



Original article

LeafMamba: A novel IoT-integrated network for accurate and efficient plant leaf disease detection

Xinming Wang, Sai Hong Tang^{ID}*, Mohd Khairul Anuar B. Mohd Ariffin,
Mohd Idris Shah B. Ismail, Ruixin Zhao

Faculty of Engineering, Universiti Putra Malaysia, Serdang, 43400, Malaysia

ARTICLE INFO

Keywords:

Plant leaf disease detection
LeafMamba network
Computational efficiency
Multi-scale information extraction
Mamba

ABSTRACT

Plant leaf disease detection plays a crucial role in agricultural production and food safety. With the rapid development of the Internet of Things (IoT) and advancements in sensor technology, plant leaf disease detection has become capable of remote and real-time monitoring, greatly enhancing the efficiency and precision of agricultural production. However, current detection methods face issues of insufficient accuracy and high model complexity, limiting their effectiveness and scalability in practical applications. To address these challenges, this paper proposes a novel network architecture, LeafMamba, which integrates the core ideas of the Mamba structure and overcomes the limitations of Transformer in complex scenarios. Through the innovatively designed LF-Mamba structure, LeafMamba efficiently captures multi-scale information and progressively enhances the spatial resolution of feature maps in the decoder, thereby improving the model's ability to extract complex disease features. Experimental results on the Plant Pathology 2020 - FGVC7 and Plant Pathology 2021 - FGVC8 datasets show that LeafMamba achieves performance of 89.5% and 92.5% in mAP@0.5, respectively, significantly outperforming existing methods. Furthermore, the model reduces the number of parameters substantially, demonstrating higher computational efficiency.

1. Introduction

Plant diseases are one of the major challenges faced in global agricultural production. They not only severely affect crop yield and quality but also threaten food security and the sustainable development of the agricultural economy [1,2]. Traditional methods of plant disease detection typically rely on manual observation, which is time-consuming and labor-intensive and is easily influenced by the subjective judgment and experience level of individuals. With the rapid development of information technology, plant disease detection techniques based on computer vision have gained widespread attention [3–5]. These techniques can quickly and accurately identify different types of diseases by analyzing leaf images, significantly improving detection efficiency and accuracy.

In the field of plant leaf disease detection, common diseases include Scab, Rust, and Multiple Diseases. Scab is a fungal disease primarily manifested by spots and scab-like lesions on the surface of leaves, which severely affect the growth of fruit trees and crops. Rust, also caused by fungi, typically appears as yellow, orange, or brown spots on leaves, leading to early leaf drop and, in severe cases, impacting the photosynthesis and yield of the crop. Multiple diseases refer to a

single plant being simultaneously affected by multiple diseases, which increases the complexity and challenges of disease detection [6,7].

The importance of plant leaf disease detection is self-evident. Firstly, accurately identifying these diseases is crucial for crop management and disease control. Early detection and identification of diseases allow for timely, targeted interventions, reducing the spread of diseases and crop losses [8,9]. Secondly, automated disease detection systems can significantly reduce dependence on agricultural experts, especially in remote areas or regions with limited agricultural resources, providing farmers and agricultural workers with timely and convenient diagnostic services. Moreover, the introduction of deep learning techniques enhances the accuracy and robustness of disease detection systems when dealing with complex diseases, such as multiple diseases [10,11].

However, current deep learning-based disease detection algorithms still encounter efficiency bottlenecks [12,13]. The increased computational complexity of these models, coupled with the large volume of data, leads to relatively slow detection speeds [14]. This inefficiency not only restricts the practical applicability of these models but also significantly hampers the deployment of disease detection systems in large-scale agricultural environments. To overcome this challenge,

* Corresponding author.

E-mail addresses: gs62785@student.upm.edu.my (X. Wang), saihong@upm.edu.my (S.H. Tang), khairul@upm.edu.my (M.K.A.B.M. Ariffin), ms_idris@upm.edu.my (M.I.S.B. Ismail), gs63714@student.upm.edu.my (R. Zhao).

<https://doi.org/10.1016/j.aej.2025.03.033>

Received 25 February 2025; Received in revised form 8 March 2025; Accepted 10 March 2025

Available online 26 March 2025

1110-0168/© 2025 The Authors. Published by Elsevier B.V. on behalf of Faculty of Engineering, Alexandria University. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

further optimization of the algorithm structure and improvements in model inference speed are essential to meet the real-time demands of practical applications [15,16].

To address the aforementioned limitations, we present the LeafMamba network, which integrates the core principles of the Mamba structure while optimizing the shortcomings of Transformers in complex scenarios. Through the innovative LF-Mamba structure, our model effectively captures multi-scale information and progressively refines the spatial resolution of feature maps in the decoder, thereby enhancing its ability to extract intricate disease characteristics. Furthermore, LeafMamba not only improves detection accuracy but also enhances computational efficiency by reducing the number of model parameters, making it suitable for both laboratory research and large-scale deployment in practical agricultural applications.

- This paper presents LeafMamba, a novel network architecture tailored for plant leaf disease detection. The model employs a multi-level, multi-scale approach to information processing, incorporating both internal and external Mamba modules to efficiently capture and represent the visual features of input images. This architecture is particularly adept at identifying subtle disease characteristics, enhancing the model's sensitivity to early or minor signs of plant pathology.
- The paper introduces the LF-Mamba block, which integrates a state space model (SSM) with an optimized self-attention mechanism. This architecture not only maintains high accuracy but also significantly reduces computational complexity. Unlike traditional self-attention mechanisms, the SSM demonstrates linear complexity growth when processing long sequence data, thereby greatly enhancing the model's computational efficiency and making it more practical for real-world applications.
- The LeafMamba model presented in this paper outperforms existing plant leaf disease detection models in experimental evaluations. These exceptional results not only validate the effectiveness of the LeafMamba architecture but also pave the way for new research and application opportunities in the field of plant disease detection.

The following is an outline of the other sections of this paper: Section 2 introduces relevant research in the field of plant leaf diseases, providing a background and theoretical foundation for this study. Section 3 first explains the theoretical framework of State Space Models (SSMs), then elaborates on the core concepts and architectural design of the LeafMamba model. Section 4 is the experimental chapter, which primarily introduces the datasets used in this paper and their preprocessing methods, and conducts a large number of comparative experiments, followed by an analysis of the results. Section 5 is the conclusion of the paper, summarizing the research findings and significance of the entire study.

2. Related work

In this section, we will first explore traditional methods of plant leaf disease detection, which are primarily based on classical image processing techniques and statistical methods. Secondly, we will delve into the application of deep learning in plant leaf disease detection, specifically examining how concepts from YOLO, Transformer, and Mamba can be utilized to enhance the accuracy and efficiency of detection.

2.1. Traditional methods in leaf disease detection

Traditional methods for detecting plant leaf diseases typically rely on manual observation and expert judgment, where agricultural specialists visually inspect the leaves of crops to identify the type and severity of the disease [17–19]. Although intuitive, this approach is limited by human subjectivity and is heavily influenced by the observer's

level of experience. Moreover, it is time-consuming and labor-intensive to implement in large-scale agricultural environments [20,21]. Additionally, this method is prone to misdiagnosis or oversight, particularly when dealing with the simultaneous presence of multiple diseases or complex symptoms [22,23]. To enhance detection efficiency, several traditional image processing-based methods have emerged in recent years. These methods rely on manually extracted features such as color histograms, shape features, texture features, and edge features. Researchers typically use various image processing techniques, including thresholding, edge detection, and morphological operations, to extract features from the diseased regions in leaf images. These extracted features are then classified using conventional machine learning algorithms such as Support Vector Machines (SVM) [24,25], K-Nearest Neighbors (K-NN) [26], Naive Bayes classifiers [27], and decision trees [28]. These methods can improve the automation of detection to some extent, reducing dependence on manual observation. However, plant disease detection based on traditional methods has several significant limitations. Firstly, these methods are highly dependent on image quality and environmental conditions, such as variations in lighting, background complexity, and noise interference, which can affect detection accuracy [29]. Secondly, the representational capacity of manually extracted features is limited, failing to capture the complex patterns and subtle variations of diseases adequately. Furthermore, traditional methods usually only recognize specific types of diseases and are significantly less effective at detecting multiple co-occurring diseases or new types of diseases [30]. With the rise of deep learning technologies, these limitations have prompted researchers to seek more intelligent and automated solutions.

2.2. Deep learning for plant leaf disease detection

In recent years, deep learning technology has made significant strides in the field of plant leaf disease detection, particularly through the use of convolutional neural networks (CNNs) and their enhanced models, which excel in image recognition tasks. Detection models based on the YOLO (You Only Look Once) series, such as YOLOv8 [31] and YOLOv9 [32], have gained widespread application in plant leaf disease detection. These models offer end-to-end detection capabilities, enabling simultaneous object localization and classification in a single forward pass, which greatly enhances both detection efficiency and accuracy. YOLOv8 has improved feature extraction and optimized network structures, making it more effective in handling plant leaf disease detection tasks, especially in complex backgrounds. YOLOv9 further refines the model's precision and robustness, excelling particularly in detecting small objects and handling multi-object scenarios.

At the same time, detection models based on Transformers, such as DETR (Detection Transformer) [33] and RT-DETR (Real-Time Detection Transformer) [34], have also started to be applied in plant disease detection. DETR effectively captures long-range dependencies in images through a self-attention mechanism, achieving high detection accuracy even under complex backgrounds and occlusion conditions [35]. RT-DETR builds on DETR with optimizations that significantly enhance real-time detection, making it more suitable for agricultural applications that require rapid response.

Additionally, the widely used transfer learning and fine-tuning techniques in glaucoma and COVID-19 detection provide valuable insights for plant disease detection [36,37]. In COVID-19 detection, deep learning models based on ensemble CNN and hybrid LSTM-CNN perform excellently, offering rapid and accurate predictions through the analysis of chest X-ray images. This is similar to plant disease detection, where models pre-trained on large-scale datasets and fine-tuned for specific agricultural scenarios can significantly enhance the model's adaptability and accuracy.

Despite the significant progress of these deep learning methods in plant leaf disease detection, they still have some limitations. One major challenge is the slow computational speed. While models like YOLOv8

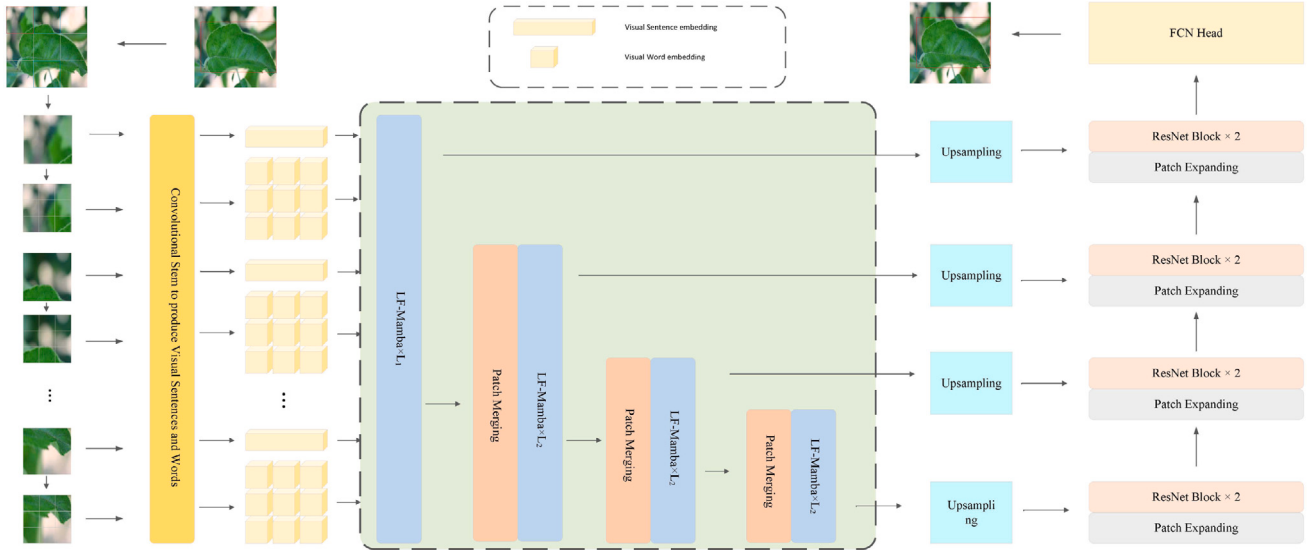


Fig. 1. LeafMamba network architecture.

and YOLOv9 have improved accuracy, their complex network structures and large number of parameters result in relatively slow inference speeds, which do not meet the real-time requirements of large-scale agricultural environments [38]. Similarly, although Transformer-based models perform well in complex scenarios, their self-attention mechanisms have high computational complexity, leading to long inference times. These computational speed bottlenecks limit the practical deployment and use of these algorithms in large-scale agricultural applications. Therefore, optimizing model structures to improve computational efficiency while maintaining high accuracy is a key research direction for applying deep learning in plant leaf disease detection.

3. Method

In this section, we begin by reviewing the fundamental concepts of State Space Models and Vision Mamba. We then introduce the LeafMamba network architecture proposed in this paper and provide a detailed description of the LF-Mamba block's composition. Lastly, we analyze the complexity of the LeafMamba model.

3.1. Preliminaries

State Space Models. State Space Models (SSMs) [39] map input $x(t) \in \mathbb{R}^L$ to output $y(t) \in \mathbb{R}^L$ through internal states $h(t) \in \mathbb{R}^N$, modeled by linear ordinary differential equations (ODEs). The key parameters include the state transition matrix A , projection vectors B and C , and skip connection D .

$$\begin{aligned} h'(t) &= Ah(t) + Bx(t) \\ y(t) &= Ch(t) + Dx(t) \end{aligned} \quad (1)$$

SSMs due to their continuous-time nature, pose significant challenges when applied to deep learning scenarios. To effectively integrate SSMs, it is often necessary to convert their described ordinary differential equations (ODEs) into discrete-time form. Eq. (2) demonstrates this conversion method. In this form, the discrete-time hidden state h_k is updated using the previous time step's state h_{k-1} and the current input x_k . Here, the transformed state transition matrix is $\bar{A} = e^{A\Delta t}$, and the input projection matrix is $\bar{B} = (e^{A\Delta t} - I)A^{-1}B$, where I is the identity matrix representing the transition from continuous to discrete time during discretization. The projection matrix $\bar{C} = C$ remains unchanged and is used to map the hidden state $h(t)$ to the output $y(t)$.

$$\begin{aligned} h_k &= \bar{A}h_{k-1} + \bar{B}x_k, \quad y(t) = Ch(t) + Dx(t) \\ \bar{A} &= e^{A\Delta t}, \quad \bar{B} = (e^{A\Delta t} - I)A^{-1}B, \quad \bar{C} = C \end{aligned} \quad (2)$$

During the discretization of state space models, to further simplify the representation of the input projection matrix, matrix $\bar{B}$ can be expanded using a Taylor series, as shown in the equation:

$$\bar{B} = (e^{A\Delta t} - I)A^{-1}B = (\Delta A)(\Delta A)^{-1}\Delta B = \Delta B \quad (3)$$

In Eq. (3), the matrix $e^{A\Delta t}$ represents the exponential form of the state transition matrix, and I is the identity matrix. By re-expressing matrix $\bar{B}$, we utilize the properties of matrix inversion and multiplication to further simplify $\bar{B}$ to ΔB . This process effectively simplifies the representation of the discretized state space model, making the model more efficient in handling the input projection matrix while maintaining system stability and accuracy.

Vision Mamba. SSMs have also achieved great success in the field of computer vision. A representative work is Vision Mamba [40], which, when processing visual tasks, first transforms a two-dimensional image $t \in \mathbb{R}^{H \times W \times C}$ into a flattened 2D patch representation $x_p \in \mathbb{R}^{J \times (P^2 \cdot C)}$, where (H, W) are the dimensions of the input image, C is the number of channels, and P is the size of each image patch. Next, x_p is linearly projected onto a vector of size D , and positional embeddings $E_{pos} \in \mathbb{R}^{(J+1) \times D}$ are added, as shown below:

$$\mathbf{T}_0 = [t_{cls}, t_p^1 \mathbf{W}, t_p^2 \mathbf{W}, \dots, t_p^J \mathbf{W}] + \mathbf{E}_{pos}, \quad (4)$$

where t_p^j is the j th patch of t , and $\mathbf{W} \in \mathbb{R}^{(P^2 \cdot C) \times D}$ is a learnable projection matrix. Vision Mamba normalizes the class token $\mathbf{T}_0^L$ and uses a Multi-Layer Perceptron to generate the final prediction $\mathbf{T}_0^p$.

$$\begin{aligned} \mathbf{T}_l &= \text{Vim}(\mathbf{T}_{l-1}) + \mathbf{T}_{l-1} \\ f &= \text{Norm}(\mathbf{T}_L^0) \\ \hat{p} &= \text{MLP}(f) \end{aligned} \quad (5)$$

where Vim is the Vision Mamba block, L is the number of layers, and Norm is the normalization layer.

3.2. LeafMamba network

As shown in Fig. 1, the proposed LeafMamba Network adopts a multi-scale and hierarchical approach to effectively capture and represent visual features from input images, aiming for accurate detection and classification of plant leaf diseases. The network structure begins with an input image of size $3 \times H \times W$, where a series of transformations are applied to generate feature embeddings that serve as the foundation for subsequent detection and classification tasks.

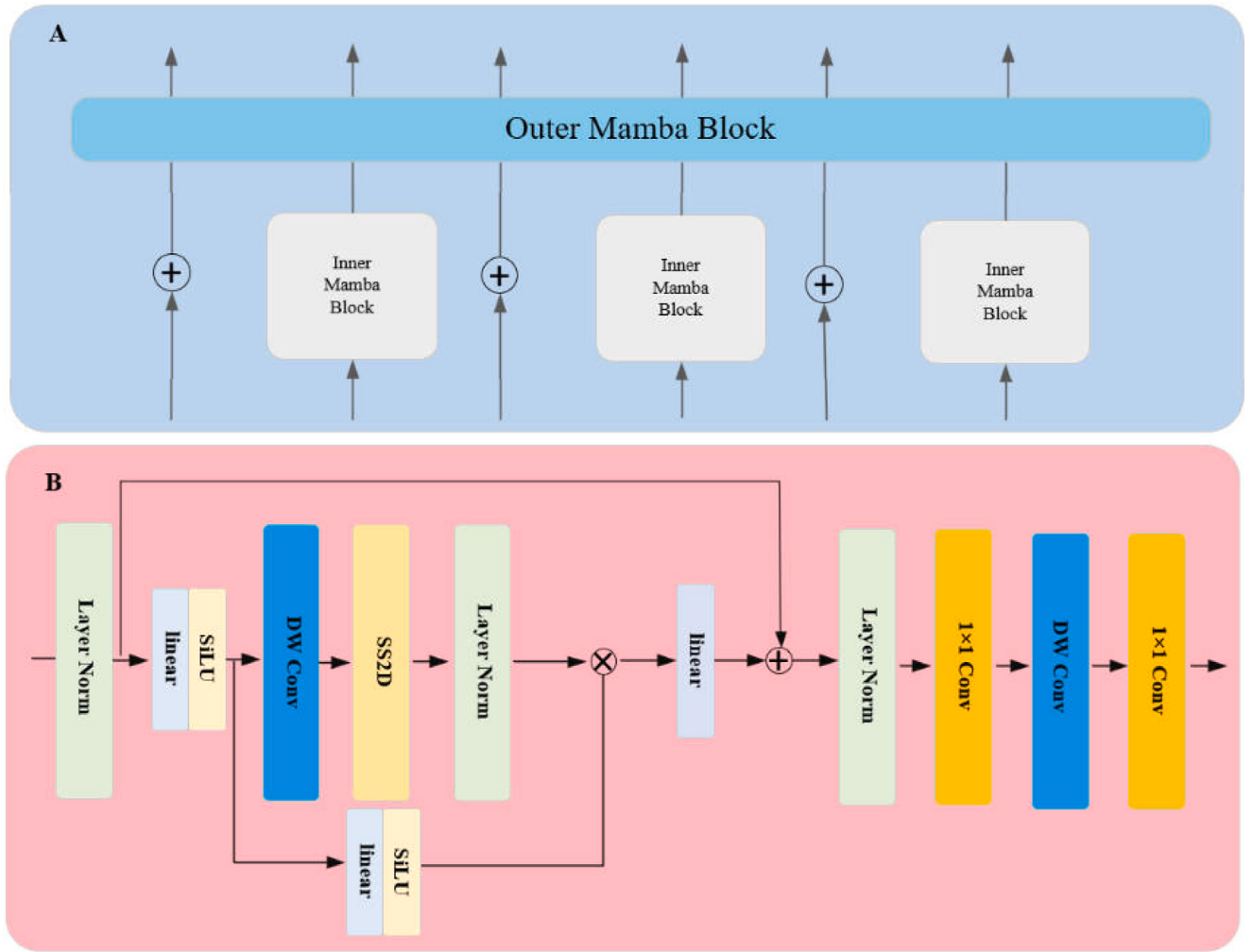


Fig. 2. LF-Mamba. (A) This paper introduces the LF-Mamba network, which consists of internal and external Mamba modules; (B) the specific structure of these modules is illustrated.

First, the network processes the input image through a Convolutional Stem module to generate “visual sentences” and “visual words”. These visual tokens are extracted through multiple convolution operations to capture local textures and patterns, which are crucial for identifying subtle variations in plant leaf diseases. The core of the network consists of four stages of LF-Mamba modules: L_1 , L_2 , L_3 , and L_4 . Each stage contains multiple Mamba-in-Mamba modules that gradually extract deeper and more abstract features from the visual embeddings. By focusing on different scales and details of the input data, these modules enhance the feature extraction capability and effectively capture complex patterns associated with various plant diseases. Between each stage, Patch Merging layers are employed to down-sample the feature maps, reducing spatial resolution while increasing the depth of feature representations, thus learning more compact and meaningful features. Before the final prediction, the network uses up-sampling modules to progressively restore the spatial resolution of the feature maps, ensuring that the output retains the necessary details for accurate localization. The features output by each stage are further refined through Residual Blocks, which use skip connections to retain critical information from earlier layers. Subsequently, Patch Expanding modules are used to up-sample the feature maps to a higher resolution, preserving spatial details for precise disease localization. Finally, the interpolated features are fed into the FCN (Fully Convolutional Network) head, which adjusts the output to the original input resolution of $1 \times H \times W$, generating the final locations and classifications of the detected diseases.

The LeafMamba Network ensures comprehensive detection capability through multi-scale and multi-level information integration, from visual word embedding to full image segmentation. By combining LF-Mamba modules, patch merging, and up-sampling modules, the network efficiently and accurately handles various types of leaf diseases, even in complex backgrounds.

LF-Mamba. The core of the LF-Mamba Network is a hierarchical encoder composed of four stages, each containing a different number of tokens. These stages are designed to progressively encode input image features into different levels of abstraction. The spatial shape of the visual word embeddings is gradually reduced across the stages to $H/2 \times W/2$, $H/4 \times W/4$, $H/8 \times W/8$, and $H/16 \times W/16$ to effectively capture multi-scale information. Patch merging serves as a down-sampling operation between each stage, reducing the spatial resolution while increasing the depth of feature representation to better extract complex features related to leaf diseases. The visual words $X^{i,j}$ are mapped to a sequence of word embeddings $w^{i,j}$ via linear projection, expressed as:

$$W^i = [w^{i,1}, w^{i,2}, \dots, w^{i,m}], \quad w^{i,j} = FC(Vec(x^{i,j})) \quad (6)$$

where W^i is the set of word embeddings for the i th visual word, $w^{i,j}$ represents the embedding of the j th patch in the i th stage, FC denotes the fully connected layer, and $Vec(x^{i,j})$ is the vectorized representation of the j th patch $x^{i,j}$. As illustrated in Fig. 2(A), the LF-Mamba module consists of two main components: one is designed to traverse visual sentences, ensuring that global contextual information is fully captured

and conveyed throughout the network; the other is responsible for managing the visual words within each sentence, allowing for detailed extraction of local features. The first component primarily utilizes a series of self-attention mechanisms to capture global information within the visual sentences, which helps the network understand the overall structure and general layout of the image. The second component focuses on extracting fine-grained features by processing visual words individually, enhancing the understanding and capture of local details, such as subtle variations and edge characteristics associated with leaf diseases. The formula is expressed as:

$$\begin{aligned} W_l^{i'} &= W_{l-1}^i + \text{Mamba}(\text{LN}(W_{l-1}^i)) \\ W_l^i &= W_l^{i'} + \text{Conv}_{FFN}(\text{LN}(W_l^{i'})) \end{aligned} \quad (7)$$

where W_l^i is the output of the i th visual word at layer l , W_{l-1}^i represents the input from the previous layer, $\text{LN}(\cdot)$ denotes the layer normalization operation, $\text{Mamba}(\cdot)$ is the function representing the Mamba operation on the normalized input, and $\text{Conv}_{FFN}(\cdot)$ is the convolutional feed-forward network applied to the normalized intermediate output $W_l^{i'}$.

Additionally, we generate sentence embedding memory as a storage mechanism for information. In Eq. (8), S_{l-1}^i represents the embedding of the i th sentence at layer $l-1$, while W_l^i denotes the embedding of the i th visual word at layer l . The visual word embedding W_l^i is mapped to a vector representation through a fully connected layer (FC). After concatenation, this results in a new sentence embedding. This sentence embedding memory mechanism allows the network to efficiently store and propagate contextual information, enhancing its understanding and representation of the complex features associated with plant leaf diseases.

$$S_{l-1}^i = S_{l-1}^i + \text{FC}(\text{Vec}(W_l^i)), \quad l = 1, 2, \dots, L \quad (8)$$

To further enhance word-level features, we employ the Outer Mamba module. This module strengthens the representation of local features by further processing visual word embeddings at each layer. The formula is:

$$\begin{aligned} S_l' &= S_{l-1} + \text{Mamba}(\text{LN}(S_{l-1})) \\ S_l &= S_l' + \text{Conv}_{FFN}(\text{LN}(S_l')) \end{aligned} \quad (9)$$

In the LF-Mamba block, both the inner and Outer Mamba blocks adopt the structural design of the Vision Mamba block and are closely integrated with a Convolutional Feed-Forward Network after each Vision Mamba block, as illustrated in Fig. 2(B). Specifically, the Vision Mamba blocks combines various operational units, including Layer Normalization, Linear Transformation, activation function (SiLU), Depthwise Separable Convolution, and Space-to-Depth Convolution. These units sequentially extract and enhance the representation of input features, improving the network's ability to capture and understand complex image patterns. The Convolutional Feed-Forward Network further optimizes the depth and breadth of feature processing, consisting of 1×1 convolutions, 3×3 Depthwise Separable Convolutions, Batch Normalization, and the GELU activation function. This design effectively captures multi-scale and multi-dimensional information while maintaining computational efficiency and the lightweight characteristics of the model, thereby significantly enhancing the inference speed while ensuring high detection accuracy.

Decoder Structure. In the decoder structure, a patch expanding layer is used for the up-sampling operation. This layer gradually increases the spatial resolution of the feature maps, allowing the network to recover fine details lost during down-sampling in the earlier layers. The output of the decoder is combined with the expanded feature set, effectively integrating high-level semantic information with low-level spatial details. Finally, these fused features are further processed through a fully connected network layer, and an interpolation operation adjusts the output to match the resolution of the original input, ensuring accurate localization and classification of plant leaf diseases.

3.3. Complexity analysis

For a visual sequence $T \in \mathbb{R}^{1 \times n \times d}$, the computational complexity of the SSMs is given by $128nd$, where N denotes the dimension of the SSM, typically set to 16, and $E = 2d$. In contrast, the self-attention mechanism has a complexity of $4nd^2 + 2n^2d$. Clearly, the complexity of self-attention grows quadratically with the sequence length n , whereas the SSM maintains a linear growth rate.

In this paper, the FLOPs for all Inner Mamba blocks and the Outer Mamba blocks within the LF-Mamba blocks are calculated as $128mnc + 3mnc^2$ and $128nd + 3nd^2$, respectively, where m is the number of visual words in a visual sentence, and c represents the word embedding dimension. Thus, the total FLOPs for the LF-Mamba block, denoted as $FLOPs_{MIM}$, sum up to $128mnc + 128nd + 3mnc^2 + 3nd^2$, maintaining an overall linear complexity. This efficiency is achieved despite the potentially large number of parameters, ensuring scalability for practical applications across various domains.

4. Experiments

4.1. Experimental setup

Training Data. We utilized two publicly available plant pathology datasets: Plant Pathology 2020 - FGVC7 [41] and Plant Pathology 2021 - FGVC8 [42]. The Plant Pathology 2020 - FGVC7 dataset comprises 3644 high-quality annotated RGB images, while the dataset from the Plant Pathology 2021 - FGVC8 competition contains 3651 RGB images, primarily used for identifying diseases on apple leaves. Specifically, these images exhibit typical symptoms of cedar apple rust (Fig. 3A) and apple scab (Fig. 3B). Additionally, the datasets include images of leaves showing complex disease symptoms (i.e., more than one disease present on the same leaf; Fig. 3C) and healthy apple leaves (Fig. 3D).

In our study, we divided the datasets into training and testing sets with a ratio of 7:3. The training set was used to train the model to learn the characteristics of different diseases, while the testing set was employed to evaluate the model's performance and verify its generalization ability on unseen data. This approach ensures that the model achieves high accuracy and robustness when handling new data in real-world applications.

Data Preprocessing. To enhance the model's generalization ability and prevent overfitting, we employed various data augmentation techniques in this study. The primary augmentation methods include horizontal flipping, vertical flipping, and image blurring. Fig. 4 displays two different flipping operations: Image A and Image B respectively show the effects of vertical and horizontal flips on the Plant Pathology 2020 - FGVC7 and Plant Pathology 2021 - FGVC8 datasets. Fig. 5 then demonstrates the outcomes of image blurring applied to these datasets.

As shown in Eq. (10), suppose an image is represented as a matrix A_{ijk} , where i and j denote the rows and columns of the image, respectively, and k represents the number of channels. The horizontal flip operation reverses each row of the image from left to right, specifically mapping the matrix element A_{ijk} to $A_{i(n+1-j)k}$, where n is the width of the image. Similarly, the vertical flip operation reverses each column of the image from top to bottom, mapping the matrix element A_{ijk} to $A_{(m+1-i)jk}$, where m is the height of the image.

$$\text{Image} = A_{ijk}$$

$$\text{Horizontal flip} : A_{ijk} \rightarrow A_{i(n+1-j)k} \quad (10)$$

$$\text{Vertical flip} : A_{ijk} \rightarrow A_{(m+1-i)jk}$$

For the blurring method, we apply a Gaussian filter to the image to smooth out details and reduce noise. This process is defined as convolving the image A_{ijk} with a Gaussian kernel G_σ , where σ represents the standard deviation of the Gaussian distribution. The blurred image is then represented as:

$$\text{Blurring} : A_{ijk} \rightarrow (A * G_\sigma)_{ijk} \quad (11)$$

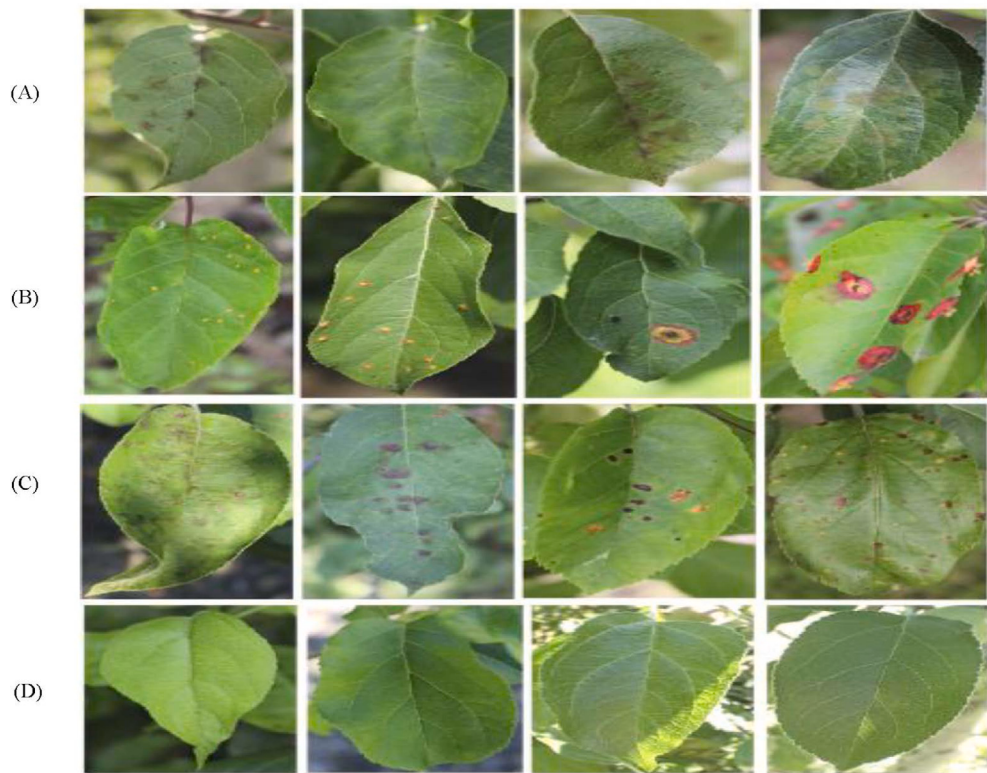


Fig. 3. The sample images in the dataset show: (A) Cedar apple rust (B) Apple scab (C) Multiple diseases on a single leaf (D) Healthy leaf.

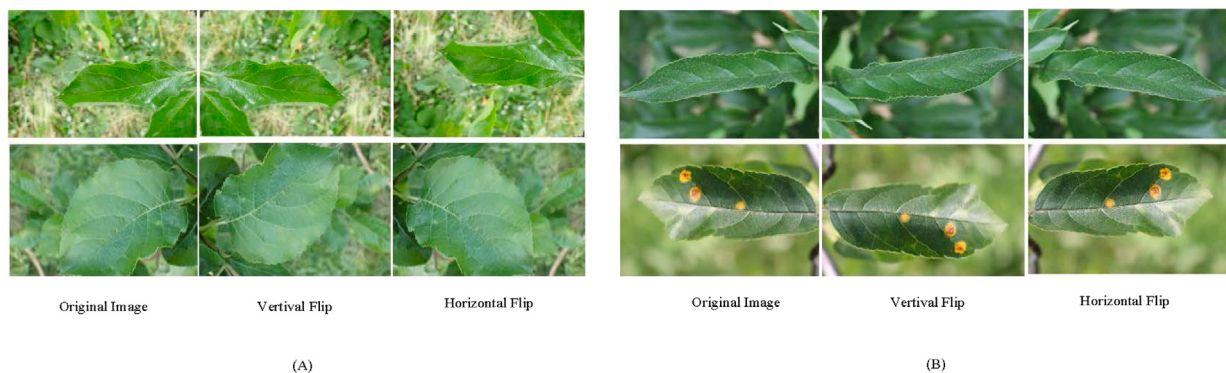


Fig. 4. Data augmentation techniques demonstrated on leaf images. Panel (A) shows the original image, vertical flip, and horizontal flip applied to a leaf from the Plant Pathology 2020 - FGVC7 dataset. Panel (B) illustrates the same set of transformations on a leaf afflicted with spots, sourced from the Plant Pathology 2021 - FGVC8 dataset. These transformations are typical augmentation methods used to enhance the robustness and generalization of machine learning models in plant disease detection.

Through flipping and blurring the data, we significantly increased the diversity of the samples, thereby enhancing the model's generalization capability. Specifically, after these data augmentation processes, the training set of Plant Pathology 2020 - FGVC7 includes a total of 7890 images, while the test set comprises 3450 images. Similarly, the training set for Plant Pathology 2021 - FGVC8 expanded to 8230 images, with the test set containing 4520 images. These data augmentation strategies ensure the robustness of the model under various conditions and changes, helping to improve diagnostic accuracy and reliability.

Experimental Environment. This study was conducted in a high-performance computing environment to ensure the efficiency and effectiveness of model training and evaluation. The experiments were executed on a computing platform with the following configuration: the operating system was Ubuntu 20.04 LTS, with an Intel Xeon Gold 6248R CPU @ 3.00 GHz, equipped with 48 physical cores, and 256 GB of RAM. The Graphics Processing Unit (GPU) used was the NVIDIA

Tesla V100 32 GB, supporting CUDA 11.2 and cuDNN 8.1, to accelerate the computation of deep learning models. The primary frameworks used for development and experimentation were Python 3.8.10 and PyTorch 1.9.0, with data preprocessing, model training, and result analysis performed through Jupyter Notebook.

To ensure the reproducibility of our experiments, we set a fixed random seed (42) for both training and testing, guaranteeing consistency in data splitting and model initialization. The training process utilized the Adam optimizer with an initial learning rate of 0.001, which was progressively reduced using a cosine annealing scheduler to facilitate smooth convergence. A batch size of 32 was employed, and the model was trained for 100 epochs to ensure comprehensive learning from the dataset. To mitigate overfitting, early stopping and weight decay regularization were implemented, with the weight decay factor set to $1e-5$.

Evaluation metrics. In this study, we employ several key metrics to quantify model performance, including Precision, Recall, and mean



Fig. 5. Illustration of image blurring as a data augmentation technique for leaf disease detection. Panel (A) demonstrates the effects of image blurring on a sample from the Plant Pathology 2020 - FGVC7 dataset. Panel (B) shows the blurring effects on a sample from the Plant Pathology 2021 - FGVC8 dataset.

Table 1

Comparison of model performance on Plant Pathology 2020 - FGVC7.

Models	mAP@0.5%	mAP@0.5-0.95%	Parameters	Model size/MB	FLOPs/G	FPS
RetinaNet [43]	68.3%	28.7%	4,570,023	9.1	14.4	126.6
EfficientDet [44]	71.7%	31.8%	5,852,863	10.5	13.1	144.9
M2Det [45]	73.5%	32.3%	4,863,463	9.7	12.1	137.9
SSD [46]	80.4%	35.8%	4,781,031	9.5	11.9	136.2
Cascade RCNN [47]	78.5%	26.8%	5,524,356	10.8	11.9	127.6
Faster RCNN [48]	82.1%	32.1%	4,872,579	9.2	11.9	127.3
YOLOv5s [49]	86.7%	37.8%	7,037,095	13.8	15.8	156.9
YOLOv5n [50]	84.5%	28.5%	2,772,695	7.7	6.2	142.3
YOLOv8s [51]	89.1%	35.8%	7,879,630	14.5	14.6	156.3
YOLOv8n [31]	88.7%	38.7%	3,152,334	6.6	8.5	142.5
Vision Mamba [40]	84.3%	39.5%	1,125,471	4.4	7.8	175.3
Mamba-in-Mamba [39]	85.4%	38.7%	1,387,524	4.3	6.4	165.8
Ours	89.5%	41.5%	994,478	3.8	5.3	178.3

Average Precision (mAP) at an Intersection over Union (IoU) threshold of 0.5. Here are detailed explanations of these metrics:

Precision (P%): Precision measures the proportion of correctly predicted positive observations to the total predicted positives. The formula for precision is:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (12)$$

where TP stands for true positives, the number of correct positive predictions, and FP stands for false positives, the number of incorrect predictions as positive.

Recall (R%): Recall measures the proportion of actual positives correctly identified by the model. The formula for recall is:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (13)$$

where FN stands for false negatives, the number of actual positives that were not detected by the model.

4.2. Comparisons with state-of-the-art detection methods

Performance Analysis on Plant Pathology 2020 - FGVC7 dataset

Table 1 provides a comprehensive performance comparison of our model against other leading models on the Plant Pathology 2020 - FGVC7 dataset. Our model achieves leading scores of 89.5% and 41.5% on key performance metrics mAP@0.5% and mAP@0.5–0.95%, respectively, surpassing YOLOv8s by 0.4% and 5.7%, and YOLOv5s by 2.8% and 3.7%. This improvement demonstrates our model's efficient capability in accurately recognizing a variety of plant diseases,

particularly its significant lead on the more stringent mAP@0.5–0.95% metric, showcasing its ability to capture subtle features. Compared to other models such as EfficientDet and SSD, which also perform well on mAP@0.5% with scores of 71.7% and 80.4% respectively, our model still exhibits superior performance on mAP@0.5–0.95%. Additionally, from a technical implementation perspective, our model has the optimal number of parameters and model size among all listed models, at 994,478 and 3.8 MB respectively, which significantly reduces resource consumption compared to larger models like YOLOv8s and YOLOv5s. In terms of frames per second (FPS), although our model's 178.3 FPS is slightly lower than YOLOv5s's 156.9 FPS and YOLOv8s's 156.3 FPS, its superior accuracy and lower resource demands make it highly feasible and efficient for practical applications. Particularly in resource-limited environments, the lower number of parameters and smaller model size make our model more suitable for deployment.

Performance Analysis on Plant Pathology 2021 - FGVC8. As demonstrated in Table 2, our model exhibits significant superiority on the Plant Pathology 2021 - FGVC8 dataset, particularly in the mAP@0.5–0.95% metric. Achieving a score of 44.5%, our model outperforms all other competitors, including the industry benchmark YOLOv8s, which attains 40.8% on the same metric. This result highlights our model's enhanced precision and adaptability in handling diverse disease types and complex image backgrounds. Furthermore, compared to YOLOv5s, which achieves an mAP@0.5% of 89.7%, our model surpasses it by 4.9 percentage points on the more rigorous mAP@0.5–0.95% metric. This performance advantage underscores our model's ability to accurately distinguish various plant diseases, particularly in identifying early-stage or minor infections. Against models

Table 2
Comparison of model performance on Plant Pathology 2021 - FGVC8.

Models	mAP@0.5%	mAP@0.5-0.95%	Parameters	Model size/MB	FLOPs/G	FPS
RetinaNet [43]	78.4%	35.7%	4,570,023	9.1	14.4	138.5
EfficientDet [44]	75.6%	37.8%	5,852,863	10.5	13.1	144.9
M2Det [45]	83.4%	37.3%	4,863,463	9.7	12.1	137.9
SSD [46]	85.7%	34.8%	4,781,031	9.5	11.9	136.2
Cascade RCNN [47]	88.6%	36.8%	5,524,356	10.8	11.9	157.4
Faster RCNN [48]	82.1%	38.1%	4,872,579	9.2	11.9	137.5
YOLOv5s [49]	89.7%	39.6%	7,037,095	13.8	15.8	176.9
YOLOv5n [50]	85.4%	37.5%	2,772,695	7.7	6.2	152.3
YOLOv8s [51]	84.8%	40.8%	7,879,630	14.5	14.6	166.3
YOLOv8n [31]	90.7%	37.7%	3,152,334	6.6	8.5	143.5
Vision Mamba [40]	88.3%	39.2%	1,125,471	4.4	7.8	185.1
Mamba-in-Mamba [39]	87.4%	37.7%	1,387,524	4.3	6.4	178.3
Ours	92.5%	44.5%	994,478	3.8	5.3	187.5



Fig. 6. Leaf disease detection results display. (A) The image shows healthy leaves, (B) detects scab disease, and (C) identifies apple rust.

such as Cascade RCNN and M2Det, which perform well under certain conditions, our model demonstrates a substantial lead of 7.7% and 7.2%, respectively, in the mAP@0.5–0.95% metric, further confirming its overall superiority. This performance gain is attributed to the deep feature extraction and optimization strategies embedded in our model, enabling it to maintain high efficiency and accuracy across a broader range of scenarios. The results on the FGVC8 dataset not only statistically outperform existing models but also demonstrate our model’s exceptional generalization capability and processing efficiency in real-world, complex conditions. These findings underscore the advanced architectural design of our model and its immense potential for practical applications, particularly in agricultural disease detection, where high accuracy and rapid response are essential.

4.3. Qualitative comparison

Fig. 6 shows the leaf disease detection results of the LeafMamba model on a dataset. Panel A identifies healthy leaves, Panel B detects scab disease, and Panel C identifies apple rust. The figure demonstrates that our model exhibits high accuracy and reliability under various disease conditions. In Panel A, the LeafMamba model successfully recognizes and marks all healthy leaves, with no false positives. Panels B and C display the model’s effectiveness in detecting complex conditions such as scab and apple rust diseases. Even in cases where the diseased areas on the leaves are small or the symptoms are not pronounced, LeafMamba accurately locates and identifies these areas,

proving its high sensitivity and diagnostic ability for different types of leaf diseases. These results not only demonstrate the model’s efficiency but also highlight its potential in practical agricultural applications, offering farmers a powerful tool to reduce crop loss and enhance the precision of crop management.

4.4. Limitations and future work

Although this study has made notable progress in the field of plant pathology, the limitations of the dataset remain a significant challenge. While the current dataset covers various types of plant diseases, it primarily focuses on specific regions and plant species, failing to represent the full global diversity of plant diseases. This limitation may hinder the model’s generalizability and applicability, making it challenging to adapt to different regions or previously unseen types of plant diseases.

To address these limitations and further improve the model, future work will focus on the following areas:

1. We aim to expand the dataset by collecting and integrating plant disease data from a broader geographical range, with an emphasis on underrepresented regions and plant types. Increasing the diversity of the dataset will significantly enhance the model’s robustness and adaptability, thus improving its utility in global agricultural production.
2. With the continued advancement of smart agriculture technologies, integrating our disease detection model with IoT devices

will be a key area of future research. Specifically, developing lightweight models optimized for edge computing will facilitate real-time monitoring and data analysis in the field, enabling farmers to receive immediate disease management recommendations.

3. Given the need for continuous optimization and adaptive learning, we plan to explore advanced machine learning techniques, including transfer learning, federated learning, and reinforcement learning. These approaches will enable the model to learn effectively with limited labeled data while improving its performance in complex and dynamic environments.

Another notable limitation is the model's sensitivity to environmental changes. While it performs well in controlled testing environments, factors such as varying lighting conditions, weather fluctuations, and background noise in real-world applications can compromise its accuracy. This sensitivity hinders the model's generalizability and reliability across diverse agricultural settings.

To address this challenge, future research should aim to enhance the model's adaptability to diverse environmental conditions. Incorporating advanced data augmentation techniques — such as the introduction of random noise, lighting variations, and simulated weather conditions — along with the development of novel algorithms to improve robustness against these dynamic factors, could significantly enhance the model's performance and reliability in real-world applications. Furthermore, integrating scene recognition technology to inform disease detection will allow the model to adjust its diagnostic strategies based on varying environmental contexts, thereby improving its overall effectiveness and dependability in practical settings.

5. Conclusion

In this paper, we introduce the LeafMamba Network, which significantly improves upon the traditional Transformer model's inefficiencies and high resource consumption when processing long sequence data. By integrating the design philosophy of Mamba, the LeafMamba Network not only enhances the depth and breadth of information processing but also significantly reduces dependency on computational resources and increases data processing speed through its innovative structure. This enables the network to handle long sequence images more efficiently, accurately extracting key features from complex backgrounds, and demonstrating its strong capabilities in practical applications. Experimental results show that the LeafMamba Network performs excellently on multiple standard datasets, especially in processing high-resolution plant images with complex backgrounds, where it can precisely locate and identify different types of plant diseases. This not only demonstrates its superior performance in traditional tasks but also validates its potential and practicality in real agricultural applications. Moreover, we plan to further expand the application scope of the LeafMamba Network, exploring its effectiveness in broader agricultural and ecological environment monitoring fields. Through these continued efforts, we hope that the LeafMamba Network will make a greater contribution to global plant disease management and sustainable agricultural development in the future.

CRedit authorship contribution statement

Xinming Wang: Writing – original draft, Validation, Supervision.
Sai Hong Tang: Writing – review & editing, Methodology, Formal analysis, Conceptualization.
Mohd Khairul Anuar B. Mohd Ariffin: Writing – review & editing, Software, Data curation.
Mohd Idris Shah B. Ismail: Writing – review & editing, Software, Formal analysis.
Ruixin Zhao: Validation, Supervision, Data curation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

- [1] S. Ashok, G. Kishore, V. Rajesh, S. Suchitra, S.G. Sophia, B. Pavithra, Tomato leaf disease detection using deep learning techniques, in: 2020 5th International Conference on Communication and Electronics Systems, ICCES, IEEE, 2020, pp. 979–983.
- [2] S.S. Harakannanavar, J.M. Rudagi, V.I. Puranikmath, A. Siddiqua, R. Pramodhini, Plant leaf disease detection using computer vision and machine learning algorithms, *Glob. Transitions Proc.* 3 (1) (2022) 305–310.
- [3] R. Kundu, U. Chauhan, S. Chauhan, Plant leaf disease detection using image processing, in: 2022 2nd International Conference on Innovative Practices in Technology and Management, ICIPTM, 2, IEEE, 2022, pp. 393–396.
- [4] M.E. Chowdhury, T. Rahman, A. Khandakar, M.A. Ayari, A.U. Khan, M.S. Khan, N. Al-Emadi, M.B.I. Reaz, M.T. Islam, S.H.M. Ali, Automatic and reliable leaf disease detection using deep learning techniques, *AgriEngineering* 3 (2) (2021) 294–312.
- [5] J. Eunice, D.E. Popescu, M.K. Chowdary, J. Hemanth, Deep learning-based leaf disease detection in crops using images for agricultural applications, *Agron.* 12 (10) (2022) 2395.
- [6] Y. Zhao, Z. Chen, X. Gao, W. Song, Q. Xiong, J. Hu, Z. Zhang, Plant disease detection using generated leaves based on DoubleGAN, *IEEE/ ACM Trans. Comput. Biology Bioinform.* 19 (3) (2021) 1817–1826.
- [7] P. Vasavi, A. Punitha, T.V.N. Rao, Crop leaf disease detection and classification using machine learning and deep learning algorithms by visual symptoms: A review, *Int. J. Electr. Comput. Eng.* 12 (2) (2022) 2079.
- [8] M.P. Mathew, T.Y. Mahesh, Leaf-based disease detection in bell pepper plant using YOLO v5, *Signal, Image Video Process.* (2022) 1–7.
- [9] L. Goyal, C.M. Sharma, A. Singh, P.K. Singh, Leaf and spike wheat disease detection & classification using an improved deep convolutional architecture, *Inform. Med. Unlocked* 25 (2021) 100642.
- [10] J. Shin, Y.K. Chang, B. Heung, T. Nguyen-Quang, G.W. Price, A. Al-Mallahi, A deep learning approach for RGB image-based powdery mildew disease detection on strawberry leaves, *Comput. Electron. Agric.* 183 (2021) 106042.
- [11] A. Khattak, M.U. Asghar, U. Batool, M.Z. Asghar, H. Ullah, M. Al-Rakhmani, A. Gumaei, Automatic detection of citrus fruit and leaves diseases using deep neural network model, *IEEE Access* 9 (2021) 112942–112954.
- [12] J. Wang, F. Li, S. Lv, L. He, C. Shen, Physically realizable adversarial creating attack against vision-based BEV space 3D object detection, *IEEE Trans. Image Process.* (2025).
- [13] T. Wang, Z. Yu, J. Fang, J. Xie, F. Yang, H. Zhang, L. Zhang, M. Du, L. Li, X. Ning, Multidimensional fusion of frequency and spatial domain information for enhanced camouflaged object detection, *Inf. Fusion* (2024) 102871.
- [14] R. Mahum, H. Munir, Z.-U.-N. Mughal, M. Awais, F. Sher Khan, M. Saqlain, S. Mahamad, I. Tlili, A novel framework for potato leaf disease detection using an efficient deep learning model, *Hum. Ecol. Risk Assessment: An Int. J.* 29 (2) (2023) 303–326.
- [15] J. Wang, F. Li, L. He, A unified framework for adversarial patch attacks against visual 3D object detection in autonomous driving, *IEEE Trans. Circuits Syst. Video Technol.* (2025).
- [16] X. Bai, P. Zhang, X. Yu, J. Zheng, E.R. Hancock, J. Zhou, L. Gu, Learning from human attention for attribute-assisted visual recognition, *IEEE Trans. Pattern Anal. Mach. Intell.* (2024).
- [17] D. Tiwari, M. Ashish, N. Gangwar, A. Sharma, S. Patel, S. Bhardwaj, Potato leaf diseases detection using deep learning, in: 2020 4th International Conference on Intelligent Computing and Control Systems, ICIACS, IEEE, 2020, pp. 461–466.
- [18] M. Kaushik, P. Prakash, R. Ajay, S. Veni, et al., Tomato leaf disease detection using convolutional neural network with data augmentation, in: 2020 5th International Conference on Communication and Electronics Systems, ICCES, IEEE, 2020, pp. 1125–1132.
- [19] C. DeMatteo, J. Jakubowski, K. Stazyk, S. Randall, S. Perrotta, R. Zhang, The headaches of developing a concussion app for youth: Balancing clinical goals and technology, *Int. J. E-Heal. Med. Communications (IJEHMC)* 15 (1) (2024) 1–20.
- [20] M.A. Islam, M.N.R. Shuvo, M. Shamsojjaman, S. Hasan, M.S. Hossain, T. Khatun, An automated convolutional neural network based approach for paddy leaf disease detection, *Int. J. Adv. Comput. Sci. Appl.* 12 (1) (2021).
- [21] T.M. Masenya, Digital transformation of medical libraries: Positioning and pioneering electronic health record systems in South Africa, *Int. J. E-Heal. Med. Communications (IJEHMC)* 15 (1) (2024) 1–13.
- [22] K.P. Panigrahi, H. Das, A.K. Sahoo, S.C. Moharana, Maize leaf disease detection and classification using machine learning algorithms, in: *Progress in Computing, Analytics and Networking: Proceedings of ICCAN 2019*, Springer, 2020, pp. 659–669.

- [23] K. Zhang, Q. Wu, Y. Chen, Detecting soybean leaf disease from synthetic image using multi-feature fusion faster R-CNN, *Comput. Electron. Agric.* 183 (2021) 106064.
- [24] D. Das, M. Singh, S.S. Mohanty, S. Chakravarty, Leaf disease detection using support vector machine, in: 2020 International Conference on Communication and Signal Processing, ICCSP, IEEE, 2020, pp. 1036–1040.
- [25] S. V. Patil, R. Saxena, SR motor TI characteristics performance simulation validation through experimental results, *Cybern. Syst.* (2023) 1–14.
- [26] B.K. Hatuwal, A. Shakya, B. Joshi, Plant leaf disease recognition using random forest, KNN, SVM and cnn., *Polibits* 62 (2020) 13–19.
- [27] C.A. Sari, I.P. Sari, E.H. Rachmawanto, E. Proborini, R.R. Ali, I. Rizqa, et al., Papaya fruit type classification using LBP features extraction and naive bayes classifier, in: 2020 International Seminar on Application for Technology of Information and Communication, ISemantic, IEEE, 2020, pp. 28–33.
- [28] R.J. Hassan, A.M. Abdulazeez, Plant leaf disease detection by using different classification techniques: Comparative, *Asian J. Res. Comput. Sci.* 8 (4) (2021) 1–11.
- [29] M. Agarwal, A. Singh, S. Arjaria, A. Sinha, S. Gupta, ToLeD: Tomato leaf disease detection using convolution neural network, *Procedia Comput. Sci.* 167 (2020) 293–301.
- [30] J. Sun, Y. Yang, X. He, X. Wu, Northern maize leaf blight detection under complex field environment based on deep learning, *IEEE Access* 8 (2020) 33679–33688.
- [31] S. Ning, F. Tan, X. Chen, X. Li, H. Shi, J. Qiu, Lightweight corn leaf detection and counting using improved YOLOv8, *Sensors* 24 (16) (2024) 5279.
- [32] C.-Y. Wang, I.-H. Yeh, H.-Y.M. Liao, Yolov9: Learning what you want to learn using programmable gradient information, 2024, arXiv preprint arXiv:2402.13616.
- [33] H. Yang, X. Deng, H. Shen, Q. Lei, S. Zhang, N. Liu, Disease detection and identification of rice leaf based on improved detection transformer, *Agriculture* 13 (7) (2023) 1361.
- [34] Y. Zhao, W. Lv, S. Xu, J. Wei, G. Wang, Q. Dang, Y. Liu, J. Chen, Detsr beat yolos on real-time object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 16965–16974.
- [35] S. Lauguico, R. Concepcion, R.R. Tobias, A. Bandala, R.R. Vicerra, E. Dadios, Grape leaf multi-disease detection with confidence value using transfer learning integrated to regions with convolutional neural networks, in: 2020 IEEE Region 10 Conference, TENCON, IEEE, 2020, pp. 767–772.
- [36] L.K. Singh, Pooja, H. Garg, M. Khanna, Deep learning system applicability for rapid glaucoma prediction from fundus images across various data sets, *Evol. Syst.* 13 (6) (2022) 807–836.
- [37] M. Khanna, A. Agarwal, L.K. Singh, S. Thawkar, A. Khanna, D. Gupta, Radiologist-level two novel and robust automated computer-aided prediction models for early detection of COVID-19 infection from chest X-ray images, *Arab. J. Sci. Eng.* 48 (8) (2023) 11051–11083.
- [38] H. LT, An energy-aware and fault-tolerant WSN routing approach using serial exponential Newton metaheuristic algorithm for agricultural lands, *Cybern. Syst.* 54 (8) (2023) 1494–1517.
- [39] T. Chen, Z. Tan, T. Gong, Q. Chu, Y. Wu, B. Liu, J. Ye, N. Yu, Mim-istd: Mamba-in-mamba for efficient infrared small target detection, 2024, arXiv preprint arXiv:2403.02148.
- [40] L. Zhu, B. Liao, Q. Zhang, X. Wang, W. Liu, X. Wang, Vision mamba: Efficient visual representation learning with bidirectional state space model, 2024, arXiv preprint arXiv:2401.09417.
- [41] R. Thapa, K. Zhang, N. Snively, S. Belongie, A. Khan, The plant pathology challenge 2020 data set to classify foliar disease of apples, *Appl. Plant Sci.* 8 (9) (2020) e11390.
- [42] S. Fruit Pathology, Plant pathology 2021-FGVC8, 2021, Accessed: [insert date here]<https://www.kaggle.com/competitions/plant-pathology-2021-fgvc8>.
- [43] H. Peng, Z. Li, Z. Zhou, Y. Shao, Weed detection in paddy field using an improved RetinaNet network, *Comput. Electron. Agric.* 199 (2022) 107179.
- [44] M. Tan, R. Pang, Q.V. Le, Efficientdet: Scalable and efficient object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 10781–10790.
- [45] Q. Zhao, T. Sheng, Y. Wang, Z. Tang, Y. Chen, L. Cai, H. Ling, M2det: A single-shot object detector based on multi-level feature pyramid network, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 9259–9266.
- [46] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A.C. Berg, Ssd: Single shot multibox detector, in: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, the Netherlands, October 11–14, 2016, Proceedings, Part I 14, Springer, 2016, pp. 21–37.
- [47] Z. Cai, N. Vasconcelos, Cascade r-cnn: Delving into high quality object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 6154–6162.
- [48] X. Chen, A. Gupta, An implementation of faster rcnn with study for region sampling, 2017, arXiv preprint arXiv:1702.02138.
- [49] Z. Chen, R. Wu, Y. Lin, C. Li, S. Chen, Z. Yuan, S. Chen, X. Zou, Plant disease recognition model based on improved YOLOv5, *Agron.* 12 (2) (2022) 365.
- [50] L. Ma, Q. Yu, H. Yu, J. Zhang, Maize leaf disease identification based on yolov5n algorithm incorporating attention mechanism, *Agron.* 13 (2) (2023) 521.
- [51] X. Wang, J. Liu, Vegetable disease detection using an improved YOLOv8 algorithm in the greenhouse plant environment, *Sci. Rep.* 14 (1) (2024) 4261.