Journal of King Saud University – Computer and Information Sciences 34 (2022) 7212–7224

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

Disease detection of apple leaf with combination of color segmentation and modified DWT

Sharad Hasan [a,*], Sarwar Jahan [b], Md. Imdadul Islam [a]

[a] Department of Computer Science and Engineering, Jahangirnagar University, Dhaka 1342, Bangladesh
[b] Department of Electronics and Communications Engineering, East West University, Dhaka 1212, Bangladesh

ARTICLE INFO

Article history:
Received 1 February 2022
Revised 11 June 2022
Accepted 4 July 2022
Available online 8 July 2022

Keywords:
Average color marker
Nearest neighbor
Pixel classification
Wavelet coefficients
$L^*a^*b^*$ color histogram
Random Forest

ABSTRACT

In this paper, we proposed a machine learning and computer vision-based automated apple disease detection and recognition system based on leaf symptoms. The proposed method is composed of three parts: diseased region segmentation, feature extraction, and classification. We have segmented the infected portion of the leaf using $L^*a^*b^*$ space-based color segmentation method. Here, average color markers in a^*b^* space and the nearest neighbor method have been used for classifying each pixel into either healthy, infected, or background regions. We have extracted two types of features: one is the proposed DWT feature and another is $L^*a^*b^*$ space-based color histogram features. Horizontal feature fusion is performed to create the final feature vector. The feature vectors have been classified using several classifiers keeping Random Forrest as the base classifier. In this paper, the experiment is made on Plant Village dataset, where image of Apple Scab, Black Rot, and Cedar Apple Rust disease are taken for both training and testing our model. The fusion of proposed DWT and color histogram features is a novel approach in detecting and recognizing apple leaf disease, which got an accuracy of 98.63%.

1. Introduction

Agriculture is the main foundation to ensure food safety all over the world. It also grasps a lion's share in the world's economy. However, in recent years, infection of diseases on plants is rising because of continuous deterioration of environment. The world agricultural industry is facing a huge loss each year because of disease infection into plants. The traditional diagnosis method of manually inspecting the trees is costly, inefficient, and difficult. To remove these problems, the need for an effective and efficient automated disease detection system is increasing to detect the diseases in an early stage and safeguard crops from production failure as well as quality degradation. There exist more than 100 types of apple disease (Chuanlei et al., 2017) and the most prominent diseases are Black Rot, Apple Scab, Cedar Apple Rust, etc. Fruit quality and production are degraded due to these disease infections.

The traditional method of disease diagnosis is observing the symptoms and spots in naked eye and taking opinions from the experts. However, experts are not always available which makes this process inconvenient and expensive therefore increases production cost (Al Bashish et al., 2011). This manual disease identification process is inefficient because of the limitation of the human eye's perception capability. This process is also very difficult and impractical while inspecting large apple orchards. This makes automatic detection and recognition of apple diseases an active research area as well as a challenging aspect in the field of machine learning and computer vision (Zhang et al., 2018).

Almost all types of apple diseases affect the leaves and thus leaves become a primary and effective source of information to detect diseases as discussed in (Vishnu and Ranjith Ram, 2015; Wang et al., 2009). Early detection of diseases can be helpful to reduce economic catastrophe, increase productivity and improve the quality of the produced fruits.

The domain of automation in agriculture is very challenging because of various global factors and changes in the environment. Some common challenges faced by the researchers under computer vision and machine learning are: (a) Accurate segmentation of disease-infected regions as they show various colors, textures,

[*] Corresponding author at: Department of Computer Science and Engineering, Jahangirnagar University, Bangladesh.

E-mail address: hasansharad@gmail.com (S. Hasan).

Peer review under responsibility of King Saud University.

ELSEVIER Production and hosting by Elsevier

https://doi.org/10.1016/j.jksuci.2022.07.004
1319-1578/© 2022 The Author(s). Published by Elsevier B.V. on behalf of King Saud University.

and shapes. It is also challenging because of the presence of multiple disease symptoms on a single image, cluttered and complex background, the similarity between diseased spots and background, poor and uneven lighting condition, and presence of heavy noise. (b) Extraction of appropriate and sufficient number of features to represent the diseased image. A myriad of features is available to be extracted but all features are not suitable for a particular task. It is challenging to create a robust feature set for disease classification. (c) Select a good and robust machine learning algorithm that can classify the diseases based on the extracted feature and produce a high level of accuracy. (d) Development of mathematical model in combining several ML algorithms to enhance the accuracy of detection.

In this paper we consider three disease classes: Apple Scab, Black Rot, and Cedar Apple Rust. The proposed method is composed of three main parts: diseased region segmentation, feature extraction, and classification. Major contributions of the paper are listed below:

a) Proposed a $L^*a^*b^*$ color space-based segmentation algorithm that does not require any preprocessing of images. Our segmentation algorithm excludes lightness channel (L^*- channel) which handles uneven lighting and exposure situation without any preprocessing. Three color markers on a^*b^* color space are created for healthy, diseased, and background regions. The a^*b^* space based color markers resemble to human viewing perception and detect small differences in color. The nearest neighbor method is then applied using proposed color markers to classify each pixel of the image.
b) Proposed a modified method of extracting DWT coefficients and used them as features. Multilevel discrete wavelet transformation is performed along each row of the image matrix until a single coefficient is remaining from each row. In this process we have ignored detail coefficients and considered approximation coefficients only. The single coefficients are then combined as a discretely sampled signal and multilevel discrete wavelet transform is performed again to get the final feature vector. The DWT feature vector derived by our proposed method is less susceptible to noise and has performed well despite the presence of peeper noise in the dataset.
c) Proposed a $L^*a^*b^*$ space-based color histogram method to extract color features from the images. We have used bin size = 8, which is empirically derived and has provided better results. Then DWT features and color histogram features are fused horizontally which is a novel fusion.
d) Tested several classifiers on different forms of feature vector and achieved the best accuracy for Random Forrest classifiers with 100 trees.

2. Literature review

In recent years, research in the field of automated plant disease detection is on the growth to provide an effective and efficient method for early detection and diagnosis of plant diseases. A plethora of methods combining computer vision, image processing and machine learning have been proposed by researchers in this respect. Automated systems are proposed to detect diseases in plants such as Tomato (Basavaiah and Anthony, 2020; Khan and Narvekar, 2020), Mango (Ramírez Alberto et al., 2022; Singh et al., 2019), Papaya (Habib et al., 2020), Olive (Cruz et al., 2017; Sinha and Shekhawat, 2020), Rice (Jiang et al., 2020; Li et al., 2020) Cassava (Abayomi-Alli et al., 2021), Palm (Hamdani et al., 2021) and so on. From various literatures, it is prominent that, plant detection methods generally have three parts: segmentation of diseased region from the leaf or fruit image, extracting effective and useful features from the segmented image, and classifying the features. Researchers have proposed different techniques for these parts. Some state of the art methods is discussed and summarized here.

Khan et al. (2019) have proposed an optimized method for image segmentation and feature selection to detect Apple leaf diseases. 3D-Box filter, de-correlation, 3D-Gaussian, and 3D-median filter have been used to enhance disease spots. The strong Correlated Pixels (SCP) based method optimized by the Expectation-Maximization (EM) is used for diseased region segmentation. LBP and color histogram-based features are extracted which are then selected by a genetic algorithm (GA).

Chuanlei et al. (2017) tried to create a globally optimum feature set for apple leaf disease detection combining genetic algorithm and correlation-based feature selection (CFS). They have removed image background using a histogram-based method with an empirically derived threshold. The diseased region is then segmented with the Region Growing Algorithm (RGA), aided by a quad-tree-based merging and splitting method. Authors have fused color features in RGB and HIS space, shape features, and texture features from Spatial Gray-level Dependence Matrices (SGDM).

Bracino et al. (2020) have used graph cut segmentation aided by lazy snapping process to separate background from the Apple leaf images. They have not segmented the diseased spots. Features are extracted from the whole image. Authors have extracted a total of 12 color and texture features from which only three features are selected using Neighborhood Component Analysis (NCA).

Dubey and Jalal (2016) have used fruit images instead of leaf images for Apple disease detection. After converting the images into L*a*b* space, K-means clustering is applied on a*b* space and four regions are created. The region with most of the infectious parts is then selected. They have extracted color features using a global color histogram (GCH) and color coherence vector (CCV), texture feature using LBP, and shape features using Zernike moments (ZM). K-means clustering is a widely used method for leaf disease segmentation. Several researchers (Habib et al., 2020; Kumari et al., 2019; Padol and Yadav, 2016; Zhang et al., 2018) have also used K-means clustering method in their studies.

Pravin Kumar et al. (2021) has separated background and foreground using Gaussian Mixture Model (GMM) which works based on Gaussian distribution of each pixel in the frame. Diseased regions are segmented using Particle Swarm Optimization (PSO) based Fuzzy C-means algorithm. Vein, shape, edge-based and texture features are extracted. As a classifier they have proposed Multi-Kernel Parallel Support Vector Machine (MK-PSVM). Apple leaves are classified as only diseased or healthy in their paper.

Khan and Narvekar (2020) proposed a method for the segmentation of tomato leaf diseases in cluttered and natural backgrounds. Simple Linear Iterative Clustering (SLIC) is used to generate 100 superpixels. The background is then discarded from this superpixel image using two proposed constraints. K-means clustering is then used to extract disease-infected regions. Their method extracted three types of features from multiple datasets: PHOG spatial features, GLCM texture features, and statistical features. On the other hand Basavaiah and Anthony (2020) have not performed any segmentation on Tomato leaves. They have directly extracted color histogram features, Hu Moments shape features and GLCM texture features. A combination of LBP and HOG descriptors is proposed in their study. For classifying the extracted features, they have used decision tree and random forest classifier.

Abisha and Jayasree (2019) have applied two different segmentation methods namely, Delta E (DE) and Expectation-Maximization (EM) on brinjal leaves. Authors extracted discrete wavelet transformation (DWT) coefficients from differently segmented images and fused those according to their proposed fusion rule. Then, a single segmented image is constructed from the DWT coefficients using inverse DWT. They extracted color, textural and structural features and classified those using ANN.

Ali et al. (2017) have made use of a template containing the symptoms of specific diseases to segment disease-affected regions in Citrus leaf. Energy difference between the enhanced $L^*a^*b^*$ color space image and the template is calculated. Then Empirically derived tolerance is applied to segment the images. Histogram-based color features, LBP texture features and principal component analysis (PCA) is incorporated for feature extraction and selection. Sharif et al. (2018) have considered both citrus leaf and fruit for disease detection. Images are enhanced using the Gaussian function and Top-Hat filter. Pixels are clustered using threshold based Chi-square distance. Their threshold function is optimized using the HDCT saliency method. The authors have extracted color features, GLCM based texture features, and geometric features. Optimum 100 features have been selected by combining PCA, skewness, and entropy.

Almadhor et al. (2021) have used ΔE method for segmenting diseased region in Guava leaf. Color features are extracted from RGB and HSV histograms. LBP features are extracted as textural feature. Different combinations of these features are classified using various ML classifiers with good image level and disease level accuracy.

Adeel et al. (2019) has proposed a Local Contrast Haze Reduction (LCHR) approach for enhancing grape leaf images. For disease segmentation they have decomposed input image into L^*, a^* and b^* channels and selected the best channel using a proposed weight function. Finally, additional pixel removal using some morphological operations are done before mapping. Color, LBP and geometrical features are extracted and fused using Canonical Correlation Analysis (CCA). Feature reduction is done using Neighborhood Component Analysis (NCA) and classified using MSVM.

Capizzi et al. (2016) proposed an automatic orange defect classification system on production line. They have separated orange from the background using specific threshold value in Saturation histogram. Five textural features are extracted from co-occurrence matrix created for each channel Hue, Saturation and Value. Authors have proposed an RBPNN model to classify the extracted features to four defect and one normal class.

Zhang et al. (2018) proposed a diseased lesion segmentation method combining super-pixel and K-means clustering. They divided the whole image into some super-pixels using the SLIC method. Then K-mans is applied to each super-pixel to classify the pixels as either diseased or non-diseased. Then features are extracted from disease segmented images using PHOG descriptors.

Yadav et al. (2019) have proposed a method for multiple leaf disease classification method. They have taken 8750 images of 23 classes from Plant Village dataset. Pre-trained AlexNet is used for feature extraction. Removing the 1000-way SoftMax layer of AlexNet, they have extracted 100 features. From these features, 34 best descriptive features are selected using proposed Particle Swarm Optimization (PSO). Multiple classifiers are then used for classification.

Besides these traditional ML methods, a plethora of deep learning-based models is proposed to detect apple disease. Rehman et al. (2021) have proposed a hybrid contrast stretching method for better distinguishability of diseased spots in Apple leaf. A modified Mask RCNN is used for spot segmentation. After extracting features with Transfer Learning, Kapur entrpy along with SVM is used for feature selection.

Luo et al. (2021) have proposed an improved ResNet architecture by improving the information flow to detect apple leaf disease. They have solved the problem of information loss in the ResNet down-sample by separating channel projection and spatial projection. The authors also replaced the 3 × 3 convolution layer with the pyramid convolution for improved performance.

Bi et al. (2020) have used MobileNet architecture to propose a lightweight detection model with high precision. Zhong and Zhao (2020) have used DenseNet-121 as their backbone model and modified the fully connected layer to achieve regression. Their focal loss function-based model has outperformed traditional cross-entropy loss, function-based models. Yu and Son (2020) have proposed a leaf spot attention network (LSA-Net), where they have created two subnetworks for better discriminative features and improved accuracy. Bansal et al. (2021) have tried to ensemble DenseNet121, EfficientNetB7, and EfficientNet NoisyStudent to increase classification accuracy. Their model can classify leaf images with multiple diseases on them.

Abayomi-Alli et al. (2021) tried to solve the problem of less accuracy due to low quality test image. They proposed a histogram transformation technique based on Chebyshev orthogonal function and PDFs for image augmentation. Authors have also adopted some models for image quality reduction to resemble real world situations.

Table 1 provides a summary of the related studies for plant disease detection along with their limitations.

3. Proposed methodology

3.1. Dataset and data augmentation

In this paper, we used the Plant Village dataset (Hughes and Salathé, 2015) which is an open access repository dataset. It contains leaf images of 14 crops including Apple, Blueberry, Cherry, Corn, etc. Each crop has multiple types of disease images and in total there are 38 classes. We have chosen Apple for our study and selected three class of diseases: Apple Scab, Black Rot, and Cedar Apple Rust. The number of samples in the Apple Cedar Rust class is less than half of the other two classes. This creates an imbalance in the dataset. The imbalanced dataset can cause skewness and bias in classification results. Larger classes can have more emphasis than smaller classes. That's why we applied image augmentation to increase the number of samples of the Cedar Apple Rust class. Outward scaling by 20%, 40-degree clockwise rotation, shifting with width shift = 0.2 and height shift = 0.2, and horizontal flipping have been used to augment image data. In case of outward scaling, the new image is cropped so that it matches the original image size. Disease classes and the associated number of samples before and after augmentation are shown in Table 2. Some samples from our dataset are shown in Fig. 1. First row is Apple Scab, the second is Black Rot and the third row represents Cedar Apple Rust.

3.2. Diseased region segmentation

Segmentation is the process where the disease-infected regions are extracted out from the full image. It is necessary because the extra background information can cause potential inherent bias while extracting features. Our proposed segmentation process can be divided into two parts: a) average color marker creation and b) segmenting diseased region.

3.2.1. Average color marker creation

1. We have selected 100 images from each of the three classes and created a sub-dataset of 300 images. In our sub dataset we have three classes: Apple Scab, Black Rot and Apple Cedar Rust and each class has 100 images. The images are selected empirically

Table 1
Summary of related studies.

Reference	Plant & Dataset	Segmentation	Feature Extraction and Selection	Classification	Limitations
(Khan et al., 2019)	Apple leaf; Plant Village dataset	a) 3D-Box, de-correlation, 3D-Gaussian, 3D-median filters for enhancement b) Strong Correlated Pixels + Expectation Maximization	a) Color histogram + LBP texture b) Genetic Algorithm feature selection	Multiple classifiers	—
(Chuanlei et al., 2017)	Apple leaf; custom dataset	Histogram method + Region Growing Algorithm + Quad tree based merging and splitting	a) RGB, HIS color + shape + SDGM texture features b) GA + Correlation Feature Selection	SVM	a) Very controlled environment (illumination box + CCD camera + software) used for image acquisition which does not resemble natural condition b) Very small dataset with only 90 samples
(Bracino et al., 2020)	Apple leaf; Plant Village dataset	Graph Cut segmentation aided by Lazy Snapping	a) RGB, HSV, L*a*b*, YCbCr color + texture features b) Neighborhood Component Analysis feature selection	KNN and ANN	a) No diseased region segmentation; only background separation b) Very small feature vector with only 3 features
(Dubey and Jalal, 2016)	Apple fruit; custom dataset	K-means Clustering	Global Color Histogram, Color Coherence Vector + LBP texture + Zernike Moments shape features	MSVM	a) Manual selection of segments containing most diseased portions b) Lacks K-fold cross validation
(Pravin Kumar et al., 2021)	Apple leaf; Plant Village dataset	Gaussian Mixture Model background subtraction + Particle Swarm Optimization (PSO) based Fuzzy C-means diseased region segmentation	Vein + shape + edge-based + GLCM texture features	MK-PSVM	a) presumes uniform standard deviation in the difference in pixel intensity, which is not a natural case b) only classifies healthy or infected leaf, no classification of disease is done
(Khan and Narvekar, 2020)	Tomato leaf; Plant Village, Internet downloaded and custom dataset	a) Simple Linear Iterative Clustering + HOG for background subtraction b) K-means clustering for region segmentation	PHOG spatial + GLCM texture + statistical feature	Multiple classifiers	Lacks K-fold cross validation
(Basavaiah and Anthony, 2020)	Tomato leaf; custom dataset	Not applicable	Color histogram + Hu Moments shape + GLCM, LBP texture features	Random Forest	No image segmentation is performed
(Abisha and Jayasree, 2019)	Brinjal leaf; custom dataset	a) Delta E + Expectation Maximization b) DWT based image fusion	RGB color + GLCM texture + structural features	ANN	Lacks K-fold cross validation
(Ali et al., 2017)	Citrus leaf; Custom dataset	Delta E + disease symptom template	a) RGB, HSV histogram + LBP texture features b) PCA based feature selection	Multiple classifiers	a) Creating template containing disease symptoms requires expert knowledge b) Dataset size is very small
(Sharif et al., 2018)	Citrus leaf and fruit; Plant Village + Citrus Image Gallery and custom dataset	a) Top-Hat filter + Gaussian function for contrast enhancement b) Chi-square distance + proposed threshold function + HDTC saliency method	a) Color + GLCM texture + geometric feature b) PCA + skewness + entropy based feature selection	SVM	—
(Almadhor et al., 2021)	Guava leaf; custom dataset	Delta E	a) RGB, HSV color histogram + LBP texture b) PCA feature selection	Multiple classifiers	Very large feature vector which can eventually lead to curse of dimensionality
(Adeel et al., 2019)	Grape leaf; Plant Village dataset	a) LCHR contrast enhancement b) LAB color transformation + channel selection using weighted function + morphological operations	a) Color + LBP + Geometric features b) CCL feature fusion + NCA feature selection	Multiple classifiers	Disease segmentation is done on only a single channel (either L^*, a^* or b^*) which may suffer from losing important image information
(Yadav et al., 2019)	Multiple fruit leaf, Plant Village dataset	N/A	a) 100 features from AlexNet b) Particle Swarm Optimization feature selection	Multiple classifiers	No diseased region segmentation is done
(Capizzi et al., 2016)	Orange fruit; custom dataset	Threshold value + saturation histogram	Co-occurrence matrix based textural feature	RBPNN	No diseased region segmentation; only background separation
(Zhang et al., 2018)	Apple and Cucumber leaf; Custom dataset	Super pixel clustering + K-means + PHOG	PHOG features from L*, a*, b* channels	C-SVM	Dataset size is small
(Rehman et al., 2021)	Apple leaf; Plant Village dataset	Hybrid contrast stretching + Mask RCNN	a) Features from ResNet-50b) Kapur entropy along with SVM (EaSVM) feature selection	Multiple classifiers	Lacks large number of ground truth image for mask RCNN.
(Abayomi-Alli et al., 2021)	Cassava leaf; Cassava disease leaf dataset	Not applicable	Deep features from MobileNetV2	Modified MobileNetV2	Since their augmentation method is based on color value distribution, sometimes the synthetic images show unnatural color.

Table 2
Disease classes and number of sample images.

Disease Classes	No. of Samples (before augmentation)	No. of Samples (after augmentation)
Apple Scab	630	630
Black Rot	621	621
Apple Cedar Rust	275	650

so that they can represent the variations in color, illumination, blurriness, and fuzziness of all the images of that specific class in the main dataset.

2. From each image I_{RGB} of our sub-dataset, we have selected three regions of polygonal shape using the *roipoly* function of Matlab. The selected sample regions are indicated as: healthy region R_1, diseased region R_2 and background region R_3. The *roipoly* function returns a mask M_i, i = 1,2,3 for each sample region R_i, i = 1,2,3. M_i is a binary mask of shape 256 × 256. Each element (x,y) of mask M_i follows the rule:

$$M_i(x,y) = \begin{cases} 1, & \textit{if } I_{RGB}(x,y) \in R_i \\ 0, & \textit{otherwise} \end{cases} \quad (1)$$

That means, any pixel (x,y) in binary mask M_i will have a pixel value 1 (white) if the corresponding pixel (x,y) of I_{RGB} falls within our selected region. Otherwise it will have a pixel value 0 (black). Thus, we will have 3 binary masks from a single image I_{RGB} for its three sample regions. Examples of selecting samples regions and associated binary masks are shown in Fig. 2.

3. Next we convert RGB image I_{RGB} into $L^*a^*b^*$ color space image I_{Lab} and decompose it into 3 color channels: I_L, I_a and I_b, where each channel has the shape 256 × 256. Here we made use of only I_a and I_b channels to create average color marker.

4. Then we apply mask M_i on I_a and I_b and perform bitwise logical AND operation. Through this operation we create two column vectors V_{ai} and V_{bi}. Each element (x,1) of the column vectors V_{ai} and V_{bi} is calculated using the following rules:

$$V_{ai}(x,1) = I_a(x,y), \text{ if } M_i(x,y) \neq 0 \quad (2)$$

$$V_{bi}(x,1) = I_b(x,y), \text{ if } M_i(x,y) \neq 0 \quad (3)$$

That means, if an element (x,y) of binary mask M_i is non zero, then the pixel value of corresponding (x,y) element of I_a will be inserted into V_{ai}, otherwise it will not be inserted. Using the same way V_{bi} will also be created. The length of V_{ai} and V_{bi} is equal to the number of non-zero elements of M_i. If the size of V_{ai} and V_{bi} is p, then the average value of V_{ai} and V_{bi} is,

$$\varepsilon_{ai} = \frac{1}{p}\sum_{j=1}^{p} V_{ai}(x_j, 1) \quad (4)$$

$$\varepsilon_{bi} = \frac{1}{p}\sum_{j=1}^{p} V_{bi}(x_j, 1) \quad (5)$$

Thus, we get the color marker $[\,\varepsilon_{ai} \quad \varepsilon_{bi}\,]$ for region R_i in a^*b^* color space. This color marker shows average color in a^*b^* space for each selected region R_i for a single image. Since we have three sample regions R_1, R_2, and R_3, therefore we get three color markers from a single image. The obtained color markers are: healthy color marker $[\,\varepsilon_{a1} \quad \varepsilon_{b1}\,]$, diseased color marker $[\,\varepsilon_{a2} \quad \varepsilon_{b2}\,]$ and background color marker $[\,\varepsilon_{a3} \quad \varepsilon_{b3}\,]$

5. Steps 1–4 are repeated for all 300 images in our sub-dataset. Each image gives us three color markers. Thus we have 300 healthy color markers, 300 diseased color markers and 300 background color markers. Since we have three disease classes namely apple scab, black rot and apple-cedar rust, therefore the

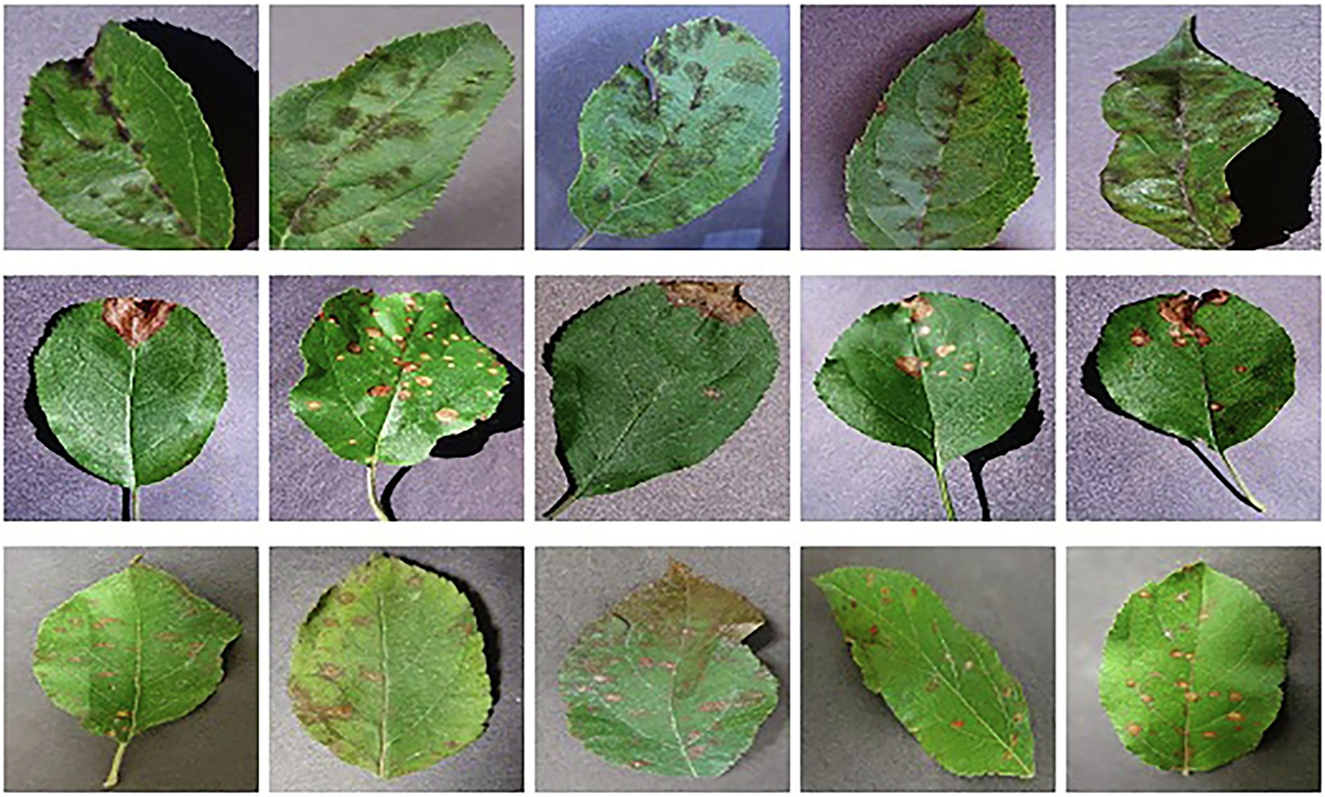

Fig. 1. Sample images from each disease class.

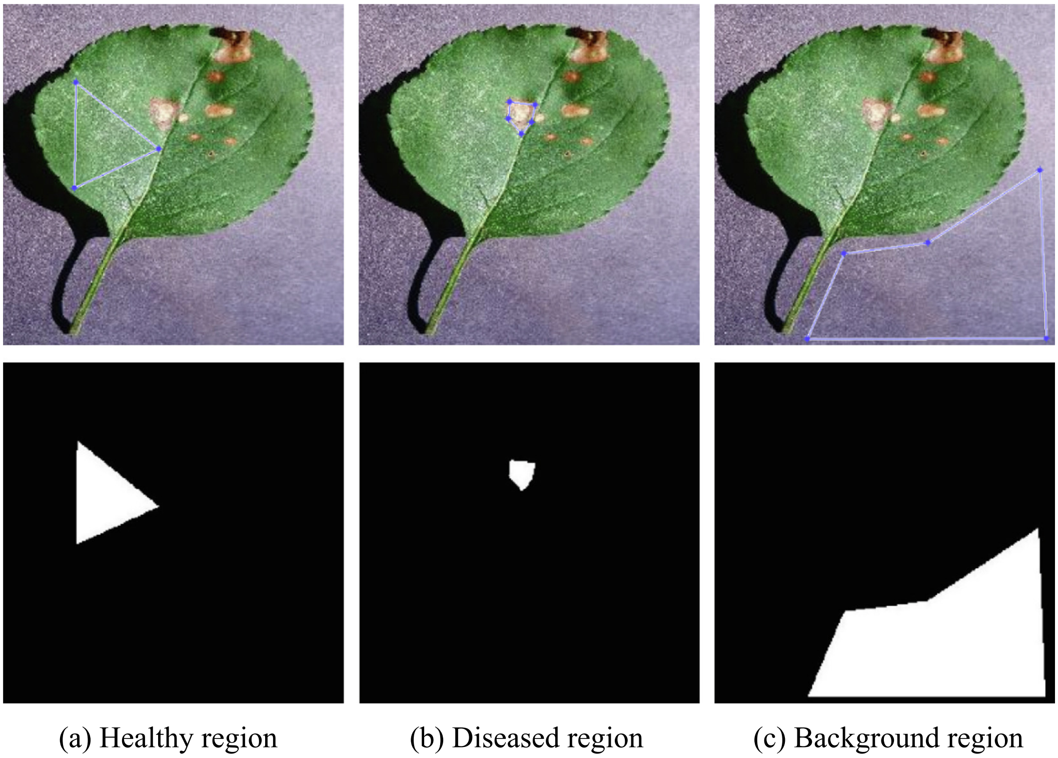
(a) Healthy region (b) Diseased region (c) Background region

Fig. 2. Selection of region using Matlab's *roipoly* function and associated binary masks.

diseased color marker $[\varepsilon_{a2} \quad \varepsilon_{b2}]$ is of three types, which are: $[\varepsilon_{a2}^{scab} \quad \varepsilon_{b2}^{scab}]$, $[\varepsilon_{a2}^{rot} \quad \varepsilon_{b2}^{rot}]$ and $[\varepsilon_{a2}^{rust} \quad \varepsilon_{b2}^{rust}]$. So we have 100 diseased color marker for each disease type. The average color marker is calculated using the following formulas:

$$\text{Average healthy color marker}, [\chi_{a1} \quad \chi_{b1}] = \frac{1}{300}\sum_{j=1}^{300}[\varepsilon_{a1} \quad \varepsilon_{b1}]_j \tag{6}$$

$$\text{Average diseased color marker (apple scab)}, [\chi_{a2}^{scab} \quad \chi_{b2}^{scab}] = \frac{1}{100}\sum_{j=1}^{100}[\varepsilon_{a2}^{scab} \quad \varepsilon_{b2}^{scab}]_j \tag{7}$$

$$\text{Average diseased color marker (black rot)}, [\chi_{a2}^{rot} \quad \chi_{b2}^{rot}] = \frac{1}{100}\sum_{j=1}^{100}[\varepsilon_{a2}^{rot} \quad \varepsilon_{b2}^{rot}]_j \tag{8}$$

$$\text{Average diseased color marker (cedar rust)}, [\chi_{a2}^{rust} \quad \chi_{b2}^{rust}] = \frac{1}{100}\sum_{j=1}^{100}[\varepsilon_{a2}^{rust} \quad \varepsilon_{b2}^{rust}]_j \tag{9}$$

$$\text{Average background color marker}, [\chi_{a3} \quad \chi_{b3}] = \frac{1}{300}\sum_{j=1}^{300}[\varepsilon_{a3} \quad \varepsilon_{b3}]_j \tag{10}$$

Combining all the average color markers into a single matrix Γ, we get.

$$\Gamma = \begin{bmatrix} \chi_{a1} & \chi_{b1} \\ \chi_{a2}^{scab} & \chi_{b2}^{scab} \\ \chi_{a2}^{rot} & \chi_{b2}^{rot} \\ \chi_{a2}^{rust} & \chi_{b2}^{rust} \\ \chi_{a3} & \chi_{b3} \end{bmatrix} \tag{11}$$

3.2.2. Segmenting diseased region

1. The average color marker Γ is used to segment all the images present in main dataset. First, an image I_{RGB} is converted into I_{Lab}. Then each pixel (x, y) of I_{Lab} is classified as either healthy, diseased, or background pixel by the nearest neighbor method using average color marker Γ. To do so, we decompose I_{Lab} into I_L, I_a, and I_b channels.
2. Then, five distances are calculated based on I_a, and I_b. The notations and description of the distances are presented in Table 3.

These distances are calculated using the following formulas:

$$D_1 = \sqrt{\{I_a(x,y) - \Gamma(1,1)\}^2 + \{I_b(x,y) - \Gamma(1,2)\}^2} \tag{12}$$

$$D_2 = \sqrt{\{I_a(x,y) - \Gamma(2,1)\}^2 + \{I_b(x,y) - \Gamma(2,2)\}^2} \tag{13}$$

$$D_3 = \sqrt{\{I_a(x,y) - \Gamma(3,1)\}^2 + \{I_b(x,y) - \Gamma(3,2)\}^2} \tag{14}$$

$$D_4 = \sqrt{\{I_a(x,y) - \Gamma(4,1)\}^2 + \{I_b(x,y) - \Gamma(4,2)\}^2} \tag{15}$$

$$D_5 = \sqrt{\{I_a(x,y) - \Gamma(5,1)\}^2 + \{I_b(x,y) - \Gamma(5,2)\}^2} \tag{16}$$

Now the classification of pixel (x, y) is done using the following rule:

Table 3
The distances and their meaning.

Distance	Description
D_1	Distance between (x, y) and healthy color marker
D_2	Distance between (x, y) and diseased color marker (apple scab)
D_3	Distance between (x, y) and diseased color marker (black rot)
D_4	Distance between (x, y) and diseased color marker (cedar rust)
D_5	Distance between (x, y) and background color marker

$$(x,y) \in \begin{cases} \textit{Healthy region}, \; \textit{if} \; \min(D_1, D_2, D_3, D_4, D_5) = D_1 \\ \textit{Diseased region}, \; \textit{if} \; \min(D_1, D_2, D_3, D_4, D_5) = D_2 \; \textit{or} \; D_3 \; \textit{or} \; D_4 \\ \textit{Background region}, \; \textit{if} \; \min(D_1, D_2, D_3, D_4, D_5) = D_5 \end{cases} \tag{17}$$

3. After classifying all the pixels of an image, we show the diseased pixels in RGB color space discarding healthy and background pixels. Thus we have our diseased region segmented image. Few examples of 'diseased region segmented images' are shown in Fig. 3.

3.3. Feature extraction

3.3.1. Discrete wavelet transformation (DWT) feature extraction

We have used DWT to extract features from the segmented images. The way we have extracted the DWT feature from the images are as follows:

1. At first, we convert the segmented image I_{RGB} into grayscale image I_{gray} with size of 256 × 256.
2. Then we take each row of our image I_{gray} and consider it as a discretely sampled signal or a vector $f(n)$ having shape 1 × 256. Then we pass $f(n)$ through a low pass filter (LPF) and a high pass filter (HPF) simultaneously. Both LPF and HPF filters are defined by the '$db1$' wavelet from the Daubechies wavelets family. A convolution operation is performed between $f(n)$ and the filters. The result of convolution between $f(n)$ and LPF is downsampled by 2 which gives us a vector of coefficients called approximation coefficients with shape 1 × 128. Similarly, we get another vector of coefficients called detail coefficients from HPF filter.
   We ignore the detail coefficients and carry on with approximation coefficients. Approximation coefficients of level 1 are again passed through LPF and HPF of level 2. This time, downsampling the result of LPF filter gives another vector of approximation coefficients with shape 1 × 64. This vector is again passed through LPF and HPF of the next level.
   We continue this process until we have a single coefficient r_i, or in other words, until the shape of the approximation coefficient vector is 1 × 1. Then we store the single coefficient into a row vector R. The whole process is depicted in Fig. 4.
3. We perform step 2 for all the 256 rows of I_{gray} and continue storing the single coefficients into R. Thus, at the end of performing proposed multi-level DWT on 256 rows of I_{gray}, the size of row vector R becomes 1 × 256. The shape of R is like, R = [r_1, r_2, r_3,. . . . ., r_{256},].
4. Then we perform multi-level DWT described in step 2 on R. Thus we create a feature vector of length 2^n, where n = 0,1, 2,. . .,8 by stopping multi-level DWT at level (8-n). In this paper we consider n = 5 i.e. 32 DWT features are extracted using the above method. The overall process of DWT feature extraction is depicted in Fig. 5.

3.3.2. Color histogram feature extraction

The following procedure is followed to extract color histogram features:

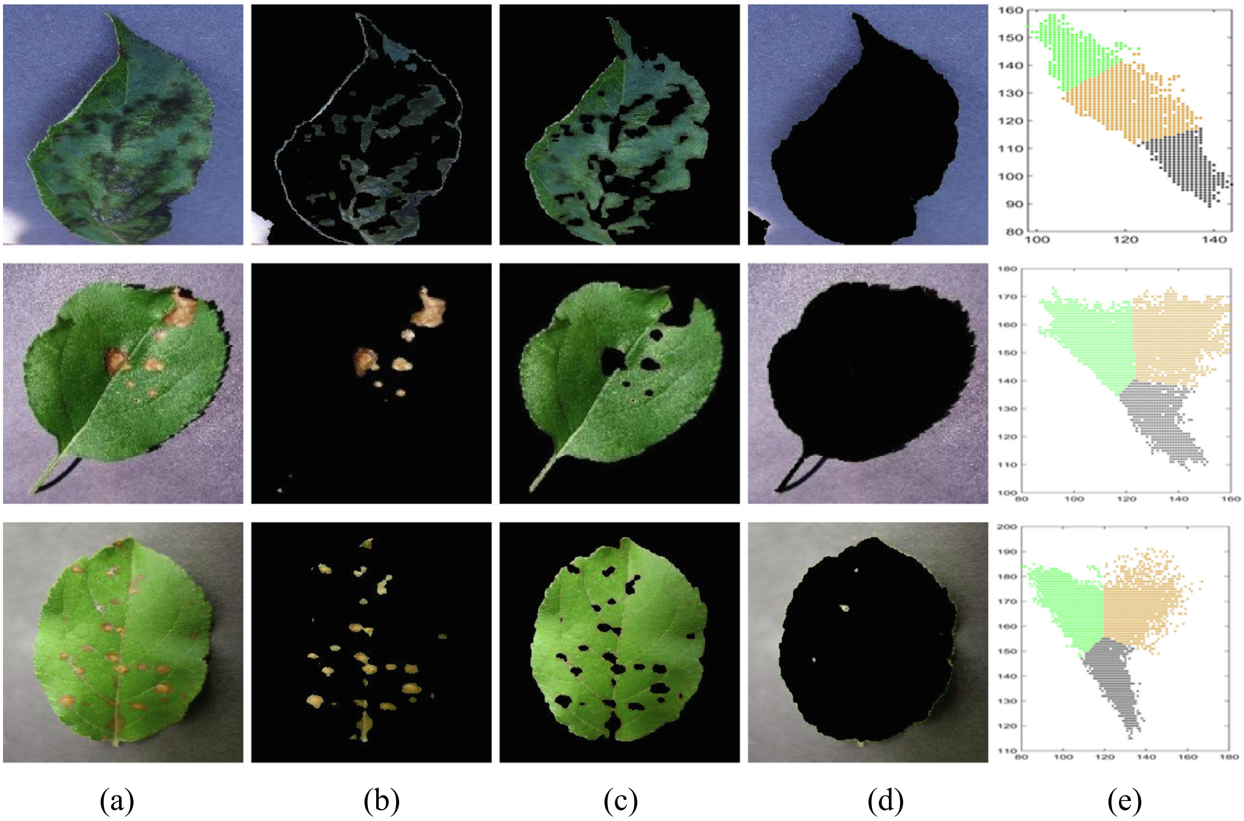

Fig. 3. Sample image segmentation. (a) Original image; (b) Diseased region; (c) Healthy region; (d) Background region; (e) Scatter plot of segmented pixels.

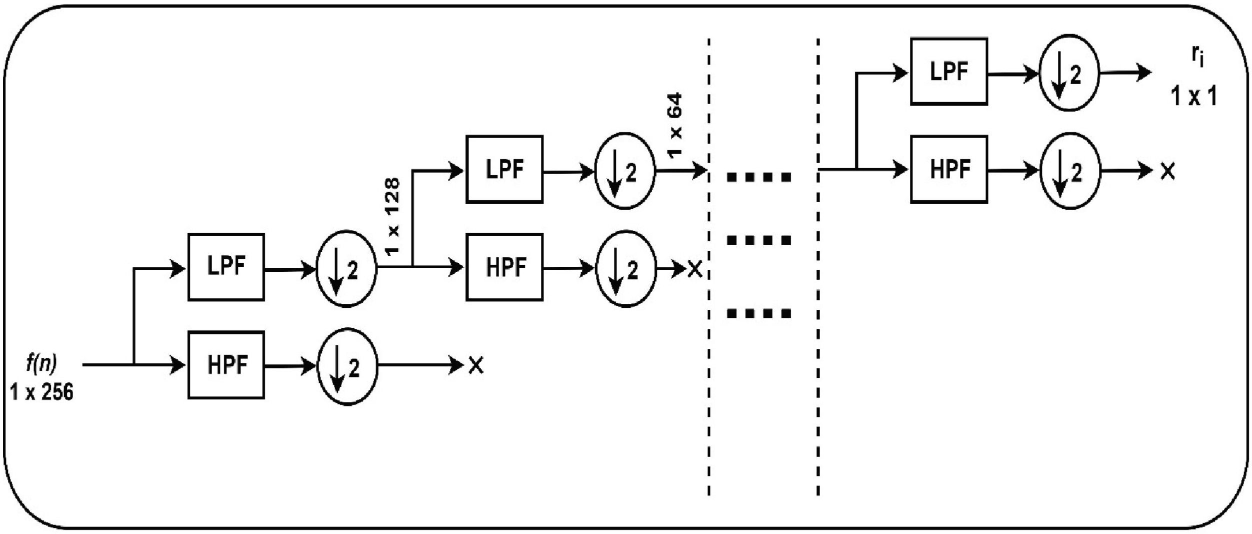

Fig. 4. Multi-level Discrete Wavelet Transformation of a single row of I_{gray}.

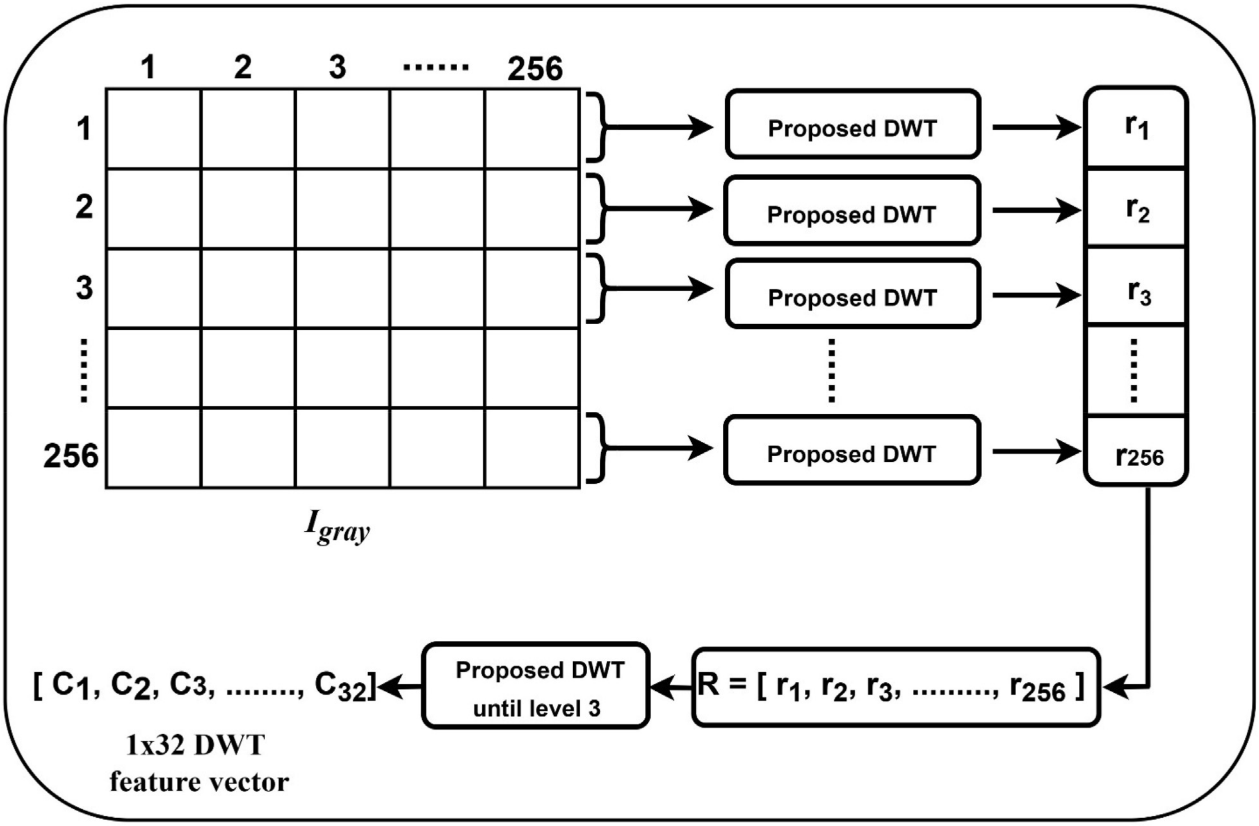
1 2 3 256
1
2
3
256
I_{gray}
Proposed DWT
r1
r2
r3
r256
R = [r1, r2, r3, ..., r256]
Proposed DWT until level 3
[C1, C2, C3, ..., C32]
1x32 DWT feature vector

Fig. 5. Overall process of extracting 32 DWT features.

1. At first, we convert the segmented RGB image I_{RGB} into $L^*a^*b^*$ color space image I_{Lab}. I_{Lab} is then decomposed into three color channels: I_L, $I_{a,}$ and I_b.
2. Next, histogram is calculated for I_L, $I_{a,}$ and I_b channels separately. Each channel has range of pixel values [0 ∼ 256]. We have divided this range into some intervals called bins. Empirically derived bin size = 8 is used in our method. Taking a bin size of 8 we get $2^8/8$ = 32 bins of equal range. If a pixel value falls within the range of any specific bin, then the count of that bin is increased by one. Then we calculate the number of pixels that fall into each of the bins. We got 32 numbers from each color channel which are considered as features for that channel.

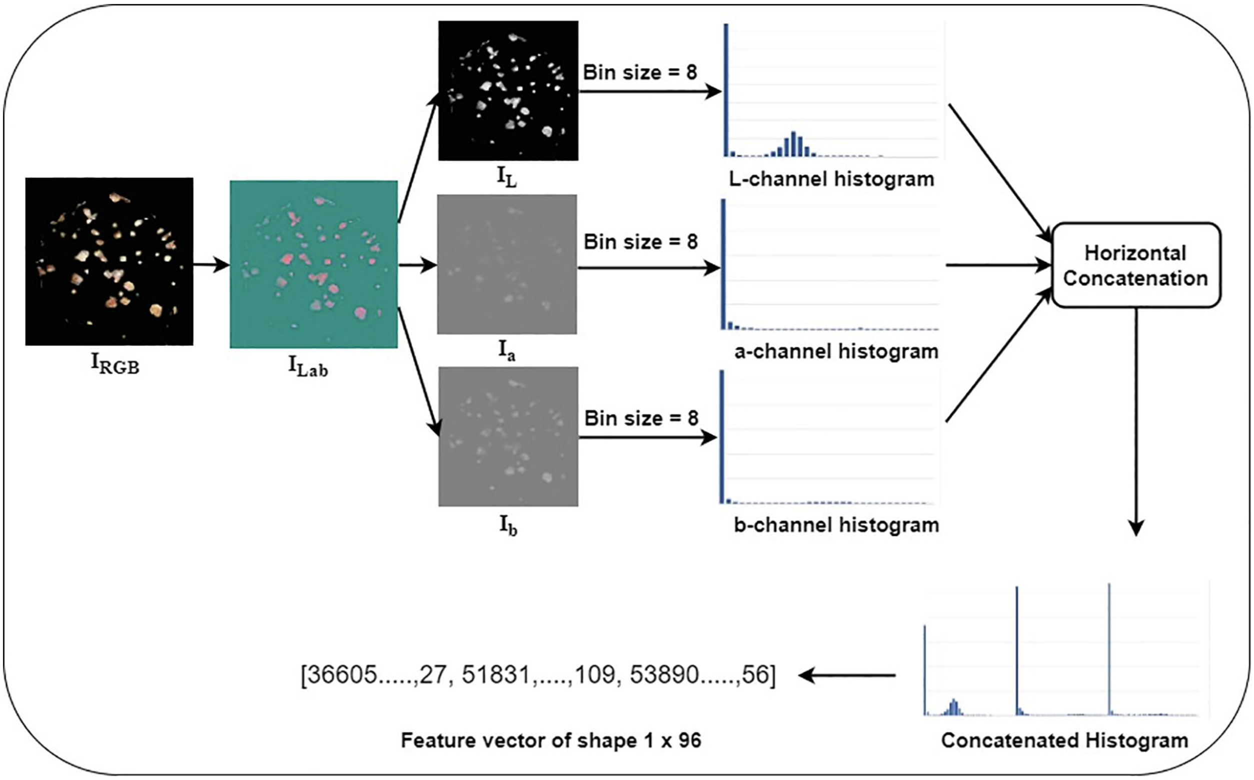

Fig. 6. Process of extracting color histogram features.

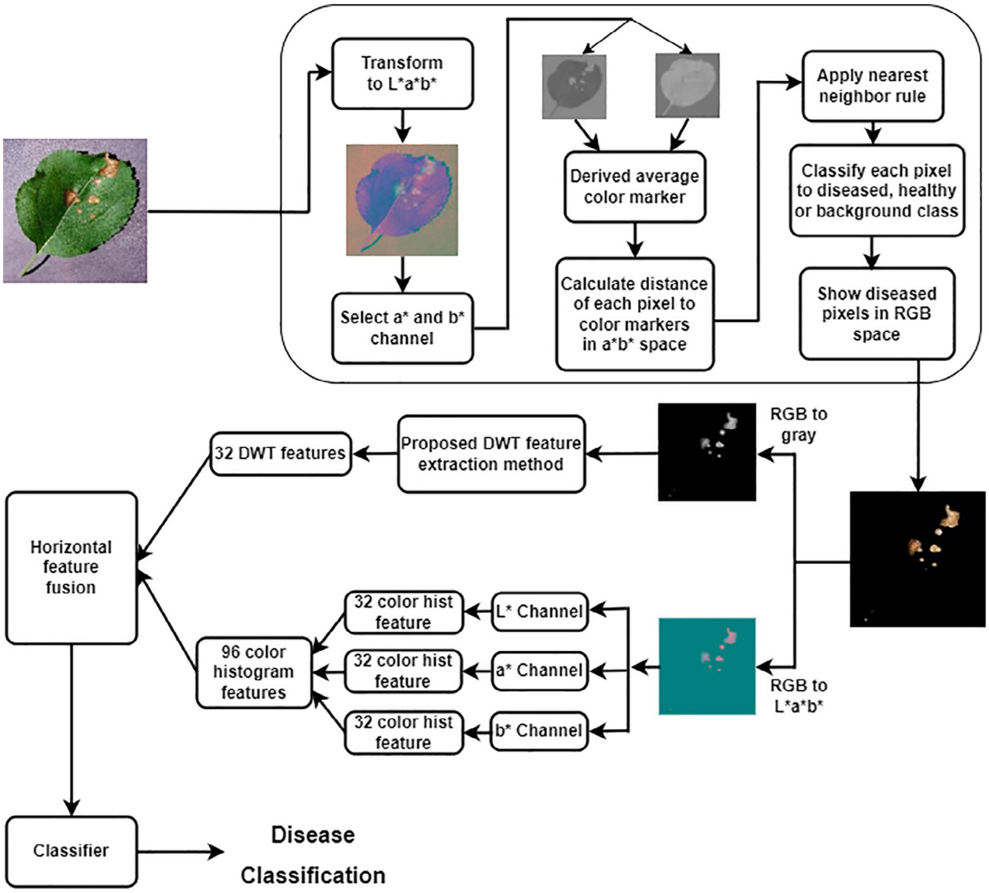
Transform to L*a*b*
Select a* and b* channel
Derived average color marker
Calculate distance of each pixel to color markers in a*b* space
Apply nearest neighbor rule
Classify each pixel to diseased, healthy or background class
Show diseased pixels in RGB space
RGB to gray
Proposed DWT feature extraction method
32 DWT features
RGB to L*a*b*
L* Channel
a* Channel
b* Channel
32 color hist feature
32 color hist feature
32 color hist feature
96 color histogram features
Horizontal feature fusion
Classifier
Disease Classification

Fig. 7. Complete block diagram of our proposed model.

Table 4
Brief description of training and testing samples from selected apple disease classes.

Disease Class	Total image	Training sample	Test sample	Background	Capturing Environment
Apple Scab	630	504	126	Fixed	Controlled
Black Rot	621	496	125	Fixed	Controlled
Cedar Apple Rust	650	520	130	Fixed	Controlled

3. Next, 32 features from each color channel are concatenated horizontally to create a 1 × 96 feature vector. This is the color histogram feature vector from segmented image I_{RGB}. The complete process of color histogram feature extraction is depicted in Fig. 6.

From discrete wavelet transform, we got a feature vector of size 1 × 32 and from the color histogram, the feature vector size is 1 × 96 for each segmented image I_{RGB}. Then we have combined these two types of feature vectors horizontally and got our final feature vector of size 1 × 128.

We create this feature vector for every image of each class to feed into the classifier for training and testing. The complete block diagram of our proposed system is depicted in Fig. 7.

Table 5
Classification performance of different classifiers for different number of DWT features.

Feature vector	Classifier	Accuracy (%)	Precision (%)	Recall (%)	F1 score (%)
DWT 8	SVM	83.22	83.41	83.27	83.20
	KNN	85.48	85.86	85.54	85.58
	XGB	86.90	87.16	86.94	86.90
	RF	**88.06**	**88.28**	**88.12**	**88.03**
DWT 16	SVM	85.48	85.49	85.60	85.44
	KNN	87.64	88.00	87.71	87.65
	XGB	90.69	90.72	90.77	90.66
	RF	**90.74**	**90.84**	**90.80**	**90.74**
DWT 32	SVM	85.90	85.90	85.99	85.85
	KNN	87.69	87.60	87.78	87.56
	XGB	92.69	92.68	92.72	92.65
	RF	**92.79**	**92.81**	**92.80**	**92.75**
DWT 64	SVM	85.90	85.90	85.99	85.85
	KNN	87.69	87.60	87.78	87.56
	XGB	92.69	92.68	92.72	92.65
	RF	**92.79**	**92.81**	**92.80**	**92.75**

Bold values indicate highest Accuracy, Precision, Recall and F1 score for each case.

4. Result and discussion

Our proposed method is evaluated by training and testing it on the Plant Village dataset. All the experimental results are obtained by using 80:20 ratios for training and testing and 10-fold cross-validation (10CV). Table 4 provides a brief detail about the total number of images from each class, the number of samples used in testing and training, type of image background, and capturing condition. The proposed method is validated by 4 classifiers namely K-nearest Neighbor (KNN) with $k = 5$, Multicategory Support Vector Machine (M-SVM), Extreme Gradient Boosting Machine (XGB), and Random Forest (RF) with 100 trees. Random Forest is our base classifier and its performance is compared with the other classifiers. The performance of our model is evaluated based on 4 metrics named accuracy, precision, recall, and F1 score. Performance is measured in three phases such as: using only DWT features, using only color histogram features, and using the combination of DWT and color histogram features. All the experiments are run using Matlab 2018a and Anaconda 4.10.3 on a 64-bit Windows 10 machine with 8 GB RAM and Intel Core i3-4100 M CPU.

At first, the model is tested using only DWT features. We have extracted 8, 16, 32, and 64 DWT features to test the performance of our model. The results of this test are shown in Table 5.

For 8 coefficients, RF gives the best performance by achieving accuracy of 88.06% and SVM has the lowest performance. When the number of coefficients is increased from 8 to 16, the overall performance of all the classifiers is increased and again RF has shown the best performance with 90.74% accuracy. Then we increase the number of coefficients to 32 and see notable performance improvement for XGB and RF. Although these two classifiers show close result but RF has slightly outperformed XGB. It provides the accuracy of 92.79%. SVM and KNN does not show any notable improvement while going from DWT 16 to DWT 32. Then when we test the model with DWT 64, all the classifiers show the same performance as DWT 32. That means increasing the number of DWT coefficients from 32 to 64 does not affect the

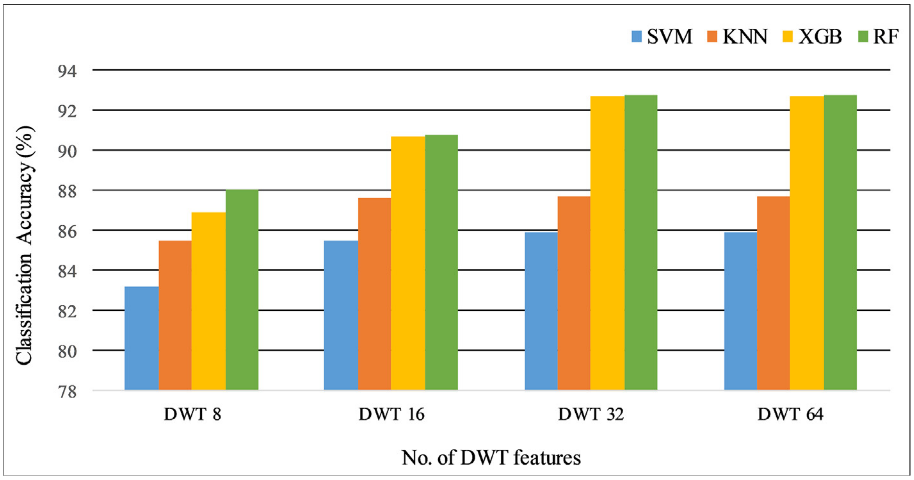

Fig. 8. Accuracy of different classifiers for different no. of DWT features.

Table 6
Classification performance of different classifiers for only color histogram feature.

Classifier	Accuracy (%)	Precision (%)	Recall (%)	F1 score (%)
SVM	84.43	84.70	84.52	84.4
KNN	91.79	91.84	91.83	91.78
XGB	94.84	94.85	94.81	95.82
RF	**95.58**	**95.56**	**95.57**	**95.56**

Bold values indicate highest Accuracy, Precision, Recall and F1 score.

Table 7
Classification performance of different classifiers for combined DWT-32 and color histogram feature.

Classifier	Accuracy (%)	Precision (%)	Recall (%)	F1 score (%)
SVM	86.17	86.46	86.25	86.14
KNN	94.42	94.39	94.42	94.39
XGB	96.84	96.83	96.93	96.85
RF	**98.63**	**98.65**	**98.64**	**98.64**

Bold values indicate highest Accuracy, Precision, Recall and F1 score.

Table 8
Confusion matrix of our proposed method.

Actual Class	No. of Test Images	Predicted Class		
		Apple Scab	Black Rot	Apple Cedar Rust
Apple Scab	630	**617** **(97.94 %)**	0 (0 %)	13 (2.06 %)
Black Rot	621	0 (0 %)	**619** **(99.68 %)**	2 (3.22 %)
Apple Cedar Rust	650	6 (0.92 %)	5 (0.77 %)	**639** **(98.31 %)**

Bold values indicate the number of correctly identified samples of each class.

performance of the classifiers. That's why 32 is our optimum number of DWT coefficients since it has less chance to face the curse of dimensionality than 64 coefficients. A graphical representation of the performance of the classifiers for different numbers of DWT coefficients in terms of accuracy is shown in Fig. 8.

After that, we have tested our model with only color histogram features. 96 color histogram features are used for the test. The test results are shown in Table 6. We can see from Table 6 that, the performance of all classifiers except SVM has been improved for color histogram features compared to DWT Features. The performance of SVM has been decreased in this case. RF has shown the best performance for color histogram features which brings the accuracy of 95.58%.

Finally, we have concatenated 32 DWT features and 96 color histogram features horizontally which gives us a combined feature vector of length 128. Test results under combined feature vector are shown in Table 7. It shows that each classifier has got a notable performance improvement for the combined feature vector than any of the single type feature vectors. Our base classifier RF has got the best accuracy in classifying apple leaf diseases based on the proposed combined feature vector. It provides the results as: 98.63% accuracy, 98.65% precision, 98.64% recall and 98.64% F1 score. These figures indicate that the fusion of DWT-32 and color histogram features has been very fruitful in detecting apple leaf diseases.

From the results, it is visualized that, classification accuracy is better for color histogram feature than DWT-32 while using them separately. But their combination significantly improves the accuracy. RF has shown better performance for all the scenarios and outperformed the other three classifiers. Fig. 9 shows a comparison of accuracy among DWT-32, color histogram, and combined feature for different classifiers.

The confusion matrix of proposed method is presented in Table 8, which shows the number of actual target values and predicted values. The class-wise accuracy is found as: Apple Scab 97.94%, Black Rot 99.68% and Apple Cedar Rust 98.31%.

To determine the trade-off between sensitivity and specificity we generated a Receiver Operator Characteristic (ROC) curve for each class following the one-vs-rest method. The ROC curves are shown in Fig. 10. For each class, the ROC curves are very close to the top left corner which indicates the very good performance of the model. Area Under the Curve (AUC) is 0.990 for apple scab, 0.999 for black rot, and 0.997 for apple cedar rust. These high values of AUC indicate the high discriminative ability of the proposed model. Our model has more than a 99% probability that it will rank a random positive sample higher than a random negative sample.

To determine the usability, we compared it with existing state-of-the-art methods for apple disease detection. Comparison results are shown in Table 9. Analyzing the results, it can be concluded that the proposed model has outperformed existing methods of detecting Apple leaf diseases.

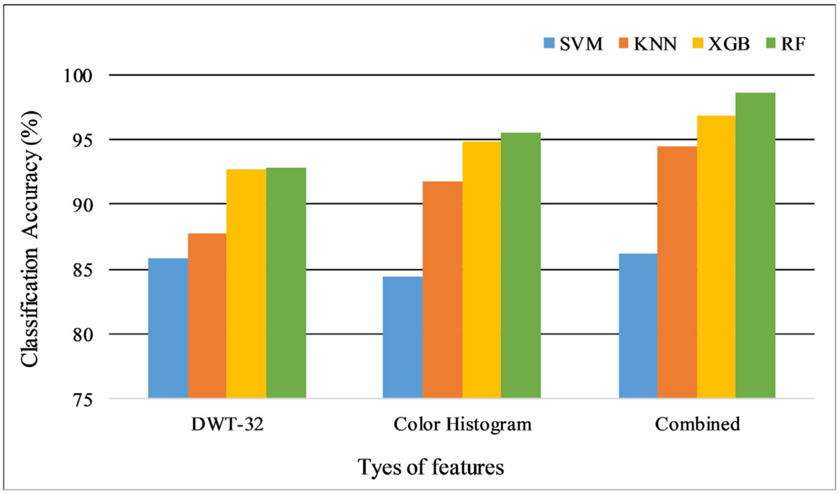

Fig. 9. Accuracy of different classifiers for different types of features.

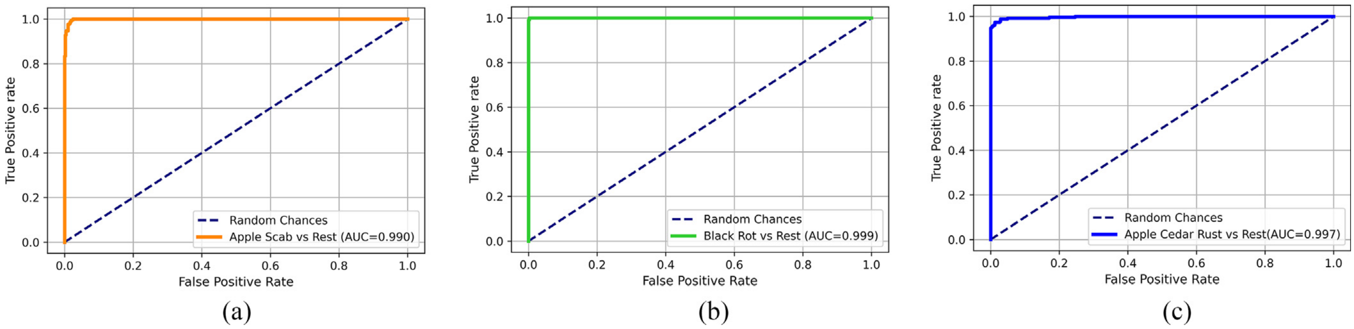

Fig. 10. ROC curve for each class using the one-vs-rest method. (a) Apple Scab; (b) Black Rot; (c) Apple Cedar Rust.

Table 9
Comparison with state-of-the-art methods for Apple leaf disease detection.

Method	Dataset	Disease Class	Class-wise Accuracy	Overall Accuracy
(Khan et al., 2019)	Plant Village	Apple Scab	97.30 %	97.20 %
		Black Rot	98.86 %	
		Cedar Apple Rust	94.62 %	
		Healthy	98.00 %	
(Bracino et al., 2020)	Plant Village	Apple Scab	64.10 %	83.3 %
		Black Rot	85.00 %	
		Cedar Apple Rust	81.70 %	
(Chuanlei et al., 2017)	Custom	Powdery Mildew	94.78 %	94.22 %
		Mosaic	93.61 %	
		Rust	94.28 %	
(Wang et al., 2017)	Plant Village	Apple Scab	–	79.30 %
		Black Rot		
		Cedar Apple Rust		
		Healthy		
(Wiesner-Hanks et al., 2018)	Plant Village	Apple Rot	98.10 %	97.50 %
		Apple Scab	96.90 %	
Proposed	**Plant Village**	**Apple Scab**	**97.94 %**	**98.63 %**
		Black Rot	**99.68 %**	
		Cedar Apple Rust	**98.31 %**	

Bold values indicate the number of correctly identified samples of each class.

5. Discussion

We have tried to propose an effective and efficient system for detecting Apple leaf disease. However, there are some limitations of our study. The images used in this study have fixed background. In real life situations, the background may not always be fixed. The reason behind this limitation is using Plant Village dataset. All the images in that dataset have fixed background.

We have considered three Apple diseases here. There are some other major Apple diseases which has not been covered in current study. We plan to include those in future study.

6. Conclusion

In this paper, we proposed a machine learning and computer vision-based automated system to detect apple disease. Proposed $L^*a^*b^*$ color space based segmentation method and derived color markers have been able to successfully segment disease infected regions. Derived color markers are based on a*b* color space which does not include L-channel. The L-channel excluded color markers limit the necessity of image preprocessing to handle uneven lighting and exposure.

We have extracted DWT features using a novel way where we have applied multilevel discrete wavelet transformation on each row of the segmented image and considered only the approximation coefficients. The DWT feature vector derived by our proposed method is less susceptible to noise and has performed well despite the presence of peeper noise in the dataset We have also extracted color features from $L^*a^*b^*$ color histogram with empirically derived bin size. Complete feature vector is the fusion of proposed DWT and color histogram features, which is a novel fusion.

The proposed segmentation, feature extraction and fusion method in this paper is found successful in detecting the three apple diseases and outperformed many other states of the art methods. Our method has achieved a predictive accuracy of 98.63%, where the nearest result is found 97.50% in existing work.

In future, we plan to collect natural-environment data with cluttered and disrupted background. Then we will modify our segmentation method to make it able to segment natural-environment data. We also plan to include more datasets to make our model more diversified. This paper considered three disease classes but there are still a lot of disease classes uncovered. We have a plan to expand our work and include more disease classes so that it can be a global solution for apple leaf disease detection.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Abayomi-Alli, O.O., Damaševičius, R., Misra, S., Maskeliūnas, R., 2021. Cassava disease recognition from low-quality images using enhanced data augmentation model and deep learning. Expert Syst. 38, e12746.

Abisha, S., Jayasree, T., 2019. Application of Image Processing Techniques and Artificial Neural Network for Detection of Diseases on Brinjal Leaf. IETE J. Res., 1–13

Adeel, A., Khan, M.A., Sharif, M., Azam, F., Shah, J.H., Umer, T., Wan, S., 2019. Diagnosis and recognition of grape leaf diseases: An automated system based on a novel saliency approach and canonical correlation analysis based multiple features fusion. Sustain. Comput. Informatics Syst. 24,. https://doi.org/10.1016/J.SUSCOM.2019.08.002 100349.

Al Bashish, D., Braik, M., Bani-Ahmad, S., 2011. Detection and classification of leaf diseases using K-means-based segmentation and. Inf. Technol. J. 10, 267–275.

Ali, H., Lali, M.I., Nawaz, M.Z., Sharif, M., Saleem, B.A., 2017. Symptom based automated detection of citrus diseases using color histogram and textural descriptors. Comput. Electron. Agric. 138, 92–104.

Almadhor, A., Rauf, H.T., Lali, M.I.U., Damaševičius, R., Alouffi, B., Alharbi, A., 2021. Ai-driven framework for recognition of guava plant diseases through machine learning from dslr camera sensor based high resolution imagery. Sensors 21, 3830.

Bansal, P., Kumar, R., Kumar, S., 2021. Disease Detection in Apple Leaves Using Deep Convolutional Neural Network. Agriculture 11, 617.

Basavaiah, J., Anthony, A.A., 2020. Tomato Leaf Disease Classification using Multiple Feature Extraction Techniques. Wirel. Pers. Commun. 115, 633–651.

Bi, C., Wang, J., Duan, Y., Fu, B., Kang, J.-R., Shi, Y., 2020. MobileNet based apple leaf diseases identification. Mob. Networks Appl., 1–9

Bracino, A.A., Concepcion, R.S., Bedruz, R.A.R., Dadios, E.P., Vicerra, R.R.P., 2020. Development of a Hybrid Machine Learning Model for Apple (Malus domestica) Health Detection and Disease Classification, in: 2020 IEEE 12th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM). IEEE, pp. 1–6.

G. Capizzi LO SCIUTO, G., Napoli, C., Tramontana, E., WOŹNIAK, M., A Novel Neural Networks-Based Texture Image Processing Algorithm for Orange Defects Classification 2016 Int. J. Comput. Sci Appl 13.

Chuanlei, Z., Shanwen, Z., Jucheng, Y., Yancui, S., Jia, C., 2017. Apple leaf disease identification using genetic algorithm and correlation based feature selection method. Int. J. Agric. Biol. Eng. 10, 74–83.

Cruz, A.C., Luvisi, A., De Bellis, L., Ampatzidis, Y., 2017. X-FIDO: An effective application for detecting olive quick decline syndrome with deep learning and data fusion. Front. Plant Sci. 8, 1741.

Dubey, S.R., Jalal, A.S., 2016. Apple disease classification using color, texture and shape features from images. Signal, Image Video Process. 10, 819–826.

Habib, M.T., Majumder, A., Jakaria, A.Z.M., Akter, M., Uddin, M.S., Ahmed, F., 2020. Machine vision based papaya disease recognition. J. King Saud Univ. Inf. Sci. 32, 300–309.

Hamdani, H., Septiarini, A., Sunyoto, A., Suyanto, S., Utaminingrum, F., 2021. Detection of oil palm leaf disease based on color histogram and supervised classifier. Optik (Stuttg). 245,. https://doi.org/10.1016/J.IJLEO.2021.167753 167753.

D. Hughes M. Salathé An open access repository of images on plant health to enable the development of mobile disease diagnostics 2015 arXiv Prepr arXiv1511.08060.

Jiang, F., Lu, Y., Chen, Y., Cai, D., Li, G., 2020. Image recognition of four rice leaf diseases based on deep learning and support vector machine. Comput. Electron. Agric. 179,. https://doi.org/10.1016/J.COMPAG.2020.105824 105824.

Khan, M.A., Lali, M.I.U., Sharif, M., Javed, K., Aurangzeb, K., Haider, S.I., Altamrah, A. S., Akram, T., 2019. An optimized method for segmentation and classification of apple diseases based on strong correlation and genetic algorithm based feature selection. IEEE Access 7, 46261–46277.

Khan, S., Narvekar, M., 2020. Novel fusion of color balancing and superpixel based approach for detection of tomato plant diseases in natural complex environment. J. King Saud Univ. Inf. Sci.

Kumari, C.U., Prasad, S.J., Mounika, G., 2019. Leaf disease detection: feature extraction with K-means clustering and classification with ANN. In: In: 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC). IEEE, pp. 1095–1098.

Li, D., Wang, R., Xie, C., Liu, L., Zhang, J., Li, R., Wang, F., Zhou, M., Liu, W., 2020. A Recognition Method for Rice Plant Diseases and Pests Video Detection Based on Deep Convolutional Neural Network. Sensors 2020, Vol. 20, Page 578 20, 578. https://doi.org/10.3390/S20030578.

Luo, Y., Sun, J., Shen, J., Wu, X., Wang, L., Zhu, W., 2021. Apple Leaf Disease Recognition and Sub-Class Categorization Based on Improved Multi-Scale Feature Fusion Network. IEEE Access 9, 95517–95527.

Padol, P.B., Yadav, A.A., 2016. SVM classifier based grape leaf disease detection. In: in: 2016 Conference on Advances in Signal Processing (CASP). IEEE, pp. 175–179.

Pravin Kumar, S.K., Sumithra, M.G., Saranya, N., 2021. Particle Swarm Optimization (PSO) with fuzzy c means (PSO-FCM)–based segmentation and machine learning classifier for leaf diseases prediction. Concurr. Comput. Pract. Exp. 33, e5312.

L. Ramírez Alberto E.C. Ardila C., Augusto Prieto Ortiz, F., A computer vision system for early detection of anthracnose in sugar mango (Mangifera indica) based on UV-A illumination 2022 Process. Agric Inf 10.1016/J.INPA.2022.02.001.

Z. Rehman ur, Khan, M.A., Ahmed, F., Damaševičius, R., Naqvi, S.R., Nisar, W., Javed, K., Recognizing apple leaf diseases using a novel parallel real-time processing framework based on MASK RCNN and transfer learning: An application for smart agriculture IET Image Process. 15 2021 2157 2168.

Sharif, M., Khan, M.A., Iqbal, Z., Azam, M.F., Lali, M.I.U., Javed, M.Y., 2018. Detection and classification of citrus diseases in agriculture based on optimized weighted segmentation and feature selection. Comput. Electron. Agric. 150, 220–234.

Singh, U.P., Chouhan, S.S., Jain, S., Jain, S., 2019. Multilayer convolution neural network for the classification of mango leaves infected by anthracnose disease. IEEE Access 7, 43721–43729.

Sinha, A., Shekhawat, R.S., 2020. Olive spot disease detection and classification using analysis of leaf image textures. Procedia Comput. Sci. 167, 2328–2336.

Vishnu, S., Ranjith Ram, A., 2015. Plant disease detection using leaf pattern: A review. Int. J. Innov. Sci. Eng. Technol. 2, 774–780.

Wang, S., He, D., Li, W., Wang, Y., 2009. Plant leaf disease recognition based on kernel K-means clustering algorithm. Nongye Jixie Xuebao= Trans. Chinese Soc. Agric. Mach. 40, 152–155.

Wang, G., Sun, Y., Wang, J., 2017. 2017. Automatic image-based plant disease severity estimation using deep learning. Comput. Intell, Neurosci.

Wiesner-Hanks, T., Stewart, E.L., Kaczmar, N., DeChant, C., Wu, H., Nelson, R.J., Lipson, H., Gore, M.A., 2018. Image set for deep learning: field images of maize annotated with disease symptoms. BMC Res. Notes 11, 1–3.

Yadav, R., Kumar Rana, Y., Nagpal, S., 2019. Plant leaf disease detection and classification using particle swarm optimization. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics) 11407 LNCS, 294–306. https://doi.org/10.1007/978-3-030-19945-6_21/COVER/.

Yu, H.-J., Son, C.-H., 2020. Leaf spot attention network for apple leaf disease identification, in. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 52–53.

Zhang, S., Wang, H., Huang, W., You, Z., 2018. Plant diseased leaf segmentation and recognition by fusion of superpixel. K-means and PHOG. Optik (Stuttg) 157, 866–872.

Zhong, Y., Zhao, M., 2020. Research on deep learning in apple leaf disease recognition. Comput. Electron. Agric. 168, 105146.