolutional neural networks. *Int. J. Inf. Technol.* 13, 2461–2470. doi: 10.1007/s41870-020-00437-5
- Jahan, M., Jahan, I. J., Fatema, K., Rony, A. H., Hasan, Z., Arefin, S., and Baten, A. (2024). “A robust deep learning approach for onion leaf disease classification. In *2024 IEEE International Conference on Computing, Applications and Systems (COMPAS)*, pp. 1–7. IEEE.
- Kareem, M. A., Murthy, K. V. M., Hasansab, A. N., and Waseem, M. A. (2012). Effect of temperature, relative humidity and light on lesion length due to *Alternaria porri* in onion. *Bioinfolet-A Q. J. Life Sci.* 9, 264–266.
- Kim, W. S., Lee, D. H., and Kim, Y. J. (2020). Machine vision-based automatic disease symptom detection of onion downy mildew. *Comput. Electron. Agric.* 168, 105099. doi: 10.1016/j.compag.2019.105099
- Larentzaki, E., Shelton, A. M., Musser, F. R., Nault, B. A., and Plate, J. (2007). Overwintering locations and hosts for onion thrips (Thysanoptera: Thripidae) in the onion cropping ecosystem in New York. *J. Econ. Entomol.* 100, 1194–1200. doi: 10.1093/jee/100.4.1194
- Li, R., and Wu, Y. (2022). Improved YOLO v5 wheat ear detection algorithm based on attention mechanism. *Electronics* 11 (11), 1673.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Generative AI statement

The author(s) declare that no Generative AI was used in the creation of this manuscript.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fpls.2025.1551794/full#supplementary-material>

- Lin, J., Bai, D., Xu, R., and Lin, H. (2023). TSBA-YOLO: An improved tea diseases detection model based on attention mechanisms and feature fusion. *Forests* 14, 619. doi: 10.3390/f14030619
- Liu, J., and Wang, X. (2025). EggplantDet: An efficient lightweight model for eggplant disease detection. *Alexandria Eng. J.* 115, 308–323. doi: 10.1016/j.aej.2024.12.037
- Mohanty, S. P., Hughes, D. P., and Salathé, M. (2016). Using deep learning for image-based plant disease detection. *Front. Plant Sci.* 7, 1419. doi: 10.3389/fpls.2016.01419
- Nault, B. A., and Shelton, A. M. (2010). Impact of insecticide efficacy on developing action thresholds for pest management: a case study of onion thrips (Thysanoptera: Thripidae) on onion. *J. Econ. Entomol.* 103, 1315–1326. doi: 10.1603/ec10096
- Orchi, H., Sadik, M., Khaldoun, M., and Sabir, E. (2023). June. Real-time detection of crop leaf diseases using enhanced YOLOv8 algorithm. In *2023 Int. Wireless Commun.* 1690–1696. doi: 10.1109/IWCMC58020.2023.10182573
- Patil, S., Nargund, V. B., Hariprasad, K., Hegde, G., Lingaraju, S., and Benagi, V. I. (2018). Etiology of twister disease complex in onion. *Int. J. Curr. Microbiol. Appl. Sci.* 7, 3644–3657. doi: 10.20546/ijcmas.2018.712.413
- Rajamohan, R., and Latha, B. C. (2023). An optimized YOLO v5 model for tomato leaf disease classification with field dataset. *Eng. Technol. Appl. Sci. Res.* 13, 12033–12038. doi: 10.48084/etasr.6377
- Sambasivam, G. A. O. G. D., and Opiyo, G. D. (2021). A predictive machine learning application in agriculture: Cassava disease detection and classification with imbalanced dataset using convolutional neural networks. *Egypt. Inform. J.* 22, 27–34. doi: 10.1016/j.eij.2020.02.007
- Sandhya, and Kashyap, A. (2024). Real-time object-removal tampering localization in surveillance videos by employing YOLO-V8. *J. Forensic Sci.* 69, 1304–1319. doi: 10.1111/1556-4029.15516
- Shen, Y., Yang, Z., Khan, Z., Liu, H., Chen, W., and Duan, S. (2025). Optimization of improved YOLOv8 for precision tomato leaf disease detection in sustainable agriculture. *Sensors* 25, 1398. doi: 10.3390/s25051398
- Slimani, H., El Mhamdi, J., and Jilbab, A. (2023). Artificial intelligence-based detection of fava bean rust disease in agricultural settings: an innovative approach. *Int. J. Adv. Comput. Sci. Appl.* 14 (6), 126–134. doi: 10.14569/ijacsa.2023.0140614
- Soeb, M. J. A., Jubayer, M. F., Tarin, T. A., Al Mamun, M. R., Ruhad, F. M., Parven, A., et al. (2023). Tea leaf disease detection and identification based on YOLOv7 (YOLO-T). *Sci. Reports* 13, 6078. doi: 10.1038/s41598-023-33270-4
- Sun, C., Zhou, X., Zhang, M., and Qin, A. (2023). SE-VisionTransformer: Hybrid network for diagnosing sugarcane leaf diseases based on attention mechanism. *Sensors* 23 (20), 8529.
- Too, E. C., Yujian, L., Njuki, S., and Yingchun, L. (2019). A comparative study of fine-tuning deep learning models for plant disease identification. *Comput. Electron. Agric.* 161, 272–279. doi: 10.1016/j.compag.2018.03.032
- Trinh, D. C., Mac, A. T., Dang, K. G., Nguyen, H. T., Nguyen, H. T., and Bui, T. D. (2024). Alpha-EIOU-YOLOv8: an improved algorithm for rice leaf disease detection. *AgriEngineering* 6, 302–317. doi: 10.3390/agriengineering6010018
- Tripathi, A., Gohokar, V., and Kute, R. (2024). Comparative analysis of YOLOv8 and YOLOv9 models for real-time plant disease detection in hydroponics. *Eng. Technol. Appl. Sci. Res.* 14, 17269–17275. doi: 10.48084/etasr.8301
- Van Tran, T., Do Ba, Q.-H., Tran, K. T., Nguyen, D. H., Dang, D. C., and Dinh, V.-L. (2024). Designing a mobile application for identifying strawberry diseases with YOLOv8 model integration. *Int. J. Adv. Comput. Sci.* 15 (3).
- Wang, X., and Liu, J. (2024). Vegetable disease detection using an improved YOLOv8 algorithm in the greenhouse plant environment. *Sci. Rep.* 14, 4261. doi: 10.1038/s41598-024-54540-9
- Wang, N., Liu, H., Li, Y., Zhou, W., and Ding, M. (2023). Segmentation and phenotype calculation of rapeseed pods based on YOLO v8 and mask R-convolution neural networks. *Plants* 12, 3328. doi: 10.3390/plants12183328
- Wang, Y., Wu, M., and Shen, Y. (2024). Identifying the growth status of hydroponic lettuce based on YOLO-efficientNet. *Plants* 13, 372. doi: 10.3390/plants13030372
- Wen, G., Li, M., Tan, Y., Shi, C., Luo, Y., and Luo, W. (2025). Enhanced YOLOv8 algorithm for leaf disease detection with lightweight GOCR-ELAN module and loss function: WSIOU. *Comput. Biol. Med.* 186, 109630. doi: 10.1016/j.combiomed.2024.109630
- Woo, S., Park, J., Lee, J. Y., and Kweon, I. S. (2018). “Cbam: Convolutional block attention module,” in *Proceedings of the European conference on computer vision (ECCV)* (Cham, Switzerland: Springer International Publishing), 3–19.
- Xie, H., and Ding, J. (2025). MSAM-YOLO: An improved YOLO v8 based on attention mechanism for grape leaf disease identification method. *J. Comput. Methods Sci. Eng.* doi: 10.1177/14727978251323226
- Yang, G., Wang, J., Nie, Z., Yang, H., and Yu, S. (2023). A lightweight YOLOv8 tomato detection algorithm combining feature enhancement and attention. *Agronomy* 13, 1824. doi: 10.3390/agronomy13071824
- Zaki, S. N., Ahmed, M., Ahsan, M., Zai, S., Anjum, M. R., and Din, N. U. (2021). Image-based onion disease (purple blotch) detection using deep convolutional neural network. *Int. J. Adv. Comput. Sci. Appl.* 12, 448–458. doi: 10.14569/ijacsa.2021.0120556
- Zayani, H. M., Ammar, I., Ghodhbani, R., Maqbool, A., Saidani, T., Slimane, J. B., et al. (2024). Deep learning for tomato disease detection with YOLOv8. *Eng. Technol. Appl. Sci. Res.* 14, 13584–13591. doi: 10.48084/etasr.7064
- Zhang, S., Wang, J., Yang, K., and Guan, M. (2024). YOLO-ACT: an adaptive cross-layer integration method for apple leaf disease detection. *Front. Plant Sci.* 15, 1451078. doi: 10.3389/fpls.2024.1451078
- Zeng, T., Li, S., Song, Q., Zhong, F., and Wei, X. (2023). Lightweight tomato real-time detection method based on improved YOLO and mobile deployment. *Comput Electron Agric.* 205, 107625.