

RESEARCH ARTICLE

Plant Leaf Disease Detection Using Ensemble Learning and Explainable AI

AMMAR OAD¹, SYED SHOAIB ABBAS², AMNA ZAFAR², BEENISH AYESHA AKRAM²,
FENG DONG¹, MIR SAJJAD HUSSAIN TALPUR³, (Member, IEEE),
AND MUEEN UDDIN⁴, (Senior Member, IEEE)

¹Faculty of Information Engineering, Shaoyang University, Shaoyang 422000, China

²Department of Computer Science, University of Engineering and Technology at Lahore, Lahore 39161, Pakistan

³Information Technology Centre, Sindh Agriculture University, Tando Jam 70060, Pakistan

⁴College of Computing and Information Technology, University of Doha for Science and Technology, Doha, Qatar

Corresponding authors: Ammar Oad (ammar.oad@hnsyu.edu.cn) and Mueen Uddin (mueen.malik@ieee.org)

This open research was funded by Qatar National Library (QNL).

ABSTRACT Plants are integral to the agriculture industry, profoundly impacting a nation's economy and environmental stability, with a significant portion of certain countries' economies reliant on crop production. Much like human health, plants face susceptibility to diseases induced by viruses and bacteria, necessitating careful attention to plant care and disease identification. This study introduces an AI (Artificial Intelligence) model that detects and explains plant diseases through image analysis. The proposed system, distinct from existing detectors, identifies numerous diseases in vegetables and fruits by employing our proposed ensemble learning classifier involving four deep learning models: VGG16, VGG19, ResNet101 V2, and Inception V3, achieving an accuracy exceeding 90%. The reason for using ensemble learning is to obtain accurate predictions. Furthermore, the system sets itself apart by providing explanations for predictions using LIME (Local Interpretable Model-Agnostic Explanations), applied to interpret the predictions of deep learning models. The visualizations generated from multiple methods point to specific pixels' influence on accurate and incorrect predictions, clearly illustrating the model's decision-making process. This technique shows areas of the image that contributed positively to the model's decision, like key regions where the object of interest was most prominent, and areas that added negative values, where irrelevant or misleading features were present. By exploring these features, we gained insights into how the model interprets and prioritizes different aspects of the image during prediction. The study aims to address existing limitations in plant disease detection, offering a comprehensive solution to enhance agricultural practices, foster economic growth, and contribute to environmental sustainability.

INDEX TERMS Deep learning (DL), convolution neural network (CNN), explainable AI (XAI), ensemble learning, plant diseases, plant village.

I. INTRODUCTION

Farming plays an essential part in every aspect of human life, including food, clothing, medication, antibiotic cures, and the environment throughout the world. Plants are a basic source of food for both human and animal species and agriculture is associated directly with the economic system of different countries as well. Similar to human health, plants

are susceptible to diseases caused by viruses and bacteria [1]. The timely detection of plant diseases is crucial to increase their yield to enhance economic growth and to ensure food security for a growing human population [2]. Therefore, there is a need for an automatic, fast, efficient, and accurate system to detect plant disease and control the disease spread for better crop yield.

Many machine learning and deep learning models have been used for plant leaf detection. They mostly cover 14 diseases in fruits and vegetables e.g. apples and cucumbers

The associate editor coordinating the review of this manuscript and approving it for publication was Jeon Gwanggil.

to a maximum of 14 diseases [3], [4], [5], [6]. AgriCure, a plant disease detector is proposed to detect disease in plants in vegetables and fruits [7]. Farmers, botanists, specialists, and consultants can use the proposed system for their benefit. However, most of the existing work on plant leaf detection is based on training single machine learning or deep learning models for disease detection models. However, in our study, we find that the predictions of a single model are not 100 percent reliable due to the complex nature of the plant data and multiple diseases. Moreover, there is a need to interpret the results of the black box deep learning models to analyze the main factors causing diseases for better mitigation.

This study proposes an innovative approach to address these concerns, by combining Ensemble learning with Explainable AI for accurate detection of multiple plant leaf diseases and analysis of the factors responsible for the diseases. Ensemble learning involves combining multiple models to improve predictive performance. Our approach utilizes a deep learning ensemble, which incorporates four models: VGG16, VGG19, ResNet101 V2, and Inception V3. By leveraging the collective learning of these models through ensemble learning, we aim to achieve an accuracy of 90% and above. An ensemble learning voting technique is employed to aggregate predictions from individual models, ensuring robust and accurate results. In addition to the ensemble approach, we use explainable AI, which provides the capability to explain clear and understandable explanations for predictions. Through the LIME explainable AI technique, the proposed model not only predicts the class of a plant leaf disease but also provides an in-depth explanation and justification for its decision.

The key contributions of the proposed work are as follows.

- We have used an image dataset of 38 diseases of different plants named apple, blueberry, cherry, grape, orange, peach, pepper, potato, raspberry, soy, squash, strawberry, and tomato. The dataset was pre-processed using image processing techniques.
- We have developed an Ensemble Learning framework for plant leaf disease detection using four deep learning models including VGG16, VGG19, ResNet101 V2, and Inception V3, each providing their output independently. The final output is determined using hard and soft voting techniques, which yield more accurate results compared to a single model.
- Explainable AI (XAI) is employed to explain predictions through their features, enhancing the accuracy of the predictions.

The rest of the manuscript is organized as follows: In Section II, we have provided a detailed literature overview, discussing existing work on plant leaf disease detection. In Section III, we explain the proposed ensemble learning and Explainable AI-based methodology. In section IV, the experimental details are presented including model training and parameters optimization. The Ensemble learning results

are discussed along with the application of Explainable AI. The final section presents a conclusion and addresses potential future work directions.

II. LITERATURE REVIEW

In this section, we present a comparative analysis of the existing work on plant leaf disease detection. Past studies have explored various machine learning and deep learning approaches for plant disease identification. For this purpose, researchers have explored different classification methods, morphology features, color and texture attributes [8], [9].

The paper provides an overview of different kinds of diseases in plant leaves and applies machine learning-based classification methods to detect plant diseases such as bacteria, viruses, and fungi [10]. For classification, the authors considered morphology features and attributes of color, intensity, and dimensions of leaves [11].

The stem plant disease detection is handled in the following work [12]. A mobile application has been developed for capturing images of jute plant stems. The images are transferred to the server, and affected portions are segmented using a threshold formula via hue-based segmentation. Features value is matched along with stored sample values and a Support Vector Machine classifier is applied to predict the disease.

The paper mainly focuses on papaya leaves. Random forest classifier is used to diagnose unhealthy plant leaves with an accuracy of nearly 70%. The methodology involves dataset creation, feature extraction, and classifier training [13].

The authors in [14] utilized different machine learning models on a diverse set of plant leaf images, achieving an average accuracy of 99% across 22 plant diseases. However, this approach was limited to traditional machine-learning techniques.

Recently, numerous studies have investigated deep-learning techniques for plant disease detection. In [5] authors, tackled the problem of plant leaf disease detection through the use of image processing [15], [16], [17] strategies and back propagation neural network (BPNN). In [18], the authors applied Otsu thresholding observed by boundary detection and detection set of rules to section inflamed elements inside the leaf. Consequently, they extracted features which include color, texture, morphology, and edges. BPNN was applied for disease detection.

In [3], the authors developed a convolutional neural network (CNN) based model for disease diagnosis. They had efficaciously labeled twelve plant diseases. The data set of three thousand high-resolution colored images was utilized for the research. However, the F1 score is very low due to a high number of false-negative predictions.

In [19] authors presented a CNN-based method to detect rice crop disease, utilizing a 500-pixel image data set of unhealthy and healthy rice plant leaves and their stems. In total, 10 rice crop diseases are handled.

In [4], authors developed a CNN based model for identifying Apple pathological images. Based on different analytics benchmarks, they show the effectiveness of their approach.

In [5], a cucumber leaf disease detection method based on CNN is proposed with 94% accuracy. However, the proposed method only detects a single plant disease.

In [6], the researchers developed a CNN based model to identify 13 plant diseases with 94% accuracy. In [20] authors applied pre-trained CNN models, including YOLO, to identify diseases in different plants. This method yielded an average accuracy of 99.06% across 25 plant diseases, demonstrating the potential of deep learning models in improving disease detection accuracy over traditional machine learning models.

These studies underscore the effectiveness of both machine learning and deep learning for plant disease detection, while also highlighting the advantages of deep learning algorithms in handling more complex datasets. Table 1 presents a comparative analysis of existing machine learning and deep learning-based plant leaf detection models. Notable observations from existing studies include:

Limited number of Plant Leaf Diseases: In the context of plant pathology, researchers have predominantly concentrated their efforts on a subset of 10 to 15 diseases [2], [7]. To the best of our knowledge, limited work has been done on detection of 38 classes of plant diseases comprehensively [1].

Scarcity of Ensemble Learning based Techniques: Notably, a prevalent trend in literature involves utilization of CNN algorithm. Although CNN based deep learning models have proven effective in yielding high accuracy rates, a notable gap emerges in ability of these models to confidently identify a single disease. Often, reliance on maximum accuracy comes at the cost of precision in disease prediction. The observed discrepancy lies in the fact that, despite achieving high accuracy, a definitive diagnosis is only achieved when 2 to 3 models converge on the same outcome [21]. This underscores need for an Ensemble learning based reliable approach for disease detection and highlights a potential area for improvement in the existing work.

Limited Work on Explainable AI: To the best of our knowledge, limited work has been done on applying explainable AI methods to identify important features used by deep learning based plant leaf detection models.

To address these issues, in this research work, we propose a novel methodology that builds an ensemble learning classifier, leveraging power of four deep learning models i.e., VGG16, VGG19, InceptionV3, and ResNet101V2 trained on Plant Village dataset containing 38 classes of plant leaf diseases. This approach aims to significantly improve disease detection accuracy and robustness by combining strengths of multiple models. An essential aspect of our proposed system

is incorporation of explainable AI, ensuring transparency and interpretability of the classification results. By using explainable AI techniques, our system can provide clear and understandable justifications for its disease predictions, enhancing user trust and facilitating the interpretation of results.

III. PROPOSED METHODOLOGY

The proposed methodology is based on integration of an Ensemble learning classifier with Explainable AI for plant leaf disease detection. The following steps are included i.e., data collection, data pre-processing, deep learning models training and testing, Ensemble learning using soft Voting, and results interpretation using LIME. We explain these steps of the proposed methodology in the following subsections.

A. DATASET DESCRIPTION

Plant Village dataset is a comprehensive collection of 54,305 images depicting both unhealthy and healthy plant leaves [22]. Figure 1 shows some images taken from the data set. These images were carefully gathered under controlled conditions to ensure data quality and reliability. The dataset encompasses a wide variety of fruits and vegetables, including apple, blueberry, cherry, grape, orange, peach, pepper, potato, raspberry, soy, squash, strawberry, and tomato, totaling 14 different plants. Within this dataset, there are images representing 17 distinct basic diseases, 4 bacterial diseases, 2 diseases caused by mold (oomycete), and 2 viral diseases. We have selected the Plant Village dataset because it offers a comprehensive set of 38 disease classes, which is not available in any other dataset. The total images for the training of the model are shown in the below table:

B. DATA PRE PROCESSING

We have implemented a series of data pre-processing techniques to improve the quality and variability of the training data. The pre-processing includes multiple steps including re-scaling pixel values to [0, 1] range, applying rotations up to 25 degrees, zooming in and out by up to 50%, shearing with a range of 0.3, and performing horizontal flips. These steps were applied to enhance the classifier's ability to generalize by training it on a wider range of image variations. The figure shows how an original image is transformed through these pre-processing steps. These steps are critical for reducing overfitting.

C. ENSEMBLE LEARNING CLASSIFIER

We have developed an Ensemble learning-based classifier that combines outputs from multiple deep learning models using majority voting to achieve higher disease prediction accuracy. This approach helps to gain an understanding of how different deep learning models perform on new, unseen data and to determine whether the models are overfitting or underfitting [23]. The proposed ensemble

TABLE 1. Comparative analysis of machine learning and deep learning based plant leaf disease detection.

Paper No.	Methodology	Dataset	Accuracy	Limitations	Explainable AI
[3]	CNN model	3,000 images	74% on greyscale, 88% on 14 classes	Used CNN model for 14 plant diseases only	NO
[10]	K-Nearest Neighbor, Radial Basis Function, Probabilistic Neural Networks, Backpropagation Network, Support Vector Machine	Not explicitly mentioned	94% accuracy	traditional ML model used for fungal disease only	NO
[12]	SVM	Stem diseases dataset	87%	Used machine learning models only	NO
[13]	Random forest	160 images of papaya	70.14%	used traditional machine learning model for a single plant disease	NO
[19]	CNN	500 rice plant images	95% accuracy only on ten rice diseases using CNN	Used machine learning and CNN just to compare accuracy. F1 score, Precision, Recall not given	NO
[4]	CNN	Apple pathological images	Effective approach for apple disease identification	Focused on apple diseases and only CNN model used	NO
[5]	CNN	Cucumber leaf images	94%	Only Cucumber plant disease	NO
[6]	CNN	4483 images, 30880 augmented images, 2589 testing images	94% average	Only CNN used for 13 plant diseases	NO
[14]	XGBoost, KNN, SVM, AdaBoost models	2100 plant images	99%, 72%, 95%, and 86%	Only machine learning models used with 22 diseases	NO
[20]	DenseNet201, EfficientNetV2, MobileNetV2, NasNetLarge, ResNet152V2, ResNet50V2, VGG16, VGG19 and Xception	1083 plant images	97.499%, 98%, 99%, 95%, 96%, 97%, 97%, 95%, 96%	CNN pre-trained models used with 25 plant diseases	NO

learning framework encompasses 4 deep learning models, i.e., VGG16, VGG19, Inceptionv3, and Resnet101v2, and soft and hard voting techniques are employed to combine outputs of the trained deep learning models. To ensure reliability of our predictions, we also utilized explainable AI technique, LIME, which helps to interpret the results of Ensemble learning and verify that the predicted disease is accurate by providing insights into the model’s decision-making process.

1) VGG16 MODEL

We have employed VGG16, a deep convolutional neural network for its well-balanced trade-off between depth and computational efficiency, making it suitable for our ensemble learning approach. The VGG16 architecture consists of 16 layers, including 13 convolutional layers and 3 fully connected layers, which are organized sequentially. The activation function used after each convolutional layer is ReLU (Rectified Linear Unit), defined as:

$$f(x) = \max(0, x)$$

This activation function introduces non-linearity, which allows the network to learn more complex patterns.

The network concludes with a softmax activation function in the final layer, which outputs a probability distribution over the target classes. Mathematically, the softmax function is defined as:

$$\sigma(z_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$$

where z_i represents the input to the i -th unit of the final layer, and K is the total number of classes. The input layers use input size (265 X 256 X 3) with stride 1 and padding to preserve the spatial dimensions, followed by max-pooling layers to reduce spatial dimensions and computational load. The network’s depth allows it to learn intricate patterns in images, which is crucial for accurate disease detection in plant leaves. The fully connected layers at end of the network aggregate learned features and produce final classification output. By using VGG16 in combination with other models in our ensemble learning classifier, we aim to leverage its strengths in feature extraction and enhance the overall prediction accuracy of our system. In Figure 3, we present VGG16 model architecture. The pre-processed image dataset is passed onto the input layer. After features extraction, we pass this to flatten layer and then output layer which contains 38 neurons, and softmax activation function is applied.

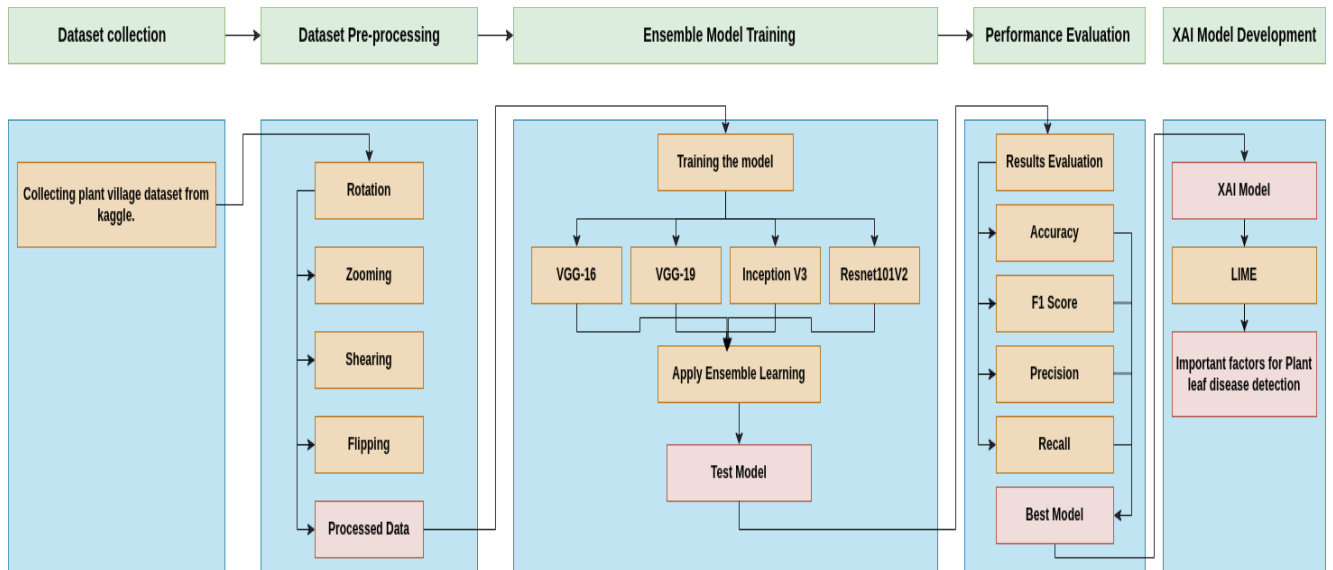


FIGURE 1. Proposed ensemble learning and explainable AI based methodology.

TABLE 2. Plant village image dataset.

Plant	Disease	Images
Apple	Scab	630
Apple	Black Rot	621
Apple	Cedar Rust	275
Apple	Healthy	1645
Blueberry	Healthy	1502
Cherry	Powdery Mildew	1034
Cherry	Healthy	850
Corn	Cercospora leaf spot Gray leaf spot	513
Corn	Common Rust	1192
Corn	Northern Leaf Blight	985
Corn	Healthy	1162
Grape	Black Rot	1180
Grape	Black Measles	1383
Grape	Isariopsis Leaf Spot	1076
Grape	Healthy	423
Orange	Huanglongbing	4507
Peach	Bacterial Spot	1797
Peach	Healthy	360
Pepper	Bell Bacterial Spot	997
Pepper	Bell Healthy	1478
Potato	Early Blight	1000
Potato	Late Blight	1000
Potato	Healthy	152
Raspberry	Healthy	371
Soybean	Healthy	5090
Squash	Powdery Mildew	1835
Strawberry	Leaf Scorch	909
Strawberry	Healthy	456
Tomato	Bacterial Spot	2130
Tomato	Early Blight	1000

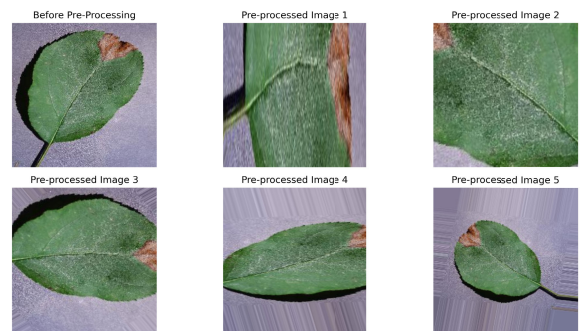


FIGURE 2. Data preprocessing applied on a plant leaf image.

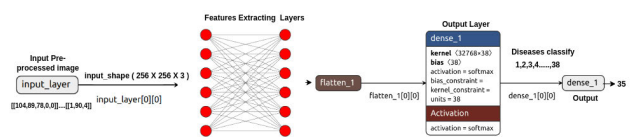


FIGURE 3. VGG16 model architecture.

2) VGG19 MODEL

In our ensemble learning classifier, the second model used is VGG19 which is a powerful deep convolutional neural network known for its ability to capture complex features in different image classification tasks. It has 19 feature layers, including 16 convolutional layers and 3 fully connected

layers. VGG19 is particularly useful for tasks requiring high precision and fine-grained details due to its increased depth compared to VGG16.

The architecture of VGG19 follows same principles as VGG16, using small receptive fields (3 × 3) with stride 1 and padding to preserve spatial dimensions. Max-pooling layers are applied after every two or four convolutional layers to reduce the dimensionality. The increased depth allows VGG19 to learn more complex features, which is beneficial for tasks such as disease detection in plant leaves. By integrating VGG19 into our ensemble learning, we aim to enhance model’s ability to generalize across different datasets and improve classification accuracy. Mathematically,

the ReLU activation function used in each convolutional layer is defined as:

$$f(x) = \max(0, x)$$

The final layer of VGG19 network uses softmax activation function to produce a probability distribution over the classes, defined as:

$$\sigma(z_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$$

where z_i is the input to the i -th unit of the final layer, and K is the total number of classes.

3) INCEPTIONV3 MODEL

InceptionV3 is a deep convolutional neural network architecture designed for image classification tasks. Key features of InceptionV3 include use of the ‘‘Inception module’’ with various filter sizes, batch normalization, and a factorization approach to reduce computational complexity. The architecture of InceptionV3 includes a series of convolutional layers, Inception modules, and pooling operations. Characterized by its innovative ‘‘inception modules,’’ InceptionV3 utilizes 1×1 , 3×3 , and 5×5 convolutional layers to extract features from input images. For example, output of a convolution operation can be mathematically represented as:

$$O = \text{Conv}(I, K) + B$$

where: - O is the output feature map, - I is the input image, - K is the convolution kernel, - B is the bias term.

Inception Modules: These modules combine multiple convolutional filters to capture features at different scales. The output O of a single inception module can be represented as:

$$O = \text{Concat}(\text{Conv}_{1 \times 1}(X), \text{Conv}_{3 \times 3}(X), \text{Conv}_{5 \times 5}(X), \text{MaxPool}(X))$$

where: - Concat indicates concatenation along the depth dimension, - $\text{Conv}_{1 \times 1}$, $\text{Conv}_{3 \times 3}$, and $\text{Conv}_{5 \times 5}$ are the respective convolution operations applied to the input X , - MaxPool is the max pooling operation.

Factorized Convolutions: To improve efficiency, larger convolution operations are factorized. **Global Average Pooling:** This layer replaces traditional fully connected layers, reducing overfitting and computational load:

$$O = \text{GlobalAvgPool}(X)$$

Softmax Activation: The final output is obtained using a softmax function for multi-class classification:

$$\text{Output} = \text{Softmax}(W \cdot O + b)$$

4) RESNET101V2 MODEL

The primary reason for choosing ResNet-101V2 in our proposed Ensemble learning classifier is its sophisticated architecture, which builds upon ResNet (Residual Networks)

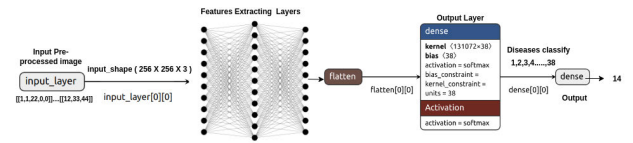


FIGURE 4. Resnet101V2 model architecture.

framework by incorporating several key enhancements. These enhancements include use of batch normalization and improved residual connections, which contribute to better training performance and reduced overfitting. ResNet-101V2 model depth, with 101 layers, enables it to capture complex features and hierarchical representations from input images, making it highly suitable for tasks requiring detailed feature extraction and fine-grained classification.

It utilizes ‘‘residual connections’’ to mitigate vanishing gradient problems in very deep networks, allowing for more effective training. ResNet101V2 features 101 layers, including an initial 7×7 convolutional layer, followed by four stages of residual blocks. The residual blocks are composed of three layers: a 1×1 convolution, a 3×3 convolution, and another 1×1 convolution, with a number of filters doubling across stages (from 64 in Stage 1 to 512 in Stage 4). The network also incorporates batch normalization and ReLU activation after each convolutional layer. The final layers include a global average pooling layer and a fully connected layer with softmax activation for classification. The operation within each residual block can be mathematically described as:

$$\text{Output} = \text{ReLU}(W_2 \cdot \text{ReLU}(W_1 \cdot \text{Input} + b_1) + b_2 + \text{Input})$$

where: - W_1 and W_2 are the weights of the convolutional layers, - b_1 and b_2 are the biases, - ReLU represents the activation function.

In Figure 4, we present architecture of our model which takes images having (256 X 256 X 3) size at input layer. The output layer contains 38 neurons depicting the 38 disease classes.

IV. EXPERIMENTATION

In our proposed Ensemble learning classifier, we have trained four deep learning models including VGG16, VGG19, InceptionV3, and Resnet101V2 trained on the preprocessed Plant Village dataset. The dataset is divided into training and test subsets. The number of images in the training dataset is 43,444 whereas, the test dataset contains 10861 images. The outputs of the trained models are combined using hard and soft voting techniques for plant leaf disease detection.

A. HARDWARE

The experiments were conducted on a Dell laptop with following specifications: Processor: Intel(R) Core(TM) i5-4210U CPU @ 1.70GHz 2.40 GHz, RAM: 4.00 GB, System type: 64-bit operating system, x64-based processor.

The hardware effectively catered to the computational needs of the experiments.

B. SOFTWARE

Open-source libraries and frameworks were used to implement and train the deep learning models. Python extensive library ecosystem, including NumPy, Pandas, and RandomOverSampler, was utilized to streamline the process. Scikit-learn offered a toolkit for machine learning tasks, while TensorFlow provided functionalities for deep learning model building and training. These libraries streamlined data pre-processing, model training, and evaluation.

C. PARAMETERS TUNING

We conducted parameter tuning for each deep learning model to optimize its performance. This involved adjusting model-specific parameters, i.e., the number of hidden layers, learning rate, and regularization parameters. Table 3 shows the values of the parameters.

TABLE 3. Parameters values for deep learning models.

Parameters	Values
Layers	Dense
Activation Function for all Layers	ReLU
Activation Function for Output Layer	Sigmoid
Epochs	50
Optimizer	Adam
Batch Size	25
Learning Rate	0.001

D. VGG16 MODEL TRAINING

In our proposed ensemble learning classifier, first deep learning model used is VGG16 model. The target image size is set to (256,256), with a batch size of 32. For model training, We have used softmax activation function, and categorical cross-entropy loss function is optimized using Adam optimizer. The training was configured to stop if accuracy did not improve over five consecutive epochs. The model has been trained over 20 epochs, but early stopping occurred at the 8th epoch, achieving an accuracy of 91.9%. Figures 5 and 6 illustrate training and validation loss and accuracy graphs.

E. VGG19 MODEL TRAINING

The second model in our Ensemble learning classifier is CNN based VGG19. The model is trained with a target image size of (256,256) and a batch size of 32. Employing softmax activation function and categorical cross-entropy loss, we optimized VGG19 model using Adam optimizer. A stopping criterion was set: training would halt if accuracy failed to improve after 2 consecutive epochs. During 50 epochs, model training was completed during 6th epoch, attaining 88.2% accuracy with minimum loss as shown in Figures 7 and 8.

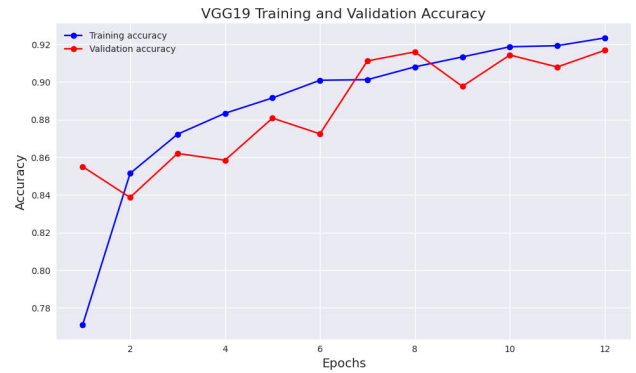


FIGURE 5. Training and validation accuracy of VGG16 model.

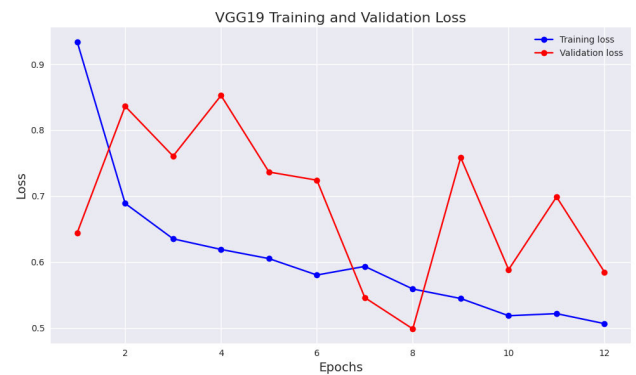


FIGURE 6. Training and validation loss of VGG16 model.

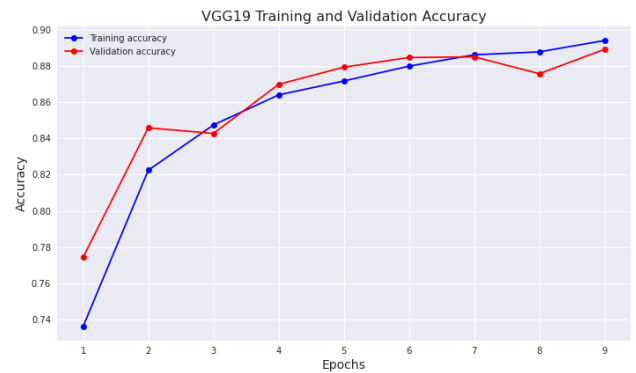


FIGURE 7. Training and validation accuracy of VGG19 model.

F. INCEPTIONV3 MODEL TRAINING

In our proposed Ensemble learning classifier, InceptionV3 model is trained with a target image size of (256,256) and a batch size of 32. The model refinement is performed by using softmax activation function and categorical cross-entropy loss, with performance optimization performed by Adam optimizer. A predefined stopping criterion mandated cessation of training if accuracy failed to improve over two consecutive epochs. Across a span of 50 epochs, model

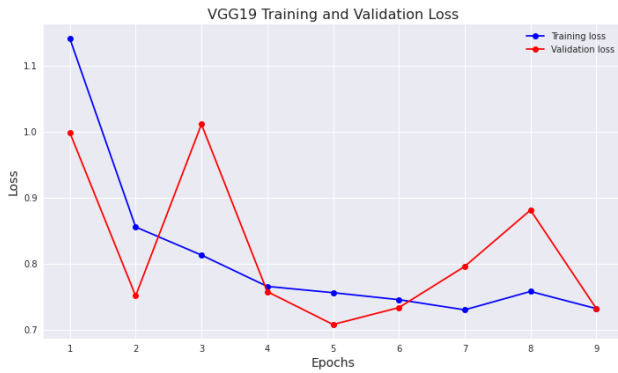


FIGURE 8. Training and validation loss of VGG19 model.

training culminated by 6th epoch, achieving an accuracy level of 88% as presented in Figures 10 and 11.

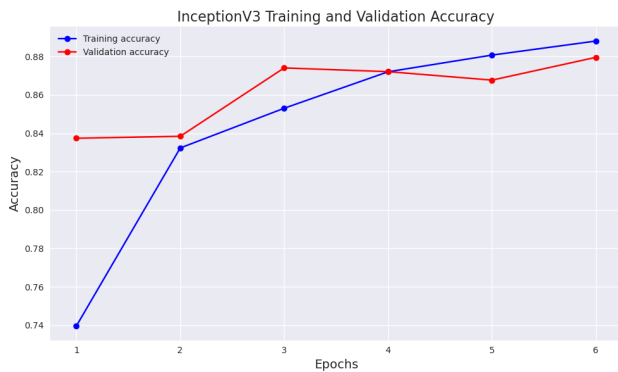


FIGURE 9. Training and validation accuracy of InceptionV3 model.



FIGURE 10. Training and validation loss of InceptionV3 model.

G. RESNET101V2 MODEL TRAINING

The fourth deep learning model in our proposed Ensemble Learning classifier is ResNet101V2. The model is trained using preprocessed images from the Plant Village dataset. Softmax function and categorical cross-entropy are used for model training. We have defined a stopping criteria. After

50 epochs, model performance improved and training was stopped at a detection accuracy of 93.6% accuracy with lower validation loss as shown in Figures 13 and 14 below.

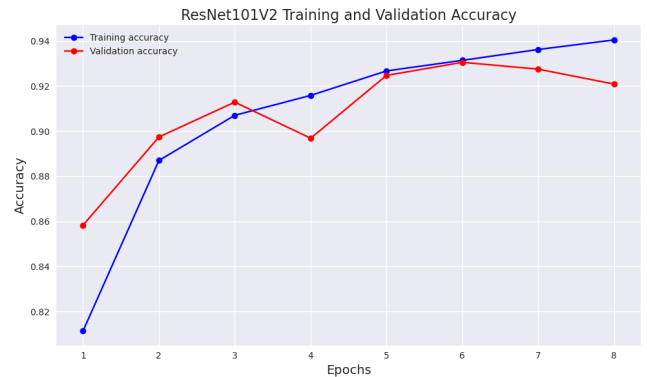


FIGURE 11. Training, validation accuracy of ResNet101V2 model.

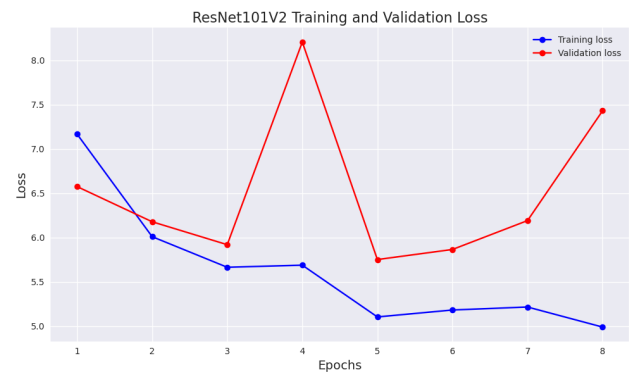


FIGURE 12. Training and validation loss of ResNet101V2 model.

The precision, recall, and F1 score values have been computed and presented in Table 4.

TABLE 4. Performance comparison of four deep learning models.

No.	Model	Accuracy	Precision	Recall	F1 Score
1	VGG 16	0.919	0.89	0.72	0.70
2	VGG 19	0.882	0.74	0.75	0.69
3	Inceptionv3	0.88	0.89	0.86	0.84
4	Resnet101v2	0.936	0.90	0.89	0.86

H. VOTING BASED ENSEMBLE LEARNING

Voting in the context of machine learning is a method to combine predictions from multiple models or classifiers to make a final decision or prediction. There are two common types of voting techniques:

Hard Voting: In hard voting, each model or classifier makes a prediction, and final decision is based on the majority vote.

In our proposed Ensemble learning classifier, we have trained four deep learning models for plant leaf disease

prediction. The outputs of the four trained models are aggregated to determine final output of ensemble learning classifier. In our case, consider the following example:

Vgg16 predicts class 0 (Apple_scab). Vgg19 predicts class 1 (Apple_Black_rot). InceptionV3 predicts class 0 (Apple_scab). Resnet101V2 predicts class 3 (Apple_healthy).

In majority voting, all votes are counted and class with a maximum number of votes is selected as final prediction of ensemble learning classifier. In this case, class 0 (Apple_scab) has highest number of votes, so this class would be chosen.

Calculating the votes: Votes for Class 0 (Apple_scab) = 2
 Votes for Class 1 (Apple_Black_rot) = 1
 Votes for Class 3 (Apple_healthy) = 1

The final prediction using hard voting would be class with the highest number of votes, which is Class 0 (Apple_scab).

This approach simplifies the decision-making process by following the majority's opinion, leading to a choice that reflects the most common prediction, as explained in Figure 13.

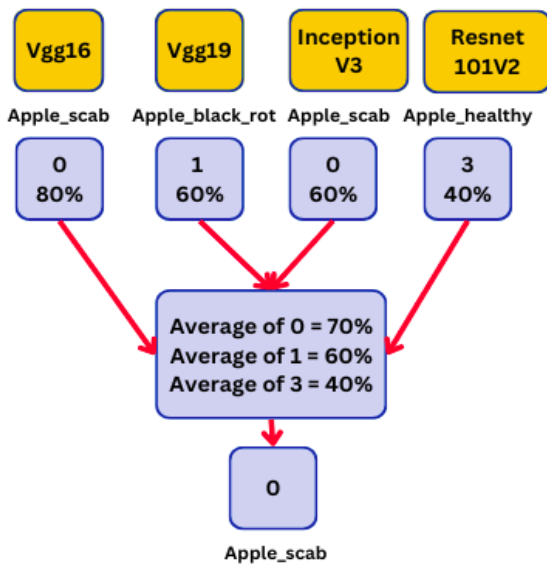


FIGURE 13. Hard voting approach.

Soft Voting: In soft voting, not only the majority votes are considered, but also confidence scores from each model's predictions are taken into account. The confidence scores for each class are summed up, and the class with the highest total confidence score is chosen.

In our case,:

- Vgg16 is confident in its prediction of class 0 (Apple_scab), with a confidence score denoted as c_1 .
- Vgg19 is confident in its prediction of class 1 (Apple_Black_rot), with a confidence score denoted as c_2 .
- InceptionV3 is confident in its prediction of class 0 (Apple_scab), with a confidence score denoted as c_3 .
- Resnet101V2 is confident

in its prediction of class 3 (Apple_healthy), with a confidence score denoted as c_4 .

The total confidence score for class 0 (Apple_scab) is calculated by summing up confidence scores from Vgg16, Vgg19, InceptionV3, and Resnet101V2:

$$\text{Total Confidence for Class 0} = c_1 + c_2 + c_3 + c_4$$

Calculating specific confidence values from your given example: - c_1 = confidence from Vgg16 - c_2 = confidence from Vgg19 - c_3 = confidence from InceptionV3 - c_4 = confidence from Resnet101V2

Use these values of confidence in the final equation:

$$\text{Total Confidence for Class 0} = c_1 + c_2 + c_3 + c_4$$

Similarly, we can also evaluate total confidence values for other classes. The class with the highest confidence value is selected as final prediction, as explained in Figure 14.

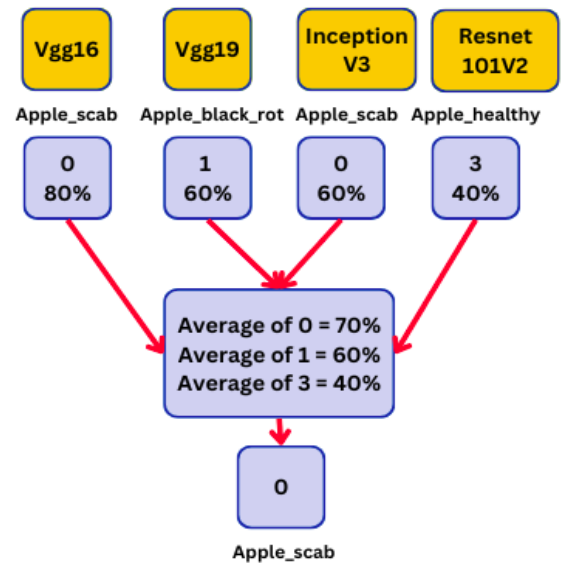


FIGURE 14. Soft voting approach.

TABLE 5. Ensemble learning performance using hard voting and soft voting.

No.	Technique	Accuracy	Precision	Recall	F1 Score
1	Hard Voting	0.87	0.86	0.87	0.86
2	Soft Voting	0.923	0.90	0.90	0.89

I. RESULTS INTERPRETATION USING EXPLAINABLE AI

In this research work, we have employed explainable AI to interpret prediction results of Ensemble learning using both hard and soft voting techniques. In contrast to machine learning methods in which explicit features are extracted from dataset and fed to the model, deep learning models automatically extract features from training data making it difficult to understand how different feature values

contribute to the final prediction. To solve this problem, we have utilized LIME (Locally Interpretable Model-Free Description) method which is one of the most widely used algorithms for interpreting the results of black box deep learning models. LIME generates explanations by taking input features and observing changes in model predictions, which allows us to understand which features of the leaf image are most influential in determining the predicted disease. For example, if the model predicts a leaf to have a particular disease like Apple_scab, LIME can highlight specific regions or characteristics of the leaf image that contributed most significantly to that prediction. This not only understand the model’s decision-making process but also helps users interpret and trust the model’s outputs more effectively. We have categorized the explanations given by LIME into the following two categories.

- Explain Actual Prediction
- Interpret Actual Predictions using Segmentation Method
- Interpret Wrong Prediction

1) EXPLAIN ACTUAL PREDICTION

We generated distinct visual representations that highlighted the pixels that have positively and negatively influenced the predictions. Initially, a random image sample was chosen from test dataset. Subsequently, both actual and predicted labels were displayed. **Pixels Enhancing True Predictions:** we generate an image and mask that shows pixels that positively impact the prediction. Figure 15 displays original image, predicted image, the masked image, and their combined view. This aids in comprehending the features used for forecasting.

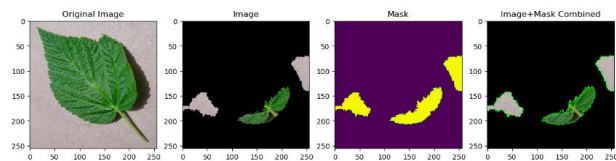


FIGURE 15. Pixels enhancing true predictions.

Pixels Enhancing Wrong Predictions: To explain wrong predictions, we have generated a visualization displaying those pixels that have a negative impact on the prediction outcome shown in the following figure. From the visualization, it is evident that certain middle pixels negatively influence prediction, suggesting they might be beneficial for another category. Figure 16 below illustrates influence of these negative pixels on predictions.

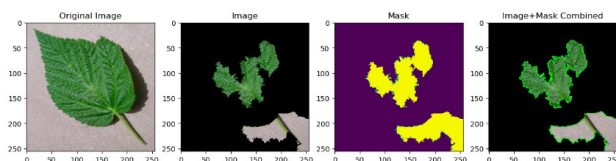


FIGURE 16. Pixels enhancing wrong predictions.

2) INTERPRET ACTUAL PREDICTIONS USING SEGMENTATION METHOD

We present Felzenszwalb segmentation method, which is accessible from scikit-image Python library. The method divides an image into regions considering color likeness and compactness, proving beneficial for tasks such as object detection and image interpretation. **Pixels Contributing Positively to Prediction:** we have generated a visualization that highlights pixels enhancing the prediction accuracy, utilizing an Explanation object generated through the Felzenszwalb segmentation technique. Figure 17 below demonstrates effectiveness of the Felzenszwalb segmentation method in elucidating predictions.

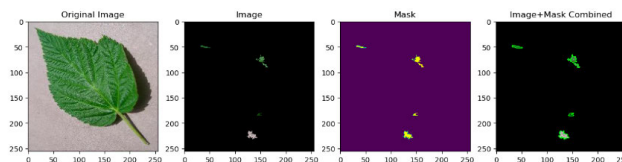


FIGURE 17. Actual predictions using segmentation method.

Pixels Enhancing Wrong to Prediction: In this section, we have created a visualization that identifies negative pixels from prediction using Felzenszwalb segmentation technique. In Figure 18, we illustrate the impact of these negative pixels on predictions.

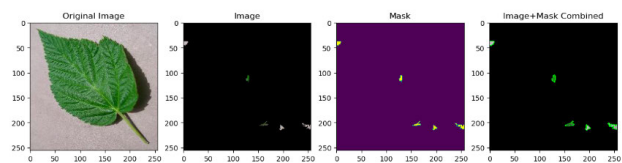


FIGURE 18. Negative pixels enhancing wrong prediction.

3) INTERPRET WRONG PREDICTION

To give interpretations of wrong predictions, we have generated sample wrong predictions produced by our model. This explanation helps us determine which pixels are influencing incorrect prediction category. Figure 19, we showcase instances where actual image differs, leading to an incorrect prediction, along with our explanation for this prediction.

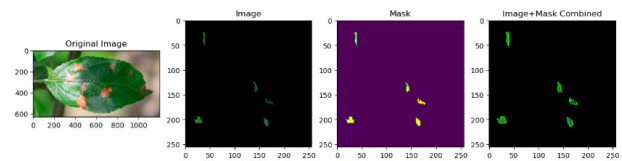


FIGURE 19. Explanation of wrong prediction.

V. PERFORMANCE COMPARISON AND DISCUSSION

In this section, we analyze results of our Ensemble learning classifier to gain valuable insights into effectiveness of

TABLE 6. Performance comparison of ensemble learning with related work.

Study	Model Used	No. of images	No. of classes	Accuracy	XAI
[19]	CNN	500	10	95%	No
[14]	KNN, Random Forest	2,100	2	97%, 98%, 99%	No
[20]	MobileNet, YOLO	1,083	25	99.06%	No
[3]	CNN	3,000	14	88%	No
Our Work	Ensemble Learning	54,305	38	92.3%	yes

the proposed methodology for plant leaf disease detection. We compare our prediction results obtained with related work on plant leaf disease detection. In comparison with work done in [24], we achieved 90% to 91% accuracy with individual deep learning models. It is evident from comparative analysis that previous studies, using Plant-Village dataset [3], [4], [5], [6], typically focused on 14 to 27 diseases. In contrast, our proposed ensemble learning classifier targets 38 diseases with greater than 92% accuracy. Moreover, most of the existing work has not employed explainable AI to interpret prediction results.

The utilization of ensemble voting technique significantly enhances accuracy of our plant disease detection system. Specifically, our soft voting method achieved highest performance, underscoring effectiveness of combining multiple models to improve overall accuracy and robustness. Therefore, the proposed method has the potential to be applicable in real-life situations to accurately detect plant diseases.

VI. CONCLUSION AND FUTURE WORK

In this study, we have developed an Ensemble learning classifier based on VGG16, VGG19, ResNet101V2, and InceptionV3 deep learning models for plant disease detection and employed LIME for interpretation of the results. Compared to previous studies, which often focused on a limited number of plant diseases and primarily used single model like CNN, our approach covers 38 classes of plant diseases. The proposed ensemble learning methodology, particularly soft voting, has proven to be highly effective, achieving an overall accuracy of over 93%. Furthermore, integration of XAI technique LIME, adds substantial value to our system. By providing clear and logical explanations for model predictions, XAI enhances user confidence and facilitates interpretation of results, addressing a critical gap in prior research where model interpretability was often overlooked. This methodology has the potential to make a meaningful impact on precision agriculture and plant disease management, contributing to economic development and environmental sustainability. Future work will focus on extending the current model, refining its capabilities, and exploring new AI techniques to further enhance system's effectiveness in diagnosing plant diseases. Furthermore, in future bagging and boosting techniques can be employed to further improve accuracy. We can also include multiple XAI techniques to improve prediction understanding with a user-friendly interface. Another idea is to use advanced

generative AI techniques to explain disease predictions in different languages.

REFERENCES

- [1] M. Chohan, A. Khan, R. Chohan, S. H. Katpar, and M. S. Mahar, "Plant disease detection using deep learning," *Int. J. Recent Technol. Eng.*, vol. 9, no. 1, pp. 909–914, May 2020.
- [2] H. Park, J.-S. Eun, and S.-H. Kim, "Image-based disease diagnosing and predicting of the crops through the deep learning mechanism," in *Proc. Int. Conf. Inf. Commun. Technol. Converg. (ICTC)*, Oct. 2017, pp. 129–131.
- [3] G. Shrestha, Deepshikha, M. Das, and N. Dey, "Plant disease detection using CNN," in *Proc. IEEE Appl. Signal Process. Conf. (ASPCON)*, Oct. 2020, pp. 109–113.
- [4] C. W. H. Tan and W. X. Zhao, "CNN intelligent early warning for apple skin lesion image acquired by infrared video sensors," *High Technol. Lett.*, vol. 10, no. 1, pp. 67–74, 2016.
- [5] Y. Kawasaki, H. Uga, S. Kagiwada, and H. Iyatomi, "Basic study of automated diagnosis of viral plant diseases using convolutional neural networks," in *Proc. Int. Symp. Vis. Comput.* Cham, Switzerland: Springer, 2015, pp. 638–645.
- [6] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, "Deep neural networks based recognition of plant diseases by leaf image classification," *Comput. Intell. Neurosci.*, vol. 2016, pp. 1–11, Mar. 2016.
- [7] M. E. H. Chowdhury, T. Rahman, A. Khandakar, M. A. Ayari, A. U. Khan, M. S. Khan, N. Al-Emadi, M. B. I. Reaz, M. T. Islam, and S. H. M. Ali, "Automatic and reliable leaf disease detection using deep learning techniques," *AgriEngineering*, vol. 3, no. 2, pp. 294–312, May 2021.
- [8] B. Liu, Y. Zhang, D. He, and Y. Li, "Identification of apple leaf diseases based on deep convolutional neural networks," *Symmetry*, vol. 10, no. 1, p. 11, Dec. 2017.
- [9] M. Brahimi, K. Boukhalfa, and A. Moussaoui, "Deep learning for tomato diseases: Classification and symptoms visualization," *Appl. Artif. Intell.*, vol. 31, no. 4, pp. 299–315, Apr. 2017.
- [10] M. K. R. Gavhale and P. U. Gawande, "An overview of the research on plant leaves disease detection using image processing techniques," *IOSR J. Comput. Eng.*, vol. 16, no. 1, pp. 10–16, 2014.
- [11] L. S. P. Annabel, T. Annapoorani, and P. Deepalakshmi, "Machine learning for plant leaf disease detection and classification—A review," in *Proc. Int. Conf. Commun. Signal Process. (ICCSP)*, Apr. 2019, pp. 0538–0542.
- [12] Z. N. Reza, F. Nuzhat, N. A. Mahsa, and Md. H. Ali, "Detecting jute plant disease using image processing and machine learning," in *Proc. 3rd Int. Conf. Electr. Eng. Inf. Commun. Technol. (ICEEICT)*, Sep. 2016, pp. 1–6.
- [13] S. Ramesh, R. Hebbar, M. Niveditha, R. Pooja, N. Shashank, and P. V. Vinod, "Plant disease detection using machine learning," in *Proc. Int. Conf. Design Innov. 3Cs Compute Communicate Control (ICDI3C)*, Apr. 2018, pp. 41–45.
- [14] B. Dey, J. Ferdous, and R. Ahmed, "Machine learning based recommendation of agricultural and horticultural crop farming in India under the regime of NPK, soil pH and three climatic variables," *Heliyon*, vol. 10, no. 3, Feb. 2024, Art. no. e25112.
- [15] A. N. Rathod, B. Tanawal, and V. Shah, "Image processing techniques for detection of leaf disease," *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 3, no. 11, pp. 397–399, 2013.
- [16] L. C. Ngugi, M. Abelhahab, and M. Abo-Zahhad, "Recent advances in image processing techniques for automated leaf pest and disease recognition—A review," *Inf. Process. Agricult.*, vol. 8, no. 1, pp. 27–51, Mar. 2021.
- [17] Y. M. Oo and N. C. Htun, "Plant leaf disease detection and classification using image processing," *Int. J. Res. Eng.*, vol. 5, no. 9, pp. 516–523, Nov. 2018.

[18] N. Karthikeyan, M. Anjana, S. Anusha, R. Divya, and A. Vinod, "Leaf disease detection using image processing," *Int. J. Innov. Sci. Eng. Technol.*, vol. 7, pp. 60–65, 2020.

[19] Y. Lu, S. Yi, N. Zeng, Y. Liu, and Y. Zhang, "Identification of Rice diseases using deep convolutional neural networks," *Neurocomputing*, vol. 267, pp. 378–384, Dec. 2017.

[20] B. Dey, R. Ahmed, J. Ferdous, M. M. U. Haque, R. Khatun, F. E. Hasan, and S. N. Uddin, "Automated plant species identification from the stomata images using deep neural network: A study of selected mangrove and freshwater swamp forest tree species of Bangladesh," *Ecol. Informat.*, vol. 75, Jul. 2023, Art. no. 102128. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1574954123001577>

[21] M. A. Ganaie, M. Hu, A. Malik, M. Tanveer, and P. Suganthan, "Ensemble deep learning: A review," *Eng. Appl. Artif. Intell.*, vol. 115, Sep. 2022, p. 105151.

[22] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Frontiers Plant Sci.*, vol. 7, p. 1419, Sep. 2016.

[23] Y. Djenouri, A. Belhadi, A. Yazidi, G. Srivastava, and J. C. Lin, "Artificial intelligence of medical things for disease detection using ensemble deep learning and attention mechanism," *Expert Syst.*, vol. 41, no. 6, Jun. 2024, Art. no. e13093.

[24] B. Dey, M. M. U. Haque, R. Khatun, and R. Ahmed, "Comparative performance of four CNN-based deep learning variants in detecting Hispa pest, two fungal diseases, and NPK deficiency symptoms of Rice (*Oryza sativa*)," *Comput. Electron. Agricult.*, vol. 202, Nov. 2022, Art. no. 107340. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0168169922006494>



BEENISH AYESHA AKRAM received the Ph.D. degree in computer science from the University of Engineering and Technology (UET) at Lahore, Lahore, Pakistan, in 2019. She was a Software Design Engineer with MicroTech Industries (Pvt.). She is currently an Assistant Professor with the Department of Computer Engineering, UET Lahore. She has published in research areas of machine learning, ambient assisted living, WSNs, and indoor localization. She serves as the Section Editor for UET research journal.



FENG DONG received the Ph.D. degree from Shanghai Maritime University, in 2023. He is currently a Lecturer with the School of Information Engineering, Shaoyang University. His research interests include deep learning, computer vision, and medical image processing.



AMMAR OAD received the Ph.D. degree from the Department of Computer Science, Central South University, in 2018. He is currently an Associate Professor with the School of Information Engineering, Shaoyang University, Shaoyang, Hunan, China. His research interests include computer graphics, medical image processing, machine learning, deep learning, cybersecurity, and the IoT.



SYED SHOAB ABBAS received the bachelor's degree from the University of Engineering and Technology (UET) at Lahore, Lahore, where he is currently pursuing the master's degree in computer science. He has been a Senior Software Engineer for past two years, involved in different machine learning and research projects. His research interests include deep learning, computer vision, image processing, medical image processing, and plant image processing.



AMNA ZAFAR received the Ph.D. degree in computer science from UET Lahore, in 2019. She is currently an Assistant Professor with the Department of Computer Science, UET Lahore. She has published numerous research articles in international reputed journals. Her research interests include the wireless IoT, E-health, artificial intelligence, and machine learning.



MIR SAJJAD HUSSAIN TALPUR (Member, IEEE) received the Ph.D. degree in computer application technology from Central South University, China, in 2015. He is currently with the Information Technology Centre, Sindh Agriculture University, Tando Jam, Pakistan, as a Professor and the HoD. His current research interests include the IoT, green AI, blockchain, and cyber security.



MUEEN UDDIN (Senior Member, IEEE) received the Ph.D. degree from Universiti Teknologi Malaysia (UTM), in 2013. He is currently an Associate Professor of data and cybersecurity with the University of Doha for Science and Technology, Qatar. He has published more than 130 international journals and conference papers. His research interests include blockchain, cybersecurity, the IoT security, and network and cloud security.

...