RESEARCH ARTICLE

Detection of Tomato Leaf Diseases for Agro-Based Industries Using Novel PCA DeepNet

KYAMELIA ROY[1], SHELI SINHA CHAUDHURI[1], (Member, IEEE), JAROSLAV FRNDA [2], (Senior Member, IEEE), SRIJITA BANDOPADHYAY[3], ISHAN JYOTI RAY[3], SOUMEN BANERJEE [4], (Senior Member, IEEE), AND JAN NEDOMA [5], (Senior Member, IEEE)

[1]Department of Electronics and Telecommunication Engineering, Jadavpur University, Kolkata 700032, India
[2]Department of Quantitative Methods and Economic Informatics, Faculty of Operation and Economics of Transport and Communication, University of Zilina, 010 26 Zilina, Slovakia
[3]Department of Electronics and Communication Engineering, University of Engineering and Management, Kolkata 700160, India
[4]Department of Electronics and Communication Engineering, Budge Budge Institute of Technology, Kolkata 700137, India
[5]Department of Telecommunications, VSB—Technical University of Ostrava, 70800 Ostrava, Czech Republic

Corresponding author: Soumen Banerjee (prof.sbanerjee@gmail.com)

ABSTRACT The advancement of Deep Learning and Computer Vision in the field of agriculture has been found to be an effective tool in detecting harmful plant diseases. Classification and detection of healthy and diseased crops play a very crucial role in determining the rate and quality of production. Thus the present work highlights a well-proposed novel method of detecting Tomato leaf diseases using Deep Neural Networks to strengthen agro-based industries. The present novel framework is utilized with a combination of classical Machine Learning model Principal Component Analysis (PCA) and a customized Deep Neural Network which has been named as PCA DeepNet. The hybridized framework also consists of Generative Adversarial Network (GAN) for obtaining a good mixture of datasets. The detection is carried out using the Faster Region-Based Convolutional Neural Network (F-RCNN). The overall work generated a classification accuracy of 99.60% with an average precision of 98.55%; giving a promising Intersection over Union (IOU) score of 0.95 in detection. Thus the presented work outperforms any other reported state-of-the-art.

INDEX TERMS Tomato leaf diseases, artificial intelligence, deep learning, computer vision, generative adversarial networks, convolutional neural network, faster region-based convolutional neural network.

I. INTRODUCTION

Agriculture is one of the oldest occupations practiced worldwide in the majority of the countries. It forms an important aspect of sustainability and survivability. India being a country primarily having an agriculture-based economy, hence a major part of India's population is either directly or indirectly linked to agriculture. Moreover, agricultural exports majorly contribute to the country's GDP. According to the available data, the global population is expected to reach 10 billion by 2050 which in turn would require agricultural productivity to increase by at least 70%. With increase in world population, the demand for agricultural products has increased manifold. As further cultivable land area cannot be increased, hence the only way to increase the amount of agricultural production is to enhance the productivity of the existing lands. Agriculture in many parts of India is still practiced manually in a traditional manner, involving lots of manpower and man-hours. Among several aspects of agriculture, keeping crops disease-free is an important facet. This is done manually and may turn out to be inaccurate owing to some limitations and incorrect judgments of humans which in this case might result

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

in catastrophic consequences; the worst being the whole crop getting ruined. This is a problem that no farmer can afford and hence resort to Artificial Intelligence (AI) based solutions. AI in agriculture serves a better pathway to analyze real-time problems faced by the farmers in day-to-day life. One of the common problems faced is the invasion of pests, which deteriorates the quality of the crops. The main challenge lies in detecting the diseases caused due to the attack of these pests and for which farmers need innovative technologies to combat such attacks. The joint venture of Computer Vision and Artificial Intelligence makes it a great way to solve such kinds of problems. The most promising factor that rules AI is that it uses real-time data not only to predict the emergence but also the identification of the pest and diseases before it takes a huge shape. Therefore, the main motto of developing such automated systems using AI is to reduce the vulnerability of pest attack and to preserve the quality of production.

Plant disease detection using Deep Learning techniques such as classification and detection has become a crucial aspect in monitoring and analyzing the productivity of each and every specific species of plant. As compared to traditional classification networks, techniques involving deep learning yields better results for real-time identification of plant leaf diseases [1]. These are all headed under the latest improvements in computer vision aided systems to efficiently provide solutions for multiple plant diseases as the existing method for disease detection is through naked eyes which require lots of effort and is time consuming as well [2]. Therefore to reduce this problem Deep Learning has been introduced which involves a robust process with higher accuracy for accurate diagnosis of the respective diseases [3].

The proposed work for detection of Tomato leaf diseases is accomplished with a Deep Learning approach combined with Machine Learning technique. The overall framework depicts the classification of Tomato leaf images into healthy and diseased ones and then implementation of the images for different categories of disease detection. The entire work has been implemented using Deep Neural Networks, especially using Convolution Neural Network (CNN) architectures and PCA and this new model is named as PCA DeepNet. The PCA work as the primary feature extractor followed by the customized deep neural networks for classification and detection purposes. The convolutional deep learning networks are basically chosen to reduce computational cost and for smooth classification; thereby helping in development of an intelligent systems assisted tomato leaf disease detection. A single shot detector (SSD) and Faster Region Based Convolutional Neural Network (F-RCNN) is used for detection purposes. In F-RCNN, the detection steps are carried out in two steps unlike SSD and hence a more accurate detection is obtained using F-RCNN [4].

The major contributions of the work are listed below which differentiates the presented PCA DeepNet from the existing works available in literature:

- The work is methodized by a hybridized framework consisting of GANs, conventional PCA, customized CNN classifier and detection architecture for respective purposes.
- The system consists of a customized neural network which is basically structured using 10 convolutional neural layers. This is formulated utilizing a feed forward network where the method of down-sampling and up-sampling is predominant and the work is tuned using several hyper-parameters like Learning rate, Optimizers, Activation Functions to enhance the performance of the model.
- The changes made in the dropout values, pooling layers and max pooling layers gives the proposed work a better formulation and thus reduces time complexity of the entire system. The lesser time consumption is compared with respect to the hardware and software specification mentioned later in the results and discussion section.
- The newer system which has not been implemented earlier generates better accuracy and other performance metrics like precision, recall, F1-measure scores.
- The novelty of the overall framework is materialized through a customized hybridized model involving classical machine learning model along with deep neural networks which are structured in such a way so as to outperform any of the existing works.

The remaining paper is presented as follows: Section II contains related works, Section III comprises methodology of the overall process, Section IV throws light on the results obtained and finally Section V concludes the entire study.

II. RELATED WORKS

Detection and classification methods for different diseases in leaves, plants and crops have been exhaustively studied in the recent past. Even studies involving the causes of diseases in crops through attack by different pests have also been investigated to reduce the crop yield losses [5]. The classification of leaf diseases using deep learning is mainly done with the help of Transfer learning techniques. In [6] a light weight transfer learning based approach is adopted for efficient detection of tomato leaf diseases. Herein a pretrained MobileNetV2 architecture along with a classifier network is used for prediction yielding 99.3% accuracy. Also the MobileNetV2 architecture has been used in [7] for Bean leaf diseases classification yielding more than 97% and 92% accuracy results on training and testing dataset respectively. In [8] a brief analysis shows that the VGG family provides better accuracy while validating the dataset for classification as well as detection and VGG 16 gave an accuracy of 99.25%. The state-of-the-art results was also generated when several other CNN architectures like Resnet-50, Xception, Mobilenet, ShuffleNet, Densenet121_Xception were used for feature extraction and studies on the comparative detailing of these architectures [9] revealed that Densenet121_Xception gave the best training accuracy of 97% followed by ShuffleNet which could only

give 83.68. Many more CNN classification architectures like AlexNet and SqueezeNet highlighted some of the good parameters for a detailed study. In [10] a compact CNN is proposed by the authors for tomato leaf disease identification involving six layers network. Another new model based on CNN Architecture was developed with Adam optimizer and with the help of Image Augmentation an accuracy of 96.55% was achieved. Also by using CaffeNet Architecture and fine-tuning the data a similar accuracy of 96.3% was obtained by authors in [11]. Leaf disease detection has been subjected to several other comparative studies using Alexnet and ResNet by [12] where AlexNet fetched a better accuracy of 97%.

Authors in [13] have developed their own feature extraction technique using GLCM, Complex Gabor Filter, Curvelet and Image moments; further they have trained their Neuro-Fuzzy logic classifier with feature extractor using MATLAB simulation tool. Another approach to implementing MATLAB is used in [14], where Image-Segmentation and feature extraction using Color co-occurrence method is proposed and finally classification is done using Back Propagation Neural Network (BPNN). In recent years many new upcoming techniques have been formulated using deep learning, some of which are Precision farming to enhance production and Soft Computing Technologies involving Segmentation processes.

In recent years, machine learning has enabled creation of newer ways for efficient disease detection. In [15] a survey on different machine learning classifiers like support vector method (SVM), k-nearest neighbor and fuzzy logic has been carried out to get an overview of these algorithms. Many changes in algorithms and techniques were done later on to create better output using clustering and classification techniques. Algorithms such as SVM and K-mean Segmentation also generated better accuracy of 90%. A new method presented in [16] used low-level features of Luminance and colour along with multi-scale analysis for determining saliency maps and then using k-means algorithm for soybean leaf disease detection. Another approach [17] is adopted using SIFT for extraction of features and analyzing the results using an SVM classifier. Some researchers use different architectures and algorithms to acquire the desired results. In [18] the images were segmented using K-Means Clustering, extracted the features with GLCM and LBP and finally classified them using SVM. Regression techniques were used by researchers as reported in [19] where SVR (Support Vector Regression) and GPR (Gaussian Process Regression) have been used. SVR was used in the estimation of biochemical and biophysical parameters of the plant while GPR, a kernel-based machine learning method, was used for nonlinear regression problems.

GANs (Generative Adversarial Networks) are widely used nowadays by researchers as tools for image augmentation. GANs increase efficiency of classification models to a greater margin. Researchers in [20] proposed the first work using GANs for synthetic augmentation of the dataset in improving performance of plant disease recognition. Authors optimized the activation reconstruction loss (ARL) function that put forward an enhanced AR-GAN, comparing it with prominent existing models.

The proposed model provided a significant increase of about 5.2% in the classification accuracy as compared to the classical ones. As reported in [21], using DCGAN to augment the dataset, researchers were able to achieve 20% higher accuracy than those using conventional tools for augmentation. GANs were also able to solve another major problem of data imbalance or class imbalance; also providing endless high-quality data. In [22], Double GAN approach is adopted wherein two GANs are used to obtain a pre-trained model and SRGAN is used to increase the residual network to prevent overfitting. Researchers in [23] have generated augmented images using C-DCGAN (Conditional Deep Convolutional GAN) as the input to VGG16 and found average accuracy to be around 28% higher than conventional methods like rotation and translation. Authors in [24] augmented four types of grape leaf disease with a novel Leaf GAN model. The experimental results revealed that the Leaf GAN model could make the images highlight the disease and could also generate enough synthetic grape leaf disease images, proving that the method is superior in comparison to DCGAN and WGAN.

III. METHODOLOGY

The proposed PCA DeepNet is meticulously designed for performance where each and every step of data preparation and analysis has been optimized for best results. The pre-processed image data is first made to go through an augmentation process using GANs, where the data is refined and made more trainable for further processes. Then the data is processed with a feature extraction technique performed using conventional Principal Component Analysis (PCA). Thereafter, the classification of the data, which is the highlight of PCA DeepNet, is done using a customized CNN Classifier specifically designed to process the data. At last, the classified outputs are detected using a faster region-based CNN. The overall system workflow is shown in Fig.1.

A. DATA SET
For training a Deep Learning model to be able to classify precisely and with high accuracy, an image dataset with proper and balanced image samples is very essential [25].

The larger the dataset size, the more accurate the deep learning model can be obtained. Hence the authors have selected the Plant Village dataset [26] which is an open-source agricultural disease dataset. It is a collection of more than 56000 images divided into 38 classes consisting of 19 crops (apple, grapes, tomato, potato etc.). The dataset consists of high-quality images of leaves in .jpeg format with a width of 5472 pixels and a height of 3648 pixels. Among the 19 crops, the authors have only taken tomatoes into consideration and some of them are shown in Fig. 2.
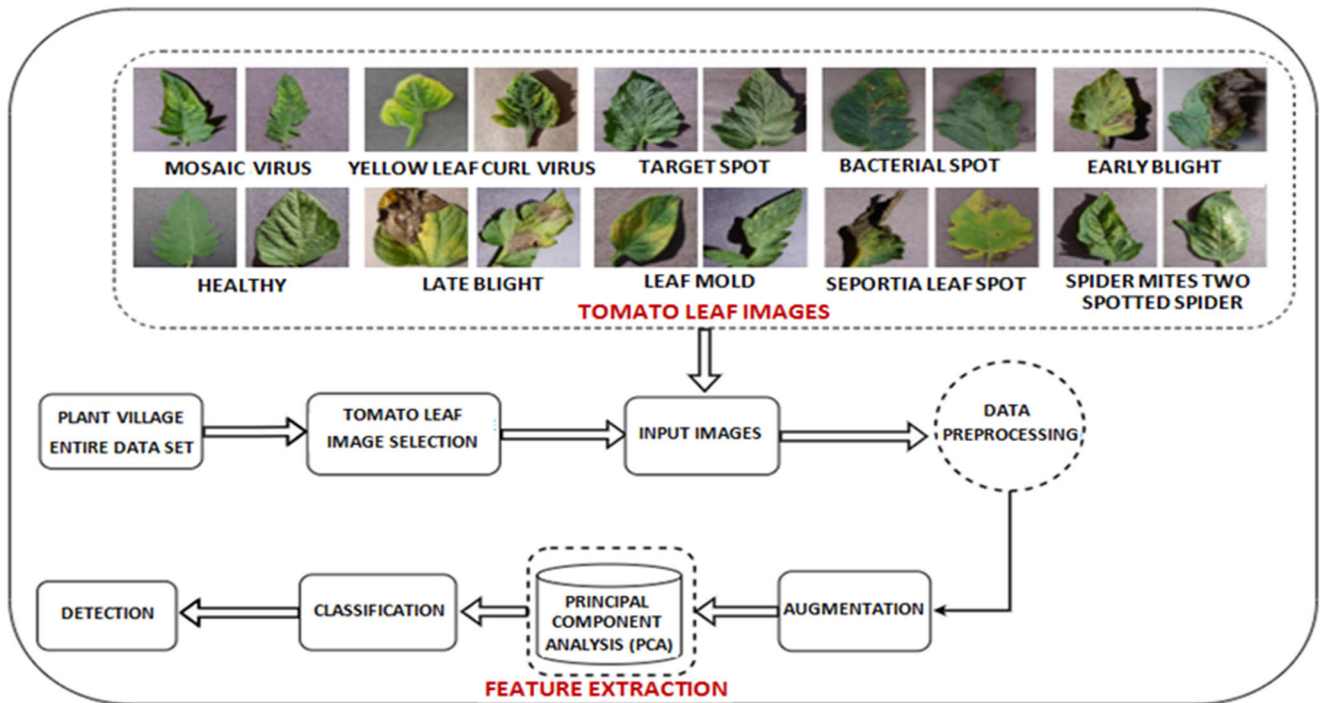
FIGURE 1. Block diagram of overall System Workflow.

The tomato data is distributed into 10 different classes namely Late_blight, Healthy, Early_blight, Seportia_leaf_ spot, Yellow_leaf_curl_virus, Bacteria_spot, Target_spot, Mosaic_virus, Leaf_mold and Spider_Mites_two_spotted_ sider which consists of a total of 18,128 images of tomato leaves. Table 1 highlights the overall data employed in this work. The enlisted table also throws light on the entire images utilized to carry forward the current framework. Thus a multi-class classification of the dataset is performed in the presented work.

B. DATA PRE-PROCESSING

The collected data which is used as input images are pre-processed to make it trainable with the proposed model. The whole dataset consisting of 18,128 images is split into training and validation sets in ratio of 7:3, where each set consists of all the 10 classes. After the segregation of the dataset, each and every image is combined separately according to the diseases. The graphical representation of the overall data pre-processing technique is exhibited in Fig.3 while its schematic representation is shown in Fig. 4. In the proposed work a classical method of splitting the data into train and test is carried out to grasp the convenience of easy compatibility with the current dataset.

C. ANNOTATION

Annotation of the images plays a major role in detection of different diseases. The annotation of the images in this work is mainly done using labeling software, an open source graphics annotation tool. The images are labeled using bounding boxes; the most commonly used type of annotation in object detection and localization tasks. Fig. 5(a) and Fig. 5(b) depict the picture and the XML document associated to label these images using python. The different kinds of diseased images have been identified by professional experts in the agriculture domain. Detailed information about the annotation of each class is given in Table 2.

The entire table contains all the detail information regarding all the images which are annotated for computing the detection process. The annotation is done in PASCAL format since the detection architecture used in the presented work is Faster Region Based Convolutional Neural Network.

D. DATA AUGMENTATION

Data augmentation is a technique of refining the dataset which facilitates the training of classification models. In agricultural disease datasets like the Plant Village dataset, the onset period of certain diseases is shorter, which makes it difficult to collect enough samples of them. In the field of deep learning, small sample size and data imbalance are major factors leading to poor recognition and classification. As a mitigation technique, data augmentation is applied to artificially increase the amount of data by generating new data points from existing data. Here the authors have used Generative Adversarial Networks (GANs) as a modern approach to data augmentation. Unlike any other conventional augmentation models, GAN aims at learning the distribution of a training dataset to generate new (synthetic) data instances.
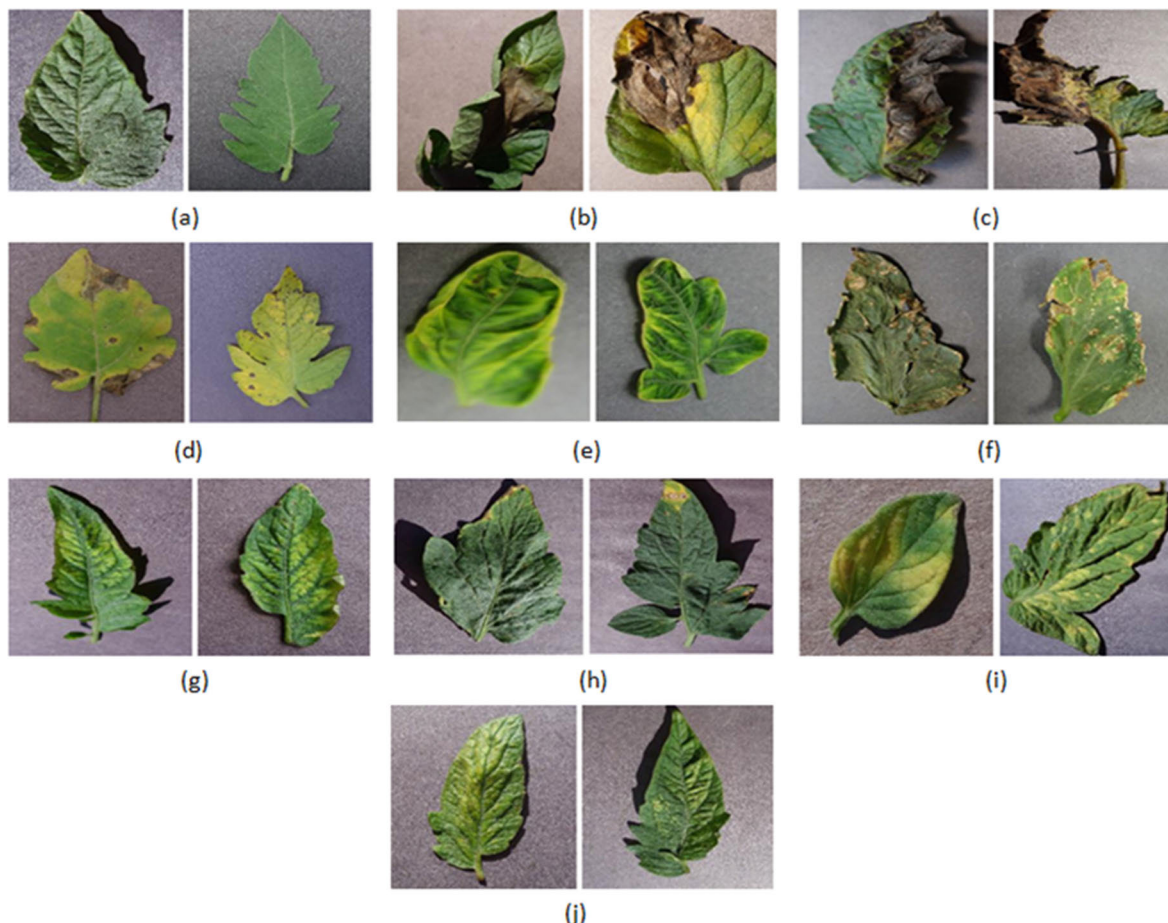
FIGURE 2. Dataset images – (a) Healthy (b) Late_blight (c) Early_blight (d) Seportia_leaf_spot (e) Yellow_leaf_curl_virus (f) Bacteria_spot (g) Target_spot (h) Mosaic_virus (i) Leaf_mold (j) Spider_Mites_two_spotted_spider.

TABLE 1. Detail information of dataset employed.

Tomato leaf Diseases Classes	Total images Per classes	Training images	Testing images
Tomato Bacterial Spot	2126	1488	637
Tomato Early Blight	1057	740	317
Tomato Late Blight	1938	1356	582
Tomato Leaf Mold	952	667	285
Tomato Septoria Leaf Spot	1860	1302	558
Tomato Spider Mites	1700	1190	510
Tomato Target Spot	1386	971	415
Tomato Yellow Leaf Curl Virus	5456	3819	1637
Tomato Mosaic Virus	400	280	120
Tomato Healthy	1662	1163	499

The GAN model comprises two sub-models: generator and discriminator, which work against each other. The discriminator is trained on both real and fake data. It learns to get better at distinguishing the generated fake data and real data and the generator learns to generate more realistic new data points from random inputs. The process continues until the
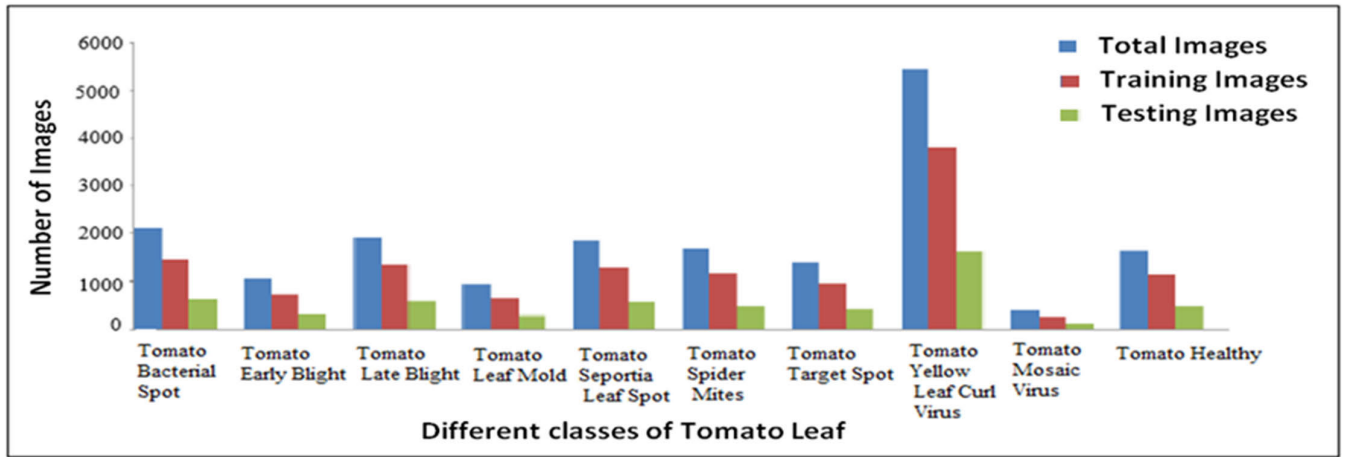
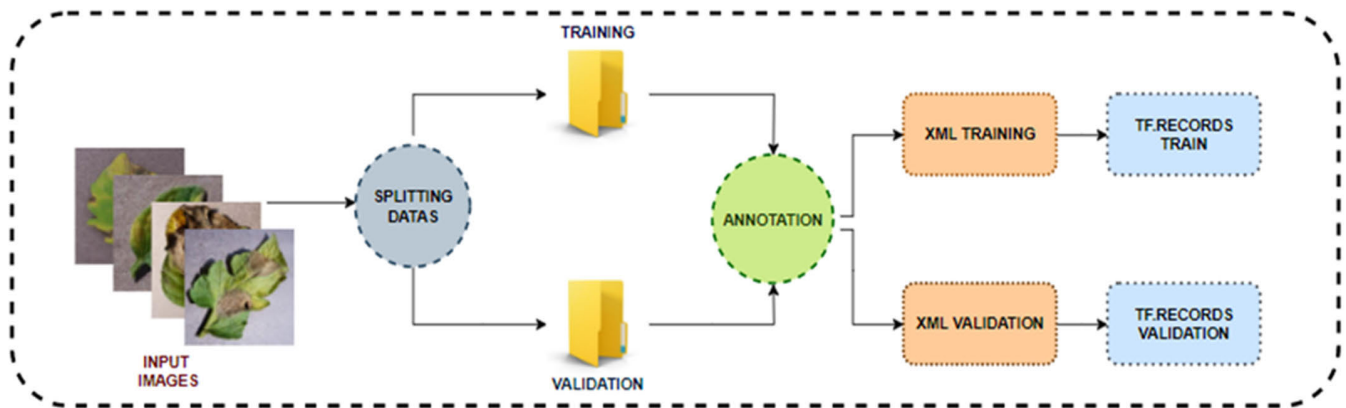**FIGURE 3.** Graphical representation of data pre-processing.

FIGURE 4. Schematic diagram of data pre-processing.

generator can create data instances that the discriminator cannot distinguish from real data. Out of the various types of GANs suitable for different purposes, here CycleGAN is used owing to its suitability for image augmentation. The most important feature of CycleGAN is that it can perform image translation on an unpaired image where there is no relation between input and output images. A diagrammatic representation of the above process is shown in Fig. 6.

E. FEATURE EXTRACTION

The datasets commonly available these days have hundreds of features. If the number of features becomes similar to the number of observations stored in a dataset, then this can certainly lead to over fitting of the model. Amongst the different available Feature Extraction techniques, herein the presented work is accomplished using a classical machine learning feature extraction method known as Principal Component Analysis which is an unsupervised linear dimensionality reduction technique mainly used for feature extraction and reduction of image dimensions. PCA is primarily chosen as it aims toward finding the direction of maximum variance in high

dimensional data; thereby helping in easy identification of the object. PCA helps to project the original data into a set of orthogonal axes and each of the axes gets ranked in the order of importance [27]. Fig.7 shows the projection of different classes of tomato leaf after evaluating the eigen vectors using PCA.

F. CLASSIFICATION

Classification of images has been done using many hybrid models which includes a mixture of machine learning and deep learning models namely viz. AlexNet-SVM [28]. Classification is a major technique for classifying the diseased one amongst the healthy leaves. This step includes the usage of several machine learning models like SVM and KNN [29]. Classification of images can also be performed using pre-trained CNN models like Resnet-50, Xception, Mobilenet, ShuffleNet, Densnet121_Xception, AlexNet, GoogleNet, VGGNet [30] etc. The pre-trained models are saved networks that had been previously trained on a large scale dataset. It includes all steps like data augmentation, feature extraction, image classification etc. unlike the proposed PCADeepNet

```
<annotation>
        <folder>Tomato___Bacterial_spot</folder>
        <filename>0d56df83-84fb-4189-a5d2-3a6da18a224d___UF.GRC_BS_Lab L
        <path>E:\desktop\leaf_data\Tomato___Bacterial_spot\0d56df83-84fb
        <source>
                <database>Unknown</database>
        </source>
        <size>
                <width>256</width>
                <height>256</height>
                <depth>3</depth>
        </size>
        <segmented>0</segmented>
        <object>
                <name>Tomato_BS</name>
                <pose>Unspecified</pose>
                <truncated>0</truncated>
                <difficult>0</difficult>
                <bndbox>
                        <xmin>34</xmin>
                        <ymin>124</ymin>
                        <xmax>86</xmax>
                        <ymax>208</ymax>
                </bndbox>
        </object>
        <object>
                <name>Tomato_BS</name>
                <pose>Unspecified</pose>
                <truncated>0</truncated>
                <difficult>0</difficult>
                <bndbox>
                        <xmin>121</xmin>
                        <ymin>64</ymin>
                        <xmax>195</xmax>
                        <ymax>175</ymax>
                </bndbox>
        </object>
</annotation>
```

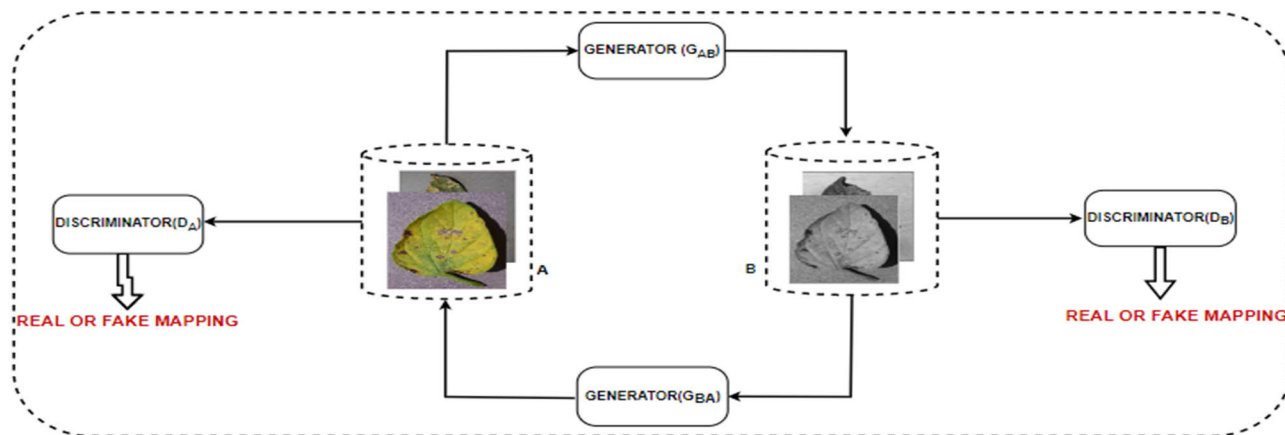**FIGURE 5.** Annotation of Tomato leaf images (a) Annotated image (b) XML document.

FIGURE 6. Image augmentation using CycleGAN.

where each step is customized according to the dataset for providing best results. For classifying the images, CNN is customized in such a way so that it contains a stack of 10 Conv2D layers where each convolutional layer consists of a pooling layer, dropout, a max-pooling layer and the activation function ReLU.

ReLU is best suited for multi-class classification and does not saturate for the positive value of the weighted sum of inputs. An input image is fed to the classifier after extracting the features using PCA. A series of 10 layers is made which consists of kernel size, stride and padding layer of different values as shown in Fig. 8. The entire classifier is

TABLE 2. Detail information of image annotation.

| Tomato leaf Classes from Plant Village Dataset	Annotation Labeling	Total images Per classes	Training images	Testing images
Tomato Bacterial Spot	Tomato_BS	2126	1488	637
Tomato Early Blight	Tomato_ER	1057	740	317
Tomato Late Blight	Tomato_LR	1938	1356	582
Tomato Leaf Mold	Tomato_LM	952	667	285
Tomato Septoria Leaf Spot	Tomato_SLS	1860	1302	558
Tomato Spider Mites	Tomato_SMP	1700	1190	510
Tomato Target Spot	Tomato_TS	1386	971	415
Tomato Yellow Leaf Curl Virus	Tomato_YLCV	5456	3819	1637
Tomato Mosaic Virus	Tomato_MV	400	280	120
Tomato Healthy	Tomato_Healthy	1662	1163	499
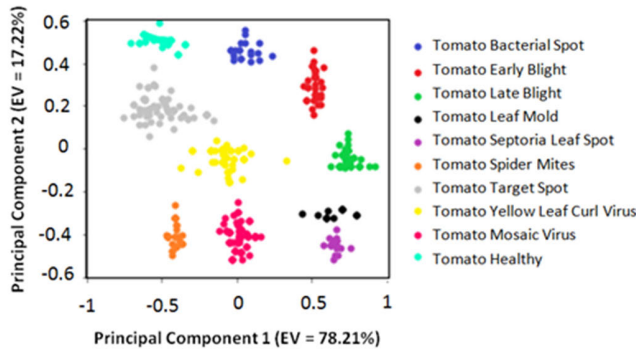
FIGURE 7. Projection of different classes of leaf after implementing PCA.

structured in such a way that the first five convolutional layers consist of a feed forward neural network that increases in size from 32 to 512. The next set of convolutional layers presenting down-sampling are concatenated with the previous layers of up-sampling. In the following model, the up-sampling method is utilized and conv6, conv7, conv8 and conv9 are designed by concatenating layers (conv5,conv4), (conv6,conv3), (conv7,conv2), (conv8,conv1) respectively. In the present work Categorical Cross-Entropy is used as the loss function, the formula for the same is given in equations 1 and 2 respectively. Equation 1 is for the Softmax activation function and equation 2 stands for the Cross Entropy.

$$f\,(s)_i = \frac{e^{s_i}}{\sum_j^C e^{s_j}} \tag{1}$$

$$CE = -\sum_i^C t_i \log(f\,(s)_i). \tag{2}$$

Here f(s) is the function, C is the class for which the probability needs to be calculated. The letter t stands for the target vector. The classifier also uses Adam as an optimizer with a learning rate of 0.01 and a momentum of 0.9 respectively. The mathematical formula associated with it is given in equation 3.

$$m_n = E[X^n] \tag{3}$$

where m is the moment and X is the random variable and n is the expected value of the moment. The outer layer is the fully connected layer which comprises the input from the convolutional layer 9 with a sigmoid activation function. The computation of the classifier using 35 epochs is performed and overall training time reaches to 25min. However an early stopping [31] function callback is used for resolving the problem of over fitting. The hybridized framework is trained using a huge dataset and the limited GPU helps in consumption of less time; thereby boosting the proposed work with utilization of less computational resources. The inference time calculated for the present classifier is less than any other models used. Thus the work highlights the efficacy of the presented work effectively.

G. PERFORMANCE PARAMETERS

The performance of the PCA DeepNet classifier model is evaluated based on several metrics which are calculated using True positive and True negative (Tp & Tn) and False positive and False negative (Fp & Fn) values obtained from the confusion matrix while training the models [32].

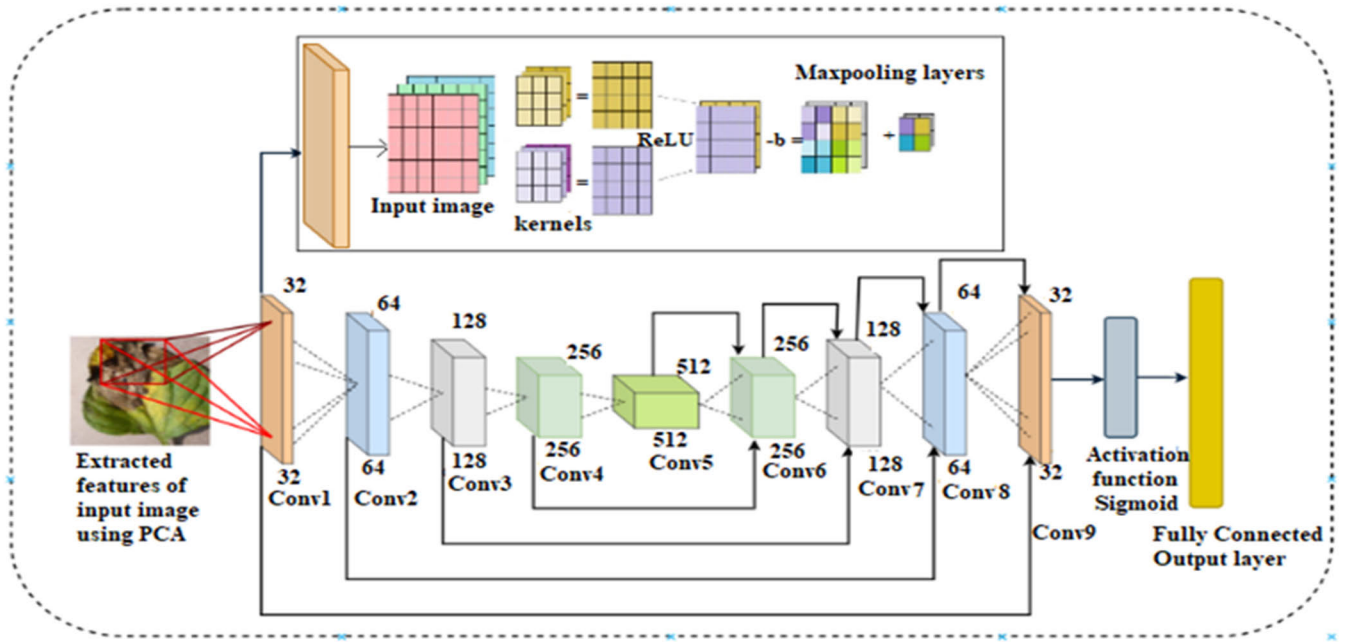1) Accuracy – It is the most crucial and intuitive performance measure which ensures the percentage of correct

FIGURE 8. Schematic representation of the classifier model.

prediction based on the total number of observations present.

$$Accuracy\,(ACC) = \frac{Tp + Tn}{Tp + Tn + Fp + Fn}$$

2) Precision – Also called Positive Predictive Value, it represents the number of samples actually and predicted as positive from the total number of samples predicted as positive. The precision measures the model's accuracy in classifying a sample as positive. In some cases, precision is preferred over recall because it does not depend on the false negative values which rules out the problems arising due to class imbalance.

$$Precision\,(PPV) = \frac{Tp}{Tp + Fp}$$

3) Recall – It is the number of samples actually and predicted as positive from the total number of samples actually positive. It gives a measure of how accurately the model is able to identify the data specifically the true positives.

$$Recall\,(Sensitivity) = \frac{Tp}{Tp + Fn}$$

4) F1-Measure – The F1-Score combines the precision and recall of a classifier into a single matrix by taking their harmonic mean. It measures a model's accuracy for a dataset. It is used to compare the performance of two classifier models.

$$F1 - Measure = 2 * \frac{Tp}{2 * Tp + Fp + Fn}$$

H. DETECTION

The proposed approach of the work is to train the data using customized PCA DeepNet model. In case of detection, the authors have used the model F-RCNN. In Faster RCNN, PCA DeepNet is used as the backbone architecture for training. An input image is used to train the model. This detected as well as perfectly localized the images due to its improved architecture. It uses two networks – one for region proposal and another for object detection. It consists of 9 anchors for creating bounding boxes of specific size. The output obtained in this architecture is due to the Region Proposal Network (RPN) which easily classifies the diseased and the healthy Tomato Leaf in the form of rectangular bounding boxes by reframing the anchor. The training accuracy of this model is higher. The architecture used produces an IoU score of 0.95 with a threshold score of 0.8. The entire process is diagrammatically represented in Fig. 9.

IV. RESULTS AND DISCUSSION

A detailed and exhaustive study has been carried out while validating the PCA DeepNet classifier model. The overall training has been processed using Google Colab with GPU specification of Tesla K80 (2496 CUDA cores). The pre-processed augmented data of 10 classes are taken as input and different classification metrics were generated while training it. The software and hardware specifications used in the work are enlisted in Table 3.

The Accuracy and loss graph of PCA DeepNet classifier is highlighted in Fig. 10(a) and Fig. 10(b). The graphs are validated against 50 epochs although for optimized fitting of the model for accuracy and loss an early stopping function
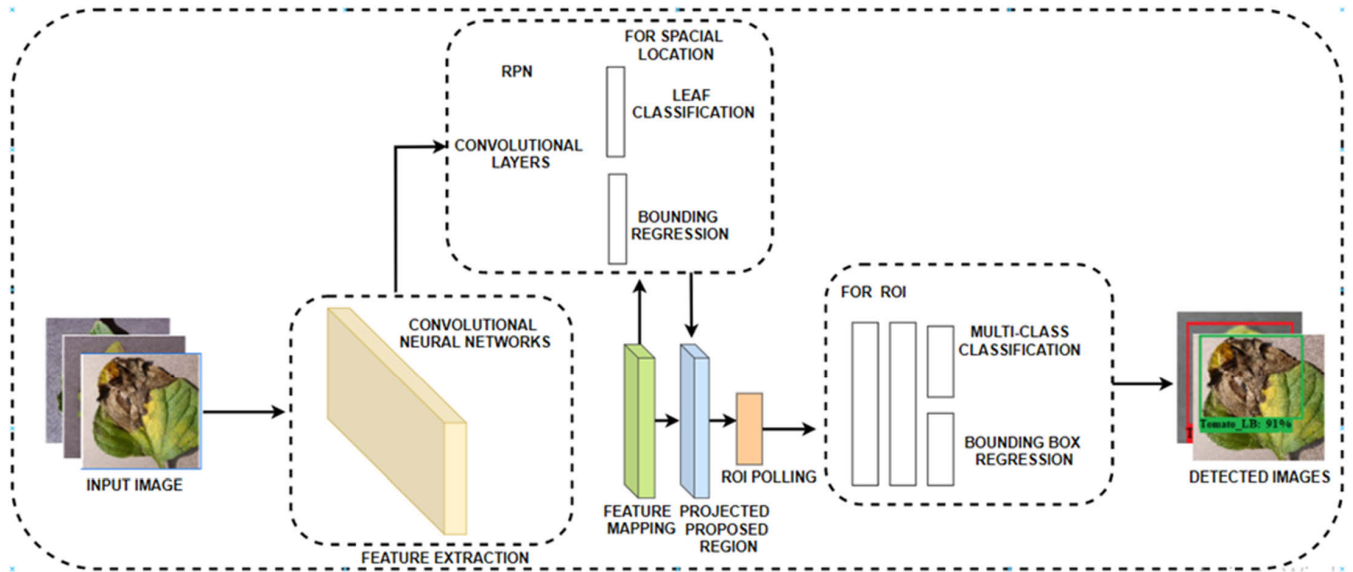
FIGURE 9. Block diagram of faster region-based convolutional neural networks.

TABLE 3. Software and hardware specifications.

Configuration	Value
CPU	Intel core i5 8th Generation
GPU	Tesla K80 (2496 CUDA Cores)
Hard Disk	1TB
Operating System	Windows 10

named callback of 30 epochs was implemented. The resulting graph generated into a well fitted optimized curve. In [33], the authors trained the apple leaves disease data using DenseNet and EfficientNet which resulted in inconsistent accuracy and loss graphs. The overall time taken to compile the process is 25 minutes.

The custom PCA DeepNet generates the confusion matrix, which qualifies the classifier's performance and provides necessary data for calculating the performance parameters like precision, recall, F1-Score etc. The diagonal elements of the confusion matrix indicate the number of points for which the predicted label is equal to the true label. Fig. 11 represents the confusion matrix generated by the classifier which has high diagonal element values indicating a large number of correct predictions. The overall summary of the result gives a concrete inference that the classifier generated the best values for each class.

The other parameters like precision, recall and F1-measure are also generated and give a promising value indicating the novelty of the classifier in specifically classifying each and every class without getting confused with others. In [34]

where the authors trained their data with ResNet34, the results generated in terms of accuracy, F1-Score etc. were less as compared to the proposed AIS. The entire structure of the presented PCA DeepNet is enlisted in Table 4 for smooth understanding of the overall work. Brief analyses of the overall training and its resulting accuracies and performance metrics have been entitled in Table 5. The main objective is to get the result which includes Accuracy, Precision, Recall and F1-measure scores obtained while training the novel PCA DeepNet model. It can be seen that the proposed architecture detected each and every class of tomato leaf disease with higher accuracy.

The study is accomplished using Adam optimizer at a learning rate of 0.01; however, the application of different optimizers on the classifier is also performed in the experiment.

Table 6 highlights the results which are generated when PCA DeepNet is compiled using different optimizers respectively. The best results are listed in the table to set a comparison between different optimizers using different learning rates and predicting the outcomes efficiently. The work proves that mostly all the optimizers could perform well using the proposed PCA DeepNet classifier. The performance parameters include Accuracy, Precision, Recall and F1-Score. Thus it can be said that in terms of Optimization, the model is fully optimized to perform the training process.

In the presented work, further experiment is carried out using different pre-trained Deep Learning models. The transfer learning process is applied for each and every DL classifier using the Plant Village dataset. The various models are utilized based on several hyper-parameters to compare the efficiency of the presented customized classifier PCA DeepNet. Table 7 enlists the various ranges of the hyper-parameters values used in different architectural
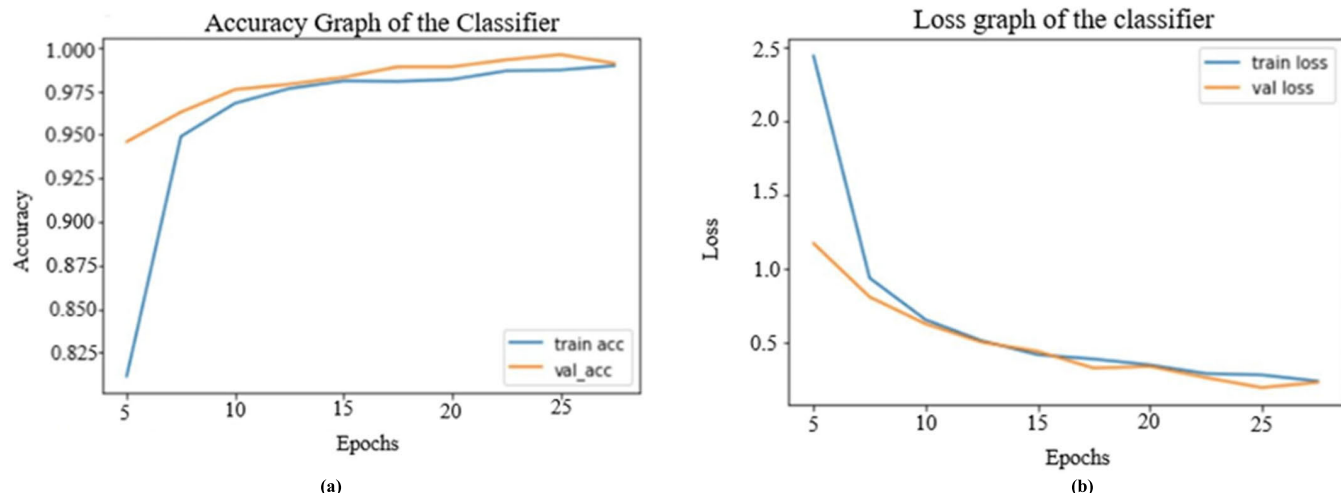
FIGURE 10. (a) Accuracy graph of the classifier (b) Loss graph of the classifier.

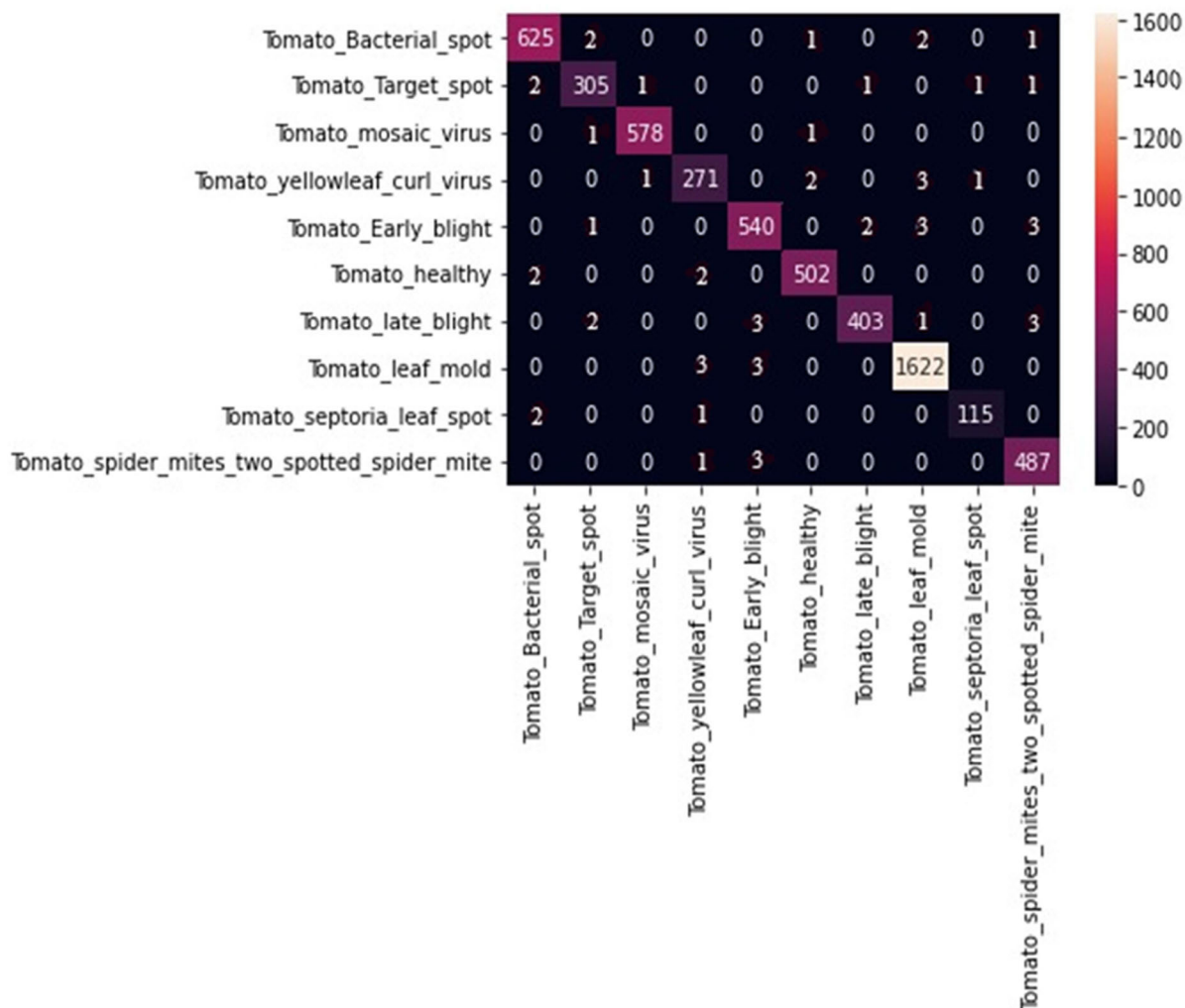**FIGURE 11.** Confusion matrix generated by PCA DeepNet.

models while compiling the same using Tomato leaf diseases dataset. Several parameters like the trainable, non-trainable layers, optimizers, learning rate, batch size, activation function, epochs and dropout values are altered. In the VGG16 model, first the base model is freezed and after training the same it is again unfreezed in order to reduce

TABLE 4. Detail structure of PCA deepnet.

Name	Kernel	Pooling Size	Number of Filters	Stride	Padding	Dropout	Activation
Input Image	-	-	-	-	-	-	-
Conv_1	5		32	3	Same	0.2	ReLU
Pool_1		2					
Conv_2	3		64	3	Same	0.25	ReLU
Pool_2		2					
Conv_3	3		128	3	Same	0.3	ReLU
Pool_3		2					
Conv_4	3		256	3	Same	0.3	ReLU
Pool_4		2					
Conv_5	3		512	3	Same	0.3	ReLU
Pool_5		2					
Up_6	2		256	2			
Conv_6	1		256	3	Same	0.3	ReLU
Up_7	2		128	2			
Conv_7	1		128	3	Same	0.3	ReLU
Up_8	2		64	2			
Conv_8	1		64	3	Same	0.3	ReLU
Up_9	2		32	2			
Conv_9	1		32	3	Same	0.3	ReLU
Conv_10		1		1			Sigmoid
Fully Connected layer							.
Total Training Parameters	7,759,521						
Trainable Parameters	7,759,521						
Non-Trainable Parameter	0						

the computational time of the model. The entire model takes 40 min to execute the results. The InceptionV3 model is trained altering several hyperparameters like hidden trainable layers, optimizers, learning rate, activation function, batch size, epochs and dropout values respectively. However in this model freezing of any layers is not done and the compilation is carried out using transfer learning technique. The training time consumption of the model is 35min. After successful completion of the process the corresponding results for the same are calculated.

The presented work also comprises of an exhaustive study of the different hyperparameters incorporated while training the dataset using InceptionResnetV2 architecture as shown in Table 7. Different parameters like Hidden trainable layers, nodes per trained layer, learning rate, momentum, optimizer, activation function, batch size, epochs and dropout values are

TABLE 5. Performance parameters obtained from the confusion matrix.

Leaf Classes	Precision (%)	Recall (%)	F1-measure (%)	Accuracy (%)
Tomato Bacterial Spot	99.04	99.04	99.04	99.20
Tomato Early Blight	98.07	98.07	98.07	98.07
Tomato Late Blight	99.48	99.82	99.65	99.65
Tomato Leaf Mold	97.48	97.48	97.48	97.48
Tomato Septoria Leaf Spot	98.36	98.36	98.36	99.36
Tomato Spider Mites	99.01	99.01	99.01	99.01
Tomato Target Spot	98.05	97.81	97.93	97.90
Tomato Yellow Leaf Curl Virus	99.51	99.51	99.51	99.51
Tomato Mosaic Virus	98.29	97.45	97.87	97.92
Tomato Healthy	98.78	98.78	98.78	98.78

TABLE 6. Performance parameters obtained using different optimizer on PCA deepnet model.

Optimizers	Learning Rate	Accuracy	Precision	Recall	F1-Score
SGD	0.01	96.90	94.32	94.30	94.31
RMSProp	0.001	94.40	93.32	93.20	93.26
Adadelta	0.001	92.31	92.30	92.30	92.30
Adagrad	0.0001	95.32	92.32	92.30	92.31
Adamax	0.001	97.32	96.90	96.92	96.91
Adam	0.01	99.60	98.55	98.49	98.52

changed. The top layer of this model was frozen so as to generate good results while compiling it. This helps in smooth training of the entire model. The time taken to complete the process is noted to be 32 min. Experimentation is also done using Resnet152V2 and the overall hyper-parameters used in training the model are listed in Table 7. The variation in the parameters like hidden trainable layers, optimizers, learning rate, momentum, activation function, batch size, epochs and dropout values resulted in valuable results for the same. The total training time for the DL architecture is 30 min and none of the layers are freezed in the architecture.

The entire approach highlights a detail analysis on how different deep learning models work on the present Tomato leaf diseases dataset. The different models used in the present experiment sets a good comparative report on the different parameters and how much time each and every model takes to generate the results. It is hereby observed that Resnet152V2 takes lesser time for computation as compared to other Deep Learning models used in the current work. Hence it can be concluded that different DL models perform differently when several parameters are altered, which is primarily due to the different model complexities that are mostly generated owing to the number of layers in it.

A detailed study has also been conducted where the Plant village dataset is utilized to validate other pre-trained Deep learning architectures. The resulting performances as well as the inference time are enumerated in Table 8 which indicates that the proposed customized classifier is better than any other state-of-the-art described and thus establishes the proof of its excellence. The paper also proposes a good utilization of the Plant Village dataset and the same is further used to train some existing Machine Learning Algorithms. Table 9

TABLE 7. Details of hyper-parameters used in different architectural models.

Hyperparameters	Experimental Value Range			
	VGG 16	Inception V3	Inception Resnet V2	Resnet 152V2
No. of Hidden trainable layers	1-3	1-3	0-3	0-3
Nodes per trained layer	64-2048	128-2048	64-2048	64-2048
Learning rate	0.01-0.00001	0.001-0.000001	0.001-0.000001	0.001-0.000001
Learning Rate Decay	Yes	-	-	-
Drop out	0.0-0.5	0.0-0.75	0.0-0.7	0.0-0.7
Batch size	64	64	64	64
Optimizer	Adam	Adam	Adam	Adam
Activation Function	Softmax	Softmax	Softmax	Softmax
Epochs	30-50	30-50	30-50	30-50
Base model	Freeze, Unfreeze	Unfreeze	-	-
Momentum	-	-	0.9	0.9

TABLE 8. Comparison of existing pre-trained DL classifiers with the proposed work using plant village dataset.

Pretrained CNN Classifier	Accuracy (%)	Precision (%)	Recall (%)	Inference Time (milli-second)
VGG 16	97.86	97.19	97.19	2.15
Inception V3	98.20	98.10	98.10	1.98
InceptionResnet V2	99.01	98.45	97.34	4.60
Resnet152V2	98.60	97.41	97.41	3.67
PCA DeepNet	**99.60**	**98.55**	**98.49**	**1.21**

TABLE 9. Hyper-parameters of the machine learning algorithms.

Machine Learning(ML) Algorithms	Hyper-parameters
K-Nearest Neighbor	Leaf_size = 5, p=1, n_neighbors = 7
Decision Tree	Max_depth = 200
Random Forest	n_estimators = 200, random_state = 5, max_depth = 200
Gradient Boosting	n_estimators = 200, random_state = 5, max_depth = 200

summarized the different values of hyper-parameters used in compiling the ML algorithms on Plant village dataset. The performance results in terms of Accuracy, Precision, Recall and F1-Score are mentioned in Table 10 while the graphical representation of the same is shown in Fig. 12. The present work is computed on Machine Learning algorithms
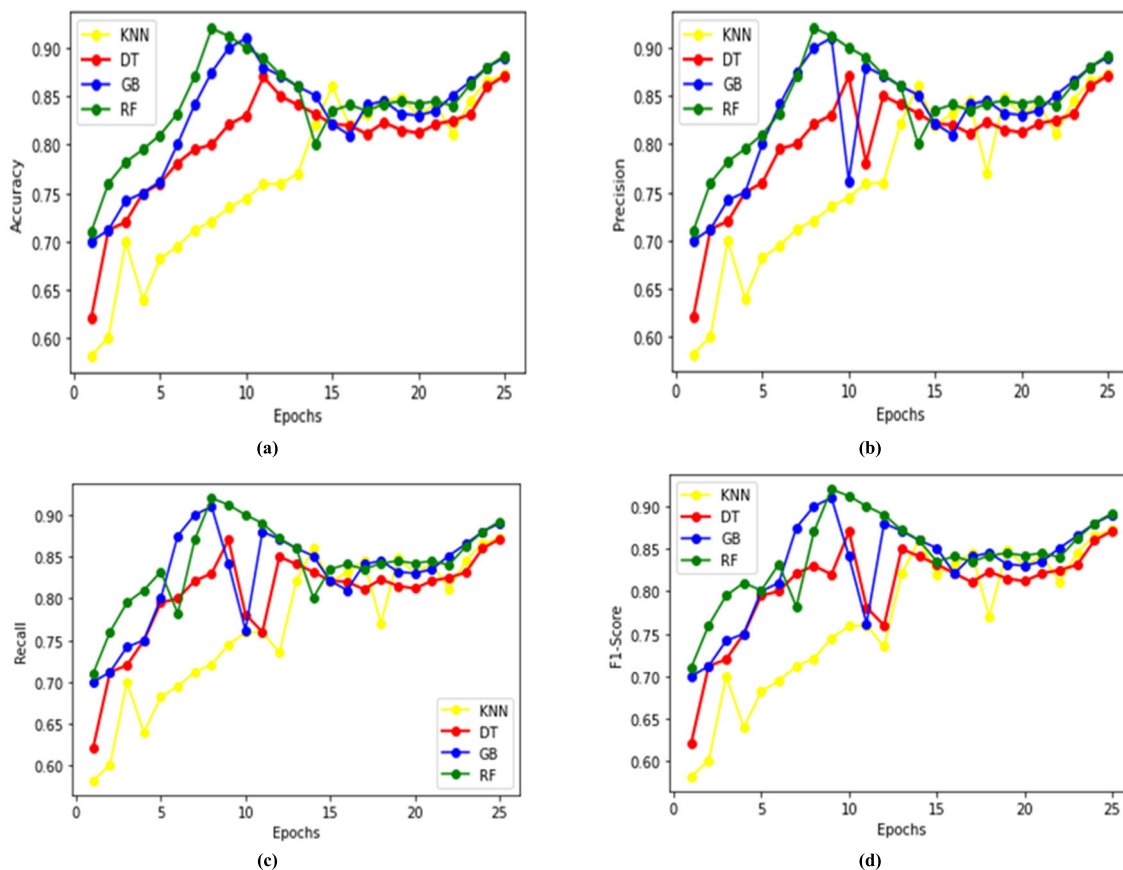
FIGURE 12. Performance evaluation of the different machine learning classifiers on the plant village dataset.

TABLE 10. Performance parameters of the existing ML algorithms using plant village dataset.

ML Algorithms	Accuracy	Precision	Recall	F1-Score
K-Nearest Neighbor (KNN)	0.767	0.767	0.767	0.767
Decision Tree (DT)	0.807	0.807	0.807	0.807
Gradient Boosting (GB)	0.830	0.830	0.830	0.830
Random Forest (RF)	0.842	0.842	0.842	0.842

as well as Deep Learning algorithms including the proposed PCA DeepNet architecture. From Table 8 and Table 11 it is clearly revealed that deep learning architecture generates promising results. The present scores give an insight that the proposed PCA DeepNet classifier is best among all the Machine Learning as well as Deep Learning classifiers. This clearly states that the proposed PCA DeepNet generates a sharp rise in the performance parameters which is more than 15% compared to Machine Learning algorithms and more than 5% compared to Deep Learning classifiers respectively. Table 11 represents a comparative analysis of different works based on the Plant Village Dataset. From the table it can be concluded that the proposed CNN architecture yields superior

results as compared to other works. Detection of diseases is a major purpose of this current work. Detection of plant diseases using Deep Learning based CNN has achieved great success in the Agricultural Industry for reducing the risk of infected diseases [35]. The authors in [36] for detecting Apple leaf diseases proposed CNN based model with GoogleNet Inception structure using Rainbow concatenation based on SSD Architecture, which generated detection performance of 78.80% mAP with a high detection speed of 23.13 FPS. The detection of the following 10 class Tomato leaf disease was formulated using F-RCNN. Some of the detected class images are shown in Fig. 13. The detection image consists of the individual annotated class names, bounded boxes and
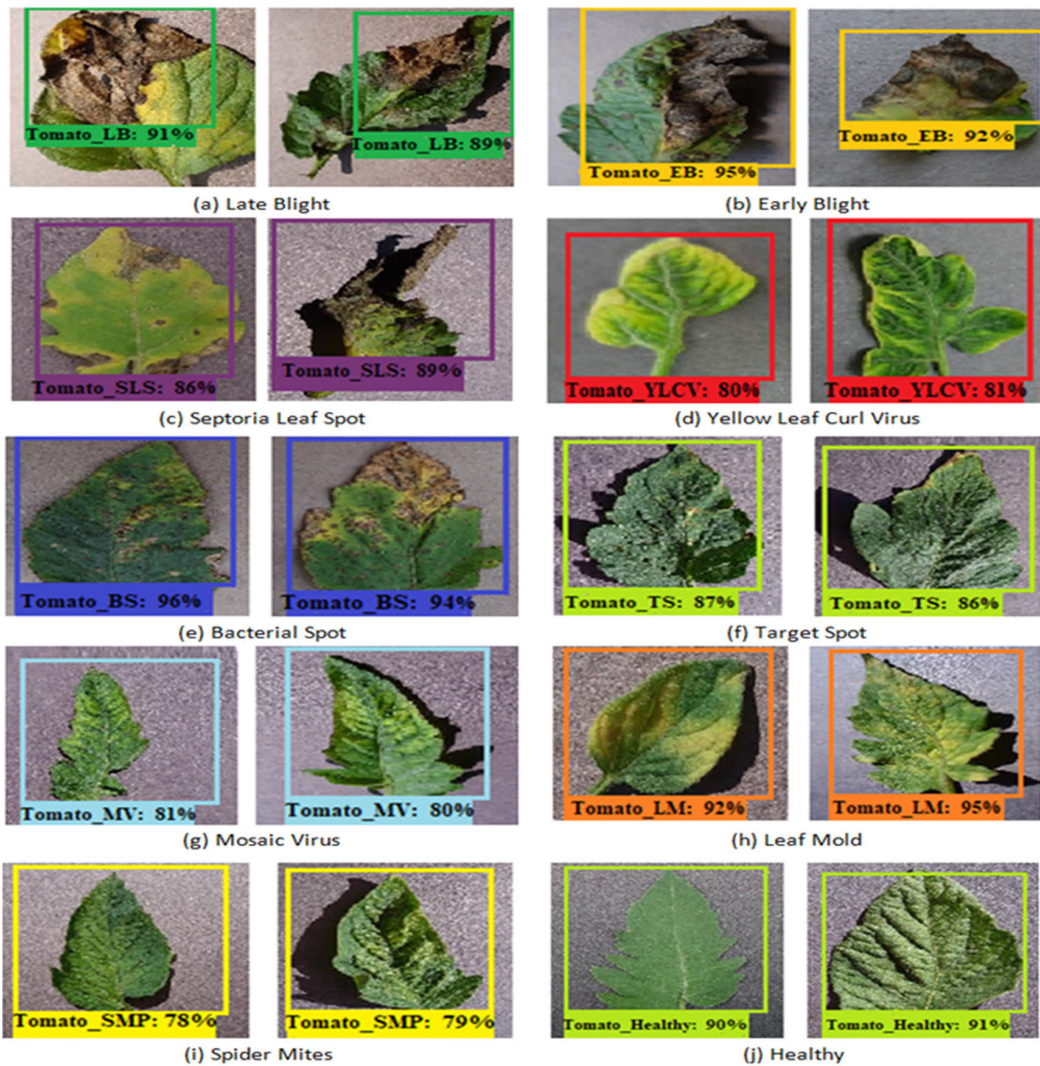
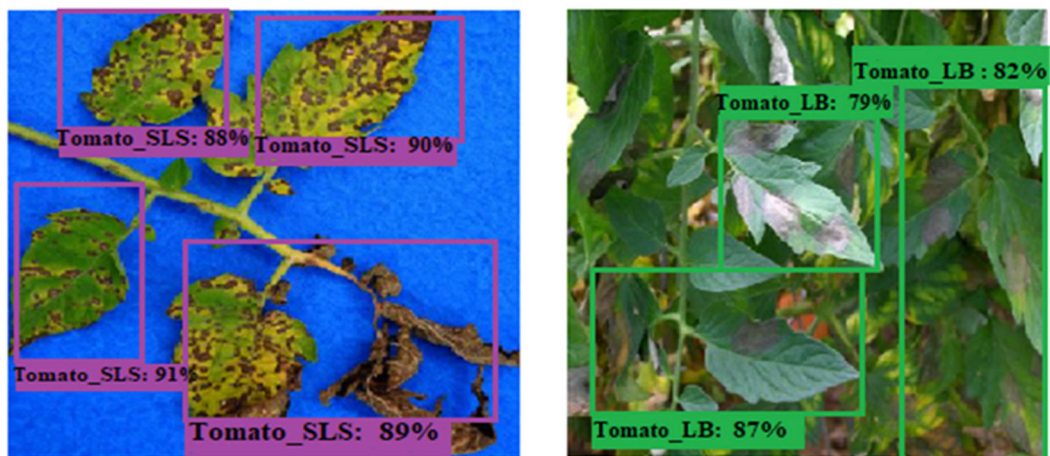**FIGURE 13.** Detected images of all the classes.

FIGURE 14. Detected image of single class in multiple position.

the detection scores. The detection scores and bounded boxes define how perfectly each class is detected. An intersection over union score of 0.95 is obtained while processing it.

The present work is also utilized in real time detection of tomato leaves to draw a major conclusion of the proposed work. In this paper we have used some more dataset and

TABLE 11. Comparison of the proposed framework with different other works on plant village dataset.

Different Techniques	Accuracy (%)	Precision (%)	F1-Measure (%)
MobileNetV2 [6]	99.30	-	-
CNN [10]	99.70	-	98.49
InceptionV3+DCGAN [22]	92.60	-	-
VGG-INCEP [13]	97.14	78.80	-
SSD-InceptionV2 [3]	-	73.07	-
ResNet-34 [29]	97.2	-	96.5
Proposed PCA DeepNet	**99.60**	**98.55**	**98.5%**

Fig. 14 shows some of the detected images of a single class in multiple positions to prove the major utility of the presented work.

The novelty of the pipeline is methodized using GANs, PCA, CNN and F-RCNN hence the name is suggested as PCA DeepNet which is used for the classification as well as detection of Tomato Leaf diseases. The entire framework is a hybridized structure of all the modules mentioned with appropriate and optimized customization. The classifier is designed to carry out a specific task of classifying each and every disease efficiently. The main challenge is to reduce the time complexity of the entire model so as to make it very reliable for proper classification process. The presented work proposes a framework where proper detection of Tomato Leaf Diseases is done. Hence, a detection process is introduced with required changes in different parameters of F-RCNN model for efficient detection of the diseases. The integration of these two helps in fulfilling the desired framework. The main objective of the current work is to classify the images based on different diseases and then detect the exact disease. This is methodized owing to the fact that if it is automated into a system then this can be very beneficial for the farmers and other people associated with agricultural work. The system will automatically analyze the different Tomato Leaf Disease images and can detect the diseases within a fraction of second. Thus this can serve a better purpose for the early detection of the Tomato Leaf diseases so as to prevent excessive loss that would have been otherwise occurred due to these diseases. This would thus strengthen agro-based industries for better production.

The presented work has huge utility in the agricultural domain; however, it possesses some limitations. The architecture of the proposed work consists of 10 CNN layers which can further be reduced to 5 – 6 layers in order to reduce computational time and complexity, still maintaining a higher value of Accuracy. Moreover, the proposed method is only limited to tomato leaf diseases detection whereas in real world, several other crops exist which require same types of pipeline for disease detection. Thus the same work can be extended for other crops as well. Also the current work is only a software based framework, while its hardware can be developed for accomplishment of the same specific task which will help to resolve the problem easily. Hardware interfacing for real time applications is encouraged to have a finished product for smart agriculture and growth of agro-based industries.

V. CONCLUSION

Agriculture has secured a great place in our day-to-day lifestyle and plays a significant role in the economic growth of the country. India, being an agricultural country, employs about 60% of its population in the agricultural sector. Thus, it is very crucial to have healthy and disease free crops. The implementation and deployment of modern methods using Deep learning and Computer vision in agriculture facilitates early detection of diseases which is very important for implementing remedial solutions and sustaining agro-based industries. The proposed work using PCA DeepNet makes disease detection very fast and accurate thereby eliminating the factor of human error. The major utility of the present work lies in helping farmers and other people engaged with agriculture to eradicate great losses incurred due to pest and other bacterial invasion on crops. Thus easy detection of such situations can enhance production to a greater extent. In near future this work can be automated into a real time system so that this can directly help farmers and others associated with agriculture. Moreover it can also be implemented as pest resistant equipment in large nurseries to detect early arrival of the diseases for its recovery. It is thus concluded that this novel framework creates a greater scope in enhancing its quality of analyzing and detecting different diseases in a better way than any other state-of-the-art.

REFERENCES

[1] Y. Wu, X. Feng, and G. Chen, "Plant leaf diseases fine-grained categorization using convolutional neural networks," *IEEE Access*, vol. 10, pp. 41087–41096, 2022.

[2] M. Adnan, K. Ali, G. Drushti, and C. Tejal, "Plant disease detection using CNN & remedy," *Int. J. Adv. Res. Electr., Electron. Instrum. Eng.*, vol. 8, no. 3, pp. 622–626, 2019.

[3] A. Abade, P. A. Ferreira, and F. de Barros Vidal, "Plant diseases recognition on images using convolutional neural networks: A systematic review," *Comput. Electron. Agricult.*, vol. 185, Jun. 2021, Art. no. 106125.

[4] 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.

[5] N. Ullah, J. A. Khan, L. A. Alharbi, A. Raza, W. Khan, and I. Ahmad, "An efficient approach for crops pests recognition and classification based on novel DeepPestNet deep learning model," *IEEE Access*, vol. 10, pp. 73019–73032, 2022.

[6] S. Ahmed, M. B. Hasan, T. Ahmed, M. R. K. Sony, and M. H. Kabir, "Less is more: Lighter and faster deep neural architecture for tomato leaf disease classification," *IEEE Access*, vol. 10, pp. 68868–68884, 2022.

[7] E. Elfatimi, R. Eryigit, and L. Elfatimi, "Beans leaf diseases classification using MobileNet models," *IEEE Access*, vol. 10, pp. 9471–9482, 2022.

[8] L. Aversano, M. L. Bernardi, M. Cimitile, M. Iammarino, and S. Rondinella, "Tomato diseases classification based on VGG and transfer learning," in *Proc. IEEE Int. Workshop Metrol. Gricult. Forestry (MetroAgriFor)*, Nov. 2020, pp. 129–133.

[9] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, "Deep neural networks based recognition of plant diseases by leaf image classification," *Comput. Intell. Neurosci.*, vol. 2016, pp. 1–11, May 2016.

[10] E. Ozbilge, M. K. Ulukok, O. Toygar, and E. Ozbilge, "Tomato disease recognition using a compact convolutional neural network," *IEEE Access*, vol. 10, pp. 77213–77224, 2022.

[11] H. Ajra, M. K. Nahar, L. Sarkar, and M. S. Islam, "Disease detection of plant leaf using image processing and CNN with preventive measures," in *Proc. Emerg. Technol. Comput., Commun. Electron. (ETCCE)*, Dec. 2020, pp. 1–6.

[12] M. Chohan, A. Khan, R. Chohan, S. Hassan, and M. Mahar, "Plant disease detection using deep learning," *Int. J. Recent Technol. Eng.*, vol. 9, no. 1, pp. 909–914, 2020.

[13] A. Rao and S. B. Kulkarni, "A hybrid approach for plant leaf disease detection and classification using digital image processing methods," *Int. J. Electr. Eng. Educ.*, Oct. 2020, Art. no. 002072092095312, doi: 10.1177/0020720920953126.

[14] M. K. Singh, S. Chetia, and M. Singh, "Detection and classification of plant leaf diseases in image processing using MATLAB," *Int. J. Life Sci. Res.*, vol. 5, no. 4, pp. 120–124, 2017.

[15] A. Patel and B. Joshi, "A survey on the plant leaf disease detection techniques," *Int. J. Adv. Res. Comput. Commun. Eng.*, vol. 6, no. 1, pp. 229–231, 2017.

[16] J. Gui, L. Hao, Q. Zhang, and X. Bao, "A new method for soybean leaf disease detection based on modified salient regions," *Int. J. Multimedia Ubiquitous Eng.*, vol. 10, no. 6, pp. 45–52, 2015.

[17] S. Pavithra, A. Priyadharshini, V. Praveena, and T. Monika, "Paddy leaf disease detection using SVM classifier," *Int. J. Commun. Comput. Technol.*, vol. 3, no. 1, pp. 16–20, 2015.

[18] Y. M. Oo and N. C. Htun, "Plant leaf disease detection and classification using image processing," *Int. J. Res. Eng.*, vol. 5, no. 9, pp. 516–523, Nov. 2018.

[19] D. Ashourloo, H. Aghighi, A. A. Matkan, M. R. Mobasheri, and A. M. Rad, "An investigation into machine learning regression techniques for the leaf rust disease detection using hyperspectral measurement," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 9, pp. 4344–4351, Sep. 2016.

[20] H. Nazki, S. Yoon, A. Fuentes, and D. S. Park, "Unsupervised image translation using adversarial networks for improved plant disease recognition," *Comput. Electron. Agricult.*, vol. 168, Jan. 2020, Art. no. 105117.

[21] Q. Zeng, X. Ma, B. Cheng, E. Zhou, and W. Pang, "GANs-based data augmentation for citrus disease severity detection using deep learning," *IEEE Access*, vol. 8, pp. 172882–172891, 2020.

[22] Y. Zhao, Z. Chen, X. Gao, W. Song, Q. Xiong, J. Hu, and Z. Zhang, "Plant disease detection using generated leaves based on DoubleGAN," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 19, no. 3, pp. 1817–1826, May/Jun. 2021.

[23] L. Li, S. Zhang, and B. Wang, "Plant disease detection and classification by deep learning—A review," *IEEE Access*, vol. 9, pp. 56683–56698, 2021.

[24] M. Brahimi, S. Mahmoudi, K. Boukhalfa, and A. Moussaoui, "Deep interpretable architecture for plant diseases classification," in *Proc. Signal Process., Algorithms, Architectures, Arrangements, Appl. (SPA)*, Sep. 2019, pp. 111–116.

[25] J. G. A. Barbedo, "Factors influencing the use of deep learning for plant disease recognition," *Biosyst. Eng.*, vol. 172, pp. 84–91, Aug. 2018.

[26] *PlantVillage Dataset*. Accessed: Aug. 2, 2022. [Online]. Available: https://www.kaggle.com/abdallahalidev/plantvillage-dataset

[27] F. Harrou, M. N. Nounou, H. N. Nounou, and M. Madakyaru, "Statistical fault detection using PCA-based GLR hypothesis testing," *J. Loss Prevention Process Ind.*, vol. 26, no. 1, pp. 129–139, 2013.

[28] A. M. Dawud, K. Yurtkan, and H. Oztoprak, "Application of deep learning in neuroradiology: Brain haemorrhage classification using transfer learning," *Comput. Intell. Neurosci.*, vol. 2019, pp. 1–12, Jun. 2019.

[29] F. Mohameth, C. Bingcai, and K. A. Sada, "Plant disease detection with deep learning and feature extraction using plant village," *J. Comput. Commun.*, vol. 8, no. 6, pp. 10–22, 2020.

[30] A. Rehman, S. Naz, M. I. Razzak, F. Akram, and M. Imran, "A deep learning-based framework for automatic brain tumors classification using transfer learning," *Circuits, Syst., Signal Process.*, vol. 39, no. 2, pp. 757–775, Feb. 2020.

[31] Y. H. Bhosale and K. Sridhar Patnaik, "IoT deployable lightweight deep learning application for COVID-19 detection with lung diseases using RaspberryPi," in *Proc. Int. Conf. IoT Blockchain Technol. (ICIBT)*, May 2022, pp. 1–6.

[32] A. Tripathy, A. Agrawal, and S. K. Rath, "Classification of sentiment reviews using n-gram machine learning approach," *Expert Syst. Appl.*, vol. 57, pp. 117–126, Sep. 2016.

[33] V. V. Srinidhi, A. Sahay, and K. Deeba, "Plant pathology disease detection in apple leaves using deep convolutional neural networks: Apple leaves disease detection using EfficientNet and DenseNet," in *Proc. 5th Int. Conf. Comput. Methodolog. Commun. (ICCMC)*, Apr. 2021, pp. 1119–1127.

[34] E. Harte, "Plant disease detection using CNN," ResearchGate, Sep. 2020, doi: 10.13140/RG.2.2.36485.99048.

[35] Y. Toda and F. Okura, "How convolutional neural networks diagnose plant disease," *Plant Phenomics*, vol. 2019, pp. 1–14, Mar. 2019.

[36] P. Jiang, Y. Chen, B. Liu, D. He, and C. Liang, "Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks," *IEEE Access*, vol. 7, pp. 59069–59080, 2019.

KYAMELIA ROY received the B.Tech. degree in electronics and communication engineering from MAKAUT, Kolkata, and the M.Tech. degree in intelligent automation and robotics from Jadavpur University, Kolkata, India, where she is currently pursuing the Ph.D. degree. She is also a Faculty Member with the Department of Electronics and Telecommunication Engineering, Siliguri Government Polytechnic, West Bengal, India. Her research interests include machine learning and image processing. She has published more than 20 papers in journals and international conferences.

SHELI SINHA CHAUDHURI (Member, IEEE) received the B.Tech. and M.E. degrees in electronics and telecommunication engineering and the Ph.D. degree in robotic vision from Jadavpur University, Kolkata, India. She is currently a Professor and the Ex-Head of the Department of Electronics and Telecommunication Engineering, Jadavpur University. Her research interests include image processing and robotic vision. She has published several research papers in reputed international journals and conferences and guided postgraduate and doctoral research in her field of interest.

JAROSLAV FRNDA (Senior Member, IEEE) was born in Slovakia, in 1989. He received the M.Sc. and Ph.D. degrees from the Department of Telecommunications, VSB—Technical University of Ostrava, Czechia, in 2013 and 2018, respectively. He is currently working as an Assistant Professor with the University of Zilina, Slovakia. He has authored or coauthored over 40 SCIE papers in WoS. His research interests include quality of multimedia services in IP networks, data analysis, and machine learning algorithms.

SRIJITA BANDOPADHYAY is currently pursuing the bachelor's degree in electronics and communication with the University of Engineering and Management, Kolkata, India. She is a keen research enthusiast in the field of deep learning and image processing. Her primary research interests include inclined towards medical image processing for disease detection and segmentation, computational neuroscience, and cognitive AI.

ISHAN JYOTI RAY is currently pursuing the bachelor's degree in electronics and communication engineering with the University of Engineering and Management, Kolkata, India. His current area of research interests include deep learning, especially neural networks. He has been working on applications, such as medical image processing and detection.

SOUMEN BANERJEE (Senior Member, IEEE) received the B.Sc. degree (Hons.) in physics from the University of Calcutta, in 1998, the B.Tech. and M.Tech. degrees in radio physics and electronics from the Institute of Radio Physics and Electronics, University of Calcutta, in 2001 and 2003, respectively, and the Ph.D. degree in engineering from the Indian Institute of Engineering Science and Technology (IIEST) Shibpur, India.

He is currently a Professor with the Department of Electronics and Communication Engineering, Budge Budge Institute of Technology, Kolkata, India. He was the Visiting Faculty Member with the Department of Applied Physics, University of Calcutta. He has a teaching and research experience of more than 20 years. His current research interests include design, fabrication, and characterization of wide band gap semiconductor-based IMPATT diodes at D band, W-band, and THz frequencies, SIW technology-based antennas, printed antennas and arrays, FSS, dielectric resonator antennas, body wearable antennas, machine learning, fuzzy systems, and evolutionary computation. He has published more than 100 contributory papers in journals and international conferences. He has authored ten books and five book chapters in fields of communication engineering, electromagnetic field theory, microwave, and antenna. He has edited three books published by Springer. His profile is included in IBC, Cambridge, England, and Marquis Who's Who in the World, USA.
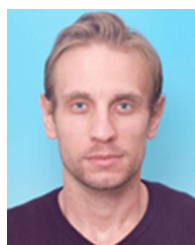
Dr. Banerjee is a Senior Member of IEEE AP Society. He is also a fellow of IETE, New Delhi, India. He is a Reviewer of several international journals, such as IEEE Transactions on Emerging Topics in Computational Intelligence, IEEE Transactions on Electron Devices, IEEE Access, IEEE Sensors Letters, *Microwave and Optical Technology Letters* (MOTL) (Wiley), *Journal of Electromagnetic Waves and Applications* (Taylor & Francis), *Radioengineering Journal* (Czech and Slovak Technical University), *Journal of Computational Electronics* (Springer-Nature), *Journal of Renewable and Sustainable Energy* (American Institute of Physics-AIP), *Cluster Computing* (Springer-Nature), and *Journal of Infrared, Millimeter and Terahertz Waves* (Springer-Nature). He acted as a Convener in international conferences, such as OPTRONIX-2019 (IEEE) and OPTRONIX-2020 (Springer), both held at Kolkata, India, and also chaired many technical sessions in several international conferences.

JAN NEDOMA (Senior Member, IEEE) is currently an Associate Professor and the Head of the Optoelectronics Laboratory, Department of Telecommunications, Faculty of Electrical Engineering and Computer Science, Technical University of Ostrava. He is also a member of the scientific council, a member of doctoral, habilitation, and professors' committees, and a guarantor of bachelor's study programs at the mentioned faculty. During his scientific career, he was the leader or a co-investigator of more than 25 projects and has more than 175 journal articles and conference papers in his research areas. He holds ten valid Czech patents. His research interests include optical communications and optical wireless communications, optoelectronics, optical measurements, measurements in telecommunication technology, signal processing, fiber-optic sensors, and biomedical engineering.

• • •