J. Integr. Sci. Technol. 2022, 10(1), 44-52 Article

Journal of Integrated
SCIENCE & TECHNOLOGY

An effective feature descriptor method to classify plant leaf diseases using Extreme Gradient Boost

A.Usha Ruby,* Chaithanya B.N., Swasthika Jain T.J., Smita Darandale, Sudarshana Kerenalli, Renuka Patil

Department of Computer Science & Engineering, GITAM School of Technology, GITAM University, Visakhapatnam, Andhra Pradesh 530045. India.

Received on: 28-Dec-2021, Accepted and Published on: 5-Mar-2022

ABSTRACT

Identifying plant leaf diseases will be highly difficult due to the difficulties in gathering lesion characteristics from a quickly changing atmosphere, imbalanced illumination reflection of the incoming light source, and numerous other factors. A practical strategy for classifying plant leaf diseases is provided in this research. Using HSV, HU moments, and color histograms, we first created a leaf feature improvement framework that can enhance leaf characteristics in a complicated environment. Then, to increase feature classification capacity, a competent extreme boost method is modelled. Batch normalization is used to avoid network overfitting while also improving the model's resilience. The plant leaf disease feature improvement approach is favorable to boosting the efficiency of the XGBoost classification, as demonstrated in studies from various perspectives. For plant leaf disease photos obtained in the natural environment, our technique displays significant resilience, serving as a benchmark for the intelligent categorization of additional plant leaf diseases.

Keywords: Image enhancement, image segmentation, plant diseases, artificial intelligence, XGBoost.

INTRODUCTION

Plants have given humanity a wide range of medications and food sources. Plants must be safeguarded at all phases of their lives since they are crucial components of human existence. Agriculture crops are recognized to be among the most beneficial plant species. These crops provide food for even more than 70% of our nation's population. Crop production, on the other hand, might be impeded by illnesses that aren't apparent to the human eye. By analyzing the afflicted leaf, plant diseases can be recognized. One option is to manually detect these illnesses with the help of a botanic specialist, however manual detection of leaf diseases is a tedious and time-consuming operation. As a result, an autonomous technique is needed. Plant leaf disease is the sole important element that contributes to a decrease in plant generation quality and quantity. Identification and categorization of plant leaf diseases are key responsibilities for improving plant efficiency and profitability.[1] The ability to identify and classify the standard of agricultural goods is becoming increasingly widespread as agricultural and image processing technology evolves.[2] Diagnosis and detection of agricultural pest and diseases have become object of current farming study, but the outcomes have been astounding. The leaf has several advantages over flowers and fruits throughout the year. Plant pathology is the study of plant diseases, their causes, and control and management procedures. Numerous studies have shown that plant diseases reduce the quality of agricultural products. Diseases are natural changes in a plant's state that affect or stop vital processes such as photosynthesis, transpiration, pollination, fertilization, germination, and so on. Many diseases are caused by infections such as fungi, bacteria, and viruses, as well as poor environmental conditions. It is vital to make good diagnoses and treatments for plant disease observed during vegetative growth[3] to enhance the quality and productivity of the primary harvests. Specialists must go into the farm to diagnose agricultural diseases, which may be time-consuming and labor-intensive. Simultaneously, the effect of numerous external environmental variables and subjective elements can quickly lead to subjective lack of judgment in the identification and treatment of various diseases.[4] To classify leaf images, machine learning technique can be used.[5] Each image contains valuable data that may be extracted using a computational model. Image segmentation is the process of

Corresponding Author name: Dr. A Usha Ruby
Tel: xx
Email: uruby@gitam.edu

Cite as: *J. Integr. Sci. Technol., 2022, 10(1), 44-52.*

©ScienceIN ISSN: 2321-4635 http://pubs.iscience.in/jist

breaking down a large image into tiny, more useful sections. It is significant to observe that it can be defined as the identification and classification of a particular selected area.[6]

In this paper, an efficient method for classifying plant diseases is implemented. The experiments are carried out on the plant village dataset. Preprocessing, spot segmentation, feature extraction, and classification are the next steps. The leaf spots are segmented in the first step for color extraction. Following that, using two feature descriptors such as hu moments and color histogram, features are extracted from images using global feature descriptors. The extracted features are then divided into training and testing datasets in an 80:20 ratio and saved in Hierarchical data format version 5.[7] Logistic Regression, Linear Discriminant Analysis, K Nearest Neighbors, CART, Random Forest, Naive Bayes, Support Vector Machine, and Extreme gradient boosting are the machine learning models used to train. Furthermore, good preprocessing always resulted in significant features that later made substantial classification accuracy. Extreme gradient boosting (XGBoost) is a decision tree algorithm augmentation proposed by Chen et al.[8] in 2016. This excels the previous gradient boosted tree approach as well as other supervised machine learning approaches.

The following are the article's key contributions:

(i) The plant leaf dataset preprocessed and classified as healthy and diseased

(ii) Comparative evaluations of current approaches are carried out to determine their benefits and drawbacks.

(iii) Different performance indicators for evaluating the efficacy of illness prediction systems are also discussed.

Related Work

Manual disease identification or detection in plants is costly, time intensive, and needs specialized specialists. As a result, there has been a lot of study into developing automated procedures that are dependable, accurate, and cost-effective. Because of recent advances in machine learning, a variety of strategies have been proposed to address this problem, all of which have shown excellent results. Jiang x et al presents study to detect leaf traits due to disease and pest.[9] Here Successive Projection Algorithm (SPA) has been developed to extract sensitive features of severity disease and mangrove pest of leaves. Study helps for quick management of mangrove health conditions by observing infections and mangrove pests at the land.[9] K. Suganya Devia, et al. the previous study of groundnut leaf disease was based on an image processing methodology that detects and classifies the disease automatically. Here the author has proposed a robust system using the KNN classifier and Histogram on Oriented Gradient for identifying and classifying groundnut leaf disease.[10] Zhang K. et al. this study first developed an image dataset of soybean leaf disease. Next, a multi-feature fusion faster R-CNN model has been designed to separate healthy and diseased leaves, addressing the issues.[11] Qian Yang et.althe study focused on exploring the role of melatonin in leaf disease. Furthermore, it analyzed the saponin attentiveness to examine the correctness of melatonin use in agricultural practice.[12] Hamdani H. et al. the author studies the disease of oil palm plants. They proposed a novel system for leaf disease of oil palm. The leaf is classified into two types: diseased and healthy leaf. Here feature extraction and clustering approaches are used to address the issues.[13] Xiao Chen et al. the author proposed a novel work for identifying infection of tomato leaf. In first stage Binary Wavelet Transform (BWT) collective with Retinex remove outlier and retained significant texture statistics. Further tomato leaves were distinguished using the Artificial Bee Colony algorithm. Tested on 8616 images and concluded with accuracy 89%.[14] Zhencun Jiang et al. the study presents the two types of infections of wheat leaf and three categories of diseases of rice leaf. Experimental work was performed on 40 images and targeted to progress model (VGG16) Visual-Geometry-Group-Network-16.[15] Wang C. et al. in this article, author proposed a model to classify cucumber leaf disease sternness in compound backgrounds. First, DeepLabV3+ is used to segment the leaves from compound circumstances using U-Net and DeepLabV3+. In second step, (U-Net) is applied in the direction of distinct the affected plants to find syndrome. Experimental results concluded with 93.27% accuracy of the model.[16] Gensheng Hu et al. identified a tea leaf disease. They designed a model using (CNN) Convolutional Neural Network and multi scale feature extraction to remove image attributes of different tea leaf diseases.[17] Zhang et al. recognized a Soybean leaf disease using bp algorithm. To control the output error, the learning rate is adjusted dynamically.[11,18] Ali H. et al. this article projected a system to recognize and categories infection of major citrus fruit. ΔE color difference algorithm is applied to distinguish affected leaves, and then it is classified using textural attributes and color histogram.[19] Zhang S. et al. proposed a method based on K-means clustering, clustering method of super-pixel, and PHOG algorithms for IOT based plant leaf disease.[20] Lv Jidong et al. developed a two-stage method to separate severity diseases apple fruits and leaves. In the first stage, to check the growth, the apple images are classified. In the second stage, OTSU dynamic threshold segmentation method is used to compare the apple images.[21] Francis et al., proposes approach grounded on an image-processing related to infection of leaf recognition for groundnut yields. This study directed to educate and highlight the farmer about the overwhelming effect of these diseases.[22] Ashourloo et. al, developed an SDI Spectral Disease Index which is used to identify the phases of Wheat leaf disease with different severity levels. For experimental work infected leaves reflectance spectra and disease severity levels were measured using spectro radiometer. Here pure spectra have been analyzed and developed a new methodology to search the wavelengths which is most sensitive to disease.[23] Lv. M et al. proposed the disease recognition method for maize leaf. In the first step, framework was designed for maize leaf feature enrichment. Next, based on Alexnet architecture, the new neural network is designed to improve the proficiency of feature extraction.[24] Chouhan S.S. et al., study focus on the automatic process compared with existing methods where human interaction is required, which is time-consuming and expensive. The authors introduced the automated technique for identifying and classifying plant leaf infections. This technique is based on a radial basis function neural network. Experimental work demonstrates that the proposed method gives better accuracy and speed.[25] Liu, B. et al., proposed a Leaf GAN model based on Generative Adversarial Network. It is for training identification model for different kinds of diseases of a grape leaf. Real and fake

infected images were discriminate against using feature extraction. Results concluded that the proposed model proficiently found the infected grape leaf images as compared with other models.[26] Zhou C. et al., these papers researched tomato leaf diseases to help the farmer to identify disease in early stage. For determining the infection, a restructured residual dense network model has been proposed. The results demonstrate that the proposed model achieved 95% accuracy on the tomato dataset.[27] Ashourloo, D. et al., the paper explored wheat leaf diseases methods based on vsupport vector regression, partial least sqaure regression, and Guassian process regression. A non-imaging spectroradiometer was used for measuring symptoms of infected and noninfected leaves.[28] Yuan Y. et al., the paper proposed a neural network for crop infection leaf segmentation. It entails region infection segmentation network and region infection identification. This method combines the three-level convolution neural network model. Results represent that the proposed method has higher segmentation accuracy.[29] Jiang, P. et al., the study focuses on disease detection in apple leaves. The deep learning approach has been proposed using CNNs neural network. The proposed novel INAR-SSD model gives a high speed and accuracy for diagnosing apple leaf diseases in the early stage.[30] Wu Q. et al., proposed a novel method of data augmentation using Generative adversarial Networks for tomato leaf disease detection. The results demonstrate that the proposed method constructs the data near to authentic images.[31] Pham T.N. et al., This paper used the ANN (artificial neural network) approach to identify the disease of plant leaves in the early stage. Pre-processed data using contrast enhancement method and then the diseased blobs are segmented for better results.[32]

PROPOSED SYSTEM

We choose the finest feature extraction and classification approach from the reviewed list to set the foundation for our suggested solution. According to the findings, the accuracy is the most important aspect to consider in this study. The key reason for selecting this study is that (Mohanty et al.; 2019)[33] employed machine learning for classification, that classifies village plant datasets into healthy and diseased utilizing hu moments and color histogram feature descriptors. After analyzing and testing many features extraction and classification strategies utilizing various machine learning techniques, XGBoost proved the efficiency in classifying the leaf diseases. The suggested model's architecture is depicted in Figure 1.

The recommended model's five steps are as follows: i.e., dataset selection, data preprocessing, data classification, data visualization and evaluation. Following figure 2 explains about the different stages of the proposed model.

Dataset Selection

Obtaining datasets containing images of plants in the actual world instead of in a controlled laboratory environment is difficult and costly; thus, this evaluation adapted a dataset that is available publicly and has been used by different researchers [https://www.kaggle.com abdallahalidev/ plantvillage-dataset] that includes images of different plants affected by multiple diseases. The dataset has a total of 38 class labels, with each class label indicating a plant-disease relationship. The dataset has previously been classified into three parts: color, segmented, and grayscale. Only the color form of the photographs is included in the analysis since the color variant of the images produced excellent results in the study. The Dataset sample is shown in table 1.

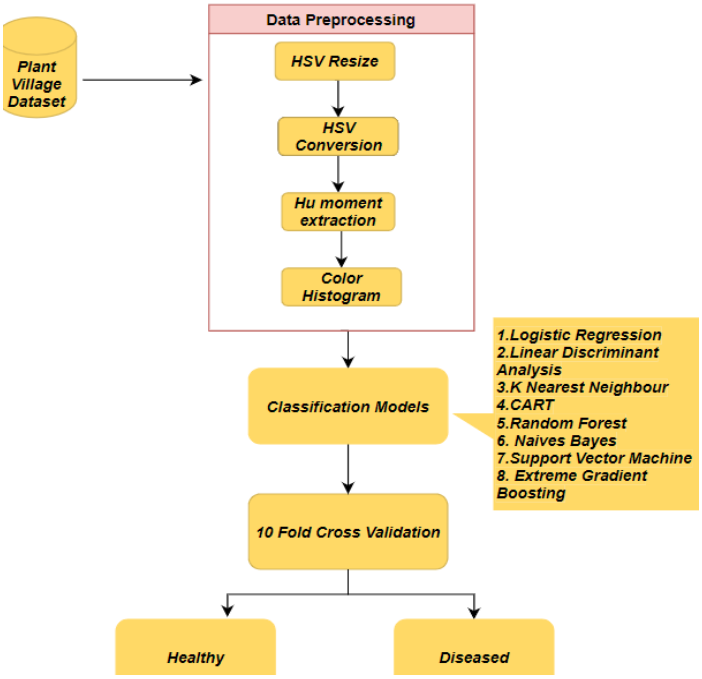
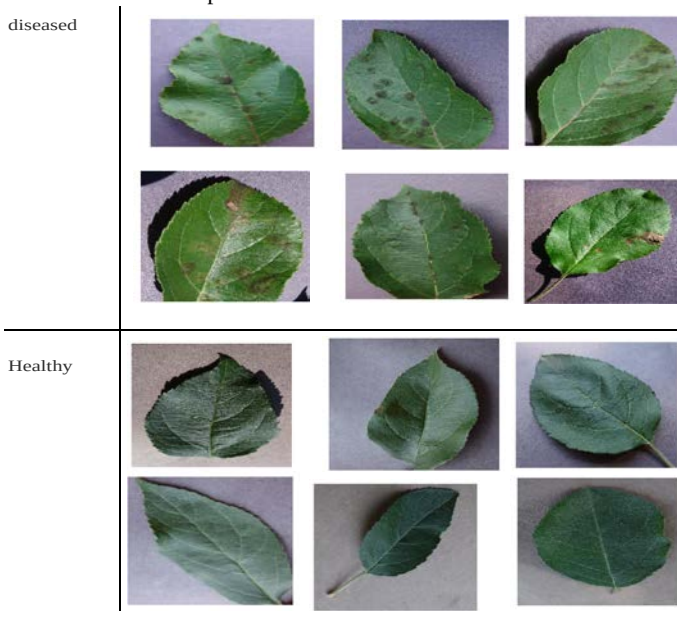


Figure 1: Illustrates the architecture of the suggested model.

Figure 2: Block diagram stages of proposed model.

Table 1: Dataset sample



Data Pre-Processing

The imagery database was set up such that each folder corresponded to a label/class of photographs with the structure <plant name>< disease name>. As a result, the first logical step was to delete all image folders that included just photographs of healthy or unhealthy plants. With an 80/20 split, the data repository was partitioned into two independent train and test groups. The 80/20 ratio was chosen since it generated the best results in the study. Therefore, around 1280 images were loaded into the train directory and 320 images into the Val(test) directory, which is more than enough for transfer learning.

Preprocessing step includes

Converting each image to RGB from BGR format

Conversion of BGR image to RGB and conversely can be done for a variety of factors, one of which is that different image processing libraries have varying pixel groupings. The cvtColor() function may be used to translate a BGR image to RGB and conversely.

Conversion to HSV image format from RGB

HSV filtration works by removing noise from the hue, saturation, and intensity value elements of a dataset image selected. For de-noising, the HSV filtering system divides a color plant image into its hue, saturation, and intensity value elements. With a color's R, G, and B variables, its H, S, and I value are derived in equation 1,2 and 3 respectively.

$$H = \begin{cases} \theta \\ 360 - \theta \end{cases} \quad if\ B < G\ if\ B > G \tag{1}$$

$$Where\ , \theta = \cos -1\ \frac{\frac{1}{2}(R-G)+(R-B)}{(R-G)^2+(R-B)(G-B)}$$

$$S = 1 - \frac{3}{R+G+B}[min(R,G,B)] \tag{2}$$

$$I = \frac{1}{3}(R+G+B) \tag{3}$$

The equation 1,2 and 3 provided above facilitate for the conversion of an RGB plant images to its HSV format of plant images.

Image Segmentation

The Segmentation technique is used to extract green and brown colors since it can effectively construct pixel-wise filters for images in the dataset, allowing us to separate the foreground from the background.

IMAGE FEATURES DESCRIPTORS

Hu moments

The ever-increasing volume of digital images need efficient retrieval. However, the dominant color descriptor has seen widespread application in image processing. The same hue in nature may seem differently due to the impact of lighting and other variables. The human eye is typically more responsive to zones of continuous color, and these zones of uniformity are frequently used to detect images. As a result, the approach suggested in this study first uses the text on pattern to identify and retrieve the constant region of a plant image, and then computes the dominant color descriptor attribute on the pixels in this persistent zone. Furthermore, the Hu moments feature's translation and rotation invariance is used to recover shape information in the same consistent zone of the plant image.[34]

Color Histogram

The image histogram is the most essential tool for developing spot behavior on plant leaf images. The digital image histogram, which is a plot or graph, represents the couple of instances of each grey level. As a result, a histogram is a one-dimensional feature with a range of 0 to the image's pixel count. Histogram equalization, also known as histogram flattening, was among the most significant nonlinear point processes. The digital image f's histogram Hf is a plot or graph that shows the number of examples of each grey level in f. As a result, Hf is a one-dimensional feature with domains 0,...,K-1 and a potential range of 0 to the pixel count of the image.

To reach the optimum balance among classification performance and prediction accuracy, many models were developed. Logistic Regression, Linear Discriminant Analysis, K Neighbors Classifier, CART, Random Forest Classifier, Gaussian NB, and SVM are utilized as classifiers in this finding. Based on previous research, Random Forest outperformed other classic machine learning classifiers, and XGBoost, the suggested model applied in this area, is used to determine if there is any advantage over other approaches. To discover the best set of parameters obtained by randomized Search, 10-fold cross-validation is used.

1. Logistic Regression

It's a strategy for creating a linear relationship between the dependent and independent variables 'x' and 'z.' As seen in equation 4, the model works the best line for guessing the value of z for a specific value of x. The linear regression hypothesis function in equation 4 determines the best regression fit line by picking the appropriate intercept 'm' and coefficient 'c' values.

$$\boldsymbol{z = m * x + c} \tag{4}$$

The cost function in equation 5, defines an attempt to minimize the Root Mean Squared Error (RMSE) between the real values of z, 'z_act,' and the estimated values, 'z_est.'

$$\boldsymbol{J = \frac{1}{n}\sum_{i=1}^{n}(z\_est - z\_act)^2} \tag{5}$$

2. Linear Discriminant Analysis

By reflecting the input data to a linear subspace containing the directions that improve class segregation, Linear Discriminant Analysis (LDA) can be utilized to accomplish supervised dimensionality reduction. LDA estimates the likelihood that a given set of inputs will correspond to each class. A forecast is prepared for the output class with the highest likelihood.

Linear Discriminant Analysis (LDA), also termed as Normal Discriminant Analysis (NDA) or Discriminant Function Analysis (DFA), is a matrix factorization technique used to solve supervised classification problems. This is used to represent class differences, such as splitting two or more classes. It's utilized to project items from a higher level to a lower dimension space.

3. The k-NN Classifier

The k-nearest neighbor method is a part of the tired beginner group. This causes a delay in the process of developing the learning

model until it is tested. If the difference between the actual and estimated labels is less than the threshold value 'k,' a test instance is assigned to a specific class. The nearest neighbor error rate is provided in equation 6 and is defined as the likel

ihood that the point to be measured x varies from the class c of the nearest - neighbor point x'.

$$p(error) = 1 - \sum_{c \in y} P(c|x) * P(c|x') \quad (6)$$

After obtaining the nearest neighbors list, the test instance is labelled using the majority class of its nearest neighbors. The algorithm for the k-NN is given below in Table 2.

Table 2: Basic k-NN Algorithm

Algorithm: k-Nearest Neighbour Algorithm

Input: 'D'-Data set having 'n' records

Output: 'y'- Class Label

1. Let 'k' be the no. of nearest neighbors
2. For every test instances z(x',y') do
   a. Calculate the dist(x', x), for each (x,y)ε D
   b. Select subset D_z of D containing the closest training examples to z
   c. $y' = \underbrace{Argmax}_{v} \sum_{(xi,yi) \in Dz} I(v = yi)$
3. End for

4. CART

The CART method separates the training dataset repeatedly to create subsets that are as simple as feasible to a specific target class. Each node in the tree defines a specific set of records T that has been divided using a feature test. T is then separated into two subsets, each leading to the tree's left and right nodes. The recursive approach for inducing the decision tree divide phase considers all possible splits for each feature and attempts to select the optimal one based on a quality metric: the condition for division: E = {A_1…,A_m,C}, where A_j is the characteristics and C is the target class. Impurity measures are frequently used to determine the best split. The parent node's impurity must be reduced because of the split. Let (E1, E2,...,Ek) denote a split caused on the collection of records E using a splitting criterion based on the impurity measure $I(.)$, ∇ is the impurity gain as shown in equation 7.

$$\nabla = I(E) - \sum_{i=1}^{k\infty} \frac{||E_i||}{||E||} * I(E) \quad (7)$$

CART use the Gini index as Standard impurity measures, which is defined for the set E as follows equation 8:

$$Gini(E) = 1 - \sum_{j=1}^{Q} p^2{}_j \quad (8)$$

5. Random Forest Classifier

The RF Classifier is a supervised ensemble classifier (Suresh et al., 2019). It is a method of categorization that is scalable, adaptive, and exact. By mixing predictions from several trees, it avoids the over-fitting problem. Random forests are more complicated, use more processing resources, and take longer to compute. Figure 3 depicts the stages required in developing the random forest approach.

1. Random models are generated from the supplied data set.
2. To gather predictive performance, build decision trees from such samples obtained.
3. Vote on all the expected outcomes using the voting method.
4. As the prediction accuracy result, use more popular forecast result.

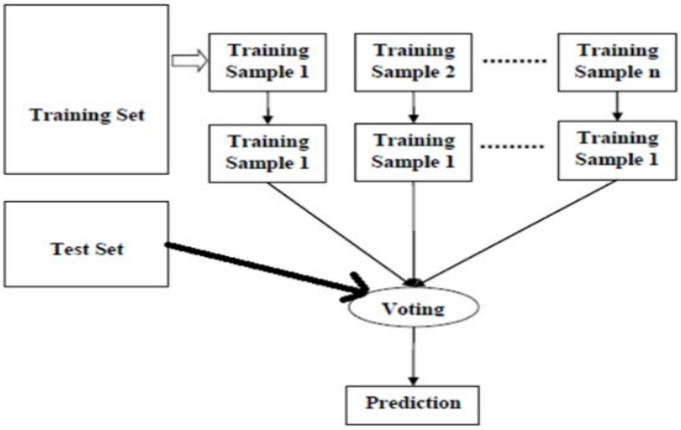


Figure 3: Basic architecture of RF algorithm.

The RF classifier turns many classification trees into a vector that is uniformly distributed among all trees in a forest. A random vector k is produced and spread throughout the whole forest in an RF approach, allowing each tree to form using the leaf image training dataset. When the random vector k is applied to the input vector x, a collection of tree-structured classifiers h(x, k), k = 1,...n, are produced. Equation 9 is used to determine the generalisation error, EG, where x and y are random vectors expressing probability in (x, y) space. The margin function calculates how much one output's average number of votes boosts the average number of votes for following outputs.

The margin function, mg(x,y), is defined in equation 10, where I() is an indicator function. The strength and correlation measures are used to measure individual classifier accuracy and confidence. A random forest with random features is formed by arbitrarily decide on a simple set of input variables on every node.

$$E_G = P_{x,y}(mg(x,y) < 0) \quad (9)$$

$$mg(x,y) = av_k * I((h_k(X) = y) - max_{j!=y} av_k * I((h_k(X) = j) \quad (10)$$

The learning system creates a classifier from the sample, then combines all the classifiers created by the different trials to create the final classifier. Every classifier keeps track a vote for the class to which an occurrence fits, and the instance is assigned to the class with the most votes.

6. The Naive Bayes Classifier

The Naive Bayes Classifier is dependent on the Bayes Rule, which says in equation 11.

$$P(Y/X) = \frac{P(X/Y) * P(Y)}{P(X)} \quad (11)$$

Rewrite equation 11 using X (input variables) and y (output variables) to make it easier to understand (output variable). Based on the supplied qualities X, this equation calculates the probability of y. The naive assumption is that the variables are independent given the class. It's possible to rewrite equation 11 as equation 12. The goal of Naive Bayes is to choose the class 'y' with the highest

probability. Argmax is a simple function that discovers the argument that yields the maximum value for the target variable. We're aiming for the highest y value in this case, as shown in equation 13.

$$P(X/y) = P(x_1/y) * P(x_2/y) * \dots\dots\dots P(x_n/y) \quad (12)$$

$$y = ArgMax_y\left[P(y) * \pi_{i=1}^{n}\left(P(x_i/y)\right)\right] \quad (13)$$

7. Support vector Machine

The SVM technique increases the disparity between the training set and the decision boundary. SVM generates a hyperplane with the greatest difference between true and false examples feasible. Kernel techniques prevent moving data into a higher-dimensional feature vector while maintaining the features and runtime required to build the classifier's prediction model, avoiding the quadratic memory growth problem. Equation 14 defines the hyperplane for the dataset of pairings of D: D = (xi, yj) | xi RP, yi +1,-1, I = 1, 2, 3... In this case, is a unit vector. When classes are discrete, the hyperplane with the biggest boundary between the training points for classes 1 and -1 may be obtained by using the function yif(x)> 0 for all I values. The optimization formula for the above issue is found in equation 15. As a result, a margin M unit away from the hyperplane forms on both sides.

$$\{x: f(x) = X^T * \beta + \beta_0 = 0\} \quad (14)$$

$$\text{Maximize } M_{\beta,\beta_0,\|\beta\|=1} \text{ , subject to } y_i \begin{matrix}(X^T * \beta + \beta_0) \geq M \\ i = 1,2,3,\dots N\end{matrix} \quad (15)$$

8. eXtreme Gradient Boosting Algorithm

The XGB is an open-source algorithm capable of handling a extensive variety of data irregularities. A more efficient gradient boosting procedure is the XGBoost method. Using weaker models, a gradient boosting approach predicts the target variable for a basic data set. After that, the data are pooled to produce a good estimate of the target variable. The minimization of XGB's objective function is defined by equation 16.

$$L^t = \sum_{i=1}^{n} l(y_{act}, {y_{pred,i}}^{(t-1)} + f_t\,(x_i)) + \Omega(f_t) \quad (16)$$

A convex differentiable loss function is defined in equation 16. It is a calculated difference between the prediction y_{pred} and the target y_{act}. Add f_t greedily to l, as illustrated in the equation 16 $\Omega(f_t)$, to reduce the objective function, L^t. At each iteration, as illustrated in equation 16, the goal function L^t is reduced. Equation 17 shows a reduced objective function for minimizing at step t.

$$\widehat{L^t} = \sum_{i=1}^{n}\left[g_i f_t(x_i) + \frac{1}{2} h_i {f^2}_t(x_i)\right] + \Omega(f_t) \quad (17)$$

Using well-known approaches, the sum of simple quadratic functions with a single parameter, L^t, is diminished. The learning model construction is the as follows. The simplest method is to start with a single root that contains all of the training instances. Iterate through all features and values per feature vector, considering each split loss reduction possibility:

$$\text{Gain} = loss_{parent} - (loss_{Left\ child} + loss_{Right\ Child}) \quad (18)$$

The equation 18 is about the gain of this best split must be positive and greater than the min split gain parameter, or the branch will stop growing. Chen et al. proposed extreme gradient boosting (XGBoost) as an extension of the decision tree algorithm in 2016.[8] This significantly outperforms other supervised learning algorithms, including the original gradient boosted tree algorithm. The objective function is represented by equation 19.

$$(\theta) = L(\theta) + \Omega(\theta) \quad (19)$$

where $L(\theta)$ and $\Omega(\theta)$ represent learning loss and normalization, i.e., outfit computational complexity, respectively. The classifier creates several weak learners and groups them together. The algorithm begins by constructing a tree based on a feature. It generates another tree using the objective function that improves on the errors or residuals of the previous tree. During the construction of a new tree, the error or residual is calculated and minimized using gradient descent. Tree pruning is done greedily in each split based on accuracy gain. Depth-first tree pruning, and gradient loss minimization speed up the decision tree construction process, resulting in faster execution and higher accuracy.

EVALUATION METRICS

We have measured our proposed techniques using various evaluation metrics such as:

1. Confusion Matrix
2. Accuracy
3. Precision or Sensitivity
4. Recall
5. F1 Score

Confusion Matrix

The confusion parameter is used to quantify the main assessment criteria. The number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) are the components of the confusion matrix (FN). To assess the outcomes, the accuracy, recall, precision, and F-1 Score were determined. The concepts listed above have the following definitions:

There are four methods for determining whether the forecasts are accurate:

1. True Positive (TP): This measure tracks the number of accurately anticipated positive cases.

2. False Positive: This measure lists the number of negative records that were wrongly projected to be positive.

3. True Negative (TN): This statistic counts the number of accurately anticipated negative situations.

4. False Negative (FN): counts the number of positive records that should be negative.

Accuracy

Accuracy is calculated as the ratio of the total number of accurately predicted healthy images to the total number of images. It's shown in the following equation 20.

$$\text{Accuracy} = \frac{(\text{True Positive} + \text{True Negative})}{(\text{Total number of images})} \quad (20)$$

Precision or Sensitivity

The ratio of true positive counts of leaf pictures to the total number of true positive and false positive leaf images is used to determine precision. Equation 21 is used to express accuracy.

$$p = \frac{(True\ Positive)}{(True\ Posiitive + False\ Positive)} \quad (21)$$

Recall

Recall is calculated by dividing the total number of true positive and false negative pictures by the sum of true positive and false negative images. The following equation 22 explains it.

$$Recall = r = \frac{(True\ \ Positive)}{(True\ Positive + False\ Negative)} \quad (22)$$

F1-Score

The F1-Score is the fraction of true positive values of the images to the total of true positive and false positive values of the images. It is provided in the equation 23.

$$F1 - Score = \frac{2 * precision * recall}{(precision + recall)} \quad (23)$$

RESULTS AND DISCUSSIONS

The experimental findings achieved by using our approach are described in depth in this part, as well as the performance under various experimental settings. We conduct all our research crossways a wide variety of train-test set separations, particularly 80-20, to obtain a result of how our models will perform on new, unknown data, as well as to keep records of whether any of our strategies are overfitting (80 percent of the whole dataset used for training, and 20 percent for testing). The plant village dataset has numerous images with the same leaf (taken from different orientations), and we have mappings for 41,112 of the 54,306 images; and even after all those test-train partitions, we make sure that all images of the same leaf go either in training or testing set. Furthermore, for each trial, we estimate the mean precision, mean recall, mean F1score, and average accuracy during the whole training phase for 10-fold cross validation.

Comparing the various ML results on the data

Modern technologies, such as ML and DL algorithms, have been used to improve the recognition rate and accuracy of the findings. Numerous studies have been conducted in the aera of machine learning for plant leaf disease classification. The histogram of oriented gradients (HOG) is an element descriptor used in image pre-processing for object recognition. In this case, we're using two feature component descriptors: 1. hu moments 2. Color Histogram Hu moments are essentially utilized to determine the form of the leaves. A color histogram is used to illustrate the distribution of colors in an image. The labeled datasets are segregated into training and testing data. The feature vector is generated for the training dataset using Hu moment and color histogram. The generated feature vector is trained using different classifiers. The trained classifier is then given the feature vector for the testing data produced by feature extraction for prediction. Then, using feature extraction, labelled training datasets are converted into their corresponding feature vectors. These extracted feature vectors are stored as training datasets. The trained feature vectors are then trained using machine learning methods. The classification accuracy of the proposed method has been verified using various machine learning models. Figure 4 show the result for the classification accuracy and loss among all traditional algorithms random forest achieved higher classification accuracy of 96.4 and loss of 0.05 when compared with that of LR, LDA, KNN, C ART, NB, SVM algorithms. Figure 5 shows the box plot visualization for all compared algorithms.

```
LR: 0.917188 (0.020432)
LDA: 0.901563 (0.027994)
KNN: 0.921094 (0.024067)
CART: 0.915625 (0.019390)
RF: 0.964063 (0.015309)
NB: 0.855469 (0.021608)
SVM: 0.922656 (0.016919)
```

Figure 4: Accuracy and Loss

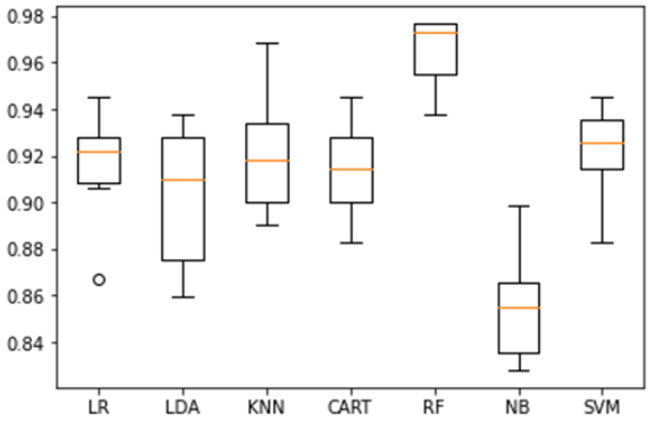


Figure 5: Box plot for different machine learning models

In an asymmetrical data gathering, precision and recall are necessary. Precision is a way of measuring how precise the final scale is and how closely it resembles the planned solution. The relevance of the findings is evaluated by recall. The lower the recall value, the lower the false-positive rates, and the higher the false-negative rates, the better the accuracy. The higher recall value is due to fewer false negative rates. When the number of false positives is lower, precision improves. Therefore, the precision accurately indicates the proximity and suitability of the result scale. Recall score also determines the quantity of relevant outcomes. For each trial, we estimate the mean precision, mean recall, mean F1score, and average accuracy during the whole training phase for 10-fold cross validation. Finally, the main aim of our work is to detect whether it is diseased or healthy leaf with the help of a XGBoost classifier which is as depicted in the Figure 6.

	precision	recall	f1-score	support
0	0.98	0.97	0.98	158
1	0.98	0.98	0.98	162
accuracy			0.98	320
macro avg	0.98	0.98	0.98	320
weighted avg	0.98	0.98	0.98	320

Figure 6: Classification report for XGB Classifier

The outcomes are as shown in figure 7 the confusion matrix. There are 158 positive class images and 162 negative class images among the 320 test images. The algorithm successfully predicted 154 positive images, yielding a true positive rate of 97.5 percent and a false positive rate of 2.5 percent. Furthermore, 159 of 162 negative class cases were properly identified, yielding a true negative rate of 98.15 percent and a false negative rate of 1.85 percent. When compared to current strategies, the proposed method outperformed them.

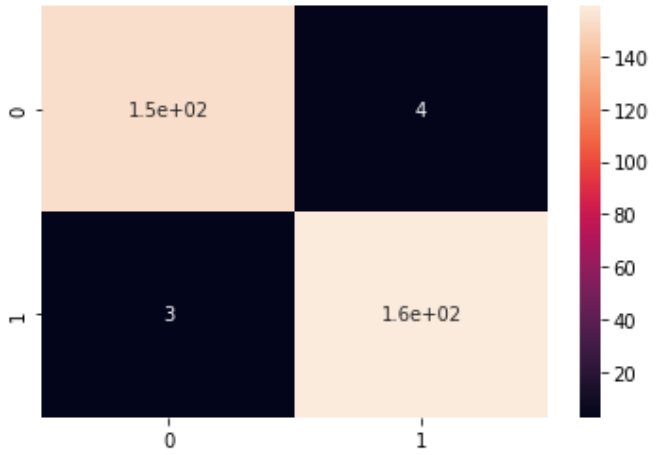


Figure 7: Confusion matrix

The average accuracy over the 10 folds was 99.35 percent with a standard deviation of 0.004 when utilizing XGB to choose the better parameters for efficient classification. It is as shown in the Figure 8.

```
Best: 0.993538 using {'scale_pos_weight': 1}
0.993538 (0.004385) with: {'scale_pos_weight': 1}
0.993343 (0.004771) with: {'scale_pos_weight': 10}
0.993025 (0.004751) with: {'scale_pos_weight': 25}
0.992358 (0.004992) with: {'scale_pos_weight': 50}
0.992179 (0.004917) with: {'scale_pos_weight': 75}
0.992252 (0.004803) with: {'scale_pos_weight': 99}
0.992244 (0.005110) with: {'scale_pos_weight': 100}
0.989339 (0.005750) with: {'scale_pos_weight': 1000}
```

Figure 8: Mean accuracy and standard deviation against the scale_pos_weight for XGB classifier.

We examined the mean accuracy and standard deviation of the various classifiers and discovered that XGB and Random Forest fared better than the others. It is as shown in the Figure 9a and 9b.

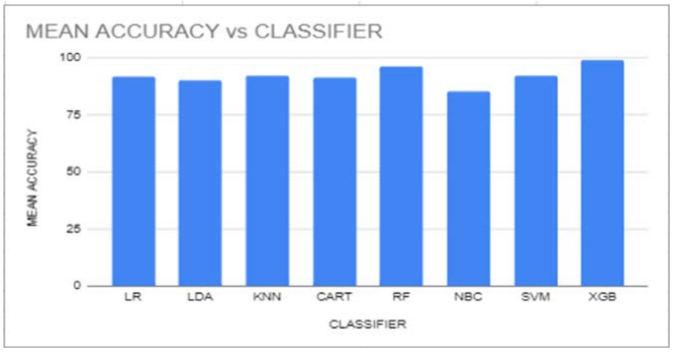


Figure 9a: Classifier vs Mean Accuracy

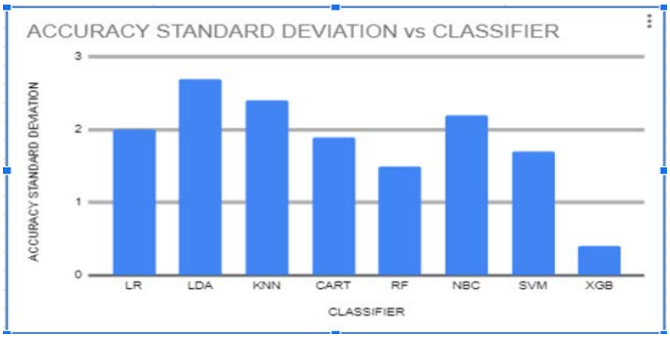


Figure 9b: Classifier vs Mean standard deviation.

Conclusion

The plant serves a fundamental need for all living things. Because of the vast range of diseases, identifying and categorising diseases using artificial eyes is not only time-consuming and labor-intensive, but it is also possible to misidentify with a high mistake rate. As a result, we demonstrate how to use XGBoost to categorise plant leaves as healthy or unhealthy. We conducted experiments using a dataset that is openly available and was used by numerous researchers that includes images of various plant species impacted by numerous diseases. Linear regression, Linear Discriminant Analysis, K-nearest neighbours Classifier, CART, Random Forest Classifier, Gaussian NB, SVM, and XGBoost were used to compare the performance of the suggested technique with state-of-the-art machine learning algorithms. In terms of Accuracy, Precision, Recall, and F1-score, the investigative findings show that the suggested technique beat the other machine learning algorithms. The proposed XGBoost algorithm beats existing classifiers based on the data. The testing shows that the proposed strategy is effective, as it achieves a classification rate of 99.35% for the dataset.

Conflict of Interest

Authors declare no conflict of interest is there for publication of this work.

References

1. N.G. Kurale, M. V. Vaidya. Classification of Leaf Disease Using Texture Feature and Neural Network Classifier. *Proc. Int. Conf. Inven. Res. Comput. Appl. ICIRCA 2018* **2018**, 1–6.
2. R.D.L. Pires, D.N. Gonçalves, J.P.M. Oruê, et al. Local descriptors for soybean disease recognition. *Comput. Electron. Agric.* **2016**, 125, 48–55.

3. I.N. Gahlawat, P. Lakra, J. Singh, B.S. Chhikara. Developmental and histochemical studies on carposporophyte of Solieria robusta (Greville) Kylin Solieriaceae, Gigartinales) from Port Okha, India. *J. Integr. Sci. Technol.* **2020**, 8 (2), 12–20.
4. S.V. Meena, V.S. Dhaka, D. Sinwar. Exploring the Role of Vegetation Indices in Plant Diseases Identification. In *2020 Sixth International Conference on Parallel, Distributed and Grid Computing (PDGC)*; IEEE, **2020**; pp 372–377.
5. A. Chlingaryan, S. Sukkarieh, B. Whelan. Machine learning approaches for crop yield prediction and nitrogen status estimation in precision agriculture: A review. *Comput. Electron. Agric.* **2018**, 151, 61–69.
6. K. Bhosle, B. Ahirwadkar. Deep learning Convolutional Neural Network (CNN) for Cotton, Mulberry and Sugarcane Classification using Hyperspectral Remote Sensing Data. *J. Integr. Sci. Technol.* **2021**, 9 (2), 70–74.
7. B. Keswani, A.G. Mohapatra, A. Mohanty, et al. Adapting weather conditions based IoT enabled smart irrigation technique in precision agriculture mechanisms. *Neural Comput. Appl.* **2019**, 31, 277–292.
8. T. Chen, C. Guestrin. XGBoost: A scalable tree boosting system. *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.* **2016**, 13-17-August-2016, 785–794.
9. X. Jiang, J. Zhen, J. Miao, et al. Assessing mangrove leaf traits under different pest and disease severity with hyperspectral imaging spectroscopy. *Ecol. Indic.* **2021**, 129, 107901.
10. K. Suganya Devi, P. Srinivasan, S. Bandhopadhyay. H2K – A robust and optimum approach for detection and classification of groundnut leaf diseases. *Comput. Electron. Agric.* **2020**, 178, 105749.
11. K. Zhang, Q. Wu, Y. Chen. Detecting soybean leaf disease from synthetic image using multi-feature fusion faster R-CNN. *Comput. Electron. Agric.* **2021**, 183, 106064.
12. Q. Yang, J. Li, W. Ma, et al. Melatonin increases leaf disease resistance and saponin biosynthesis in Panax notogiseng. *J. Plant Physiol.* **2021**, 263, 153466.
13. H. Hamdani, A. Septiarini, A. Sunyoto, S. Suyanto, F. Utaminingrum. Detection of oil palm leaf disease based on color histogram and supervised classifier. *Optik (Stuttg).* **2021**, 245, 167753.
14. X. Chen, G. Zhou, A. Chen, et al. Identification of tomato leaf diseases based on combination of ABCK-BWTR and B-ARNet. *Comput. Electron. Agric.* **2020**, 178, 105730.
15. Z. Jiang, Z. Dong, W. Jiang, Y. Yang. Recognition of rice leaf diseases and wheat leaf diseases based on multi-task deep transfer learning. *Comput. Electron. Agric.* **2021**, 186, 106184.
16. C. Wang, P. Du, H. Wu, et al. A cucumber leaf disease severity classification method based on the fusion of DeepLabV3+ and U-Net. *Comput. Electron. Agric.* **2021**, 189, 106373.
17. G. Hu, X. Yang, Y. Zhang, M. Wan. Identification of tea leaf diseases by using an improved deep convolutional neural network. *Sustain. Comput. Informatics Syst.* **2019**, 24, 100353.
18. S. Shrivastava, S.K. Singh, D.S. Hooda. Soybean plant foliar disease detection using image retrieval approaches. *Multimed. Tools Appl.* **2017**, 76 (24), 26647–26674.
19. H. Ali, M.I. Lali, M.Z. Nawaz, M. Sharif, B.A. Saleem. Symptom based automated detection of citrus diseases using color histogram and textural descriptors. *Comput. Electron. Agric.* **2017**, 138, 92–104.
20. S. Zhang, H. Wang, W. Huang, Z. You. Plant diseased leaf segmentation and recognition by fusion of superpixel, K-means and PHOG. *Optik (Stuttg).* **2018**, 157, 866–872.
21. L. Jidong, Z. De-An, J. Wei, D. Shihong. Recognition of apple fruit in natural environment. *Optik (Stuttg).* **2016**, 127 (3), 1354–1362.
22. J. Francis, Anto Sahaya Dhas D, Anoop B K. Identification of leaf diseases in pepper plants using soft computing techniques. In *2016 Conference on Emerging Devices and Smart Systems (ICEDSS)*; IEEE, **2016**; pp 168–173.
23. D. Ashourloo, A.A. Matkan, A. Huete, H. Aghighi, M.R. Mobasheri. Developing an Index for Detection and Identification of Disease Stages. *IEEE Geosci. Remote Sens. Lett.* **2016**, 13 (6), 851–855.
24. M. Lv, G. Zhou, M. He, et al. Maize Leaf Disease Identification Based on Feature Enhancement and DMS-Robust Alexnet. *IEEE Access* **2020**, 8, 57952–57966.
25. S.S. Chouhan, A. Kaul, U.P. Singh, S. Jain. Bacterial foraging optimization based radial basis function neural network (BRBFNN) for identification and classification of plant leaf diseases: An automatic approach towards plant pathology. *IEEE Access* **2018**, 6, 8852–8863.
26. B. Liu, C. Tan, S. Li, J. He, H. Wang. A Data Augmentation Method Based on Generative Adversarial Networks for Grape Leaf Disease Identification. *IEEE Access* **2020**, 8, 102188–102198.
27. C. Zhou, S. Zhou, J. Xing, J. Song. Tomato Leaf Disease Identification by Restructured Deep Residual Dense Network. *IEEE Access* **2021**, 9, 28822–28831.
28. D. Ashourloo, H. Aghighi, A.A. Matkan, M.R. Mobasheri, A.M. Rad. An Investigation Into Machine Learning Regression Techniques for the Leaf Rust Disease Detection Using Hyperspectral Measurement. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2016**, 9 (9), 4344–4351.
29. Y. Yuan, Z. Xu, G. Lu. SPEDCCNN: Spatial pyramid-oriented encoder-decoder cascade convolution neural network for crop disease leaf segmentation. *IEEE Access* **2021**, 9, 14849–14866.
30. P. Jiang, Y. Chen, B. Liu, D. He, C. Liang. Real-Time Detection of Apple Leaf Diseases Using Deep Learning Approach Based on Improved Convolutional Neural Networks. *IEEE Access* **2019**, 7, 59069–59080.
31. Q. Wu, Y. Chen, J. Meng. Dcgan-based data augmentation for tomato leaf disease identification. *IEEE Access* **2020**, 8, 98716–98728.
32. T.N. Pham, L. Van Tran, S.V.T. Dao. Early Disease Classification of Mango Leaves Using Feed-Forward Neural Network and Hybrid Metaheuristic Feature Selection. *IEEE Access* **2020**, 8, 189960–189973.
33. V. Pallagani, V. Khandelwal, B. Chandra, et al. dCrop: A Deep-Learning Based Framework for Accurate Prediction of Diseases of Crops in Smart Agriculture. In *2019 IEEE International Symposium on Smart Electronic Systems (iSES) (Formerly iNiS)*; IEEE, **2019**; pp 29–33.
34. G. Xie, B. Guo, Z. Huang, Y. Zheng, Y. Yan. Combination of Dominant Color Descriptor and Hu Moments in Consistent Zone for Content Based Image Retrieval. *IEEE Access* **2020**, 8, 146284–146299.