Wireless Personal Communications
https://doi.org/10.1007/s11277-020-07590-x

Tomato Leaf Disease Classification using Multiple Feature Extraction Techniques

Jagadeesh Basavaiah[1] · **Audre Arlene Anthony**[1]

Abstract
Agriculture, with its allied sectors, is the largest source of livelihoods in India. Diseases in plants cause a substantial decrease in quality as well as quantity of crops or agricultural products. Detection of these diseases is the solution to prevent losses in the harvest and amount of agricultural products. The main objective of the proposed method is to develop a technique to identify leaf disease in tomato plant by improving the classification accuracy and reducing computational time. The novelty of the work is fusion of multiple features in order to improve classification accuracy. Color histograms, Hu Moments, Haralick and Local Binary Pattern features are used for training and testing purpose. Rrandom forest and decision tree classification algorithms are uses for leaf disease classification. Based on the experiments conducted, it showed that the random forest classifier is more accurate than decision tree classifier. The classification accuracy is 90% for decision tree classifier and 94% for random forest classifier respectively.

Keywords Decision tree · Feature extraction · Leaf disease · Random forest

1 Introduction

Tomato is a member of the family solanaceace and it is one among the main crops of India. India has become the second largest leading nation in the production of tomatoes in the world producing 18.7 million tons annually. Most of the common variants of tomatoes that are grown in India are pear, beefsteak, cherry, heirloom and roma. Several tomato diseases such as bacterial canker, viruses, bacterial speck, leaf blights, bacterial spots, early blight, anthracnose, late blight and bacterial soft rot can cause crop losses. Since prevention is better than cure, detection of the diseases is one of the solutions to reduce the crop losses.

Pests and diseases in plants result in devastation of plants or portion of the plant causing in declined yield of crops which leads to food disruption. Moreover, the awareness

✉ Jagadeesh Basavaiah
jagadeesh.b@vvce.ac.in

Audre Arlene Anthony
audre.arlene@vvce.ac.in

[1] Department of Electronics and Communication Engineering, Vidyavardhaka College of Engineering, Mysuru, India

Published online: 16 June 2020

about management of pests or disease control is not much in various parts of the country. Various new methods have been developed to reduce post-yield processing, to strengthen the agricultural sustainability and to increase the production. Many approaches like thermography, mass spectrometry, gas chromatography, polymerase chain reaction and hyper spectral techniques are being used for identifying the diseases. The above approaches are costly and computational time is high. Currently, mobile based and server based techniques are being used for disease identification. Some of the aspects of these techniques includes good resolution camera, higher configuration processor with good performance and high end built-in accessories [1] resulting in accurate and efficient disease identification. Recent methodologies like deep learning and machine learning are being utilized to improve the disease detection, recognition and classification. Many investigations are going on in the field of machine learning to detect and classify diseases in plants like fuzzy logic, neural networks and deep neural networks.

The objective of the work is to detect and classify tomato leaf diseases using multiple feature extraction techniques. The features that are commonly extracted are color features, shape features, texture and so on. In this work, the features that are extracted are haralick, Hu moments, local binary pattern and color histograms. These features are fed as an input to the random forest and decision tree classifiers for classifying the diseases. Once the features are extracted, classification is performed using decision tree and random forest classifiers. Random Forests is an ensembling technique which utilizes several decision tree models to predict the disease. A decision tree is a pictorial illustration of all the possible results to a decision depending on certain conditions. Finally, cross validation is done in order to check which classifier shows higher accuracy.

2 Leaf Diseases and Its Symptoms

Some of the common tomato leaf diseases are discussed in this section.

2.1 Bacterial Spots

Bacterial spot is due to disease causing bacterial species Xanthomonas occurring worldwide where tomatoes are being grown. It causes leaf and fruit spots leading to defoliation, sun-burnt fruit, and loss of yield. Because of variety in the bacterial spot pathogens, this disease can be found in wide range of temperatures and it is a menace to tomato production globally. Increase in disease is favored by temperatures of 75°–86° F and high precipitation. The lesions in the leaf are firstly round in shape, water-soaked and may be enclosed by a faint yellow region. Generally, the color of these spots may vary from dark brown to black and appears to be circular or rounded on leaves and stems. Spots hardly grow to more than a diameter of 3 mm. The leaf lesions may cause a blighted appearance in leaves and a in general yellowing or browning or withering or dying of leaves occur with increase in lesions or multiple lesions.

Figure 1 shows the tomato leaf having bacterial spots.

2.2 Mosaic Virus

There are numerous ways in which a tomato plant can get infected with mosaic virus. Commonly, the infection may be due to the remains of virally infected plants which still exist in

Fig. 1 Tomato leaf having bacterial spots

the soil. The virus can live for a minimum of 50 years in dead and dried remains of infected plants. Tomato mosaic virus has no cure or remedy. If a plant is infected, the virus can easily spread to other plants. The infection may spread from farmers who told infected plants and then touch a healthy plant. The other minor source of transmission may be insects which migrate from infected plant region to healthy plant region. The infection can also spread from gardening tools, pots, or planters which are used in infected plant region. Light and dark green spotted regions appears on the tomato leaves if it is infected with mosaic virus. Other symptoms are diminutive growth, deformities in tomato fruit, and decrease in the quantity of yield. Leaves may become curled, yellowish and fern-like in appearance.

Figure 2 shows the tomato leaf with mosaic virus.

2.3 Septoria Spots

Septoria spots are instigated by a fungus by name Septoria lycopersici. Septoria spots are one among the damaging tomato plant diseases and are predominantly more in the regions where humid and wet weather conditions prevails for longer extended periods. Septoria spots normally appear in the lower leaves when the first fruit starts to grow. These spots are rounded or circular in shape with a diameter of around 1/16 to 1/4 inch characterized by tan to gray centers with small black fruiting structures and dark brown margins. Typically, on a single leaf there can be many spots. The disease will spread towards the upper leaves from the lower leaves. When lesions on the leaf are more, the leaf will gradually turn to light yellow, later to brown which later withers. Normally the fruit in this type of disease rarely get infected.

Figure 3 shows the tomato leaf with Septoria Spots.

2.4 Yellow Curl

Tomato Yellow Leaf Curl Virus (TYLCV) is a type of DNA virus from Geminiviridae family. It causes one of the utmost damaging diseases of tomato. Typically it appears in

Fig. 2 Tomato leaf with mosaic virus

Fig. 3 Tomato leaf with septoria spots

subtropical and tropical geographical regions. This virus is transmitted through insect belonging to Aleyrodidae family. The main host for TYLCV is the tomato plant and other plant hosts are tobacco, potatoes, beans, eggplants and peppers. The swift transmission and spread of TYLCV in recent times has resulted in increased attention in research to apprehend and control the harmful virus as well as disease.

Figure 4 shows the tomato leaf with Yellow Curl.

3 Literature Review

Hang et al. [2] proposed a deep learning-based method to identify and classify plant leaf diseases. The proposed method took the advantages of the neural network to extract the characteristics of diseased parts, and thus to classify target disease areas. To address the issues of long training convergence time and too-large model parameters, the traditional convolutional neural network was improved by combining a structure of inception module, a squeeze-and-excitation (SE) module and a global pooling layer to identify diseases. The feature data of the convolutional layer were fused in multi-scales to improve the accuracy on the leaf disease dataset.The global average pooling layer was used instead of the fully connected layer to reduce the number of model parameters. Compared with some traditional convolutional neural networks, the proposed model yielded better performance and achieved an accuracy of 91.7% on the test data set. At the same time, the number of model parameters and training time have also been greatly reduced.

Vamsidhar et al. [3] developed a segmentation technique for automatic detection and classification of plant leaf diseases. Features are extracted and selected features are used for training and support vector machine (SVM) and artificial neural network (ANN) classifiers. The texture and colour features are extracted and these features are selected to get the better feature set as input to the classification algorithms. The k-means can also be used as classification and hence it is used as classifier and found out that the accuracy presented by it 85.3%. Initially input to the SVM is given without feature selection and using k-means for segmentation of the images. The SVM used for the classification and it obtained 90% accuracy. The Linear Kernel is applied in SVM and found out that classification accuracy is 89%. The RBF kernel is considered 88.8% and SVM polynomial kernel gave 90.2%. The Naive Bayes method obtained an accuracy of 86%. The MLP with the Back-propagation as the learning algorithm is used and accuracy obtained 88.59%. The optimized MLP have obtained accuracy of 91.45%.

Sannakki and Rajpurohit [4] developed a method which works primarily on the scheme of segmenting. In this method, features used were color and texture. For the classification

Fig. 4 Tomato leaf with yellow curl

purpose, neural network classifier was used. The important benefit of using color and texture features with neural network was that L*a*b was calculated to extract the chromaticity layers in an image and categorization efficiency was 97.30%. The major drawback of this method is that it is applicable only for few number of crops.

Rothe and Kshirsagar [5] proposed a method using snake segmentation approach. Hu moments were used as distinguishing features. Active contour model was employed to minimize the vitality within the infected regions. Back Propagation Neural Network (BPNN) was used to classify the leaf disease. The average classification accuracy was 85.52%.

Rastogi et al. [6] proposed a technique using K-means clustering to segment the infected regions. Gray Level Co-occurrence Matrix (GLCM) features were extracted and Fuzzy Logic and Artificial Neural Network (ANN) were used as classify the diseases.

Owomugisha et al. [7] introduced a technique in which color histogram features were extracted and transformed the color space from RGB to HSV color model, RGB to L*a*b color model. Five shape features were utilized and region under the curve investigation was used to classify the diseases. Different classification algorithms like Random Forest, SVM, Nearest Neighbors, Naïve Bayes, Decision Tree, Extremely Randomized Tree were used. Out of these, extremely randomized trees proved to be more efficient than other classifiers.

Sladojevic et al. [8] discussed a method of using deep learning methods. The Deep learning techniques were used for automated detection and classification of plant diseases using images of plant leafs. The proposed model was efficient to detect and differentiate healthy leaves and thirteen different plant diseases.

4 Methodology

The proposed methodology is as shown in Fig. 5. The proposed methodology includes the following steps: Collection of Database, Feature Extraction, Training, Testing and Classifying.

4.1 Collection of Database

Collection of Database step includes collecting data samples, separating the collected data samples and save them in appropriate folders. The data samples are stored in five folders with four leaf diseases and one healthy leaf as: Bacterial, Healthy, Mosaic, Septorial and Yellow curved. The images in the train folder are strictly based for training the classifier model and hence we do not test the images which are in train folder. So, for the purpose of testing, we split the dataset into two parts 60% of them will be saved in train folder and other 40% will be stored in test folder. All the classes of tomato leaves are shuffled in the test folder and no separate folders are created in test folder. The train folder consists of 300 images in which each class folder consists of 60 images and test folder consists of 200 images, 40 from each class mixed.

4.1.1 Feature Extraction

In computer vision and machine learning, feature extraction refers to a primary set of data which are measured and constructs features anticipated to be non-redundant and informative simplifying the succeeding learning process which leads to enhanced human understanding and interpretations. Feature extraction is basically a dimensionality reduction technique in

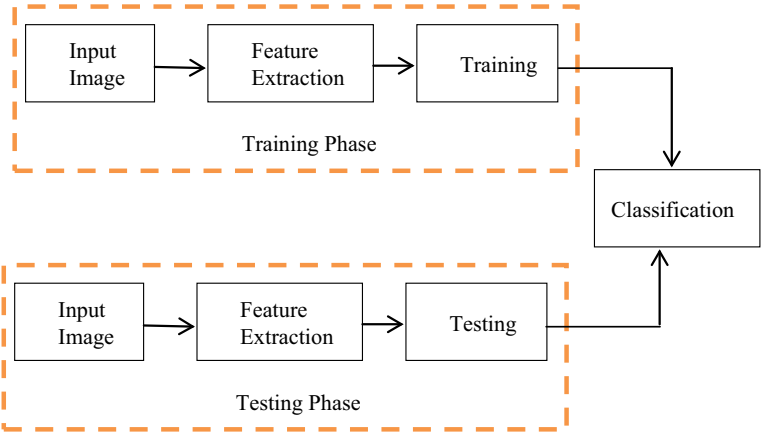

Fig. 5 Proposed methodology

which a set of raw variables is reduced to adaptable groups of features for further processing keeping the original data set as it is. In order to give data for the classifier, features must be extracted from the dataset. The work deals with the multiple feature extraction for tomato leaf disease classification. For color features, color histograms are extracted. For shape features Hu-moments are extracted and for texture features haralick features are extracted.

1. Color Histograms

Color histogram in an image depicts the dispersal of colors. In image processing, color histogram symbolizes number of appearances of a particular color. Color histograms may be constructed in any color space. Generally, color space is separated into suitable number of ranges, each consisting of same values of color. It can be also exemplified as a smooth function which is determined over the given color space that estimates the number of pixels. The color histograms of the tomato leaves with diseases is as shown in Fig. 6.

2. Hu Moments

For calculating the second feature that is the shape feature, Hu-moments are extracted. Basically, image moments is defined as the weighted average of image pixel intensities. The image moments $\boldsymbol{M_{ij}}$ for an image $\boldsymbol{I}(\boldsymbol{x},\boldsymbol{y})$ given by the sum of pixel intensities, expressed as,

$$M_{ij} = \sum_{i} \sum_{j} I(x, y) \tag{1}$$

In Eq. (1), the summation of pixel intensities is calculated. In other terms, the intensities of pixels are weighted on the basis of its intensity and not depending on its location. For a binary image, moment is considered as the number of white pixels or white region in the image. For two shapes to be similar, the image moment will be essentially the same, but it is not an adequate form. Two images with same moment can be constructed even if the images appear to be different. Hence there is a need to find stable moments. The Central moments are much identical to the raw image moments which is given by,

$$\mu_{ij} = \sum_{x} \sum_{y} (x - \bar{x})^i (y - \bar{y})^j I(x, y) \tag{2}$$

The central moments are invariant to scale, translation and rotation. It would be sufficient to the state of shape matching. Moments [7] are a group of seven numbers determined using

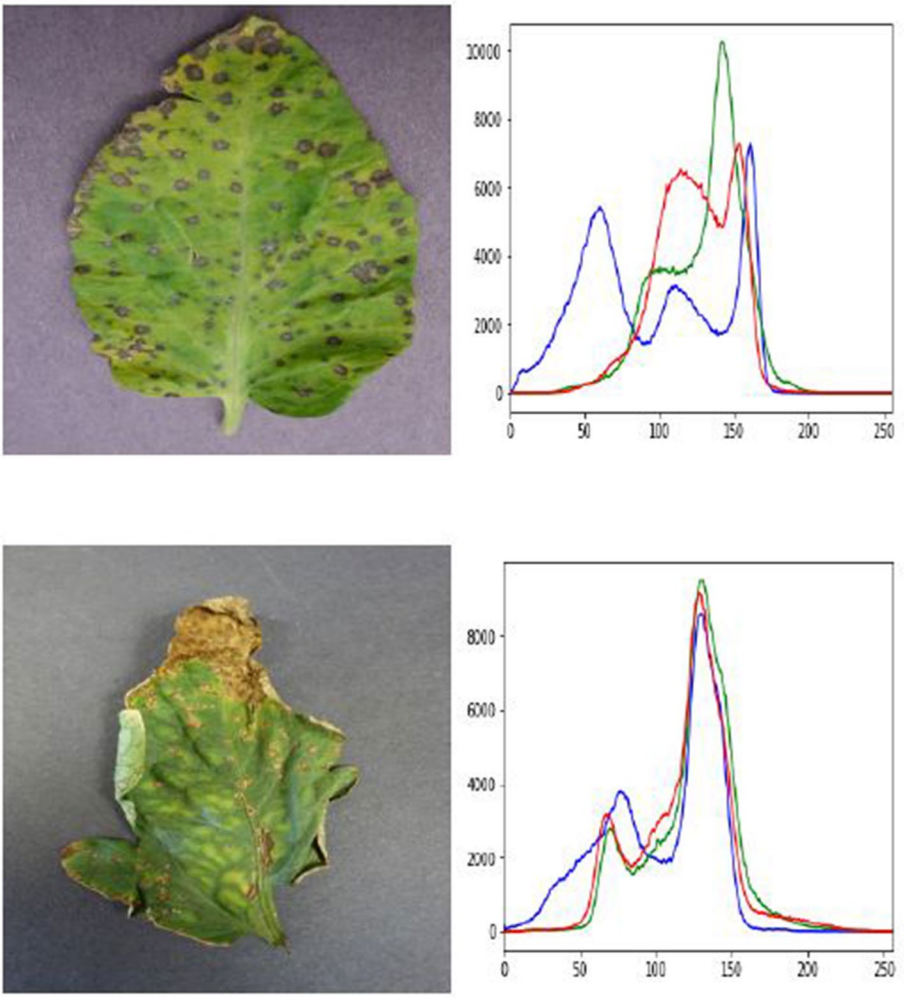

Fig. 6 Color histograms of the tomato leaves with diseases

central moments which are invariant to image transformations. The first six moments are proved to be invariant to scale, translation, reflection and rotation. The sign of the seventh moment will change with image reflection. The equations for seven moments are as follows:

$$h_0 = \eta_{20} + \eta_{02} \tag{3}$$

$$h_1 = \left(\eta_{20} - \eta_{02}\right)^2 + 4\eta_{11}^2 \tag{4}$$

$$h_2 = \left(n_{30} - 3\eta_{12}\right)^2 + \left(3\eta_{21} - \eta_{03}\right)^2 \tag{5}$$

$$h_3 = \left(\eta_{30} + \eta_{12}\right)^2 + \left(\eta_{21} + \eta_{03}\right)^2 \tag{6}$$

$$\begin{aligned} h_4 = {} & \left(\eta_{30} - 3\eta_{12}\right)\left(\eta_{30} + \eta_{12}\right)\left[\left(\eta_{30} + \eta_{12}\right)^2 - 3\left(\eta_{21} + \eta_{03}\right)^2\right] \\ & + \left(3\eta_{21} - \eta_{03}\right)\left[3\left(\eta_{30} + \eta_{12}\right)^2 + \left(\eta_{21} + \eta_{30}\right)^2\right] \end{aligned} \tag{7}$$

$$h_5 = \left(\eta_{20} - \eta_{02}\right)[\left(\eta_{30} + \eta_{12}\right)^2 - \left(\eta_{21} + \eta_{03}\right)^2 + 4\eta_{11}\left(\eta_{30} + \eta_{12}\right)\left(\eta_{21} + \eta_{03}\right) \tag{8}$$

$$\begin{aligned} h_6 = {} & \left(3\eta_{21} - \eta_{03}\right)\left(\eta_{30} + \eta_{12}\right)\left[\left(\eta_{30} + \eta_{12}\right)^2 - 3\left(\eta_{21} + \eta_{03}\right)^2\right] \\ & + \left(\eta_{30} - 3\eta_{12}\right)\left(\eta_{21} + \eta_{03}\right)\left[3\left(\eta_{30} + \eta_{12}\right)^2 - \left(\eta_{21} + \eta_{03}\right)^2\right]. \end{aligned} \tag{9}$$

3. Haralick Features

Texture is one of the significant element in human visual perception. Statistical texture approaches consider the distribution of gray values in spatial domain by finding the local features of every pixel in the given image and deducing a set of statistical parameters using the distributions of the local features. This approach is widely employed in many image understanding and analysis applications. It has two steps: GLCM computation and calculation of texture features based on GLCM. GLCM is a square matrix having a dimension N_g, where N_g corresponds to the number of gray levels. The matrix elements are obtained by summing the number of times a pixel with value 'i' is adjacent to a pixel with value 'j' and then dividing the whole matrix by the total number of such comparisons made. Each entry is thus considered to be the probability that a pixel with value 'i' will be found adjacent to a pixel of value 'j'.

Fourteen statistical terms is determined using co-occurrence matrix to define the texture, which are as follows:

$$Angular\ Second\ Moment = \sum_i \sum_j p(i,j)^2 \tag{10}$$

$$Contrast = \sum_{n=0}^{N_g - 1} n^2 \left\{ \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p(i,j) \right\}, |i - j| = n \tag{11}$$

$$Correlation = \frac{\sum_i \sum_j (ij) p(i,j) - \mu_x \mu_y}{\sigma_x \sigma_y} \tag{12}$$

$$Variance = \sum_i \sum_j (i - \mu)^2 p(i,j) \tag{13}$$

$$Inverse\ Difference\ Method = \sum_i \sum_j \frac{1}{1 + (i - j)^2} p(i,j) \tag{14}$$

$$Sum\ Average = \sum_{i=2}^{2N_g} ip_{x+y}(i) \tag{15}$$

$$Sum\ Variance = \sum_{i-2}^{2N_g} \left(i - f_s\right)^{2p_{x+y}}(i) \tag{16}$$

$$Sum\ Entropy = -\sum_{i=2}^{2N_g} p_{x+y}(i) \log\left\{p_{x+y}(i)\right\} = f_s \tag{17}$$

$$Entropy = -\sum_{i}\sum_{j} p(i,j)\log(p(i,j)) \tag{18}$$

$$Difference\ Variance = \sum_{i=0}^{N_g-1} i^2 p_{x-y}(i) \tag{19}$$

$$Difference\ Entropy = -\sum_{i=0}^{N_g-1} p_{x-y}(i) log\left\{p_{x-y}(i)\right\} \tag{20}$$

$$Info.\ Measure\ of\ Correlation\ 1 = \frac{HXY - HXY1}{\max\{HX, HY\}} \tag{21}$$

$$Info.\ Measure\ of\ Correlation\ 2 = (1 - \exp\left[-2(HXY2 - HXY)\right])^{\frac{1}{2}} \tag{22}$$

$$Max\ Correlation\ Coeff. = Square\ root\ of\ the\ second\ largest\ Eigen\ values. \tag{23}$$

4. Local Binary Patterns

Local binary patterns (LBP) are visual descriptors which are employed in computer vision for classification. It is validated that when LBP and Histogram of Oriented Gradients (HOG) descriptors are combined, detection efficiency can be enhanced substantially.

The LBP feature vector is constructed using the following steps:

- Divide the window into cells.
- Every pixel within the cell is compared with its eight neighbors.
- Write "0″ if the center pixel value is greater than the neighbor pixel value, otherwise, write "1". This gives an 8- bit binary number, and its decimal equivalent is found.
- Find the histogram of the image. it may be perceived as a 256-dimensional feature vector.
- The histogram is normalized.
- The histograms are concatenated. The feature vector for the entire window is thus obtained.

Figure 7 shows the local binary patterns and corresponding histogram plots of two different leaves.

The feature vector is trained by employing machine-learning algorithm for classification of images. In this work, Decision tree and Random Forest algorithms are used. In this work, texture analysis is employed for classification purpose.

4.1.2 Training

The major step in the proposed methodology is the training process. Figure 8 shows the flowchart of training process. The images are resized to a resolution of 500 * 500. This is because to maintain uniformity in extracted features. The next step after resizing the image is feature extraction. Hu moments, haralick, color histogram and LBP features are extracted, concatenated and stored in a number format. After all the features

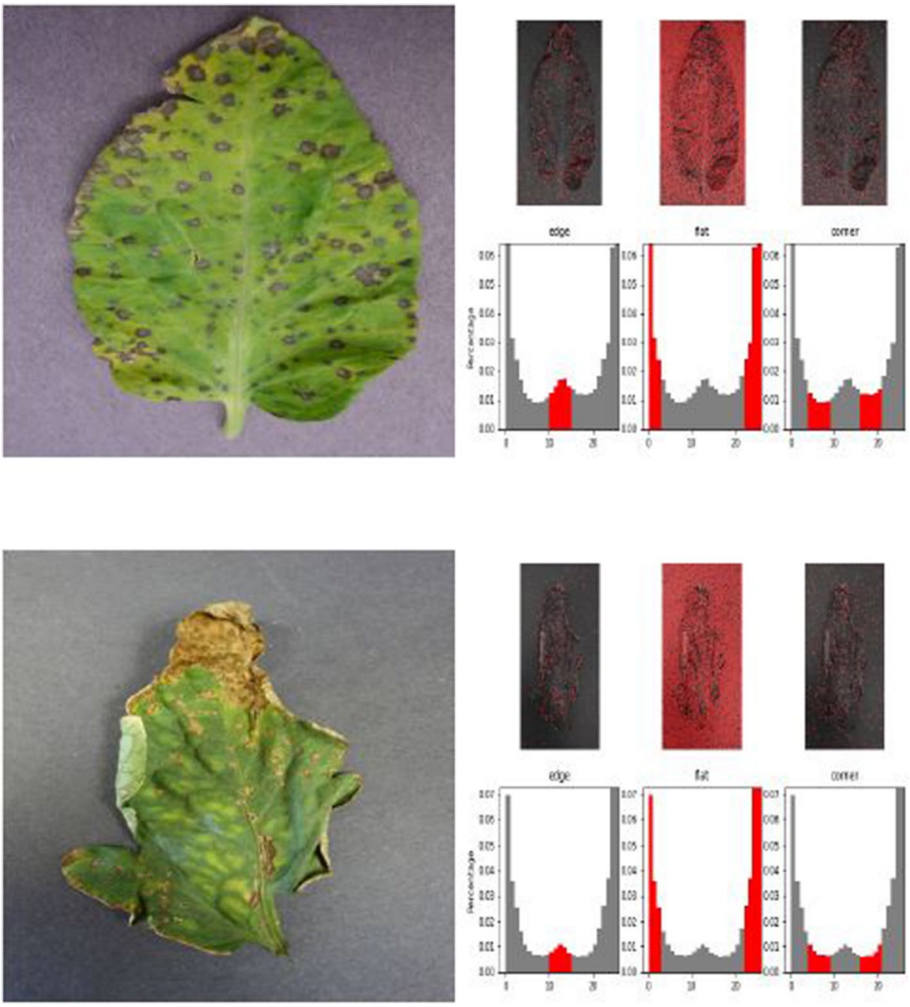

Fig. 7 Local binary patterns and corresponding histogram plot

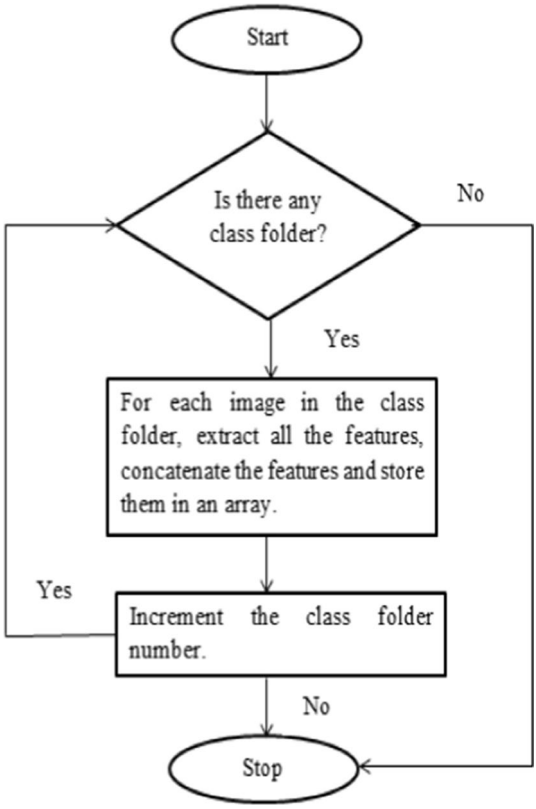

Fig. 8 Flowchart of training process

are extracted from the class folder, increment the class folder by 1 till the features are extracted from all the five folders.

In this work, the extracted features for 300 training images and encoded with labels from 0 to 4 with 0 being first folder class and 4 being fifth folder class. After the end of last image of the last folder, the flow of the process gets broken and it gives the message training has been completed.

The next process after the training process is the testing process where the test images are given as query.

4.1.3 Testing and Classification

After the training process, the next step involves testing the created algorithm by the test data. Figure 9 depicts the flowchart of the testing process. There are 14 statistical parameters which can be determined from the co-occurrence matrix for describing the texture of the image. They are as stated below:

The testing process flow begins by loading the extracted features. The files are loaded since it has to be compared with the test data. Once the files are loaded, the classifier is created using the create classifier function. Now the classifiers are created and needed to be trained to find out the cross validation score. To find out the cross validation score we need to give the testing data and labels. These have been already loaded at the beginning of

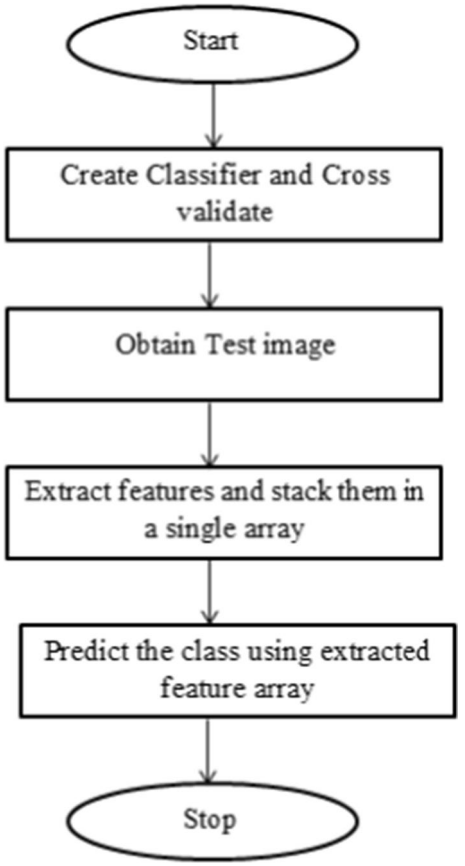

Fig. 9 Flowchart of testing process

testing. With test split as 0.1 that is splitting the test data itself as train data 90 percent and test data 10 percent, the project shows a cross validation score of 95%. Once the classifier is trained it is now required to test with new samples i.e. images in the test folder.

Now, test data path is given as an input to the created algorithm. Once again, all the processes which were applied to the training data that is for the training image the same features are extracted: Color Histogram, Hu Moments, Haralick texture and Local Binary Feature patterns features are extracted in the same order and are stacked to one another to create an array. This feature array is fed as input to the trained algorithm to predict the label. The classifier predicts the label from the input array. The encoded label output is nothing but the prediction: 0 being first class and 4 being fifth class. Following subsections show the working of the classifiers that has been used in this work: Decision Tree and Random Forest Classifiers.

1. Decision Tree Classifier

It is a tree structure similar to a flowchart in which an internal node characterize the attributes or features. The branch portrays a decision rule, and the outcome is defined by the leaf node. The upper node is denotes the root node. It learns the partitioning depending on the feature values. Recursive method is used for partitioning the trees. This tree structure is used in framing decisions. This easily imitates reasoning at human level. Therefore

decision trees can be understandable easily. Figure 10 shows the basic structure of Decision Tree Algorithm.

The decision tree is a type of non-parametric or distribution free method and is independent of probability distribution functions. Decision trees is able to handle high dimensional data with better precision.

Decision Tree Classifier works in three steps:

1. Choose the finest feature or attribute utilizing Attribute Selection Measures (ASM) to partition the records.
2. Select the attribute as a decision node and divide the dataset into subsets of smaller size.
3. Begin tree building process recursively for every child till at least one of the following conditions match:

   - All the tuples belong to the same attribute value.
   - There are no more remaining attributes.
   - There are no more instances

The decision tree generation algorithm is as shown in Fig. 11.

2. Random Forest Classifier

Random forest is a type of supervised learning algorithm. This can be employed for classification as well as regression. This algorithm is highly flexible and simple

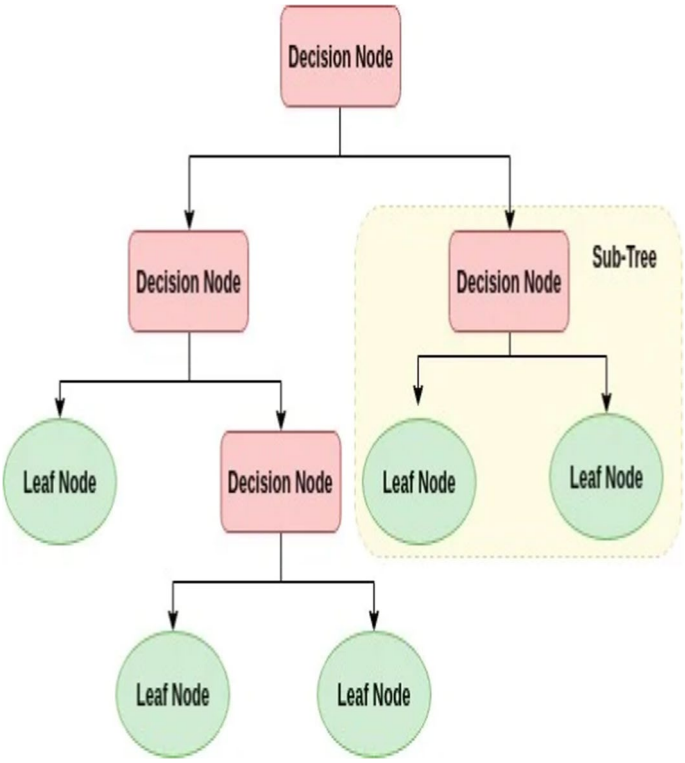

Fig. 10 Basic structure of decision tree algorithm

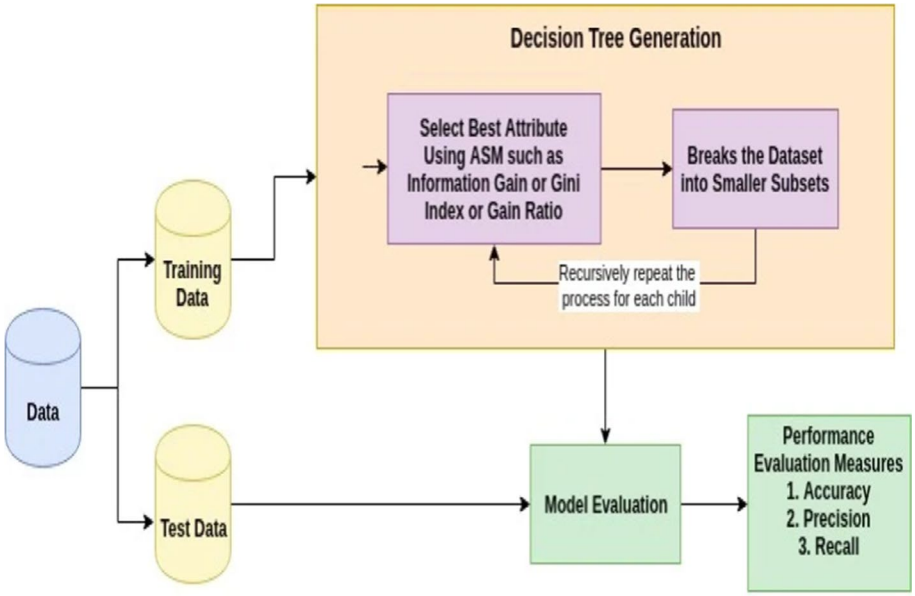

Fig. 11 Decision tree generation algorithm

to employ. A forest is said to be consisting of trees, more the number of trees, more will be robustness of the forest. Random forests generates decision trees on arbitrarily chosen data samples, collects estimate from every tree and chooses the simplest solution by voting. Random forests have a wide range of applications like feature selection, recommendation engines and classification. It is based on Boruta algorithm that selects significant features in the given dataset. It is basically an ensemble method of decision trees created on an arbitrarily split dataset. This group of decision tree classifiers is referred as a forest. The decision trees are created using feature selection indicators like Gain ratio, Gini index and Information gain for each attribute. Each tree is influenced by an independent random sample. During classification, each and every tree votes and the best class is selected as an outcome. For regression, the mean of all the outputs of trees are considered as the final outcome. It is very simple and dominant when compared with other non-linear classification algorithms. Random Forest classifier works in four steps:

1. Choose random samples from the given dataset.
2. Build a decision tree for every sample and get a likelihood result from every decision tree.
3. Perform voting for every predicted result.
4. Choose the prediction result with highest votes as the final prediction.

Figure 11 shows the Random Forest Classifier Algorithm. Random forest is a good feature selection pointer. It automatically computes the relevance score for every feature in the training phase. The relevance score is scaled down to make the sum of all scores

as "1". The score helps in selecting highly significant features and neglect the least significant features for developing the model. Random forest utilizes the Gini importance or Mean Decrease in Impurity (MDI) to compute the significance of every feature. Gini importance is often referred as the total reduction in node impurity. This indicates how the model fits or accuracy decreases when a variable is dropped. More the reduction, more significant the variable will be. The average reduction is an important parameter for selecting the variable. The Gini index defines the total descriptive power of the variables. Figure 12 shows the architecture of Random Forest algorithm.

5 Experimental Analysis and Results

In this work, the dataset is divided into train and test folders. In train folder, 5 classes are made namely: "Bacterial", "Healthy", "Septoria", "Mosaic" and "Yellow Curl". Each of these classes contains 60 images. Therefore, the train folder consists of 300 images. In test folder, there are 200 images where each class consists of 40 images. Overall the dataset contains 500 images.

After training and testing of images, confusion matrix is calculated. Confusion matrix gives the information about correct and incorrect matching and hence the overall accuracy can be found. Tables 1 and 2 show the Confusion Matrix for Decision Tree classifier and Confusion Matrix of Random Forest classifier respectively.

From Table 1, it is clearly seen that 90% of bacterial leaf spot diseases are correctly classified, 100% of healthy leaves are correctly classified, 82.5% of mosaic leaves are correctly classified, 92.5% of septoria leaves are correctly classified and 85% of yellow-curl leaves are correctly classified. Overall, the decision tree classifier gave 90% accuracy.

From Table 2, it is clearly seen that 90% of bacterial leaf spot diseases are correctly classified, 100% of healthy leaves are correctly classified, 92.5% of mosaic leaves are correctly classified, 95% of septoria leaves are correctly classified and 92.5% of yellow-curl leaves are correctly classified. Overall, the decision tree classifier gave 94% accuracy.

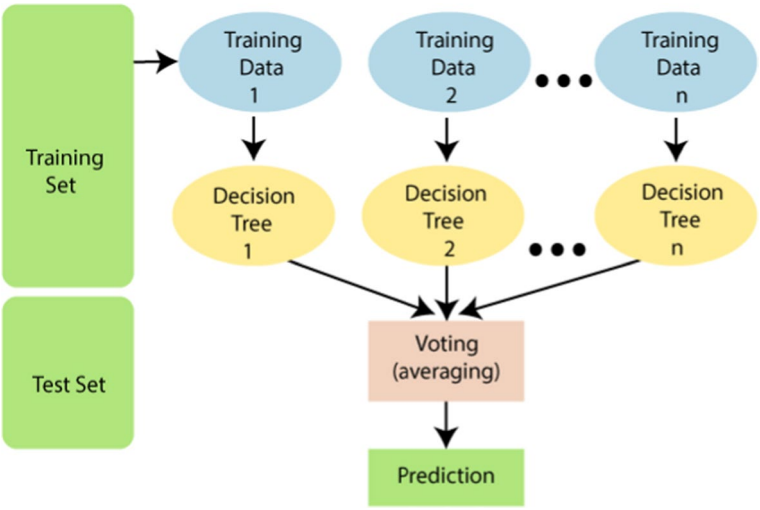

Fig. 12 Architecture of random forest algorithm

Table 1 Confusion matrix for decision tree classifier

	Bacterial spot	Healthy	Mosaic	Septoria spot	Yellow curl
Bacterial spot	90	5	0	2.5	2.5
Healthy	0	100	0	0	0
Mosaic	0	10	82.5	0	7.5
Septoria spot	0	0	5	92.5	2.5
Yellow curl	2.5	0	5	7.5	85
Overall			90		

Table 2 Confusion matrix for random forest classifier

	Bacterial spot	Healthy	Mosaic	Septoria spot	Yellow curl
Bacterial spot	90	0	0	5	5
Healthy	0	100	0	0	0
Mosaic	0	0	92.5	0	7.5
Septoria spot	0	2.5	0	95	2.5
Yellow curl	0	0	2.5	5	92.5
Overall			94		

Table 3 Comparison table of proposed method with other state-of-the-arts techniques

Authors/references	Techniques used	Classification accuracy (%)
Hang et al. [2]	Convolutional Neural network + inception module + squeeze-and-excitation (SE) module + global pooling layer	91.7
Vamsidhar et al. [3]	K-Means Classifier	85.3
Vamsidhar et al. [3]	Support Vector Machine	89
Vamsidhar et al. [3]	RBF Kernel	88.8
Vamsidhar et al. [3]	SVM Polynomial Kernel	90.2
Vamsidhar et al. [3]	Optimized MLP	91.45
Sannakki and Rajpurohit [4]	Neural Network Classifier	97.30
Rothe and Kshirsagar	Back Propagation Neural Networks (BPNN)	85.52
Proposed method	Decision Tree Classifier	90
Proposed method	Random Forests Classifier	94

The classification accuracy of the proposed method is compared with other state-of-the-art techniques. Table 3 shows the comparison table of methods used and classification accuracy of different methods.

6 Conclusion and Future Work

In the proposed work four main diseases of tomato leaves: bacterial spot, septoria spot, mosaic virus and yellow-curl are detected using multiple feature extraction methods. Also the diseases are classified using decision tree classifier and random forest classifier with an accuracy of 90% for decision tree classifier and 94% for random forest classifier respectively. The results show that the random forest classifier is more accurate than the decision tree classifier. The main advantage of the proposed method is reduced computational time and the classification accuracy is quite high compared to other state-of-the-art techniques.

Future work includes extending the number of disease classification in both intra and inter class meaning classification of various diseases along with various species, improving the accuracy by using features such as Visual Bag of Words features, local features such as scale-invariant feature transform (SIFT), speeded up robust features (SURF), oriented FAST and rotated BRIEF (ORB), developing a system that will help in automatic capture, detect and classify the disease and take necessary action using suitable hardware and software requirement.

References

1. Ramesh, S., Hebbar, R., Niveditha, M., Pooja, R., Prasad Bhat, N., Shashank, N., & Vinod, P. V. (2018). Plant disease detection using machine learning. In *2018 International conference on design innovations for 3Cs compute communicate control (ICDI3C)*.
2. Hang, J., Zhang, D., Chen, P., Zhang, J., & Wang, B. (2019). Classification of plant leaf diseases based on improved convolutional neural network. *Sensors (Basel), 19*(19), 4161. https://doi.org/10.3390/s19194161.
3. Vamsidhar, E., Jhansi Rani, P., & Rajesh Babu, K. (2019). Plant disease identification and classification using image processing. *International Journal of Engineering and Advanced Technology (IJEAT)*. ISSN 2249-8958, Vol. 8, No. 3S.
4. Sannakki, S., & Rajpurohit, V. (2015). Classification of pomegranate diseases based on back propagation neural network. *International Research Journal of Engineering and Technology (IRJET)*, *2*(2).
5. Rothe, P., & Kshirsagar, R. (2015). Cotton leaf disease identification using pattern recognition techniques. In *2015 International conference on pervasive computing (ICPC)*, IEEE, pp. 1–6.
6. Rastogi, A., Arora, R., & Sharma, S. (2015). Leaf disease detection and grading using computer vision technology and fuzzy logic. In *2015 2nd international conference on signal processing and integrated networks (SPIN)*, IEEE, pp. 500–505.
7. Owomugisha, G., Quinn, J. A., Mwebaze, E., & Lwasa, J. (2014). Automated vision-based diagnosis of banana bacterial wilt disease and black SIGATOKA disease. In *International conference on the use of mobile ICT in Africa*, Citeseer.
8. 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*. https://doi.org/10.1155/2016/3289801.

Jagadeesh Basavaiah has received his master degree in Industrial Electronics. His area of interests is image and video processing, computer vision. He has published many papers in national and international journals.

Audre Arlene Anthony has received her master degree in Digital Electronics and Communication Systems. Her area of interests is digital electronics, signal processing and computer vision. She has published many papers in national and international journals.