

An Ensemble Learning Framework with Explainable AI for interpretable leaf disease detection

Mohammad Rifat Ahmmad Rashid^{ID *}, Md. AL Ehtesum Korim, Mahamudul Hasan, Md Sawkat Ali, Mohammad Manzurul Islam, Taskeed Jabid, Raihan Ul Islam, Maheen Islam

Department of Computer Science And Engineering, East West University, A/2 Jahurul Islam Ave, Dhaka, 1212, Bangladesh

ARTICLE INFO

Keywords:

Leaf diseases
Deep learning
Transfer learning
Convolutional neural networks (CNNs)
Cucumber disease detection
Machine learning

ABSTRACT

The early and accurate detection of plant diseases is critical for sustainable agriculture, ensuring crop health, reducing losses, and supporting food security. To address this challenge, we present an Ensemble Learning Framework with Explainable AI (XAI) tailored to disease detection, using cucumber leaf diagnosis as a key use case. In this study, we experimented with a dataset comprising 6,400 images capturing six prevalent cucumber leaf diseases – Gummy Stem Blight, Downy Mildew, Anthracnose, Bacterial Wilt, Belly Rot, and Pythium Fruit Rot – alongside two healthy categories. Prior to training, the images underwent preprocessing steps such as resizing, rescaling, and data augmentation (through random rotations, flips, zooms, and contrast adjustments) to enhance model generalization. The proposed framework unites multiple architectures – CNN, DenseNet121, EfficientNetB0, InceptionV3, MobileNetV2, ResNet50, and Xception – into an ensemble that attained overall accuracy of 99%, alongside high recall and F1-scores. Individual models demonstrated accuracy ranging from 88.71% to 99%, underscoring the robustness of the ensemble. Integrating XAI methods further ensures interpretable outputs, granting valuable insights into the decision-making process and heightening transparency for researchers and agronomists. The findings confirm that transfer learning, model ensembling, and interpretability methods significantly enhance classification performance, especially in cases of limited data, offering a scalable solution for improved disease management in agriculture. Additionally, the framework is scalable for real-world deployment by enabling real-time disease monitoring on edge devices (e.g., Raspberry Pi, IoT systems), seamless integration with smart farming platforms, and continuous learning for adaptive crop management.

1. Introduction

Agriculture is a critical sector of the economy, and effective farming practices are essential for maintaining crop quality. Protecting plants from leaf diseases is vital for economic viability and aligns with the goals of responsible farming practices. Early and accurate detection of plant diseases is fundamental for efficient crop management. The growth of cucumber crops is often hindered by a variety of diseases, which can spread quickly. Diseases such as downy mildew and powdery mildew primarily attack the leaves, causing serious damage that leads to significant losses in agricultural output. Other diseases including Pythium fruit rot, anthracnose, bacterial wilt, gummy stem blight, and belly rot also pose risks to crop health and productivity. Controlling these diseases is crucial for sustaining productive cucumber farming [1].

A range of technologies, including visual inspection, digital imaging, remote sensing, machine learning, DNA-based diagnostics, immunological assays, spectroscopy, smartphone apps, expert systems, and sensor networks, are employed to diagnose and manage diseases promptly and accurately. These methods play a critical role in safeguarding crop yields and ensuring food security [1]. This study addresses the challenge of rapid and precise detection of foliar diseases in crops. The main goal is to develop an automated system for identifying leaf diseases caused by various pathogens using an ensembling framework of machine learning and image processing techniques. We propose a method that could diminish the need for labor-intensive manual inspection and enhance the health and yield of crops. This goal is pursued by harnessing diverse algorithms in the domain of image processing [2].

* Corresponding author.

E-mail address: rifat.rashid@ewubd.edu (M.R.A. Rashid).

The field of computer vision has significantly advanced the detection and classification of plant diseases. Historically, disease identification has relied on traditional image processing techniques, which require meticulous segmentation and manual feature engineering. These methods, while effective, are labor-intensive and rely heavily on hand-crafted features. The advent of artificial intelligence (AI) has introduced machine learning and deep learning models, which offer improvements in accuracy for disease detection tasks. Despite the ongoing use of traditional methods, a discernible shift towards AI-driven approaches is evident in the realm of plant disease identification [3].

The fields of image processing and computer vision are experiencing a surge in growth, with numerous studies emerging annually. This paper advocates for an ensembling approach to the identification and classification of plant diseases, with a particular focus on Convolutional Neural Networks (CNNs) [4]. CNNs represent a class of deep learning models that have excelled in various image analysis tasks, including classification, object recognition, and segmentation [5]. They are especially appropriate for detecting plant diseases, given their proficiency in extracting distinctive features from images of leaves. A prominent CNN architecture in the domain of plant disease detection is ResNet. It is designed to capture deep image representations by employing layers of residual blocks, which enable the modeling of more complex attributes without sacrificing computational efficiency. The appeal of deep learning in plant pathology is linked to its capability to process extensive datasets and discern intricate image patterns, paving the way for the creation of precise and effective disease detection models that can substantially benefit crop management and yield [5].

This study introduces an ensembling framework for leaf disease detection, using cucumber leaf disease as a use case. Employing transfer learning (TL) [6], features are extracted from several pre-trained CNN models: ResNet-50 [7], EfficientNetB0 [8], DenseNet-121 [9], Xception [10], MobileNet V2 [11], and Inception V3 [12]. TL is beneficial for scenarios where a small dataset is available for the target task but a larger dataset exists for a related task. It is commonly applied to adjust deep learning models that have been pre-trained on extensive image datasets. The main contributions of this work are as follows:

- A comprehensive ensemble model integrating ResNet50, EfficientNetB0, DenseNet121, Xception, MobileNetV2, and InceptionV3 is proposed. This approach combines the strengths of individual architectures, achieving superior performance with a test accuracy of 99%.
- To overcome limitations of small datasets and ensure generalization, a rigorous data augmentation pipeline was implemented. Techniques such as rotation, scaling, and contrast normalization enriched the training dataset, increasing its variability and robustness.
- The study incorporates Grad-CAM, Grad-CAM++, and Eigen-CAM to provide interpretable visualizations of the model's predictions. These tools highlight critical regions in images used for disease classification, enhancing transparency and user trust.
- The ensemble approach is scalable to other crop diseases and adaptable to varying environmental conditions, making it a promising tool for precision agriculture and sustainable farming practices.

The paper is structured as follows: Section 2 provides a thorough literature review, discussing prior advancements in plant disease detection and highlighting gaps our study addresses. Section 3 details the dataset, outlining preprocessing steps and augmentation techniques used to improve model robustness. Section 4 introduces our methodology, explaining the ensembling framework, model architecture choices, and training strategies. Section 5 presents experimental results, evaluating individual and ensemble model performance and demonstrating interpretability through Grad-CAM visualizations. Section 6 offers an in-depth discussion of the model's accuracy, interpretability, cost analysis, and limitations, providing insights for future work. Section 7 concludes with reflections on the framework's contributions to agricultural disease detection.

2. Related work

There has been a substantial amount of research focused on leveraging machine learning and deep learning techniques to accurately diagnose plant illnesses. In this work, we explored a variety of computational models and feature extraction methods aimed at enhancing the precision and reliability of plant disease detection. This section provides an overview of recent advancements in this field, presenting a comparative analysis of different approaches and their efficacy in recognizing and classifying various plant diseases.

2.1. Deep learning architectures

In the study [13], a range of machine learning classifiers was assessed for their effectiveness in plant disease detection, including Support Vector Machine (SVM), Artificial Neural Network (ANN), K-Nearest Neighbor (KNN), Fuzzy C-Means, and Convolutional Neural Network (CNN). The study reports that CNNs outshine their neural network counterparts in accuracy and disease detection breadth.

A thorough review of paper [1] plant disease detection and diagnosis approaches are given in this study, along with feature extraction methods like color histogram, SIFT, Gabor filter, and CCM and classification classifiers like ANN, SVM, BPNN, RBFNN, and PNN. Additionally, it covers lesion segmentation algorithms and supervised classification techniques for the identification and diagnosis of alfalfa leaf diseases utilizing pattern recognition and image-processing algorithms. The research emphasizes the shortcomings of conventional techniques in obtaining the best characteristics from complex and irregular photos of sick leaves, resulting in poorer identification rates.

The techniques that are currently in use in the literature [14] concentrate on classifying agricultural diseases but do not offer any preventative actions. In this work, a method is suggested that emphasizes preventative actions in addition to crop disease detection and treatment. A strategy that is often used in the literature for crop disease identification is the use of image processing techniques and artificial neural networks in this research. The work in [15] proposed detecting plant leaf disease is a complex task that requires a combination of feature extraction algorithms, deep learning models like CNN, and an IoT-based strategy. These methods focus on discriminating feature selection of disease components using feature fusion, clustering, and PHOG feature extraction. Conventional methods have outperformed CNN models in terms of accuracy when using photos from datasets.

To automate the diagnosis of viral plant diseases, the research [16] introduces a unique plant disease detection system based on convolutional neural networks (CNN). The method achieves great classification performance by using training photos to automatically gather features. The goal of the study is to identify the melon yellow spot virus (MYSV) and zucchini yellow mosaic virus (ZYMV), two severe viral diseases that affect cucumber plants. 800 photos of cucumber leaves – 300 with MYSV, 200 with ZYMV, and 300 without the disease – make up the training dataset. The suggested CNN-based approach classifies cucumbers into two illness classes and a non-disease class, with an average accuracy of 94.9%. The study's restricted dataset and the necessity for future system development to manage a wider range of illnesses in varied photographic settings are acknowledged by the authors.

Recent literature highlights the efficacy of hybrid optimization models [17] that merge advanced evolutionary computation methods with AI-driven optimization frameworks to address complex real-world challenges [18]. Techniques such as genetic algorithms, particle swarm optimization, and differential evolution have been effectively combined to enhance the convergence speed and solution quality in diverse applications [19]. In the healthcare domain, machine learning-based prediction models are increasingly employed to forecast patient outcomes and disease trajectories, leveraging extensive clinical data to improve diagnostic accuracy [20]. Furthermore, ensemble learning approaches, which integrate multiple predictive models, have shown

substantial promise in healthcare by enhancing robustness, accuracy, and interpretability [21]. This convergence of hybrid optimization and ensemble methodologies not only propels the advancement of predictive analytics but also paves the way for more adaptive, resilient, and reliable decision-support systems in real-world healthcare applications [22].

The paper's [23] main concern is how to deal with crop output losses brought on by pests and diseases, which in India account for about half of all crop losses. To categorize illnesses in cucumber plants and determine the severity of disease spots on cucumber leaves, it suggests an image processing technique. The ratio of the infected region to the overall leaf area is used to gauge the severity of the illness. The research uses machine learning methods to create a model that can correctly identify and categorize illness in plant photos, including support vector machines (SVM), K-nearest neighbor classifiers (K-NN), and neural networks (NN). The workflow for the system, which focuses on early identification of plant illnesses, involves image processing, feature extraction, and model training. Another good example is [24].

The research [25] examines the potential of deep learning (DL) models for more accurate detection and classification of plant diseases. To recognize the signs of plant illnesses, a variety of DL architectures and visualization methods are utilized. Performance measures are then used to assess these models. DL models are frequently evaluated using the PlantVillage dataset, even though it has a basic backdrop and cannot accurately reflect real-world situations. The system described in the research [4] uses deep learning methods to categorize and identify plant leaf diseases. The system classifies 15 types of plant leaf illnesses, including bacteria, fungus, etc., and 3 classifications for healthy leaves using a convolutional neural network (CNN). The dataset known as Plant Village comprises 20,636 photos of plants and the illnesses that affect them. These photographs were utilized in the system.

The research [2] focuses on the detection and classification of tomato plant illnesses using image processing and machine learning methods, which is a critical component of Indian agriculture. Every Year there are plenty of losses affected by disease. The disease detection method goes through four stages: preprocessing, leaf segmentation, feature extraction, and classification. For classification and regression, the k-nearest neighbors (KNN) method is used, which offers answers for both classification and treatment suggestions.

The article [26] suggests using a deep convolutional neural network (DCNN) to identify cucumber illnesses based on symptom-wise categorization. The four cucumber diseases – anthracnose, downy mildew, powdery mildew, and target leaf spots – are represented as symptom pictures in the image dataset used to train the DCNN. To increase the dataset and to minimize the chance of overfitting data augmentation techniques are used. With an accuracy of 93.4%, the DCNN produces positive recognition results. The findings of comparative studies utilizing traditional classifiers (Random Forest and Support Vector Machines) and AlexNet demonstrate that the DCNN is a reliable technique for identifying cucumber illnesses in field situations. Using the pre-trained network AlexNet, transfer learning is also carried out, and its effectiveness is measured.

The study [27] focuses on the automated identification and diagnosis of plant diseases in the context of agricultural information. Deep learning is believed to be the most successful strategy, particularly transfer learning using trained models. The authors selected the pre-trained VGGNet on ImageNet and the Inception module for transfer learning, and they trained the freshly built neural networks using their own datasets. The recommended technique was able to verify photos of rice plants with at least 91.83% accuracy on the public dataset and forecast rice plant images with an average accuracy of 92.00% despite the difficult background.

In the paper [28] study, a CNN algorithm is presented for differentiating between diseased and healthy cucumber leaves. The accuracy of the method was 98.19% with an enriched dataset and 100% with the publicly accessible plant disease dataset. With data augmentation

techniques being utilized to avoid overfitting, the algorithm comprises picture improvement, feature extraction, and classification. To address the lack of a public dataset, a new data set of illnesses affecting cucumber leaves was created. The study [6] suggests a new deep learning architecture with an optimization approach for identifying cucumber and potato leaf diseases. The suggested architecture consists of five steps: data augmentation, fine-tuning a pre-trained deep model, extracting deep features, refining the features using an enhanced cuckoo search method, and classifying the selected features using machine learning classifiers like SVM.

The main objective of the article [5] is to employ deep learning methods to identify illnesses in cotton leaves. Since early discovery is essential for efficient treatment, cotton is vulnerable to several illnesses that can seriously harm crops. For precise illness detection, the authors suggest a meta-deep learning model that performs 98.53% better than the Cotton Dataset. Approximately, 2385 photos of healthy and sick cotton leaves made up the dataset used to train the model. In the study [3], convolutional neural networks (CNNs), a type of deep learning methodology, are discussed as a method for utilizing photographs to detect plant illnesses in crops. The authors evaluated the model on the PlantVillage dataset after fine-tuning the hyperparameters of pre-trained models including DenseNet-121, ResNet-50, VGG-16, and Inception V4.

In the article [29], convolutional neural networks (CNN) are used to identify plant leaf diseases. It makes use of image processing and a freely accessible dataset called Plant Village, which has 12 disease-related classifications and 3 healthy classes. Overfitting is avoided, and classification outcomes are enhanced using data augmentation, dropout, and weight regularization. The study [30] investigates the application of convolutional neural network algorithms (CNN) to identify downy and powdery mildew infections in Vietnamese cucumbers. Using a dataset of 7640 photos gathered over the course of three months, researchers trained a CNN model that had an accuracy rate of over 80%. The device can instantly identify illnesses and notify farmers. The research [31] focuses on the use of deep learning methods, particularly deep convolutional neural networks (CNN), in the detection of plant diseases. Plant disease is very harmful and decreases farmer's profits. In the presentation of many research projects using various deep learning models for plant disease diagnosis, the authors present a comparative analysis of various deep learning models applied to the detection of plant diseases. The CaffeNet architecture, a customized version of AlexNet, is used in the study to highlight the application of transfer learning for plant disease identification.

The authors in [32] addressed the challenge of improving edge detection in digital images, particularly noting the inefficiencies in traditional methods like the Baljit and Amar technique, which are computationally expensive and time-intensive. Their approach leverages Shannon entropy and divides images into four regions, each with an optimal threshold, rather than a global threshold, which reduces CPU runtime and enhances edge detection quality. Experimental results demonstrated superior performance over the Baljit and Amar method in gray-scale images, though the study focused solely on grayscale. In [33], the authors tackled the problem of filtering inappropriate images, such as pornographic content, embedded in PDFs. Their proposed system extracts images from PDFs, converts them to JPEG format, and detects skin regions using the Log Opponent and HSV color spaces. To increase accuracy, these techniques were combined with the YIQ skin detection method, addressing complex background detection challenges. Results indicated that the modified Log Opponent technique outperformed the HSV method, though challenges remained with complex backgrounds and lighting conditions.

In [34], the authors identified a lack of effective methods for blocking inappropriate images on the web. They proposed a Browser Helper Object (BHO) that extracts images and URLs from web pages, comparing RGB-based skin detection with a YUV-YIQ approach enhanced by a saturation parameter. The YUV-YIQ method outperformed RGB

Table 1

Previous studies for plant leaf disease feature extraction and proposed model.

Paper	Feature extraction technique
[1]	–
[14]	Color, shape, texture
[15]	HOG, LBP, and COLOR
[16]	CNN
[23]	Shape, texture, and color
[4]	–
[2]	GLCM, HOG
[26]	Texture and color
[27]	–
[28]	–
[6]	Contrast Enhancement
[5]	–
[3]	Deep CNN
[29]	–
Proposed Model	–

Table 3

Previous studies for plant leaf disease dataset size and proposed model.

Paper	Dataset size
[1]	Primary 700, After Augmentation 35000
[14]	–
[15]	Images 1262
[16]	800,500 images with disease and 300 healthy
[23]	–
[4]	20636 images, disease 13 class and healthy 3 class
[2]	200 images
[26]	14,208 images
[27]	1000 leaf images
[28]	4868 cucumber leaf images
[6]	3 datasets used 15,938, 4863, and 2152 images
[5]	2385 images
[3]	54,305 images with 38 classes
[29]	34,934 images, 12 disease, 3 healthy
Proposed Model	6400 images with 8 classes

Table 2

Previous studies for plant leaf disease classification and proposed model.

Paper	Model architecture
[1]	GPDCNN, PNNs, SRC, AlexNet
[14]	SVM, ANN
[15]	KNN, SVM
[16]	CNN
[23]	SVM, LDA, NN
[4]	CNN
[2]	ANN, KNN
[26]	Random Forest, AlexNet, SVM
[27]	VGGNet3
[28]	AlexNet, Inception-V3, and ResNet-50
[6]	DarkNet19
[5]	CNN, VGG16, ResNet-50, Meta learning
[3]	DenseNet-121, ResNet-50, VGG-16, Inception V4
[29]	CNN
Proposed Model	Ensembling Framework

in skin detection but faced limitations with similar color backgrounds. Meanwhile, [35] presented an effective face recognition system incorporating preprocessing, feature extraction with Gabor filters, feature reduction via PCA, and feature selection using Grey Wolf Optimization (GWO). The classification was performed using k-nearest neighbors (k-NN), achieving a 97% recognition rate on the Yale face database. Although successful, this study was limited to facial features. In [36], a method was proposed that combines skin detection with the HSV color system, Haar cascade files, and eye and nose detection to improve face detection accuracy. Tested on challenging images, it achieved a 96% detection rate, surpassing the Viola-Jones and Zahra systems, though occasional false positives or missed faces occurred. Lastly, [37] proposed a face detection and recognition system incorporating Haar cascades and the Local Binary Pattern Histogram (LBPH) algorithm, yielding a 96.5% recognition rate with a 3.5% error rate, while [38] utilized deep learning-based mask detection and facial landmarks to address COVID-19 related challenges in face recognition with masks, achieving up to 97.8% accuracy. This was further extended in [39] to integrate RPCA with KNN, showing improvement in classification rates. Additionally, [40] explored a deep-learning model for poultry audio signal classification using a custom Burn Layer in ResNet, achieving a 98.55% accuracy with limitations in noisy data sensitivity.

2.2. Summary

Recent research on plant disease detection has centered on improving image processing techniques, refining feature extraction methods, and designing robust classification algorithms. A wide range of computational approaches have been employed, including Support Vector Machines (SVMs), Artificial Neural Networks (ANNs), K-Nearest Neighbors (KNNs), Fuzzy C-Means (FCM), and notably, Convolutional Neural Networks (CNNs). CNNs, in particular, have been successful in

automatically extracting distinguishing features from images, making them especially effective in recognizing plant diseases.

Despite their advantages, CNNs often require extensive and varied datasets to generalize well. In plant pathology, this requirement becomes even more challenging due to the limited volume and diversity of disease-specific data. To mitigate these constraints, our research has identified several feature extraction techniques (summarized in Table 1) designed to handle smaller datasets. These methods focus on color, shape, and texture analysis, as well as more sophisticated descriptors like Histogram of Oriented Gradients (HOG), Local Binary Patterns (LBP), and Gray Level Co-occurrence Matrix (GLCM), alongside contrast enhancement approaches. Such strategies enable effective characterization of plant diseases, even when only modest training data is available. Table 3 presents previous studies on plant leaf disease dataset sizes and the proposed model.

Furthermore, Table 2 reviews commonly used classification models – ranging from SVMs and KNNs to ANNs and CNNs – that have been applied in plant disease detection. Our proposed methodology sets itself apart by adopting an ensemble strategy, leveraging the strengths of multiple CNN architectures, namely Xception, MobileNetV2, ResNet50, EfficientNetB0, InceptionV3, and DenseNet121. This ensemble approach aims to enhance the overall accuracy and robustness of plant disease recognition .

3. Dataset description

To experimentally evaluate our proposed framework, we utilized a cucumber dataset [41] composed of both original and augmented images. This section provides an overview of the cucumber leaf diseases addressed in this study, followed by a detailed description of the dataset used and the preprocessing steps applied.

3.1. Cucumber diseases

Cucumber plants are vulnerable to a range of diseases that can significantly reduce both yield and quality. Below is a summary of the most common cucumber leaf diseases, highlighting their symptoms and potential impact on plant health:

- **Anthracnose** [42] is primarily caused by the fungus *Colletotrichum orbiculare*. It often appears as small yellow, circular spots on leaves, progressing to more severe leaf spots, fruit rot, and stem cankers under warm, humid conditions. The disease can result in substantial foliage loss, reduced yield, and poorer fruit quality if not managed promptly.
- **Bacterial Wilt** [43], triggered by the bacterium *Erwinia tracheiphila*, is a serious threat to cucumber plants. Early signs include the wilting of individual leaves, particularly those damaged by cucumber beetles. If unchecked, wilting spreads throughout the vine and ultimately kills the entire plant.

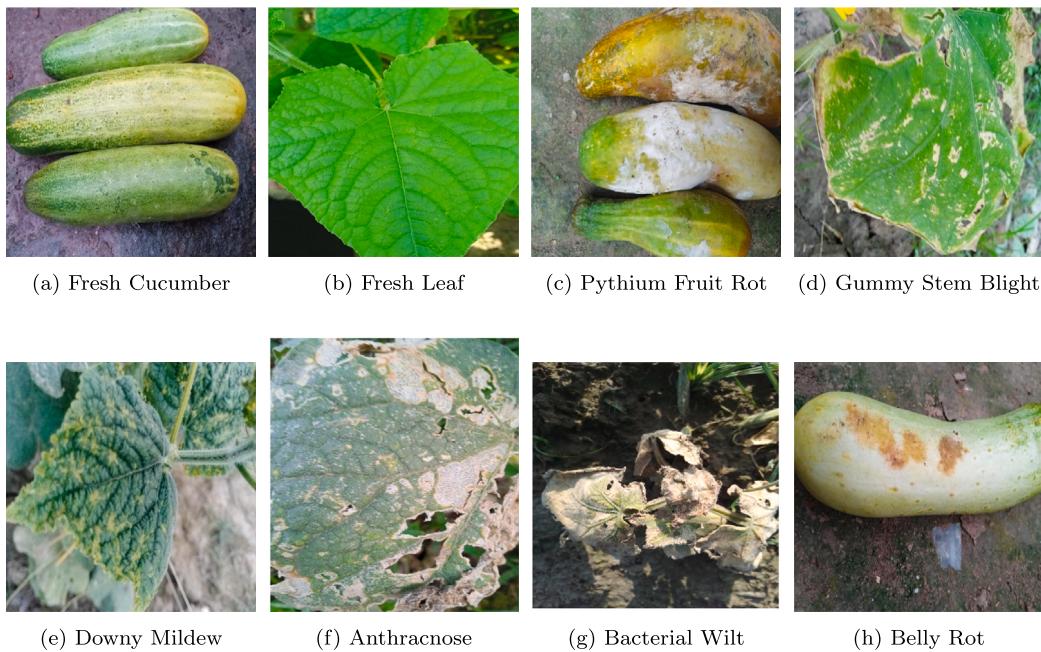


Fig. 1. Sample images from each class in the dataset [41]: (a) Fresh Cucumber, (b) Fresh Leaf, (c) Pythium Fruit Rot, (d) Gummy Stem Blight, (e) Downy Mildew, (f) Anthracnose, (g) Bacterial Wilt, and (h) Belly Rot.

- **Belly Rot** [43] is linked to the fungal pathogen *Rhizoctonia solani* and tends to flourish in conditions of high temperature and humidity. Infected fruits develop tan-to-brown lesions at the blossom end or underside, forming crater-like depressions. Although the lesions expand and dry out, affected fruits typically remain firm, showing infrequent soft rot.
- **Pythium Fruit Rot** [43] is caused by *Pythium* species. Symptoms begin as water-soaked, brownish spots that rapidly enlarge into watery, soft rotting areas, often on the fruit portions in direct contact with the soil. In humid conditions, a white, cottony mycelium may appear on the decaying tissue.
- **Gummy Stem Blight** [44] poses a significant fungal threat, initially forming brown or grayish-brown lesions with a sticky or gummy appearance on leaves. These lesions can merge and spread, causing foliage to wilt and drop. In severe cases, the disease results in stem rotting, potentially destroying the plant.
- **Downy Mildew** [45] is particularly prevalent in moist environments. Early indications include yellow or pale green blotches on the upper leaf surfaces, while a white or grayish-purple downy growth appears underneath. Progression of the disease leads to brown, necrotic spots and eventual leaf curling, distortion, and drop.

3.2. Dataset overview

To evaluate our proposed framework, we conducted experiments on a cucumber dataset [41] comprising both original and augmented images. Initially, the dataset contained 1280 images – 160 images per class – across eight categories, including two healthy classes and six disease classes. Our primary objective was to distinguish healthy cucumber samples (Fresh Cucumber and Fresh Leaf) from the six disease categories (Pythium Fruit Rot, Gummy Stem Blight, Downy Mildew, Anthracnose, Bacterial Wilt, and Belly Rot). Sample images from each category are presented in Fig. 1.

3.2.1. Data augmentation

Class imbalance and insufficient data diversity can hinder model robustness and generalization. To address these issues, each class was augmented to contain 800 images, resulting in a total of 6400 images.

Table 4

Dataset summary and augmentation techniques.

Dataset summary	Value
Original Images per Class	160
Total Original Images	1280
Augmented Images per Class	800
Total Augmented Images	6400
Data augmentation techniques	
Flip	Horizontal, Vertical
Shear	0.2
Zoom	Range: 0.8 to 1.2
Rotation	Angles: 45°, 60°, and 90°
Brightness Adjustment	Histogram equalization
Scaling	Bicubic, Bilinear, Nearest Neighbor

Table 5

Original and augmented image sizes for each class.

Class name	Original images	Augmented images
Fresh Cucumber	160	800
Fresh Leaf	160	800
Pythium Fruit Rot	160	800
Gummy Stem Blight	160	800
Downy Mildew	160	800
Anthracnose	160	800
Bacterial Wilt	160	800
Belly Rot	160	800
Total	1280	6400

Various augmentation methods – flipping, shearing, zooming, rotation, brightness adjustment, and scaling – were applied. Table 4 summarizes the data augmentation techniques used. After augmentation, each of the eight categories contained 800 images, as shown in Table 5.

3.3. Data preprocessing

Data preprocessing is a crucial step in preparing an effective machine learning model, especially for image classification tasks. The following steps were applied to preprocess the dataset of 6400 cucumber images across eight labels (two healthy and six disease categories):

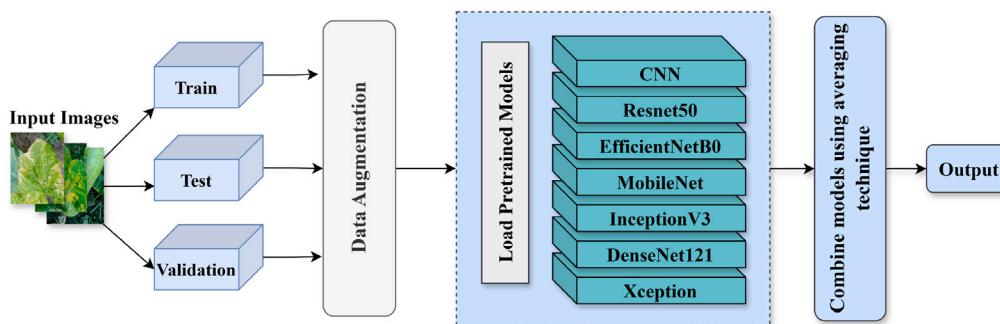


Fig. 2. Proposed methodology.

Table 6

Dataset Distribution for training and testing with an 80-20 split.

Training size (80%)	Testing Size (20%)
5120 images	1280 images

- Resize images to a consistent dimension of 224×224 pixels to ensure uniform input size for the models.
- Normalize pixel values to have a mean of 0.5 and a standard deviation of 0.5 to standardize the pixel intensity across the dataset.
- Apply data augmentation techniques, including random cropping and resizing, to increase diversity and improve model generalization during training.

The dataset was initially split into training and test subsets, adhering to an 80–20 ratio. Shuffling and a fixed random seed were employed to ensure reproducibility and randomness in the data split. Further segmentation was applied, dividing the training data into separate training and validation sets, as detailed in [Table 6](#). The `FlowFromDataFrame` method¹ was used to generate batches of images for training and validation. Configurations such as image target size, color mode, and batch size (set to 32) were adjusted to meet the requirements of our model.

4. Methodology

In our study, we propose an ensembling framework that leverages a variety of notable deep learning architectures such as ResNet50, InceptionV3, EfficientNetB0, DenseNet121, Xception, and MobileNetV2. The structure of our proposed methodology is illustrated in [Fig. 2](#). This particular combination was selected based on extensive evaluation for their synergistic balance of accuracy and computational efficiency, alongside their demand on computational resources. Initially, raw cucumber images undergo a series of preprocessing stages, involving normalization and data augmentation, to enhance model training efficacy. Techniques such as resizing, normalization, and data augmentation through random rotations, flips, zooms, and contrast adjustments are applied to increase data diversity and model reliability. The dataset is thoughtfully partitioned to facilitate effective training, validation, and testing processes. [Fig. 3](#) presents an overview of the proposed ensemble framework for the detection of leaf disease.

We employ a robust training strategy involving various CNN architectures like ResNet50, EfficientNetB0, MobileNetV2, InceptionV3, DenseNet121, and Xception. Each of these models is trained individually using strategies such as checkpoints and early stopping to optimize performance. These individual models are then combined

to form an ensemble model. Our ensemble approach aims to improve prediction performance by combining the strengths of multiple deep learning models. This method reduces the risk of overfitting and enhances generalization to a variety of visual patterns by merging predictions from different model ensembles. Ensembles provide more accurate predictions by minimizing errors and improving reliability and performance.

4.1. Deep learning models

Deep learning, a subfield of machine learning, trains artificial neural networks to recognize and encode complex patterns in data. Deep learning is well-suited for tasks involving large and complex datasets. Artificial neural networks (ANNs) are made up of neurons that process and transmit information. ANNs can capture complex data interactions, which has enabled significant progress in a wide range of fields. In this section, we provide an overview of each Deep Learning algorithm used in this work.

4.2. Convolution Neural Network (CNN)

A deep learning architecture in [Fig. 4](#) created specifically for visual data problems is the convolutional neural network (CNN) [46]. It makes use of pooling layers for spatial sampling reduction, convolutional layers for automatically detecting features like edges and textures, and fully linked layers for classification. Rotating convolutional and pooling layers, followed by fully connected layers, could make up a customized CNN architecture. Architecture is Conv-Pool-Conv, -Pool and Fully Connected are the steps. Due to their capacity to infer hierarchical characteristics from input, these architectures are effective for image recognition applications.

4.3. Transfer learning

Transfer learning in Convolutional Neural Networks (CNNs) involves adapting pre-trained models, originally developed for one task, to new tasks. It involves retaining the convolutional base, enriched with patterns and features learned from extensive datasets while customizing the top layers to specialize in the specific tasks. In this approach, the pre-trained models, usually enhanced with knowledge from expansive datasets like ImageNet, bring a wealth of foundational image representation skills to the new tasks, allowing for effective learning even with smaller labeled datasets in the new domain.

Our study strategically leverages this approach, applying a suite of models: ResNet 50 [7], EfficientNetB0 [8], DenseNet-121 [9], Xception [10], MobileNet V2 [11], Inception V3 [12], and CNN [46]—that have been pre-trained on ImageNet. The use of transfer learning aims to capitalize on the rich feature representations learned by these models, ensuring swift and efficient training processes and yielding highly competent models, as visualized in [Fig. 5](#). This approach stands as a pivotal asset in our methodology, catalyzing enhanced performance and adaptability in our cucumber disease detection models.

¹ https://www.tensorflow.org/api_docs

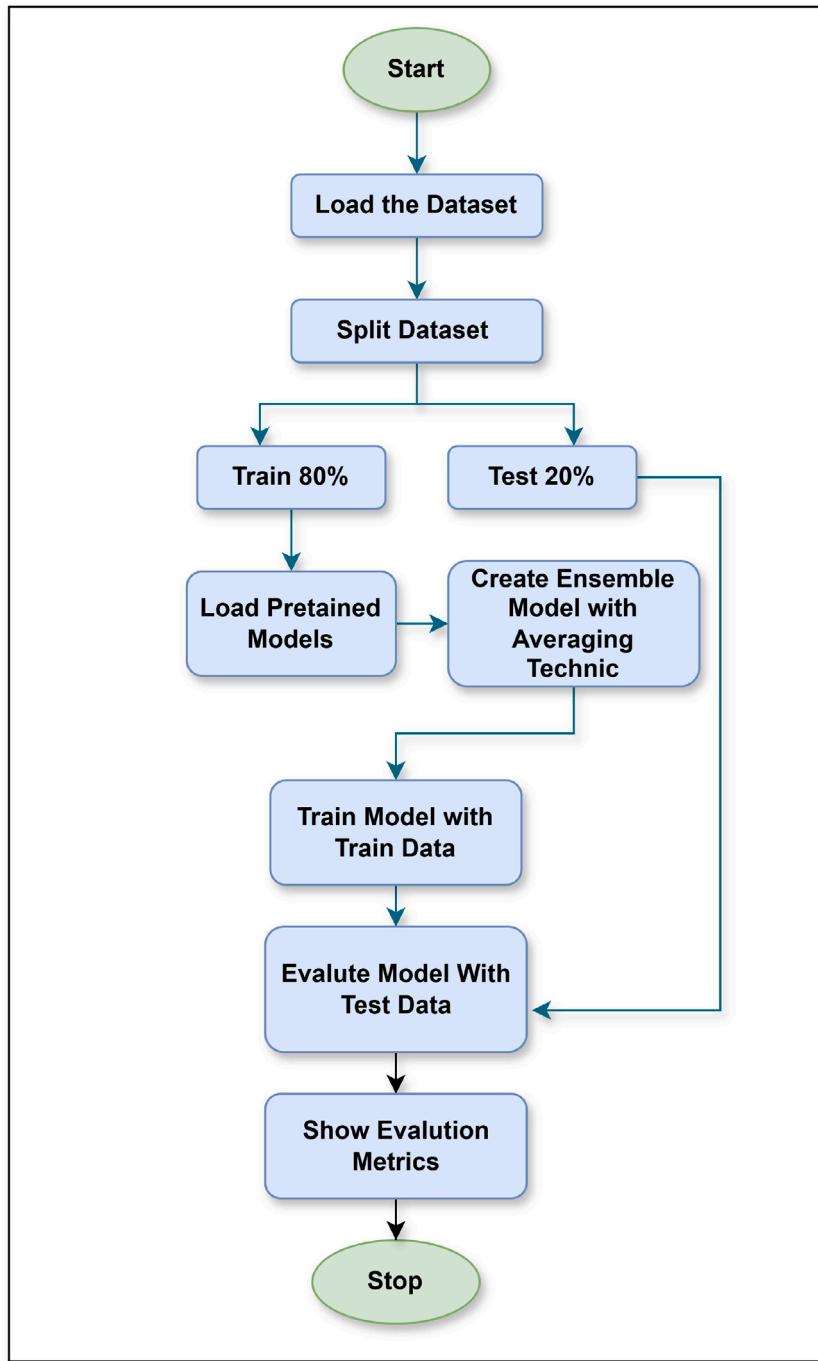


Fig. 3. Flowchart of the proposed ensemble framework.

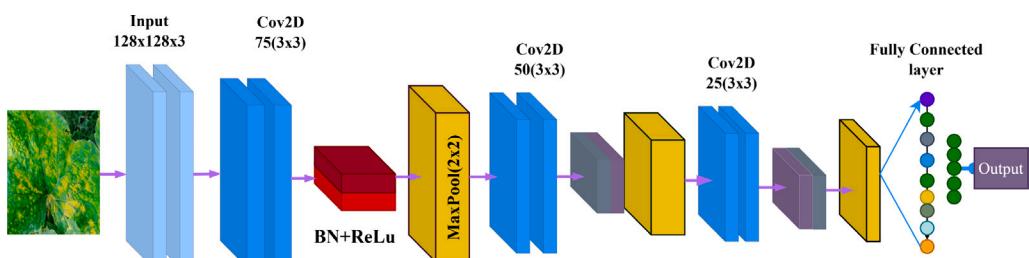


Fig. 4. CNN architecture.

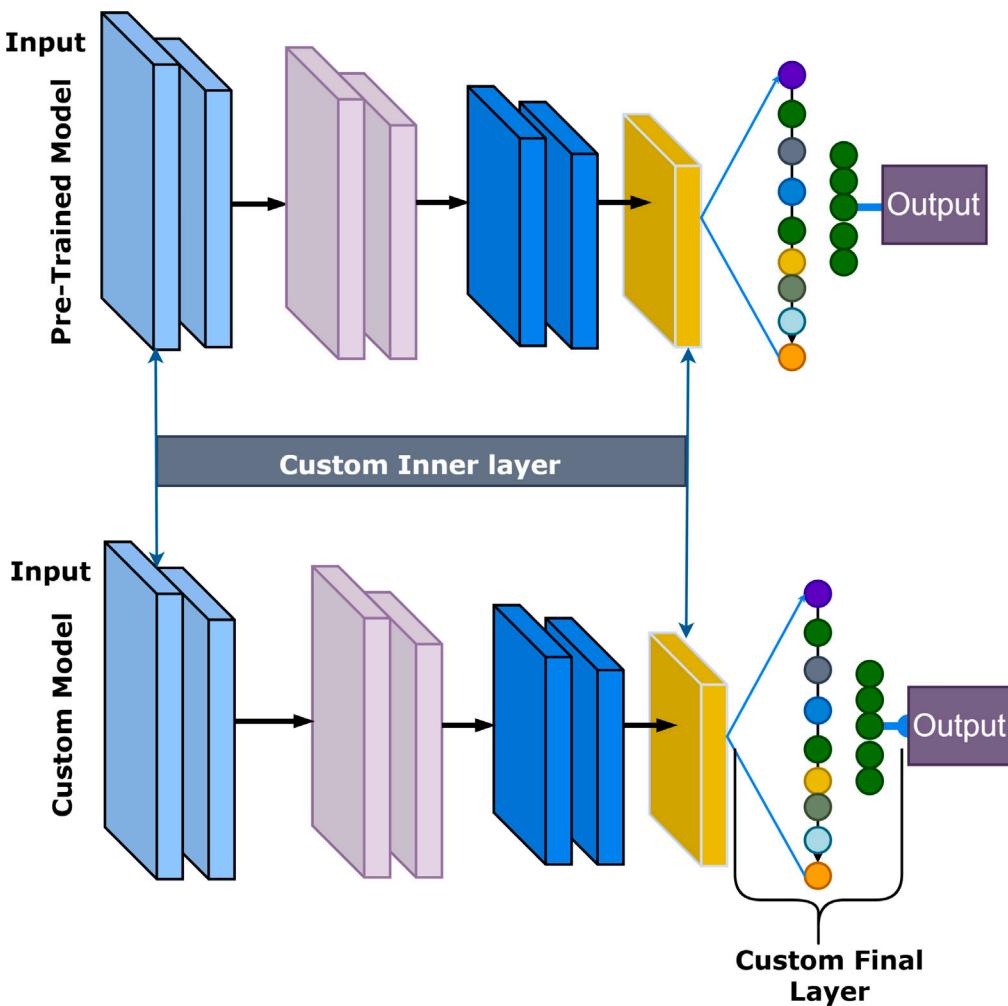


Fig. 5. Transfer learning architecture.

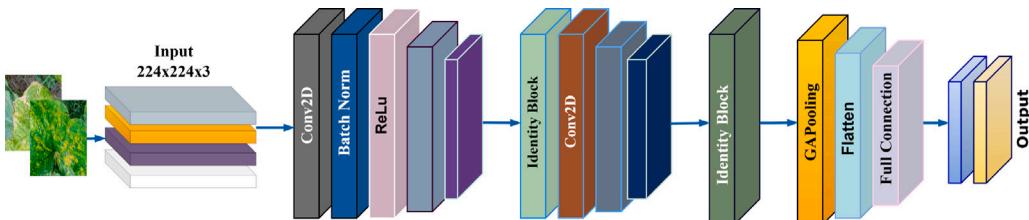


Fig. 6. ResNet-50 architecture.

4.3.1. ResNet-50

Fig. 6 showcases ResNet-50, a variant of Residual Networks (ResNet), renowned for introducing residual connections to mitigate the vanishing gradient problem. This innovative modification facilitates the training of profoundly deep networks. ResNet-50, with its assembly of 50 layers, is particularly noted for its remarkable depth and performance, making it a popular choice for tasks involving image classification and a multitude of computer vision applications.

4.3.2. InceptionV3

Fig. 7 presents InceptionV3 is a deep convolutional neural network (CNN) architecture that was developed by Google AI. It is a member of the Inception model family, which is known for its efficiency and accuracy in image classification and object detection tasks. InceptionV3 uses a series of inception modules to extract features from images at multiple scales. This allows the model to learn complex patterns that

may not be visible at a single scale. Inception modules are composed of multiple convolutional layers with different filter sizes. This allows the model to extract features of different sizes and spatial relationships. Inception modules also use a technique called residual connections, which allows the model to learn deeper representations of the input image.

4.3.3. EfficientNet

Fig. 8 illustrates the EfficientNetB0 model, a part of the EfficientNet family that is renowned for blending computational efficiency with exceptional performance. EfficientNetB0 adeptly balances network depth, width, and resolution through a compound scaling approach. This design characteristic enhances its applicability, particularly in scenarios where computational resources are constrained, allowing for a versatile deployment in various applications without compromising on performance.

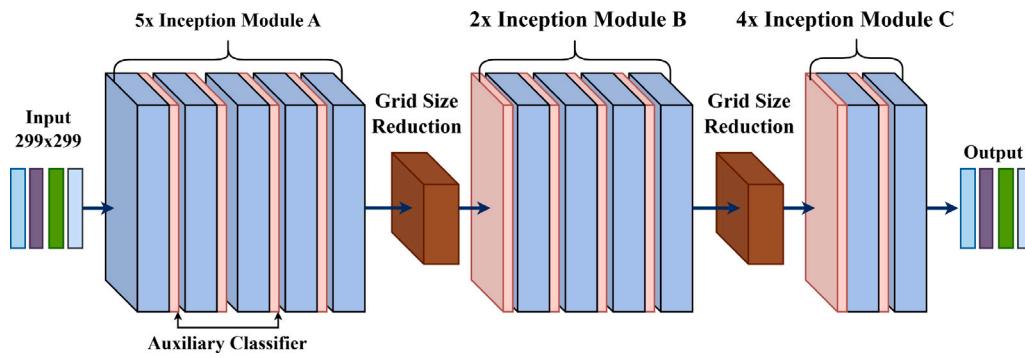


Fig. 7. InceptionV3 architecture.

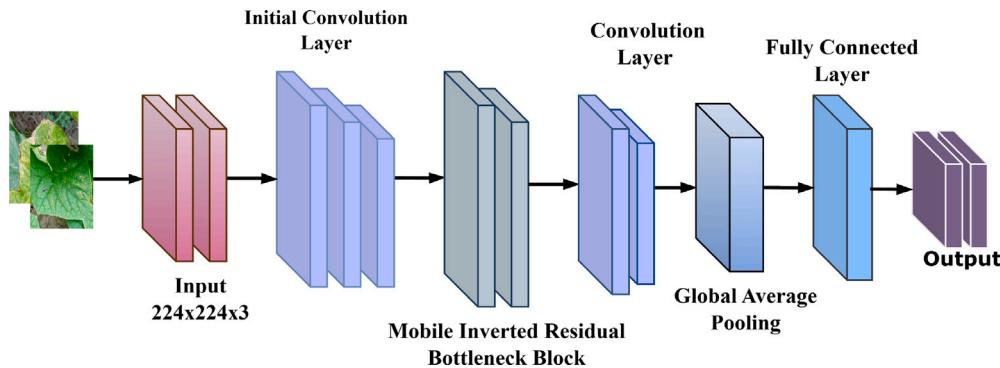


Fig. 8. EfficientNetB0 architecture.

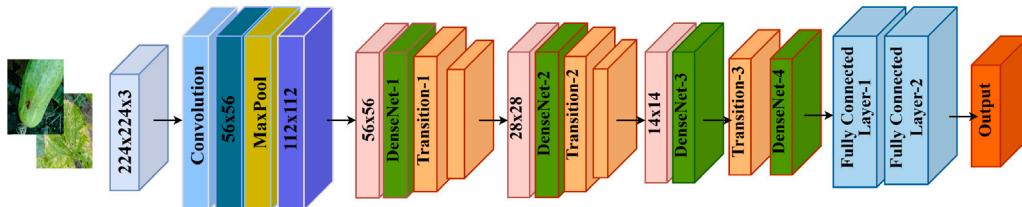


Fig. 9. DenseNet121 architecture.

4.3.4. DenseNet121

As depicted in Fig. 9, DenseNet121 is characterized by its dense connections between layers, known as densely connected convolutional networks. This design allows each layer to receive gradients directly from all preceding layers, enhancing feature propagation throughout the network. Comprising 121 layers, DenseNet121 is lauded for its efficient reuse of features and economical use of parameters, promoting a rich representation of learned features while maintaining model compactness.

4.3.5. Xception

As illustrated in Fig. 10, Xception is a modified version of the Inception architecture. It employs depth-wise separable convolutions, replacing the traditional convolutional layers, aiming to optimize computational efficiency without compromising performance. Xception is recognized for its ability to adeptly balance proficiency and accuracy in computational tasks, enhancing the model's overall effectiveness.

4.3.6. MobileNetV2

Fig. 11 presents MobileNetV2 is a lightweight and efficient deep convolutional neural network (CNN) architecture that was developed by Google AI. It is designed for mobile and embedded vision applications, where processing resources are limited. MobileNetV2 uses

a number of techniques to reduce the number of parameters and FLOPs required, such as depthwise separable convolutions and inverted residual blocks. Depthwise separable convolutions factorize a standard convolution into two separate convolutions: a depthwise convolution and a pointwise convolution. This reduces the number of parameters and FLOPs by up to 90%. Inverted residual blocks are a type of residual block that places the non-linearity before the summation operation. This allows MobileNetV2 to learn more complex features without sacrificing efficiency.

4.4. Ensembling approach

Ensemble learning involves combining multiple deep learning models, each trained independently, to improve overall predictive performance. As shown in Fig. 12, the process starts by splitting the image dataset into training, validation, and testing sets, followed by applying data augmentation techniques (e.g., random rotations, flips, zooms, and contrast adjustments) to enrich the dataset. In this framework, we train seven models individually: CNN, DenseNet121, EfficientNetB0, InceptionV3, MobileNetV2, ResNet50, and Xception.

After training, the outputs (probability distributions) of these models are aggregated. While a simple average pooling layer can be used to

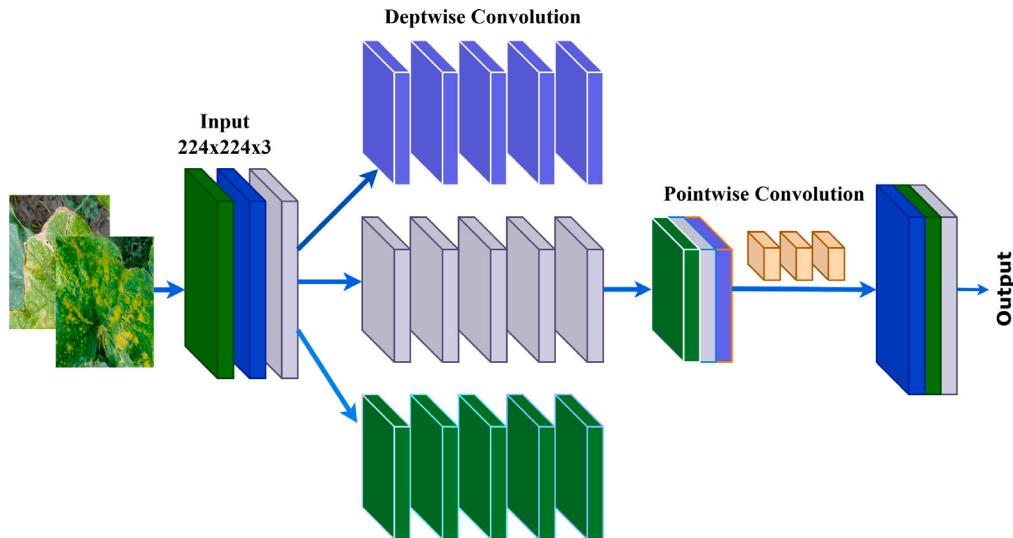


Fig. 10. Xception architecture.

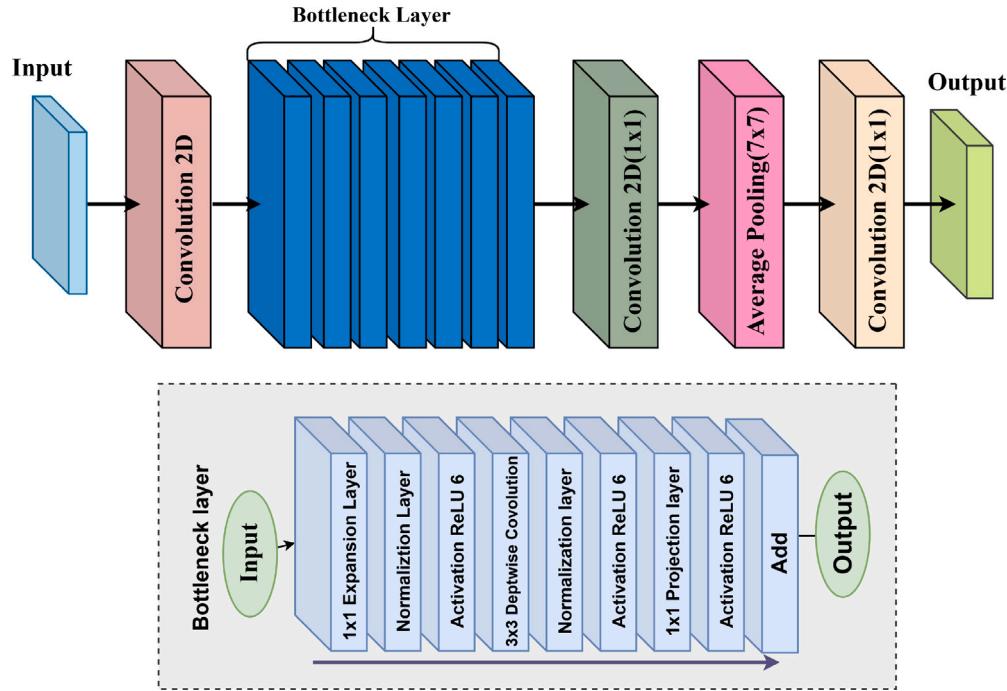


Fig. 11. MobileNetV2 architecture.

combine the logits or probability vectors directly, we adopt a weighted averaging approach to ensure that more accurate models contribute more significantly to the final prediction. Let each model M_i produce a probability vector

$$\mathbf{p}_i = (p_i(c_1), p_i(c_2), \dots, p_i(c_K)),$$

where $p_i(c_j)$ represents the predicted probability that the input belongs to class c_j , and K is the total number of classes. We define a set of weights

$$w_1, w_2, \dots, w_7,$$

one for each of the seven models. The final ensemble prediction $\hat{\mathbf{p}}$ for each class c_j is computed as:

$$\hat{p}(c_j) = \frac{\sum_{i=1}^7 w_i p_i(c_j)}{\sum_{i=1}^7 w_i}.$$

The predicted class \hat{y} is then determined by:

$$\hat{y} = \arg \max_{c_j \in \{c_1, \dots, c_K\}} \hat{p}(c_j).$$

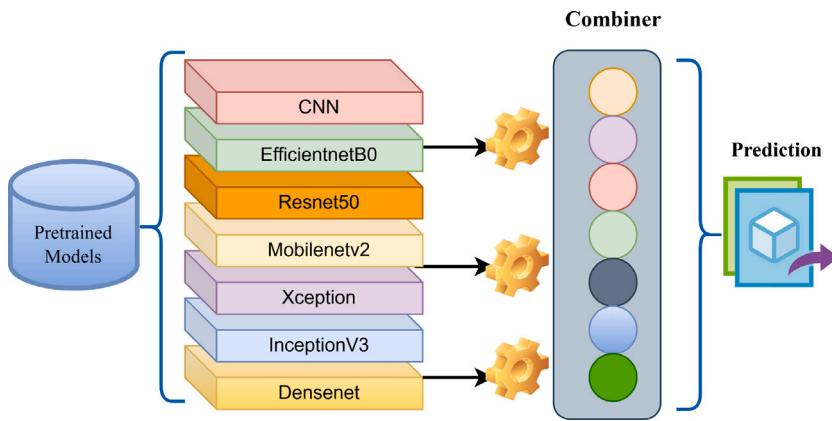


Fig. 12. Ensembling approach for leaf disease detection, combining multiple deep learning models for enhanced accuracy and stability.

By adjusting the weights w_i , one can give higher importance to more reliable models based on metrics such as validation accuracy or F1-scores. During the creation of the ensemble model, the choice of loss function, optimizer, and evaluation metrics is specified, ensuring effective end-to-end training. The training phase monitors validation performance to enable early stopping and dynamic learning rate adjustments if necessary. This procedure exploits the complementary advantages of diverse architectures, ultimately boosting the accuracy and stability of leaf disease detection. The detailed steps of this ensembling process are summarized in Algorithm 1.

Algorithm 1 Ensemble Learning for Image Classification

```

1: Input: Image dataset  $D$ 
2: Output: Trained ensemble model  $E$ 
3: Split the dataset  $D$  into training set  $D_{train}$ , validation set  $D_{val}$ , and testing set  $D_{test}$ 
4: Apply data augmentation techniques on  $D_{train}$  to create augmented dataset  $D_{aug}$ 
5: Initialize seven models:  $M_1$  (CNN),  $M_2$  (DenseNet121),  $M_3$  (EfficientNetB0),  $M_4$  (InceptionV3),  $M_5$  (MobileNetV2),  $M_6$  (ResNet50),  $M_7$  (Xception)
6: for each model  $M_i$  in  $\{M_1, M_2, M_3, M_4, M_5, M_6, M_7\}$  do
7:   Train  $M_i$  on  $D_{aug}$ 
8: end for
9: Create an average pooling layer to combine the outputs of the seven trained models
10: Define the ensemble model  $E$  by stacking the average pooling layer on top of the combined models
11: Provide evaluation metrics, loss functions, and optimizers for model compilation
12: Compile the ensemble model  $E$ 
13: Train the ensemble model  $E$  on  $D_{train}$ 
14: while not converged do
15:   Monitor validation performance on  $D_{val}$ 
16:   Adjust learning rates dynamically
17:   if early stopping criteria met then
18:     Stop training
19:   end if
20: end while
21: Evaluate the ensemble model  $E$  on  $D_{test}$  to assess predictive performance
22: Return Trained ensemble model  $E$ 
```

4.5. Performance measures

The following formula defines accuracy:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

where in Eq. (1), TP denotes true positives, TN denotes true negatives, FP denotes false positives, and FN denotes false negatives.

Recall (R) quantifies the number of correct positive class predictions made out of all positive examples in the dataset. Precision (P) quantifies the number of correct positive class predictions made out of all positive predictions. The F1-score ($F1$) is another metric for assessing a model's accuracy on a dataset, which combines precision and recall into a single statistic. The following equations illustrate the formulas for precision, recall, and the F1-score:

$$\text{Precision}, P = \frac{TP}{TP + FP} \quad (2)$$

$$\text{Recall}, R = \frac{TP}{TP + FN} \quad (3)$$

$$F1 = \frac{2 \cdot P \cdot R}{P + R} \quad (4)$$

In Eq. (2), P represents precision, where TP is true positives and FP is false positives. In Eq. (3), R represents recall, where TP is true positives and FN is false negatives. In Eq. (4), $F1$ represents the F1-score, which is the harmonic mean of precision and recall.

5. Experimental analysis

5.1. Experimental setup

The proposed model includes data augmentation techniques to enhance the quality and diversity of the training dataset. Techniques such as random cropping, flipping, and rotating of images were applied to expand the range of training examples. To thoroughly evaluate the performance of the CNN algorithms used in this study, various performance metrics were considered. Experiments were conducted on Google Colab² using a T4 GPU. The setup included PyTorch 1.13.0, CUDA 11.3, and Python 3.8.12. For the selected models, the Adam optimizer with a cross-entropy loss function was used during the training phase. Performance evaluation was carried out using test accuracy and loss curves to understand the model's learning behavior and its ability to generalize. In addition, classification reports and confusion matrices were generated to assess the model's precision, recall, and F1-score across different classes, which are essential for evaluating its effectiveness in cucumber leaf disease detection. Table 7 presents the summary of the parameters used for deep learning models.

The total training time for the seven individual deep learning models was approximately 21 h, with each model taking an average of 3 h to train. The proposed ensemble model, which combines all individual models, required an additional 4 h for training, bringing the total training time to 25 h.

² <https://colab.google/>

Table 7

Hyper parameters used for deep learning models.

Parameter name	Value/Description
Batch Size	32
Target Image Size	(224 x 224) pixels
Preprocessing Function	imagenet_utils.preprocess_input
Data Augmentation	Resizing, Rescaling, Random flip, Random Rotation, Random Zoom, Contrast
Training/Validation Size	80% training, 20% validation
Testing Size	20% testing
Loss Function	Categorical Crossentropy
Optimizer	Adam (learning rate 0.0001)
Epochs	150
Callbacks	Early stopping (patience:5), Model checkpoint, Reduce_lr
Ensemble Aggregation	Averaging

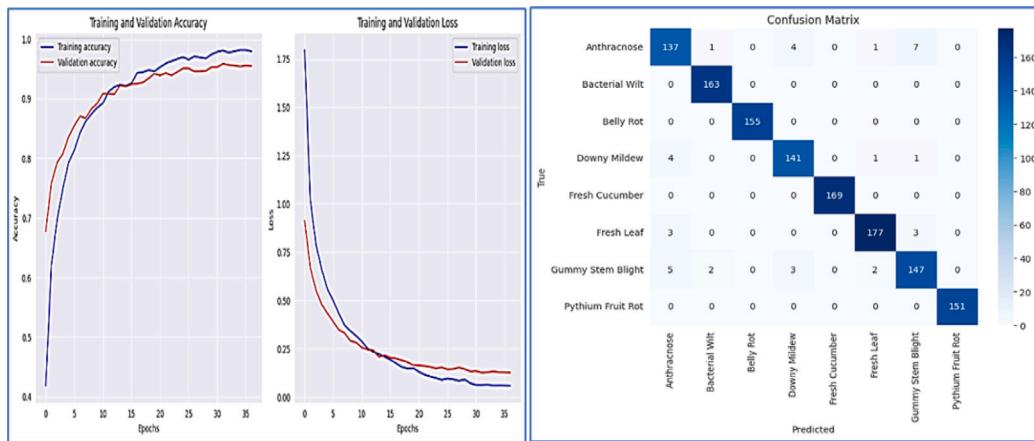


Fig. 13. Model evaluation results of EfficientNetB0.

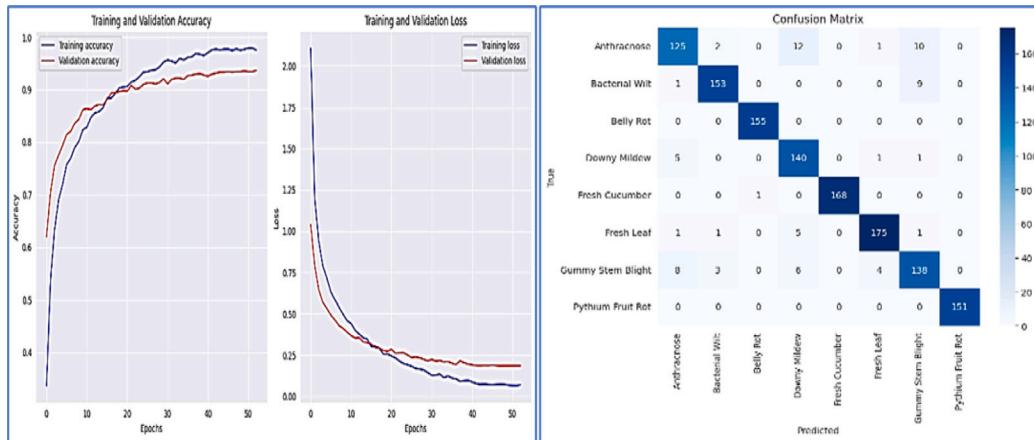


Fig. 14. Model evaluation results of DenseNet121.

5.2. Individual model performance

5.2.1. EfficientNetB0

EfficientNetB0 demonstrated proficiency in classification tasks, achieving a commendable accuracy of 97.10%, as depicted in Fig. 13. Belonging to the distinguished EfficientNet family, this model is celebrated for its enhanced efficiency and exceptional performance metrics. The architecture of EfficientNetB0 is meticulously designed to amplify network depth, width, and resolution, which is instrumental in its ability to adeptly identify and encapsulate pivotal features and patterns within the dataset.

5.2.2. DenseNet121

DenseNet121 achieved a notable accuracy rate of 94.36%, as visualized in Fig. 14. The architecture of DenseNet models is characterized by feed-forward connections across every layer. This design promotes the reuse of features and fosters an environment conducive to efficient feature learning. Such structural considerations significantly contribute to DenseNet121's competitive accuracy in identifying and classifying various states of cucumber diseases.

5.2.3. ResNet50

The ResNet-50 model demonstrated notable success by achieving an accuracy rate of 94.60%, as exhibited in Fig. 15. ResNet architectures are uniquely equipped with skip connections, a feature that effectively mitigates the vanishing gradient problem, enhancing the model's

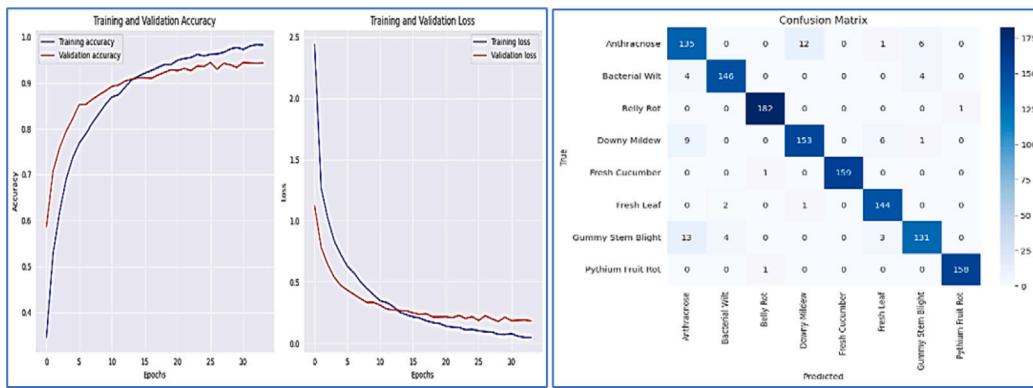


Fig. 15. Model evaluation results of ResNet-50.

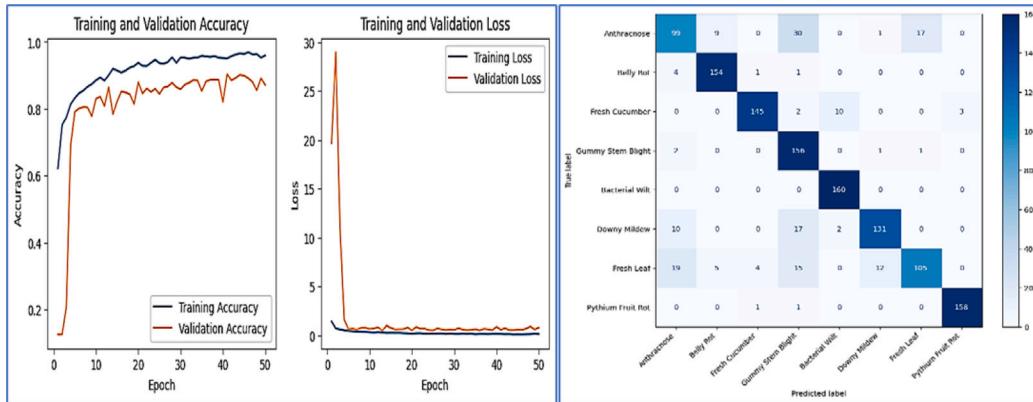


Fig. 16. Model evaluation results of custom CNN.

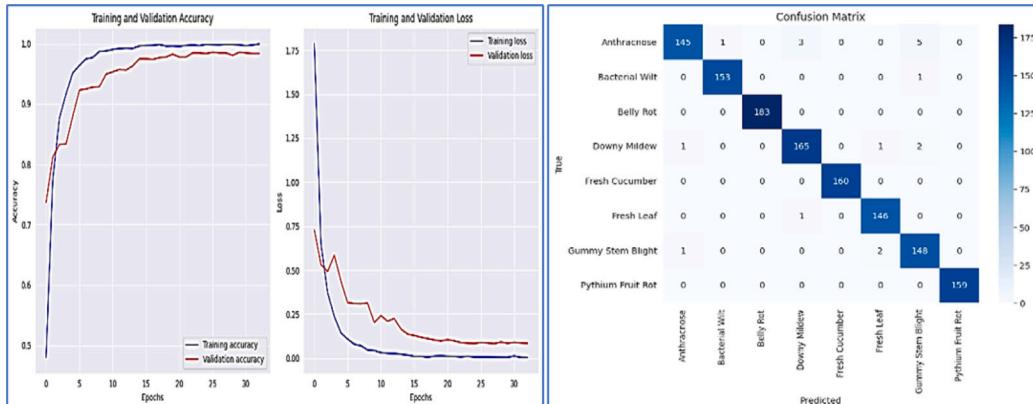


Fig. 17. Model evaluation results of MobileNetV2.

learning capacity. Due to this architecture, ResNet models, including ResNet-50, excel at training profoundly deep networks, enabling the capture of complex features and patterns, which significantly contribute to their high precision in various classification tasks, including cucumber disease detection.

5.2.4. Custom CNN

The Convolutional Neural Network (CNN) used in our study attained an accuracy rate of 88.71%, as illustrated in Fig. 16. CNNs, foundational models in image classification tasks, inherently depend on their specific architecture and training configurations to determine their performance efficacy. In this context, the CNN demonstrates a commendable level of accuracy, although it appears slightly lower compared to other specialized model architectures employed in this study.

This outcome underscores the influence of model design and training strategies on the achievement of desirable classification accuracy.

5.2.5. MobileNetV2

MobileNetV2 exhibited remarkable performance in our study, achieving an accuracy rate of 98.59%, as displayed in Fig. 17. Designed explicitly for mobile and embedded devices, MobileNetV2 optimally balances accuracy with computational efficiency. Its architecture, while being lightweight, does not compromise on precision, making it exceptionally suitable for scenarios with constrained computational resources. This intrinsic adaptability and performance efficacy underline MobileNetV2's robustness as a favorable model in the domain of cucumber disease classification.

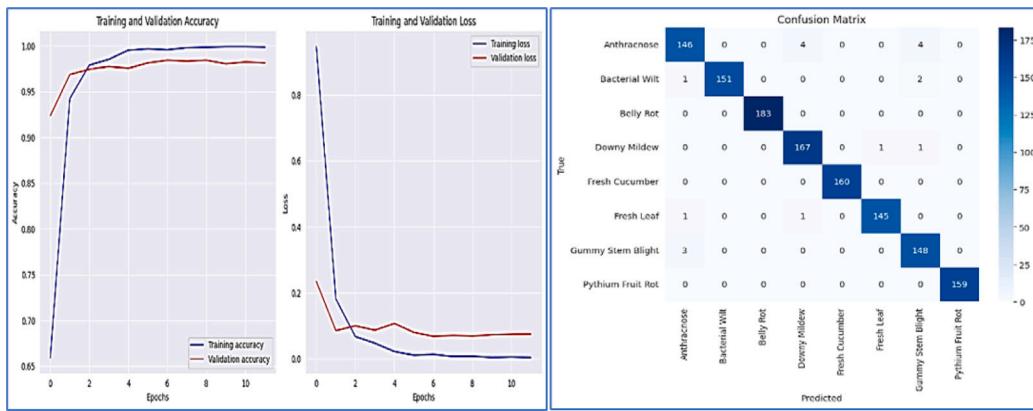


Fig. 18. Model evaluation results of Xception.

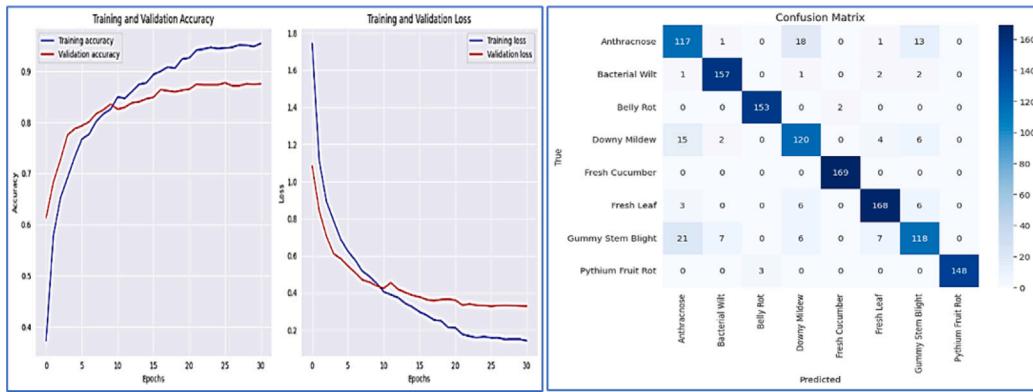


Fig. 19. Model evaluation results of InceptionV3.

5.2.6. Xception

Xception demonstrated stellar performance in our experiments, achieving an accuracy rate of 98.59%, as visualized in Fig. 18. Xception's architecture, leveraging depth-wise separable convolutions, is fundamentally adept at capturing complex patterns and features within images. This innovative architectural approach enables Xception to exhibit exceptional accuracy, marking it as particularly proficient for image classification tasks, including the detection and categorization of cucumber diseases. The sophisticated design of Xception contributes to its effectiveness in extracting meaningful representations from images, showcasing its capability in the field of image analysis.

5.2.7. InceptionV3

InceptionV3 achieved a notable accuracy of 90.05%, as depicted in Fig. 19. Known for its architectural ingenuity, InceptionV3 presents a reliable model for various classification tasks. The model delivers moderate performance in this study, suggesting its suitability and adaptability to specific datasets and tasks. The attained accuracy underscores its capability; however, it also indicates that performance may be influenced by the nature of the applied dataset or the particularities of the classification task at hand. Thus, while InceptionV3 has demonstrated competence in the classification of cucumber diseases, its performance reflects the interaction with the specific challenges and characteristics of the given task and dataset.

5.2.8. Ensembling framework

The accuracy levels for each of the different models ranged from 88.71% to 99%. These models included CNN, DenseNet121, EfficientNetB0, InceptionV3, MobileNetV2, ResNet50, and Xception. When these models were combined into an ensemble model, their performance increased to a total accuracy of 98.67%. Accurate predictions

Table 8

Performance analysis of the architectures.

No.	Model name	Precision	Recall	F1-Score	Accuracy
1	CNN	87%	87%	87%	88.71%
2	DenseNet121	94%	94%	94%	94.36%
3	EfficientNetB0	97%	97%	97%	97.10%
4	InceptionV3	90%	90%	90%	90.05%
5	MobileNetV2	99%	99%	99%	98.59%
6	ResNet50	95%	95%	95%	94.60%
7	Xception	99%	99%	99%	98.59%
8	Ensembling	99%	99%	99%	99.89%

were made by the ensemble model, as evidenced by its high accuracy, recall, and F1-score of 99%. The ensemble model's test loss was 0.220, indicating strong generalization. Overall, the accuracy and performance were significantly higher when multiple models were combined using ensemble learning compared to using individual models (see Figs. 20 and 21).

5.3. Performance analysis

Representative visualizations, including classification reports and confusion matrices (Fig. 22), were used to display the model's performance, showcasing the proportions of correctly and incorrectly classified instances. These evaluations demonstrate the model's robustness and reliability in addressing the challenges posed by cucumber disease detection.

The classification report, as presented in Table 8 and Fig. 23, provides a detailed evaluation of model performance in classifying cucumber diseases. This table compares several deep learning models, highlighting precision, recall, F1-score, and accuracy. The baseline

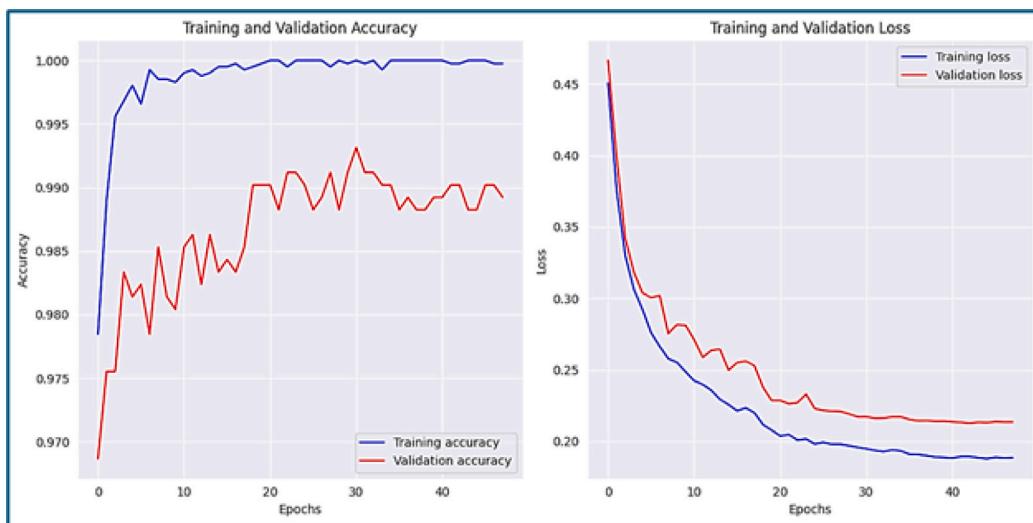


Fig. 20. Ensemble approach accuracy curve.

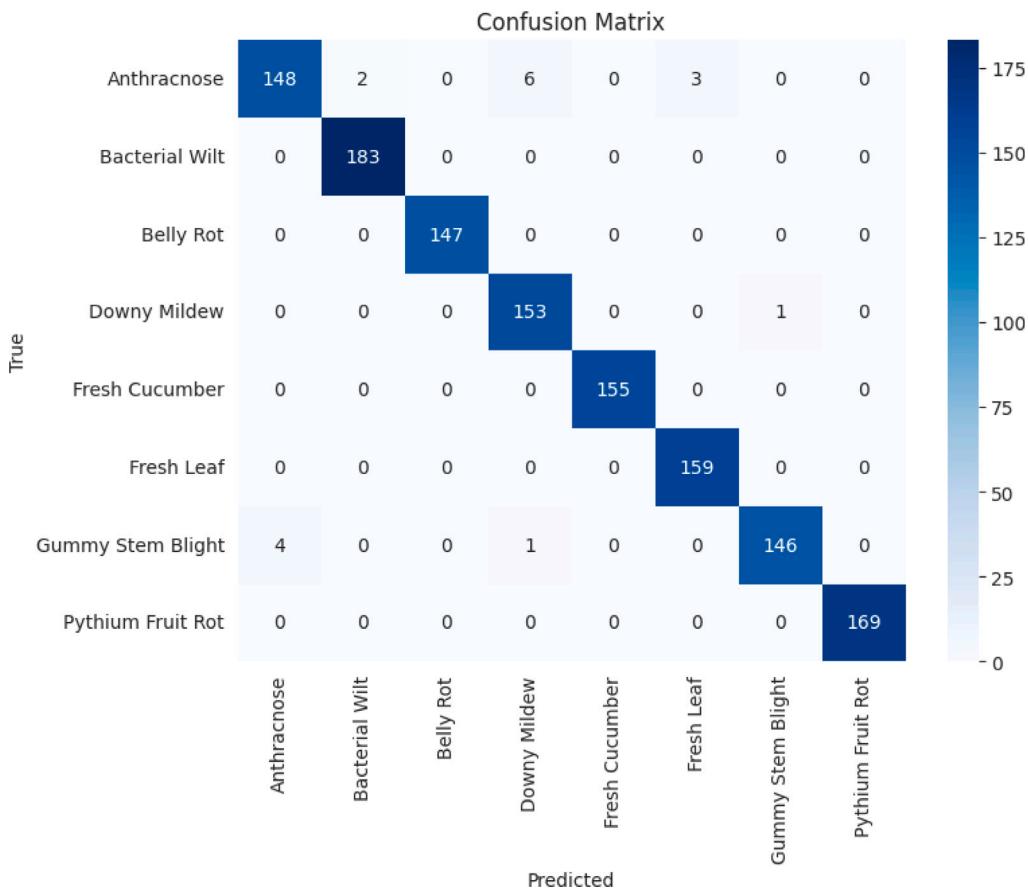


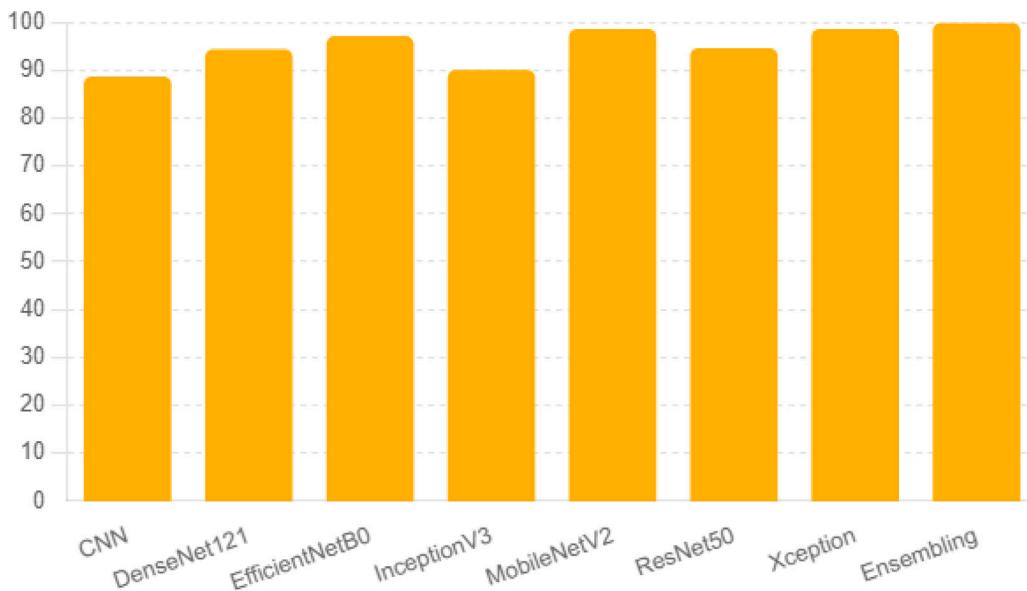
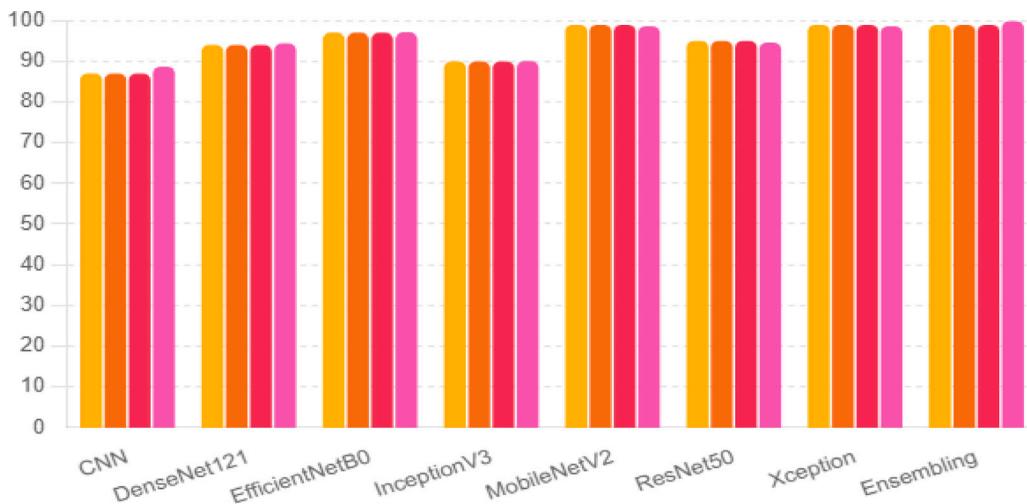
Fig. 21. Ensemble approach confusion matrix.

CNN model shows moderate results, with 87% on key metrics and 88.71% accuracy. DenseNet121 demonstrates improved performance, reaching 94% across metrics and 94.36% accuracy. EfficientNetB0 performs even better, achieving 97% in precision, recall, and F1-score, with an accuracy of 97.10%. MobileNetV2 and Xception achieve near-perfect performance, both with 99% in precision, recall, and F1-scores, reaching 98.59% accuracy. The ensemble model, which combines the strengths of these individual models, achieves 99% across key metrics

with an overall accuracy of 99.89%, underscoring the value of ensemble learning for enhanced classification performance.

5.4. Explainable AI for model interpretability

In this study, we utilized multiple explainable AI (XAI) techniques – Grad-CAM (Gradient-weighted Class Activation Mapping) [47], Grad-CAM++ [48], and Eigen-CAM [49] – to visualize and interpret the model's decision-making process during cucumber disease detection.

**Fig. 22.** Performance comparison based on Model Accuracy.**Fig. 23.** Comparison of model performance based on precision, recall, F1-score, test loss, and test accuracy.

5.4.1. Grad-CAM

Grad-CAM [47] highlights regions of the image that influence the model's predictions by using the gradient information flowing into the final convolutional layer. As shown in Fig. 24, warmer colors (reds and yellows) correspond to regions identified as critical for disease recognition, while cooler colors (blues and greens) indicate less influential areas.

5.4.2. Grad-CAM++

Grad-CAM++ [48] refines Grad-CAM by considering higher-order gradients, enabling more precise localization of important regions. This is particularly useful for images with multiple regions of interest, as it better delineates fine-grained features relevant to disease detection. In Fig. 24, Grad-CAM++ outputs show sharper and more focused heatmaps, making it easier to identify the exact regions associated with disease symptoms.

5.4.3. Eigen-CAM

Eigen-CAM [49] uses principal components of feature maps to generate heatmaps, offering a class-independent visualization of important regions. This method provides unique insights into the global patterns

captured by the model, making it a valuable complement to Grad-CAM and Grad-CAM++. As shown in Fig. 24, Eigen-CAM highlights broader structural features that may contribute to disease identification, ensuring a more comprehensive interpretability framework.

5.4.4. Application development for real-world use

To translate research into practical solutions, an interactive web application was developed, incorporating the explainable AI (XAI) techniques used in this study. As shown in Fig. 25, the application enables agricultural experts and end-users to upload images of cucumber plants and view the model's predictions alongside detailed heatmaps generated by Grad-CAM, Grad-CAM++, and Eigen-CAM. These heatmaps provide a visual explanation of the model's focus areas during disease detection, enhancing transparency and interoperability. This application bridges the gap between technical advancements and field-level usability by offering a user-friendly interface that supports effective decision-making. By visualizing the areas of interest identified by the model, users can gain deeper insights into disease patterns and symptoms.

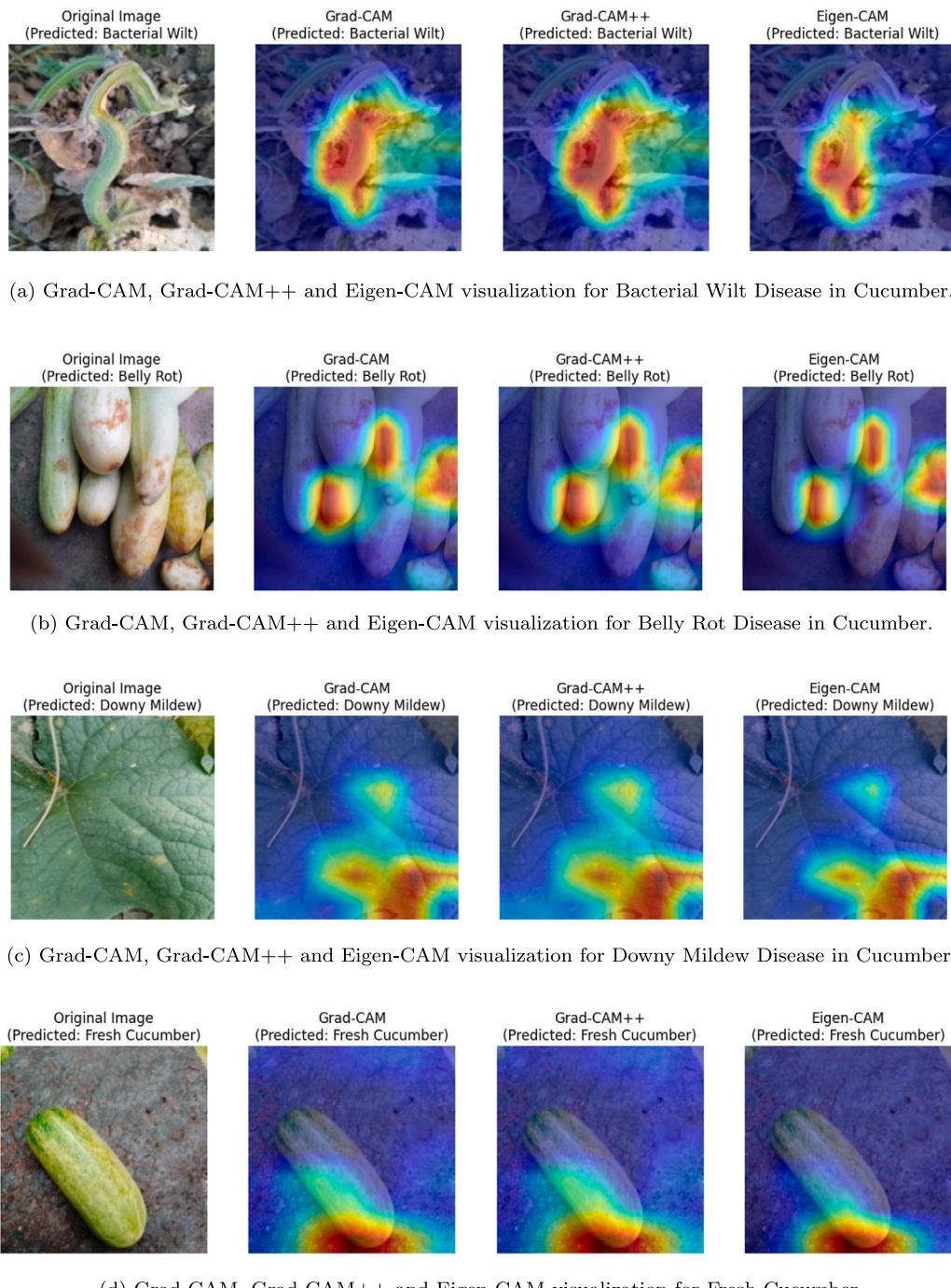


Fig. 24. Grad-CAM, Grad-CAM++ and Eigen-CAM visualizations highlighting areas of interest for various cucumber diseases.

6. Discussion

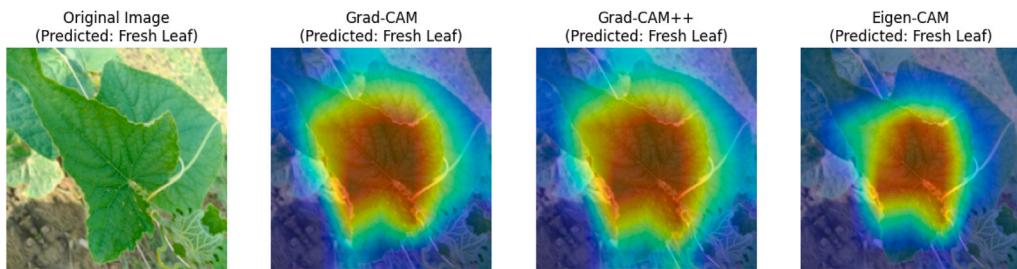
In this section, we analyze the performance of the deep learning models, the impact of the ensemble approach, the role of Explainable AI (XAI) in enhancing model interpretability, limitations of the study, cost analysis, and a comparison with state-of-the-art models.

6.1. Explainable AI and model interpretability

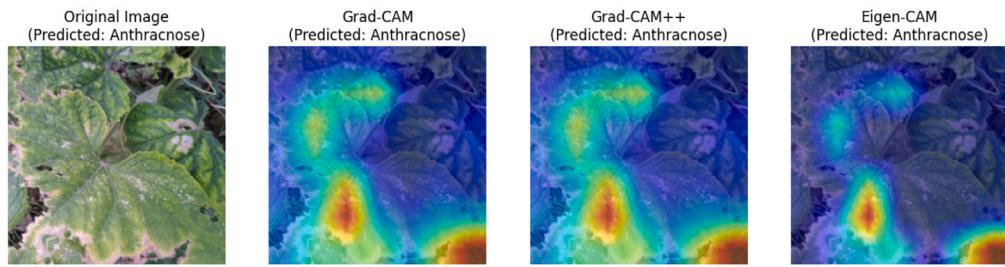
The proposed ensemble framework, comprising CNN, DenseNet121, EfficientNetB0, InceptionV3, MobileNetV2, ResNet50, and Xception, achieved an impressive overall accuracy of 99%. With recall and F1-score values of 99% and a test loss of 0.220, the ensemble model

demonstrated strong generalization capabilities. By leveraging the complementary strengths of these individual architectures, the ensemble approach enhanced prediction reliability and robustness, outperforming any single model.

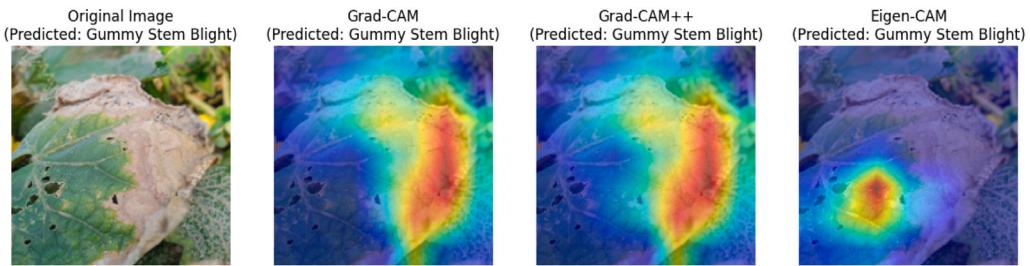
To further improve model interpretability, Explainable AI (XAI) techniques, including Grad-CAM (Gradient-weighted Class Activation Mapping), Grad-CAM++, and Eigen-CAM, were employed. These techniques provided detailed visualizations highlighting the regions of input images where the models focused their attention during predictions. Grad-CAM generated heatmaps that revealed the critical areas contributing most to disease classification, with warmer colors (reds and yellows) indicating regions of higher importance. Grad-CAM++ refined this process by incorporating higher-order gradients, enabling



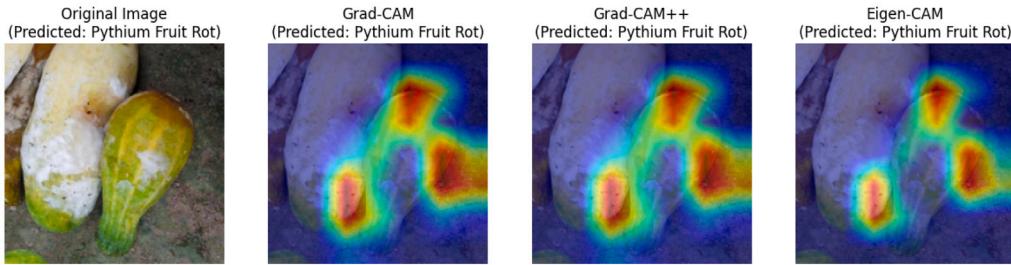
(e) Grad-CAM, Grad-CAM++ and Eigen-CAM visualization for Fresh Leaf in Cucumber.



(f) Grad-CAM, Grad-CAM++ and Eigen-CAM visualization for Anthracnose Disease in Cucumber.



(g) Grad-CAM, Grad-CAM++ and Eigen-CAM visualization for Gummy Stem Blight Disease in Cucumber.



(h) Grad-CAM, Grad-CAM++ and Eigen-CAM visualization for Pythium Fruit Rot Disease in Cucumber.

Fig. 24. (continued).

more precise localization of important features, particularly in complex cases. Eigen-CAM offered a broader perspective by using principal components of feature maps to visualize global patterns captured by the models. Together, these XAI techniques offered a comprehensive understanding of how the ensemble identified key features associated with different disease types.

The heatmaps, as shown in Fig. 24, not only enhanced interpretability but also demonstrated the ensemble's ability to integrate diverse strengths from its constituent models. By employing Grad-CAM, Grad-CAM++, and Eigen-CAM, this study provides granular insights into the decision-making processes of both individual models and the ensemble. These visualizations offer several practical benefits:

- **Disease-Specific Insights:** Enable the identification of critical regions on cucumber leaves that signify specific diseases.
- **Model Validation and Refinement:** Allow researchers to validate and fine-tune predictions in complex or uncertain scenarios.
- **Enhanced Transparency:** Foster trust in the model by providing clear and interpretable visual explanations for its classification outcomes.

6.2. Comparison with similar studies

As shown in Table 9, our proposed ensemble framework achieved a notable accuracy rate of 99%, positioning it among the top-performing

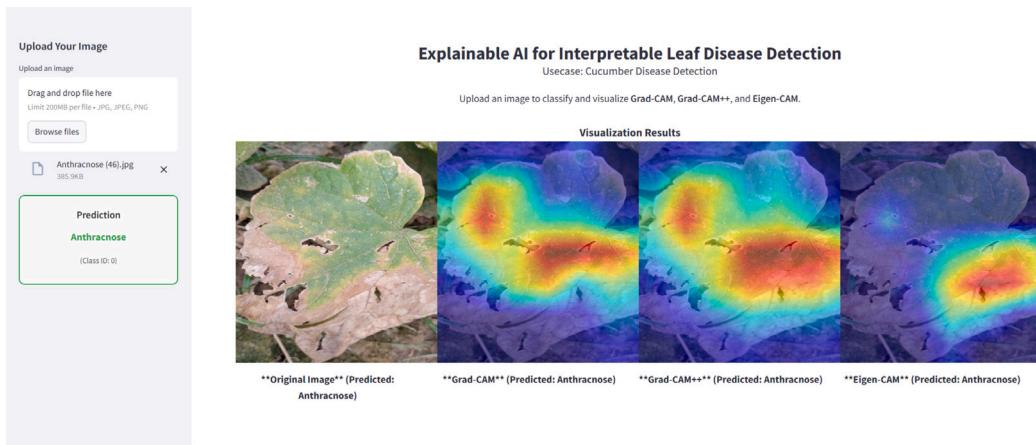


Fig. 25. Interactive web application for cucumber disease detection, showcasing model predictions and heatmaps generated by Grad-CAM, Grad-CAM++, and Eigen-CAM.

Table 9

Comparison of accuracy rates in plant leaf disease detection studies with the proposed model.

Paper	Model and performance (Accuracy in %)
[1]	GPDCNN - 90%
[14]	ANN - 80.45%
[15]	KNN - 94.30%
[16]	CNN - 94.9%
[23]	SVM - 86%
[4]	CNN - 98.029%
[2]	ANN - 89.9%
[26]	DCNN - 93.4%
[27]	VGGNet3 - 92.00%
[28]	CNN - 98.19%
[5]	Meta learning - 98.53%
Proposed Model	Ensembling framework 99%

models in plant leaf disease detection. Existing studies report accuracy levels ranging from 80.45% to an optimal 100%, highlighting a broad spectrum of results depending on the model and dataset used. **Table 9** provides a comparative overview of accuracy rates from various studies. The accuracy spans a wide range, with Artificial Neural Networks (ANN) reaching 80.45% [14]. Our proposed ensemble model, with an accuracy of 99%, stands competitively with some of the highest results in the literature. The ensemble approach effectively balances high accuracy with robustness by combining multiple architectures.

6.3. Cost analysis and feasibility

The cost analysis presented provides a clear breakdown of the investment required for deploying this cucumber disease detection system, which relies on deep learning models. The estimated expenses are divided into hardware and software costs, totaling approximately 147,860 BDT. This includes a hardware cost of 137,160 BDT, covering computational resources necessary for training and running the models, and a software cost of 10,700 BDT, which accounts for any specialized software or libraries used for model deployment.

This investment is feasible given the potential benefits for large-scale agricultural applications. By automating disease detection, the system can help farmers and agricultural professionals identify and manage plant diseases early, reducing crop loss and enhancing yield. The return on investment becomes apparent when considering the reduction in labor costs and potential improvements in crop quality and yield.

For real-world application in diverse agricultural environments, further cost reductions may be achievable. Optimizing the model architectures to operate on less resource-intensive hardware could lower

operational expenses, making the system more accessible to farmers with limited resources. Additionally, the option to deploy the model on mobile devices or edge computing devices could further improve feasibility, especially in regions with limited access to high-performance hardware. The affordability and accessibility of this system make it a practical tool for modernizing disease management practices in agriculture.

6.4. Scalability and real-world integration

The proposed ensemble framework is designed not only for high accuracy in controlled experimental settings but also with practical deployment and scalability in mind. This section discusses the potential for integrating the system into real-world agricultural environments, emphasizing its suitability for edge devices, smart farming systems, and a maintenance framework for continuous learning.

By leveraging lightweight architectures such as MobileNetV2 and EfficientNetB0, the proposed framework is well-suited for implementation on resource-constrained edge devices (e.g., Raspberry Pi and IoT-based systems). Deploying the model on such devices enables real-time disease monitoring directly in the field, reducing latency in decision-making and facilitating prompt responses to emerging plant health issues. The efficient design minimizes computational demands, allowing for low-power, cost-effective monitoring solutions in remote or underserved areas.

For enhanced agricultural management, the system can be seamlessly integrated with smart farming platforms. The disease detection outputs can be linked with existing sensor networks that monitor environmental parameters such as soil moisture, temperature, and humidity. This integration creates a comprehensive decision-support system, where data from image analysis and environmental sensors collectively inform targeted interventions, such as precise irrigation or pesticide application.

Real-world agricultural environments are dynamic, and the performance of disease detection models can be affected by changes in environmental conditions and pathogen evolution. To address this, the system is designed with a continuous learning component. Regular updates, driven by newly collected field data, can be integrated via cloud-based retraining pipelines. This adaptive mechanism ensures that the model remains robust and relevant over time. Additionally, remote monitoring and maintenance protocols can help in diagnosing issues early, enabling ongoing performance optimization without extensive manual intervention.

6.5. Limitations and future work

While this study demonstrates promising results in cucumber disease detection, several limitations must be acknowledged. Despite expanding the dataset to 6400 images, a minor class imbalance remains, with the Anthracnose class containing only 782 images compared to 800 images for other classes. This imbalance could impact the model's accuracy for underrepresented classes. Additionally, images were captured under diverse field conditions, introducing environmental noise and lighting variations, which were not fully addressed in this study. These factors may limit the model's ability to generalize effectively in real-world applications. Although the dataset was increased, further expansion could enhance model robustness, particularly for deep learning models that benefit from larger datasets.

Another limitation is the reliance on freely available resources, such as Google Colab, which restricted the scope of experiments due to server time limitations. Consequently, extensive hyperparameter tuning and the application of various optimizers like Adadelta, FTRL, and NAdam were not fully explored. Moreover, the research was conducted on publicly available secondary data rather than primary data directly collected from agricultural fields, which could limit the model's relevance to actual farming conditions.

Considering these limitations, this study still highlights the potential of deep learning and transfer learning approaches for automating disease diagnosis in crop management. The ensemble model, in particular, demonstrates enhanced performance and robustness, achieving an accuracy of 99%. These findings suggest promising directions for future research, which could focus on several key areas:

- Future work could involve collecting more balanced and high-quality primary data directly from agricultural fields, which would enhance the model's accuracy and relevance.
- Techniques to mitigate the effects of environmental noise and lighting inconsistencies should be explored to improve the model's generalization capability in real-world scenarios.
- Access to more flexible computational resources would allow for thorough hyperparameter tuning and testing with a wider range of optimizers, which could further optimize model performance.
- Plans include creating a user-friendly interface for cucumber disease detection. Such an interface would not only facilitate easy identification of diseases but also offer farmers practical advice on disease management and prevention.
- Given the widespread use of mobile phones, particularly in developing regions, a mobile-based application is envisioned to assist farmers in diagnosing cucumber diseases in real-time. This application would leverage the proposed ensemble model to deliver accurate and reliable disease diagnostics, supporting farmers in maintaining crop health effectively.

7. Conclusion

In this study, we proposed a robust ensemble learning framework for cucumber leaf disease detection, integrating multiple state-of-the-art deep learning models and Explainable AI (XAI) techniques. The framework effectively combines the strengths of individual architectures such as ResNet50, EfficientNetB0, DenseNet121, MobileNetV2, Xception, and InceptionV3, achieving a test accuracy of 99%. Rigorous data augmentation techniques enhanced the diversity and robustness of the training dataset, mitigating overfitting and enabling the model to generalize well across varied environmental conditions.

To ensure transparency and user trust, we incorporated advanced XAI techniques, including Grad-CAM, Grad-CAM++, and Eigen-CAM, which provided interpretable visualizations of the model's predictions. These techniques highlight critical regions in the input images, offering practical insights into the decision-making process. Furthermore,

the lightweight architectures included in the ensemble, such as MobileNetV2 and EfficientNetB0, make the framework suitable for deployment in resource-constrained environments, such as mobile devices or edge computing platforms.

While this framework demonstrates significant potential in agricultural disease detection, some limitations remain, including the reliance on secondary datasets and the need for broader validation in real-world field conditions. Future work will focus on expanding the dataset through primary data collection, exploring advanced hyperparameter optimization techniques, and deploying the system in real-world agricultural settings to assess its practical impact.

CRediT authorship contribution statement

Mohammad Rifat Ahmmad Rashid: Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Md. AL Ehtesum Korim:** Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Mahamudul Hasan:** Funding acquisition, Formal analysis, Data curation, Conceptualization. **Md Sawkat Ali:** Project administration, Formal analysis. **Mohammad Manzurul Islam:** Methodology, Data curation, Conceptualization. **Taskeed Jabid:** Funding acquisition, Formal analysis, Data curation. **Raihan Ul Islam:** Project administration, Investigation, Funding acquisition. **Maheen Islam:** Investigation, Funding acquisition, Formal analysis.

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.

Data availability

Data will be made available on request.

References

- [1] Zhang S, Zhang S, Zhang C, Wang X, Shi Y. Cucumber leaf disease identification with global pooling dilated convolutional neural network. *Comput Electron Agric* 2019;162:422–30.
- [2] Bhosale YH, Zanwar SR, Ali SS, Vaidya NS, Auti RA, Patil DH. Multi-plant and multi-crop leaf disease detection and classification using deep neural networks, machine learning, image processing with precision agriculture-a review. In: 2023 international conference on computer communication and informatics. ICCCI, IEEE; 2023, p. 1–7.
- [3] Eunice J, Popescu DE, Chowdary MK, Hemanth J. Deep learning-based leaf disease detection in crops using images for agricultural applications. *Agronomy* 2022;12(10):2395.
- [4] Jasim MA, Al-Tuwaijari JM. Plant leaf diseases detection and classification using image processing and deep learning techniques. In: 2020 international conference on computer science and software engineering. CSASE, IEEE; 2020, p. 259–65.
- [5] Memori MS, Kumar P, Iqbal R. Meta deep learn leaf disease identification model for cotton crop. *Computers* 2022;11(7):102.
- [6] Abbas S, Khan MA, Alhaisoni M, Tariq U, Armghan A, Alenezi F, Majumdar A, Thinnukool O. Crops leaf diseases recognition: A framework of optimum deep learning features. *Comput Mater Contin* 2023;74(1).
- [7] He F, Liu T, Tao D. Why resnet works? residuals generalize. *IEEE Trans Neural Netw Learn Syst* 2020;31(12):5349–62.
- [8] Tan M, Le Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In: International conference on machine learning. PMLR; 2019, p. 6105–14.
- [9] Iandola F, Moskewicz M, Karayev S, Girshick R, Darrell T, Keutzer K. Densenet: Implementing efficient convnet descriptor pyramids. 2014, arXiv preprint arXiv: 1404.1869.
- [10] Wu X, Liu R, Yang H, Chen Z. An xception based convolutional neural network for scene image classification with transfer learning. In: 2020 2nd international conference on information technology and computer application. ITCA, IEEE; 2020, p. 262–7.
- [11] Sandler M, Howard A, Zhu M, Zhmoginov A, Chen L-C. Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2018, p. 4510–20.

- [12] Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2016, p. 2818–26.
- [13] Shruthi U, Nagaveni V, Raghavendra B. A review on machine learning classification techniques for plant disease detection. In: 2019 5th international conference on advanced computing & communication systems. ICACCS, IEEE; 2019, p. 281–4.
- [14] Pawar P, Turkar V, Patil P. Cucumber disease detection using artificial neural network. In: 2016 international conference on inventive computation technologies. ICICT, 3, IEEE; 2016, p. 1–5.
- [15] Kainat J, Sajid Ullah S, Alharithi FS, Alroobaea R, Hussain S, Nazir S. Blended features classification of leaf-based cucumber disease using image processing techniques. Complexity 2021;2021:1–12.
- [16] Kawasaki Y, Uga H, Kagiwada S, Iyatomi H. Basic study of automated diagnosis of viral plant diseases using convolutional neural networks. In: Advances in visual computing: 11th international symposium, ISVC 2015, las vegas, NV, USA, December 14–16, 2015, proceedings, Part II 11. Springer; 2015, p. 638–45.
- [17] El-Kenawy E-SM, Khodadadi N, Mirjalili S, Abdelhamid AA, Eid MM, Ibrahim A. Greylag goose optimization: nature-inspired optimization algorithm. Expert Syst Appl 2024;238:122147.
- [18] Ahmed M, Gamal M, Ismail I, El-Din H. An AI-based system for predicting renewable energy power output using advanced optimization algorithms. J Artif Intell Metaheuristics 2024;8(1):1–8.
- [19] Jiang D, Hao M, Ding F, Fu J, Li M. Mapping the transmission risk of zika virus using machine learning models. Acta Trop 2018;185:391–9.
- [20] Martínez-Del-Río-Ortega R, Civit-Masot J, Luna-Perejón F, Domínguez-Morales M. Brain tumor detection using magnetic resonance imaging and convolutional neural networks. Big Data Cogn Comput 2024;8(9):123.
- [21] Civit-Masot J, Luna-Perejón F, Muñoz-Saavedra L, Domínguez-Morales M, Civit A. A lightweight xAI approach to cervical cancer classification. Med Biol Eng Comput 2024;62(8):2281–304.
- [22] Muñoz-Saavedra L, Escobar-Linero E, Civit-Masot J, Luna-Perejón F, Civit A, Domínguez-Morales M. A robust ensemble of convolutional neural networks for the detection of monkeypox disease from skin images. Sensors 2023;23(16):7134.
- [23] Krishnakumar A, Narayanan A. A system for plant disease classification and severity estimation using machine learning techniques. In: Proceedings of the international conference on ISMAC in computational vision and bio-engineering 2018. ISMAC-CVB, Springer; 2019, p. 447–57.
- [24] Isty MN, Vasker N, Anjum F, Parsub WR, Roy A, Haque A, Hasan M, Islam M. Deep learning for real-time leaf disease detection: Revolutionizing apple orchard health. In: 2023 4th international conference on big data analytics and practices. IBDAP, 2023, p. 1–6. <http://dx.doi.org/10.1109/IBDAP58581.2023.10271987>.
- [25] Saleem MH, Potgieter J, Arif KM. Plant disease detection and classification by deep learning. Plants 2019;8(11):468.
- [26] Ma J, Du K, Zheng F, Zhang L, Gong Z, Sun Z. A recognition method for cucumber diseases using leaf symptom images based on deep convolutional neural network. Comput Electron Agric 2018;154:18–24.
- [27] Chen J, Chen J, Zhang D, Sun Y, Nanekaran YA. Using deep transfer learning for image-based plant disease identification. Comput Electron Agric 2020;173:105393.
- [28] Omer SM, Ghafoor KZ, Askar SK, et al. An intelligent system for cucumber leaf disease diagnosis based on the tuned convolutional neural network algorithm. Mob Inf Syst 2022;2022.
- [29] Al-Akkam RMJ, Altaei MSM. Plants leaf diseases detection using deep learning. Iraqi J Sci 2022;801–16.
- [30] Uoc NQ, Duong NT, Son LA, Thanh BD. A novel automatic detecting system for cucumber disease based on the convolution neural network algorithm. GMSARN Int J 2022;16:295–301.
- [31] Kaur H, Prashar D, Kumar V. Deep CNN models in plant disease identification. Adv Appl Math Sci 2022;21(5):2681–93.
- [32] Elsayed MA, Hafeez TA-E. New edge detection technique based on the shannon entropy in gray level images. 2012, arXiv preprint [arXiv:1211.2502](https://arxiv.org/abs/1211.2502).
- [33] Abd ElHafeez T. A new system for extracting and detecting skin color regions from pdf documents. Int J Comput Sci Eng (IJCSE) 2010;9(2):2838–46.
- [34] Gergis M, Mahmoud TM, Abd-El-Hafeez T. An approach to image extraction and accurate skin detection from web pages. Int J Comput Inf Eng 2007. (Accessed 20 October 2024).
- [35] El-Hafeez TA, Zaki A, Saabia AA-BR, El-Hafeez T, Zaki AM. Face recognition based on grey wolf optimization for feature selection. In: Proceedings of the international conference on advanced intelligent systems, vol. 845, Springer; 2019, p. 273–83.
- [36] Ali A, El-Hafeez T, Mohany Y. A robust and efficient system to detect human faces based on facial features. Asian J Res Comput Sci 2019. (Accessed 20 October 2024).
- [37] Ali A, El-Hafeez T, Mohany YK. An accurate system for face detection and recognition. J Adv Math Comput Sci 2019;33(3):1–19. <http://dx.doi.org/10.9734/JAMCS/2019/v33i330178>.
- [38] Eman M, Mahmoud T, Ibrahim M, El-Hafeez TA. Innovative hybrid approach for masked face recognition using pretrained mask detection and segmentation, robust PCA, and KNN classifier. Sensors 2023. <http://dx.doi.org/10.3390/s23156727>.
- [39] Taha M, Mostafa T, El-Rahman A, El-Hafeez TA. A novel hybrid approach to masked face recognition using robust PCA and GOA optimizer. Sci J Damietta Fac Sci 2023;13(3):25–35. <http://dx.doi.org/10.21608/SJDPS.2023.222524.1117>.
- [40] Hassan E, Elbedwehy S, Shams M, El-Hafeez TA, El-Rashidy N. Optimizing poultry audio signal classification with deep learning and burn layer fusion. J Big Data 2024;11(1):135.
- [41] Sultana N, Sharif SB, Akter M, Uddin MS. A dataset for successful recognition of cucumber diseases. Data Brief 2023;109320.
- [42] Chen Y, Dai G. Antifungal activity of plant extracts against colletotrichum lagenarium, the causal agent of anthracnose in cucumber. J Sci Food Agric 2012;92(9):1937–43.
- [43] Zhang S, Raza W, Yang X, Hu J, Huang Q, Xu Y, Liu X, Ran W, Shen Q. Control of fusarium wilt disease of cucumber plants with the application of a bioorganic fertilizer. Biol Fertil Soils 2008;44:1073–80.
- [44] Wehner TC, Amand PCS. Field tests for cucumber resistance to gummy stem blight in north carolina. HortScience 1993;28(4):327–9.
- [45] Savory EA, Granke LL, Quesada-Ocampo LM, Varbanova M, Hausbeck MK, Day B. The cucurbit downy mildew pathogen pseudoperonospora cubensis. Mol Plant Pathol 2011;12(3):217–26.
- [46] Wu J. Introduction to convolutional neural networks. Natl Key Lab Nov Soft Technol Nanjing Univ China 2017;5(23):495.
- [47] Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D. Grad-CAM: visual explanations from deep networks via gradient-based localization. Int J Comput Vis 2020;128:336–59.
- [48] Chattopadhyay A, Sarkar A, Howlader P, Balasubramanian VN. Grad-CAM++: Generalized gradient-based visual explanations for deep convolutional networks. 2017, CoRR abs/1710.11063, arXiv:1710.11063, URL <http://arxiv.org/abs/1710.11063>.
- [49] Muhammad MB, Yeasin M. Eigen-CAM: Class activation map using principal components. 2020, CoRR abs/2008.00299, arXiv:2008.00299, URL <https://arxiv.org/abs/2008.00299>.