RESEARCH ARTICLE

Robust Detection of Maize Foliage Fungal Diseases using Tree-Based Ensemble Methods

Prabhat Chandra Shrivastava[1], Mohammad Saheeb Masood[1], Aman Kumar[1], Ramesh Kumar Bhukya[2,*], Harmanjit Kaur[3], Rupesh Kumar[4] and Derrick Mario Denis[5]

[1] Department of Electronics and Communication, University of Allahabad, Prayagraj, Uttar Pradesh 211002, India
[2] Department of Electronics and Communication Engineering, Indian Institute of Information Technology, Allahabad, Prayagraj, Uttar Pradesh 211015, India
[3] Department of Botany, University of Allahabad, Prayagraj, Uttar Pradesh 211002, India
[4] Yamaha Motor Solution Private Limited, Faridabad, Haryana 121003, India
[5] Department of Irrigation and Drainage Engineering, Vaugh Institute of Agricultural Engineering and Technology, Sam Higginbottom University of Agriculture, Technology and Sciences, Prayagraj, Uttar Pradesh 211007, India

Abstract

Maize productivity in India, a major global producer, is severely threatened by leaf diseases. Accurate identification of Common Rust (CR), Northern Corn Leaf Blight (NCLB), and Gray Leaf Spot (GLS) remains challenging with traditional methods. This study evaluated traditional and ensemble-based classifiers for classifying these diseases alongside healthy (HL) leaves. Using accuracy, precision, recall, and F1-score, we assessed k-NN, DT, RF, ETs, AdaBoost, SGD, GB, XGBoost, LightGBM, and a Stacking model on a four-class dataset. Ensemble methods demonstrated clear superiority. The Stacking model achieved the highest accuracy (98.50%), followed by LightGBM (98.46%) and XGBoost (98.01%). Among conventional models, ETs (97.38%) and RF (96.93%) outperformed others. While HL was consistently identified, GLS proved most challenging, especially for non-ensemble methods. The results underscore the robustness and superior generalization capability of tree-based ensemble methods for imbalanced multi-class disease classification.

Keywords: maize leaf disease detection, image-based disease classification, plant pathology informatics, agricultural data analytics, artificial intelligence in agriculture, precision agriculture, machine learning, ensemble learning.

1 Introduction

India is the second-largest agricultural producer in the world. About 60-70% of India's rural population is engaged in agriculture (soil cultivation, crop production, livestock farming), which accounts for roughly around 15-17% of the country's *Gross Domestic Product* (GDP). India relies heavily on the agricultural

Submitted: 23 October 2025
Accepted: 23 November 2025
Published: 18 December 2025

Vol. 1, **No.** 2, 2025.
10.62762/DIA.2025.672386

***Corresponding author:**
✉ Ramesh Kumar Bhukya
rkbhukya@iiita.ac.in

Citation
Shrivastava, P. C., Masood, M. S., Kumar, A., Bhukya, R. K., Kaur, H., Kumar, R., & Denis, D. M. (2025). Robust Detection of Maize Foliage Fungal Diseases using Tree-Based Ensemble Methods. *Digital Intelligence in Agriculture*, 1(2), 79–95.

sector for food, employment, and raw materials for industries. Among all the crops grown, maize (*Zea mays L.*), commonly known as corn, is the third most important food grain crop after rice and wheat in India [66]. Maize is important due to its versatility and high demand [1]. In India and many other developing nations, corn is not just a staple food for people but also serves as essential feed for livestock. Plant diseases are a major threat to agriculture around the world as they can result in reduced harvests, increased costs of production, and even cause food insecurity [65]. Plant diseases are caused by several factors, including living organisms such as fungi, bacteria, and viruses, as well as abiotic environmental conditions such as temperature variation, high humidity, drought, and poor soil conditions [3]. Amongst biotic factors, pathogenic fungi are the most prevalent and destructive cause of crop diseases, including corn [4, 5]. The growth of the maize plant can be impacted by several leaf spot diseases, like *Northern Corn Leaf Blight* (NCLB), *Gray Leaf Spot* (GLS), and *Common Rust* (CR), which lower its yield and grain quality, ultimately posing a serious risk to farmer's livelihood and country's overall food security [2, 6, 67]. NCBL, also called *turcicum leaf blight* (TLB), is caused by a fungal pathogen. *Setosphaeria turcica*, also known as *Exserohilum turcicum* (previously classified as Helminthosporium turcicum), leads to the production of atypically elongated or cigar-shaped lesions that disrupt photosynthesis dependent on chloroplast structure. GLS is caused by two fungal species, *Cercospora zeae-maydis* and *Cercospora Zeina*, which results in rectangular-shaped spots that are gray to tan, disrupting photosynthesis [7]. The infectious agent that causes CR is *Puccinia sorghi*, which severely constraints the plant growth, leading to lower corn yields. It is important to recognize the above-mentioned fungal diseases at an early stage and prevent them from spreading and increasing yield loss [68]. If fungal diseases are timely identified, it provides an accurate opportunity for their management or control, including strategies such as fungicide application, use of resistant cultivars, and crop rotation [8]. However, traditional means of responding to visible plant disease symptoms involve manual field visits to collect samples of infected plant species by trained pathologists and agricultural specialists.

While traditional methods are effective, they require a significant amount of time and labor, making them impractical in practice, especially for small producers located in remote areas with limited resources [9]. Reliance on plants, organisms, or other human tools also contributes to variability in diagnosis, leading to inconsistent diagnoses and a lack of scalability in monitoring efforts. Considering these limitations, there is a huge demand these days for developing automated, accurate, and scalable *plant disease detection* (PDD) systems. Advances in *artificial intelligence* (AI), *image processing* (IP), and *machine learning* (ML) have created new opportunities in this area [8, 70]. Computer models can process and analyze images of a plant leaves, allowing for fast and accurate disease detection and classification. Automated systems can not only reduce dependence on expert intervention, and can even provide real-time monitoring via smartphones or IoT-based devices [10]. These innovations could transform the future of plant disease management, making it easier, cheaper, more efficient, and more accessible to both large-scale agricultural operations and smallholder farmers who need to protect their crops and incomes [11].

The rise of AI, particularly in the fields of ML and *deep learning* (DL), has opened new avenues for solving complex agricultural problems such as plant disease detection. These technologies have revolutionized the way complex data-driven problems are approached, offering automation, efficiency, and high accuracy in decision-making processes [12]. In the context of agriculture, and more specifically PDD, AI is believed to provide scalable and cost-effective solutions that could significantly benefit farmers by enabling early disease diagnosis and timely treatment [13]. The advent of DL, particularly *convolutional neural networks* (CNNs), has transformed PDD. CNNs, which are frequently employed in target detection, exhibit great promise for the identification of crop diseases [14–17]. CNNs are useful for tasks like object detection [18, 19], segmentation [20], and recognition [21] because they can accurately detect diseases without the need for manually created features by training on large image datasets. For instance, Lin et al. [22] created a three-channel CNN that, using colour differences, was able to identify vegetable leaf diseases with 87.15% accuracy. Using data augmentation to lessen overfitting [23, 24] developed a *deep* CNN (DCNN) that achieved high accuracy on more than 14,000 images in order to detect four cucumber diseases. On similar lines, Sladojevic et al. [25] classified plant diseases from leaf photos with 91% accuracy. Although these studies demonstrate the efficiency of CNNs, the majority of the datasets

were collected in controlled laboratory settings with unnatural backgrounds, which differ from actual agricultural field settings. Furthermore, the small lesion size of early-stage maize leaf blight makes it particularly difficult to detect, necessitating high accuracy in identifying small targets. Singh et al. [26] developed a hybrid DL framework, combining *Long Short-term memory* (LSTM) units with convolutional layers to identify *corn leaf eye spot* (CLES) disease at different degrees of severity. Six thousand images collected from actual agricultural fields made up a real-time dataset used for training the model. With an accuracy of 95.88%, the system proved to be most effective among the four defined stages in spotting early-stage infections. This work emphasized the need to combine spatial and temporal aspects for more accurate disease diagnosis in agricultural fields with enhanced improvements. CNNs can automatically extract hierarchical features from raw input images, making them highly effective for image classification tasks [27]. CNNs are particularly well-suited for analyzing visual patterns, such as those found in leaf textures, shapes, and discolorations caused by different diseases [28, 29]. By learning spatial hierarchies of features through convolutional layers, CNNs can distinguish between healthy and diseased leaves with remarkable precision [30]. In recent studies, CNN-based models have achieved outstanding performance in detecting and classifying corn leaf diseases, with accuracy rates reported as high as 98.78% [31–33]. This level of precision underscores the potential of CNNs in real-world agricultural applications, where accurate disease identification can lead to better crop management and reduced yield loss [34]. Furthermore, advancements in *transfer learning* (TL), data augmentation, and lightweight CNN architectures have made it feasible to deploy such models even on mobile devices and low-cost embedded systems, thereby increasing accessibility and practical usability for farmers in remote and resource-limited areas [35].

The structure of the paper follows this organizational order: Section 2, gives the brief idea on the previous work. Section 3 outlines the description of the PlantVillage database. In Section 4, the ML models employed for the evaluation of the dataset, along with the evaluation metrics, are presented. Section 5 presents the experimental results, their respective associated discussions, and Section 6 summarizes the conclusion.

2 Literature Review

A study demonstrated the effectiveness of CNNs in detecting corn leaf diseases like NCLB (BL), GLS (GL), and CR, reporting a remarkable 99.58% accuracy using a relatively shallow architecture with optimized layers, ReLU activation, and the Adam optimizer [36]. Similarly, provide a comprehensive review of both ML and DL techniques for disease prediction and seed quality classification. Their findings emphasize that while DL models perform exceptionally well with large datasets, ML models like *support vector machine* (SVM), *random forest* (RF), and *Gradient Boosting* (GB) are still valuable, especially when paired with efficient feature extraction methods [37]. Despite their high performance, CNNs have certain limitations. CNN typically requires large, well-labeled datasets and considerable computational power for training. On the other hand, traditional ML models like SVM, *k*-Nearest Neighbors (*k*-NN), RF, and GB methods are more resource-efficient. When paired with carefully designed feature extraction techniques-such as GLCM, HoG, LBP, and color histograms. The models can still deliver strong performance and offer greater interpretability [2, 37]. Song et al. [38] used a method called SVM to spot different maize leaf diseases and achieved an accuracy of 89.6%. SVMs can handle both small and large datasets, but they don't always provide the best accuracy, especially when compared to some newer techniques. Dash et al. [39] tried combinations of traditional ML and DL models, used the DenseNet201 model to extract features from the images, and then fed those into an SVM that had been fine-tuned using *Bayesian optimization*. This combination performed better than a plain SVM, achieving a solid 94.6% accuracy on nearly 5,000 maize leaf images. Authors also addressed some common problems encountered with real-world photos, such as reflections and changes in lighting, which can interfere with model learning. Another approach from Daneshwari et al. [40], who worked on a modified *k*-NN model aimed at early detection of corn leaf diseases. Researchers designed it to extract both detailed and general features and used advanced mathematical techniques behind the scenes, including something called *restricted intensity*, DOR, and a technique called the *Directional Set* for optimization. Their model performed exceptionally well by achieving 99.86% accuracy, with strong sensitivity and specificity scores too. In fact, it outperformed several existing methods. Appalanaidu et al. [41] investigated the performance of various ML algorithms for classifying plant diseases using

the *PlantVillage* benchmark dataset, including *k*-NN, *Naïve Bayes* (NB), SVM, DT, and *Artificial Neural Network* (ANN). Their research centered on eight crops: *potatoes, corn, apples*, and *grapes*. With an average accuracy of 83.71%, the results showed that ANN consistently outperformed other models. On the other hand, DT and SVM achieved average precision scores of 80.42% and 80.27%, respectively; NB and *k*-NN performed least, with average precision scores of 75.62% and 66.23%. With an accuracy of 91.35%, ANN specifically outperformed SVM (63.6%) and DT (68.22%) for maize leaf diseases. These outcomes highlight the superior plant classification accuracy of neural network models across a variety of crop plants [64]. Priyaradhikadevi et al. [42] suggested ML models for identifying plant leaf diseases by applying DT and GB algorithms. The model was trained and validated on the *PlantVillage* dataset, which consists of images of healthy and infected leaves of *apple, grape, tomato*, and *corn* plants. The proposed methodology included pre-processing steps such as grayscale conversion, median filtering, extraction of textural features using the *Gray-Level Co-occurrence Matrix* (GLCM), and using these textural features for disease prediction via classifiers. The results indicated that GB achieved 94.59% accuracy on corn and 80.02% global accuracy, while the DT achieved 95.76% accuracy on corn but only 69.89% on the joint dataset, clearly evidencing GB to be superior than DT. Furthermore, this research demonstrated that the exploitation of ensemble learning for classification may increase classification accuracy in the area of plant disease diagnosis, especially in the early detection of diseases affecting food production. Jaisakthi et al. [43] created an automatic system to detect grape leaf disease using ML and image processing. The system identifies the primary area in images using the *GrabCut* segmentation algorithm. It extracts the diseased part using global thresholding and a semi-supervised color filter. The system retrieves texture features from the segmented parts using GLCM and also extracts color features. It then classifies grape leaves into four groups, such as *healthy, black rot, esca*, and *leaf blight*. The study tested three ML classifiers-SVM, RF, and AdaBoost on a dataset of 1,135 grape leaf images. The SVM classifier performed best with global thresholding, achieving a test accuracy of 93.03%. AdaBoost, which uses DTs, achieved 83%, while RF reached 74.79%. This work revealed that SVM performs well for disease detection. Amin et al. [33] combined features from two pre-trained CNN architectures: EfficientNetB0 and DenseNet121. This model identified four categories: *GLS, CR, NLB*, and *healthy corn* (HL) leaves, using a portion of the *PlantVillage* dataset. The system extracted deep features from each CNN and then combined them to build a more robust feature representation, improving the classification performance. The model achieved a classification accuracy of 98.56%. This result outperformed other architectures tested, such as ResNet152 (98.37%), DenseNet121 (97.82%), EfficientNetB0 (97.91%), and InceptionV3 (96.26%). Data augmentation, early stopping, and the use of separate preprocessing layers for each CNN helped the model perform well on new data. The study concluded that combining features from lightweight CNNs can outperform larger models with many parameters, offering a promising direction for future work. Sami et al. [44] investigated a method to classify plant leaf diseases based on texture, using SVM, XGBoost, and CNNs. The study employed *Local Binary Patterns* (LBP) to extract texture details from images of five plant types. These details included ridges, hairs, and waxy coatings. Along with mango, the plants studied included Alstonia, Guava, Jamun, and Lemon. The images were captured in various outdoor settings. They trained several models using the texture details. These models performed a two-category classification—healthy versus diseased. The CNN that used LBP features (CNN-LBP) performed best among all the models, achieving accuracies of 97% for Guava, 95% for Mango and Lemon, 98% for Jamun, and 97% for Alstonia. SVM-LBP and XGBoost-LBP models, however, showed lower accuracies. This was especially true for leaves with complex structures, such as Guava and Alstonia. The study showed that CNN-LBP effectively handles variations in lighting, orientation, and size, making it a suitable option for classifying plant diseases in farm settings [69]. Lokhande et al. [45] conducted a comparative study on AlexNet and ResNet50, focusing on their effectiveness in classifying plant leaf diseases in maize and soybean crops. They worked with over 6,500 labeled images sourced from the *PlantVillage* dataset and local agricultural fields, which included prevalent diseases such as GLS, CR, and NCLB in maize, and *frogeye leaf spot, powdery mildew*, and *downy mildew* in soybean. The models were trained with an $80:20$ split for training and testing, employing pre-processing and data augmentation techniques to enhance generalization. Both feature extraction and classification were performed end-to-end using CNNs, and TL was used to fine-tune both models for improved performance on the specific plant data.

The experimental results clearly showed that ResNet50 consistently outperformed AlexNet in both crop types. Specifically, ResNet50 achieved an impressive 97.41% accuracy in classifying soybean diseases and 96.74% for maize. In contrast, AlexNet achieved 96.40% accuracy for soybeans and 95.99% for maize. The study also highlighted ResNet50's greater depth and its effectiveness in addressing vanishing gradient issues, making it a better choice for tackling complex PDD tasks. These results emphasize how deeper CNN architectures can significantly improve accuracy and reliability in practical agricultural applications. Reddy et al. [46] presented a system for automatically detecting plant diseases using DL methods. They utilized image pre-processing along with classification techniques involving SVM and *k*-NNs. Despite this, the results indicated that a CNN model outperformed both methods. The experimental results revealed that the CNN model achieved an impressive accuracy of 96%, compared to 94% for SVM and just 82% for *k*-NN. Hassan et al. [47] conducted a comparative study on various DL models for detecting plant leaf diseases using TL. Their research utilized a large and diverse dataset that included 87,612 images representing 38 different plant disease categories. The dataset comprised leaf samples from different crops, featuring both healthy and infected specimens of apple, corn, tomato, grape, and pepper. Researchers assessed five CNN-based architectures: *DenseNet*, *ResNet50*, *MobileNet*, *Xception*, and *EfficientNetB3*, all while applying the same pre-processing and augmentation techniques. Each model underwent fine-tuning with early stopping and optimized hyperparameters. In the end, EfficientNetB3 stood out as the best performer, excelling in both accuracy and training efficiency. The experimental findings indicated that EfficientNetB3 achieved an outstanding test accuracy of 99.92% and an F1-score of 0.999, which was a significant improvement over DenseNet (92.59%), ResNet50 (95.27%), MobileNet (96.39%), and Xception (98.92%). The study highlighted the critical importance of choosing suitable architectures and TL methods for effective PDD, particularly when handling extensive multi-class datasets. The approach presented in this research shows great promise for real-time applications in smart agriculture systems, aiming to reduce crop losses and enhance disease management efforts.

Despite substantial advances in traditional and modern ML approaches, the existing literature indicates several ongoing challenges. Deep neural networks, although highly accurate, typically require extensive computational resources, large volumes of annotated data, and meticulously tuning of hyperparameters. These requirements limit their practical deployment, particularly in real-world agricultural settings where computational infrastructure may be constrained. In contrast, classical ML models offer greater interpretability and lower computational overhead, but often struggle to maintain consistent performance across diverse maize leaf diseases, especially in the presence of class imbalance, natural variability in field conditions, and heterogeneous imaging environments. Collectively, these limitations underscore the need for models that combine high predictive accuracy with robustness, scalability, and operational efficiency. In response to these gaps, the present study focuses on evaluating the potential of tree-based ensemble learning methods including RF, ETs, GB, LightGBM, XGBoost, and a Stacking ensemble for the accurate classification of key maize foliage fungal diseases such as NCLB, GLS, CR, and healthy leaf categories. Supporting a large and balanced subset of the PlantVillage dataset, this work systematically compares the performance of these algorithms to identify a computationally lightweight yet highly reliable diagnostic solution. The overarching objective is to establish a generalizable and field-ready framework capable of enabling timely, automated disease detection in precision-agriculture applications.

In summary, while previous studies have demonstrated the potential of digital and intelligent systems in various agricultural domains, there remains a lack of integrated approaches that combine data-driven analytics, IoT-based sensing, and AI-powered decision models for sustainable agricultural management. To address these gaps, the present study focuses on developing and validating an intelligent agricultural framework that supports digital technologies for improved productivity and resource optimization.

3 Database Description

Corn is one of the major staple foods consumed worldwide, alongside rice and cassava, and plays a particularly important role in Indonesia as a primary source of carbohydrates [59]. The high demand for corn necessitates its large-scale production to ensure an adequate supply. Any decline in production levels can significantly disrupt market stability, thereby negatively impacting consumers in

Table 1. The number of corn leaves distribution is represented from the PlantVillage dataset.

Diseases	HL	GLS	NCLB	CR
Number of Images	3,486	2,629	3,548	3,682

regions such as Indonesia, where corn consumption is substantial. Multiple factors contribute to reduced corn yields, among which plant health is a critical determinant. Unhealthy corn plants often exhibit stunted growth, fail to develop kernels, suffer severe damage, ultimately leading to yield losses and, in extreme cases, crop failure [60]. The dataset used in this study is an organized subset of the publicly available PlantVillage dataset hosted on Kaggle [61]. The complete PlantVillage dataset, originally introduced by Hughes and Salathe [62, 63], contains approximately 2,17,000 images that span 38 categories of healthy and diseased plant leaves. The selected subset comprised of four classes: NCLB (3,548), CR (3,682), GLS (2,629), and HL (3,486) leaves. The classwise distribution of the images is provided in Table 1, while the respective representatives are provided in Figure 1 to highlight the variability in the diseases of corn. Thus, this subset of the PlantVillage dataset gives a diverse and balanced representation of both diseased and healthy corn leaves, enabling the early detection of corn plant diseases and ultimately assisting farmers in improving crop health and yield.

4 Model Architectures

***k*-NN:** It is a straightforward and useful instance-based learning method. This method is often applied to classification and regression problems. It does not rely on any specific distribution of the data, making it a suitable and baseline option for real-world datasets with irregular class boundaries [48]. For a classification problem, the model checks the classes of its *k*-NNs in the feature space to decide the class of a new data point that is not in the training set. The neighbors are projected based on the Euclidean distance, which defines similarity based on the label for an unknown sample that can be estimated by majority voting of its neighbors. The *k*-NN model was set to $k = 300$ neighbors with distance weighting. While *k*-NN with majority vote calculation is the more common method, this *k*-NN model assigns greater relative importance to closer neighbors based on inverse distance weighting. The class $\hat{\chi}$ can be predicted as:

$$\hat{\chi} = \underset{c}{argmax} \sum_{i \in N_k(\chi)} \frac{1}{\|\chi - \hat{\chi}\| + \epsilon} \cdot \mathbb{I}(\chi_i = c) \qquad (1)$$

where N_k is the set of *k*-NNs of the test instance χ, $\hat{\chi}$ is the class label of neighbor i, and ϵ is a small constant to prevent division by zero. A large k value ensures smoother decision boundaries and minimizes overfitting, especially when combined with feature selection and normalization.

DT: A popular supervised learning algorithm, the DT classifier learns a set of decision rules from the input features to solve classification problems. The model is organized in a tree-like fashion, with each internal node standing for a feature test, each branch for the test's outcome, and each leaf node for the final class assignment. The algorithm keeps breaking the dataset down into smaller chunks, and with each split, the groups get more uniform in terms of the target class. At every step or node, it picks one feature and a cutoff value that helps separate the classes more clearly. This choice is usually guided by impurity measures like entropy or the Gini index, which provide the algorithm with information about how mixed the data is at that point [49]. The trained DT classifier was produced using a hyperparameter optimization routine $GridSearchCV$. The parameters optimized included: $max\_depth, min\_samples\_split, min\_samples\_leaf$, ccp_alpha, and $max\_features$. The parameter $class\_weight = 'balanced'$ was also specified to appropriately counter class imbalance in the model training. The DT classifier recursively splits the feature space by finding the feature and threshold that yields the smallest Gini impurity, defined as:

$$Gini(S) = 1 - \sum_{k=1}^{N} p_k^2 \qquad (2)$$

where p_k is the proportion of samples belonging to class k in a node S. At every node, the DT classifier finds the feature and threshold that yields the greatest decrease in Gini impurity before splitting this node. In this way, the classifier develops purer child nodes and, consequently, better class separation in the training model. The final model was chosen based on the maximum cross-validation accuracy throughout the grid search, while ensuring the resulting model provided a good balance between explainable outcomes and predictive power.

SVM: The SVM is a popular supervised training

Figure 1. The diseased leaves from the maize of PlantVillage dataset represent HL, GLS (GL), NCLB (BL), and CR used for plant leaf disease classification. The classification of various diseases of maize leaves by learning the local and global features together.

algorithm for binary and multiple-class classification problems. It performs extremely well with datasets that include a large number of characteristics, regardless of whether there are fewer samples than features. Identifying the best hyperplane, or boundary, that divides classes by providing the greatest space between them is the main goal of SVM. Since they are the data points closest to this boundary, support vectors are the most important in developing it [50]. The SVM has been configured with a *Radial basis function* (RBF) kernel, regularization parameter $C = 10$, and $class\_weight =' balanced'$. This model creates the best separating hyperplane in the transformed feature space using a nonlinear kernel. The decision function is defined as:

$$f(\chi) = sign(\sum_{l=1}^{L} \xi_l \gamma_l \pounds(\chi_l, \chi) + \alpha) \tag{3}$$

where α_l are the Lagrange multipliers, ξ_l are the class labels, and $\pounds(\chi_l, \chi)$ is the RBF kernel, defined as:

$$\pounds(\chi_l, \chi_m) = exp(-\beta \left\| \chi_l - \chi_m \right\|^2) \tag{4}$$

The $class\_weight =' balanced'$ parameter enables the model to account for class imbalance during training. The RBF kernel allows the model to capture nonlinear and complex patterns in the manually constructed feature space, enabling strong generalization performance in multiclass classification problems.

RF: A single DT may be quite weak, especially if the data used to train it has noise or is small. RF works on this by having many trees. Each tree looks at a slightly different portion of the data and selects from a random group of features. Then, when it's time to make a prediction, the model averages the results or takes a vote for classification. It's a straightforward concept, but it performs effectively because it reduces overfitting without requiring much tuning [51].

The RF classifier was employed with 200 estimators, $max\_depth = 19$, and $class\_weight =$ "$balanced$". It is an ensemble model that aggregates the outputs of multiple DTs trained on bootstrapped samples and random feature subsets. Each tree uses $Giniimpurity$ to determine the best split.

$$Gini(S) = 1 - \sum_{k=1}^{N} p_k^2 \tag{5}$$

where p_k is the proportion of samples of class k in a node S. The final prediction is made by majority vote.

$$\hat{\chi} = mode(h_1(x), h_2(x), \ldots, h_K(x)) \tag{6}$$

where $h_K(x)$ is the prediction of the K^{th} DT. RF reduces variance and is robust to noise and overfitting, especially when combined with balanced class weights and limited tree depth.

ETs: *Extremely randomized trees* (ERT) is a classification method that makes many DTs to make more right guesses and fewer wrong ones. This way of making trees is often used in the field of fixed data because it learns fast and is steady. The best part of ERT is that it makes many different trees by adding randomness, when the basic part of ERT is that it makes many different trees by adding randomness when choosing features and setting a split to go into a new branch at each node. This way of making trees makes the model work better and does not matter if the data are changed. ERT works well with structured or tabular data sets, where every feature has its own meaning, and is not in a way that is spatial or makes sense in a row. In these cases, models based on trees tend to do better than DL ways of working [52]. DL often needs a lot of data that is in a spatial or row type of way. ERT also lets us know how it makes its results. This is very good for health and farm jobs. It is good to know how the model makes its calls.

The ET classifier was created with 250 estimators and $max\_depth = 22$. In contrast to RFs, which seek the best splitting thresholds, ETs introduce more randomization by selecting split points randomly within the feature's value range. This contributes to lower variance and better generalization. The decision at each node is made by: $Split : x_f \leq \theta$, Where $\theta \sim \mathcal{U}(a, b)$ with a and b being the minimum and maximum values of feature f in the data subset. Like RFs, ETs aggregate decisions using majority voting across all trees. $\hat{\chi} = mode(h_1(x), h_2(x), \ldots, h_k(x))$. The model employed $class\_weight =$ '$balanced$' to address class imbalance and leveraged parallel processing $n\_jobs = -1$ to accelerate training.

GB: GB is an ensemble learning method that builds a strong predictive model by combining several weak learners, usually DTs, one after another. In each step, the algorithm adds a new model that tries to fix the mistakes made by the earlier combined models. This fix is done by fitting the new model to the negative gradient, or pseudo-residuals, of a specific loss function based on the current ensemble prediction. Unlike methods like bagging that build models at the same time, GB improves the model step by step using gradient descent in function space [53]. This approach effectively captures complex non-linear patterns and works well with many types of loss functions. The GB model was configured with 200 estimators, a learning rate of 0.05, and a maximum depth of 5 for each tree. This ensemble method builds DTs in a sequential manner, where each new tree is trained to minimize the residual error from the previously built ensemble. The model is updated iteratively as:

$$F_m(x) = F_{m-1}(x) + \xi h_m(x) \tag{7}$$

where $F_m(x)$ is the ensemble prediction at iteration m, ξ is the learning rate, and $h_m(x)$ is the weak learner fit to the negative gradient of the loss function. A $min\_samples\_s[lit$ of 5 was used to regulate the minimum number of samples required to split an internal node, which aids in generalization. This configuration enables effective learning of complex relationships in the feature space while minimizing overfitting through controlled model complexity and shrinkage from the learning rate.

SGD: SGD is an optimization algorithm, and it helps to train large linear models. The method changes the model parameters one step at a time. It uses samples from the training set, which are picked at random, instead of using all the data. This way, the process uses less computation time and less memory -it works well with data that has many dimensions. It also performs well in online learning setups. The SGD classifier was improved by using the $modified\_huber$ loss function, which provides a smooth approximation of the hinge loss and is robust against outliers [54]. The classifier was trained with $max\_iter = 5000$ to ensure sufficient convergence, and $\alpha = 0.00005$ to apply weaker regularization and avoid underfitting. An $L_2 - norm$ penalty was applied to manage the complexity of the model. The weight update rule for SGD followed the standard gradient descent formulation.

$$\beta_{k+1} = \beta_k - \xi_k \nabla \psi(\omega_k) \tag{8}$$

where ξ_k is the adaptive learning rate with $learning\_rate =' optimal'$ and $\nabla \psi(\omega_k)$ is the gradient of the loss function. The flag $early\_stopping = True$ monitored validation performance, using 10% of the training dataset. The system is stopped training if no further improvement is observed for 15 consecutive epochs $n\_iter\_no\_change = 15$. These configurations were chosen for their robustness in generalization, reducing training time, and ability to tolerate noise in the corn leaf disease detection dataset.

AdaBoost: AdaBoost is a step-by-step way to use weak learners to make one strong one. It trains weak models, shallow DTs, on weighted data. It focuses on correcting samples that were wrong. In each round, the weights of the misclassified samples increase. This causes the next learner to focus more on these tough cases. This reweighting process is what makes AdaBoost different from other boosting methods. The AdaBoost framework assumes that each base learner does a bit better than random chance. It combines their outputs using weighted majority voting. A weak learner's role in the final prediction depends on its accuracy, which is based on the classification error related to the weighted distribution of training samples. This method allows AdaBoost to reduce both bias and variance over several iterations while keeping it easy to understand by using decision stumps or shallow trees [55].

The AdaBoost classifier was improved by setting the $DecisionTreeClassifier$ with $max\_depth = 4$, and $min\_samples\_split = 5$ as the weak learner. The model was trained using 400 estimators and a learning rate of 0.1 to constrain the contribution of each weak learner and stabilize training. AdaBoost is an ensemble of weak classifiers trained sequentially, with each new classifier emphasizing the observations that previous ones misclassified. The final hypothesis is the weighted

sum of the weak hypotheses:

$$\hat{\chi} = sign(\sum_{k=}^{K} \xi_k h_k(x)) \tag{9}$$

where $h_k(x)$ is the k^{th} weak learner, and χ_k is its weight, calculated as

$$\xi_k = \frac{1}{2} ln(\frac{1-\varepsilon_k}{\varepsilon_k}) \tag{10}$$

where ε_k denotes the weighted classification error associated with h_k. This formulation ensures that more accurate learners contribute more to the final decision. The use of shallow trees as base learners helps reduce overfitting while maintaining performance on complex patterns.

LightGBM: LightGBM is a fast and efficient version of the *Gradient Boosting DT* (GBDT) framework. Microsoft introduced it in 2017. LightGBM stands out from traditional GBDT algorithms because it uses a leaf-wise tree growth strategy instead of a level-wise approach. In each iteration, it picks the leaf with the highest delta loss for splitting. This results in quicker convergence and improved accuracy. LightGBM also employs a histogram-based feature binning method, which groups continuous feature values into discrete bins [56]. This significantly reduces memory use and speeds up training. The LightGBM classifier was trained using 200 estimators, with a learning rate of 0.05, a maximum depth of 7, and 31 leaves per tree. The LightGBM model extends the benefits of GB by incorporating *Gradient-based One-Side Sampling* (GOSS) and *Exclusive Feature Bundling* (EFB) to accelerate training time while preserving predictive accuracy. Unlike traditional boosting frameworks such as XGBoost that grow trees level-wise, LightGBM grows trees leaf-wise, which can achieve lower loss but may risk overfitting. This was mitigated using $min\_child\_samples = 10$. Similarly to regular boosting, the ensemble is constructed additively as follows:

$$F_k(x) = F_{k-1}(x) + \xi \cdot h_k(x) \tag{11}$$

where $F_k(x)$ is the model prediction at iteration k, ξ is the learning rate, and $h_k(x)$ is the weak learner fit to the residuals or negative gradients. This formulation enables LightGBM to prioritize informative data points and minimize loss efficiently. The chosen parameters facilitate fast computation and strong generalization across the high-dimensional feature space extracted from corn leaf images.

XGBoost: It is an advanced ensemble learning method that generates additive models sequentially in a stage-wise fashion to optimize a regularized objective function. XGBoost differs from traditional GB in that it uses the first-order gradients (residuals) and second-order derivatives (Hessians) in formulating the search direction for boosting, allowing for more accurate approximations and faster convergence times. It is very well regarded for its scalability, regularized optimization approach, and ability to handle structured/tabular data [53].

The XGBoost model was configured with 200 estimators, a learning rate of 0.05, maximum tree depth of 6, and both $subsamples$ and $colsample\_bytree$ set to 0.8 to reduce overfitting. XGBoost extends classical GB by applying second-order Taylor expansion to the loss function, capturing both the gradient and curvature information. The objective minimized at each boosting iteration is:

$$\mathcal{L}^{(t)} = \sum_{k=1}^{K} [g_k f_t(x_k) + \frac{1}{2} h_k f_t^2(x_k)] + \Omega(f_t) \tag{12}$$

where g_k and h_k are the first and second derivatives (gradient and Hessian) of the loss with respect to the prediction, and $\Omega(f_t)$. This formulation allows for more precise optimization and improved convergence speed. By incorporating regularization, feature sampling, and row subsampling, XGBoost achieves better generalization and is well-suited for high-dimensional and class-imbalanced datasets such as those derived from handcrafted corn leaf features.

5 Experimental Results and Discussion

The performance of several ML models was tested on the data set consisting of 4 classes: NCLB, CR, GLS, and HL. The former compared the conventional classifiers: k-NN, DT, SVM, RF, and ETs, and the latter compared the ensemble and boosting techniques such as GB, SGD, AdaBoost, LightGBM, XGBoost, and stacking model. The performance was evaluated in terms of precision (P), recall (R), F1-score (F), overall accuracy (%), macro average, and weighted average scores. The macro average precision: Macro-averaging combines the contributions of every class to create a global performance measure. True positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) for every class are summed, and the precision is calculated based on aggregated TP, FP, and FN. Macro-averaged precision is the total number of TP predictions divided by the total number of predicted

Table 2. The comparative performance of ML models such as k-NN, DT, SVM, RF, and ETs evaluated across multiple metrics on the classification task. The performance was assessed in terms of P, R, and F for four label categories: NCLB, CR, GLS, and HL. In addition, overall accuracy (%), macro average, and weighted average values are provided to capture model-level performance.

Models	k-NN			DT			SVM			RF			ETs		
Metrics	P	R	F	P	R	F	P	R	F	P	R	F	P	R	F
NCLB	0.90	0.93	0.92	0.89	0.90	0.90	0.91	0.95	0.93	0.95	0.96	0.96	0.96	0.96	0.96
CR	0.99	0.94	0.97	0.99	0.96	0.97	0.99	0.98	0.98	1.00	0.96	0.98	1.00	0.97	0.98
GLS	0.85	0.90	0.87	0.84	0.87	0.86	0.92	0.87	0.89	0.92	0.95	0.93	0.92	0.96	0.94
HL	0.99	0.96	0.97	1.00	0.99	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Accuracy (%)		93.52			93.44			95.39			**96.93**			**97.38**	
Macro Avg.	0.93	0.93	0.93	0.93	0.93	0.93	0.95	0.95	0.95	0.97	0.97	0.97	0.97	0.97	0.97
Weighted Avg.	0.94	0.94	0.94	0.94	0.93	0.93	0.95	0.95	0.95	0.97	0.97	0.97	0.97	0.97	0.97

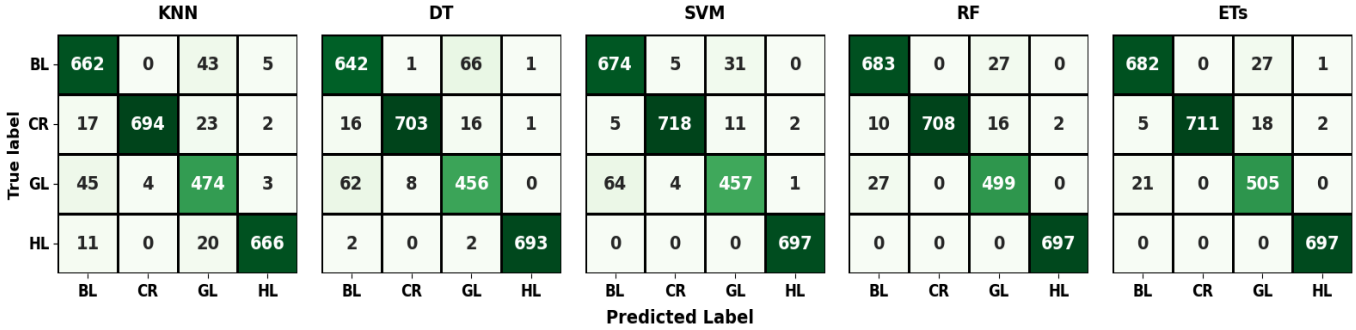

Figure 2. Confusion matrices of different ML classifiers for maize disease classification. Each matrix compares the true and predicted labels for four disease categories: NCBL, CR, GLS, and HL. The diagonal values represent correctly classified samples, while off-diagonal cells indicate misclassification. Among all models, RF and ETs exhibited the highest classification accuracy with minimal misclassification across all disease classes.

positives across all classes. Macro-averaged precision gives equal weight to every instance, regardless of its class, and is therefore beneficial for multi-class classification problems with imbalanced datasets, adding more weight to classes that have more samples [57]. The weighted-average precision is especially valuable in problems with multiple classes because it helps with class imbalance by specifying weights based on the number of class instances. Thus, classes with larger sample sizes will weigh more in the overall metric, providing a more accurate indication of overall system performance than a macro-average, which could treat each class equally [58].

The different algorithms tested in the dataset in Table 2 showed that ETs and RF consistently achieved the highest accuracy scores of 97.38% and 96.93%, along with macro and weighted average P, R, and F scores of 0.97 each. As both methods are ensemble learning algorithms based on multiple DTs combining in a strong classifier, this was reflective of the complexity of the dataset being evaluated. The k-NN classifier, while slightly lower in absolute terms (93.52%), at least suggested that nearest-neighbor-based classification could be a useful method to explore for the dataset. The SVM classifier suggested an accuracy of 95.39%. Although SVM obtained balanced class prediction performance for the dataset, the performance was lower than the ensemble methods. The DT model obtained the lowest accuracy (94.08%) overall, suggesting poor generalization for all classes, although the DT model appropriately characterized the HL class. Among the individual class-level metrics, HL demonstrated near-perfect classification across all models, with several achieving P=R=F=1.00. Conversely, the GLS class exhibited more variability, where models such as DT and k-NN yielded slightly lower F1-scores (0.86-0.87), while ensemble models like RF and ETs improved the performance to 0.93-0.94 can be observed from Figure 2.

Table 3 represents the data, which indicates that the stacking model had the best accuracy (98.50%), closely followed by the LightGBM accuracy (98.46%) and XGBoost (98.01%), demonstrating the strong generalization capability of ensemble approaches. These boosting and stacking-based methods are able to produce strong classifiers from several weak learners and are effective for complex data sets. The customized GB method also performed well with solid results at 97.94%, whereas AdaBoost records

Table 3. The comparative performance of ML models such as GB, SGD, AdaBoost, LightGBM, XGBoost, and stacking models evaluated across multiple metrics on the classification task. The performance is assessed in terms of P, R, and F for four label categories: NCLB, CR, GLS, and HL. In addition, overall accuracy (%), macro average, and weighted average values are provided to capture model-level performance.

Models	GB			SGD			AdaBoost			LightGBM			XGBoost			Stacking Model		
Metrics	P	R	F	P	R	F	P	R	F	P	R	F	P	R	F	P	R	F
NCLB	0.97	0.98	0.97	0.90	0.85	0.87	0.92	0.95	0.93	0.97	0.98	0.98	0.97	0.98	0.98	0.98	0.98	0.98
CR	0.99	0.97	0.98	0.97	0.95	0.96	0.99	0.97	0.98	0.99	0.98	0.99	0.99	0.97	0.98	0.99	0.98	0.98
GLS	0.96	0.96	0.96	0.77	0.86	0.81	0.91	0.90	0.90	0.97	0.98	0.97	0.95	0.96	0.96	1.00	1.00	1.00
HL	1.00	1.00	1.00	0.99	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.98	0.98	0.98
Accuracy (%)		**97.94**			91.61			95.84			**98.46**			**98.01**			**98.50**	
Macro Avg.	0.98	0.98	0.98	0.91	0.91	0.91	0.96	0.95	0.95	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98
Weighted Avg.	0.98	0.98	0.98	0.92	0.92	0.92	0.96	0.96	0.96	0.98	0.98	0.98	0.98	0.98	0.98	0.99	0.99	0.99

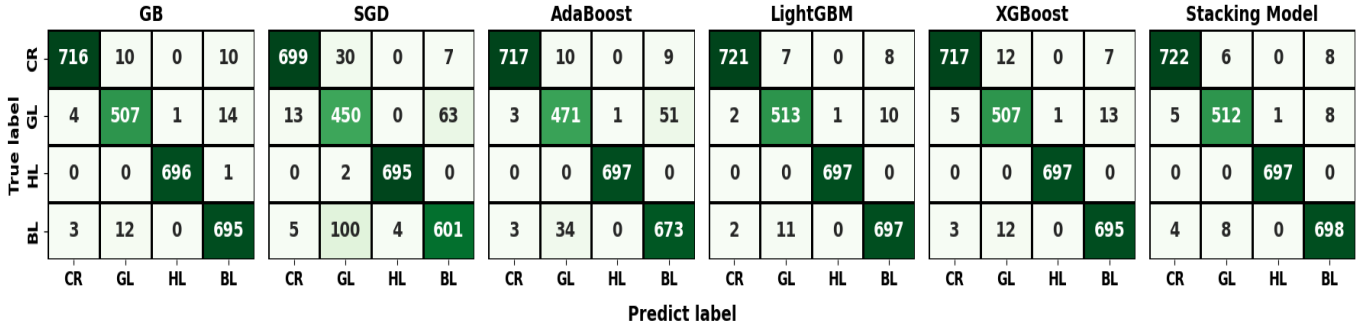

Figure 3. Confusion matrices of ensemble ML classifiers for maize disease classification. The matrices display the correspondence between true and predicted labels for four classes: CR, GLS, NCLB, and HL. Diagonal values indicate correct classification, while off-diagonal values represent misclassifications. Among all models, LightGBM, XGBoost and Stacking Model demonstrate superior performance with the highest number of correctly classified samples across all disease categories.

slightly lower performance (95.84%). On the other hand, SGD produced the lowest with 91.61% accuracy, which reflected limited effectiveness for this data set, indicating its limited capability to capture complex non-linear relationships within the dataset. The overall performance matrices in Table 3 indicated that the stacking model, LightGBM, and XGBoost models had the highest results on P, R, and F, whereas SGD, AdaBoost, and GB performed slightly lower than other ensemble methods.

The overall performance matrices present in Table 3 suggested the following. At the class level, near-perfect classification was observed for the HL category, where most models achieved P=R=F=1.00, except for the stacking model (0.98). The NCLB and CR classes also achieved consistently high scores across models, particularly under ensemble methods. However, the GLS class exhibited noticeable variation, while the stacking model achieves perfect P, R, and F (1.00). SGD lagged with an F of 0.81, highlighting the benefits of ensemble learning in handling class imbalance and variability. The macro and weighted averages further reinforced these findings, where ensemble-based models (LightGBM, XGBoost, and stacking) consistently achieved values of 0.98-0.99 across all three evaluation metrics, significantly outperforming traditional approaches, as observed in Figure 3.

Taking into consideration both Table 2 and Table 3, it seems that the stacking model and LightGBM achieved the best results, and they were followed closely by GB, XGBoost, ETs, and RF. The SVM classifier also showed performance competent enough to come in the same category, while DT had moderate success. On the other hand, k-NN and SGD were two classifiers that were the least reliable in this study. These results show that ensemble and boosting-based methods have a significant advantage over traditional classifiers and may be the best candidates for plant disease classification. Based on the algorithms applied in Table 2 and Table 3, the following conclusions can be drawn:

- The stacking model culminated in the best overall

accuracy of 98.50%, thus providing the best performance when being learned in the data. It consistently provided the best P, R, and F through all classes.

- LightGBM came in at a high accuracy of 98.46%. This shows strong power for generalization ability; LightGBM has been seen to be efficient, scalable, and thus is very useful for large-scale data.
- The XGBoost achieved accuracy scores of 98.01%, slightly lower than LightGBM and GB, but still a high-performance learner. XGBoost also yielded one of the best unweighted average precision and recall.
- The GB algorithm was able to achieve an accuracy of 97.94%. Being an ensemble, meaning it takes advantage of multiple DT learners. The GB algorithm was consistent through all classes, with its metrics balanced.
- ETs and RF classifiers had an accuracy of 97.38% and 96.93%, respectively, confirming the power of ensemble methods. Both ETs and RF were able to classify effectively, especially for the CR and HL classes.
- The k-NN classifier achieved an accuracy of 93.52% and was the most competitive base model despite its simplicity. However, it was slightly less consistent than the boosting and stacking methods.
- The SVM classifier achieved 95.39% accuracy, which suggests there was balanced performance across classes, but ultimately did not do as well as other ensemble and boosting-based methods.
- The DT classifier had the lowest accuracy in Table 2 (93.44%), indicating it had weaker generalization despite performing well for the HL class.
- AdaBoost achieved an accuracy of 95.84%, which was lower than other boosting-based approaches, namely GB and XGBoost. It was effective in its usage but not consistent across all classes.
- The last classifier is SGD, which had the lowest overall accuracy of 91.61%, ranked it as the least effective method in all the experiments performed.

Ultimately, ensemble-based methods such as the stacking model, LightGBM, XGBoost, GB, ETs, and RF demonstrated superior performance over classical classifiers. k-NN and SVM produced acceptable scores, but were outweighed by DT, AdaBoost, and SGD, which were the weakest. These results confirm that the stacking and boosting algorithms are the most reliable for the classification of plant diseases.

Table 4 presents a comprehensive analysis of the related literature, which reveals that earlier ML approaches, such as classical SVM, yielded moderate accuracy of around 89.6% in [38]. With the rise of DL, hybrid models such as DenseNet201 + SVM achieved improved performance 94.6%, presented in [39], while advanced mathematical feature-engineering methods like modified k-NN [40], reported very high accuracy 99.86%. Neural networks and CNN-based studies [41, 42] achieved accuracy in the 91-96% range for Maize, especially when applied on PlantVillage subsets. DL architectures, particularly ResNet50 (96.74%) and EfficientNetB3 (99.92%), demonstrated strong performance but typically require high computational cost and extensive training data. Compared with these works, our study shows that tree-based ensemble methods such as Stacking, LightGBM, and XGBoost consistently achieve superior performance, reaching up to 98.50% accuracy with excellent macro and weighted average metrics across all four (CR, NCLB, GLS, and HL) classes. Particularly, our stack-based ensemble outperforms ML models such as SVM: 95.39%, DT: 93.44%, and approaches the accuracy of deep CNNs while maintaining far lower computation cost and faster training speed. This positions our method as a practical, robust, and computationally efficient alternative to DL heavyweights-especially valuable for real-time deployments, mobile devices, and low-resource environments.

6 Conclusion

The experimental evaluation demonstrates that the ensemble-based methods consistently outperform traditional classifiers for the classification of NCLB, CR, GLS, and HL in maize plants. Among conventional models, ETs, and RF achieved the highest accuracies, 97.38% and 96.93%, respectively, with strong and balanced macro and weighted average scores across all metrics. In contrast, DT, and k-NN yielded lower accuracies (93-94%) and exhibited reduced performance for the GLS class, although HL classification remained near perfect for all models. Similarly, for advanced ensemble and boosting techniques, the stacking model achieved the best performance (98.50% accuracy), closely followed by LightGBM (98.46%) and XGBoost (98.01%), all delivering near-perfect macro and weighted average

Table 4. **The comparative performance of ML models—including GB, SGD, AdaBoost, LightGBM, XGBoost, and stacking ensembles—was evaluated across multiple metrics. Their classification accuracy was further compared with the results reported in existing literature to assess relative effectiveness.**

Authors'	Model(s) Used	Dataset	Accuracy Reported	Key Notes
Song et al. [38]	SVM	Maize Leaves	**89.6%**	Early ML approach; lower accuracy than modern ensemble-based models
Dash et al. [39]	DenseNet201+SVM (BO optimized)	∼5000 Maize Images	**94.6%**	Deep features + optimized SVM improve performance over classical SVM
Daneshwari et al. [40]	Modified k-NN	Maize Leaf Images	**99.86%**	Very high accuracy; method designed for early detection; heavy mathematical preprocessing
Appalanaidu et al. [41]	ANN, k-NN, NB, SVM, DT	PlantVillage (Maize + other crops)	**ANN: 91.35%, SVM:63.6% DT: 68.22% for Maize**	ANN outperforms classical ML but still lower than modern ensembles.
Priyaradhikadevi et al. [42]	DT, GB	PlantVillage (Corn + other crops)	**GB: 94.59%, DT: 95.76% (Corn Only)**	Shows boosting >DT; ensemble trend consistent with our findings.
Lokhande et al. [45]	ResNet50, AlexNet	PlantVillage + Field Images	**ResNet50: 96.74% AlexNet: 95.99%**	Deep CNNs outperform earlier architectures; still slightly below our ensemble performance.
Hassan et al. [47]	CNN, TL	Multi-crop Datasets	**∼96 - 99% (Varies)**	CNNs strong but require heavy compute + data augmentation.
Sambana et al. [11]	EfficientNetB3	Multi-crop PlantVillage	**99.92%**	Highest DL accuracy reported; very deep network; high complexity.
Ensemble Models (Our Work)	Stacking, LightGBM, XGBoost, RF, ETs, DT, k-NN, AdaBoost, SGD	PlantVillage (Maize Subset: CR, GLS, NCLB, HL)	**Stacking: 98.50%, LightGBM: 98.46% XGBoost: 98.01%**	Best classical-ML ensemble performance; high generalization and robustness, particularly for difficult GLS class.

precision, recall, and F1-scores. GB also performed competitively at 97.94%, while AdaBoost and SGD showed reduced effectiveness, with SGD recording the lowest accuracy (91.61%) due to its limited ability to capture complex non-linear patterns. Class-level analysis revealed that the HL category is consistently classified with near-perfect precision and recall across all models, while the GLS class posed more challenges, particularly for non-ensemble methods. Overall, the results confirm that tree-based ensemble methods work well. Especially, stacking models, LightGBM, and XGBoost models offer superior generalization and robustness for imbalanced, multi-class datasets. Future work would focus on the development of automated, reliable, and accurate disease detection systems to strengthen global food security by enabling early diagnosis and targeted intervention, thereby optimizing pesticide usage, reducing crop mortality, and promoting sustainable agricultural productivity. The future work will be focus on integrating the proposed model with IoT and mobile-based diagnostic platforms to enable real-time, on-field disease detection and monitoring. The approach can also be extended to other crop species to develop a generalized bioinformatics framework for plant disease classification. Additionally, exploring deep learning-ensemble hybrid architectures will help capture spatial, temporal features under real agricultural conditions, further improving robustness, scalability, and prediction accuracy.

Data Availability Statement

The dataset used in this study is an organized subset of the publicly available PlantVillage dataset hosted on Kaggle (https://www.kaggle.com/datasets/abdallahalidev/plantvillage-dataset). The complete PlantVillage dataset, originally introduced by Hughes and Salathé, contains approximately 217,000 images that span 38 categories of healthy and diseased plant leaves, and can be accessed via the original source at https://github.com/spMohanty/PlantVillage-Dataset.

Funding

This work was supported by the Telecommunications Consultants India Limited (TCIL), a Government of India company under administrative control of the Department of Telecommunications (DoT), as part of the prestigious "100 5G Use Case Labs" initiative.

Conflicts of Interest

The authors declare no conflicts of interest. Rupesh Kumar is an employee of Yamaha Motor Solution Private Limited, Faridabad, Haryana 121003, India.

Ethical Approval and Consent to Participate

Not applicable.

References

[1] Banik, A., Patil, T., Vartak, P., & Jadhav, V. (2023, May). Machine learning in agriculture: A neural network approach. In *2023 4th International Conference for Emerging Technology (INCET)* (pp. 1-6). IEEE. [CrossRef]

[2] Panigrahi, K. P., Sahoo, A. K., & Das, H. (2020, June). A cnn approach for corn leaves disease detection to support digital agricultural system. In *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184)* (pp. 678-683). IEEE. [CrossRef]

[3] Rout, J. K., Rout, M., & Das, H. (2020). *Machine learning for intelligent decision science*. Springer. [CrossRef]

[4] Ashwini, C., & Sellam, V. (2022). Corn disease detection based on deep neural network for substantiating the crop yield. *Appl Math, 16*(3), 423–433. [CrossRef]

[5] Patrekar, S., Samrat, R., Kunder, S. R., Bhat, A., & Nikhil, P. B. (2025, February). AgriScan: Next-Gen Corn Leaf Disease Detection and Analysis Using Deep Learning Models. In *2025 3rd IEEE International Conference on Industrial Electronics: Developments & Applications (ICIDeA)* (pp. 1-6). IEEE. [CrossRef]

[6] Ma, L., Yu, Q., Yu, H., & Zhang, J. (2023). Maize leaf disease identification based on yolov5n algorithm incorporating attention mechanism. *Agronomy, 13*(2), 521. [CrossRef]

[7] Liu, J., He, C., Jiang, Y., Wang, M., Ye, Z., & He, M. (2024). A high-precision identification method for maize leaf diseases and pests based on LFMNet under complex backgrounds. *Plants, 13*(13), 1827. [CrossRef]

[8] Sun, X., & Huo, H. (2025). Corn leaf disease recognition based on improved EfficientNet. *IET Image Processing, 19*(1), e13288. [CrossRef]

[9] Santhi, S., Srinivasan, T., & RS, S. G. (2025, March). Smart Agriculture: An AI Based Solution for Early Corn Leaf Disease Detection Using Deep Learning. In *2025 International Conference on Machine Learning and Autonomous Systems (ICMLAS)* (pp. 648-654). IEEE. [CrossRef]

[10] Thorat, A., Kumari, S., & Valakunde, N. D. (2017, December). An IoT based smart solution for leaf disease detection. In *2017 international conference on big data, IoT and data science (BID)* (pp. 193-198). IEEE. [CrossRef]

[11] Sambana, B., Nnadi, H. S., Wajid, M. A., Fidelia, N. O., Camacho-Zuñiga, C., Ajuzie, H. D., & Onyema, E. M. (2025). An efficient plant disease detection using transfer learning approach. *Scientific Reports, 15*(1), 19082. [CrossRef]

[12] Upadhyay, A., Chandel, N. S., Singh, K. P., Chakraborty, S. K., Nandede, B. M., Kumar, M., Subeesh, A., Upendar, K., Salem, A., & Elbeltagi, A. (2025). Deep learning and computer vision in plant disease detection: a comprehensive review of techniques, models, and trends in precision agriculture. *Artificial Intelligence Review, 58*(3), 92. [CrossRef]

[13] Rani, R., Sahoo, J., Bellamkonda, S., Kumar, S., & Pippal, S. K. (2023). Role of artificial intelligence in agriculture: An analysis and advancements with focus on plant diseases. *IEEE Access, 11*, 137999–138019. [CrossRef]

[14] Lin, Z., Mu, S., Shi, A., Pang, C., Sun, X., & others. (2018). A novel method of maize leaf disease image identification based on a multichannel convolutional neural network. *Transactions of the ASABE, 61*(5), 1461–1474.

[15] Zhou, G., Zhang, W., Chen, A., He, M., & Ma, X. (2019). Rapid detection of rice disease based on FCM-KM and faster R-CNN fusion. *IEEE Access, 7*, 143190–143206. [CrossRef]

[16] Mwebaze, E., Gebru, T., Frome, A., Nsumba, S., & Tusubira, J. (2019). iCassava 2019 fine-grained visual categorization challenge. *arXiv preprint arXiv:1908.02900*.

[17] Jiang, P., Chen, Y., Liu, B., He, D., & Liang, C. (2019). Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks. *IEEE Access, 7*, 59069–59080. [CrossRef]

[18] Gao, A., Geng, A., Song, Y., Ren, L., Zhang, Y., & Han, X. (2023). Detection of maize leaf diseases using improved MobileNet V3-small. *International Journal of Agricultural and Biological Engineering, 16*(3), 225-232. [CrossRef]

[19] Ferentinos, K. P. (2018). Deep learning models for plant disease detection and diagnosis. *Computers and electronics in agriculture, 145*, 311-318. [CrossRef]

[20] Ma, J., Du, K., Zhang, L., Zheng, F., Chu, J., & Sun, Z. (2017). A segmentation method for greenhouse vegetable foliar disease spots images using color information and region growing. *Computers and Electronics in Agriculture, 142*, 110-117. [CrossRef]

[21] Khan, M. A., Akram, T., Sharif, M., Awais, M., Javed, K., Ali, H., & Saba, T. (2018). CCDF: Automatic system for segmentation and recognition of fruit crops diseases based on correlation coefficient and deep CNN features. *Computers and Electronics in Agriculture, 155*, 220–236. [CrossRef]

[22] Lin, Z., Mu, S., Huang, F., Mateen, K. A., Wang, M., Gao, W., & Jia, J. (2019). A unified matrix-based convolutional neural network for fine-grained image classification of wheat leaf diseases. *IEEE Access, 7*, 11570–11590. [CrossRef]

[23] Zhang, S., Huang, W., & Zhang, C. (2019). Three-channel convolutional neural networks for vegetable leaf disease recognition. *Cognitive Systems Research, 53*, 31–41. [CrossRef]

[24] Ma, J., Du, K., Zheng, F., Zhang, L., Gong, Z., & Sun, Z. (2018). A recognition method for cucumber

diseases using leaf symptom images based on deep convolutional neural network. *Computers and Electronics in Agriculture, 154*, 18–24. [CrossRef]

[25] Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., & Stefanovic, D. (2016). Deep neural networks based recognition of plant diseases by leaf image classification. *Computational Intelligence and Neuroscience, 2016*(1), 3289801. [CrossRef]

[26] Singh, D., Rana, A., Gupta, A., Sharma, R., & Kukreja, V. (2023). An enhanced CNN-LSTM based hybrid deep learning model for corn leaf eye spot disease classification. In *2023 IEEE 12th International Conference on Communication Systems and Network Technologies (CSNT)* (pp. 147–151). IEEE. [CrossRef]

[27] Pai, P., Amutha, S., Basthikodi, M., Ahamed Shafeeq, B. M., Chaitra, K. M., & Gurpur, A. P. (2025). A twin CNN-based framework for optimized rice leaf disease classification with feature fusion. *Journal of Big Data, 12*(1), 89. [CrossRef]

[28] Maglogiannis, I., Papadopoulos, H., & Iliadis, L. (2021). *Artificial intelligence applications and innovations*. Springer. [CrossRef]

[29] Singh, T., Bhadwaj, H., Verma, L., Navadia, N. R., Singh, D., Sakalle, A., & Bhardwaj, A. (2022). Applications of AI in agriculture. *Challenges and opportunities for deep learning applications in industry 4.0, 181*.

[30] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[31] Ouppaphan, P. (2017, November). Corn disease identification from leaf images using convolutional neural networks. In *2017 21st International computer science and engineering conference (ICSEC)* (pp. 1-5). IEEE. [CrossRef]

[32] Pacal, I., & Işik, G. (2025). Utilizing convolutional neural networks and vision transformers for precise corn leaf disease identification. *Neural Computing and Applications, 37*(4), 2479–2496. [CrossRef]

[33] Amin, H., Darwish, A., Hassanien, A. E., & Soliman, M. (2022). End-to-end deep learning model for corn leaf disease classification. *Ieee Access, 10*, 31103–31115. [CrossRef]

[34] Balaska, V., Adamidou, Z., Vryzas, Z., & Gasteratos, A. (2023). Sustainable crop protection via robotics and artificial intelligence solutions. *Machines, 11*(8), 774. [CrossRef]

[35] Reda, M., Suwwan, R., Alkafri, S., Rashed, Y., & Shanableh, T. (2022). Agroaid: A mobile app system for visual classification of plant species and diseases using deep learning and tensorflow lite. *Informatics, 9*(3), 55. [CrossRef]

[36] Bhujel, A., Kim, N. E., Arulmozhi, E., Basak, J. K., & Kim, H. T. (2022). A lightweight attention-based convolutional neural networks for tomato leaf disease classification. *Agriculture, 12*(2), 228. [CrossRef]

[37] Barbedo, J. G. A. (2018). Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification. *Computers and electronics in agriculture, 153*, 46-53. [CrossRef]

[38] Song, K., Sun, X. Y., & Ji, J. W. (2007). Corn leaf disease recognition based on support vector machine method. *Transactions of the CSAE, 23*(1), 155–157.

[39] Dash, A., Sethy, P. K., & Behera, S. K. (2023). Maize disease identification based on optimized support vector machine using deep feature of DenseNet201. *Journal of Agriculture and Food Research, 14*, 100824. [CrossRef]

[40] Daneshwari, A. N., & Basavaraju, D. R. (2022). Corn leaf image classification based on machine learning techniques for accurate leaf disease detection. *International Journal of Electrical and Computer Engineering, 12*(3), 2509. [CrossRef]

[41] Appalanaidu, M. V., & Kumaravelan, G. (2024). Classification of Plant Disease using Machine Learning Algorithms. In *2024 Sixth International Conference on Computational Intelligence and Communication Technologies (CCICT)* (pp. 1–7). IEEE. [CrossRef]

[42] Priyaradhikadevi, T., Mohan, R., Ragupathi, T., Prasanna, S., Madhan, K., & Ananthi, R. (2023). Leaf disease detection using machine learning algorithm. In *2023 Eighth International Conference on Science Technology Engineering and Mathematics (ICONSTEM)* (pp. 1–5). IEEE. [CrossRef]

[43] Jaisakthi, S. M., Mirunalini, P., & Thenmozhi, D. (2019, February). Grape leaf disease identification using machine learning techniques. In *2019 International conference on computational intelligence in data science (ICCIDS)* (pp. 1-6). IEEE. [CrossRef]

[44] Sami, A. K., & Al-Dulaimi, K. (2024). Plant Leaf Diseases Classification Using SVMs, XGBoost, and CNNs Based Texture Features. In *2024 IEEE International Conference on Artificial Intelligence in Engineering and Technology (IICAIET)* (pp. 354–359). IEEE. [CrossRef]

[45] Lokhande, N., Thool, V., & Vikhe, P. (2024). Comparative analysis of different plant leaf disease classification and detection using CNN. In *2024 International Conference on Recent Innovation in Smart and Sustainable Technology (ICRISST)* (pp. 1–4). IEEE. [CrossRef]

[46] Reddy, A. M., & Shaik, S. S. (2022). An automatic plant disease detection system using deep learning technique. *Mathematical Statistician and Engineering Applications, 71*(3), 152–158. [CrossRef]

[47] Hassan, S. M., Maji, A. K., Jasiński, M., Leonowicz, Z., & Jasińska, E. (2021). Identification of plant-leaf diseases using CNN and transfer-learning approach. *Electronics, 10*(12), 1388. [CrossRef]

[48] Taunk, K., De, S., Verma, S., & Swetapadma, A.

(2019). A brief review of nearest neighbor algorithm for learning and classification. In *2019 international conference on intelligent computing and control systems (ICCS)* (pp. 1255–1260). IEEE. [CrossRef]

[49] Mienye, I. D., & Jere, N. (2024). A survey of decision trees: Concepts, algorithms, and applications. *IEEE Access, 12*, 86716–86727. [CrossRef]

[50] Suthaharan, S. (2016). Support vector machine. In *Machine learning models and algorithms for big data classification: thinking with examples for effective learning* (pp. 207-235). Boston, MA: Springer US. [CrossRef]

[51] Iranzad, R., & Liu, X. (2025). A review of random forest-based feature selection methods for data science education and applications. *International Journal of Data Science and Analytics, 20*(2), 197-211. [CrossRef]

[52] Aminifar, A., Shokri, M., Rabbi, F., Pun, V. K. I., & Lamo, Y. (2022). Extremely randomized trees with privacy preservation for distributed structured health data. *IEEE Access, 10*, 6010–6027. [CrossRef]

[53] Bentéjac, C., Csörgő, A., & Martínez-Muñoz, G. (2021). A comparative analysis of gradient boosting algorithms. *Artificial Intelligence Review, 54*(3), 1937–1967. [CrossRef]

[54] Catapang, J. K. (2022). Optimizing speed and accuracy trade-off in machine learning models via stochastic gradient descent approximation. In *2022 9th International Conference on Soft Computing & Machine Intelligence (ISCMI)* (pp. 124–128). IEEE. [CrossRef]

[55] Chengsheng, T., Huacheng, L., & Bing, X. (2017). AdaBoost typical Algorithm and its application research. *MATEC Web of Conferences, 139*, 00222. [CrossRef]

[56] Jayasooriya, R. I. L., & Arachchi, S. M. (2022). Diagnosis of bacterial leaf blight, brown spots, and leaf smut rice plant diseases using light GBM. *Int. J. Comput. Appl, 183*(48), 53–58.

[57] Takahashi, K., Yamamoto, K., Kuchiba, A., & Koyama, T. (2022). Confidence interval for micro-averaged F 1 and macro-averaged F 1 scores. *Applied Intelligence, 52*(5), 4961–4972. [CrossRef]

[58] Sharma, A., Singh, S. K., Kumar, S., Thakur, R., Gupta, B. B., & Arya, V. (2024). IoT-enabled smart farming with Industry 5.0. *Journal of High Speed Networks, 30*(3), 477-496. [CrossRef]

[59] Subedi, S. (2015). A review on important maize diseases and their management in Nepal. *Journal of Maize Research and Development, 1*(1), 28–52. [CrossRef]

[60] Putro, S. S., Syakur, M. A., Rochman, E. M. S., & Rachmad, A. (2023). Prediction of corn crop yield using backpropagation neural network algorithm. *AIP Conference Proceedings, 2679*(1), 020006. [CrossRef]

[61] Aboelenin, S., Elbasheer, F. A., Eltoukhy, M. M., El-Hady, W. M., & Hosny, K. M. (2025). A hybrid Framework for plant leaf disease detection and classification using convolutional neural networks and vision transformer. *Complex & Intelligent Systems, 11*(2), 142. [CrossRef]

[62] Arya, S., & Singh, R. (2019). An analysis of deep learning techniques for plant leaf disease detection. *International Journal of Computer Science and Information Security (IJCSIS), 17*(7), 73–80.

[63] Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016). Using deep learning for image-based plant disease detection. *Frontiers in plant science, 7*, 215232. [CrossRef]

[64] Nazari, L., & Zinati, Z. (2023). Gene expression classification for biomarker identification in maize subjected to various biotic stresses. *IEEE/ACM Transactions on Computational Biology and Bioinformatics, 20*(3), 2170–2176. [CrossRef]

[65] Li, P., Cao, W., Fang, H., Xu, S., Yin, S., Zhang, Y., ... & Yang, Z. (2017). Transcriptomic profiling of the maize (Zea mays L.) leaf response to abiotic stresses at the seedling stage. *Frontiers in plant science, 8*, 290. [CrossRef]

[66] Gadag, R. N., Bhat, J. S., Mukri, G., Gogoi, R., Suby, S. B., Das, A. K., ... & Shilpa, K. (2021). Resistance to biotic stress: theory and applications in maize breeding. In *Genomic designing for biotic stress resistant cereal crops* (pp. 129-175). Cham: Springer International Publishing. [CrossRef]

[67] Quan, S., Wang, J., Jia, Z., Xu, Q., & Yang, M. (2024). Real-time field disease identification based on a lightweight model. *Computers and Electronics in Agriculture, 226*, 109467. [CrossRef]

[68] Daphal, S. D., & Koli, S. M. (2023). Enhancing sugarcane disease classification with ensemble deep learning: A comparative study with transfer learning techniques. *Heliyon, 9*(8). [CrossRef]

[69] Ojo, M. O., & Zahid, A. (2023). Improving deep learning classifiers performance via preprocessing and class imbalance approaches in a plant disease detection pipeline. *Agronomy, 13*(3), 887. [CrossRef]

[70] Rodríguez-Lira, D. C., Córdova-Esparza, D. M., Álvarez-Alvarado, J. M., Terven, J., Romero-González, J. A., & Rodríguez-Reséndiz, J. (2024). Trends in machine and deep learning techniques for plant disease identification: a systematic review. *Agriculture, 14*(12), 2188. [CrossRef]

Dr. Prabhat Chandra Shrivastava received the PhD. degree in Electronics & Communication Engineering from Motilal Nehru National Institute of Technology Allahabad, Prayagraj, India in 2019. His research interests include Digital Signal Processing, Artificial Intelligence, Machine Learning and Architecture design of Digital Filter. (Email: pcshrivastava@allduniv.ac.in)

Mohammad Saheeb Masood received bachelor degree in the Department of Electronics and Communication from Faculty of Science, University of Allahabad, Old Katra, Prayagraj-211002, Uttar Pradesh, India, in 2025. His research interests include machine learning, deep learning, Internet of Things (IoT), and Computer Vision. (Email: saheebmasood@gmail.com)

Aman Kumar received bachelor degree in the Department of Electronics and Communication from Faculty of Science, University of Allahabad, Old Katra, Prayagraj-211002, Uttar Pradesh, India, in 2025. His research interests include machine learning, deep learning, Internet of Things (IoT), and Computer Vision. (Email: reachiaman@gmail.com)

Dr. Ramesh Kumar Bhukya received the B.Tech. degree in Electronics and Communication Engineering from Mahatma Gandhi Institute of Technology (MGIT) Gandipet, Hyderabad, Telangana, India, in 2008, the M.Tech. degree in Electronics and Communication Engineering from Motilal Nehru National Institute of Technology (MNNIT) Allahabad, Prayagraj, Uttar Pradesh, India in 2011, and the Ph.D. degree in Electronics and Electrical Engineering from Indian Institute of Technology Guwahati (IITG), Guwahati, Assam, India in 2019. His Ph.D. work focuses on Approaches for robust text-dependent speaker verification under degraded conditions using speech-specific knowledge from the perspective of practical application-oriented systems. After completing doctoral studies, he joined as an Assistant Professor in Electronics and Communication Engineering from the National Institute of Technology (NIT), Hamirpur, Himachal Pradesh, India, in 2019-2020. In 2020, he joined in the Department of Electronics and Communication Engineering at National Institute of Technology Andhra Pradesh (NIT-AP), Tadepalligudem, Andhra Pradesh, India. He is currently working as an Assistant Professor in the Department of Electronics and Communication Engineering from the Indian Institute of Information Technology Allahabad (IIIT-A), Prayagraj, Uttar Pradesh, India. His research interests are speech signal processing, speaker verification, biomedical signal processing, machine learning and pattern recognition. (Email: rkbhukya@iiita.ac.in)

Dr. Harmanjit Kaur completed her M.Sc. Botany (Hons. School), Gold Medalist and Ph.D. (Plant Physiology) from Panjab University, Chandigarh. Her research field includes Plant Abiotic Stress Physiology, Agricultural Plant Sciences and Plant Stress Physiology. She is a life member of Indian Society for Plant Physiology (ISPP) since 2016. She has published around 20 research and review articles in various SCI and SCOPUS indexed international journals. She has also published 11 chapters and 1 book. She has presented her research work in several National and International conferences. She has 10 years of teaching and research experience at both college and University level and serves as a reviewer for several high-impact international journals. (Email: harmanjitkaur@allduniv.ac.in)

Rupesh Kumar received the PhD. degree in Department of Electrical Engineering from Indian Institute of Technology, Kanpur UP, India in 2018. Her research interests include Computer Vision: Video Processing, Spatio-Temporal analysis of Video, Motion based video applications, Pattern recognition, Feature detection and extraction, Machine Learning, Deep Learning, 3D Point Cloud, Image Segmentation. Large Language Model, Short Language Model. (Email: rupeshkumar@yamaha-motor-india.com)

Prof. (Dr.) Ir. Derrick Mario Denis received the PhD. degree in Agricultural Engineering, University of Allahabad, Prayagraj, India in 2004. His research interests include Remote Sensing of Evapotranspiration, Water footprints and Deficit irrigation, Water and Food Systems, Farming system-based water budgeting, and Modelling of Watersheds. (Email: derrickmdenis@gmail.com)