TELKOMNIKA Telecommunication Computing Electronics and Control
Vol. 24, No. 2, April 2026, pp. 574~587
ISSN: 1693-6930, DOI: 10.12928/TELKOMNIKA.v24i2.27599

A comprehensive analysis of feature selection and XAI for machine learning classifiers to recognize guava disease

Sujon Chandra Sutradhar, Md. Mehedi Hasan
Department of General Education, Faculty of Digital Transformation Engineering, University of Frontier Technology, Gazipur, Bangladesh

Article Info

Article history:

Received Oct 11, 2025
Revised Jan 9, 2026
Accepted Jan 30, 2026

Keywords:

AdaBoost
Explainable artificial intelligence
Guava disease
Machine learning
Recognition

ABSTRACT

Recognizing and classifying diseases in guava is crucial for managing farms to keep crops healthy and increase harvest quality. Cultivators face the most severe challenges when it comes to recognizing and diagnosing guava fruit and leaf illnesses, a task that is nearly impossible to perform manually. This research focuses on developing a robust disease identification model using image data collected locally from guava trees. After data collection, various image processing techniques, including scaling and contrast enhancement, are utilized to make the data more suitable for use. K-means clustering is employed to quickly divide the images into groups, followed by the extraction of important characteristics. Two separate feature ranking approaches, analysis of variance (ANOVA) and least absolute shrinkage selection operator (LASSO), are used to select the best characteristics, identifying the 10 most important attributes. The adaptive boosting (AdaBoost) classifier achieves the highest accuracy among six classifiers for the top seven characteristics indicated by LASSO among the specified features. To enhance the model's interpretability, two explanation methods, local interpretable model-agnostic explanations (LIME) and shapley additive explanations (SHAP), are employed to illustrate how the classifier reaches its conclusions. This approach not only simplifies disease identification but also clarifies the reasoning behind predictions, opening the door to real-world applications in detecting and preventing dangerous diseases.

Corresponding Author:

Md. Mehedi Hasan
Department of General Education, Faculty of Digital Transformation Engineering
University of Frontier Technology
Kaliakoir, Gazipur-1750, Bangladesh
Email: mehedi0001@uftb.ac.bd

1. INTRODUCTION

Guava (*Psidium guajava*) is one of the most widely cultivated tropical fruits, contributing substantially to global agricultural output and trade. In 2024, the annual global production of guava was estimated at nearly 55 million tonnes, with India alone accounting for 45% of the total [1]. In most cases, human beings have been the ones to grow the guava seedling. There are instances in which the seeds of the guava have been dispersed by birds and other creatures with four feet for such a long period of time that their origin is unknown. Nevertheless, it is thought to be a region that ranges from the southern part of Mexico into Central America or via Central America. It is not uncommon to come across guava bushes in all warm regions of tropical America, as well as in the West Indies (since 1526), the Bahamas, Bermuda, and southern Florida. It is said to have been first implemented in the year 1847, and by the year 1886, it had spread throughout more than half of the state [1]. Beyond its economic importance, guava holds nutritional and

medicinal value, making it a vital crop for both farmers and consumers. However, guava production is highly vulnerable to plant diseases, which threaten crop yield, reduce quality, and ultimately impact farmers' livelihoods and the agricultural economy. Timely and precise detection of guava diseases is crucial to minimize crop losses and ensure sustainable production.

It is possible to consider artificial intelligence (AI) and data-driven methodologies to be the contemporary equivalents of the experimental component of the scientific method, which entails the methodical collection of data, the examination of patterns, correlations, and the establishment of links between events that have been observed [2]. AI systems recapture this process by recognising patterns and generating predictions only based on data that has been seen, which is accomplished via the use of machine learning (ML) algorithms. The association process that is intrinsic to AI is often lacking in transparency, which makes it difficult for humans to appreciate how judgments are arrived at by these systems. The black box aspect of many AI models raises questions about accountability, justice, and trust. As AI becomes an increasingly driving force in decision-making across a variety of sectors, the need for explanations that are both visible and understandable for these connections becomes of the utmost importance.

This research aims to develop and assess an automated system to aid farmers in identifying guava diseases by prompt detection and analysis, thus averting substantial agricultural losses. A system that is easy to use and set up could improve guava production, which is a major contributor to global gross domestic product (GDP) through large-scale exports, while also solving longstanding problems in agriculture. The proposed method lowers the chance of widespread harm by making it easier to find infections early on. It also gives advice on safe growing practices and, in the end, helps increase profits by optimizing growth circumstances.

This study is mostly on explainable AI (XAI). ML has emerged as a very successful methodology for pattern recognition in extensive and intricate datasets; nevertheless, its deficiency in interpretability hinders its implementation in essential sectors such as agriculture. ML has already shown promising results in diagnosing plant diseases [3]. Algorithms made for automatic disease detection can give useful hints that help find problems early on, making it easier to treat and manage them quickly. However, traditional disease detection still relies heavily on expert botanists looking at plants, which is costly, takes a lot of time, and isn't always effective.

The main goal of this research is to use six distinct ML models in an image-based framework to tell the difference between healthy and defective guava leaves and fruits. To accomplish this, different image processing methods were used before disease segmentation and feature extraction. Also, two methods for selecting features were used to rank and prioritize them, which made the model more accurate and efficient. Lastly, XAI methods were used to explain the predictive results, which made the model's decision-making process clear and ensured the proposed system was reliable. The rest of this paper is structured as: section 2 reviews related work in detecting guava diseases and ML. Section 3 describes the proposed framework and methodologies. Section 4 presents the experimental setup and results, followed by a discussion in section 5. Finally, the paper concludes with future research directions.

2. LITERATURE REVIEW

In the modern day, the majority of research on ML and deep learning is mostly concentrated on agricultural problems since this industry makes a significant contribution to the economy of the whole globe. On the other hand, there is a limited amount of study on the disease identification of fruits like guava, mango, jackfruits, and so on. For the purpose of identifying guava leaf disease, Howlader *et al.* [4] developed a deep convolutional neural network (DCNN)- model. For the purpose of developing the model, 2705 photos illustrating four different diseases were used. It was during the training and testing phase that they used 25 epochs that they attained an accuracy of 98.74% and 99.43% respectively.

The identification of plant leaf disease was accomplished by Geetharamani and Pandian [5] via the development of a model that used a nine-layer convolutional neural network (CNN) architecture. Both the Plant Village dataset and the Kaggle dataset were used by them. The Kaggle dataset had 55448 photos of 13 different plant leaves that were categorized into 38 different groups. Comparing the suggested model with other classification methods such as support vector machines (SVM), logistic regression (LR), decision trees (DT), and k-nearest neighbors (KNN) classifiers revealed that the CNN-model performed the best, with an impressive prediction accuracy of 96.46%.

Turkoglu *et al.* [6] presented a multi-model pre-trained CNN model for identifying Apple illness and pests. The model used the AlexNet, GoogleNet, and DenseNet201 models, and it was trained on 1192 pictures that depicted four prominent apple diseases. With a score of 96.10%, the DenseNet201 achieved the greatest accuracy score among the models that were applied. For the purpose of evaluating the effectiveness of deep learning approaches in detecting sweetness and quality, Jupudi [7] used an image classification system on an orange. Even though the source of the dataset was not disclosed, the objective of the research

project was applied to a total of five thousand images. The model was trained using SVM, AlexNet, stacked autoencoder (SAE), and kernel sparse stacked autoencoder (KSSAE), with KSSAE reaching the highest possible accuracy of 92.1% throughout the course of the training process. DenseNet201 seems to have the most outstanding performance, as seen by its score of 96.1% from the total.

A deep residual network (ResNet) that utilizes a contrast enhancement and transfer learning strategy was suggested by Trang *et al.* [8] in order to recognize mango illness. With an accuracy rate of 88.46%, the proposed algorithm was able to properly detect three prevalent ailments based on a total of 394 images. It was suggested by Nikhitha *et al.* [9] that the Inception V3 model be used for the processes of illness detection and fruit identification. Bananas, apples, and cherries were selected as disease detection targets, and the Inception V3 model was applied only to these fruits. GitHub was the source of this information as well.

A DCNN was proposed by Ma *et al*. [10] for the purpose of symptom-wise diagnosis of four cucumber disorders. The network achieved a recognition rate of 93.4%. Preprocessing and classification are two examples of well-known image processing processes that are used in the method that was presented by Prakash *et al*. [11] for the purpose of identifying illnesses that affect leaves. An evaluation of the offered method is carried out on a collection of sixty images, of which thirty-five are cancerous and twenty-five are benign, with an accuracy rate of 90%. The K-means clustering method is used to segment the disease-affected region, and then the gray-level co-occurrence matrix (GLCM) algorithm is used to extract features from the segmented area. After the feature vector has been formed, it is next categorized by utilizing the SVM classifier. With the use of 688 images, Al Buhaisi [12] was able to determine the kind of pineapple by using the VGG16 model. A hundred percent accuracy was achieved by the trained model, and it is quite probable that this dataset was overfitting; otherwise, the accuracy would not have been achievable.

A diagnostic technique that utilizes deep learning was described by Elleuch *et al.* [13]. During this investigation, they made use of their recently developed dataset, which included five different kinds of plant data. As part of the training process for their model, they used transfer learning architecture using VGG-16 and ResNet. t. In order to evaluate the validity of this model, they compared the suggested model to both actual data and data that contains augmentations. With the use of transfer learning, VGG-16 progressively gave results that were both promising and realistic in terms of accuracy, with 99.02% and 98.35% respectively.

An approach to the identification of guava illness that is based on computer vision was developed by Elleuch *et al.* [13]. This approach makes use of three CNN-based models with distinct optimizers. On the other hand, they do not specify any trustworthy online sources for the data that was acquired. Both the dropout value and the third optimizer showed remarkable accuracy when the dropout was 50%, which was 96.1%. A DCNN based technique was presented by Mostafa *et al.* [14] for the purpose of detecting guava illness. This approach used five different neural network network architectures. The dataset that they utilized was one that was acquired locally in Pakistan. Having achieved an accuracy rate of 97.74%, the classification result demonstrated that ResNet-101 was the model that was the most suitable for their purpose.

An application of nine important classifiers was proposed by Habib *et al.* [15] for the purpose of developing a machine vision-based disease detection system for the purpose of identifying illness in three different species of fruit, namely guava, papaya, and jackfruit. When it came to recognizing illnesses that affect guava and jackfruit, the random forest (RF) classifier achieved the best accuracy, with a rate of 96.8% and 89.59%, respectively. From the above analysis it can be said that most of research on guava diseases recognition has been performed using ML or deep learning classisifier. There is not any research on feature selection and XAI based implementation which is our motivation to work with.

3. METHOD

Diseases that affect guava fruit may be classified using a variety of methods, each of which is further subdivided into a great number of categories. The first thing that we did was go out into the field and capture the image. Before beginning this inquiry, the second step is to do some preliminary processing on the data. Furthermore, by using GLCM and statistical feature extraction, we were able to extract thirteen distinct characteristics from one picture. There are two distinct features selection procedures that are used in order to ascertain the mutual score of each feature. These strategies include analysis of variance (ANOVA) and least absolute shrinkage selection operator (LASSO). Following the completion of the process of selecting features, the top ten features were selected for further examination. Contrast (CON), correlation (COR), skewness (SKEN), kurtosis (KTS), variance (VAR), standard deviation (STD), entropy (ENT), energy (ENG), mean (MN), and homogeneity (HGN) are some of the qualities that are included in this category. After that, we divided the dataset into two halves, applied algorithms, and used seven different performance assessment indicators to evaluate each technique. The overall working procedure is presented in Figure 1.

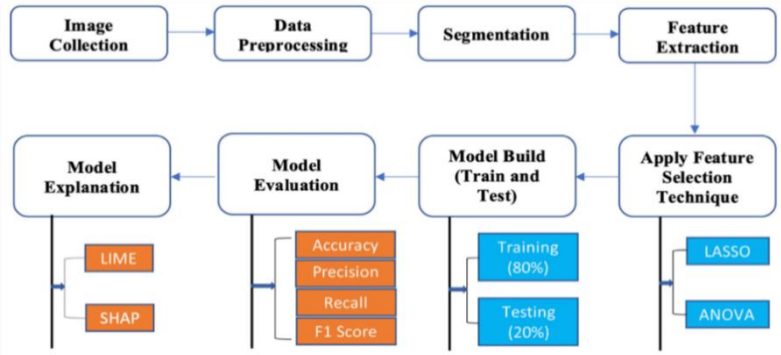
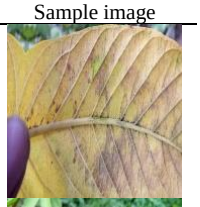


Figure 1. Working procedure for guava disease classification

3.1. Data description

In this research, the Samsung S24 smartphone equipped with a main camera that has a resolution of 50 megapixels, a storage capacity of 128 gigabytes, and 8 gigabytes of random access memory (RAM) was used to collect the images as well as for the image acquisition process. 239 disease-free images and 287 disease-affected images are among the total of 526 guava images that have been gathered from the guava field. The overall data distribution is depicted in Table 1.

Table 1. Data frequency for acquisition of image

Class	Binary	Num. of images	Sample image
Disease-affected	1	287	
Disease-free	0	239	
Total	20	526	15.3

3.2. Image preprocessing

Image preprocessing is essential for effective computer vision tasks. Initially, our images were captured at a resolution of 2160×2160 pixels and resized to 300×300 pixels using the bilinear interpolation method [16], which calculates pixel values in the compressed image by interpolating distances between four neighboring pixels using the formula $U(x) = 1 - |x|$ for $|x| \leq 1$ and 0 otherwise. Some images contained noise, which was addressed using a Gaussian filter to enhance quality via gamma correlation, followed by histogram equalization to improve contrast [17]. This process normalizes the image intensity range (0 to 1), transforming the intensity distribution density function $p(x)$ to a uniform density of one post-equalization.

3.3. Segmentation

This research used the K-means clustering algorithm, supplemented by boundary and spot detection techniques, to segment the pictures [18]. The 8-connected pixel approach is utilized for boundary detection. Euclidean distance is employed for K-means clustering in this instance. The K-means clustering technique primarily assigns each pixel in the image to the cluster with the minimum distance from the cluster's centroid, subsequently performs color segmentation on the image, and ultimately selects the cluster that exclusively contains regions of interest (ROIs). We have used the K-means clustering approach to segment guava image data into smaller pieces in this paper. K-means is an unsupervised method used to find separate groups in the data based on how similar the data is. This is one of the most used clustering algorithms, where k stands for the number of clusters. We have chosen $k = 3$ for this task to segment an image, which means that it will find 3 groups in the image. The K-means clustering algorithm works well on a limited set of data [19].

3.4. Extraction of relevant features

Images are represented as large pixel matrices, from which dimensionality can be reduced through feature extraction [20]. In this research, thirteen features were extracted, including GLCM measures and statistical descriptors: CON, COR, SKEN, KTS, VAR, STD, ENT, ENG, MN, HGN, root mean square (RMS), smoothness (SM), and inverse difference moment (IDM). For two pixels at $z(x,y)$ separated by distance d and angle α, the GLCM can be defined as:

$$Q(j,k,d,\alpha) = \{(a_1,b_1),(a_2,b_2) \in S \times T : d,\alpha,z(a_1,b_1) = j, z(a_2,b_2) = k\} \quad (1)$$

In this formulation, $Q(j,k)$ represents the $(j,k)-th$ element of the computed GLCM, t_g is the number of gray levels, and $\mu_x, \mu_y, \sigma_x, \sigma_y$ are the means and standard deviations of row and column sums. The main GLCM-based features are defined as:

$$CNT = \sum_{k=0}^{t_g-1}\sum_{j=0}^{t_g-1}(k-j)^2 Q(k,j), \quad (2)$$

$$CRL = \frac{\sum_{j=0}^{t_g-1}\sum_{k=0}^{t_g-1} j.k.Q(j,k) - \mu_x\mu_y}{\sigma_x\sigma_y}, \quad (3)$$

$$EG = \sum_{k=0}^{t_g-1}\sum_{j=0}^{t_g-1} Q(j,k)^2, \quad (4)$$

$$ENT = -\sum_{k=0}^{t_g-1}\sum_{j=0}^{t_g-1} Q(j,k)\log Q(j,k), \quad (5)$$

$$HGN = \sum_{k=0}^{t_g-1}\sum_{j=0}^{t_g-1} \frac{Q(j,k)}{1+(\mathrm{j}-\mathrm{k})^2}, \quad (6)$$

$$IDM = \sum_j \sum_k \frac{Q(j,k)}{1+(\mathrm{j}-\mathrm{k})^2} \quad (7)$$

In addition to GLCM features, statistical descriptors were extracted. For a set of pixels with intensity P_x, MN, STD, VAR, KTS, and skewness (SKEN) are defined as:

$$MN = \frac{1}{P_x}\sum_{n=1}^{P_x} G_n \quad (8)$$

$$STD = \sqrt{\frac{\sum_{n=1}^{P_x}(G_n-C)^2}{P_x}} \quad (9)$$

$$VAR = \frac{1}{P_x}\sum_{n=1}^{P_x}(G_n - C)^2 \quad (10)$$

$$KTS = \frac{\frac{1}{P_x}\sum_{n=1}^{P_x}(G_n-C)^4}{\left(\frac{1}{P_x}\sum_{n=1}^{P_x}(G_n-C)^2\right)} - 3 \quad (11)$$

$$SKEN = \frac{\sigma - G}{Q} \quad (12)$$

Here, C is the mean intensity, G the pixel count at a given intensity, and Q the normalization factor. These measures characterize intensity distributions in both defective and defect-free regions of grayscale images.

3.5. Description of feature selection techniques

Feature selection was applied to retain relevant attributes and remove redundancy [21]. The methods used for ranking features were ANOVA and LASSO. All three metrics such as RMS, smoothness, and IDM are amplitude-based measures of the same underlying surface signal, often linearly related or derived from squared deviations. Unless the surface profile varies widely in frequency content, they will respond almost identically [22] and were excluded, leaving the top ten features: CNT, CRL, SKEN, KTS, VAR, STD, ENT, EG, MN, and HGN.

3.5.1. ANOVA

ANOVA is a statistical method that is used to investigate whether there is any equal variation among groups of categorical variables that pertain to numerical response. ANOVA can be represented by (13):

$$F = \frac{SSB/(K-1)}{SSW/(n-K)} \quad (13)$$

ANOVA tests whether group means differ significantly (13), where SSB and SSW are the between-group and within-group variances, K is the number of classes, and n is the sample size [23].

3.5.2. LASSO

LASSO is a regression-based feature selection technique that penalizes the sum of absolute coefficient values to prevent overfitting (14), with α as the regularization parameter [24].

$$Minimize = \frac{1}{2n_{samples}} \left|\left|y - X\omega\right|\right|^2 + \alpha\left|\left|\omega\right|\right|_1 \quad (14)$$

3.6. Classifier training and testing

In this section, three splitting ratios of the dataset, such as train (60%), validation (20%), and test (20%), are used. Training data is used for training a ML model, where test data evaluates the efficiency of each model. Comparing the performances of various trained models is measured with the validation data.

3.7. Hyperparameter tuning

A type of parameter whose value is determined prior to algorithm training is known as a hyperparameter. Fine-tuning a parameter is a well-used technique for upgrading the accuracy of the ML model. In this research, Table 2 includes all the employed algorithm's hyperparameter values. Here, LBFGS means limited-memory Broyden–Fletcher–Goldfarb-Shanno, gini refers to gini impurity, and radial basis function (RBF).

Table 2. Hyperparameters of classifiers used for guava disease recognition

Model	Hyperparameter	Value
AdaBoost	n estimators	100
	Min samples leaf	1
	Criterion	Gini
	Min samples split	2
	Max features	Square root (sqrt)
DT	Min samples leaf	1
	Criterion	gini
	Min samples split	2
KNN	n neighbors	5
	Weight	Uniform
	Algorithm	Auto
	Leaf size	30
	P	2
	Metric	Minkowski
SVM	C	1
	Kernel	RBF
	Gamma	Scale
RF	n estimators	100
	Min samples leaf	1
	Criterion	Gini
	Min samples split	2
	Max features	Auto
LR	C	1
	Max_iteration	1000
	Solver	LBFGS

3.8. Description of utilized XAI techniques

XAI is crucial for elucidating AI model functionality, anticipated impacts, and potential biases, ensuring accuracy, fairness, and transparency in AI-driven decision-making. In this article, two most effective explainable techniques LIME and SHAP are used. LIME and SHAP are model-agnostic XAI tools that differently attribute a single prediction to its input features. While LIME builds a simple, interpretable surrogate (often a sparse linear model) around the specific instance by perturbing its features and weighting

nearby samples, yielding quick, human-readable local importance scores that can vary with perturbation choices. Essentially, LIME perturbs the input features around a given sample and learns a local linear approximation of the black-box model, from which it derives feature importance for that particular prediction [25]. SHAP uses game-theoretic Shapley values to distribute the prediction (relative to a baseline) fairly across features, offering consistent attributions that sum to the prediction difference and scaling efficiently for tree models via TreeSHAP, though it depends on the chosen background data and can be computationally heavier. SHAP considers all possible combinations of features and allocates credit in a manner that is fair and mathematically rigorous, ensuring that the sum of attributions equals the difference between the prediction and the dataset baseline [25]. In short, LIME is suitable for fast, approximate local insight, and SHAP is more suitable for principled, additive attributions of both local and global analysis.

3.9. Performance evaluation metrics

Performance evaluation matrices are essential for evaluating the performance of a model. In this research, 7 performance matrices such as accuracy (ACC), sensitivity (SEN), specificity (SPE), false positive rate (FPR), false negative rate (FNR), F1-Score, and Precision are used. All the equations from (15) to (21) are used.

$$ACC = \left(\frac{TP+TN}{TP+FP+FN+TN}\right) \times 100\% \quad (15)$$

$$SEN = \left(\frac{TP}{TP+FN}\right) \times 100\% \quad (16)$$

$$SPE = \left(\frac{TN}{TN+FP}\right) \times 100\% \quad (17)$$

$$FPR = \left(\frac{FP}{FP+TN}\right) \times 100\% \quad (18)$$

$$FNR = \left(\frac{FN}{TP+FP}\right) \times 100\% \quad (19)$$

$$Precision = \left(\frac{TP}{TP+FP}\right) \times 100\% \quad (20)$$

$$F1 - Score = \left(2 \times \frac{SEN \times Precision}{SEN+Precision}\right) \times 100\% \quad (21)$$

4. RESULTS AND DISCUSSION

The gathering of image data was the first stage in the process of achieving the work with the identification of guava illnesses. Upon the end of the picture collection process, the images that have been acquired are subsequently scaled to a resolution of 300×300 pixels. Increasing the contrast of the pictures that are shown in Figure 2 is accomplished by the use of the mapping of colour intensity. After then, the colour pictures are divided up into a number of different clusters. The K-means clustering algorithm, which is seen in Figure 3, is used in order to carry out this segmentation. It has been shown that K-means clustering is superior to other techniques that are used for segmentation. In order to get the feature vectors that are shown in Figure 4, we extracted them from each cluster. Following the conclusion of the segmentation process, we retrieved feature vectors from each segmented image. These feature vectors were then used for the training of the classifiers. ANOVA and LASSO are the two feature-selection approaches that are used once the feature extraction process has been completed. These techniques are utilised to establish the relevance of features by taking into consideration the rank values of the features.

Upon completion of the feature selection, we discovered that three characteristics (RMS, smoothness, IDM) had values that are almost identical to one another. As a result, we exclude these features based on the feature ranking score in order to resolve the duplication concerns, and we ultimately choose the top 10 features to conduct this research. Both the ANOVA and LASSO feature selections are represented by their respective mutual scores in Tables 3 and 4.

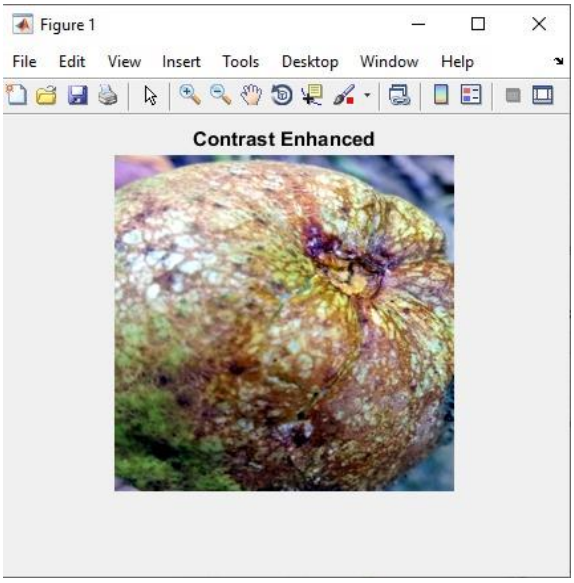


Figure 2. Contrast enhancement from the original image

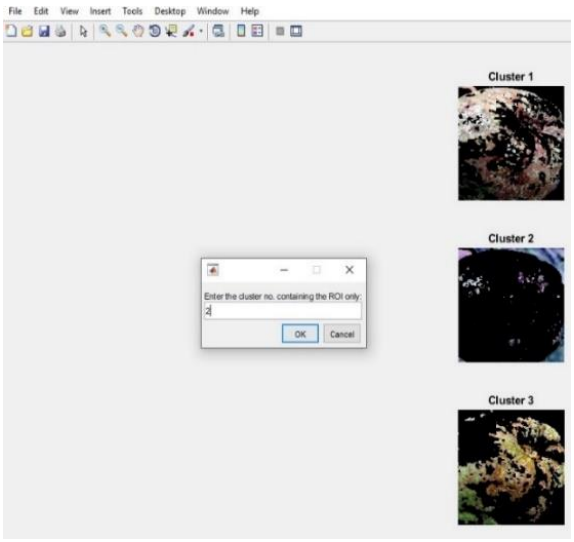


Figure 3. Enhanced image segmentation using K-means clustering

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Contrast	Correlation	Energy	Homogeneity	Mean	Standard_Deviation	Entropy	RMS	Variance	Smoothness	Kurtosis	Skewness	IDM
2	1.39877451	0.87718806	0.256841652	0.874654857	73.22196452	81.90179751	4.607674556	10.99082055	6230.39293	0.999999931	1.591699924	0.464294055	255
3	0.584099265	0.94276404	0.26742324	0.937655229	66.82184855	76.43452202	4.920358978	10.16005187	3326.985147	0.999999924	1.952908943	0.636280408	255
4	0.51239277	0.95728305	0.26010706	0.932417626	73.20480855	84.64175251	4.942532078	11.10069514	5850.289148	0.999999931	1.847646288	0.639421316	255
5	0.67221201	0.78145387	0.655483658	0.940486731	18.66759745	47.02885045	1.979768055	5.425997198	1909.542552	0.999999728	8.688276574	2.578799055	255
6	0.865150123	0.88254763	0.199530126	0.914720727	68.09713745	68.81160422	5.540059866	12.47155457	3121.136173	0.999999925	1.663092529	0.44971035	255
7	0.377757353	0.96414043	0.141913306	0.910233883	100.0650686	83.90220133	6.217810606	13.32909625	5388.002339	0.999999949	1.582957812	0.129919848	255
8	0.821875	0.80991882	0.651081652	0.935931482	18.31561788	49.13116928	1.924722783	5.902819986	2146.635442	0.999999722	9.198845454	2.714260076	255
9	2.143244485	0.79588264	0.274126158	0.866822497	68.72873942	86.06558212	4.498433427	9.985362567	6110.777116	0.999999926	2.074327533	0.781370765	255
10	1.146262255	0.83363572	0.275774872	0.906217302	58.2621816	73.61353497	4.696004477	10.28851566	4329.75525	0.999999913	2.49416775	0.908379078	255
11	1.430713848	0.82304212	0.409292042	0.922041791	42.8542277	66.99290824	3.552903244	8.448824552	3280.908698	0.999999881	3.133676464	1.240762359	255
12	1.613893995	0.86836898	0.395273896	0.913221143	55.51547241	82.79890485	3.671373703	8.678654716	5217.11106	0.999999908	2.402013339	1.054295581	255
13	0.867202819	0.92375835	0.399853646	0.950274513	56.84268188	79.16923622	3.691600117	9.551297518	5526.836954	0.999999911	2.116785233	0.892485356	255
14	0.369347426	0.97641819	0.266397311	0.953908621	108.0076599	101.0639404	5.656947287	11.65981728	5201.056382	0.999999953	1.297847133	0.149926038	255
15	1.519806985	0.91019348	0.26434826	0.9093597	93.77296956	100.322115	4.856525216	11.2535089	9146.809289	0.999999946	1.257588081	0.278540584	255
16	1.294699755	0.86315476	0.128580113	0.89589355	102.8023326	75.8853845	6.60317321	13.79721246	3763.396515	0.999999951	1.727318349	-0.028782806	255
17	0.881954657	0.91815782	0.296519349	0.933835857	65.69983927	81.63499235	4.556303123	10.88734473	5528.552397	0.999999923	1.938426183	0.749655495	255
18	0.308823529	0.95786112	0.223031276	0.917697209	68.03832499	69.26405428	5.158426916	11.48754079	3122.607335	0.999999925	1.957553757	0.477552741	255
19	0.995557598	0.88493987	0.520684051	0.939484875	36.14952087	68.44318107	2.79264397	7.67910654	3979.027158	0.999999859	4.271883167	1.655818196	255
20	1.622809436	0.88384287	0.388697871	0.928736614	66.33401998	90.11946341	3.630962747	9.369776826	7455.47454	0.999999923	1.789191896	0.748711884	255
21	1.067095588	0.86969112	0.556177858	0.917588465	33.16110738	69.47835188	2.743583894	7.477763082	4505.703103	0.999999847	5.434727708	1.974633156	255
22	0.416467525	0.964723	0.335383305	0.940046011	62.51382446	82.1059104	4.242887151	8.870837842	3217.515106	0.999999919	2.100705023	0.856535566	255
23	0.501700368	0.95200438	0.283209516	0.926677353	66.15894572	80.83821287	4.737736914	10.01032627	3274.530246	0.999999923	1.87470152	0.726689924	255
24	0.37611826	0.95485233	0.494163416	0.947019305	39.47191874	70.48973509	3.133840353	8.235891685	4123.165476	0.999999871	3.882040967	1.553213291	255
25	0.49765625	0.95821232	0.299114897	0.930340037	66.81606547	83.2904372	4.538430897	9.364738121	2891.136133	0.999999924	1.917877606	0.757119415	255
26	0.503538603	0.9562261	0.361786627	0.93060835	58.16264343	80.69549837	4.010769398	7.591996005	2247.247447	0.999999913	2.285475079	0.960833607	255

Figure 4. Sample of extracted features

Table 3. Feature rankings and scores for ANOVA feature selection

Ranking	Feature	Score	Ranking	Feature	Score
1	HGN	0.1472	6	COR	0.0846
2	CON	0.13	7	MN	0.0842
3	VAR	0.1237	8	SKEN	0.0814
4	STD	0.1153	9	KTS	0.0729
5	ENT	0.09	10	ENG	0.0708

Table 4. Feature rankings and scores for LASSO feature selection

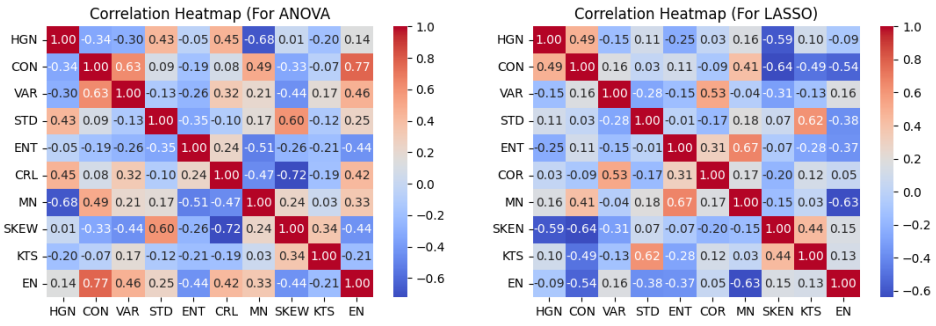
Ranking	Feature	Score	Ranking	Feature	Score
1	STD	1	6	SKEN	0.3281
2	HGN	0.9360	7	ENT	0.1759
3	CON	0.7206	8	ENG	0.1101
4	MN	0.6532	9	COR	0.0882
5	VAR	0.6450	10	KTS	0.01

The performance of six classifiers has been calculated using the confusion matrix. Table 5 provides the performance metrics for six classifiers with 10 features. From Table 5, it can be claimed that the highest accuracy is 80.19%, which is obtained by the SVM classifier. On the other hand, the lowest accuracy is 72.64% achieved by the AdaBoost classifier. It is claimed from many research articles that an odd number of feature selection works well in prediction. Also, from the correlation heatmap shown in Figure 5(a), which applies the ANOVA technique, and Figure 5(b), which applies the LASSO technique, together with the LIME and SHAP feature extraction technique shown in Figure 6(a) and (b) we select the top seven (7) and nine (9) features. In addition, the top seven and nine features that were gained via the use of ANOVA feature selection approaches are utilized to train and test the classifier that is shown in Table 6. It can be seen from Table 6 that the RF classifier achieves the maximum accuracy of 87.74% for the top seven characteristics under consideration. On the other side, the DT classifier achieves the lowest accuracy, which is 72.64%. At some point in time, the assessment metrics for other classifiers will be sufficiently accurate to constitute the outcome of the top feature set. Another feature set is also manipulated by applying LASSO feature selection techniques. Table 7 depicts the overall evaluation metrics for six classifiers. From Table 7, it is claimed that most of the classifier's accuracy is well enough compared to the ANOVA feature selected set. The highest accuracy is found for the AdaBoost classifier, which is 88.68%. Whereas the lowest accuracy is 73.58%, that obtained by KNN.

Compared to the feature set outcomes, it is said that the best performance is achieved by the AdaBoost classifier by utilizing the top 7 feature sets obtained by LASSO feature techniques. The greatest area under the curve (AUC) value for the ANOVA was 0.93 for AdaBoost and RF, which was a sufficient amount. Additionally, the LR, SVM, and KNN classifiers wereable to reach the greatest results, which were 0.86, 0.85, and 0.81 respectively. Last but not least, the lowest AUC value that can be reached using DT is 0.73 for all of the models that are applied. In the LASSO algorithm, the logistic AdaBoost classifier was able to obtain the highlighted AUC value of 0.93, which is indicative of its high level of discriminating. A number of other classifiers, including RF, LR, SVM, and KNN, also performed well, with AUC values of 0.92, 0.86, 0.85, and 0.81 respectively. Among all of the models that were evaluated, the DT had the lowest AUC value, which was 0.75.

Table 5. Performance metrics of different algorithms using 10 features

Algorithm	ACC (%)	SEN (%)	SPE (%)	FPR (%)	FNR (%)	F1-score (%)	Precision (%)
LR	78.30	90.38	66.67	33.33	9.62	80.34	72.31
KNN	74.53	80.77	68.52	31.48	19.23	75.67	71.19
DT	75.47	75.00	75.93	24.07	25.00	75.00	75.00
AdaBoost	72.64	92.31	53.70	46.30	7.69	76.80	65.75
RF	79.25	88.46	70.37	29.63	11.54	80.74	74.19
SVM	80.19	96.15	64.81	35.19	3.85	82.60	72.50



(a) (b)

Figure 5. Correlation heatmap using: (a) ANOVA and (b) LASSO techniques

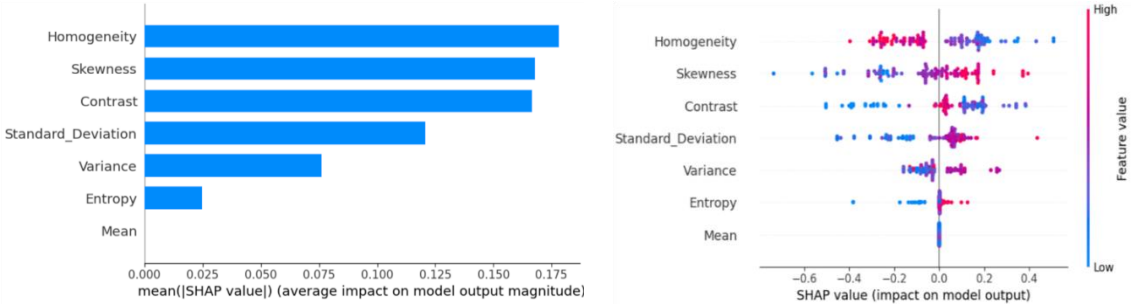


Figure 6. Model explainability visualized using: (a) LIME and (b) SHAP techniques

Table 6. Performance metrics for different algorithms using 7 and 9 features with ANOVA selection technique

Feature number	Algorithm	ACC (%)	SEN (%)	SPE (%)	FPR (%)	FNR (%)	F1-score (%)	Precision (%)
7 Features	LR	76.42	80.95	69.76	30.23	19.05	80.31	79.69
	KNN	73.58	76.19	69.77	30.23	23.81	74.42	78.69
	DT	84.13	84.13	76.44	23.56	15.87	84.13	84.13
	AdaBoost	85.85	95.24	72.09	27.91	4.76	88.00	82.76
	RF	87.74	95.24	79.07	20.93	4.76	91.67	88.46
	SVM	80.19	87.30	72.09	27.91	12.69	85.90	85.29
9 Features	LR	80.19	82.54	76.74	23.26	17.46	81.82	81.82
	KNN	73.58	76.19	69.77	30.23	23.81	74.42	78.69
	DT	72.64	76.19	67.44	32.56	23.81	71.70	77.42
	AdaBoost	85.85	88.89	81.39	18.61	11.11	88.89	88.89
	RF	83.02	88.57	77.91	22.09	14.28	85.71	85.71
	SVM	80.19	87.30	69.77	30.23	12.67	80.87	80.88

Table 7. Performance metrics for different algorithms using 7 and 9 features with LASSO selection technique

Feature number	Algorithm	ACC (%)	SEN (%)	SPE (%)	FPR (%)	FNR (%)	F1-score (%)	Precision (%)
7 Features	LR	81.13	84.13	76.74	23.26	15.87	84.13	84.13
	KNN	73.58	76.19	69.77	30.23	23.81	77.42	78.69
	DT	75.47	79.37	69.77	30.34	20.63	79.37	79.37
	AdaBoost	88.68	92.06	83.72	16.28	7.93	90.63	89.23
	RF	83.02	85.71	79.07	20.93	14.29	85.71	85.71
	SVM	80.19	87.30	69.77	30.23	12.69	83.97	80.89
9 Features	LR	79.25	82.54	74.42	25.58	17.46	82.54	82.54
	KNN	73.58	76.19	69.77	30.23	23.81	77.42	78.69
	DT	75.47	77.77	72.09	27.91	22.22	79.03	80.33
	AdaBoost	85.85	88.89	81.39	18.60	11.11	87.50	87.50
	RF	84.91	87.30	81.40	18.60	12.70	87.30	87.30
	SVM	80.19	87.30	69.77	30.23	12.70	83.97	80.88

The receiver operating characteristic (ROC) curve illustrates the most significant portion of the model in terms of its performance. When the ROC curve is evaluated, it is determined that the LASSO feature selection performs better than the ANOVA feature selection. The overall details are presented in Figures 7 and 8.

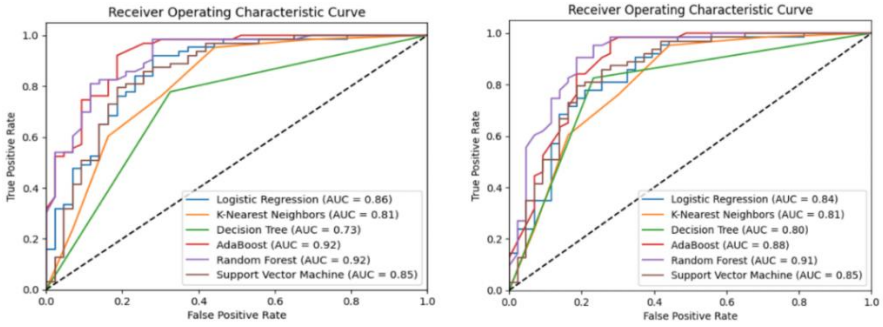


Figure 7. ROC curve for model performance visualization using ANOVA feature set

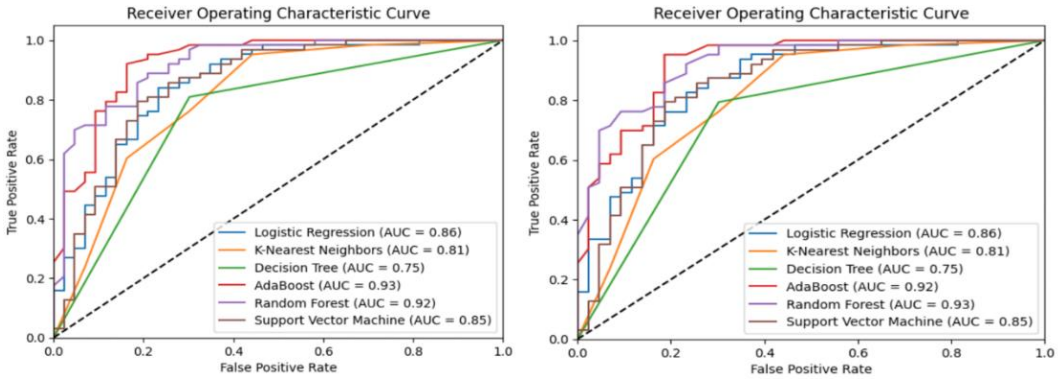


Figure 8. ROC curve for model performance visualization using LASSO feature set

The AdaBoost classifier with LASSO-selected features outperformed others due to its ability to reduce bias and focus on difficult samples, while LASSO eliminated redundant features. Lower-performing models like KNN and DT struggled with noise and overfitting. The ROC and AUC results confirm AdaBoost's strong discriminative power, highlighting its reliability for practical applications. Eventually, the performance of the best model is analyzed applying XAI techniques named LIME and SHAP, which are presented in the following Figures 6(a) and (b). From the ANOVA feature selection technique and XAI techniques, it can be said that the homogeneity features have the most influence on the prediction, whereas the mean feature conveys the lowest.

Table 8 shows that theproposed method's superior accuracy and explainable framework offer practical benefits for guava cultivators, enabling early disease detection and management. Its agro-based automation system, leveraging locally sourced data, and the XAI-based technique for illness identification contribute to sustainable farming practices.

Table 8. Comparison of related works with the proposed method

Research work	Dealing object	Class	Dataset size	Segmentation algorithm	Feature selection	Set of features	XAI	Employed classifier	Accuracy (%)
[26]	Dragon root and leaf	Three	81	K-means clustering	Color moment and GLCM	–	No	SVM, KNN	87.50
[27]	–	Binary	1000	–	–	–	No	KNN, SGD, RF, CNN-SGD	93.60
[28]	Cauliflower	Four	766	K-means clustering	–	10	No	RF	88.61
[29]	Rice	Ten	500	–	–	–	No	CNN	95.48
[30]	Tomato	Binary	800	Manual cropping	–	36	No	Linear SVM	93.25
[31]	Jackfruit	Four	480	K-means clustering	–	10	No	RF	89.59
[32]	Apple, banana, tomato	Three	63	Modified K-means, Otsu	–	–	No	–	–
[33]	Fruit and vegetable	Fifteen	2612	K-means clustering	–	2	No	SVM	93.77
[34]	Paddy	Three	–	Local entropy threshold	–	–	No	–	94.70
[35]	Cauliflower	Four	708	K-means clustering	ANOVA	10	No	LR	90.77
Proposed method (this work)	Guava fruit and leaf	Binary	526	K-means clustering	ANOVA, LASSO	10 (7 selected)	Yes	AdaBoost (best), RF, SVM, KNN, DT, LR	88.68

5. CONCLUSION

This study developed an agro-based feature selection technique to identify guava diseases, utilizing 526 images to extract two distinct feature types from segmented data. Two feature selection models, ANOVA and LASSO, were employed to rank and select the top five out of ten features. Six ML algorithms, tailored to feature rankings, were tested on sets of seven, nine, and ten features. The AdaBoost classifier achieved the highest accuracy of 88.68% with the top seven features using LASSO, and 85.85% with the top

seven and nine features using ANOVA, outperforming other methods. Two XAI methodologies were applied to enhance model interpretability, reinforcing its reliability. Despite these achievements, the study faces limitations, including dependence on representative data, sensitivity to algorithmic choices, and environmental variability, which may influence the model's applicability and stability. These problems happen because the performance of a model depends on how diverse the data is, how the algorithms are designed, and how things change in the actual world. Also, using the model in real-time applications like an Android app is hard because it needs to be fast, scalable, and able to handle different field circumstances. The models were trained on controlled images; thus, they might not work as well in real life when the lighting, backdrop, or severity of the disease changes. These challenges highlight the need for robust data collection and adaptable algorithms to ensure practical applicability in agricultural disease management. Looking ahead, this research offers significant potential for broader agricultural applications. While focused on guava diseases, the methodology could be extended to detect and classify diseases in other crops across diverse regions. Future work should focus on refining the proposed approach, addressing its limitations, and developing scalable solutions. Future work will focus on making datasets more diverse, improving algorithms, and coming up with adaptive deployment strategies that will help solve these problems, providing timely and actionable insights to enhance disease management and improve agricultural outcomes.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Sujon Chandra Sutradhar	✓	✓	✓	✓	✓	✓		✓	✓	✓		✓		
Md. Mehedi Hasan		✓	✓			✓		✓	✓		✓		✓	

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

Data available on interest due to privacy/ethical restrictions.

REFERENCES

[1] R. Shakil, B. Akter, A. Rajbongshi, U. Sara, M. R. Barman, and A. Dhali, "A transfer learning approach to the development of an automation system for recognizing guava disease using CNN models for feasible fruit production," in *Lecture Notes in Networks and Systems*, vol. 647 LNNS, 2023, pp. 127–141, doi: 10.1007/978-3-031-27409-1_12.

[2] S. Warnick *et al.*, "Explainable AI: motivations and connections with system identification," *IFAC-PapersOnLine*, vol. 58, no. 15, pp. 502–507, 2024, doi: 10.1016/j.ifacol.2024.08.579.

[3] I. Z. Mukti and D. Biswas, "Transfer learning based plant diseases detection using ResNet50," in *2019 4th International Conference on Electrical Information and Communication Technology (EICT)*, IEEE, Dec. 2019, pp. 1–6, doi: 10.1109/EICT48899.2019.9068805.

[4] M. R. Howlader, U. Habiba, R. H. Faisal, and M. M. Rahman, "Automatic recognition of guava leaf diseases using deep convolution neural network," in *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, IEEE, Feb. 2019, pp. 1–5, doi: 10.1109/ECACE.2019.8679421.

[5] G. Geetharamani and J. A. Pandian, "Identification of plant leaf diseases using a nine-layer deep convolutional neural network," *Computers & Electrical Engineering*, vol. 76, pp. 323–338, Jun. 2019, doi: 10.1016/j.compeleceng.2019.04.011.

[6] M. Turkoglu, D. Hanbay, and A. Sengur, "Multi-model LSTM-based convolutional neural networks for detection of apple diseases and pests," *Journal of Ambient Intelligence and Humanized Computing*, vol. 13, no. 7, pp. 3335–3345, Jul. 2022, doi: 10.1007/s12652-019-01591-w.

[7] L. Jupudi, "Image classification algorithm on oranges to perceive sweetness using deep learning techniques," in *AICTE Sponsored National Level E-Conference on Machine Learning as a Service for Industries MLS*, 2021.
[8] K. Trang, L. Tonthat, N. Gia Minh Thao, and N. Tran Ta Thi, "Mango diseases identification by a deep residual network with contrast enhancement and transfer learning," *2019 IEEE Conference on Sustainable Utilization and Development in Engineering and Technologies, CSUDET 2019*, pp. 138–142, 2019, doi: 10.1109/CSUDET47057.2019.9214620.
[9] M. Nikhitha, S. Roopa Sri, and B. Uma Maheswari, "Fruit recognition and grade of disease detection using inception V3 model," *Proceedings of the 3rd International Conference on Electronics and Communication and Aerospace Technology, ICECA 2019*, pp. 1040–1043, 2019, doi: 10.1109/ICECA.2019.8822095.
[10] J. Ma, K. Du, F. Zheng, L. Zhang, Z. Gong, and Z. Sun, "A recognition method for cucumber diseases using leaf symptom images based on deep convolutional neural network," *Computers and Electronics in Agriculture*, vol. 154, pp. 18–24, 2018, doi: 10.1016/j.compag.2018.08.048.
[11] R. M. Prakash, G. P. Saraswathy, G. Ramalakshmi, K. H. Mangaleswari, and T. Kaviya, "Detection of leaf diseases and classification using digital image processing," *Proceedings of 2017 International Conference on Innovations in Information, Embedded and Communication Systems, ICIIECS 2017*, vol. 2018–January, pp. 1–4, 2017, doi: 10.1109/ICIIECS.2017.8275915.
[12] H. N. Al Buhaisi, "Image-based pineapple type detection using deep learning," *International Journal of Academic Information Systems Research*, vol. 5, no. 1, pp. 2643–9026, 2021.
[13] M. Elleuch, F. Marzougui, and M. Kherallah, "Diagnostic method based dl approach to detect the lack of elements from the leaves of diseased plants," *International Journal of Hybrid Intelligent Systems*, vol. 17, no. 1-2, pp. 33–42, 2021, doi: 10.3233/HIS-210002.
[14] A. M. Mostafa, S. A. Kumar, T. Meraj, H. T. Rauf, A. A. Alnuaim, and M. A. Alkhayyal, "Guava disease detection using deep convolutional neural networks: A case study of guava plants," *Applied Sciences (Switzerland)*, vol. 12, no. 1, 2022, doi: 10.3390/app12010239.
[15] M. T. Habib, M. J. Mia, M. S. Uddin, and F. Ahmed, "An explorative analysis on the machine-vision-based disease recognition of three available fruits of Bangladesh," *Vietnam Journal of Computer Science*, vol. 9, no. 2, pp. 115–134, 2022, doi: 10.1142/S2196888822500087.
[16] S. Fadnavis, "Image interpolation techniques in digital image processing: an overview," *International Journal of Engineering Research and Applications*, vol. 4, pp. 70–73, 2014.
[17] H. D. Cheng and X. J. Shi, "A simple and effective histogram equalization approach to image enhancement," *Digital Signal Processing*, vol. 14, no. 2, pp. 158–170, Mar. 2004, doi: 10.1016/j.dsp.2003.07.002.
[18] A. Likas, N. Vlassis, and J. J. Verbeek, "The global k-means clustering algorithm," *Pattern Recognition*, vol. 36, no. 2, pp. 451–461, Feb. 2003, doi: 10.1016/s0031-3203(02)00060-2.
[19] A. Rajbongshi, A. A. Biswas, J. Biswas, R. Shakil, B. Akter, and M. R. Barman, "Sunflower diseases recognition using computer vision-based approach," *IEEE Region 10 Humanitarian Technology Conference, R10-HTC*, vol. 2021–September, 2021, doi: 10.1109/R10-HTC53172.2021.9641588.
[20] S. Khalid, T. Khalil, and S. Nasreen, "A survey of feature selection and feature extraction techniques in machine learning," *Proceedings of 2014 Science and Information Conference, SAI 2014*, pp. 372–378, 2014, doi: 10.1109/SAI.2014.6918213.
[21] J. Cai, J. Luo, S. Wang, and S. Yang, "Feature selection in machine learning: A new perspective," *Neurocomputing*, vol. 300, pp. 70–79, Jul. 2018, doi: 10.1016/j.neucom.2017.11.077.
[22] K. Singh, N. Paliwal, and K. Kasamias, "Surface roughness characterization using representative elementary area (REA) analysis," *Scientific Reports*, vol. 14, no. 1, 2024, doi: 10.1038/s41598-024-52329-4.
[23] L. St»hle and S. Wold, "Analysis of variance (ANOVA)," *Chemometrics and Intelligent Laboratory Systems*, vol. 6, no. 4, pp. 259–272, Nov. 1989, doi: 10.1016/0169-7439(89)80095-4.
[24] R. Muthukrishnan and R. Rohini, "LASSO: a feature selection technique in predictive modeling for machine learning," in *2016 IEEE International Conference on Advances in Computer Applications (ICACA)*, IEEE, Oct. 2016, pp. 18–20, doi: 10.1109/ICACA.2016.7887916.
[25] T. Etem, "Interpretable machine learning for battery health insights: A LIME and SHAP-based study on EIS-derived features," *Bulletin of the Polish Academy of Sciences: Technical Sciences*, vol. 73, no. 5, 2025, doi: 10.24425/bpasts.2025.155033.
[26] L. Hakim, S. P. Kristanto, D. Yusuf, M. N. Shodiq, and W. A. Setiawan, "Disease detection of dragon fruit stem based on the combined features of color and texture," *INTENSIF: Jurnal Ilmiah Penelitian dan Penerapan Teknologi Sistem Informasi*, vol. 5, no. 2, pp. 161–175, 2021, doi: 10.29407/intensif.v5i2.15287.
[27] A. BaniMustafa, H. Qattous, I. Ghabeish, and M. Karajeh, "A machine learning hybrid approach for diagnosing plants bacterial and fungal diseases," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 1, pp. 912–921, 2023, doi: 10.14569/IJACSA.2023.0140198.
[28] A. Rajbongshi, M. E. Islam, M. J. Mia, T. I. Sakif, and A. Majumder, "A comprehensive investigation to cauliflower diseases recognition: an automated machine learning approach," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 12, no. 1, pp. 32–41, 2022, doi: 10.18517/ijaseit.12.1.15189.
[29] Y. Lu, X. Tao, F. Jiang, J. Du, G. Li, and Y. Liu, "Image recognition of rice leaf diseases using atrous convolutional neural network and improved transfer learning algorithm," *Multimedia Tools and Applications*, vol. 83, no. 5, pp. 12799–12817, 2024, doi: 10.1007/s11042-023-16047-9.
[30] U. Mokhtar *et al.*, "SVM-based detection of tomato leaves diseases," *Advances in Intelligent Systems and Computing*, vol. 323, pp. 641–652, 2015, doi: 10.1007/978-3-319-11310-4_55.
[31] M. T. Habib, M. J. Mia, M. S. Uddin, and F. Ahmed, "An in-depth exploration of automated jackfruit disease recognition," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 4, pp. 1200–1209, 2022, doi: 10.1016/j.jksuci.2020.04.018.
[32] L. J. Rozario, T. Rahman, and M. S. Uddin, "Segmentation of the region of defects in fruits and vegetables," *International Journal of Computer Science and Information Security* (*IJCSIS*), vol. 14, no. 5, pp. 399–406, 2016.
[33] S. R. Dubey and A. S. Jalal, "Fruit and vegetable recognition by fusing colour and texture features of the image using machine learning," *International Journal of Applied Pattern Recognition*, vol. 2, no. 2, p. 160, 2015, doi: 10.1504/ijapr.2015.069538.
[34] N. N. Kurniawati, S. N. H. S. Abdullah, S. Abdullah, and S. Abdullah, "Investigation on image processing techniques for diagnosing paddy diseases," *SoCPaR 2009 - Soft Computing and Pattern Recognition*, pp. 272–277, 2009, doi: 10.1109/SoCPaR.2009.62.
[35] R. Shakil, B. Akter, F. M. J. M. Shamrat, and S. R. H. Noori, "A novel automated feature selection based approach to recognize cauliflower disease," *Bulletin of Electrical Engineering and Informatics (BEEI)*, vol. 12, no. 6, pp. 3541–3551, 2023, doi: 10.11591/eei.v12i6.5359.

BIOGRAPHIES OF AUTHORS

Sujon Chandra Sutradhar is currently serving as a lecturer in mathematics, Department of General Education at University of Frontier Technology, Bangladesh. He previously held positions as an adjunct Faculty Member in the Department of Mathematics and Natural Sciences at BRAC University, and as a full-time lecturer in mathematics in the Department of Electrical and Electronic Engineering at Southeast University. He completed both his Bachelor's and Master's degrees in Applied Mathematics from the University of Dhaka, where he achieved first class, second position in both programs. He is currently serving as the chief researcher on a UGC-funded project and is actively engaged in several collaborative research initiatives with fellow academics. His research interests include: machine learning, deep learning, computer vision, financial mathematics, and mathematical biology. He can be contacted at email: sujon0001@uftb.ac.bd.

Md. Mehedi Hasan received his B.S. (Hons.) degree in Mathematics and M.S. in Pure Mathematics from the University of Dhaka. He is working as a lecturer in the Department of General Education, University of Frontier Technology, Bangladesh. His research interest is in mathematical programming and different areas of operations research, optimization and numerical analysis. He can be contacted at email: mehedi0001@uftb.ac.bd.