

How to Cite:

Kumar, P. D., & Suhasini, A. (2022). Feature extraction and classification based crop disease detection using deep neural network and ensemble convolutional neural network architecture. *International Journal of Health Sciences*, 6(S3), 2257–2273.
<https://doi.org/10.53730/ijhs.v6nS3.5997>

Feature extraction and classification based crop disease detection using deep neural network and ensemble convolutional neural network architecture

P. Dharmendra Kumar

Department of Computer Science and Engineering Annamalai University

A. Suhasini

Department of Computer Science and Engineering Annamalai University

Abstract---Crop disease identification is crucial for avoiding production losses and lowering the amount of agricultural items produced. Machine learning-based approaches can be utilized to solve these types of problems. Deep Learning Techniques, which are commonly employed in image processing, have recently been involved in a number of agricultural applications. This research suggests detecting agricultural diseases using feature extraction and classifying those using deep learning algorithms. For the extraction of the segmented feature, we employ a pre-training model-based Deep Neural Network. The segmented feature will then be categorized using an ensemble classifier composed of Visual Geometry Group (19) and Inception V3++ architecture. This Ensemble Convolution Neural Networks (EN-CNN) classification will increase illness detection accuracy while minimizing time. Pre-trained Deep Convolution Neural Network (PDCNN) and EN-CNN are used to classify damaged and healthy leaves in a dataset of healthy and diseased leaves. The suggested model's resilience is empirically shown for crop disease detection in tomatoes and grapes. The suggested approach evaluated parameters with a 97 percent accuracy rate, a 98 percent precision rate, a 98 percent recall rate, and a 98 percent F-1 score rate.

Keywords---crop diseases, deep learning, feature extraction, classification, PDCNN, EN-CNN, VGG-19, inception V3++.

Introduction

The leaves of the agricultural crops provide vital information related to the nature and productivity of horticulture. Production of food is affected by numerous factors like change in climate, infertile soil and presence of weed. The threat to the increase in agricultural products is the disease found either on the leaf or plant which introduces economic loss [1]. When infections/bacteria/virus in crops are failed to diagnose, pesticide/fungicide cannot be used efficiently. Thus biological features of crop and plant diseases have to be focused and considered. Most advanced technologies have to be used to optimize decision-making. Experts perform inspection and review the biological aspects of plants and diagnose when necessary which increases cost and is time-consuming [2]. The leaves of agricultural crops contain important information on the nature and productivity of horticulture. Food production is influenced by a variety of factors, including climate change, infertile soil, and the presence of weeds. The illness discovered on the leaf or plant, which causes economic loss, is a threat to the rise in agricultural goods [1]. Pesticides and fungicides cannot be applied effectively if infections/bacteria/viruses in crops are not diagnosed. Thus, the biological characteristics of crop and plant diseases must be focused on and examined. To maximise decision-making, the most modern technology must be applied. Experts do inspections, analyse biological characteristics of plants, and diagnose when required, which adds expense and time to the process.

CNN-based algorithms, in particular, appear to hold the promise of automatically learning decisive and discriminative characteristics. Deep Learning (DL) is made up of many convolution layers that represent learning data characteristics [4]. Plant disease detection is carried out using a deep-learning model, which has a few drawbacks in that large amounts of data are necessary for training the network. For tiny datasets, the poorest performance is experienced. Transfer learning has various advantages when a large number of data is not required for training the network. Transfer learning is commonly utilised in illness detection techniques because it reduces training time, computing cost, and generalisation error while developing a Deep Learning model [5]. This work uses various DL models for detecting plant diseases. Here the crop disease image dataset has been extracted using pre-training model based neural networks, and then the extracted feature has been classified using Ensemble architecture of CNN. The ensemble architecture uses Visual Geometry Group (VGG-19) and inception V3++. The experimental results show the enhanced accuracy.

The leaves of the agricultural crops provide vital information related to the nature and productivity of horticulture. Production of food is affected by numerous factors like change in climate, infertile soil, and the presence of weeds. The threat to the increase in agricultural products is the disease found either on the leaf or plant which introduces economic loss [1]. When infections/bacteria/viruses in crops are failed to diagnose, pesticide/fungicide cannot be used efficiently. Thus biological features of crop and plant diseases have to be focused on and considered. Most advanced technologies have to be used to optimize decision-making. Experts perform inspection and review the biological aspects of plants and diagnose when necessary which increases cost and is time-consuming [2]. The use of advanced and intellectual techniques solves these issues by detecting

plant diseases, thus Machine Learning (ML) approaches are involved in numerous studies to perform operations related to agriculture [3]. But in recent works, Deep Learning (DL) approaches that are based on ML are effectively involved in detecting, recognizing, and classifying objects in real-life applications. Thus research on agriculture uses techniques based on DL which produces satisfactory results in discriminating crop/weed and harvesting fruits.

Particularly, CNN-based techniques are promising to learn decisive and discriminative features automatically. Deep learning (DL) comprises several convolution layers representing learning data features [4]. Detection of plant disease is performed with a deep-learning model which also has a few limitations where huge data is required for training the network. For small datasets, the very worst performance is performance experienced. Transfer learning exhibits numerous benefits where a huge volume of data is not necessary for training the network. Transfer learning is widely used in approaches involved for the detection of diseases as they decrease the training time, computation cost, generalization error while designing a Deep Learning model [5].

This study uses a variety of DL models to identify plant illnesses. In this case, the crop disease picture dataset was extracted using pre-training model-based neural networks, and the retrieved feature was categorised using CNN's Ensemble architecture. Visual Geometry Group (VGG-19) and conception V3++ are used in the ensemble architecture. The experimental findings indicate improved precision. This research is provided as follows: Section 2 summarises the existing research on plant disease detection. The illness detection model presented in this study is elaborated in Section 3. Section 4 demonstrates the model's performance by examining the generated data based on prediction abilities. Section 5 brings everything to a close and shows the future scope.

Related works

Several techniques were used to determine image pathology accurately. Generally, image processing techniques, K-means clustering, Support Vector Machine (SVM) classifiers, and so forth are used but are not more attractive. Recently, field-based approaches are also used. Approaches based on deep neural networks are better than traditional image processing and recognition methods. Standard DL approaches are involved in classifying plant diseases. Moreover, few DL algorithms were modified to produce better classification performance with the disease in plants. Few of the promising recent works are discussed here. In [6], a CNN model was used for classifying diseases in tea leaves. In [7], MobileNet models were revised for plant diseases classification [7]. In [8], two deep learning architectures were used which were based on residual learning and attention methods for classifying diseases in tomato leaf. In [9], other CNN-based architecture was applied on the Plant dataset for disease classification. In [10], a CNN-based model was utilized for groundnut disease classification [10]. In [11], the performance of advanced training approaches like AlexNet and GoogLeNet were evaluated. In [12], a comparative study was presented showing the significance of fine-tuned techniques with standard DL architectures to classify plant disease. In [13], recent plant disease classification approaches were presented in detail. In [14], a Region-based Convolution Neural Network (RCNN)

was developed for image detection. In [15], deep learning approaches were designed to localize and diagnose diseases in tomato leaves. In [16], two approaches based on KNN learning were implemented for automatic pest detection in greenhouse tomato and pepper crops and were compared. DL methods were better than ML algorithms. In [17], the DL technique was designed for diagnosing Cassava leaves disease with Single Shot MultiBox Detector (SSD). In [18], used CNN to recognize plant disease and estimated the defects in plant leaves. From the above discussion, it is concluded that several recent works are related to the classification of plant diseases.

However, it is analyzed that localization and classification of plant disease is a complex task and are not focused on that much. Not even a single work has presented a comparative study of classifying 38 classes of plant disease using DL architectures. Thus, this research evaluates three standard DL-based object detection approaches namely Single Feed-forward Neural Network, Region Proposal Network, and Region-based Fully Convolution Network using a transfer learning method on agricultural issues in identifying plant diseases. Transfer learning models are successful in recognizing objects. Practically, reusing or transferring information from the learned tasks increases the accuracy of learning new tasks using DL architectures.

Proposed crop disease detection model

Here the proposed technique is based on feature extraction and classification using Deep Learning Architecture. The collected dataset of tomatoes and grapes has been processed for noise removal and gradient-based active contour segmentation. Then this segmented data has been extracted using Pre-trained Deep Convolution Neural Network (PDCNN), by this, the image has been pre-trained based on a neural network. Then the extracted pre-trained data will be classified using Ensemble CNN which is comprised of VGG-19 and Inception V3++ architectures of CNN. The overall research architecture is given in figure-1.

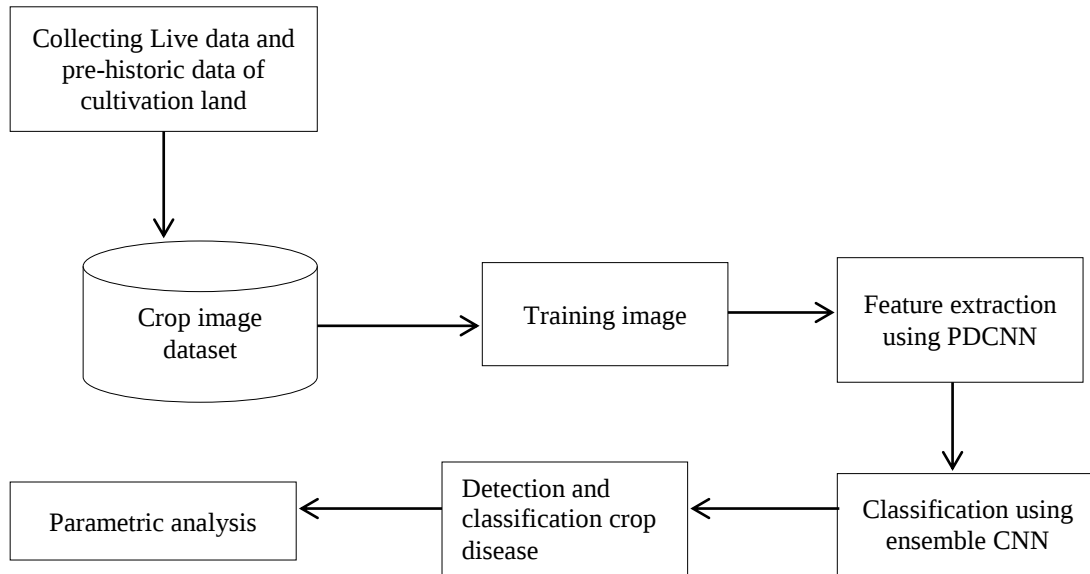


Figure 1. Overall proposed extraction and classification model

Dataset description

To train and test the developed model, open-access standard dataset consisting of 54,305 diseased and healthy leaf images of 38 classes from 14 species captured under laboratory conditions is used. Here, three formats of Plant datasets are used. Model was implemented with colored, segmented and gray-scale leaf images. The smoothed background segmented images provided more useful information making analysis easier. The dataset images were partitioned as training and testing set. The model was evaluated using three sets, namely 80–20 (where 80% were involved in training and 20% in testing), 70–30 (70% were involved in training and 30% in testing), and 60–40 (60% were involved in training and 40% in testing).

Disease segmentation using active contour with gradient descent method

Curves enclosing objects which have to be identified are inserted to define the actual level contours. Gradient operator against discrete image x is given by first order finite differences represented as eqn (1).

$$(\nabla x)_i = (x_{i_1,1_2} - x_{i_1-1,l_2}, x_{i_1,l_2} - x_{i_1,l_2-1}) \quad (1)$$

With first order finite differences, divergence operator is described for continuous vector field v is given as eqn (2)

$$\text{div} = -\nabla \quad (2)$$

whose discrete form is given in eqn (3),

$$\text{div}(v)_i = v^1_{i_1} + 1, i_2 - v^1_{i_1, i_2} + v^2_{i_1, i_2+1} - v^2_{i_1, i_2} \quad (3)$$

When the cost function is minimized, optimized image is obtained in eqn (4)

$$\min_{x \in \mathbb{R}^A} f(x) = \frac{1}{2} \|y - x\|^2 + \lambda J_\epsilon(x) \quad (4)$$

In which, $J_\epsilon(x)$ is $\sum_i \|(Gx)_i\|_i$ is the complete smoothed variation and $(Gx)_i$ is an approximated image gradient over pixel is $\lambda > 0$ is a Lagrange multiplier parameter where data fit $\|y - x\|^2$ is controlled by weights and $J_\epsilon(x)$ is a term specifying regularization. L^2 norm in $\mathbb{R}^2$ is smoothed and used to compute u with $1 > \epsilon > 0$ by eqn (5),

$$\|u\|_\epsilon = \sqrt{\epsilon^2 + \|u\|^2} \quad (5)$$

ϵ represent the small regularization parameter and $u = \nabla f(x)$. The gradient of any function f is defined as eqn (6),

$$\nabla f(x) = x - y + \lambda \nabla J_\epsilon(x) \quad (6)$$

The total variation gradient norm after smoothing is given as eqn (7)

$$\nabla J_\epsilon(x)_i = G^*(u), \text{ in which } \|u\|_i = \frac{(Gx)_i}{\|(Gx)_i\|_\epsilon} \quad (7)$$

While segmenting images using the proposed model, once the boundary is limited, saliency map was applied to the input fundus.

Pre-trained deep convolution neural network (PDCNN) based feature extraction

The layers of PDCNN are a layer of input, a layer of output, and hidden layers. The layer which is hidden generally comprises the layer of convolution, the layer of ReLU, a layer of pooling, as well as the layer which is fully connected. A layer of convolution has been operated with the function of convolution for input. Here the image has been permitted to the subsequent layer. The combination of neuron clusters in the output with the neuron which is single from the subsequent layer is done in the pooling layer. Each neuron from each layer has been connected by the fully connected layer with the subsequent layer. The neurons in the convolution layer acquire the input from the previous layer and every neuron acquire the input from each component of the previous layer. The proposed PDCNN classification is done for feature extraction from the images. Here the feature extraction is done automatically. They have been trained for extracting the features. So the neural network has higher accuracy in classification for the application of computer vision. The diseased feature has been extracted by this PDCNN from 10's and 100's of layers which are hidden. Every layer maximizes the difficulty of classifying the extracted features. The architecture of PDCNN for feature extraction is given in below figure-2.

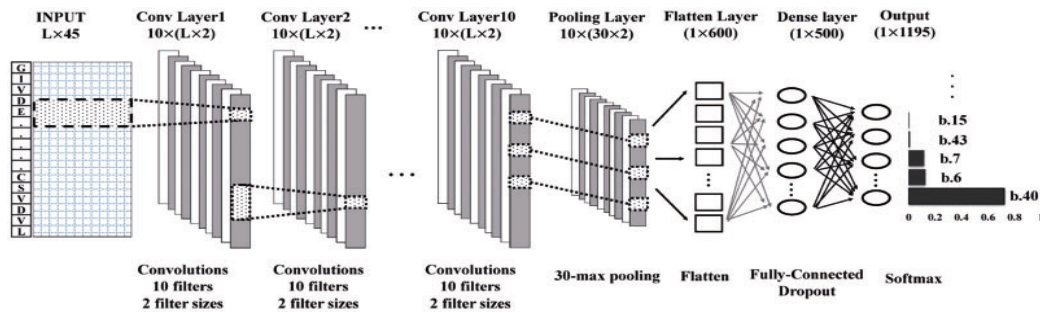


Figure 2. feature extraction architecture of PDCNN

Steps for PDCNN

- Initializes with input image.
- For generating the feature map of various filter has been used.
- Maximizing the non-linearity by applying RELU operation.
- For every map of features the pooling layer has been applied
- Suppress the images which is pooled to single maximized vector.
- The vector of input has been turned to fully connected deep neural networks.
- The features have been processed by the network. Lastly the layer of fully connected has been offered by classes of voting whether the fruit image is diseased or not.
- For maximum number of epochs the images has to be trained by forward propagation and back-propagation. This will be iterated till the defined trained image with similar weight and feature has been detected.

The level of feature extraction has been developed through these stages that certainly comprises of 3 stages which is essential stages in PDCNN technique:

- *Convolution deep Layer* – Basically, the input image is resized to $3 \times 224 \times 224$ in the PDCNN model as standard size. The resized image undergoes a stack of multiple layers as convolution layers of various receptive fields. In Convolution Recurrent layer, the basic operation is a Convolution, the Convolution is stags with several mathematical operation sliding kernel matrix over the input matrix where feature data are extracted and mapped to the successive layers. At each co-ordinate, an element-wise matrix multiplication is carried out, and the results are accumulated obtaining a feature matrix. Convolution recurrent is a distinguishable kind of linear model assist in various platforms such as image processing, statistics, physics etc. Convolution is estimated more than one axis. Where, two dimensional I is the input image and K kernel filter respectively, then the convoluted image is calculated in eqn (8):

$$S(i, j) = \sum_m \sum_n I(m, n) k(i - m, j - n) \quad (8)$$

- *Pooling layer*: This follows the convolution layer and is employed in reducing the spatial domain thereby reducing the computation in the network. In CNN, usually the kernel size is 2×2 with stride 2. Here, pooling layer

performs max operation over small spatial region R and a feature map is obtained in eqn (9):

$$p^l = \max_{i \in R} \alpha_i^l \quad (9)$$

- *Fully connected (FC) layer:* FC are simulated in PDCNN using convolution with $n1 \times n2$ as its size, where $n1$ and $n2$ are the size of the input tensor and output tensor respectively. Generally, $n1$ is a triplet ($7 \times 7 \times 512$) while $n2$ is an integer.
- *Dropout:* This layer also termed as “Drop” is usually employed to remove the input overfit; improves the conjecture of DL algorithm. Normally, it assigns the weights to the network nodes in PDCNN the percentage of 0.5 is assigned to the two drop layers).
- *Softmax:* Normally it is represented as “ σ ”, the DL model with several layers and convolution layer is followed by a ReLU layer which determines the nonlinearity in PDCNN model.

The edges have been repeated along with the classification of the plant images into 3-D groups, which has been paired into groups with size b and then directed to the PDCNN. Then, the groups are directed as volume of input to the initial layer of convolution ($c1$), comprised of filters where the stride is stable to 1, and there is no padding. Once the function of ReLU is applied, the feature maps have been created through $c1$, and this has been directed to the initial layer of MaxPool ($mp1$), with a kernel, a stride of 2, and padding. The volume of the simulated result has been directed to the following convolution layer ($c2$) with filters of size, where that has a similar stride of convolution initially as well as without padding either. Once the ReLU function is applied, the feature maps created through $c2$ are directed to the following layer of MaxPool ($mp2$), with a kernel, a stride of 2, and padding. The volume of the simulated result has been directed to the following convolution layer ($c3$), which has filters of size. The further defining of function has been done through these layer-by-feature maps in processing every component one by one, so and so. Since the network does not have max pool layer 3, the volume of output has been reformed to direct the resultant to the layer of fully-connected. The simulation of four layers of fully connected has been implemented ($fc1$; $fc2$; $fc3$ and $fc4$) with nodes, correspondingly. The output of the fully connected first 3 layers is given as, where weight matrices are given as and bias vectors are given as bfc , the previous layer output is given as (in the principal case, i.e., the output of the $C3$ layer) along with the activation function is ReLU. Lastly, the matrix for the final output is and this has been directed to $fc4$, which evaluates the networks' results by the function of softmax, which comprises the preferred labels for the original input data. In PDCNN, the structure of the function is observed for the proposed method. The cross-entropy technique has been used to represent the PDCNN system loss. It is defined as, where y and y_0 are the expected likelihood distribution and actual distribution, respectively. Thus, cross-entropy determines how predictions are calculated to describe the truth.

Ensemble architecture of CNN based classification

Here the extracted feature of disease detected part has been classified using ensemble architecture of CNN. The architecture comprises of VGG-19 and inception V3++ neural network as ensembled. **Inception-v3++** is a CNN architecture that provides improvements using Label Smoothing, Factorized 7 x 7 convolutions, and an auxiliary classifier that propagates label information. This is the improved version of the previous architectures and is efficient in computational cost. Inception modules are the basic building blocks of this model, which is computationally efficient and reduces dimension with stacked 11 convolutions. Thus, these modules overcome computational cost, overfitting, and other issues. This module permits various filters of different sizes to operate in parallel. An additional 1x1 convolutional layer is present before the 3x3 and 5x5 convolutional layers, making the process robust and inexpensive. This study imports a pre-trained Inception v3 model, which was trained on the plant Imaging Archive dataset. The dense layers of 128 x 1 replace the classification modules of this model. To obtain better performance on feature extraction, the model is fine-tuned. To train Inception v3, an input image with a size of 224 x 224 x 3 is used, which passes through several inception modules, preventing overfit and reducing computational cost. Finally, the input passes over a dense layer with 128 x 1 and either 3 x 1 or 2 x 1 as dimension for segmentation. The Inception V3 architecture is depicted in Figure 3.

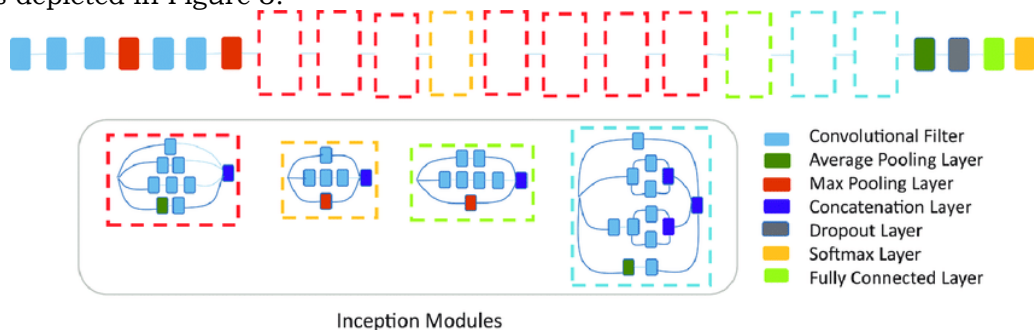


Figure 3. Inception V3 architecture

VGG-19 architecture

Numerous applications employed CNN for recognising objects, faces, and signs and proved its excellence. CNN techniques were based on supervised learning and thus trained with labelled data. CNN essentially learned the relationships between objects and their labelled classes. Here, hidden layers were used for feature extraction and the FC layer for classification. Traditionally, each layer comprises neurons that are linked to the previous layer. But this is not the case on CNN. In CNN, VGG (Visual Geometry Group) Net has around 143 million parameters, which are learnt using the ImageNet dataset. VGG-19 has 19 DNN trainable layers whose size is 574 MB in terms of fully connected nodes. When there is an increase in the number of layers, higher accuracy is obtained. The layers of VGG-19 perform convolution, and the convolution layer is trained for classification, while the dropout layer is used to regularize. The VGG-19 architecture is as illustrated in Figure 4. VGG-19 uses a 3 x 3 convolution net and max-pooling layers are used to decrease the size. Two FC layers use 4096 neurons each. Based

on individual lesions, VGG is trained and every stage is considered for testing, thereby reducing false positives. At every image pixel, a convolution process is performed using convolution layers whose output is passed to the succeeding layer. The convolution layer employs three-dimensional filters that have been trained for feature extraction. Every stacked convolution layer is linked with ReLU and max-pooling layers. The positive input region alone is allowed by the ReLU layer, which is a non-linear activation function.

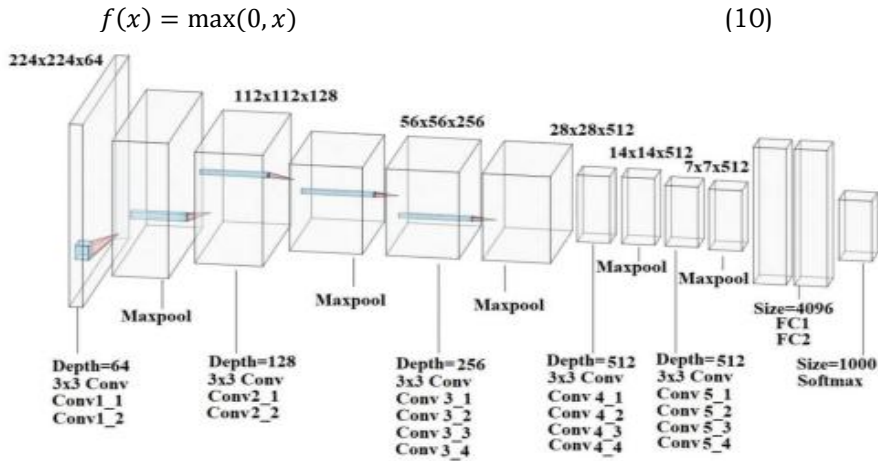


Figure 4. VGG-19 architecture

When the ReLU function is compared with the sigmoid function, the ReLU effectively indicates the best behaviour of convergence, which solves gradient problems. A max-pooling down-sampling layer follows the ReLU activation function. In general, a filter with a dimension of 2 2 is of the same step size, and in every region, the output is the maximum value. An activation function for the dense layer has to be designed while random activation makes its value zero. During random activation, neurons are eliminated in training, thereby reducing over-fitting issues.

Results and Discussion

Accuracy, precision, recall, and F1-score for each epoch are all factors in determining performance. The Plant picture dataset is represented in three ways: segmented, greyscale, and colour. In each situation, it displays a variety of performances. The color-image dataset outperformed the others. For each scenario, the CNN settings were the same. The proposed method is compared against CNN, ANN, and K-Nearest Neighbour (KNN).

System requirements

The proposed PDCNN with EN-CNN (Inception V3- Xception) segmentation is implemented in Python tool and the system configurations are: PC with Ubuntu, 4GB RAM, and Intel i3 processor.

Performance metrics

- *Accuracy*: It is defined as the total precisely predicted values to the total number of predictions as in equation (11)

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (11)$$

- *Recall*: It is the precisely predicted values to the total prediction values as in equation (12)

$$Recall = \frac{TP}{TP+FN} \quad (12)$$

- *Precision*: It is the true positive values to the total predicted values as in equation (13)

$$Precision = \frac{TP}{TP+FP} \quad (13)$$

- *F1 - Score*: It is the ratio of average precision and recall as stated in equation (14)

$$F1 - Score = 2 * \frac{Precision*Recall}{Precision+Recall} \quad (14)$$

- *Confusion Matrix*: It provides the efficiency of the proposed model in comparison with actual and predicted values. The analysis is based on the estimation of True Positive (TP), False Negative (FN), True Negative (TN), and False Positive (FP) as represented in equation (15)

$$Confusion\ Matrix = \begin{bmatrix} TP & FP \\ FN & TN \end{bmatrix} \quad (15)$$

Here, TP is the predicted value, which is anticipated as positive. FP is the predicted value which is initially estimated as negative and later as positive. TN is the value predicted as negative and anticipated as unfavourable. FN is the predicted value which is initially estimated as positive and later as negative.

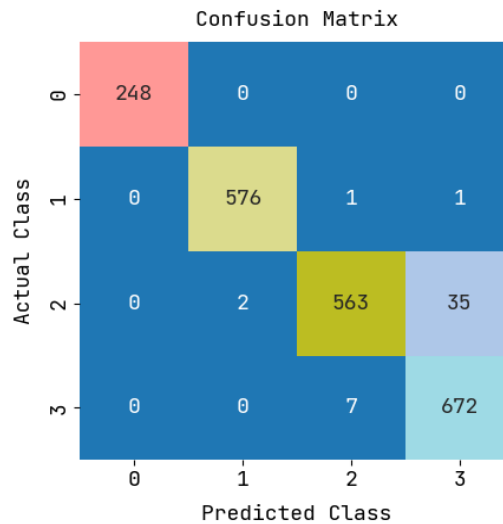
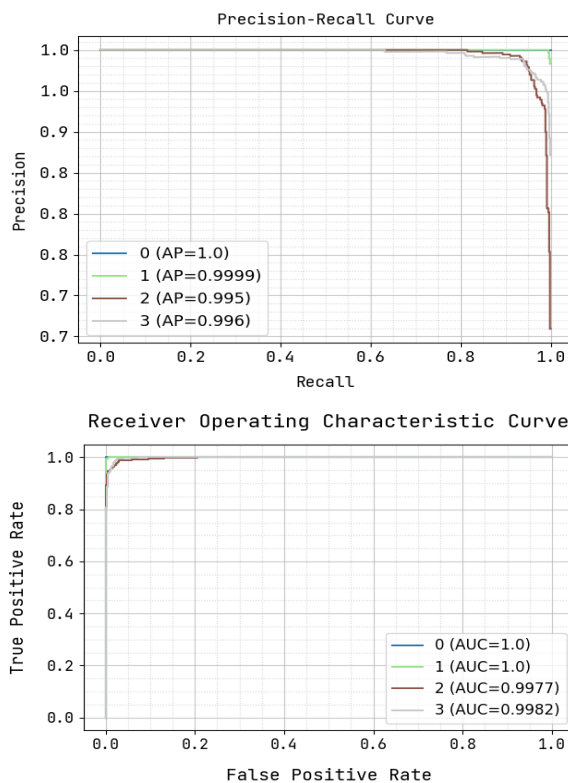


Figure 7. Confusion matrix for disease detection using PDCNN- EN-CNN

The above image shows the confusion matrix for crop disease detection using the proposed PDCNN and EN-CNN. This image shows the classes for prediction with a minimum accuracy of prediction between 8 diseased region classes, which is obtained before the loss in convergence and the error in training. Though the maximum iteration number is 60, this has certainly developed in terms of accuracy for various classes of prediction because of the errors in loss and training. Despite being the trained model, which is appropriate in terms of statistics, this model lacks efficient prediction capacity. Because of this regularization, the simulation has been maximised when the simulation for enhanced tests ranges between the values for the regularisation parameter. The above confusion matrix, the prediction class has been validated along with the parameters of precision and recall of a deep neural network classification model. The precision- recall curve and ROC curve of proposed classification technique is shown in figure-8 (a) and (b).



(a) (b)
Figure 8. (a) Precision recall curve, (b) ROC curve of proposed classification technique

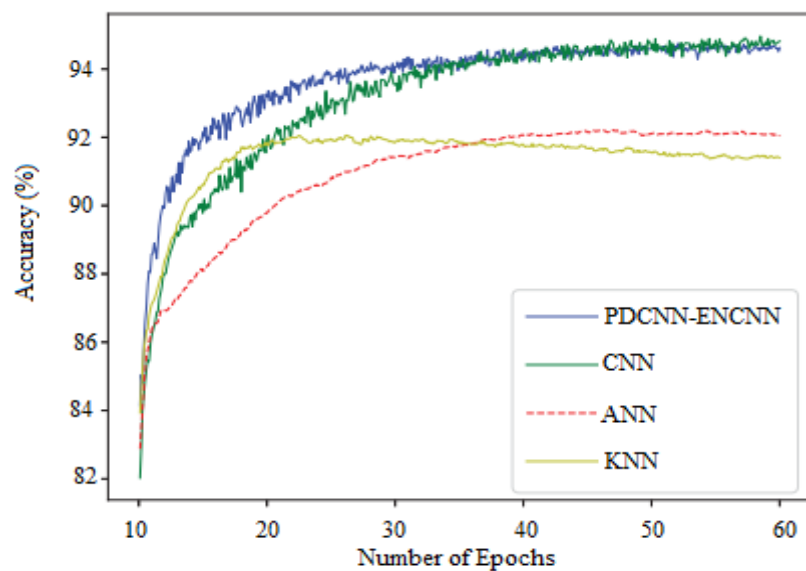
Comparative analysis of PDCNN-EN-CNN

Figure 9. Comparison of Accuracy

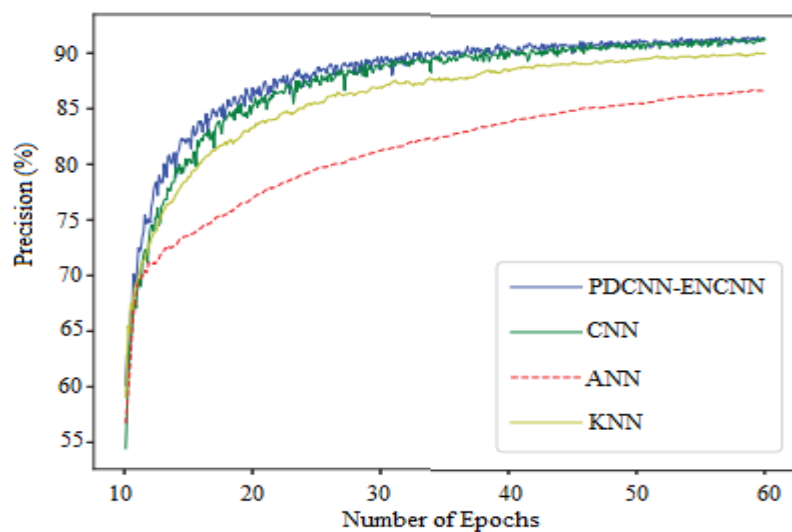


Figure 10. Comparison of Precision

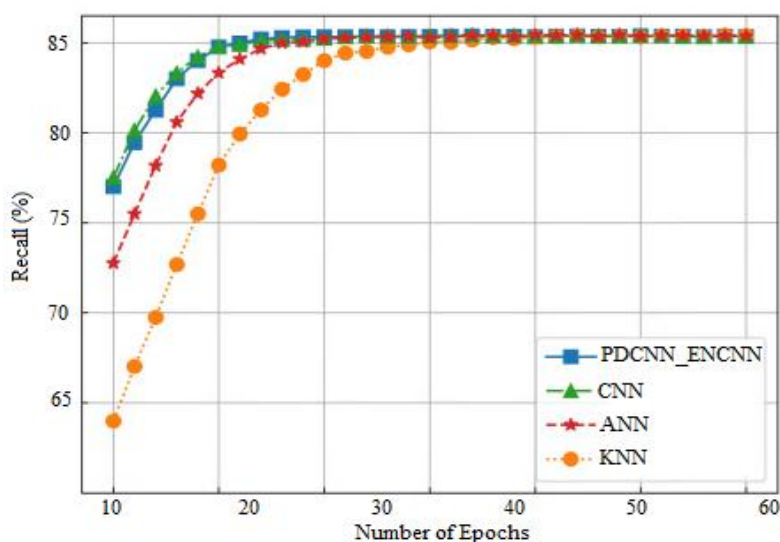


Figure 11. Comparison of Recall

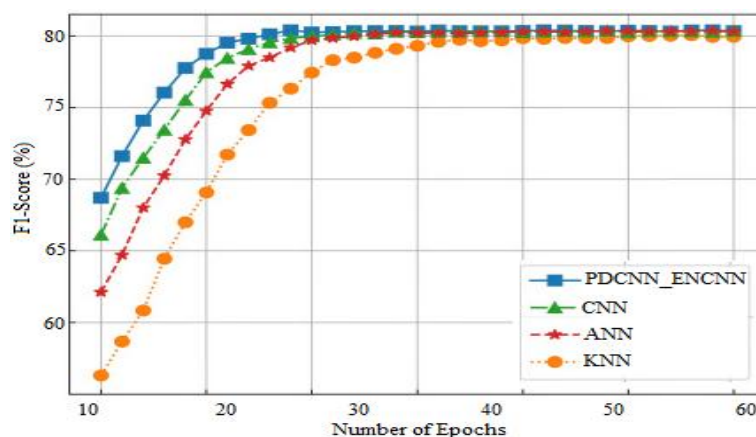


Figure 12. Comparison of F1- Score

Figure 9-12 gives the comparative analysis for the proposed PDCNN-EN-CNN gives an accuracy value of 82.93%, ANN gives 91.91%, KNN gives 75.33%, and PDCNN-EN-CNN gives 97%. For value of precision, CNN gives 81.91%, ANN gives 89.19%, KNN gives 78.45%, and PDCNN-EN-CNN gives 98%. Of these techniques, PDCNN-EN-CNN gives the optimum value. For recall value, CNN is 79.02%, ANN is 88.28%, KNN gives a recall of 74.19%, and PDCNN-EN-CNN gives 98%. The F1 score varies with CNN at 83.9%, ANN at 85.19%, KNN at 72.45%, and PDCNN-EN-CNN at 98%. The ROC-AUC value obtained by the proposed ensemble classifier is 99%, and the FPR is 0.008. Among all the techniques proposed, PDCNN-EN-CNN has obtained enhanced value.

Conclusion

There are several methods has been developed for detecting and classifying plant diseases using their leaves. Still, there is a need for efficient and effective disease

identification methods. This work uses various DL models for identifying plant diseases. Here, the crop disease image dataset has been extracted using pre-training model-based neural networks, and then the extracted feature has been classified using the Ensemble architecture of CNN. The ensemble architecture uses VGG-19 and inception V3++. The experimental results show the enhanced accuracy. The model is trained and tested using the standard dataset, which comprises 53,407 healthy and diseased images of 38 classes from 14 species captured under laboratory conditions. 80% of the entire dataset was used for training, while 20% was used for testing. An accuracy rate of 97%, a precision rate of 98%, a recall rate of 98%, and an F-1 score rate of 98% were achieved by the model. When compared with other DL models, the proposed model has better accuracy in prediction. In the future, a noisy image has to be evaluated using this model. However, the dataset has to be extended with a wide range of land area and several diseased images and improved with aerial photos captured by drones.

References

1. Sethy, P.K.; Barpanda, N.K.; Rath, A.K.; Behera, S.K. Deep feature based rice leaf disease identification using support vector machine. *Comput. Electron. Agric.* 2020, 175, 105527.
2. Chen, J.; Chen, J.; Zhang, D.; Sun, Y.; Nanekaran, Y.A. Using deep transfer learning for image-based plant disease identification. *Comput. Electron. Agric.* 2020, 173, 105393.
3. Bai, X.; Cao, Z.; Zhao, L.; Zhang, J.; Lv, C.; Li, C.; Xie, J. Rice heading stage automatic observation by multi-classifier cascade based rice spike detection method. *Agric. For. Meteorol.* 2018, 259, 260–270.
4. Ramcharan, A.; Baranowski, K.; McCloskey, P.; Ahmed, B.; Legg, J.; Hughes, D.P. Deep Learning for Image-Based Cassava Disease Detection. *Front. Plant Sci.* 2017, 8, 1852.
5. Singh, J.; Kaur, H. Plant disease detection based on region-based segmentation and KNN classifier. In *Proceedings of the International Conference on ISMAC in Computational Vision and Bio-Engineering 2018*, Palladam, India, 16–17 May 2018; pp. 1667–1675.
6. *Proceedings of the International Conference on ISMAC in Computational Vision and Bio-Engineering 2018*, Palladam, India, 16–17 May 2018; pp. 1667–1675.
7. Li, Y.; Yang, J. Few-shot cotton pest recognition and terminal realization. *Comput. Electron. Agric.* 2020, 169, 105240. [CrossRef]
8. Lee, S.H.; Goëau, H.; Bonnet, P.; Joly, A. New perspectives on plant disease characterization based on deep learning. *Comput. Electron. Agric.* 2020, 170, 105220.
9. Too, E.C.; Yujian, L.; Njuki, S.; Yingchun, L. A comparative study of fine-tuning deep learning models for plant disease identification. *Comput. Electron. Agric.* 2019, 161, 272–279.
10. Sehgal, P.; Kumar, B.; Sharma, M.; Salameh, A.A.; Kumar, S.; Asha, P. (2022), Role of IoT In Transformation Of Marketing: A Quantitative Study Of Opportunities and Challenges, *Webology*, Vol. 18, no.3, pp 1-11
11. Kumar, S. (2022). A quest for sustainium (sustainability Premium): review of sustainable bonds. *Academy of Accounting and Financial Studies Journal*, Vol. 26, no.2, pp. 1-18

12. Nisha Nandal, Dr. Naveen Nandal, Dr. Aarushi. (2020). Women Social Entrepreneurs: A growing trend in Indian Economy. *International Journal of Advanced Science and Technology*, 29(4s), 2246 - 2253.
13. Hussain, M.; Bird, J.J.; Faria, D.R. A study on cnn transfer learning for image classification. In *UK Workshop on Computational Intelligence*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 191–202.
14. Tan, M.; Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the 2019 International Conference on Machine Learning*, Long Beach, CA, USA, 9–15 June 2019; pp. 6105–6114.
15. Kaur, S.; Pandey, S.; Goel, S. Semi-automatic Leaf Disease Detection and Classification System for Soybean Culture. *IET Image Process.* 2018, 12, 1038–1048. [CrossRef]
16. Babu, M.P.; Rao, B.S. Leaves Recognition Using Back Propagation Neural Network-Advice for Pest and Disease Control on Crops. Available online: <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.110.3632&rep=rep1&type=pdf> (accessed on 1 May 2021).
17. Kamilaris, A.; Prenafeta-Boldú, F.X. Deep learning in agriculture: A survey. *Comput. Electron. Agric.* 2018, 147, 70–90.
18. Geetharamani, G.; Pandian, A. Identification of plant leaf diseases using a nine-layer deep convolutional neural network. *Comput. Electr. Eng.* 2019, 76, 323–338.
19. Jadhav, S.B.; Udupi, V.R.; Patil, S.B. Identification of plant diseases using convolutional neural networks. *Int. J. Inf. Technol.* 2020, 1–10.
20. Rangarajan Aravind, K.; Raja, P. Automated disease classification in (Selected) agricultural crops using transfer learning. *Autom. Casopis Autom. Mjer. Elektron. Računarstvo Komun.* 2020, 61, 260–272.
21. Rangarajan, A.K.; Purushothaman, R. Disease classification in eggplant using pre-trained vgg16 and msvm. *Sci. Rep.* 2020, 10, 1–11.
22. Arora, J.; Agrawal, U. Classification of Maize leaf diseases from healthy leaves using Deep Forest. *J. Artif. Intell. Syst.* 2020, 2, 14–26.