Indonesian Journal of Electrical Engineering and Informatics (IJEEI)
Vol. 12, No. 4, December 2024, pp. 802~817
ISSN: 2089-3272, DOI: 10.52549/ijeei.v12i4.5455

A Twelve-layer Deep Convolution Neural Network for Fast, Efficient and Reliable Identification and Classification of Plant Diseases in Smart Farming

Hafsa A. Jassim[1], Zahraa K. Taha[2], Abbas Khalifa Nawar [3]
[1]Department of Electrical Engineering, University of Baghdad, Baghdad, Iraq
[2]Department of Network Engineering, Al-Iraqia University, Baghdad, Iraq
[3]Department of Computer Techniques Engineering, Imam Alkadhim University College, Baghdad, Iraq

Article Info

Article history:

Received Feb 20, 2024
Revised Sep 29, 2024
Accepted Oct 8, 2024

Keywords:

Smart farming
Plant leaf diseases
Deep learning
Deep convolutional neural networks
Image augmentation
Transfer learning

ABSTRACT

Smart farming that uses information and communication technology is developed as a critical technique to address the challenges related to agricultural production, environmental effects (climate change), food security, and supply chain. The recent statistics reveal that the world's population has been increased significantly, which is expected to reach 7.7 billion. It is essential to achieve a significant rise in food output to meet the requirement of such a massive growth of population. However, due to the natural conditions and a variety of plant illnesses, food productivity and farms are reduced. In order to diagnose food diseases in farming, new technologies like the Internet of Things and artificial intelligence are now essential. To this end, the research paper introduces a novel artificial intelligence model represented by a twelve-layer deep convolution neural network to identify and classify plant image diseases. 38 distinct types of plant leaf photos are used for training and testing the suggested model, which are obtained by adjusting different parameters such as (a) hyperparameters; (b) coevolutionary layers; (c) and pooling layers in number. The proposed model consists of an extractor and classifier of functions. The first section involves three phases, i.e., it consists of two convolution layers and a maximum pooling layer for each phase. The second section consists of three levels: flattening, hidden, and output layers. The proposed model is compared with LeNet, VGG16, AlexNet, and Inception v3, which are considered state-of-the-art pre-trained models. The results demonstrate that the accuracy of LeNet, VGG16, AlexNet, and Inception v3 is given as 89%, 93%, 96.11%, and 97.6%, respectively. The findings provided in this research show that the suggested model outperforms state-of-the-art models in terms of training speed and computing time. Also, the results show that the proposed model achieves a considerable improvement in terms of accuracy and the mean square error compared to the state-of-the-art methods. In particular, The outcomes demonstrate that the suggested model achieves a mean square error and prediction accuracy of 98.76% and 0.0580, respectively. The results also depict that the proposed model is more reliable, allows fast convergence time in obtaining the results, and requires only a small number of trained parameters to identify the plant diseases accurately.

Corresponding Author:

Abbas Khalifa Nawar,
Department of Computer Techniques Engineering,
Imam Alkadhim University College, Baghdad, Iraq.
Email: abbas.altimimy@iku.edu.iq

1. INTRODUCTION

Farm management requires effective plant leaf disease diagnosis and detection. This is due to the significant roles these identification and detection play in plant leaf diseases in agricultural and agronomical. However, to now it is thought to be a substantial challenge to accurately identify and detect plant leaf diseases.

Agriculture is impacted by a number of plant diseases, which have an impact on the productivity and quality of the plants.. Most diseases are assessed based on their symptoms, which can be detected by the naked eyes of a specialist in this field. However, due to the increased cost and dearth of plants professionals, this detection technique may prove to be very difficult in practice [1]. In addition, failing to detect the diseases that harm the plants may cause insufficient food production for humans, and in worse circumstances, could lead to malnutrition, and thus, death. For instance, the late potato blight disease, brought on by Phytophthora infestans, has destroyed Ireland's primary crop of potatoes between 1845 and 1850. This resulted in widespread starvation, which caused about a million deaths and caused millions of people to emigrate to countries like Canada, the United States, and others [2].

Leaf symptoms in many different types of plants are considered an essential source of disease knowledge that could be detected. However, the productivity and quality of food are affected by the attack of various diseases such as Grapevine measles, Bacterial spot, rust, and late blight. The development of automated systems has played an essential role in detecting and identifying plant diseases at an early stage. Early identification of these plant diseases could help to increase the plant's production efficiency. In order to address this problem and reduce the shortage of plant experts, computerized identification of plant diseases by image analysis is presented [3]. The contributions of paper are

1. The primary contribution of this work is the development of the structure of a CNN by employing a combination of different convolution layers with filters of sizes 64, 128, and 256, along with max pooling of 2x2, to extract relevant features from plant diseases. The CNN layers capture basic patterns at three levels: low-level features, mid-level features, and high-level features, to build a reliable and accurate framework for classifying plant diseases.
2. We conduct numerous experiments using extensive real-world datasets, including 38 imbalanced disease classes. This demonstrates the model's scalability and reliability in accessing the diversity of plant disease datasets. The proposed framework is compared with other frameworks and shows that it has better performance than the state-of-the-art technology.
3. By combining a specific number of convolution layers with max pooling, an effective model can reduce the spatial dimension of data, which results in reduced computational complexity, allowing the model to process massive amounts of data more efficiently. The strategic use of these structures ensures only the most important features are extracted and no irrelevant features, resulting in faster training times without compromising accuracy.

2. RELATED WORK

So much research on isolated crops has been performed, such as tomatoes [4], potatoes [5], maize [6], apple [7] and pepper [8]. More recently, deep learning techniques, particularly the coevolution neural network (CNN), have rapidly become the preferred method for facing the challenges associated with detecting and identifying plant diseases. For both big and small-scale problems, CNN is the popular generality algorithm for computer vision. CNN has demonstrated excellent image processing and classification capabilities [9]. For instance, it has been used in [10] for training to recognize diseases in rice and maize images. The results showed that using deep transfer learning accuracy of 92.00% in recognizing the correct diseases can be achieved [10]. CNN has also been used in [11] to classify ten widespread rice illnesses via image analysis. The results showed that CNN could achieve 95% accuracy in diseases identification. In [12], CNN is utilized to classify tomato leaf diseases. The results demonstrated that CNN achieves 91.20% accuracy in identifying the tomato leaf diseases [12]. While several research works have been proposed in the literature and showed promising results related to identifying the diseases, these studies focus on a limited number of diseases. Thus, these are not sufficient. In particular, the most popular 38 pairs of crop diseases, as shown in Fig. 1, need to be considered. Therefore, there is essential to propose algorithms that can identify and classify different diseases more accurately. These algorithms could use pictures only to identify different diseases.

Figure 1. Illustration of 38 classes of crop diseases and pairs of plant diseases.

The goal of this work is to use a Deep CNN-based model to address the difficulties of plant leaf disease recognition and classification. Deep CNN is a type of algorithm for deep learning. Deep CNNs have numerous applications in image analysis, object recognition, speech recognition, suggestion systems, and natural language processing. Reduced training data quantity, training time, and computing costs are achieved by information sharing known as transfer learning [13]. A pre-trained model's learning can be transferred to a new model with the use of transfer learning. LeNet [14] is a design of the convolutional neural network proposed in 1998 by Yann LeCun et al. LeNet corresponds to Lenet-5 in general and is a primary convolutionary neural network. VGG 16 [12] is the CNN architecture built for the ILSVRC-2014 (ImageNet Large Scale Visual Recognition Challenge) by VGG (Visual Geometry Group, University of Oxford). VGG16 is a 16-layer network containing 13 convolutional layers, five max-pooling layers, and three dense layers. Alex-Net was designed by 2012 Image-Net competition winner Hinton and his student Alex Krizhevsky. The network has five convolution layers, followed by the highest pooling layer, and finally, three total connection layers. Inception V3 [15] is a deep learning network with 42 layers. Throughout this study, the success of the proposed Deep CNN model was contrasted with some of the standard networks such as LeNet, VGG16, AlexNet and Inception v3.

The twelve-layer deep convolution neural network model is presented to enhance plant diseases' identification and classification performance. In particular, the suggested approach seeks to increase effectiveness in comparison to contemporary techniques for detecting and classifying plant diseases, which are shown to provide long convergence times and require huge groups of design parameters are needed. Unlike the most recent models, our suggested model has quick convergence times and only needs a small number of training parameters.

The plant diseases dataset, which consists of 87867 images, commonly 38 different types of leaf images, is used to train and test the suggested model. We have evaluated different batches sizes, dropout rate, number of convolutions layer, pooling layer, and epochs. In contrast to pre-train models, the proposed model achieves 98.76% classification accuracy with less trainable parameters.This accuracy improvement is superior in comparison to other traditional machine learning approaches.

This study aims to help the farmer identify and classify plant diseases at an early stage. We aim to provide a practical, flexible, feasible and time-saving approach for achieving efficient plant diseases identification and classification. Intense learning, a type of machine learning, is suggested as a means of achieving this goal by providing a flexible and reliable decision-making structure, without the need for feature engineering and labelling of data and also to achieve best outcomes with unstructured data. Table 1 presents some of the previous works on plant leaf diseases:

Table 1. Previous works on plant leaf diseases.

Reference	**Publication & Year**	**Aims and Purpose**	**Method/Technique**	**Result and performance metrics**	**Advantages**	**Case study/size of data**
[4]	2016, IEEE	Classify the leaves of tomato plants into six groups to efficiently identify the diseases	Classification tree	Accuracy=97.3%	Provides a helpful identification of plant diseases, image recognition and classification	Tomato Plant Disease/ 383 digital images
[5]	2017, IEEE	Diagnosing diseases from leaf images	Image processing and SVM	Accuracy = 95% Recall=95% Precision=95% F1=95% ROC=96%	Provides the farmers with a reliable, effective, and fast way to identify diseases	Potato Diseases/ 300 images of potato leaves
[3]	2017, Frontiers in Plant Science	(missing)	Deep convolutional neural networks	Accuracy = 93%	Provides digital plant diseases detection that is fast, inexpensive and easy to be deployed	Cassava / 2,756 images
[11]	2017, Science direct	Classify the ten common diseases of rice	Deep convolutional neural networks	Accuracy rate = 95% report of Missing rates =5 report of False rate =0	Provides a higher convergence time and a better ability to recognize diseases	Rice diseases /500 images

Reference	Publication & Year	Aims and Purpose	Method/Technique	Result and performance metrics	Advantages	Case study/size of data
[6]	2018, IEEE	Identification of 9 kinds of corn leaves	GoogLeNet model Cifar10	Accuracy rate =98.9%	Enhances the accuracy of identifying maize leaf diseases and decreases convergence iterations	Maize leaf diseases/ 500 images
[16]	2018, IEEE	Identification of anomalies that are happened in plants, particularly in their greenhouses or natural environment	Random Forest	Accuracy=70.14%	Offers a straightforward recognition of the plant's diseases	Dataset creation
[17]	2018, Science direct	Classifying 6 different tomato diseases	AlexNet and VGG16 net	Accuracy= 97.29%	(Missing)	6 diseases Tomato / 13,262 images
[7]	2019, IEEE	Detection of apple leaf diseases in real-time	GoogLeNet Inception structure and Rainbow concatenation	Confusion Matrix Accuracy=78.80% speed =23.13 FPS	Provides a highly accurate real-time identification of the five different forms of apple leaf diseases	Apple leaf disease/ 26,377 images
[18]	2019, Scientific Reports	Diseases classification, particularly a rice blast disease data package	CNN SVM	Accuracy=95.83% AUC=0.99	Helps the plant health experts in a rice blast disease	Rice disease/ 5808 images
[8]	2019, Science direct	Identify four distinct classes of pepper plants	Artificial Neural Networks (ANN), Naive Bayes (NB) and K-nearest Neighbor (KNN)	Accuracy=100%	(missing)	Diseases related to peppers/ (missing)
[1]	2019, Science direct	Recognition of apple diseases	Genetic algorithm and classified by One-vs-All M-SVM	Accuracy= 96.98% Time=0.6511sec	Offers greater efficiency in terms of precision and time execution	Apple diseases classes/ 300 images
[19]	2019, Science direct	Considering the problems of plant leaf disease detection	Deep convolutional neural network	Accuracy=96.46% Precision=96.47% Recall=99.8% F1=98.15	Provides a superior ability to forecast and improves both the accuracy and reliability	39 Plant disease/ 55636 images
[12]	2020, Science direct	Diagnose the disease of leaves in tomato crops	Convolution Neural Network	Accuracy=91.2%	It helps the farmer to address the plant diseases fastly and increase both the quality and quantity of foods	Tomato disease/17500 images
[20]	2020, Science direct	Classification of rice diseases	Deep Neural Network with Jaya Optimization Algorithm	Accuracy F1-score Precision	(missing)	Rice plant/ 650 images
[21]	2020, IEEE	Detect and classify diseases of tomatoes,	Deep convolution neural network	Accuracy=98.029% Confusion matrix	Provides accurate and fast detection and classification	Tomatoes, pepper, and potatoes diseases/ 20636

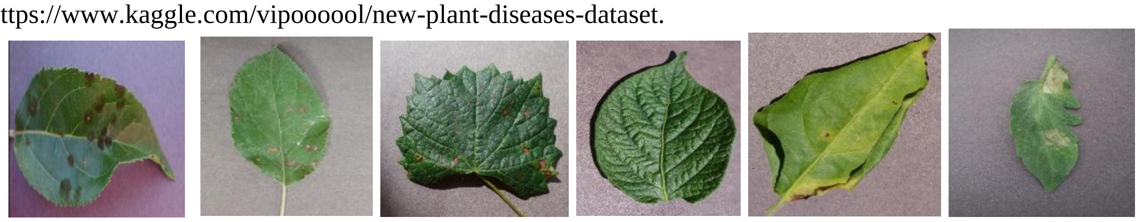
Reference	Publication & Year	Aims and Purpose	Method/Technique	Result and performance metrics	Advantages	Case study/size of data
		pepper, and potatoes				
[22]	2020, Science direct	Plant disease detection and diagnosis	VGG16 and VGG19	Accuracy=98.2% F1-score =97% Precision =98% Recall=97%	Helps to recognize infected and healthy maize leaves correctly and effectively	Maize plant/ 15408 images
[23]	2021, Science direct	It is inevitable to use effective models with fewer parameters that can be learned more quickly without sacrificing performance.	AlexNet VGG16 net ResNet Inception V3			original /55448 augmented /61486
[24]	2021, Elsevier		VirLeafNet-3	97.403%		3031 images
[25]	2021, Frontiers in Plant Science		GoogLeNet VGG16 Resnet50	98.06% original 99.60% feature datasets,		5 types of disease 792 origin 1306 images

A study in [19] has been selected to be a benchmark study for comparison. This study aligns with our scope of improving the identification and classification of plant diseases, which signifies its use in the comparison.

3. IMPLEMENTATATION OF THE PROPOSED

3.1. Data acquisition and augmentation

The dataset utilized in this paper was downloaded from Kaggle. This dataset comprises of about 1GB of RGB photos that have been divided into 38 groups representing both healthy and diseased crop leaves. A dataset with 87867 images is used for the proposed deep CNN model's training and testing. It is partitioned into 80/20 training to test a set ratio that preserves the structure of the directory. In Fig. 2, representative photographs of random groups are introduced. It is a public dataset and can be found at the following web link: https://www.kaggle.com/vipoooool/new-plant-diseases-dataset.



(a) (b) (c) (d) (e) (f)

Figure 2. (a) Apple with scab (b) Apple with cedar apple rust (c) Grape with black rot (d) Potato healthy (e) Pepper bell healthy (f) Tomato with late blight.

Data augmentation is the process of incorporating slightly altered versions of already existing data or brand-new synthetic data created from already existing data. This is done to expand the volume of data, Additionally, it is used to rotate, scale, resize, and flip the photos. Fig. 3 shows the augmentation process applied in this paper to the sample images and used in the data training. Table 2 shows the total number of samples in the training and testing of the dataset, as well as the number of samples in each group. Fig. 4 illustrates the distribution of samples to each class.

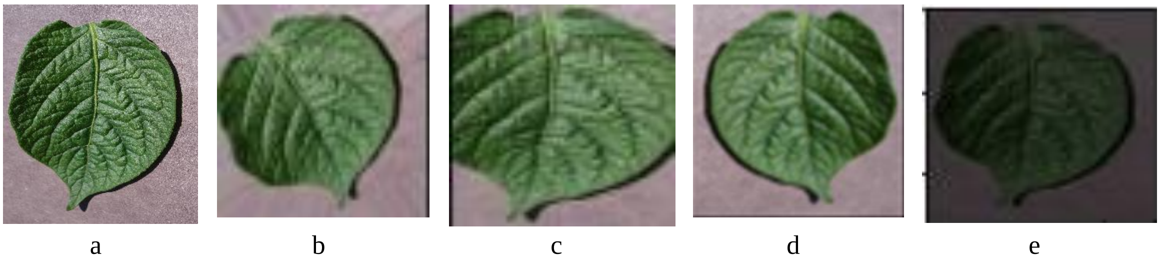


Figure 3. (a) original image (b) to (e) are augmented image.

Table 2. Number of samples in each class.

No.	Class Name	Size of the test set	Size of the train set
	Apple with scab	504	2016
	Apple with black rot	497	1987
	Apple with cedar apple rust	440	1760
	Healthy apple	502	2008
	Blueberry with healthy	454	1816
	Cherry (including sour) powdery mildew	421	1683
	Cherry with powdery mildew	456	1826
	Corn gray leaf spot	410	1642
	Corn (maize) common rust_	477	1907
	Corn (maize) northern leaf blight	477	1908
	Corn (maize) healthy	465	1859
	Grape black rot	472	1888
	Grape esca (black Measles)	480	1920
	Grape leaf blight	430	1722
	Grape healthy	423	1692
	Orange Huanglongbing (citrus greening)	503	2010
	Peach bacterial spot	459	1838
	Peach healthy	432	1728
	Pepper, bell bacterial spot	478	1913
	Pepper bell healthy	497	1988
	Potato early blight	485	1939
	Potato late blight	485	1939
	Potato healthy	456	1824
	Raspberry healthy	445	1781
	Soybean healthy	505	2022
	Squash powdery mildew	434	1736
	Strawberry leaf scorch	444	1774
	Strawberry healthy	456	1824
	Tomato bacterial spot	425	1702
	Tomato early blight	480	1920
	Tomato late blight	463	1851
	Tomato leaf mold	470	1882
	Tomato Septoria leaf spot	436	1745
	Tomato spider mites two-spotted spider mite	435	1741
	Tomato target spot	457	1827
	Tomato yellow leaf curl virus	490	1961
	Tomato mosaic virus	448	1790
	Tomato healthy	481	1926
	Average	17572	70295

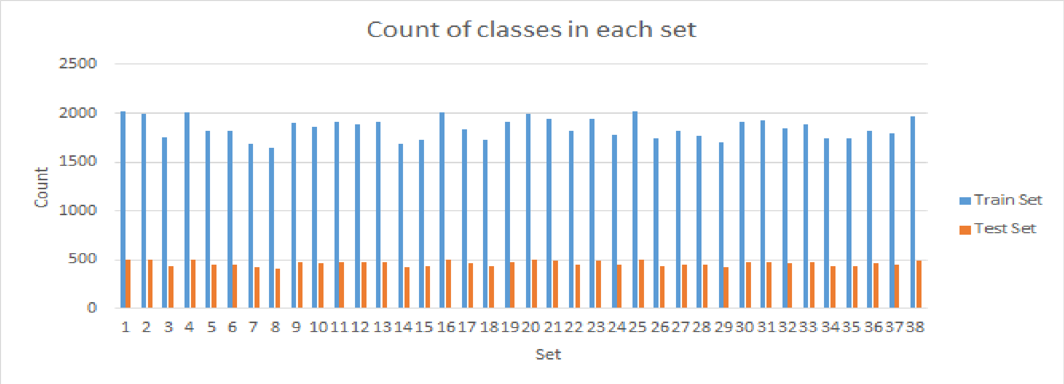


Figure 4. Counts of classes in the train set and test set.

3.2. Model Architecture and Rationale

A deep CNN with twelve layers was proposed to classify plant diseases due to the complex and high-dimensional nature of these images. A special structure was chosen for the CNN because of its ability to learn and automatically extract features from images without the need for a specific method of feature extraction. It provides a satisfactory balance between extracting and learning complex patterns while maintaining computational efficiency. The proposed framework uses convolutional layers with 64, 128, and 256 filters to extract features at different levels (low, medium, and high). This is followed by max pooling to reduce the feature map while retaining important features. The sequence of the proposed method not only reduces computational complexity but also helps prevent overfitting.

3.3. Training Setup and Hyperparameter Tuning

In the training phase, the Adam optimizer is proposed to carry out the training procedure. The learning rate is set to 0.01 to ensure faster learning and convergence compared to the gradient descent optimizer. The batch size was selected as 16 to balance the speed of training and memory usage. The dropout layer is used to reduce input to another layer: 0.5 for the internal layer and 0.1 for the external layer. This technique avoids overfitting and, at the same time, ensures the model learns more robustly from features. Twelve epochs are selected according to the convergence and performance rate of the test data. This factor must be chosen carefully to prevent insufficient learning of the model with fewer epochs (underfitting) or excessive learning, where the model memorizes all details of the training data (noise and specific patterns) but does not generalize with many epochs (overfitting).

4. EVALUATIONS PHASES

The efficiency of the introduced model was measured using five measurement metrics of classification problems, which are defined as precision, accuracy, recall, F1-score and confusion matrix. These measurement metrics are commonly used in the open literature of classification problems.

The number of accurate predictions that a model can make about the overall forecasts that have been occurred, is known as the accuracy. Accordingly, accuracy can be described in eq. (4) as

$$\text{Accuracy} = \frac{\text{TP+FN}}{\text{TP+FP+FN+FP}}. \tag{4}$$

Moreover, a precision definition is the ratio of the number of actual positive outcomes (TP) to the number of positive outcomes (TP + FP) that the model predicted. The precision ranges from 0 to 1 and can be determined as in eq. (5).

$$\text{precision} = \frac{\text{TP}}{\text{TP+FP}} \tag{5}$$

To determine the percentage of accurate identifications, precision is utilized. Additionally, the number of true positives is divided by the number of all related sample data (TP + FN) of the recall.

The recall is used to compute the percentage of true positives that have been properly identified according to eq. (6). To compute the recall, the following equation is used,

$$\text{recall} = \frac{\text{2TP}}{\text{2TP+FP+FN}} \tag{6}$$

Since the harmonic lies between accuracy and recall, the F1 score is established. The F1 score represents the number of instances that the learning systems can identify correctly. The F1 score ranges between 0 and 1 and can be determined using the following equation,

$$\text{recall} = \frac{\text{2TP}}{\text{TP+FN}} \tag{7}$$

The confusion matrix is a table that is usually familiar to reveal the performance of a classification model on the test set over which the correct values are acknowledged by mapping the predicted outputs across actual outputs [22].

5. RESULTS AND DISCUSSION

5.1. EXPERIMENTAL SETUP

The twelve-layer deep CNN model and the other transfer learning models are implemented using the python-based scikit-learn, Keras, and OpenCV libraries. In Kaggle, a graphics processing unit (GPU) is used to build and simulate the model of the suggested solution in order to speed up the simulation process.

5.2. TRAINING THE TWELVE-LAYER DEEP CONVOLUTION NEURAL NETWORK

Machine learning is used in Deep CNN. CNN is employed in a number of farming picture classification tasks. One of the key advantages of utilizing deep CNN for image classification is that it lessens the need for the feature engineering procedure. The suggested CNN trains the picture of plant leaf diseases using a 12-layer deep CNN. Identification and classification processes utilise the datasets that were covered in the preceding section. Figure 5 introduces the architecture of deep CNN, which comprises several layers.

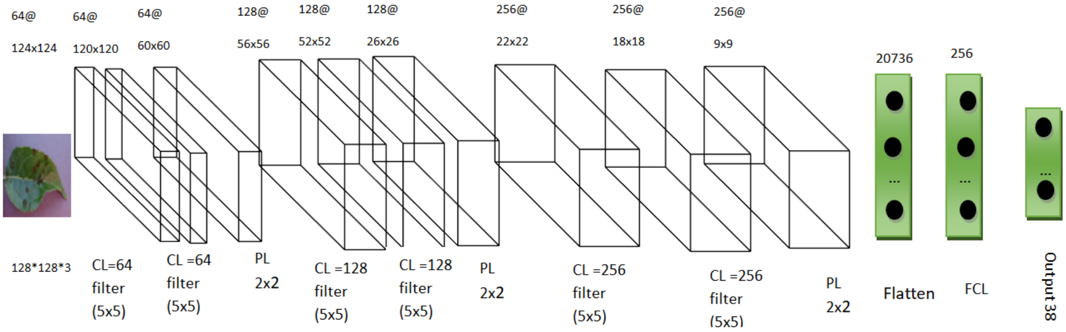


Figure 5. Twelve-layer deep CNN to identify and classify of 38 plant diseases.

Convolutional Layer (CL): The structure of deep CNN consists of several blocks. This structure reflects the extractor of features. The feature extractor pulls distinguishing traits from the raw images using layers of feature maps. As a local feature extractor, convolutional filtering is used to analyze the connections between several raw image pixels. As such, the high-level features, which are considered efficient and suitable, can be extracted to improve the CNN model's capacity for generalization. [18] [26]. The visual representation of the image was provided in Figure 6 when a different number of convolution layers were used.

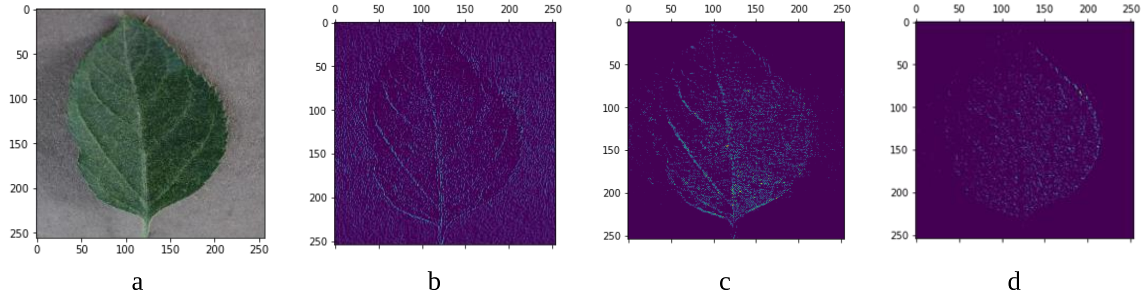


Figure 6. (a) Input image, (b) convolutional layer –1, (c) convolutional layer –6 and (d) convolutional layer –15

In order to generate feature maps, a convolutional layer is used, and this is done by first applying convolution kernels to the input data and then an activation function as shown in eq. (1), (2),

$$y_j^l = f(z_j^l) \tag{1}$$

$$z_j^l = \sum_{M_j} x_i^{l-1} * k_{ij}^l + b_j^l \tag{2}$$

where, y_j^l is the output feature maps at layer l; f () is the activation function (commonly used functions include sigmoid, tanh, and ReLU, of which ReLU was chosen); $z_j^l$is the activation of the j channel at layer l; x_i^{l-1} feature maps of the $l-1$ layer; M_j is the subset of input feature maps; k_{ij}^l is convolution kernel matrix at layer l; * is the convolution operation; and finally b_j^l is the offset [18].

As shown in Fig. 5, applying a batch normalization on the output function maps after each CL is desirable. This is because batch normalization can be used to overcome uncertainty, speed up learning, and maximize classification outcomes [27]. Another essential layer that can also be used is a dropout, which applies to eliminating input from the network. The suggested dropout rate values for the external layer are 0.1 and for the internal layer [28] are 0.5 to 0.8. Therefore, we set the dropout value in this paper to be 0.1 for the external layer and 0.5 for the inner layers.

Pooling Layer (PL): The down-sampling process is done by the pooling filters to decrease the data dimension of the y_j^l input function maps. The max-pooling is achieved by selecting the maximum element of a 2 x 2 region of the input feature map with a step of 2 by adding a 2 x 2 pooling window. As a result, the spatial scale of the feature maps is vigorously reduced, and the extracted feature is condensed with details. In addition, the pooling layers add less data redundancy to CNN, and thus, could produce less overhead for data analysis [29].

Completely Connected Layer (FCL): Black circle in Fig. 5 represents one neuron in the FCL, which is connected to all previous maps of the input elements. Based on the features extracted by the CLs, the FCL carries out a comprehensive feature evaluation and generates N-dimensional probability vectors, where N is the number of targets of the classification. A SoftMax function is used after the final layer of FCL to produce the final likelihood of classification as described in eq. 3,

$$P_i = \frac{e^{a_i}}{\sum_{N=1}^{38} e^{a_n}} \ , \tag{3}$$

where a_i is ith neuron output in the final FCL layer. This function normalizes the final FCL layer output to a vector of values between zero and one, which gives a probability over all 38 classes.

The deep CNN converts the input image layer by layer from the initial pixel values to the final likelihood vector p, which is achieved by adding those mentioned above hierarchically organized layers, in which the largest P_i represents the most regular class [27] [29]. Deep CNN is a frequent technique, and it can train a number of models to find the correct function. Gradient descent, often known as batch gradient descent, is a fundamental optimization technique used to carry out the gradient steps using all training data for each step. The application of gradient descent with comprehensive data is challenging. On the other hand, the mini-batch has the ability to update the network weights and calculate the gradient of the loss function.

5.3. Batch size effect

Batch size determines the number of samples trained before the model's weights are updated. It is important to select an appropriate batch size to balance model performance and memory constraints. The selected value reduces computation costs and enhances the performance of the proposed model. Figure 7 illustrates the test accuracy for different batch sizes in the suggested versions.

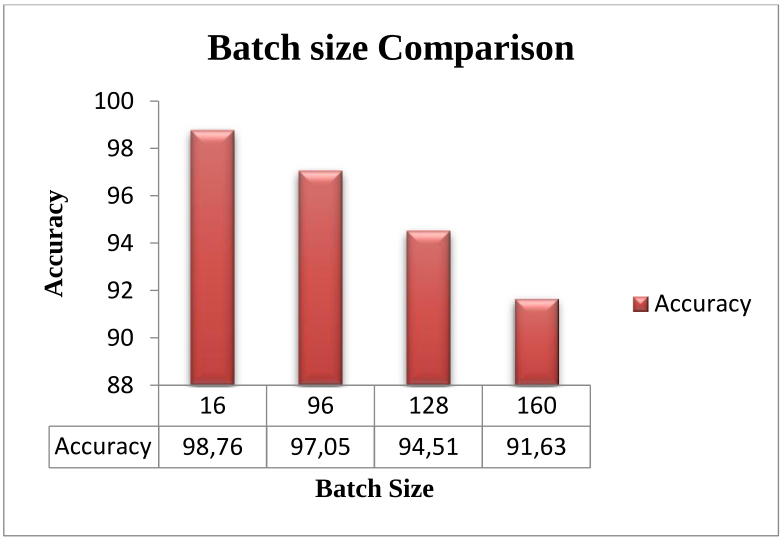


Figure 7. The accuracy of the proposed models of various batch sizes.

5.4. Number of epoch effect

When a deep learning model is trained, epoch is an important hyperparameter. Figure 8 shows that with 12 epochs, the proposed deep CNN model achieves a maximum test accuracy of 98.76%. The number of epochs should be carefully chosen to ensure sufficient training without overfitting.

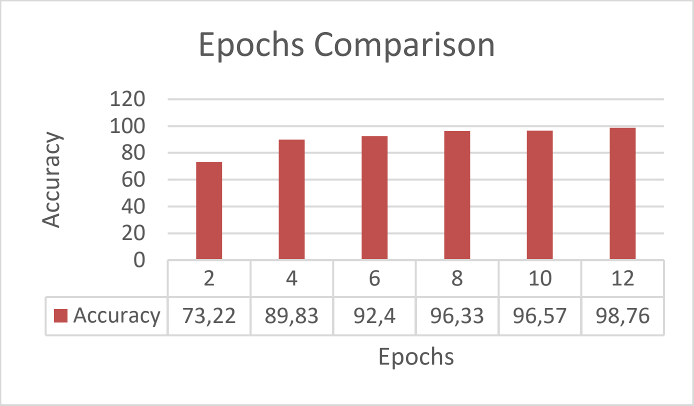
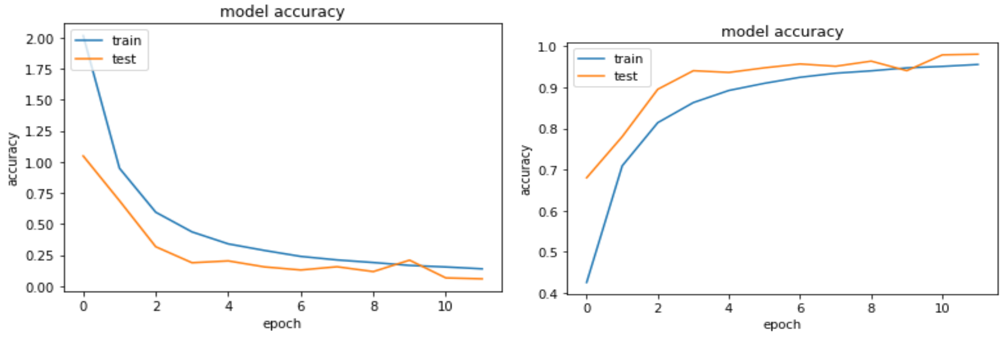


Figure 8. The test evaluation of the suggested model for various epochs.

The findings show in the training and testing phase that the model trained with the optimized hyperparameters using the augmented dataset performs well. Table 3 introduces the optimized hyperparameters of the twelve-layer deep CNN model proposed. Figure 9 provided a mean square error and accuracy evaluation of the newly presented model during the training and testing phases.



(a) (b)

Figure 9. Accuracy and mean square error for the training and testing of the proposed model.

Table 3. Hyperparameters of the twelve-layer deep CNN model.

Parameters	***Values***
Training epochs	12
Mini batch size	16
Dropout value (input layer)	0.1
Dropout value (input layer)	0.5
Learning rate	10^{-2}
Training set size	70295
Testing set size	17572

Transfer learning involves developing a reliable neural network and then using it to do related tasks.Compared to traditional transfer learning models, such as inception v3, AlexNet, VGG16, and LeNet, the efficiency of the proposed deep CNN model is compared. Figure 10 demonstrates the consequences of the research methodology and regular transfer learning. Compared with initiation v3, AlexNet, VGG16, and LeNet pre-trained versions, the introduced model yielded the maximum test accuracy.

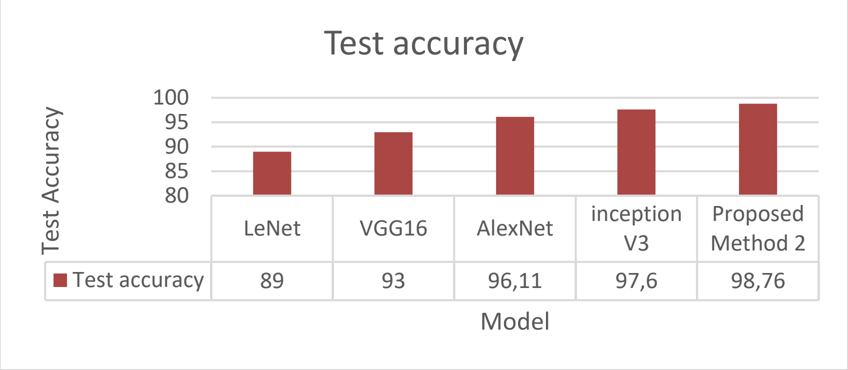
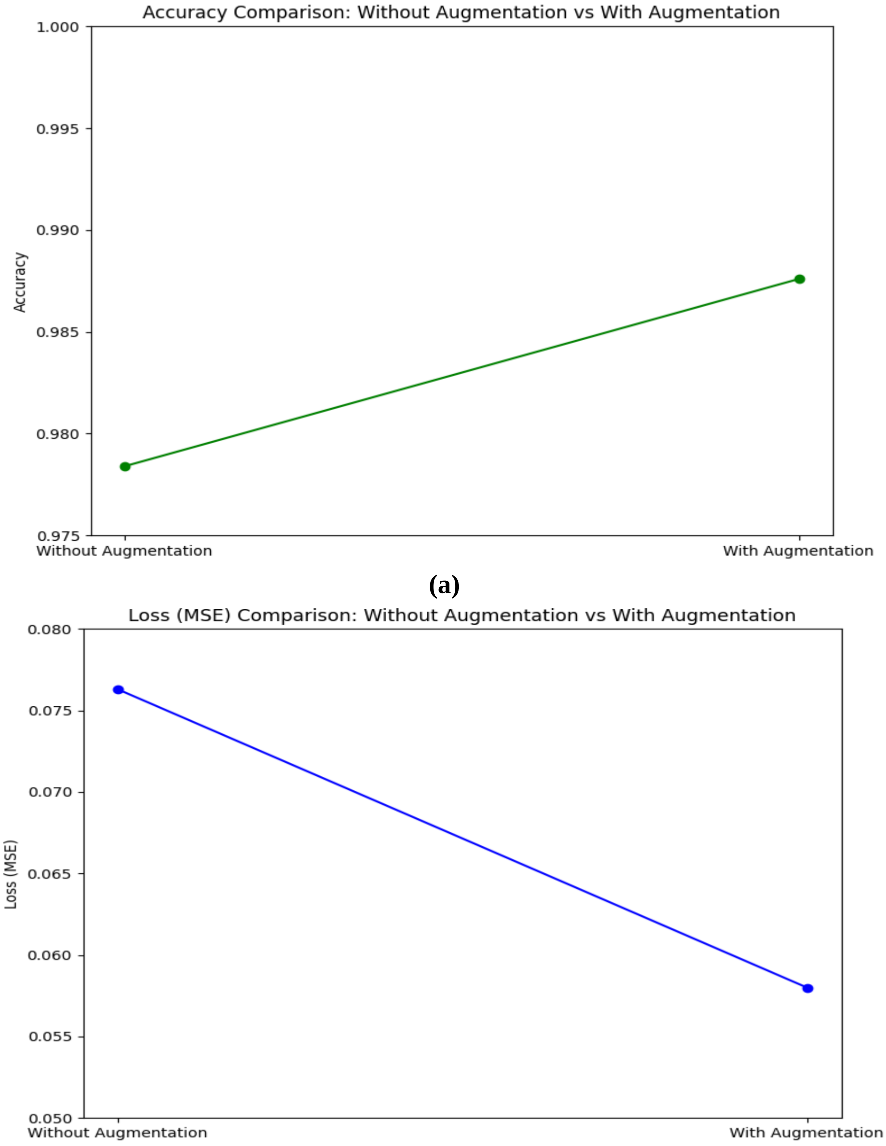


Figure 10. Compare the proposed model's accuracy against that of the trained models.

5.5. Effect of augument data on performance of framework

Data augmentation can improve the performance of a model by reducing overfitting and enhancing generalization. It also increases the variance of the input data, which helps to improve the model's performance in terms of accuracy and mean squared error (MSE), as shown in Figure 11.



(a)

(b)

Figure 11. Effect of Data Augmentation on Model Performance

5.6. Performance of the Suggested Model and Comparison with State-of-the-Art Techniques

The plant leaf diseases dataset was used to train and verify a twelve-layer deep CNN model. For the training and testing sets, the dataset is split into 70295 and 17572 pictures, respectively. and is labelled by 38 distinct classes. The suggested deep CNN model accomplishes an accuracy of 98.76 % in classification. Table 4 shows the accuracy of testing the suggested prototype for each class using the leaf image test dataset. As can see in table 4, all categories of the recommended deep CNN model accomplish an improved test accuracy ranging between 93.125 % and 100 %.

Table 4. evaluation of the proposed prototype's accuracy for each individual class.

No.	Class Name	Accuracy (%)
	Apple with scab	99.80159
	Apple with black rot	99.59759
	Apple with cedar apple rust	100
	Healthy apple	100
	Blueberry with healthy	99.33921
	Cherry with powdery mildew	100
	Cherry (including sour) powdery mildew	99.5614
	Corn (maize Cercospora leaf spot cray leaf spot	97.56098
	Corn_(maize) common rust_	99.58071
	Corn (maize) is healthy	94.75891
	Corn (maize) northern leaf blight	99.56989
	Grape black rot	99.36441
	Grape esca (black measles)	100
	Grape healthy	100
	Grape Leaf blight (Isariopsis eaf Spot)	100
	Orange Haung longbing (citrus greening)	98.40954
	Peach bacterial spot	98.91068
	Peach healthy	99.76852
	Pepper, bell bacterial spot	99.58159
	Pepper bell healthy	93.15895
	Potato early blight	99.79381
	Potato healthy	99.58763
	Potato late blight	99.34211
	Raspberry healthy	99.77528
	Soybean healthy	98.81188
	Squash Powdery mildew	100
	Strawberry healthy	99.54955
	Strawberry leaf scorch	100
	Tomato bacterial spot	98.58824
	Tomato early blight	93.125
	Tomato healthy	96.11231
	Tomato late blight	98.7234
	Tomato leaf mold	98.62385
	Tomato Septoria leaf spot	95.4023
	Tomato Spider mites two-spotted spider mite	98.24945
	Tomato target spot	98.97959
	Tomato mosaic virus	99.77679
	Tomato yellow leaf curl virus	99.7921
	Average accuracy	98.76835

The reported averages are only displayed for multi-label or multiclass with a subset of classes, and they include the macro average (averaging the unweighted mean per-label), weighted average (averaging the support-weighted mean per-label), and micro average (averaging the total true positives, false negatives, and false positives).The calculated parameters are shown in table 5 for the proposed works, which are compared with the pre-trained model.

From the classification report in table 5, LeNet's trainable parameters are 45789602. This is a considerable number that needs more space for storage and consumes more time than our proposed model. The report classification of plant diseases using VGG is provided in Table 6. While the findings in this table seem fine, we believe that the benchmark work is more expensive than ours and uses considerably more storage and parameters. The first fully connected layer is responsible for the majority of these factors. The findings of AlexNet seem attractive, but it also takes a large number of resources in terms of time and computation power, which seems unfeasible in practice.

Furthermore, this could cause overload on the network because a large number of trainable parameters are needed. Additionally, the proposed model takes up less space than these parameters. The results show that the classification report is marginally less than our proposed model. The proposed model outperforms all the pre-trained models, and it has achieved higher values in all the evaluations considered. In comparison to the current models, it offers a better performance model and exhibits a consistent and efficient performance across all performance measures. Table 9 shows that the proposed model provides above 98% accuracy. It also shows that it requires less time and computation power in comparison to the existing models.

Table 5.The comparison between the suggested model and transfer learning models.

Method	Accuracy%	Classification report									Trainable Parameters
		Micro			Macro			Weighted			
		Precision	recall	F1	Precision	recall	F1	Precision	recall	F1	
LeNet	89	89	89	89	90	89	89	90	89	89	45789602
Alex Net	96	96	96	96	96	96	96	96	96	96	41463370
VGG 16	92	92	92	92	93	92	92	93	92	92	14714688
Inception V3	97	97	97	97	97	97	97	97	97	97	21846214
Proposed 2	**98.76**	**98**	**98**	**98**	**98**	**98**	**98**	**98**	**98**	**98**	**8567142**

As seen in the appendix, the confusion matrix on our dataset is drawn to display the proper classification of 38 plant diseases. The correctly labelled images are depicted in this figure by the elements of the main diagonal of the confusion matrix. Thus, it can be shown that the model correctly classified 17353 objects from the test collection and 219 objects were misclassified.

As shown in Table 6, the true positive (TP), false positive (FP), false negative (FN), and true negative (TN) statistical parameters can be measured using the confusion matrix. This allows the details of the classification using our models to be presented. The actual positive for apple with scab is found to be 503. The false-positive and false-negative are provided as 2, 1, respectively. Notice that a false positive is provided by approximately 1.9%. The TP, FP, and FN of the second class are 495, 2 and 2, respectively. Corn (maize) healthy achieves accuracy 94.75%, as shown in table 4 and 5.25% for FN. The twenty-five items belonging to the Corn (maize) healthy were incorrectly predicted as twenty-four objects to Corn (maize Cercospora leaf spot Gray leaf spot), one object to Corn (maize) Common rust. Also, pepper bell healthy gives 463, 34 for TP and FN. Thirty-four objects were incorrectly predicted as two objects to apple with cedar apple rust, five to blueberry with healthy, seventeen to peach Bacterial spot, three to peach healthy, six potato late blight and one to tomato spider mites two-spotted spider mite. The two results are less than other classes. Also, two results provide a more negligible effect on the performance and average accuracy of the model. The actual testing dataset and the quantity of correctly categorized photos varied by a small amount. These statistics offer some specifics and extra information, demonstrating that our algorithm performs well in real datasets and is more applicable for classifying plant diseases.

Table 6. examples of statistical parameters from the examination of the confusion matrix.

class	Data set size	True Positive (TP)	False Positive (FP)	False Negative (FN)	True Negative (TN)
Apple with scab	504	503	2	1	16850
Apple with black rot	497	495	2	2	16858
Corn (maize) healthy	477	452	25	25	16901
Pepper bell healthy	497	463	1	34	16890

The suggested model has been contrasted with a number of machine learning methods, including decision trees, support vector machines, and k-nearest neighbors (KNN). Correspondingly, the following evaluation procedures are compared in terms of the efficiency of the models. Finally, The testing accuracy for various models is displayed in Table 7.In conclusion, findings suggest that when compared to the other models mentioned above, the proposed model is superior. In addition, this paper considers many diseases and data collection than most papers exist in the literature.

Table 7. Machine learning comparisons between the various proposed models.

Reference	**Method**	**Accuracy (%)**	**Number of diseases**	**Size of data set**
[30]	KNN	96.76	5 types of diseases	237 leaf images
[4]	DT	97.3	Five types of tomato diseases	383 digital images
[5]	SVM	95	Three types of diseases	300 images of potato leaves
Proposed study	Deep CNN	98.76	38 types of diseases	87867 leaf images

Table 8 provides a comparison between the benchmark [19] and the proposed framework for plant disease classification. The benchmark model integrates a 9-layer CNN with PCA dimensionality reduction, while the suggested use is only a 12-layer CNN without the need for additional preprocessing steps like PCA. The suggested model improves accuracy to an average of 98.76% over the benchmark study's 96.46% results, resulting in a relative improvement of approximately 2.3%. This improvement demonstrates that a deeper architecture of the network can capture complex patterns in plant disease images.

On the other hand, the benchmark model processes data after applying PCA, which results in a reduction in dimensionality with the loss of some information that affects the performance of the model, while the proposed model learns automatically from different levels of extracted features which enhance the performance of the model and make it suitable for use in plant disease classification.

Furthermore, the drop in mean squared error (MSE) of approximately 88.08% compared to the benchmark reflects the ability of the model to minimize the error between actual and predicted classifications. The last key improvement is that the proposed model needs only 12 epochs to reach optimal performance, while the benchmark needs 3000 epochs. This indicates that the framework is more computationally efficient and faster

Table 8. Comparison between the twelve-layer deep CNN and the benchmark.

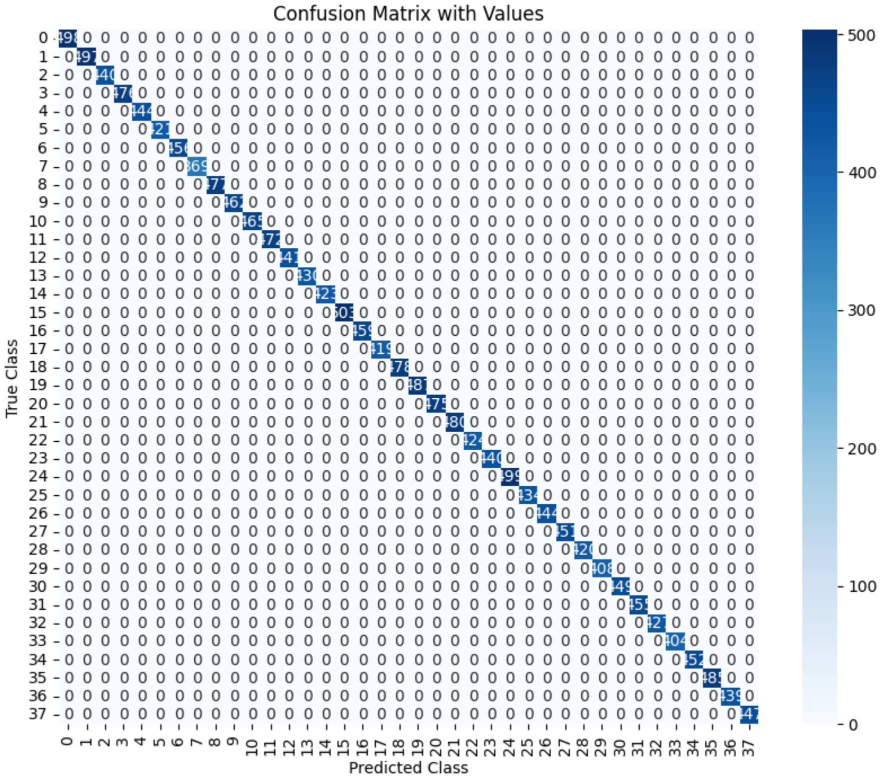
Key parameters indicators	Benchmark	Proposed model
Test size	1950	17572
Number of layers	9	12
Number of epochs	3000	12
Average accuracy	96.46%	98.76%
Missed classifier	3.5%	1.2%
Mean square error	0.4867	0.0580
Range accuracy of all classes	92%-100%	93.125%-100%

6. CONCLUSION

This paper proposed a twelve-layer deep CNN model used to identify and classify plant diseases. The proposed deep learning approach automatically removed the discrimination related to the plants' diseases and classified 38 different categories of plant leaf images with high accuracy. In addition, augmentation of data has been used in this paper to enhance the model's efficiency and accuracy. The powerful twelve-layer deep CNN model was learned and tested using an augmented dataset of 87,867 images and 12 training epochs. As a result, the proposed model achieved an average accuracy of 98.76 % in classifying the test collection of plant leaf images and between 93.125% and 100% for the individual class. Furthermore, the number of training iterations, batch size, and dropout iterations showed a significant impact on the respective performance.

Furthermore, the results showed that the max-pooling method did much better than average pooling. The proposed deep learning approach has been applied using the Kaggle and GPU processer. We have used 87867 images of leaves diseases as an input dataset. The results demonstrated that the performance reaches 98% for precision, recall and F1 scores. The findings in this paper revealed that using the proposed model from the plant image diseases for identification and classification achieve a very successful prediction performance and demonstrates a high degree of accuracy. These findings significantly underpin the contribution of this research. Finally, the results showed that the proposed model outperforms the state-of-the-art pre-trained models such as LeNet, VGG 16, AlexNet and inception v3. In particular, the comparison carried out in this paper showed that the suggested model achieves higher efficiency than the state-of-the-art pre-trained models. The results demonstrated that the proposed model had superior results, which can be summarized as the following: (1) reducing computation time, (2) improved the convergence speed, (3) enhancing the scalability, which markedly improves the performance of identification and classification of plant diseases in comparison to the state-of-the-art works.

Appendix: Confusion matrix of twelve-layer deep CNN.



REFERENCES

[1] M. A. Khan, M. I. Lali, and M. Sharif, "An Optimized Method for Segmentation and Classification of Apple Diseases based on Strong Correlation and Genetic Algorithm based Feature Selection," IEEE Access, vol. 7, pp. 46261–46277, 2019.

[2] A. Chandarakesan, S. Muruhan, and R. R. A. Sayanam, “of Plant Diseases on Human Health,” Int. J. Nutr. Pharmacol. Neurol. Dis. |, vol. 8, pp. 41–46, 2018.

[3] A. Ramcharan, K. Baranowski, P. McCloskey, B. Ahmed, J. Legg, and D. P. Hughes, "Deep learning for image-based cassava disease detection," Front. Plant Sci., vol. 8, pp. 1–7, 2017.

[4] H. Sabrol and K. Satish, "Tomato Plant Disease Classification in Digital Images using Classification Tree," 2016 Int. Conf. Commun. Signal Process. (ICCSP). IEEE, pp. 1242–1246, 2016.

[5] M. Islam, A. Dinh, and K. Wahid, "Detection of Potato Diseases Using Image Segmentation and Multiclass Support Vector Machine," 2017 IEEE 30th Can. Conf. Electr. Comput. Eng. (CCECE). IEEE, pp. 8–11, 2017.

[6] X. Zhang, Y. U. E. Qiao, F. Meng, C. Fan, and M. Zhang, "Identification of Maize Leaf Diseases Using Improved Deep Convolutional Neural Networks," IEEE Access, vol. 6, pp. 30370–30377, 2018.

[7] P. Jiang, Y. Chen, and B. I. N. Liu, "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.

[8] M. E. Tenekeci and R. Tas, "Detection of pepper fusarium disease using machine learning algorithms based on spectral reflectanc," Sustain. Comput. Informatics Syst., pp. 1–8, 2019.

[9] A. Kamilaris and F. X. Prenafeta-boldú, "Deep learning in agriculture : A survey," Comput. Electron. Agric., vol. 147, no. February, pp. 70–90, 2018.

[10] J. Chen, J. Chen, D. Zhang, Y. Sun, and Y. A. Nanehkaran, "Using deep transfer learning for image-based plant disease identification," Comput. Electron. Agric., vol. 173, no. November 2019, p. 105393, 2020.

[11] Y. Lu, S. Yi, N. Zeng, Y. Liu, and Y. Zhang, "Identification of rice diseases using deep convolutional neural networks," Neurocomputing, vol. 267, pp. 378–384, 2017.

[12] M. Agarwal, A. Singh, S. Arjaria, A. Sinha, and S. Gupta, "ToLeD: Tomato Leaf Disease Detection using Convolution Neural Network," Procedia Comput. Sci., vol. 167, pp. 293–301, 2020.

[13] M. Shaha and M. Pawar, "Transfer Learning for Image Classification," Proc. 2nd Int. Conf. Electron. Commun. Aerosp. Technol. ICECA 2018, no. Iceca, pp. 656–660, 2018.

[14] S. Cheng and G. Zhou, "Facial Expression Recognition Method Based on Improved VGG Convolutional Neural Network," Int. J. Pattern Recognit. Artif. Intell., vol. 34, no. 7, pp. 1–16, 2020.

[15] R. Jain, P. Nagrath, G. Kataria, V. Sirish Kaushik, and D. Jude Hemanth, "Pneumonia detection in chest X-ray images using convolutional neural networks and transfer learning," Meas. J. Int. Meas. Confed., vol. 165, p. 108046, 2020.

[16] S. Ramesh, "Plant Disease Detection Using Machine Learning," 2018 Int. Conf. Des. Innov. 3Cs Comput. Commun. Control (ICDI3C). IEEE, pp. 41–45, 2018.

[17] A. K. Rangarajan and R. Purushothaman, "Tomato crop disease classification using pre-trained deep learning algorithm," Procedia Comput. Sci., vol. 133, pp. 1040–1047, 2018.

[18] W. jie Liang, H. Zhang, G. feng Zhang, and H. xin Cao, "Rice Blast Disease Recognition Using a Deep Convolutional Neural Network," Sci. Rep., vol. 9, no. 1, pp. 1–10, 2019.

[19] G. Geetharamani and A. P. J, "Identification of plant leaf diseases using a nine-layer deep convolutional neural network," Comput. Electr. Eng., vol. 76, pp. 323–338, 2019.

[20] S. Ramesh and D. Vydeki, "Recognition and classification of paddy leaf diseases using Optimized Deep Neural network with Jaya algorithm," Inf. Process. Agric., vol. 7, no. 2, pp. 249–260, 2020.

[21] M. A. Jasim and J. M. Al-Tuwaijari, "Plant Leaf Diseases Detection and Classification Using Image Processing and Deep Learning Techniques," P2020 Int. Conf. Comput. Sci. Softw. Eng. (CSASE). IEEE, pp. 259–265, 2020.

[22] A. Darwish, D. Ezzat, and A. E. Hassanien, "An optimized model based on convolutional neural networks and orthogonal learning particle swarm optimization algorithm for plant diseases diagnosis," Swarm Evol. Comput., vol. 52, p. 100616, 2020.

[23] Ü. Atila, M. Uçar, K. Akyol, and E. Uçar, "Plant leaf disease classification using EfficientNet deep learning model," Ecol. Inform., vol. 61, p. 101182, 2021.

[24] R. C. Joshi, M. Kaushik, M. K. Dutta, A. Srivastava, and N. Choudhary, "VirLeafNet: Automatic analysis and viral disease diagnosis using deep-learning in Vigna mungo plant," Ecol. Inform., vol. 61, p. 101197, 2021.

[25] B. Kim, Y. K. Han, J. H. Park, and J. Lee, "Improved Vision-Based Detection of Strawberry Diseases Using a Deep Neural Network," Front. Plant Sci., vol. 11, pp. 1–14, 2021.

[26] A. Kamilaris and F. X. Prenafeta-boldú, "Deep learning in agriculture : A survey," Comput. Electron. Agric., vol. 147, pp. 70–90, 2018.

[27] F. Lei, X. Liu, Q. Dai, and B. W.-K. Ling, "Shallow convolutional neural network for image classification," SN Appl. Sci., vol. 2, no. 1, pp. 1–8, 2020.

[28] C. Garbin, X. Zhu, and O. Marques, "Dropout vs. batch normalization: an empirical study of their impact to deep learning," Multimed. Tools Appl., vol. 79, no. 19–20, pp. 12777–12815, 2020.

[29] Z. Qin, F. Yu, C. Liu, and X. Chen, "How convolutional neural networks see the world --- A survey of convolutional neural network visualization methods," Math. Found. Comput., vol. 1, no. 2, pp. 149–180, 2018.

[30] E. Hossain, M. F. Hossain, and M. A. Rahaman, "A Color and Texture Based Approach for the Detection and Classification of Plant Leaf Disease Using KNN Classifier," 2nd Int. Conf. Electr. Comput. Commun. Eng. ECCE 2019, pp. 1–6, 2019.