Journal of Information Technology Management

Print ISSN: 2008-5893

Online ISSN: 2423-5059

Cucumber Leaf Disease Detection and Classification Using a Deep Convolutional Neural Network

Manish K. Singh*

*Corresponding Author, School of Computer and Engineering, Galgotia's University, India. E-mail: manish5004@gmail.com

Avadhesh Kumar

School of Computer and Engineering, Galgotia's University, India. E-mail: avadheshkumar@galgotiasuniversity.edu.in

Abstract

Due to obstruction in photosynthesis, the leaves of the plants get affected by the disease. Powdery mildew is the main disease in cucumber plants which generally occurs in the middle and late stages. Cucumber plant leaves are affected by various diseases, such as powdery mildew, downy mildew and Alternaria leaf spot, which ultimately affect the photosynthesis process; that's why it is necessary to detect diseases at the right time to prevent the loss of plants. This paper aims to identify and classify diseases of cucumber leaves at the right time using a deep convolutional neural network (DCNN). In this work, the Deep-CNN model based on disease classification is used to enhance the performance of the ResNet50 model. The proposed model generates the most accurate results for cucumber disease detection using data enhancement based on a different data set. The data augmentation method plays an important role in enhancing the characteristics of cucumber leaves. Due to the requirements of the large number of parameters and the expensive computations required to modify standard CNNs, the pytorch library was used in this work which provides a wide range of deep learning algorithms. To assess the model accuracy large quantity of four types of healthy and diseased leaves and specific parameters such as batch size and epochs were compared with various machine learning algorithms such as support vector machine method, self-organizing map, convolutional neural network and proposed method in which the proposed DCNN model gave better results.

Keywords: DCNNs (Deep Convolution Neural Network), CNNs (Convolution Neural Network), Classification.

Journal of Information Technology Management, 2023, Vol. 15, Special Issue, pp. 94- 110
Published by University of Tehran, Faculty of Management
doi: https://doi.org/ 10.22059/jitm.2023.95248
Article Type: Research Paper

Received: July 06, 2023
Received in revised form: August 24, 2023
Accepted: November 09, 2023
Published online: December 24, 2023

Introduction

Plant disease is one of the major causes of the reduction in quantity and degraded quality of the product (Sannakki et al., 2013). Heat and wind can do a lot of damage to plants that have already been badly hurt. Brown, sunken spots develop on the surface of infected fruit. The lesions on the crop's character can change its colour from olive to black, powdery, and brittle. If infected fruit isn't found until after harvest, it could cost money to ship or store it (Manavalan et al., 2021). Pseudoperonospora cubensis is responsible for several diseases that affect cucumber leaves, including downy mildew. In most cases, the first visible sign of the disease is a tiny yellow spot on the upper surface of the affected leaves. As the lesions progress and the diseased areas merge, the irregular margins of the leaf can turn brown, and the plant may eventually lose all of its leaves. Infected plants will also show signs of grey mould growth on their undersides. Cantaloupes are unaffected, but their sweetness decreases. This illness thrives in damp environments. Powdery mildew is caused by two fungi that live on the upper surface of leaves: Podosphaera fuliginea and Erysiphe cichoracearum. When a plant is infected, it can become stunted, deformed, and die unexpectedly. Fruits typically aren't contaminated, but their growth and size could be affected (Manavalan, et al. 2021). Many people and organizations are actively engaged in boosting production in agriculture. In the past, many methods have been tried to stop the spread of disease in tomato plants. Tomato disease detection has become more accessible thanks to advances in technology. For this reason, many different approaches were made, such as the KNN algorithm used to find diseases (Sankaran et al., 2010). Deep learning is used to classify vast amounts of data efficiently. The proposed DCNN method requires many labelled data to train effectively. However, in many cases, the available data may need to be more diverse to prepare a robust model. Preprocessing techniques such as transformation and augmentation enhance the dataset's size and introduce variability in the training data. Transfer learning can help improve generalization performance and avoid over fitting by leveraging the pre-trained model's learned features and weights. The carefully chosen ResNet model weights demonstrate that the pre-trained ResNet model has been improved on the new dataset.

Literature Review

(Liu et al., 2018) proposed a plant leaf classification model. In this model, plant leaf classification is done by ten layers of CNN. The result on 32 kinds of 4800 images of Flavia leaf achieved 87.92% accuracy. Jiang et al. (2019) proposed a real-time detection model. In this model, a deep convolutional neural network identifies apple leaf disease. This work uses a Google Neural Network and the Rainbow concatenation method to automatically pull out and train features of an apple leaf disease. As a result, five types of apple leaf disease have been identified with high accuracy (Ertam & Aydın, 2017). In this work, various multiple activation functions such as Rectified Linear Unit (ReLu), sigmoid, Hyperbolic Tangent (tanH), and soft plus have been compared for classification purposes. The results show that the most accurate classification was given by the ReLu activation function, which achieved 98.43% accuracy. Luna et al. (2018) proposed automated image capture and transfer learning methods for identifying tomato leaf diseases such as Target Spot, Phoma Rot, Leaf Miner and healthy leaves. In this work, transfer learning and an automated capturing-based model were applied to 4,923 dataset images of tomato plant leaves, and as a result, they achieved 95.75% and 91.67% accuracy rates, respectively. Hari et al. (2019) proposed a new CNN model for detecting plant diseases. In this study, 16 layers of the CNN network with a 32 x 32 filter and the Max Pool achieved 86% accuracy on 14810 plant images.

To detect and classify disease on leaves, K Mean Clustering has been used, Grey Level Co-occurrence Matrix (GLCM) has been used for extraction of the features of an image, and a Support Vector Machine (SVM) has been used for classification. As a result, the model has achieved 90% accuracy (Prakash et al., 2017). A self-organizing map with a neural network is used to detect the rice disease; here zooming method extracts the features of images, and RGB spots have been identified with 92% accuracy (Phadikar & Sil, 2008). Image processing and deep learning techniques are used to detect and categorise diseases, where SVM is utilised to classify the images and achieve up to a 92.4% accuracy rate (Pooja et al., 2017). Nashrullah (2021) proposed an algorithm and identified tomato leaf disease with 90.37% accuracy. This technique included the following steps: preprocessing the dataset, segmentation, feature extraction, classification, and evaluating the model's performance. Here, colour-based and Gabor filters have been used for the texture features of the tomato leaf (Zhou et al., 2015). Proposed a model SVM classification. In this after-image processing, SVM classified three types of cucumber diseases, i.e. downy mildew, powdery mildew, and leaf rust and achieved 90% accuracy for the identification of cucumber leaves condition (Khan et al., 2020). Applied a deep feature selection technique on five cucumber leaf diseases and achieved a classification accuracy of 98.08% in 10.52 seconds.

Ma et al. (2018) proposed a deep convolutional neural network (DCNN) for identifying four types of cucumber disease, i.e., powdery mildew, downy mildew, anthracnose, and target leaf spots, and achieved an accuracy of 93.4%. They also compare it with other classifiers (random forest and support vector machines) and find that DCNN is more potent than the others. Atila et al. (2021) proposed the EfficientNetB0 architecture, which was trained on 55448 and 61486 images of original and augmented datasets and achieved the highest accuracy of 99.91% and 99.97%, respectively Şengür et al. (2017) proposed a model for detecting retinal vascular segmentation. In this model, retinal vascular detection is done by two convolutional layers, one loss layer, one dropout layer, and two pooling layers, achieving an accuracy of 91.78%. Muhammad Shoaib et al. (2022) worked on tomato leaves. The InceptionNet and Modified U-Net techniques have been used to classify six different types of tomato leaf diseases, with accuracy rates of 99.12% and 98.73%, respectively (Shoaib et al., 2022). Alshammari et al. proposed a model to find the most convenient feature and trained 3,400 images of different olive leaves. The result has been compared with three deep learning algorithms: Vision Transformer, VGG-16, and VGG-19. After comparing various deep learning algorithms, they found that the hybrid deep learning method, i.e., the combination of the VIT model and the VGG-15 model, achieved better results with 97% accuracy (Alshammari et al., 2022). Borhani et al. proposed the Vision Transformer (ViT) and the CNN approach for classifying various diseases in the Rice Leaf and Wheat Rust datasets. After training the model, they found that attention and convolutional blocks achieved better accuracy (Borhani et al., 2022). Table 1 lists the key distinguishing features of the study under review.

Table 1. Main characteristics of the relevant works under review.

Related work	Methodology
(Liu et al., 2018; Hari et al., 2019)	Develop a CNN
(Jiang et al., 2019)	Google Neural Network and the Rainbow concatenation method
(Ertam & Aydın, 2017)	Comparison of Rectified Linear Unit (ReLu), sigmoid, Hyperbolic Tangent (tanH), and soft plus.
(De Luna et al., 2018)	Transfer learning method
(Prakash et al., 2017)	K Mean Clustering, Grey Level Co-occurrence Matrix (GLCM) and Support Vector Machine
(Phadikar & Sil, 2008)	A self-organising map with a neural network
(Pooja et al., 2017)	Image processing and Deep learning techniques
(Nashrullah et al., 2021)	Gabor filters
(Zhou et al., 2015)	Support Vector Machine
(Khan et al., 2020)	Deep feature selection technique
(Ma et al., 2018)	Comparison of deep convolutional neural network with random forest and support vector machines
(Atila et al., 2021)	EfficientNetB0
(Şengür et al., 2017)	InceptionNet and Modified U-Net techniques
(Shoaib et al., 2022)	VGG-16
(Alshammari et al., 2022; Borhani et al., 2022)	Vision Transformer (ViT) and the CNN approach

Data augmentation

PyTorch offers an instrumental library with various methods to help with data augmentation (Jain et al., 2019). The relevant features have been extracted during training to represent the model. This approach allows the difference in network classes to be differentiated by using data augmentation to remove the most appropriate features (Ismael et al., 2020).

Overview of Residual Network

Alexnet consists of eight layers, where five layers are max pooling, and three are connected layers that use Relu activation. The Visual Geometry Group (VGG) is a deep convolutional neural network (CNN). VGG is classified into two categories based on layers: VGG16 and VGG19, where the 16 and 19 numbers indicate the number of layers. GoogleNet has more layers than AlexNet and VGG combined. Due to their vanishing gradient, deep networks are challenging to train. The slope becomes very small due to repeated multiplication, leading its performance to become saturated or decline rapidly. The residual network (ResNet 50) is used to overcome the rapidly declining problem. It consists of thousands of convolutional layers that perform better for shallower networks that overcome the vanishing and gradient problems. Network complexity has been analysed by adding more layers that accurately reflect the features (He et al., 2016). Nonlinear layers are stacked to fit the mapping shown in Equation 1a, where F(x) is the original mapping recast into Equations 1b. The residual network working diagram is shown in Figure 1.

$$\mathrm{F(X) = Y(X) - X} \tag{1a}$$

$$\mathrm{F(X) = Y(X) + X} \tag{1b}$$

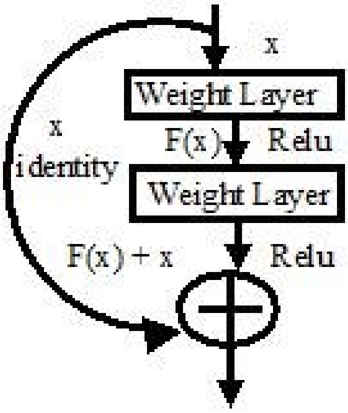


Figure 1. Residual network working diagram He et al., 2016

Training of image datasets becomes more complicated when the neural network becomes deeper. Residual networks overcome the difficulty of training datasets by using neural networks. The filter becomes deeper if all the previous, output and following layers have the exact resolution. As a result, the feature map becomes half as deep when the depth is doubled.

Overview of convolutional neural networks (CNN)

A convolutional neural network consists of mainly three layers: convolution, pooling, and fully connected layers, which establish the metaphysical pair of an image dataset, so it requires less preprocessing than other classification algorithms with enough training. CNN can learn filters and extract image features. Excellent performance with image and audio signal inputs can cover the entire area efficiently. That's why convolutional networks differ from other neural networks (Jain et al., 2019).

Methodology

The block diagram represents the proposed system for detecting cucumber leaf disease. Collection of dataset, Data augmentation, Creation of Model, Training and Validation and Classification are prominent among them. Figures 2 and 3 depict the proposed model architecture and process flow below.

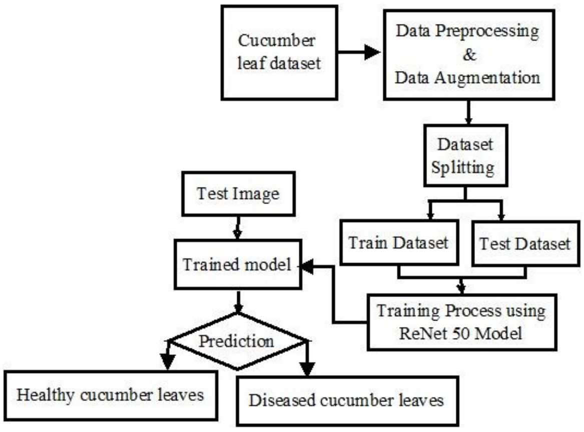


Figure 2. Proposed Model Architecture

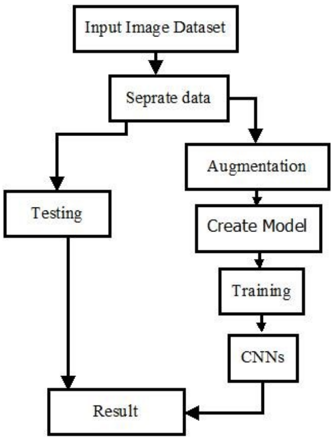


Figure 3. Work Flow Diagram (Kaushik et al., 2020.)

Collection of dataset

This work requires massive image datasets to train the model, which predicts better results. Figure 4 depicts 3,700 images of diseased cucumber leaves from a vast repository, representing four disease categories (Powdery mildew, Downy mildew, Alternaria leaf spot, and healthy cucumber leaves). Table 3 indicates the number of images used for training and testing purposes and also provides Plant Village's symptoms of various cucumber diseases. Whole image datasets are grouped into 67% and 33% for training and validation purposes. Figure 4 indicates the total number of images considered for work carried out, total number of images considered for training and testing purposes, respectively. The study of cucumber disease symptoms and weather conditions for this work is shown in Table 2.

Table 2.Cucumber Diseases, Symptoms and Condition

Cucumber Diseases	Symptoms	Conditions
Downy mildew	Diseased leaves can turn yellow and fall off the tree.	Soil moisture levels are high, and rain falls frequently.
Powdery mildew	Dark, brittle spots appear where the disease is present, signaling impending defoliation and death.	Conditions are cool and dry.
Alternaria leaf blight	Tiny brown spots on the oldest leaves surrounded by a yellow halo could be a symptom of several plant diseases or environmental factors.	The waning days of spring and the first warm days of summer

Table 3. Various cucumber disease descriptions

Types of diseases	No. of Training Image	No. of Testing Images	Symptoms
Powdery mildew	600	300	Pale yellow and white powdery leaf spots.
Downy mildew	700	320	Light green to yellow angular spots on the upper surface of leaves.
Alternaria leaf spot	800	400	Yellow, dark brown to black circular leaf spots with target concentric rings.
Healthy	400	180	No such symptoms.

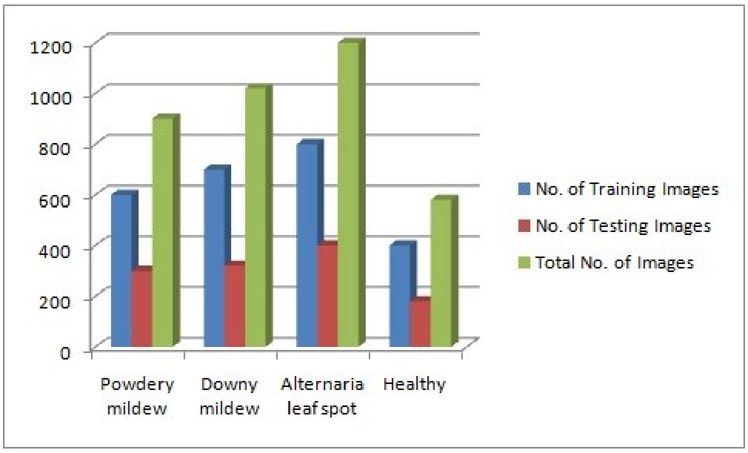


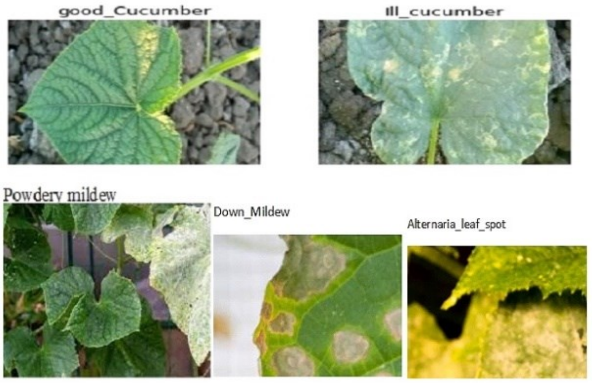
Figure 4. Number of separate image data in the testing and training data set with the total number of images taken for the work done.

Data augmentation

The proposed work includes three methods—Random Rotation, Random Resized Crop, and a combination of the first two, Rotation and Resized Crop—that use 3700 images from the data augmentation training set. The images rotate a random 30-degree spin to the left or right. Each provided image is resized randomly between 0.07 and 1.0 with a predefined aspect ratio using a random resized crop method. In the second step, use the over-centre technique to crop the images. The third approach utilises the compose function to combine the first and second processes. This function makes it easier to connect several transformations. After using the three techniques mentioned above, our dataset has expanded from 3700 to 11,100, three times its initial size. Overfitting occurred due to the large number of images in the training set, and the system failed to categorise the new image data set. The proposed model learned from various types of patterns to minimise over- and under-fitting for testing, and its overall performance has been improved. Figure 5 shows the healthy and ill images of cucumber leaves for three diseases: powdery mildew, down mildew, and Alternaria leaf spot. Augmented images of Downy mildew and values are shown in Figure 6 and Table 4.

Table 4. Values for data augmentation used for this approaches

Augmentation	Values
Zoom	30%
Rotation	30
Flipping image horizontal	True



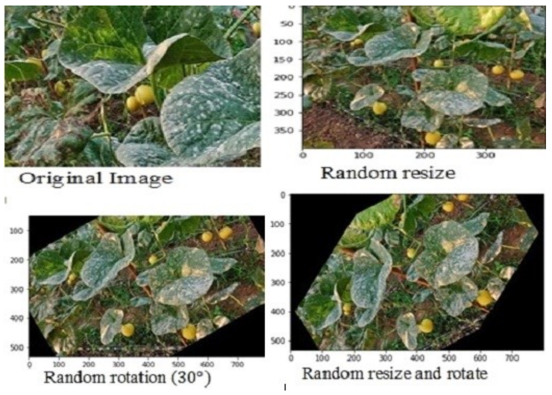


Figure 5. Healthy and diseased cucumber leaves image Figure 6.Downy mildew leaf images after augmentation

Transfer Learning (Creation of Model)

The pre-trained ResNet-50 model extends the transfer learning paradigm, which is used on CNN in two ways (Bodhwani et al., 2019). Phase one modifies the CNN based on given parameters, while phase two uses fixed feature extraction. CNN's default parameters, including those in the final layer, were preserved to use a feature extractor approach.The following is how ResNet-50 is included in our image datasets: As a first step, the pre-trained model has been introduced. The fully connected final layers of the ReNet-50 model were modified and replaced with six fully connected Soft Max layers to maintain the same number of classes in our dataset.The convolutional layers' weights and parameters are fixed. The model is trained to recognise the various cucumber illnesses. During the training, all parameters are set using an optimisation algorithm. Transfer learning is the most useful deep learning approach, where we reuse the pre-trained model for further tasks. Pre-trained models enhance skills based on problems by using a neural network. The neural network extends the skills by training thousands of images (Rezende et al., 2017). For image classification, the ResNet 50 model pulls out different features from the given image datasets, such as corners, edges, and curves. The bottleneck block has accomplished the feature extraction stage, leaving only the classification phase. The fully connected bottommost layers of the model are modified. Since our classification tasks differ, the weights, biases, and other hyper-parameters do not alter when the model is imported.

ResNet 50 Architecture

The Resnet-50 architecture is made up of 50 layers of neural networks. It employs a three-layer stack. Resnet 50 outperforms Resnet 34 in terms of accuracy. Convolution layers using 3x3 convolutions independently, followed by batch normalisation and ReLU for nonlinearity. The convolutions of each stage double the number of channels and decrease the spatial size of the feature maps by half. Global average pooling is used to produce the classification output. Resnet 50 performs 3.8 billion FLOPs, consists of 23.5 million parameters, and is relatively lower than similar networks, making ResNet50 a lightweight model. Resnet employs skip connections to connect the output layers.

Figure 7 depicts the Resnet 50 architectural diagram, while Figures 8 (a) and 8 (b) illustrate the skip connection and without the skip connection.

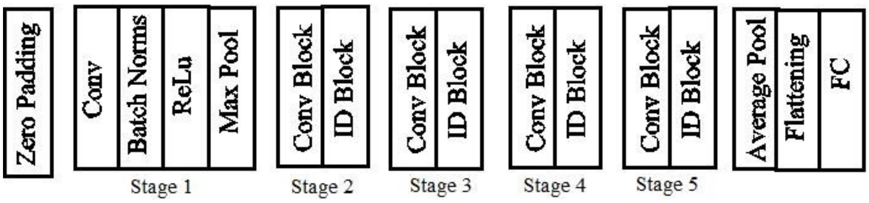


Figure 7. Resnet-50 Architecture

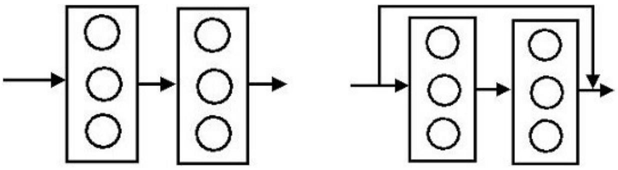


Figure 8(a). And 8(b). Without Skip Connection and skip connection

ResNet 50 has several layers, which are described further below.

Convolution layer

The convolution matrix $C_{x,y}$, shown in Equation 2, is calculated if x and y are two variables in matrix A.

$$C_{x,y} = \sum_u \sum_v a_{u,v} b_{x-u+1,y-v+1} \tag{2}$$

Where u and v cover all legal subscripts for $a_{u,v}$ and $b_{x-u+1,y-v+1}$.This layer is used to extract features from input images. It keeps the pixels' relationships together. An image matrix and filter have been used in this process.ResNet-50 is a popular deep convolutional neural network architecture that has 50 layers. It was introduced by Microsoft Research in 2015 and has achieved state-of-the-art performance on various object detection, image classification, and semantic segmentation techniques'. Here is the high-level architecture of ResNet-50:

RELU layer

The primary purpose of using the Rectified Linear Unit (RELU) layer is to increase the nonlinearity of a given input.

Pooling Layer

Pooling layers combine into a single neuron by combining each group of the preceding layer. This work uses maximum pooling to select the job and a maximum of 2*2 map element features.

Flatten layer

The data from the pooling layer is converted into a 1D continuous layer using the flattened layer.

Fully connected layer

The output of the convolution layer is likely flattened and sent to the output layer. A fully connected layer performs nonlinear feature combination learning. It is shown in Figure 9.

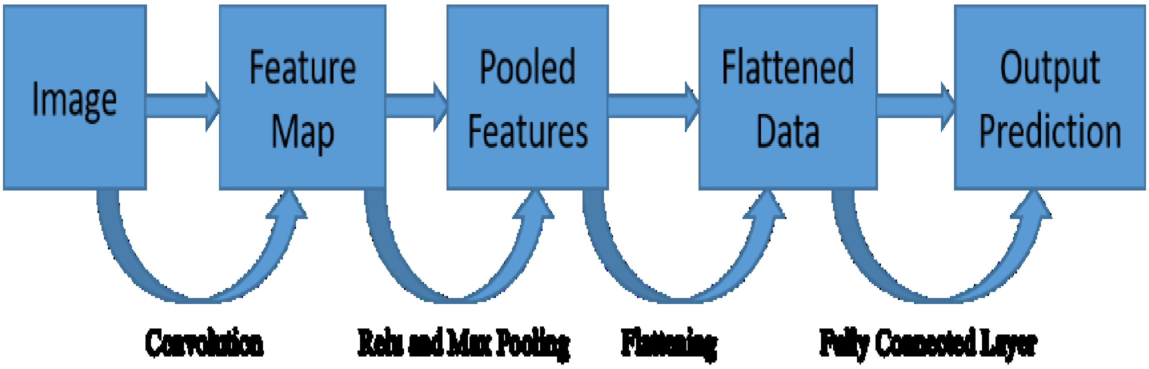


Figure 9. Fully Connected Layer

Softmax function

An activation function that can forecast problem distribution is the softmax function. Multiclass classification problems are usually solved by requiring membership in a class across more than two class labels.

Training and Validation

Residual functions trained by residual networks regarding the input layers are forced to learn unreferenced functions, and residual networks let these layers fit a residual mapping instead of assuming that every few layers directly provide a desired underlying mapping. The study demonstrates that a model has been easily optimised to improve the system's performance. In networks, existing blocks have been stacked on top of each other. In this work, researchers found that a residual network with several dense layers worked well for classifying images for analysis in leaf disease, which is essentially a hybrid residual-dense structure. ResNet50 depends on the convolution layer due to the absence of dense layers for identifying and classifying disease in leaves. Batch sizes of 40 and 8 x 8 pixels have been fixed. Six Fully connected layers are trained to be converted into new layers. Due to active gradients, previous layers will back propagate. For this purpose, the model weights have been updated using the Adam optimizer, which has been trained for 50 epochs.

Classification using CNN

Deep learning experiments include a significant portion of CNN-based image classification. Image classification and a deep learning phenomenon provide an image with a class and a label that identify convolutional neural networks (CNNs) and allow for reading 2D images in input, output, and hidden layers while using filters. Convolutional, ReLU, pooling, and fully connected layers are hidden layers. All these significantly affect the classification and processing of images in CNN, and the softmax function is used to identify the correct class by the probability distribution function. The negative value is considered positive for an exponential role for this purpose. A particular type of probability is shown in Equation 3, where N is the total number of classifications for particular diseases and Xi is the input value (Jain et al., 2019).

$$\sigma(\mathrm{X})_{\mathrm{i}} = \frac{\mathrm{e}^{\mathrm{X}_{\mathrm{i}}}}{\sum_{\mathrm{j}=1}^{\mathrm{N}} \mathrm{e}^{\mathrm{X}_{\mathrm{j}}}} \text{ for } \mathrm{i} = 1, \dots, \mathrm{N} \text{ and } \mathrm{X} = (\mathrm{X}_1, \dots\dots, \mathrm{X}_{\mathrm{N}}) \tag{3}$$

Results

The proposed deep convolutional neural network (DCNN) model is implemented in Python and utilizes the TensorFlow and PyTorch libraries. To evaluate the model's performance, the cucumber leaf dataset from the Plant Village repository is utilized. The classification process consists of two phases. In the initial stage, the model separates healthy and unhealthy leaves. Figure 10 compares different models based on quantitative metrics such as accuracy and F1 score to evaluate their performance. The first level of classification focuses on identifying unhealthy leaves, while the second level aims to detect specific diseases among three given types of diseases. To enhance the training process and improve the model's accuracy without acquiring additional datasets, a data augmentation technique is employed. This process involves adding more images to the dataset. During the model's training, back propagation of the gradient is utilized to update the model parameters in case of any loss occurrence. The ResNet-50, a deep convolutional neural network pre-trained on a large dataset, is employed for classifying images into different categories. The output of ResNet-50 is a probability distribution that represents the likelihood of an input image belonging to each class. The softmax function is applied to convert the ResNet-50's output into a probability distribution ranging between 0 and 1.By adjusting the input value provided to the softmax function, the output rate of ResNet-50 can be transformed into a probability distribution. A lower learning rate value results in slower training but generally leads to better results. In this experiment, the default learning rate of 0.001 from the PyTorch library's optimizer (Adam) was utilized. Equation 4 compares the log-probabilities for all classes with the target values. The predicted value with the highest probability is selected as the output. In this equation, 'p' represents the output values and 'q' represents the predicted values for a given model.

$$\log(p, q) = -\log q \tag{4}$$

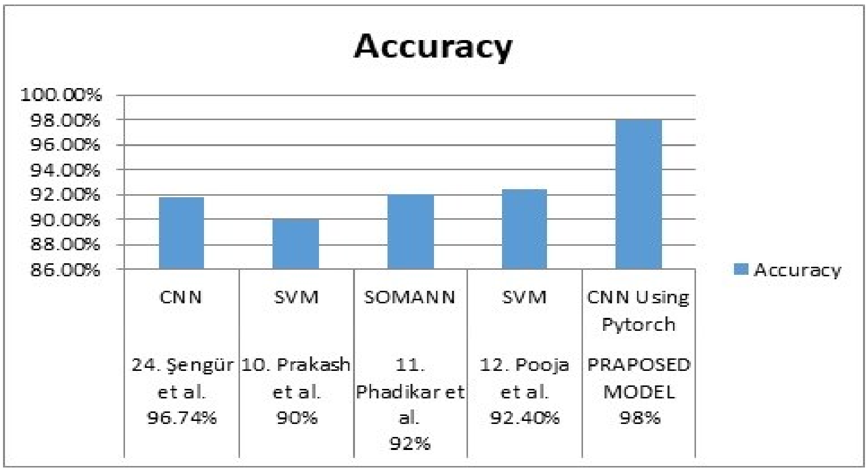


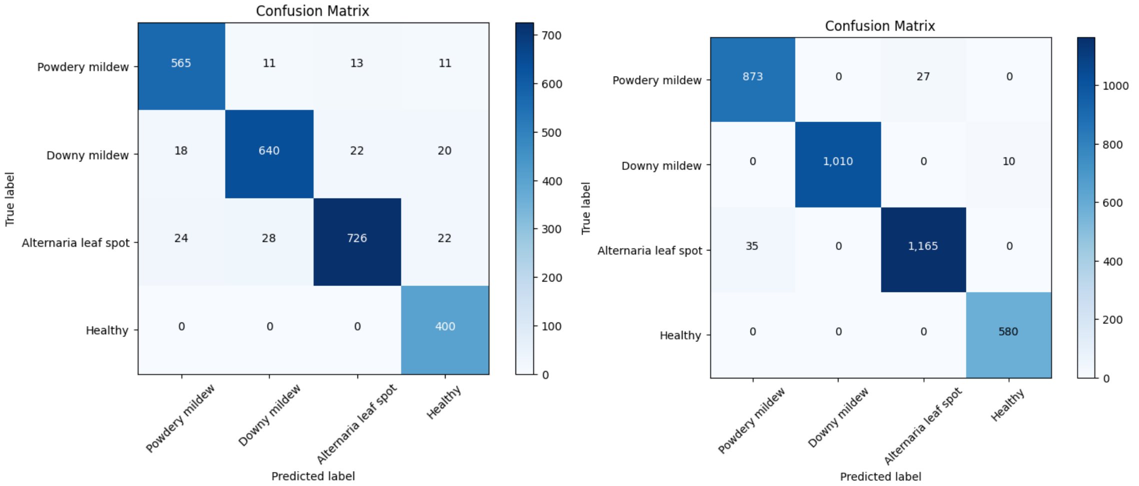
Figure 10. Comparison of different models and their accuracy

Table 5.1. Before data augmentation, the overall confusion matrix for all three classes

Leaf disease	Powdery Mildew	Downy Mildew	Alternaria leaf spot	Healthy	Accuracy
Powdery Mildew	565	11	13	11	94.2%
Downy Mildew	18	640	22	20	91.4%
Alternaria leaf spot	24	28	726	22	90.8%
Healthy	0	0	0	400	100%
Average					**94.1%**

Table 5.2. After data augmentation, the overall confusion matrix for all three classes

Leaf disease	Powdery Mildew	Downy Mildew	Alternaria leaf spot	Healthy	Accuracy
Powdery Mildew	580	6	8	6	96.7%
Downy Mildew	4	684	6	6	97.9%
Alternaria leaf spot	10	10	774	6	97.5%
Healthy	0	0	0	400	100%
Average					**98.0%**



True label \ Predicted label	Powdery mildew	Downy mildew	Alternaria leaf spot	Healthy
Powdery mildew	565	11	13	11
Downy mildew	18	640	22	20
Alternaria leaf spot	24	28	726	22
Healthy	0	0	0	400

True label \ Predicted label	Powdery mildew	Downy mildew	Alternaria leaf spot	Healthy
Powdery mildew	873	0	27	0
Downy mildew	0	1,010	0	10
Alternaria leaf spot	35	0	1,165	0
Healthy	0	0	0	580

Figure 11(a) and 11(b) show the confusion matrix before and after augmentation

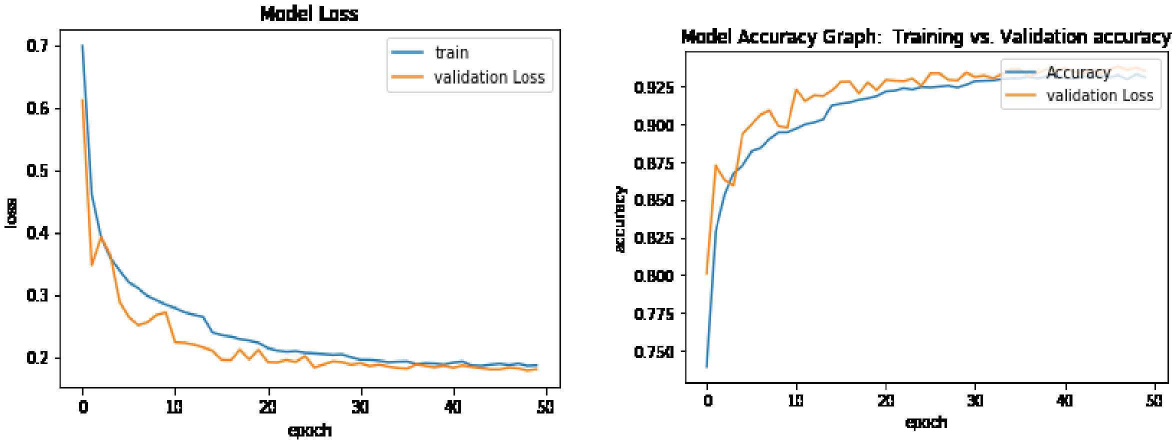


Figure 12(a) and 12(b) show the model accuracy and loss before augmentation

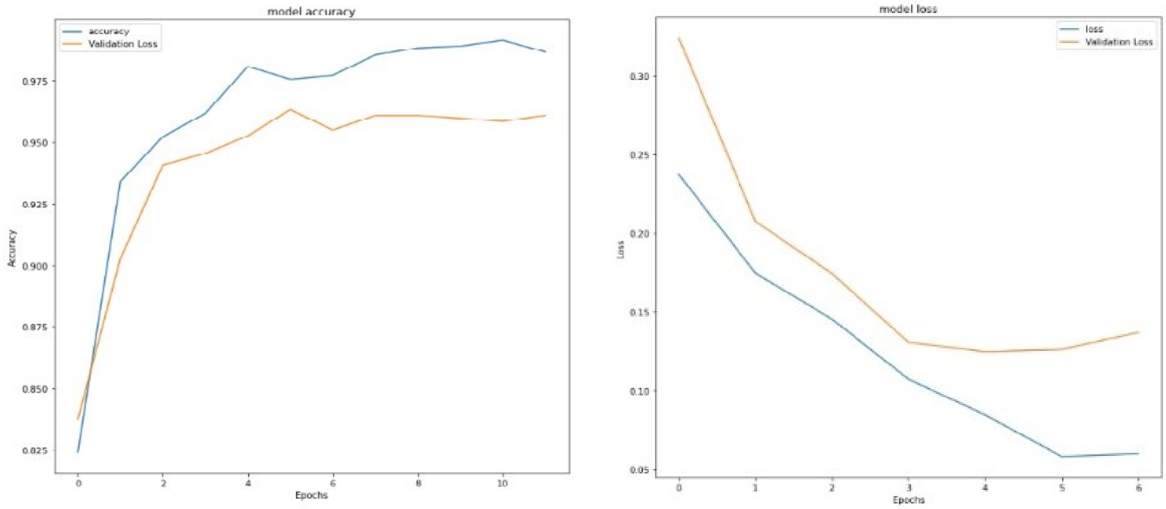
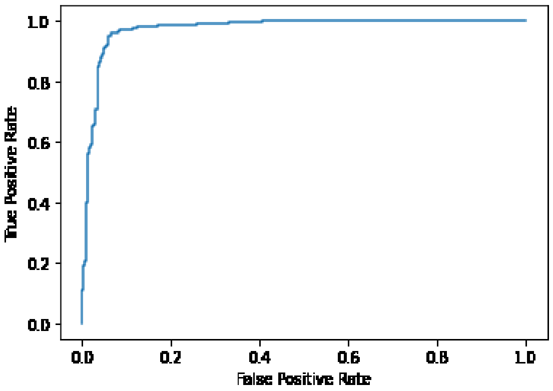


Figure 13(a) and 13(b) show the model accuracy and loss after augmentation

The quantity of epochs used to train the model is determined by the loss experienced after each iteration and validation. The loss has been predicted using log-likelihood by the negative NLLoss function indicated in Equation 4, where p is the output and q is the expected value. The confusion matrix of before and after data augmentation predicts the accuracy of cucumber leaf disease here. Tables 5.1 and 5.2 show the overall accuracy of various diseases such as powdery mildew, downy mildew, and Alternaria leaf spot based on augmentation applied before and after the proposed CDD model covered a large area and that the training model evaluated better accuracy and that the loss will be reduced. The confusion matrix is shown in Figures 11(a) and 11(b) before and after augmentation, The model accuracy and loss before augmentation are shown in Figures 12(a) and 12(b) while the model accuracy and loss after augmentation are shown in Figures 13(a) and 13(b). We can clearly see that after 30 epochs the loss becomes stable. ROC curve (Receiver Operating Characteristics curve, Figure 14(a) and Figure 14(b). Denotes a graphical representation of the model's performance before and after augmentation. The plot targeted true and false positives as part of a validation technique for evaluating the model's performance. As per ROC plot observation, the proposed model's curve indicated in Figure 14(b) performs better than the existing curve shown in Figure 14(a). This implies that the proposed model accurately predicts all significant diseases and has a larger area under the curve, indicating better classification.



,

Figure 14(a). ROC Plot before data augmentation

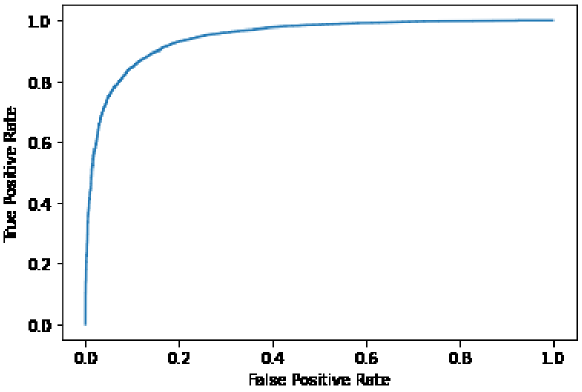


Figure 14(b). ROC Plot after data augmentation

Conclusion

Different machine learning algorithms, including support vector machines, self-organizing maps, and convolutional neural networks, have been utilized for the detection and classification of plant diseases. These algorithms have achieved accuracy rates of 90%, 92%, and 95%, respectively. In this work, a Deep-CNN-based disease detection model is employed to improve the performance of the ResNet50 model specifically for cucumber disease detection. The model incorporates data augmentation techniques using a diverse dataset. Transfer learning is employed to address overfitting issues and enhance the model's performance by adjusting its weights. The proposed model demonstrates an accuracy rate of 98%, surpassing the current method. The model's specifications include an Intel i7 processor, 16 GB of RAM, and a 250 GB SSD. However, it's crucial to consider the context and dataset of the task at hand. While a 98% accuracy rate is excellent for some tasks, it may not be sufficient for others. The proposed model is based on the ResNet50 architecture, utilizing 50 layers. Only the last layer of the model is modified after training, while the remaining layers remain unchanged. Furthermore, it is important to assess the model's performance on a separate test set. The extraction of deeper image features can be achieved by utilizing the Gabor filter for disease detection in various plants such as coffee, maize, pumpkin, and so on.

Acknowledgements

The fundamental part of this research was held at the Galgotia's university India.

Conflict of interest

The authors of this paper state that they do not have any competing financial interests or personal relationships that could have influenced their work. We would like to verify that there are no conflicts of interest related to this publication and that no significant financial support has been received that could have influenced the outcome of the research. This statement indicates that the authors have taken steps to ensure that their work is unbiased and free from any undue influence.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

References

Alshammari, H., Gasmi, K., Ben Ltaifa, I., Krichen, M., Ben Ammar, L., & Mahmood, M. A. (2022). Olive disease classification based on vision transformer and CNN models. Computational Intelligence and Neuroscience, 2022.

Atila, Ü. Uçar, M., Akyol, K., & Uçar, E. (2021). Plant leaf disease classification using EfficientNet deep learning model. *Ecological Informatics*, 61, 101182.

Bodhwani, V., Acharjya, D. P., & Bodhwani, U. (2019). Deep residual networks for plant identification. *Procedia Computer Science*, 152, 186-194.

Borhani, Y., Khoramdel, J., & Najafi, E. (2022). A deep learning based approach for automated plant disease classification using vision transformer. *Scientific Reports*, 12(1), 11554.

Ertam, F., & Aydın, G. (2017, October). Data classification with deep learning using Tensorflow. In 201

Hari, S. S., Sivakumar, M., Renuga, P., & Suriya, S. (2019, March). Detection of plant disease by leaf image using convolutional neural network. In 2019 International conference on vision towards emerging trends in communication and networking (ViTECoN) (pp. 1-5). IEEE.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).

International conference on computer science and engineering (UBMK) (pp. 755-758). IEEE. De Luna, R. G., Dadios, E. P., & Bandala, A. A. (2018, October). Automated image capturing system for deep learning-based tomato plant leaf disease detection and recognition. In TENCON 2018-2018 IEEE Region 10 Conference (pp. 1414-1419). IEEE.

Ismael, S. A. A., Mohammed, A., & Hefny, H. (2020). An enhanced deep learning approach for brain cancer MRI images classification using residual networks. Artificial intelligence in medicine, 102, 101779.

Jain, A., Awan, A. A., Anthony, Q., Subramoni, H., & Panda, D. K. D. (2019, September). Performance characterization of dnn training using tensorflow and pytorch on modern clusters. In 2019 IEEE International Conference on Cluster Computing (CLUSTER) (pp. 1-11). IEEE.

Jiang, P., Chen, Y., Liu, B., He, D., & Liang, C. (2019). Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks. IEEE Access, 7, 59069-59080.

Kaushik, M., Prakash, P., Ajay, R., & Veni, S. (2020, June). Tomato leaf disease detection using convolutional neural network with data augmentation. In 2020 5th International Conference on Communication and Electronics Systems (ICCES) (pp. 1125-1132). IEEE.

Khan, M. A., Akram, T., Sharif, M., Javed, K., Raza, M., & Saba, T. (2020). An automated system for cucumber leaf diseased spot detection and classification using improved saliency method and deep features selection. *Multimedia Tools and Applications*, 79, 18627-18656.

Liu, J., Yang, S., Cheng, Y., & Song, Z. (2018, November). Plant leaf classification based on deep learning. In 2018 Chinese Automation Congress (CAC) (pp. 3165-3169). IEEE.

Ma, J., Du, K., Zheng, F., Zhang, L., Gong, Z., & Sun, Z. (2018). A recognition method for cucumber diseases using leaf symptom images based on deep convolutional neural network. *Computers and electronics in agriculture*, 154, 18-24.

Manavalan, R. (2021). Automatic Detection of Fungal and Bacterial Plant Leaves Diseases: A Review. *Asian Basic and Applied Research Journal*, 133-143.

Manavalan, R. (2021). Cucumber Leaves Diseases Detection through Computational Approaches: A Review. *Asian Journal of Research in Biosciences*, 100-110.

Nashrullah, F. H., Suryani, E., Salamah, U., Prakisya, N. P. T., & Setyawan, S. (2021). Texture-Based Feature Extraction Using Gabor Filters to Detect Diseases of Tomato Leaves. *Revue* d'Intelligence Artificielle, 35(4).

Phadikar, S., & Sil, J. (2008, December). Rice disease identification using pattern recognition techniques. In 2008 11th International Conference on Computer and Information Technology (pp. 420-423). IEEE.

Pooja, V., Das, R., & Kanchana, V. (2017, April). Identification of plant leaf diseases using image processing techniques. In 2017 IEEE Technological Innovations in ICT for Agriculture and Rural Development (TIAR) (pp. 130-133). IEEE.

Prakash, R. M., Saraswathy, G. P., Ramalakshmi, G., Mangaleswari, K. H., & Kaviya, T. (2017, March). Detection of leaf diseases and classification using digital image processing. In 2017 international conference on innovations in information, embedded and communication systems (ICIIECS) (pp. 1-4). IEEE.

Rezende, E., Ruppert, G., Carvalho, T., Ramos, F., & De Geus, P. (2017, December). Malicious software classification using transfer learning of resnet-50 deep neural network. In 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA) (pp. 1011 1014). IEEE.

Sankaran, S., Mishra, A., Ehsani, R., & Davis, C. (2010). A review of advanced techniques for detecting plant diseases. *Computers and electronics in agriculture*, 72(1), 1-13.

Sannakki, S. S., Rajpurohit, V. S., Nargund, V. B., & Kulkarni, P. (2013, July). Diagnosis and classification of grape leaf diseases using neural networks. In 2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT) (pp. 1-5). IEEE.

Şengür, A., Guo, Y., Budak, Ü. & Vespa, L. J. (2017, September). A retinal vessel detection approach using convolution neural network. In 2017 International Artificial Intelligence and Data Processing Symposium (IDAP) (pp. 1-4). Ieee.

Shoaib, M., Hussain, T., Shah, B., Ullah, I., Shah, S. M., Ali, F., & Park, S. H. (2022). Deep learning based segmentation and classification of leaf images for detection of tomato plant disease. *Frontiers in Plant Science*, 13, 1031748.

Zhou, B., Xu, J., Zhao, J., Li, A., & Xia, Q. (2015, August). Research on cucumber downy mildew detection system based on SVM classification algorithm. In 3rd international conference on material, *mechanical and manufacturing engineering* (IC3ME 2015) (pp. 1681-1684). Atlantis Press.

Bibliographic information of this paper for citing:

Singh, Manish K. & Kumar, Avadhesh (2023). Cucumber Leaf Disease Detection and Classification Using a Deep Convolutional Neural Network. *Journal of Information Technology Management,* 15 (Special Issue), 94-110. https://doi.org/10.22059/jitm.2023.95248