IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 27, 2021, accepted December 16, 2021, date of publication December 27, 2021,
date of current version January 5, 2022.

Digital Object Identifier 10.1109/ACCESS.2021.3138920

Cardamom Plant Disease Detection Approach Using EfficientNetV2

SUNIL C. K., (Student Member, IEEE), JAIDHAR C. D., AND NAGAMMA PATIL
Department of Information Technology, National Institute of Technology Karnataka, Surathkal, Mangaluru 575025, India

Corresponding author: Sunil C. K. (sunilck.187it005@nitk.edu.in)

This work was supported by the National Institute of Technology Karnataka, Surathkal, India.

ABSTRACT Cardamom is a queen of spices. It is indigenously grown in the evergreen forests of Karnataka, Kerala, Tamil Nadu, and the northeastern states of India. India is the third largest producer of cardamom. Plant diseases cause a catastrophic influence on food production safety; they reduce the eminence and quantum of agricultural products. Plant diseases may cause significantly high loss or no harvest in dreadful cases. Various diseases and pests affect the growth of cardamom plants at different stages and crop yields. This study concentrated on two diseases of cardamom plants, Colletotrichum Blight and Phyllosticta Leaf Spot of cardamom and three diseases of grape, Black Rot, ESCA, and Isariopsis Leaf Spot. Various methods have been proposed for plant disease detection, and deep learning has become the preferred method because of its spectacular accomplishment. In this study, U^2-Net was used to remove the unwanted background of an input image by selecting multiscale features. This work proposes a cardamom plant disease detection approach using the EfficientNetV2 model. A comprehensive set of experiments was carried out to ascertain the performance of the proposed approach and compare it with other models such as EfficientNet and Convolutional Neural Network (CNN). The experimental results showed that the proposed approach achieved a detection accuracy of 98.26%.

INDEX TERMS Cardamom plant, deep learning, machine learning, neural networks, plant pathology.

I. INTRODUCTION

Cardamom is widely used as a flavoring agent and is widely used in medicine, including allopathy and Ayurveda [1]. It is a money-mint crop; modern technology for agro production has been developed and widely accepted in all cardamom-growing territories in India. Still, the spread of various pests and diseases remains a challenge that is considered a significant production barrier experienced by the cardamom sector. Small cardamom is affected by a host of pathogenic bacteria, which seriously damages the crop and is often harmful. Diseases infected with cardamom plants, such as Colletotrichum Blight and Leaf Spot, have emerged frequently in fields where crop management is not considered [1].

The emergence of plant diseases distresses agrarian production. If vegetation disorders are not diagnosed in time, food scarcity will intensify [2]. Plant diseases, pests, and weeds threaten production and quality farming, resulting in crop loss and economic loss. That means about 15-25% of food production in India [3]. Various other factors degrade the eminence and quantum of agricultural products, such as climate change and modern cultivation techniques with large amounts of chemical fertilizers. Infected plants often show apparent signs or sores on plant leaves, trunks, flowers, or fruits. In general, each disease or insect environment produces a single visual archetype that can be used to interpret anomalies. Generally, plant leaves are a significant source of plant disease, and most prophetic significance of the disease may initiate to emerge on the plant leaves [4].

In habitual, agricultural, and plant pathology experts visit the farmland or farmers to identify plant disorders and pests based on acquaintances. This approach is not only humble, but also ambitious and ineffective. Agriculturists with less knowledge may misjudge and use pesticides or insecticides indiscriminately during the screening process. This has resulted in indispensable economic losses. To address these challenges, image processing using an automatic plant leaf disease detection approach is essential. Timely perception is

The associate editor coordinating the review of this manuscript and approving it for publication was Zhouyang Ren.

the baseline for effective interdiction and supervision of plant leaf diseases, and they play an essential role in the supervision and decision-making of agrarian products.

In a recent study, computer vision and machine learning-based techniques were developed for plant leaf disease detection. Real-time plant disease detection has some significant challenges, such as complex background and severity of the disease due to the images being captured in real-time scenarios from the farm field.

In this study, the detection of cardamom plant disease was proposed. Cardamom plant leaf images are captured in the farm field with complex backgrounds and a dataset generated, which measures the detection ability of the proposed approach. U^2-Net architecture [5] used in this work, which uses multiscale features to remove the background from the image. Cutting-edge deep learning models such as EfficientNetV2 [6] were used in this work.

The key contributions of this work are:

- The cardamom plant leaf dataset was collected from a cardamom plantation in Chinnahalli, Sakaleshpur, India, from April to June 2021, using different electronic devices and set as a benchmark dataset for the subsequent study.
- Complex background of the images removed by extracting the multiscale features with U^2-Net.
- A cardamom plant disease detection approach was proposed using EfficientNetV2. A set of experiments was conducted to ascertain the detection efficiency of the proposed approach. The grape plant leaf dataset was also used to assess the performance of the proposed approach.

This paper is organized as follows: A literature survey is discussed in Section II, Section III describes the dataset used in this work, Section IV explains the proposed method and experimental results. Conclusions and future works are discussed in Section V.

II. LITERATURE SURVEY

Modern image processing and deep learning-based techniques are widely used for the detection of plant leaf disease. Many diagnostic methods use a Convolutional Neural Network (CNN) and a pre-trained model to detect and classify healthy and unhealthy plants. Manso *et al.* used segmentation to remove background data and applied a trained neural network for classification [7]. Zhang *et al.* proposed a method to diagnose cucumber plant diseases by separating images with diseased patches by combining K-means, exploring the condition and color of infected leaf lesions, and separating unhealthy leaf images using scant resentment [8]. Yeh *et al.* proposed improved ocular-heed deep neural networks for classification by reading the feature maps highlighting the essential regions that also weaken the meaningless connected layers [9]. Various artificial intelligence and image-based plant disease detection approaches have been proposed [10]. Many technological approaches have been proposed for the diagnosis of plant diseases [11]–[13] and the general diagnosis of the disease [14]–[16].

The appropriate colorimetric scheme was selected, and the relevant parameters of the appropriate elements were selected from the two-color elements and texture [17]. Pattern recognition of tomato leaves was performed using K-means clustering. However, the detection accuracy is affected by the input selection of the feature parameters. Hang *et al.* proposed disease detection by developing a traditional backpropagation algorithm, and the learning rate of the model was adjusted dynamically [18]. A region-based single-shot multi-box detector was used with Visual Geometry Group (VGG) to detect the cotton plant disease that was proposed by [19]. Sibiya and Sumbwanyambe proposed a CNN model for maize plant disease detection on mobile captured images and achieved a detection accuracy of 92.85% [20]. Shin *et al.* proposed an approach using six different pre-trained deep learning-based models for detecting powdery mildew disease of a strawberry dataset [21]. Karthik *et al.* proposed a tomato plant disease detection method based on a CNN [22]. Some of the studies used transfer learning for plant disease detection on various plant datasets [23], [24], peach dataset [25], and pre-trained AlexNet was employed for cucumber [26], [27] and tomato [28]. Pre-trained ResNet was employed for coffee and soybean plant disease detection [29], [30], and DenseNet was employed for apple leaves [31]. Jiang *et al.* proposed a novel multi-task approach by employing transfer learning with VGGNet, in which the authors extracted the independent features from multiple datasets and trained independently for multiple related tasks on wheat and rice plant datasets [32]. Singh *et al.* employed a multi-layer CNN to detect mango leaf diseases and achieved 97.13% detection accuracy [33]. The Internet of Things was employed with fuzzy networks to detect pauropsylla disease [34].

Chouhan *et al.* proposed a Wi-Fi-enabled plant detection approach by employing Wi-Fi cameras to capture images, and for classification, a Radial Basis Neural Network (RBNN) was employed to detect mango leaf diseases. The RBNN uses radial basis functions in the intermediate layers of the network [35]. Bacterial foraging escalation was assessed using RBNN. Furthermore, features were extracted from the region-growing perspective [36].

Some studies considered the combination of segmentation and classification for plant disease detection; Lu *et al.* proposed an approach for wheat disease detection by segmenting the lesion [37]. Johannes *et al.* proposed an approach for mobile captured images by segmenting the lesions of wheat leaf images [38]. Ma *et al.* proposed an approach by segmenting the lesion of a cucumber leaf and employed a CNN for classification [27]. Arsenovic *et al.* proposed an approach using Faster R-CNN, and YOLO was employed with AlexNet as a classifier and attained an average detection accuracy of 93.67% [39]. Santos *et al.* proposed an approach for segmenting the grapefruit by employing mask R-CNN and CNN for classification and attained an 0.91 F1-Score for the grape dataset [40]. Ganesh *et al.* proposed an approach

TABLE 1. Dataset used in this study.

Sl.No	Category	# of Images	Train/Test Split	Total # of images
Cardamom plant leaf dataset generated in this study				
1	Colletotrichum Blight	280	252/28	1724
2	Cardamom Healthy	781	700/81	
3	Phyllosticta Leaf Spot	663	597/66	
PlantVillage dataset (Grape) [47]				
4	Black Rot	1180	1062/118	4062
5	ESCA	1383	1245/138	
6	Grape Healthy	423	381/42	
7	Isariopsis Leaf Spot	1076	968/108	

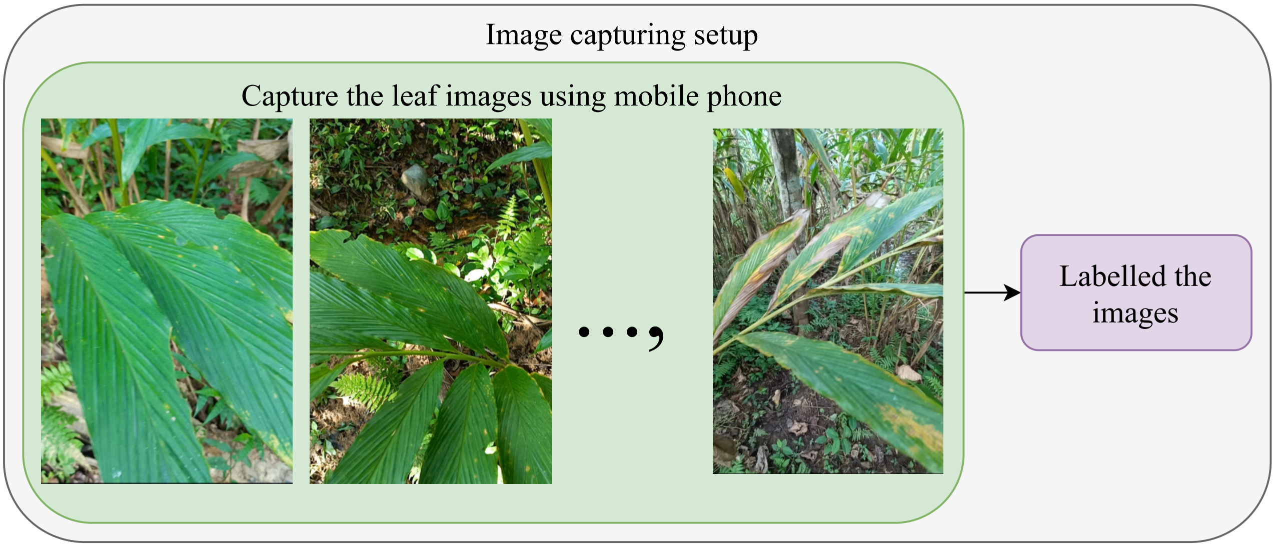


FIGURE 1. Real-time image capturing setup.

for recognizing orange fruits by employing mask R-CNN and instance segmentation tasks in an image [41]. Meyer Fernand proposed an approach for segmenting leaves by setting a certain threshold [42]. Morris Daniel employed CNN, for instance, segmenting for selecting a leaf from the image [43].

Singh *et al.* proposed a segmentation approach by employing K-means clustering, watershed segmentation, and threshold-based exemption on coconut leaf image datasets to detect leaf blight disease and attained a 96.94% detection accuracy by employing CNN [44]. Tassis *et al.* proposed an approach by employing mask R-CNN, for instance, segmentation and removal of the background using semantic segmentation by employing UNet and attained a 94.27% detection accuracy on coffee plant disease detection [45]. Chouhan *et al.* proposed a neural network model with superpixel clustering for segmentation and achieved 98.57% detection accuracy [46].

Most of the studies considered the public dataset [47]; this study collected the cardamom plant leaf images of complex backgrounds from farm fields. However, to the best of our knowledge, this is the first study on cardamom plant leaf disease detection using a deep learning-based approach.

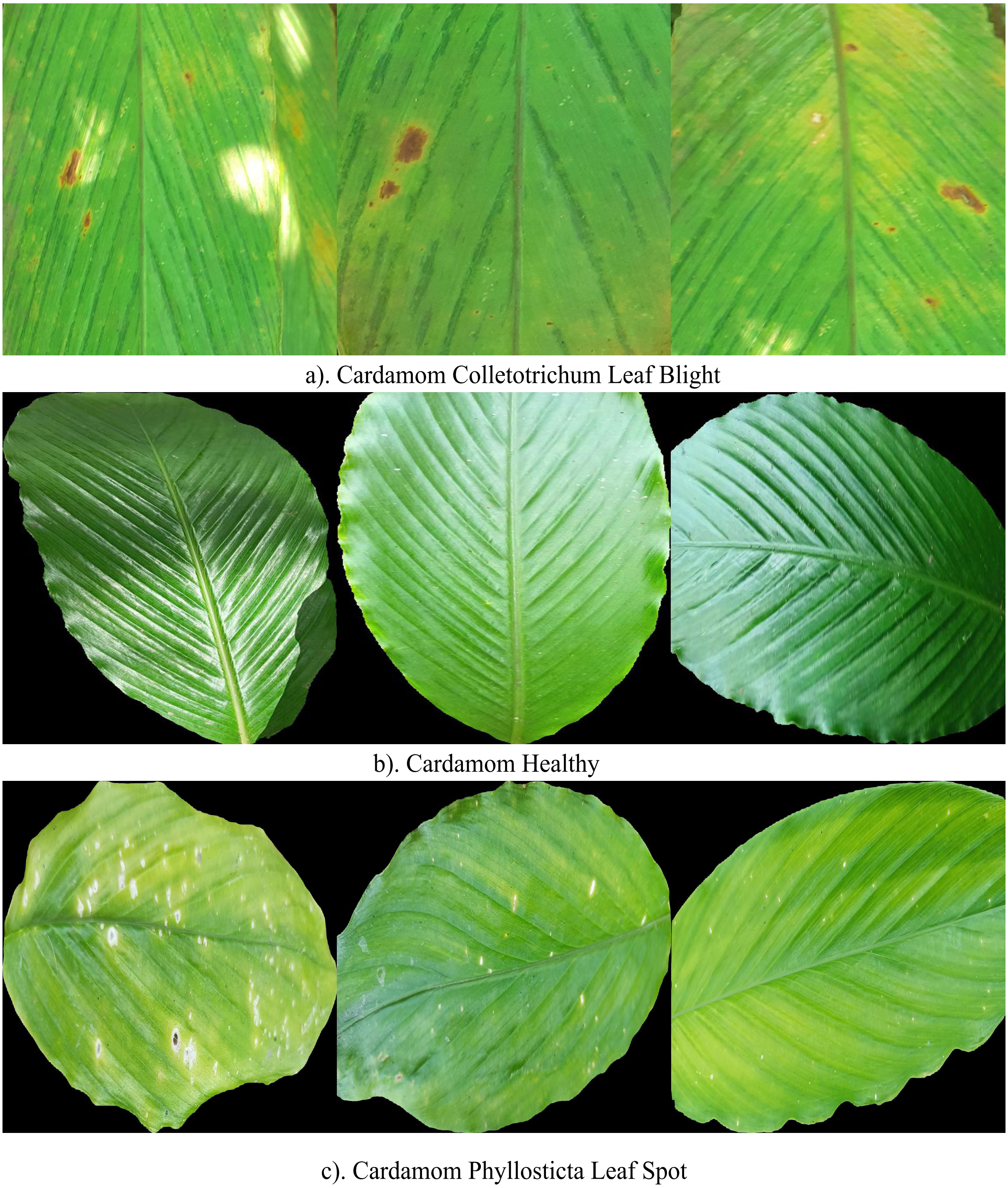
III. DATASET DESCRIPTION

A. IMAGE CAPTURING SETUP

Figure 1 shows the image capturing setup. Real-time cardamom plant leaf images were captured in the farmland using different mobile phones. To encourage real-time plant disease classification, images were captured with background, noise, different light illumination, and different angles.

B. CARDAMOM DATASET 2021

In this study, we collected 1724 cardamom plant leaf images of three classes, namely, Colletotrichum Blight and Phyllosticta Leaf Spot and healthy category. These are labeled with the help of the Indian Cardamom Research Institute officers, Regional Station Sakaleshpur, Karnataka(State), a wing of Spices Board India. All images were captured in daylight from 10 AM to 5 PM from April to June 2021. Table 1 presents the cardamom dataset of 2021. The diseases mentioned in Table 1 are common to the cardamom plant, affecting the growth and yield of the crop. Each image is further captured under a farm field scenario, without technical means, preserving all archive information and removing the background image in this work. The original cardamom plant leaf images had a complex background with different


a). Cardamom Colletotrichum Leaf Blight

b). Cardamom Healthy

c). Cardamom Phyllosticta Leaf Spot

FIGURE 2. Cardamom plant leaf images: a) cardamom Colletotrichum leaf blight, b) cardamom healthy, and c) cardamom Phyllosticta leaf spot.

dimensions and capturing conditions. The three different types of cardamom plant leaf images are shown in Figure 2.

C. PLANTVILLAGE DATASET

PlantVillage [47] is a freely available dataset that is widely used in the field of plant disease classification. It contains over 54,284 images, all of which were annotated. In these images, it is difficult to find the inclement conditions such as the complex background. In this study, we used the grape dataset available in the PlantVillage dataset was also used. The details of the dataset used in this study are described in Table 1.

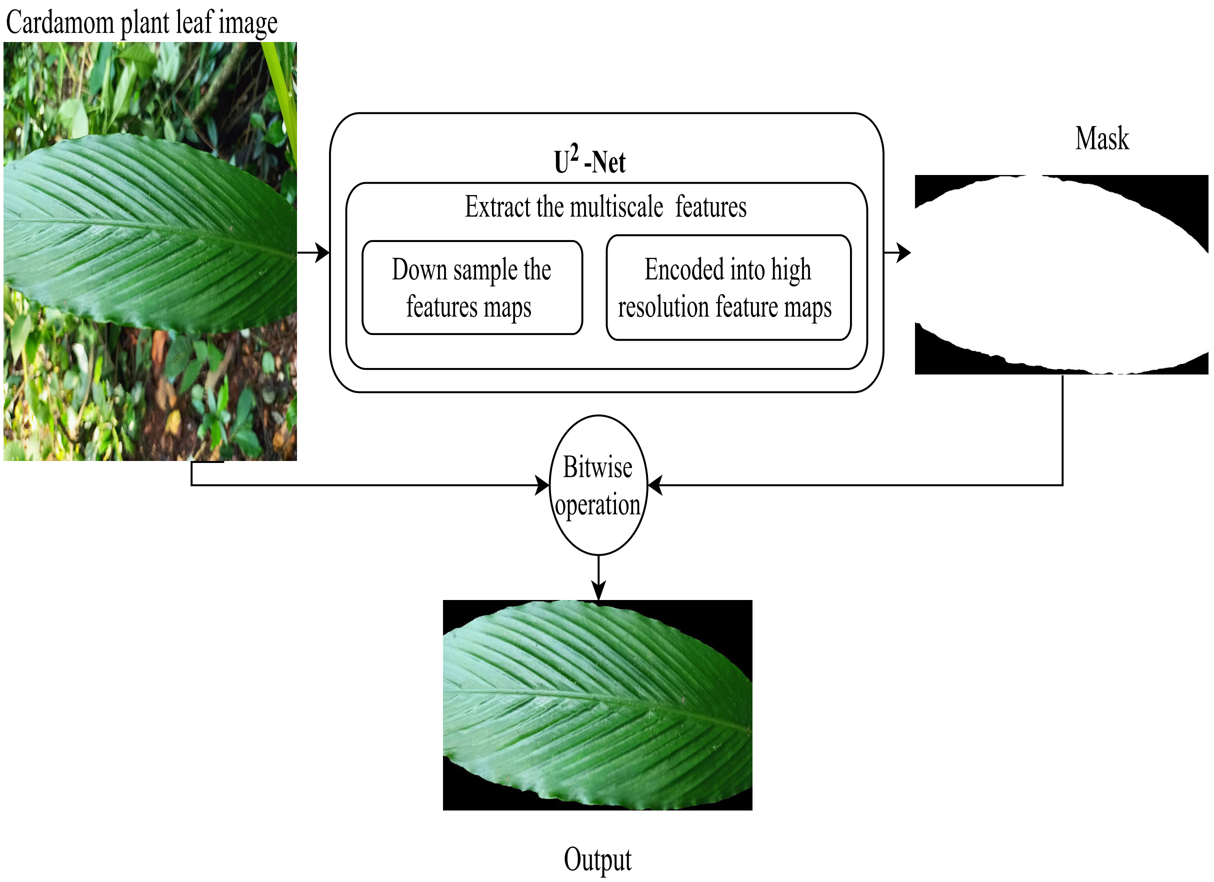

Cardamom plant leaf image
U^2-Net
Extract the multiscale features
Down sample the features maps
Encoded into high resolution feature maps
Mask
Bitwise operation
Output

FIGURE 3. Background removal by using U^2-Net.

IV. PROPOSED METHOD

In this study, we proposed a cardamom plant leaf disease detection approach by employing a background removal technique to remove the complex background of the image by using U^2-Net. EfficientNetV2 deep learning model is used for classification.

A. BACKGROUND REMOVAL

Cardamom plant leaf images are of RGB, collected with a complex background with different dimensions and resolution, and the leaf is surrounded by several other factors, generally in the environment.

In most cases, computer vision algorithms remove the background from an image, such as image thresholding in OpenCV and grab hut techniques [48]. These techniques help when the background color differs from the interesting object; in such cases, it is easy to remove the background by utilizing green and blue screens to eradicate the foundation and replace it with another scene.

Removing the background from an image is a highly challenging task without explicit pre-or post-processing. If the item has a very similar color to the background, it tends to be highly challenging to track down a perfect form because of soft edges or shadows.

The background removal approach used in this study was U^2-Net [5] as shown in Figure 3.. It takes the input image to produce a mask of the region of interest. Further, it applies a bitwise operation on the original image and the mask produced by U^2-Net. U^2-Net architecture is a twofold interlaced U-structure, as shown in Figure 4. It consists of 3 parts. The first part is a six-stage encoder; in this stage, it uses a ReSidual U-Block (RSU):

$$H_{RSU} = U(FM(x)) + FM(x) \tag{1}$$

RSU block has three components described in Figure 5; extricate local features, an input convolutional layer generates the intermediate activation map $FM(x)$, next is encoder-decoder, which is like U-Net that takes $FM(x)$ as input. It learns to extricate and encode the multiscale contingent attributes $U(FM(x))$. Minimizing the loss during direct upsampling extricates the multiscale attributes from gradually down-scaled activation maps. It encodes them into high-aspiration activation maps by incremental upsampling, concatenation, and convolution. Finally, the residual connection combines the surplus connection, which combines local attributes and multiscale attributes as shown in Equation 1.

The second part of the U^2-Net architecture is a five-stage decoder that uses the dilated version of the RSU. Finally,

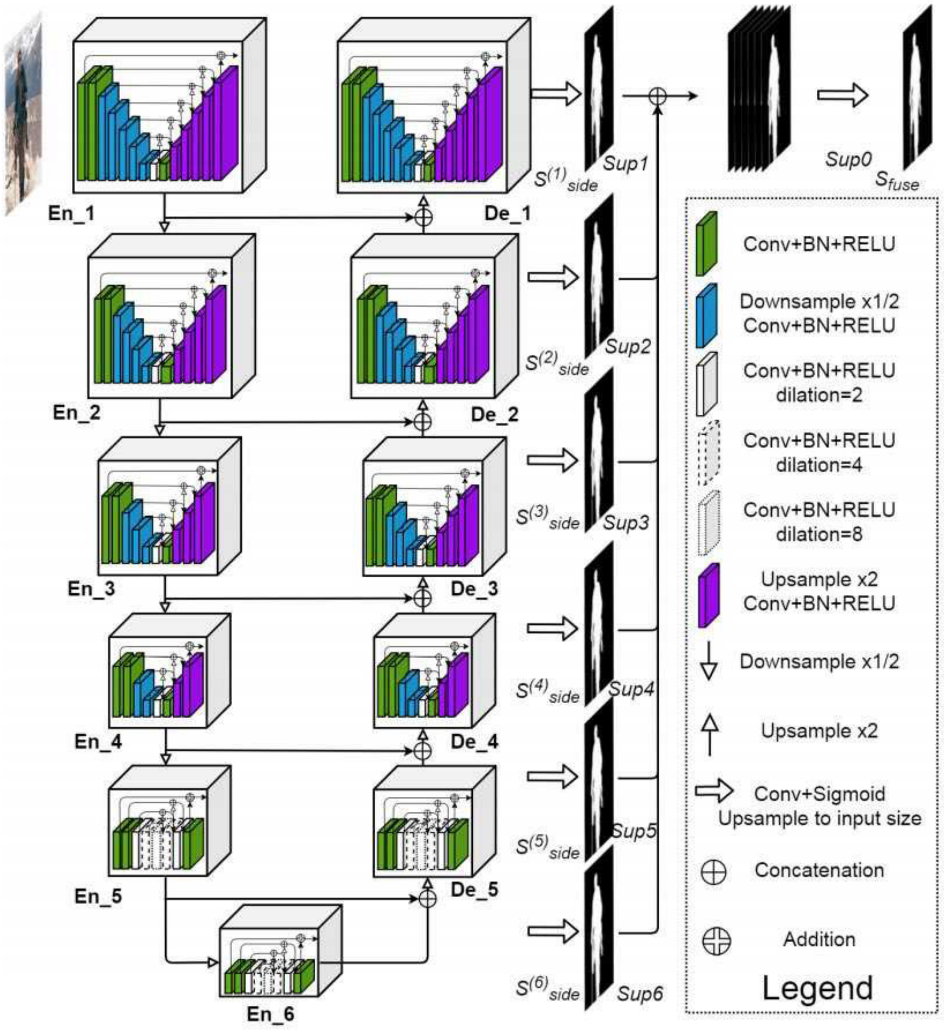


FIGURE 4. U^2-Net architecture [5].

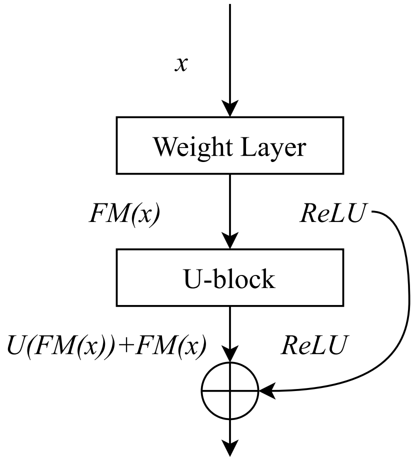


FIGURE 5. ReSidual U-block.

saliency probability maps were generated by attaching the decoder stages to the encoder stage. Saliency-map shows each pixel's unique quality. A saliency-map helps to distinguish an interesting part from the background. Deep supervision was employed with the following loss function (Equation 2) to minimize the loss and to manifest robust regularization for the learned features.

$$\text{Loss} = \sum_{n=1}^{N} \mathrm{w}_{side}^{n} l_{side}^{n} + \mathrm{w}_{fusion} l_{fusion} \qquad (2)$$

where N is number of saliency-maps generated by the encoder and decoder, l_{side}^{n} is the loss of side-saliency-maps for each stage,
l_{fusion} is the fusion for the ultimate loss of the saliency-map,
w_{side}^{n} is weight for the side-saliency-map, and
w_{fusion} is weight for the loss-term.

B. CLASSIFICATION MODELS

CNNs consist of various layers, such as convolutional, pooling, and fully connected layers [49]. The convolutional

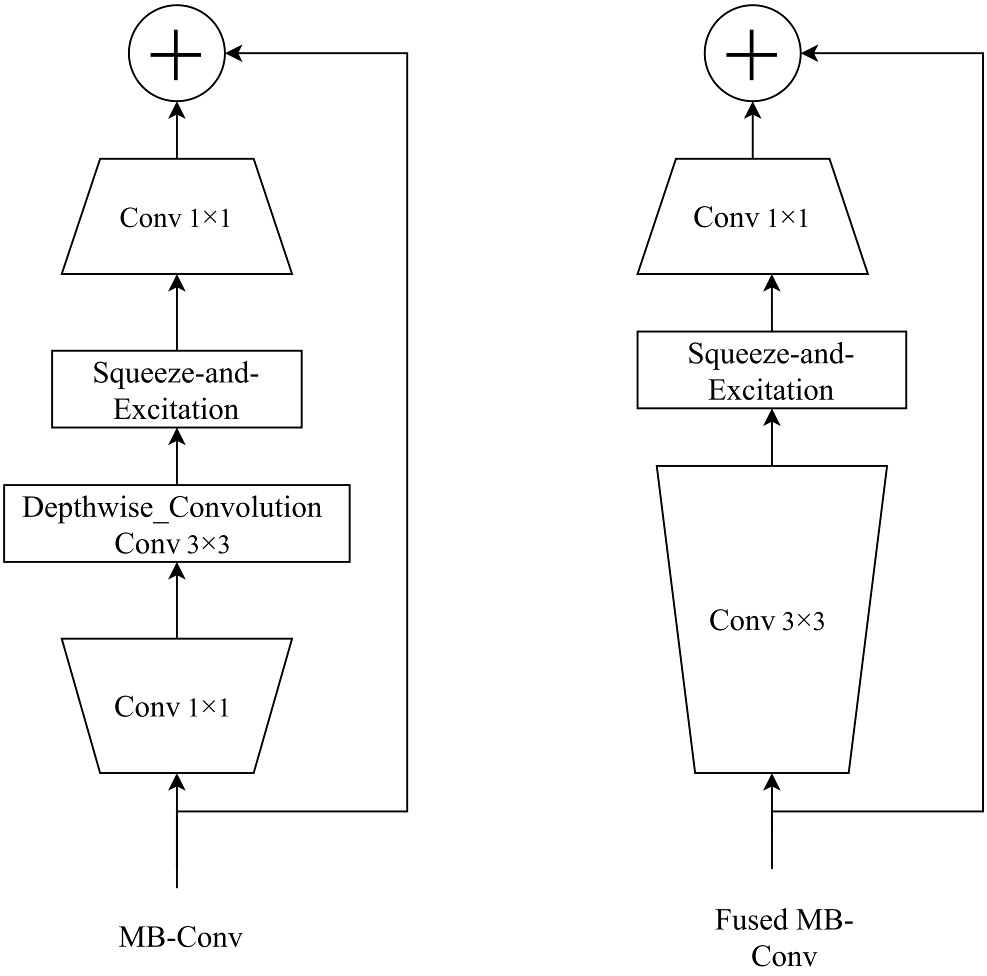


FIGURE 6. Structure of MB-Conv and fused MB-Conv [6].

layer is an essential part of the CNN because it extracts detailed information of the input images using different convolution kernels. Several convolutional layers extract the set of feature maps, known as the color and edges of the input image. The feature map function is defined in Equation 3 [49]:

$$FM_i = f(FM_{i-1}W_i + b_i) \tag{3}$$

FM denotes the feature map, W denotes the weight, b denotes the offset vector, and f(.) defines the ReLU activation function as defined in Equation 4 [50]:

$$ReLU(z) = Max(0, z) \tag{4}$$

where z is the input.

The pooling layer reduces the possibility of overfitting by minimizing the spatial dimension and convolution. Which is defined in the Equation 5 [49]:

$$y_i^l = down(y_i^{l-1}, s) \tag{5}$$

y_i^l indicates the feature vector, s defines the pooling size, and down(.) indicates the downsampling.

Finally, the one or more fully connected layers defined, which flatten the network by connecting all the previous layer neurons, final fully connected layer predicts the class label, where Softmax activation function is used in the pre-trained models which are used in this work. Softmax is defined in the Equation 6 [49]:

$$\text{Softmax}(y_i) = \frac{\exp(y_i)}{\sum_j \exp(y_j)} \tag{6}$$

where y denotes input vector
$\exp(y_i)$ denotes the exponential function for input vector.
$\exp(y_j)$ denotes the exponential function for output vector.

In CNN, hyperparameters are determined before training; on the other hand, parameters such as weights and biases are changed during training. There are two types of hyperparameters: those that deal with a network structure and those that deal with training. Kernel sizes and the number of layers of the model are the hyperparameters that deal with the network structure; kernel size plays a role in extracting the features on a large scale. The deeper the layers, the higher the classification rate. Hyperparameters that deal with

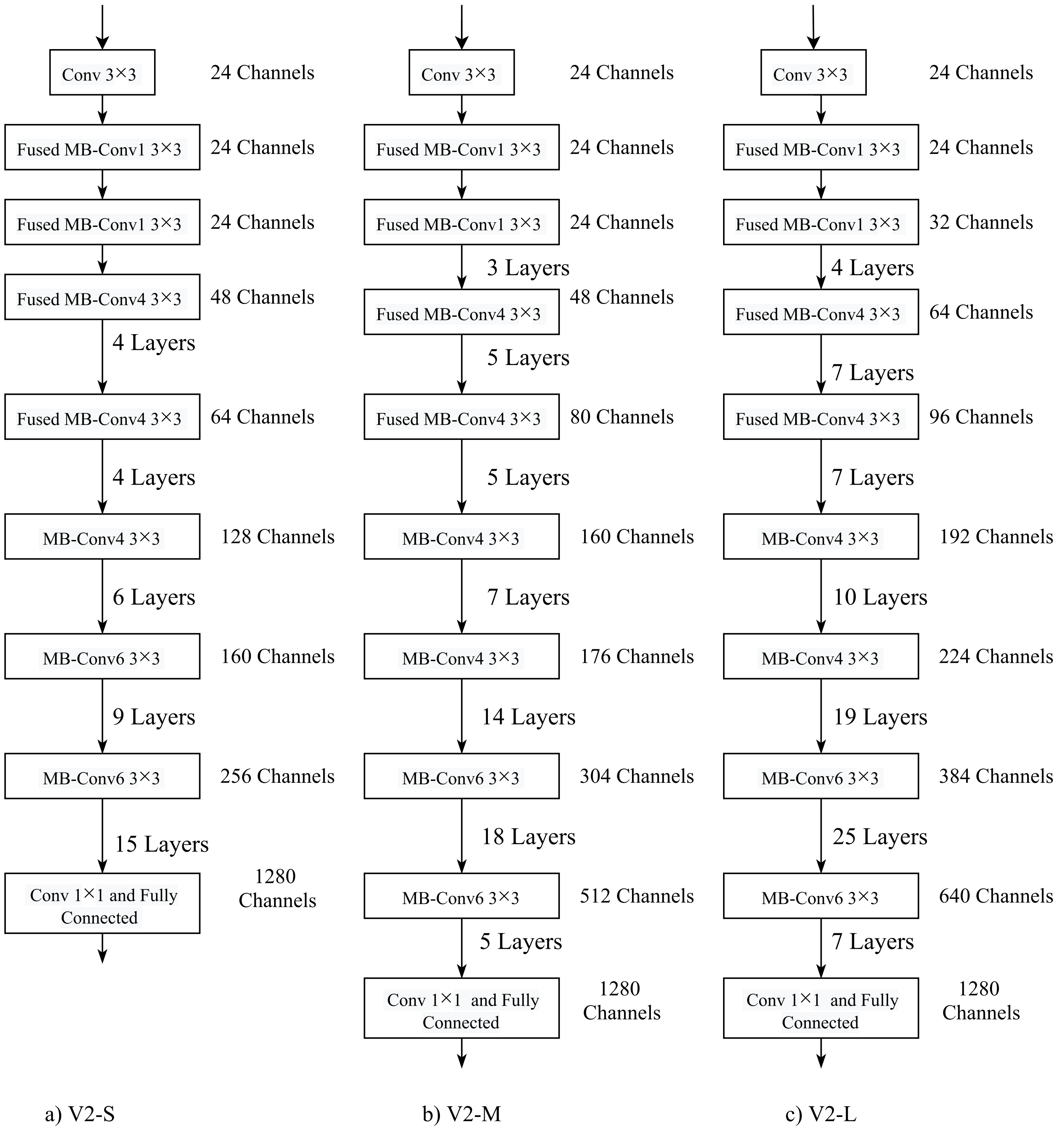


FIGURE 7. EfficientNetV2 architecture: a) EfficientNetV2-S, b) EfficientNetV2-M, c) EfficientNetV2-L [6].

training include batch size, learning rate, and dropout. The loss function is expressed by Equation 7.

EfficientNet is a family of CNNs proposed by [51]; it does the scaling on CNN such as depth, that is, how depth the network in terms of several layers, width is how wide the network, and resolution in terms of the image resolution. EfficientNet scaled up the CNN using a compound scaling method to scale the dimensions of the network. It uses Mobile inverted Bottleneck Convolution (MB-Conv) as a baseline network and scale-up this network as EfficientNet. EfficientNetV2 [6] is also a family of CNNs; it produces a higher performance and a shorter training

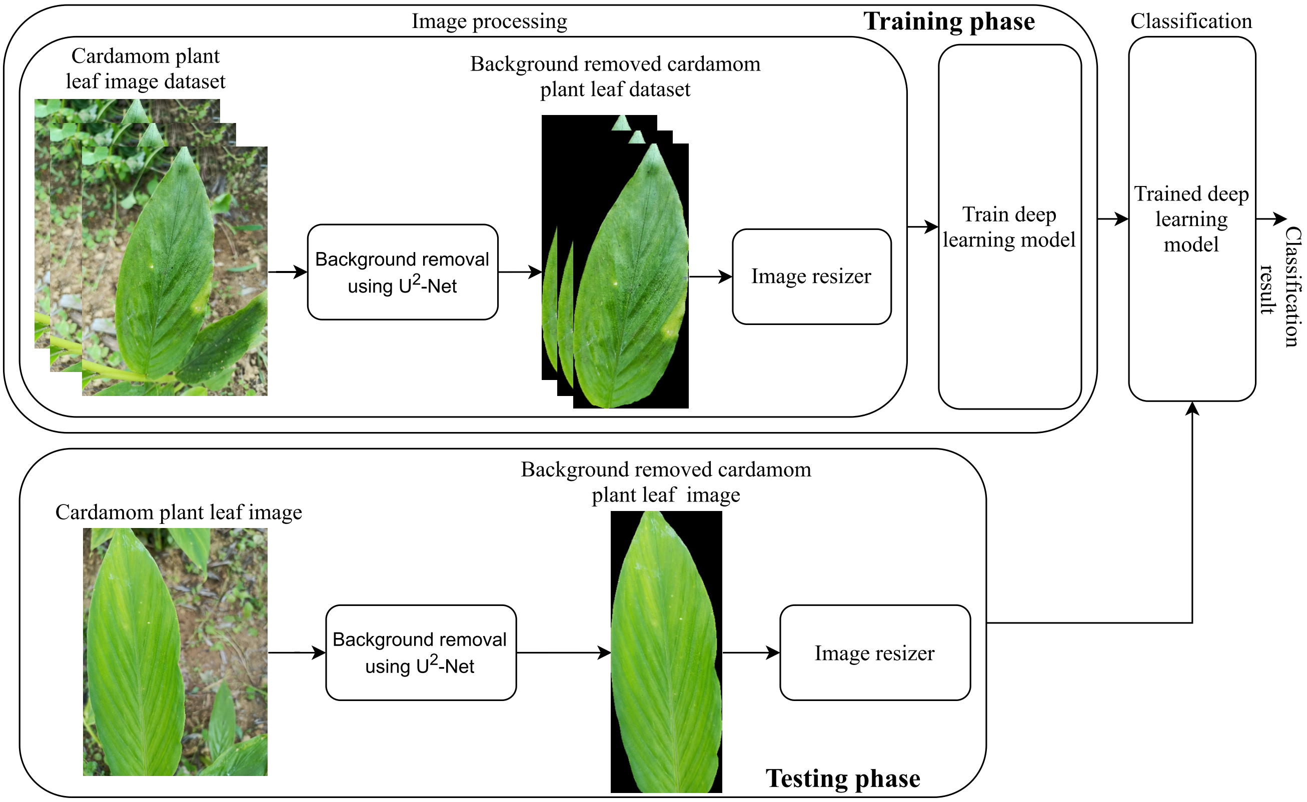


Image processing
Training phase
Classification
Cardamom plant leaf image dataset
Background removed cardamom plant leaf dataset
Background removal using U2-Net
Image resizer
Train deep learning model
Trained deep learning model
Classification result
Background removed cardamom plant leaf image
Cardamom plant leaf image
Background removal using U2-Net
Image resizer
Testing phase

FIGURE 8. **Proposed cardamom plant leaf disease detection approach.**

period. To enhance the training and efficacy, it uses Fused MB-Conv for the first three stages and MB-Conv for the subsequent stages (Figure 6), and this is faster than existing models, which are up to 6.8x smaller. The architecture of the three different EfficientNetV2 versions are shown in Figure 7.

The proposed pipeline of the cardamom plant leaf disease detection approach has four stages. The first stage depicts the dataset preparation; in this stage, we collected the real-time cardamom plant leaf images from the cardamom plantation and labeled them. The second stage is used to remove the background of the leaf image and noise, as explained earlier in this paper. The third stage is training the deep learning-based model from scratch using the generated dataset. The next and final stages are the performance evaluations of the trained model. The proposed approach is illustrated in Figure 8. This has two phases; the first is the training phase, which is used as the processing stage to remove the complex background from the input image by using U^2-Net. The background removed images are further processed to resize the images using an image resizer and fed into the next stage. The next stage is employed to train the deep learning-based models such as CNN, EfficientNet, and EfficientNetV2. The three different versions of EfficientNetV2 used in this work are EfficientNetV2-S (Small with 22 million parameters), EfficientNetV2-M (Medium with 54 million parameters), and EfficientNetV2-L (Large with 120 million parameters). Finally, the trained model produces the classification results.

In the testing phase, the cardamom plant leaf image was fed to a trained deep learning-based model after the completion of pre-processing operations, such as background removal and resizing of the image, the trained deep learning-based model produces the classification results.

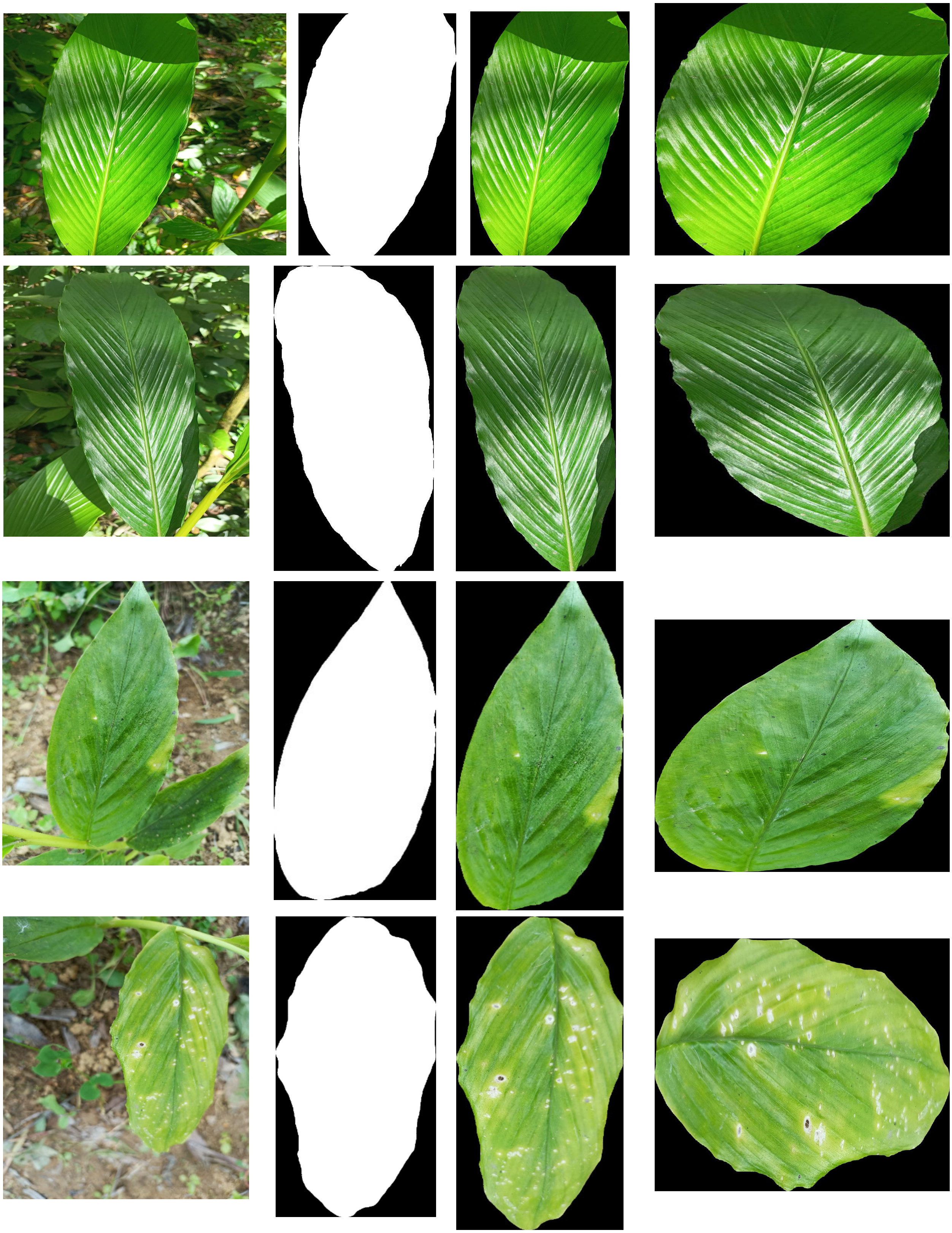
$$\mathrm{L(M)} = -\frac{1}{n}\sum_{Y_i=1}^{n}\sum_{c=1}^{C}[\mathrm{Y}_{ic} logP(x_i = c) + (1 - \mathrm{Y}_{ic}) log(1 - P(\mathrm{Y}_i = c))] \quad (7)$$

where M represents the weight matrix, n represents the number of training samples, C represents class labels, and P is the predicted probability.

C. EXPERIMENTAL RESULTS

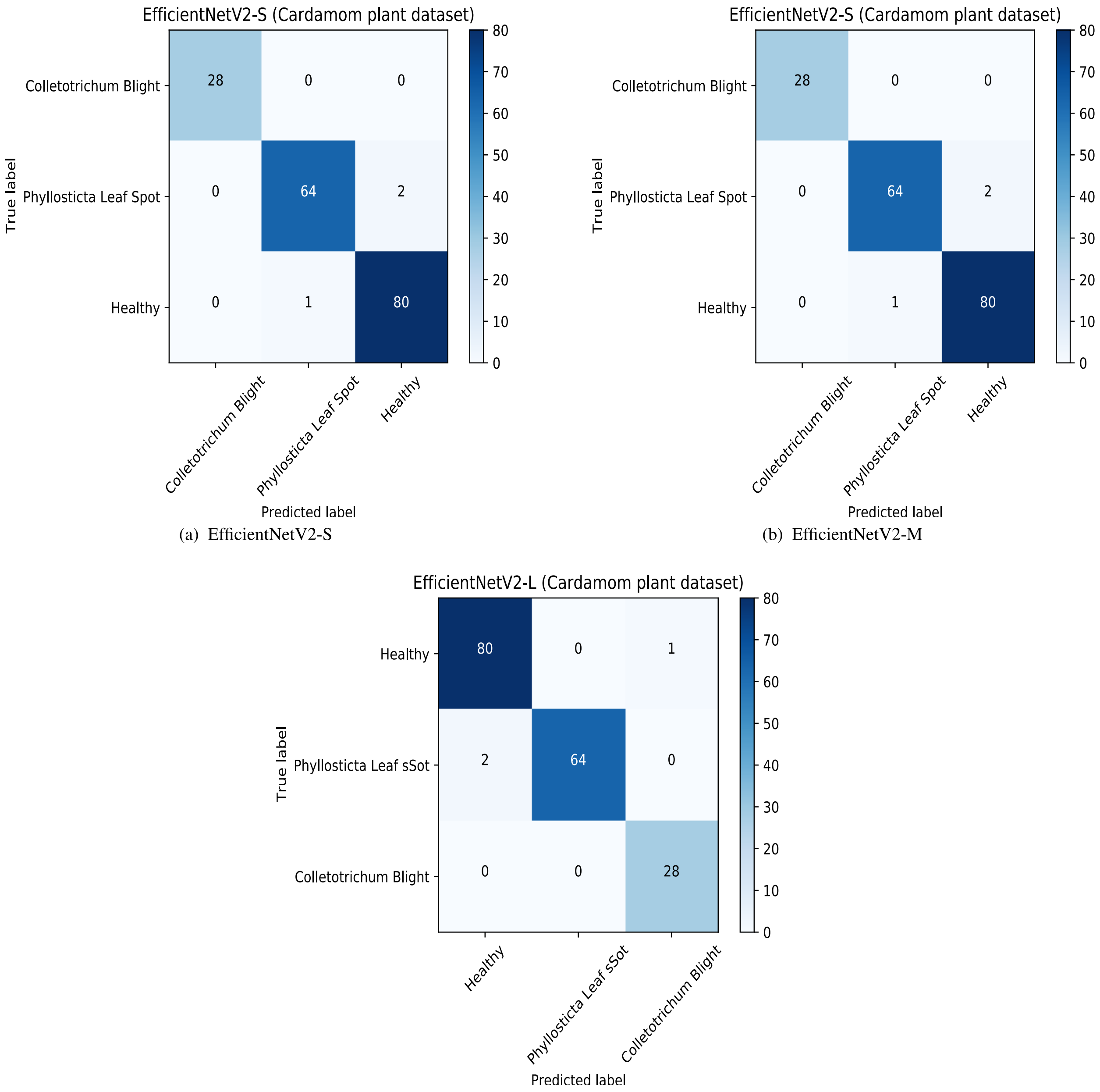
All the experiments were implemented using Python 3.6 programming language and executed using NVIDIA DGX Station server with 4X Tesla V100 and 500 TFLOPS.

As discussed in Section IV-A, all the cardamom plant leaf images are captured in complex backgrounds, and all of them are of different dimensions; to remove the complex background from the captured image, U^2-Net is used [5]. Figure 9 shows the background removal of the cardamom plant leaf images using U^2-Net. Figure 9 a) shows the original cardamom plant leaf images, Figure 9 b)



a) Cardamom leaf image b) Mask c) Output d) Resized cardamom leaf image

FIGURE 9. Background removal of cardamom plant leaf images: a) cardamom plant leaf images, b) mask, c) output, and d) resized cardamom plant leaf images.



(a) EfficientNetV2-S

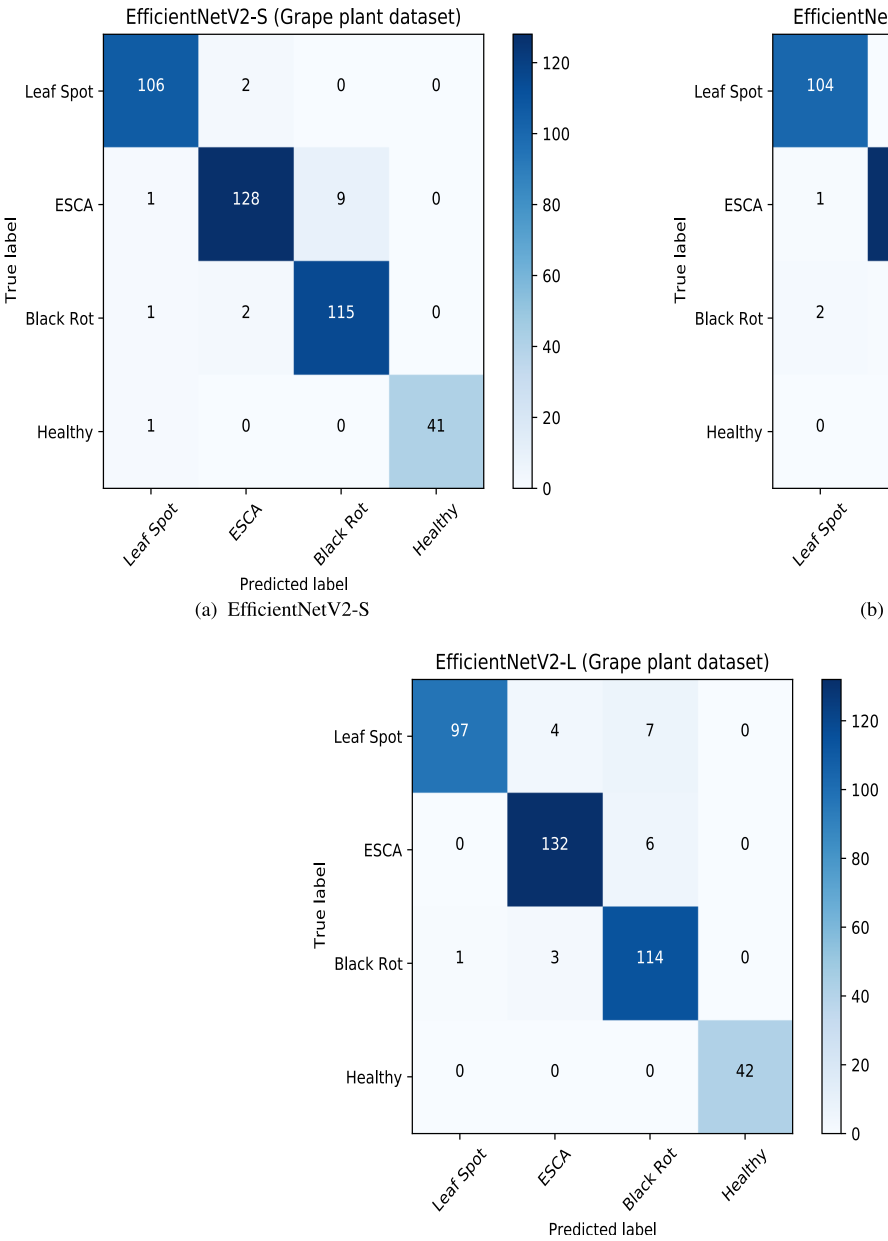
(b) EfficientNetV2-M

(c) EfficientNetV2-L

FIGURE 10. Confusion matrix for cardamom plant dataset (external testing): a) EfficientNetV2-S model, b) EfficientNetV2-M model, and c) EfficientNetV2-L model.

TABLE 2. Performance comparisons of proposed approach with other approach.

Sl.No	Plant Name	Performance Metrics	CNN	EfficientNet	EfficientNetV2-S	EfficientNetV2-M	EfficientNetV2-L
1	Cardamom	Accuracy (%)	91.30	94.10	95.59	88.44	98.26
		F1-Score	0.91	0.94	0.96	0.88	0.98
		Precision	0.91	0.94	0.96	0.88	0.98
		Recall	0.91	0.94	0.96	0.88	0.98
2	Grape	Accuracy (%)	94.24	97.81	96.44	93.72	96.45
		F1-Score	0.94	0.98	0.96	0.94	0.96
		Precision	0.94	0.98	0.96	0.95	0.96
		Recall	0.94	0.98	0.96	0.94	0.96



(a) EfficientNetV2-S

(b) EfficientNetV2-M

(c) EfficientNetV2-L

FIGURE 11. Confusion matrix for grape plant dataset (external testing): a) EfficientNetV2-S model, b) EfficientNetV2-M model, and c) EfficientNetV2-L model.

shows the mask generated by U^2-Net, Figure 9 c). The output generated by the background removal approach, and Figure 9 d) shows the resized images using the image resizer.

Accuracy, precision, recall, and F1-Score are used as performance metrics in this experiential study [37], [52], [53].

All the original input images are resized to 224 × 224 for all the three deep learning-based models used in this study. In the experiment, 90% of the dataset was used for training and 10% for testing. A set of experiments were conducted to measure the performance of the proposed approach for 100 epochs on the cardamom plant dataset. The same set of experiments was conducted using a publicly available grape dataset to assess the performance of the proposed approach. Further, an additional set of experiments was carried out using other deep learning models such as CNN and EfficientNet. The CNN attained a maximum detection accuracy of 91.30% on the cardamom plant dataset and 94.24% detection accuracy on the grape dataset. EfficientNet attained maximum detection accuracy of 94.10% and 97.81% for the cardamom and grape plant datasets, respectively. EfficientNetV2-S attained a maximum detection accuracy of

TABLE 3. Performance evaluation on external testing for cardamom plant leaf dataset.

Category	Performance Metrics	CNN	EfficientNet	EfficientNetV2-S	EfficientNetV2-M	EfficientNetV2-L
Cardamom plant dataset						
Colletotrichum Blight	Accuracy (%)	96.48	100	100	92.85	100
	F1-Score	0.98	0.98	1.00	0.95	0.98
	Precision	1.00	0.97	1.00	0.96	0.97
	Recall	0.96	1.00	1.00	0.93	1.00
Healthy	Accuracy (%)	98.76	98.76	98.76	97.53	98.76
	F1-Score	0.98	0.98	0.98	0.96	0.98
	Precision	0.98	.0.98	0.98	0.95	0.98
	Recall	0.98	0.99	0.99	0.98	0.99
Phyllosticta Leaf Spot	Accuracy (%)	98.48	96.96	96.96	96.96	96.96
	F1-Score	0.98	0.98	0.98	0.98	0.98
	Precision	0.98	1	0.98	0.98	1.00
	Recall	0.99	0.97	0.97	0.97	0.97

TABLE 4. Performance evaluation on external testing for grape plant leaf dataset.

Category	Performance Metrics	CNN	EfficientNet	EfficientNetV2-S	EfficientNetV2-M	EfficientNetV2-L
Grape plant dataset						
Black rot	Accuracy (%)	100	97.45	97.45	97.45	96.61
	F1-Score	0.97	0.96	0.95	0.95	0.93
	Precision	0.94	0.95	0.93	0.93	0.90
	Recall	1.00	0.97	0.97	0.97	0.97
ESCA	Accuracy (%)	98.55	97.82	92.75	95.65	95.65
	F1-Score	0.99	0.98	0.95	0.97	0.95
	Precision	1.00	0.99	0.95	0.99	0.95
	Recall	0.99	0.98	0.93	0.96	0.96
Healthy	Accuracy (%)	100	100	97.61	97.61	100
	F1-Score	0.99	1.00	0.99	0.98	1.00
	Precision	0.98	1	1.00	0.98	1.00
	Recall	1.00	1.00	0.98	0.98	1.00
Leaf Spot	Accuracy (%)	93.51	96.29	98.14	96.29	89.98
	F1-Score	0.97	0.97	0.98	0.97	0.94
	Precision	1.00	0.98	0.97	0.97	0.99
	Recall	0.94	0.96	0.98	0.96	0.90

TABLE 5. 5 fold cross validation results [accuracy (%)].

Sl.No	Plant Name	CNN	EfficientNet	EfficientNetV2-L
1	Cardamom	87.59	87.02	91.42
2	Grape	94.60	96.00	94.62

95.59% and 96.44% on cardamom and grape plant datasets, respectively. EfficientNetV2-M obtained a maximum detection accuracy of 88.44% and 93.72% on the cardamom and grape plant datasets, respectively. EfficientNetV2-L attained a maximum detection accuracy of 98.26% and 96.45% on the cardamom and grape plant datasets, respectively. Table 2 presents performance evaluation of the proposed approach.

Testing is broadly distinguished as either internal or external testing. Internal testing splits the dataset into training set and test set, training set is used for training the model, and the test is used for performance evaluation of the model such as accuracy, precision, recall, and F1-Score. External testing involves testing the trained model using an independently derived dataset,that is, a dataset that is not used for training and internal testing. External testing is a task that exigents the trained model with an additional dataset that is different from the original dataset used during training. This process demonstrated the generalizability of the model.

To understand the behavior of the trained models, external testing is essential; this examines the dialects learned during training, and this helps to measure the performance of the trained models. A set of external testing was conducted using trained models such as CNN, EfficientNet, and Efficinet-NetV2 models on cardamom plant and grape plant datasets. Table 3 describes the performance evaluation of the external testing for the cardamom plant dataset. In the external testing, EfficientNetV2-S outperformed the other models for the cardamom plant dataset. Figure 10 shows the confusion matrix for external testing on trained EfficientNetV2 models for the cardamom plant dataset. Table 4 describes the performance evaluation of the external testing for the grape plant dataset. In external testing, CNN outperformed the other models for the grape plant dataset. Figure 11 shows the confusion matrix for external testing on trained EfficientNetV2 models for the grape plant dataset.

A set of experiments was conducted with 5 fold cross validation on CNN, EfficientNet, and EfficientNetV2-L models; Table 5 describes the 5 fold cross validation results

TABLE 6. Comparison of proposed model with state-of-the-art methods.

Sl.No	Plant	Approach	Accuracy (%)
1	Coffee [7]	Artificial Neural Network	95.8
2	Cucumber [8]	K-means and sparse representation	85.7
3	Multiple [12]	Genetic Algorithm	96.7
4	Citrus [13]	Vision sensor and SVM	97
5	Multiple [18]	VGG16 and InceptionNet	91.7
6	Maize [20]	CNN	92.85
7	Strawberry [21]	SqueezeNet	92.61
8	Tomato [22]	CNN	98
9	Cucumber [26]	CNN	95.18
10	Multiple [27]	DCNN	93.4
11	Coffee [29]	ResNet50	95.24
12	Soybean [30]	ResNet	94.29
13	Apple [31]	DenseNet121	92.29
14	Paddy [32]	VGG16	97.22
15	Coconut [44]	MobileNet	82.10
		Proposed Model	
16	Cardamom	EfficientNetV2	98.26
17	Grape	EfficientNetV2	96.44

for 100 epochs on cardamom and grape plant leaf datasets. The EfficientNetV2-L model outperforms the other two models on the cardamom plant dataset with a 91.42% detection accuracy.

A wide set of experiments was conducted on the CNN, EfficientNet, and EfficientNetV2 models. The results shown in Table 2-4 shows that the EfficientNetV2 model outperforms the other models and attained almost consistent results on the grape plant dataset, and EfficientNetV2-L attained a maximum result of 98.26% detection accuracy on the cardamom plant dataset. Table 6 shows a comparison of the proposed approach with the state-of-the-art methods. Employing U^2-Net with EfficientNetV2 outperforms for real-time cardamom plant leaf disease detection with a 98.26% detection accuracy.

V. CONCLUSION AND FUTURE WORK

An efficient plant leaf disease detection approach is essential to detect plant diseases in real-time. In this regard, the cardamom plant leaf disease detection approach is proposed, where the cardamom plant leaf dataset was collected from a farmland with a complex background. Segmenting and detecting diseases in real-time images is a challenging task, as the images are associated with other factors such as the background of the image, environmental factors such as lighting, and angle of the capturing conditions. In the proposed method, the U^2-Net architecture is employed to remove the complex background, which produces results without deteriorating the quality of the original image. For classification, in this work, CNN, EfficientNet, and EfficientNetV2 models were trained instead of using the pre-trained weights for EfficientNet, and EfficientNetV2. EfficientNetV2-S and EfficientNetV2-L models outperformed the other models; EfficientNetV2-L achieved 98.26% detection accuracy for the cardamom plant dataset, and EfficientNetV2-S achieved 98.28% detection accuracy for the cardamom plant dataset on external testing.

The dataset can be enhanced by collecting images of cardamom diseases with nutrition deficiency in future work. Furthermore, the model can be extended to identify the severity of the disease and nutrition deficiency.

ACKNOWLEDGMENT

The cardamom plant leaf images were labeled with the help of Dr. Harsha K. N., Scientist, Indian Cardamom Research Institute, Regional Station Sakaleshpur, Karnataka, India.

DATASET AND CODE

Dataset and Code is available, upon reasonable request. https://github.com/sunilchinnahalli/Cardamom-Dataset-2021; e-mail: sunilchinnahalli@gmail.com

REFERENCES

[1] M. J. Manju, S. K. Mushrif, T. H. Shankarappa, L. N. Hegde, A. Gowda, M. S. Lokesh, and N. Naik, "Evaluation of new fungicides against the leaf blight disease of cardamom [Elettaria cardamomum Maton] in arecanut based intercropping system," *J. Pharmacognosy Phytochem.*, pp. 283–285, Mar. 2018.

[2] F. Fina, P. Birch, R. Young, J. Obu, B. Faithpraise, and C. Chatwin, "Automatic plant pest detection and recognition using K-means clustering algorithm and correspondence filters," *Int. J. Adv. Biotechnol. Res.*, vol. 4, no. 2, pp. 189–199, Jul. 2013.

[3] A.-K. Mahlein, M. T. Kuska, J. Behmann, G. Polder, and A. Walter, "Hyperspectral sensors and imaging technologies in phytopathology: State of the art," *Annu. Rev. Phytopathol.*, vol. 56, no. 1, pp. 535–558, Aug. 2018.

[4] M. Ebrahimi, M. Khoshtaghaza, S. Minaei, and B. Jamshidi, "Vision-based pest detection based on SVM classification method," *Comput. Electron. Agricult.*, vol. 137, pp. 52–58, May 2017.

[5] X. Qin, Z. Zhang, C. Huang, M. Dehghan, O. R. Zaiane, and M. Jagersand, "U^2-Net: Going deeper with nested U-structure for salient object detection," *Pattern Recognit.*, vol. 106, Oct. 2020, Art. no. 107404.

[6] M. Tan and Q. V. Le, "EfficientNetV2: Smaller models and faster training," 2021, *arXiv:2104.00298*.

[7] G. L. Manso, H. Knidel, R. A. Krohling, and J. A. Ventura, "A smartphone application to detection and classification of coffee leaf miner and coffee leaf rust," 2019, *arXiv:1904.00742*.

[8] S. Zhang, X. Wu, Z. You, and L. Zhang, "Leaf image based cucumber disease recognition using sparse representation classification," *Comput. Electron. Agricult.*, vol. 134, pp. 135–141, Mar. 2017.

[9] C.-H. Yeh, M.-H. Lin, P.-C. Chang, and L.-W. Kang, "Enhanced visual attention-guided deep neural networks for image classification," *IEEE Access*, vol. 8, pp. 163447–163457, 2020.

[10] S. S. Chouhan, U. P. Singh, and S. Jain, "Applications of computer vision in plant pathology: A survey," *Arch. Comput. Methods Eng.*, vol. 27, no. 2, pp. 611–632, Apr. 2020.

[11] J. Xiong, D. Yu, S. Liu, L. Shu, X. Wang, and Z. Liu, "A review of plant phenotypic image recognition technology based on deep learning," *Electronics*, vol. 10, no. 1, p. 81, Jan. 2021.

[12] V. Singh and A. K. Misra, "Detection of plant leaf diseases using image segmentation and soft computing techniques," *Inf. Process. Agricult.*, vol. 4, pp. 41–49, Mar. 2017.

[13] A. Pourreza, W. S. Lee, R. Ehsani, J. K. Schueller, and E. Raveh, "An optimum method for real-time in-field detection of huanglongbing disease using a vision sensor," *Comput. Electron. Agricult.*, vol. 110, pp. 221–232, Jan. 2015.

[14] M. H. Saleem, S. Khanchi, J. Potgieter, and K. M. Arif, "Image-based plant disease identification by deep learning meta-architectures," *Plants*, vol. 9, no. 11, p. 1451, Oct. 2020.

[15] D. Tabernik, S. Sela, J. Skvarc, and D. Skocaj, "Segmentation-based deep-learning approach for surface-defect detection," *J. Intell. Manuf.*, vol. 31, pp. 759–776, May 2010.

[16] H. Weng, J. Lv, H. Cen, M. He, Y. Zeng, S. Hua, H. Li, Y. Meng, H. Fang, and Y. He, "Hyperspectral reflectance imaging combined with carbohydrate metabolism analysis for diagnosis of citrus huanglongbing in different seasons and cultivars," *Sens. Actuators B, Chem.*, vol. 275, pp. 50–60, Dec. 2018.

[17] Y. L. Cui, P. F. Cheng, X. Z. Dong, Z. H. Liu, and S. X. Wang, "Image processing and extracting color features of greenhouse diseased leaf," *Trans. CSAE*, vol. 21, pp. 32–35, Dec. 2005.

[18] J. Hang, D. Zhang, P. Chen, J. Zhang, and B. Wang, "Classification of plant leaf diseases based on improved convolutional neural network," *Sensors*, vol. 19, no. 19, p. 4161, Sep. 2019.

[19] J. Zhang, F. Kong, J. Wu, Z. Zhai, S. Han, and S. Cao, "Cotton disease identification model based on improved VGG convolution neural network," *J. China Agricult. Univ.*, vol. 23, no. 11, pp. 161–171, 2018.

[20] M. Sibiya and M. Sumbwanyambe, "A computational procedure for the recognition and classification of maize leaf diseases out of healthy leaves using convolutional neural networks," *AgriEngineering*, vol. 1, no. 1, pp. 119–131, Mar. 2019.

[21] J. Shin, Y. K. Chang, B. Heung, T. Nguyen-Quang, G. W. Price, and A. Al-Mallahi, "A deep learning approach for RGB image-based powdery mildew disease detection on strawberry leaves," *Comput. Electron. Agricult.*, vol. 183, Apr. 2021, Art. no. 106042.

[22] R. Karthik, M. Hariharan, S. Anand, P. Mathikshara, A. Johnson, and R. Menaka, "Attention embedded residual CNN for disease detection in tomato leaves," *Appl. Soft Comput.*, vol. 86, Jan. 2020, Art. no. 105933.

[23] J. G. Arnal Barbedo, "Plant disease identification from individual lesions and spots using deep learning," *Biosystems Eng.*, vol. 180, pp. 96–107, Apr. 2019, doi: 10.1016/j.biosystemseng.2019.02.002.

[24] E. C. Too, L. Yujian, S. Njuki, and L. Yingchun, "A comparative study of fine-tuning deep learning models for plant disease identification," *Comput. Electron. Agric.*, vol. 161, pp. 272–279, Jun. 2019, doi: 10.1016/j.compag.2018.03.032.

[25] K. Zhang, Z. Xu, S. Dong, C. Cen, and Q. Wu, "Identification of peach leaf disease infected by Xanthomonas campestris with deep learning," *Eng. Agricult., Environ. Food*, vol. 12, no. 4, pp. 388–396, Oct. 2019, doi: 10.1016/j.eaef.2019.05.001.

[26] S. Zhang, S. Zhang, C. Zhang, X. Wang, and Y. Shi, "Cucumber leaf disease identification with global pooling dilated convolutional neural network," *Comput. Electron. Agricult.*, vol. 162, pp. 422–430, Jul. 2019, doi: 10.1016/j.compag.2019.03.012.

[27] 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," *Comput. Electron. Agricult.*, vol. 154, pp. 18–24, Nov. 2018, doi: 10.1016/j.compag.2018.08.048.

[28] M. Brahimi, K. Boukhalfa, and A. Moussaoui, "Deep learning for tomato diseases: Classification and symptoms visualization," *Appl. Artif. Intell.*, vol. 31, no. 4, pp. 299–315, Apr. 2017, doi: 10.1080/08839514.2017.1315516.

[29] J. G. M. Esgario, R. A. Krohling, and J. A. Ventura, "Deep learning for classification and severity estimation of coffee leaf biotic stress," *Comput. Electron. Agricult.*, vol. 169, Feb. 2020, Art. no. 105162, doi: 10.1016/j.compag.2019.105162.

[30] Q. Wu, K. Zhang, and J. Meng, "Identification of soybean leaf diseases via deep learning," *J. Inst. Eng. (India): A*, vol. 100, no. 4, pp. 659–666, Dec. 2019.

[31] Y. Zhong and M. Zhao, "Research on deep learning in apple leaf disease recognition," *Comput. Electron. Agricult.*, vol. 168, Jan. 2020, Art. no. 105146.

[32] Z. Jiang, Z. Dong, W. Jiang, and Y. Yang, "Recognition of Rice leaf diseases and wheat leaf diseases based on multi-task deep transfer learning," *Comput. Electron. Agricult.*, vol. 186, Jul. 2021, Art. no. 106184.

[33] U. P. Singh, S. S. Chouhan, S. Jain, and S. Jain, "Multilayer convolution neural network for the classification of mango leaves infected by anthracnose disease," *IEEE Access*, vol. 7, pp. 43721–43729, 2019.

[34] S. S. Chouhan, U. P. Singh, and S. Jain, "Automated plant leaf disease detection and classification using fuzzy based function network," *Wireless Pers. Commun.*, vol. 12, pp. 1757–1779, Jul. 2021.

[35] S. S. Chouhan, U. P. Singh, and S. Jain, "Web facilitated anthracnose disease segmentation from the leaf of mango tree using radial basis function (RBF) neural network," *Wireless Pers. Commun.*, vol. 113, no. 2, pp. 1279–1296, Jul. 2020.

[36] S. S. Chouhan, A. Kaul, U. P. Singh, and S. Jain, "Bacterial foraging optimization based radial basis function neural network (BRBFNN) for identification and classification of plant leaf diseases: An automatic approach towards plant pathology," *IEEE Access*, vol. 6, pp. 8852–8863, 2018.

[37] J. Lu, J. Hu, G. Zhao, F. Mei, and C. Zhang, "An in-field automatic wheat disease diagnosis system," *Comput. Electron. Agricult.*, vol. 142, pp. 369–379, Nov. 2017.

[38] A. Johannes, A. Picon, A. Alvarez-Gila, J. Echazarra, S. Rodriguez-Vaamonde, A. D. Navajas, and A. Ortiz-Barredo, "Automatic plant disease diagnosis using mobile capture devices, applied on a wheat use case," *Comput. Electron. Agricult.*, vol. 138, pp. 200–209, Jun. 2017.

[39] M. Arsenovic, M. Karanovic, S. Sladojevic, A. Anderla, and D. Stefanovic, "Solving current limitations of deep learning based approaches for plant disease detection," *Symmetry*, vol. 11, no. 7, p. 939, Jul. 2019.

[40] T. T. Santos, L. L. de Souza, A. A. dos Santos, and S. Avila, "Grape detection, segmentation, and tracking using deep neural networks and three-dimensional association," *Comput. Electron. Agricult.*, vol. 170, Mar. 2020, Art. no. 105247.

[41] P. Ganesh, K. Volle, T. F. Burks, and S. S. Mehta, "Deep orange: Mask R-CNN based orange detection and segmentation," *IFAC-PapersOnLine*, vol. 52, no. 30, pp. 70–75, 2019.

[42] F. Meyer, "Topographic distance and watershed lines," *Signal Process.*, vol. 38, no. 1, pp. 113–125, 1994.

[43] D. Morris, "A pyramid CNN for dense-leaves segmentation," in *Proc. 15th Conf. Comput. Robot Vis. (CRV)*, May 2018, pp. 238–245.

[44] P. Singh, A. Verma, and J. S. R. Alex, "Disease and pest infection detection in coconut tree through deep learning techniques," *Comput. Electron. Agricult.*, vol. 182, Mar. 2021, Art. no. 105986.

[45] L. M. Tassis, J. E. Tozzi de Souza, and R. A. Krohling, "A deep learning approach combining instance and semantic segmentation to identify diseases and pests of coffee leaves from in-field images," *Comput. Electron. Agricult.*, vol. 186, Jul. 2021, Art. no. 106191.

[46] S. S. Chouhan, U. P. Singh, U. Sharma, and S. Jain, "Leaf disease segmentation and classification of jatropha curcas L. And pongamia pinnata L. Biofuel plants using computer vision based approaches," *Measurement*, vol. 171, Feb. 2021, Art. no. 108796.

[47] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Frontiers Plant Sci.*, vol. 7, p. 1419, Sep. 2016. [Online]. Available: https://github.com/spMohanty/PlantVillage-Dataset

[48] Y. Xiong, L. Liang, L. Wang, J. She, and M. Wu, "Identification of cash crop diseases using automatic image segmentation algorithm and deep learning with expanded dataset," *Comput. Electron. Agricult.*, vol. 177, Oct. 2020, Art. no. 105712.

[49] J. Chen, J. Chen, D. Zhang, Y. Sun, and Y. A. Nanehkaran, "Using deep transfer learning for image-based plant disease identification," *Comput. Electron. Agricult.*, vol. 173, Jun. 2020, Art. no. 105393, doi: 10.1016/j.compag.2020.105393.

[50] A. Cruz, Y. Ampatzidis, R. Pierro, A. Materazzi, A. Panattoni, L. De Bellis, and A. Luvisi, "Detection of grapevine yellows symptoms in vitis vinifera L. With artificial intelligence," *Comput. Electron. Agricult.*, vol. 157, pp. 63–76, Feb. 2019.

[51] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6105–6114.

[52] J. Chen, H. Yin, and D. Zhang, "A self-adaptive classification method for plant disease detection using GMDH-logistic model," *Sustain. Comput., Informat. Syst.*, vol. 28, Dec. 2020, Art. no. 100415.

[53] Ü. Atila, M. Uçar, K. Akyol, and E. Uçar, "Plant leaf disease classification using efficientnet deep learning model," *Ecol. Informat.*, vol. 61, Mar. 2021, Art. no. 101182.

SUNIL C. K. (Student Member, IEEE) is currently pursuing the Ph.D. degree with the Department of Information Technology, National Institute of Technology Karnataka, Surathkal, India. His research interests include machine learning and deep learning.

JAIDHAR C. D. is currently working as an Associate Professor at the National Institute of Technology Karnataka, Surathkal, India. His primary research interests include computer networks, cyber security, the Internet of Things, and precision agriculture.

NAGAMMA PATIL received the Ph.D. degree in computer science and engineering from the Indian Institute of Technology Roorkee, India. She is with the Department of Information Technology, National Institute of Technology Karnataka, Surathkal, India. Her research interests include data mining, soft computing, big data analytics, machine learning, and bioinformatics.