Multi-Disciplinary Publishing Institute Pakistan
Vol. 01, Issue. 02

Submitted: 16 MAR 2025 Accepted: 13 OCT 2025 Published: 15 OCT 2025

# Citrus Leaves Disease Detection and Classification on Image Recognition Using Deep Convolutional Neural Networks: One Health Perspective

Original Article

**Dr. Khalid Mahmood**
Institute of Computing and Information Technology, Gomal University, Dera Ismail Khan, Khyber Pakhtunkhwa, Pakistan
Email: khalid@gu.edu.pk

**Dr. Ghulam Muhammad Kundi**
Health Informatics Department, College of Applied Medical Sciences, Qassim University, Buraydah, Kingdom of Saudi Arabia.
Email: g.muhammad@qu.edu.sa

**Dr. Yasir Hayat Mughal**
Health Informatics Department, College of Applied Medical Sciences, Qassim University, Buraydah, Kingdom of Saudi Arabia.
Email: y.hayat@qu.edu.sa

## Citation

Mahmood, K., Kundi, G.M., & Mughal, Y.H. (2025). Citrus leaves disease detection and classification on image recognition using deep convolutional neural networks: One health perspective. *Open Access Digital Management and Governance Review*, 1(2), 1-21.

**WEBSITE:** www.mdpip.com
**PUBLISHER:** MDPIP

## Abstract

Agriculture production is a crucial economic backbone for any country and is vital in meeting human food needs. At the same time, plant disease poses a significant threat to this sector, leading to decreased yields and heavy losses. Automated systems for disease detection and classification can aid in combating this issue and promoting growth and development. In recent years, deep learning approaches have demonstrated promising results in various artificial intelligence tasks, specifically in the Smart Agriculture domain. Smart Agriculture applications include Plant disease detection, water and soil management, crop distribution, crop cultivation, fruit counting, and yield prediction. This paper presents an integrated and enhanced approach for detecting citrus leaf disease detection using a deep convolutional neural network. The proposed model can distinguish healthy citrus leaves from seven common diseases: bacterial spot, black spot, canker, citrus powdery mildew, greening, melanose, and health. The proposed model extracts the complementary features by incorporating multiple hidden layers and using data augmentation for improved image recognition and classification. The proposed model is tested against other deep learning models on the citrus and Plant Village dataset and outperformed previous studies in various performance measurement metrics. With a test accuracy of 97.66%, our model serves as a reliable tool for citrus plant disease detection.

**Keywords:** Citrus Leaves, Disease Detection, Disease Classification, Image Recognition, Deep Convolutional Neural Networks, Plant Diseases, Agriculture, Machine Learning.

## Introduction

In Pakistan, sustainable agricultural expansion is essential for rural development and food security. It generates 22.7% of the GDP, employs about 37.4% of the labor force, manages the rural environment, and acts as an environmental buffer to safeguard and improve the climate-resilient ecosystem and production (Naqvi, Wang, Malik, Umar, Hasnain, Sohail, *et al.*, 2022). The agriculture industry had a tremendous increase of 4.40 percent between 2021 and 2022, exceeding the goal of 3.5% and the growth of 3.48% the previous year. Second, only to bananas in terms of global fruit output, citrus is grown on more than 200,400 hectares and produces 158 million tons of fruit annually. China produces the most citrus fruit on the planet, with 35.2% of all fruit production, or 2.4 million tons, produced in the citrus industry in 2014–2015. Mandarins (Kinnow), oranges, grapefruit, lemons, and limes are citrus fruits, and mandarins (Kinnow) are particularly significant to Pakistan (Siddique & Garnevska, 2018). There are more than 140 countries in the world where citrus is grown. Brazil, China, Mexico, the United States, Spain, and India are the world's top citrus-producing nations. Pakistan is one of the top ten countries in the world for citrus production (Source: FAOSTAT). With a 95% market share, Kinnow (Citrus reticulate) is Pakistan's most common citrus species. Punjab portions 94% and 96% in the area and production of citrus, respectively (Catara & Polizzi, 1999). Citrus has a high nutritional value and is a rich source of vitamins C and E, sugar, organic acids, amino acids, and minerals, including calcium and magnesium. The average yield in Pakistan is 2.36million tons, spread across 1,000 hectares, with an annual export of 282,000 tons, selling for 7,313 million rupees (Dandurand &. Menge, 1992). However, diseases are an important factor that restricts citrus growth and development. The disease affects different parts of the citrus plants, including the stem, leaves, fruits, and branches. The traditional methods for disease detection are manual and rely on human expertise, observation, and judgment.

There are problems of inaccurate identification and low efficiency, researcher user image processing, computer vision, and different machine learning and deep learning techniques to detect plant diseases. Citrus fruit plants are very vulnerable to various infections and bacterial diseases, including Bacterial Spot, Black Spot, Canker, Citrus Powdery Mildew, Greening, and Melanose; they represent a persistent threat to citrus farming and have a significant economic impact on all citrus-growing regions worldwide (Mendonça, Zambolim, & Badel, 2017). Citrus trees can get the highly contagious canker, typically found on the leaves or fruit. In recent years, deep learning techniques have emerged as a powerful tool for disease detection in various fields, including agriculture. Different Machine learning models like SVM (Kour & Arora, 2019), Random forest (Wójtowicz, Piekarczyk, Czernecki, & Ratajkiewicz, 2021), Ensemble, and methods (Hu, Yin, Wan, Zhang, & Fang, 2020) use image processing to obtain disease feat. However, these machine learning models use manual and hand-crafted features, which are highly subjective. In contrast, on the other hand, deep learning models, specifically convolutional neural networks (CNNs), have shown promising results in detecting and classifying various citrus plant diseases (Liaqat, Hassan, Shoaib, Khurshid, & Shamseldin, 2022; Sultana, Sufian, & Dutta, 2018). This study proposes a deep learning-based approach for detecting citrus plant leaf disease using CNNs. The proposed model distinguishes healthy citrus leaves from common diseases, including bacterial spots, blackspots, canker, citrus powdery mildew, greening, melanose, and health. The model incorporates multiple hidden layers and data augmentation techniques to extract relevant features and improve image recognition and classification accuracy. The proposed model is tested on a citrus leaves dataset and evaluated against other deep-learning models. The research study results are expected to provide a reliable and efficient tool for citrus plant disease detection at early stages and contribute to the sustainable expansion of the agriculture industry in Pakistan.

## Literature Review

Plant diseases can significantly impact food safety and agricultural product output. Many automated systems developed for plant disease detection have been based on digital images, allowing for the swift implementation of algorithms. The challenges associated with the autonomous identification of plant illness have been addressed using traditional machine learning techniques such as Support Vector Machines (SVMs), Multilayer Perceptron Neural Networks, and Decision Trees. Varshney *et al.* (2022) propose a novel DL method for leaf plant disease detection utilizing a transfer learning methodology, where a Convolutional Neural Network (CNN) is utilized as a feature extractor, and SVM is employed for classification. The proposed model is evaluated using a benchmark dataset called PlantVillage. Results indicate that the proposed model outperforms previous work, achieving an accuracy of 88.77% on a training dataset. Bharateet *et al*. (2017) provide an overview of various image-processing techniques used for plant disease detection. These methods include research on disease detection in various plants such as apples, grapes, peppers, pomegranates, and tomatoes. Researchers have developed various methods for recognizing and classifying different types of fruit diseases, such as the YCR color model, the HSV color model (Dhanesha & Shrinivasa, 2019), and VGG16. Dhanesha *et al*. (2018) used the YCR color model to segment the Arecanut bunches, using volumetric overlap error and dice similarity coefficient to evaluate the degree of similarity between the input image and the ground truth. Ghosal and Sarkar (2020) developed a VGG16 model (Qassim, Verma, & Feinzimer, 2018) that incorporates transfer learning to diagnose diseases affecting rice plants. The researchers trained this classification system with the help of four different image classes, and now VGG16 is accurate up to 92.4% of the time. Kumar *et al.* (2020) developed a system that can recognize illnesses based on the appearance of the leaves of coffee plants, using radial basis function neural networks, fuzzy logic-based expert systems, transfer learning techniques, and CNN with data augmentation. Coulibaly *et al*. (2019) use a GG16 model trained through transfer learning to detect diseases in millet crops; 124 images of leaves are separated into two categories: those with mild infections and those that appeared healthy. The level of precision achieved by the VGG16 model is 95%.

### *Deep Learning Models for Plant Disease Detection*

Deep learning models have been effectively used to detect plant diseases by recognizing patterns and features in images of both healthy and diseased plants through training on large datasets. Convolutional Neural Networks (CNNs) are a specific type of deep learning model that has shown success in classifying images of plant leaves, and RetinaNet (Wang, Wang, Zhang, Dong, & Wei, 2019) is another example of a deep learning model that can detect multiple diseases in a single image, which can help farmers to prevent disease spread and increase crop yield.

Pavan *et al.* (2021) present a comprehensive review of convolutional neural networks (CNNs) for plant disease detection from images. The authors discuss various CNN architectures, such as AlexNet, VGG, and ResNet, and their performance on different plant species and diseases. They also discuss the challenges and limitations of using CNNs for plant disease detection and the current state of the art. Saleem *et al*. (2020) employed DL meta-architectures and TensorFlow object detection framework to tackle the intricate tasks of identifying the location and categorizing diseases in plant leaves. This led to high accuracy in recognizing different types of damaged and healthy leaves, achieving a mean average precision of 73.07%. The methodology can be utilized in other areas of agriculture and has the potential for future utilization in the real-time detection of plant diseases in controlled and non-controlled environments. Tan *et al*. (2020) introduce EfficientDet, a family of convolutional neural network (CNN) models designed to be accurate and efficient for object detection tasks. The authors present results of using EfficientDet models for plant disease detection, showing that they achieve comparable or higher accuracy than existing methods while being more scalable and efficient.

### *Image Datasets for Plant Disease Detection*

Image datasets for plant disease detection typically consist of many images of both healthy and diseased plants. They can include a wide variety of plant species and disease types. These datasets train deep-learning models for plant disease detection and are critical for developing accurate and reliable algorithms. They are also used to evaluate and compare the performance of different models on a common benchmark. Table 1 lists various datasets that are used for plant disease detection. The plant species and disease categories also vary among the datasets. Some datasets, such as PlantVillage and Embrapa Dataset, have specific information about the number of plant species and disease categories included. Other datasets, such as IPM and Bing, do not provide this information. Additionally, the backgrounds of the images in the datasets vary, with some taken in lab conditions with fixed backgrounds, while others are taken in real-field conditions.

**Table 1**
*Plant Diseases Datasets*

| Dataset Name | Institution | Number of Images | Plant Species | Disease Categories | Background |
|---|---|---|---|---|---|
| PlantVillage (Hughes & Salathe, 2015) | Penn State University | 54,305 | 14 | 38 | Lab conditions with a fixed background |
| IPM[24] | N/A | 119 | N/A | N/A | Fixed and background conditions |
| PlantVillage (extended) (Ferentinos, 2018) | N/A | 87,848 | 25 | 58 | Infield and laboratory conditions |
| Embrapa Dataset (Barbedo, Koenigkan, Halfeld-Vieira, *et al.,* 2018) | Embrapa Agriculture Institute | 46,513 | 18 | 93 | In-field |
| Strawberry Dataset (Nie, L. Wang, Ding, &M. Xu, 2019) | N/A | 3531 | N/A | 4 | N/A |
| Rice dataset (Rahman, Arko, Ali, 2020) | Bangladesh Rice Research Institute | 1426 | N/A | N/A | Real-field conditions |
| Apple Dataset (Thapa, Zhang, Snavely, 2020) | N/A | 3651 | N/A | 4 | N/A |
| Maize Dataset (Chen, Chen, Zhang, 2020) | Fujian Institute of Subtropical Botany | 481 | 1 | 4 | In-field |
| Rice Dataset | Fujian Institute of Subtropical Botany | 560 | N/A | 5 | Laboratory and in-field |
| PlantDoc (Wang, Du, Wu, 2021) | | 2598 | N/A | 17 | Field |
| Turkey-PlantDataset (Kumar, Belhumeur, Biswas, 2012) | Agricultural Faculty of Bingol and Inonu Universities | 4447 | N/A | 15 | N/A |

Other datasets on plant diseases are also available in the literature, not limited to (Falaschetti, Manoni, Di Leo, Pau, Tomaselli, & Turchetti, 2022; Li & Yang, 2020).

***Transfer Learning for Plant Disease Detection***

Transfer learning is a technique used in deep learning to utilize pre-trained models on related tasks to improve performance and reduce training time for a new task. It is especially useful when data is limited. It allows for better generalization and helps avoid overfitting by fine-tuning the pre-trained models on large datasets for the new task (Tan, Sun, Kong, Zhang, Yang, & Liu, 2018). Sagar *et al.* (2021) address the problem of multi-class classification and show how neural networks can be used for plant disease recognition in the context of image classification. The study compares the performance of transfer learning approaches for plant disease detection, including fine-tuning, feature extraction, and training on the publicly available Plant Village dataset, which has 38 classes of diseases. Five different architectures are compared, including VGG16, ResNet50, InceptionV3, InceptionResNet, and DenseNet169 as the backbones for the network; ResNet50 outperforms other networks on the test set. Various metrics such as accuracy, precision, recall, F1 score, and class-wise confusion metrics are used for evaluation. The model achieved the best results using ResNet50 with an accuracy of 0.982, precision of 0.94, recall of 0,.94, and F1 score of 0.94.

Other studies that use transfer learning for plant disease detection can be found at Udawant and Srinath (2022), Sharma, Nath, Sharma, Kumar, and Chaudhary (2022) Vallabhajosyula, Sistla, and Kolli (2022). The study proposed a deep neural network model for automatically detecting citrus fruit and leaves diseases. The model achieved high accuracy compared to baseline machine learning models and outperformed a four-convolutional-layer CNN model. Future work involves exploring different datasets, using larger datasets, and investigating other deep learning architectures and techniques (Khattak, Asghar, Batool, Asghar, Ullah, Al-Rakhami, *et al.,* 2021). The study successfully generated high-quality citrus disease and nutritional deficiency images using FastGAN2 and achieved high accuracy in classification. The method shows potential for aiding fruit farmers in disease identification. Further development includes extending the approach to other plant diseases and creating a user-friendly application or software (Dai, Guo, Li, Song, Lyu, Sun *et al*, 2023). The study used CNN models to conduct a hyperspectral imaging analysis of citrus fruit peels. The results demonstrated that training the CNN model with PCA-selected bands yielded higher classification accuracy than randomly selected bands. Moreover, using images with multiple fruit instances resulted in slightly better classification performance than using single fruit instances. The findings contribute to developing efficient and accurate citrus peel condition detection methods. Future work involves deploying the trained CNN model in real-time systems, exploring generalizability, optimizing model parameters, and expanding the research to other agricultural applications (Yadav, Burks, Frederick, Qin, Kim, & Ritenour, 2022).

The study proposes the MF-RANet network for citrus disease recognition. AMSR preprocessing improves accuracy. The ELU activation function is effective. MF-RANet outperforms other networks. Limitations and future work were identified (Yang, Liao, Zhao, Zhou, He, & Li, 2022). The authors Atila, Uçar, Akyol, and Uçar (2021) in this research study proposed an EfficientNet deep learning model for plant disease detection. This research study uses the PlantVillage dataset to investigate various diseases in plants using different deep learning models. All the models were trained on original and augmented datasets comprising 55,448 and 61,486 images. The results obtained from the EfficientNet B4 on the augmented dataset achieve an accuracy of 99.91% and 99.07% accuracy score on the original dataset with compared to other transfer learning and deep learning models. The reason behind the high accuracy score is due to 90% training data, whereas only 7% data is used for testing and 3% for validation purposes (Gayathri, Wise, Shamini, & Muthukumaran, 2020). Tea leaf disease was addressed in this paper using convolutional neural networks. Four diseases were identified and classified in this research study by applying a CNN with data augmentation and annotation. The authors achieve an accuracy of 94% for the red scab disease, and for the remaining three diseases, tea leaf blight, blight disease, and tea red leaf spot, an accuracy between 84 to 93% is achieved. A very limited dataset containing only 80 images is used for training and testing purposes. A lightweight ResNet (LW-ResNet) (Yu, Cheng, Li, Cai, & Bi, 2022) architecture based on ResNet-18 to detect and classify the six different types of apple leaf disease was proposed in this study. The dataset was acquired for Kaggle. The proposed architecture uses a multi-scale layer for feature extraction, the parameter memory is 92% less compared to the traditional ResNet-18 model, and provides low computational cost, low storage cost, and strong real-time performance. While comparing its efficiency in terms of precision, recall and F-1-measure with SqueezeNet and MobileNet, it outperforms the competitors with a score of 97.80%, 97.92% and 97.85% respectively.

***Handheld Device and Smartphone-Assisted Plant Disease Diagnosis***

One way to adapt deep learning models to run on handheld devices and smartphones is by reducing the model's complexity and size. This can be achieved through techniques such as model compression and pruning, which remove redundant and unnecessary parameters from the model. Another way is to use specialized deep learning architectures designed for mobile devices, such as MobileNet, which use depth-wise separable convolutions to reduce computational complexity while maintaining high accuracy. Additionally, quantization of the model can also be done to run on specific hardware of mobile devices. Elhassouny *et al*. (2019) proposed an efficient smart mobile application model inspired by the MobileNet CNN model, which can recognize the ten most common types of tomato leaf disease using 7176 images of tomato leaves in the dataset. Andrianto *et al.* (2020) propose a deep learning-based rice disease detection system that consists of a machine learning application on a cloud server and an application on a smartphone. The system uses the VGG16 architecture and has a training accuracy of 100% and a test accuracy of 60%. The test accuracy can be improved by adding more data and increasing the dataset's quality. This system aims to improve the control of rice plant disease and maximize yields. Valdoria *et al.* (2019) emphasize the detection of common diseases on terrestrial plants in the Philippines using image processing and deep-learning neural networks. Android-based smartphones were used to capture images of the terrestrial plant to detect the plant's disease; at the same time, a deep learning neural network algorithm was utilized to distinguish the disease of terrestrial plants.

The results were trained using classification models that could identify the diseases at a certain rate and accuracy, considering the number of images used. Diah *et al.* (2021) use a deep learning model based on a convolutional neural network (CNN) architecture to detect and classify tomato plant diseases using information from plant leaves such as color, texture, and shape. The model achieved a 95.8% validation and training accuracy rate for the classification results. The proposed OplusVNet network effectively classifies citrus disease images, outperforming other networks on small and unbalanced datasets. Future work includes mobile application development and agricultural automation (Yang, Teng, Dong, Lin, Chen, & Wang, 2022).

### *Key Findings from the Literature Review*

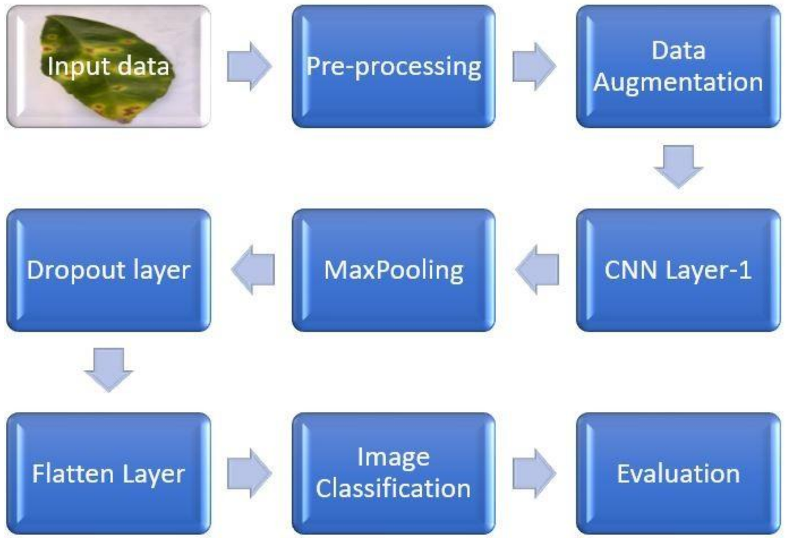
Deep learning has the potential to greatly impact plant disease detection and the future of agricultural management. Deep learning techniques, such as convolutional neural networks (CNNs), have achieved high accuracy in detecting plant diseases, often surpassing traditional image processing methods (Kaushik, Prakash, Ajay, & Veni, 2020). Additionally, deep learning models can be trained to detect diseases using IoT-based systems using edge devices for farmers with rapid and reliable information to make decisions about treatment and management (Rumy, Hossain, Jahan, & Tanvin, 2021). One potential impact of deep learning in plant disease detection is the ability to improve crop yields and reduce crop losses. Early detection of plant diseases can enable farmers to take preventative measures, such as applying pesticides or isolating infected plants, to reduce the spread of disease and minimize damage (Ferentinos, 2018). This could increase crop yields and improve food security, particularly in regions where crop diseases are prevalent.

Another potential impact of deep learning in plant disease detection is the ability to use resources more efficiently. Deep learning models can analyze large amounts of data, including images and sensor data, to detect plant diseases quickly and accurately (Pandian, Kumar, Geman, Hnatiuc, Arif, & Kanchanadevi, 2022). This can enable farmers to focus their efforts on specific areas of a field or specific plants rather than treating the entire field, reducing the resources required for disease management. Moreover, deep learning can improve the precision of plant disease detection and reduce the number of false positives and negatives caused by traditional methods (Saheb, Narayanan, & Rao, 2022). Using deep learning models in crop management can help farmers to make more informed decisions on disease management and reduce the number of chemical inputs used in agriculture. In summary, deep learning has the potential to greatly impact plant disease detection and the future of agricultural management by enabling farmers to detect diseases, improve crop yields, and make more efficient use of resources quickly and accurately.

## Methods and Materialsials

This research study proposes a Deep Convolutional Neural Network (DCNN). The proposed system is elaborated with a flowchart depicted in Figure 1.

**Figure 1**
*Workflow Model for the Citrus Leaf Disease Detection*




***Dataset Acquisition and Splitting***

Datasets are required at all steps, from training to algorithm evaluation during the image analysis study. A total of 2489 images were collected from the citrus dataset (Ali, Lali, Nawaz, Sharif, & Saleem, 2017), the self-collected dataset, and the Plat Village dataset (Hughes & Salathe, 2015). The images are divided into infected and healthy images grouped into seven categories: Bacterial spot, Black spot, Canker, Citrus Powdery Mildew, Greening, Melanose, and healthy images shown in Table 2.

**Table 2**
*Different Categories of Plants*

| Category | Disease | Image Count (Training) | Image Count (Testing) |
|---|---|---|---|
| Leaf | Bacterial Spot | 88 | 58 |
| | Black Spot | 471 | 61 |
| | Canker | 513 | 147 |
| | Citrus Powdery Mildew | 35 | 14 |
| | Greening | 684 | 121 |
| | Melanose | 65 | 31 |
| | Healthy | 128 | 76 |
| | Total | 1984 | 505 |
| | Grand Total | **2489** | |

This table provides information about the number of images of different categories of plant diseases and healthy plants used for training and testing using a machine learning model. Plants belonging to various categories are shown.

The dataset is divided into three parts: (i) training data, (ii) test data, and (iii) validation data using the Python split folders library. Intel CORE i7, a 7th-generation, 64-bit operating system with 8 GB RAM, is used for experiments. The Keras package, TensorFlow, and Anaconda Environment are utilized to run and execute the proposed DCNN model.

### *Train Data*

A DCNN model is developed using 75% of the training data. This data is used to train our model. The input and the anticipated outcomes are both included in the training data.

### *Test Data*

25% of the initial dataset is used for evaluation, which makes up the testing data. After the model has finished training, the test data forecasts its behavior.

### Validation Data

To minimize the underfitting and overfitting phenomena of the model (Singh, Rani, & Mahajan, 2018), which usually occur during the testing phase, when a machine behaves very well during the training phase, but its performance degrades during the testing phase with new data. The final model is then tested on the test dataset to evaluate its generalization ability and estimate its performance on unseen data. It is important to note that the validation dataset represents the unseen data the model encounters in practice; otherwise, it may perform poorly when used on real-world data.

### Overview of the Proposed Approach

The following modules make up the suggested technique: (i) pre-processing input image dataset, (ii) Data Augmentation, (iii) CNN layer-1, (iv) MaxPooling Layer-1, (v) CNN layer-2, (iv) MaxPooling Layer-2 (v) Dropout Layer, (vi) Flatten layer, and (vii) Classification Layer.

**Figure 2**
*The DCNN Model*

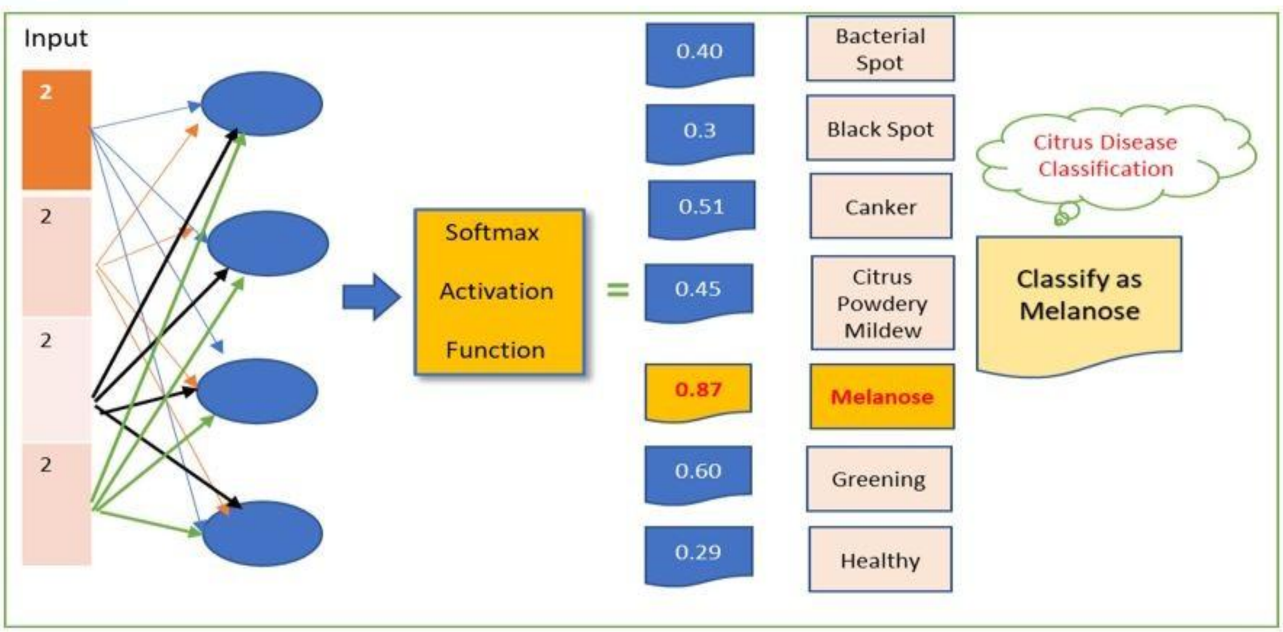


Figure 2 depicts the DCNN model module consisting of several layers stacked on each other. The input image is passed through a series of convolutional layers that apply filters to extract features from the image, max pooling layers to reduce spatial resolution and increase robustness, a flattened layer to convert the image into a one-dimensional array, and a dropout layer to reduce overfitting. A fully connected layer generates the model output and is the probability of each class, with the highest probability representing the predicted class. The DCNN model extracts features from the input image and uses them to classify the image into one of the predefined classes.

***Pre-processing of the Input Image:***

The input image is pre-processed in this module using the Keras image Data Generator module (Salamon & Bello, 2017). Image scaling, normalization, and data normalization functions are applied in this stage to normalize and scale the image repository for further processing.

***Data Augmentation***

Data augmentation aims to help better image recognition and reduce overfitting problems (Taylor & Nitschke, 2018). It helps with better data visualization by artificially expanding datasets using different options like Image Zoom, Rotation, Flipping, and Blur.

***CNN Layer-1***

The objective of this layer is to obtain a feature map of the input image. This objective is achieved using a convolution operation on the input image (Albawi, Mohammed, & Al-Zawi, 2017).

**MaxPooling of Layer-1**

The features obtained from CNN Layer-1 are reduced in size when transmitted to this layer; different pooling options can be chosen for feature selection. This layer reduces the noise and variations (Giusti, Cireşan, Masci, Gambardella, & Schmidhuber, 2013).

***CNN Layer-2***

This layer works similarly to Layer-1 of CNN, but it gets low-level features while Layer-1 gets high-level features.

***MaxPooling of Layer-2***

This layer works similarly to MaxPooling Layer-1, with the aim of dimensionality reduction.

***Dropout Layer***

To avoid overfitting the data in the testing phase dropout layer is used (Baldi & Sadowski, 2013). In the proposed study, we have set the rate=0.2.

***Flatten Layer***

This layer converts the resultant two-dimensional arrays from pool feature maps into single continuous vectors.

### *Classification Layer*

The output obtained from the flattened layer is used to get one of the possible images for classification using the activation function. The leaf disease is inspected using the SoftMax Activation function (Liang, Wang, Lei, Liao, & Li, 2017).

## Experimental Results and Discussion

### *Dataset Description*

We used sample images from the benchmark databases to experiment. The image dataset has been collected from the Kaggle Citrus dataset[68], Plant Village dataset (Hughes & Salathe, 2015), and our amassed image library to analyze the proposed study. Citrus diseases, including Bacterial spot, Black spot, Canker, Citrus Powdery Mildew, Greening, Melanose, and healthy images, are detected and classified using these datasets shown in Fig. 3.

### *Preprocessing of the Input Image*

The system initiates by taking input images from the citrus leaf dataset. The input shape parameter is set to accept a three-dimensional matrix image. After input, the image is rescaled and normalized for further processing.

### *Data Augmentation*

Deep learning models for classification tasks require large amounts of data for accurate prediction. On the other hand, the need for updated image collections for citrus illnesses and the small number of image repositories hinder the automatic detection of diseases that affect citrus leaves. To avoid this problem, a data augmentation operation on the training set was performed to increase the data size and avoid the problem of overfitting (Shorten & Khoshgoftaar, 2019). The overfitting problem arises when the model performs very well on the training data, but its performance deteriorates on the test data. Data augmentation parameters, such as Random Rotation, Random Zoom, Random Flip, Random Contrast, and Random Mirroring employing principal component analysis, are applied to the datasets. The input images are then resized to a standard 256 x 256 dimensions to facilitate implementation and reduce processing time.

**Figure 3**

*The Citrus Leaves (infected and healthy images).*


Figure 3 shows a dataset of citrus leaf images divided into infected and healthy images. The images are grouped into seven categories, each representing a different citrus leaf disease: bacterial spot, black spot, canker, citrus powdery mildew, greening, melanose, and healthy images. The dataset trains and tests a DCNN model for citrus plant disease detection. The model is trained to classify the images into one of the seven disease categories based on the features extracted from the images.

**Figure 4**

*The figure shows the DCNN Architecture*

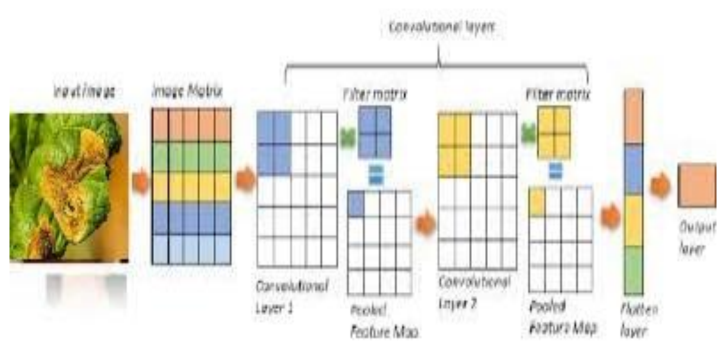



Figure 4 shows the DCNN Architecture. The images with the different diseases are fed as input. The DCNN architecture consists of Convolution, Pooling, and the SoftMax layer. The output layer presents the image with the highest probability.

***CNN Layer-1***

The fundamental units of CNN are the convolutional layers. The convolution Layer-1 performs the function of feature extraction. Convolution is a mathematical operation that filters an input image 1 concerning its dimensions $W_{in}\ X\ H_{in}$. Its hyperparameters filter dimensions $K\ x\ K$ size, stride S, and padding P. A feature map is generated as an output of repeated applications of the same filter, which indicates the locations and strength of detected features of an input image. The output size can be estimated by the following equation (1).

$$Mx = \frac{Ix-Kx}{Sx} \qquad My = \frac{Iy-Ky}{Sy} \qquad (1)$$

Where in the above equation, *(Mx, My)* represents the matrix, *(Ix, Iy)* represents the input size, *(Kx, Ky)* represents the Kernel size, and *(Sx, Sy)* represents the Stride concerning row and column, respectively.

***MaxPooling Layer-1***

The pooling layer performs the down-sampling (sub-sampling) operation on the output generated by the convolution layer to reduce its dimensionality. Particularly, the max and average pooling are special types of pooling in which the maximum and, in the latter case, the average value is considered for the pooling operation. The following equation determines the output by applying the Max operation on the input feature matrix.

$$M(r,c) = Max(Mx, My) \qquad (2)$$

***CNN Layer-2***

Layer 2 of CNN is responsible for extracting high-level features received from the previous MaxPooling layer. The second layer of CNN is comparable in function to the first convolution layer. Equation 1 is used to calculate the CNN layer 2.

***MaxPooling Layer-2***

The objective of this layer is to condense the matrix scale. This layer is comparable in function to the first pooling layer. MaxPooling layer-2 is obtained using equation 2.

***DropOut Layer***

Co-adoption is the phenomenon that occurs in fully connected layers due to the many neurons. Due to this co-adoption issue, multiple neurons extract similar hidden features from the input data because of identical neurons. It leads to wastage of machine resources and, more importantly, leads to overfitting in the test data due to duplicated extracted features (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014). So, in the training phase, we randomly drop out some fraction of the layer's neurons by zeroing out the values of neurons. In the proposed study, we have set the dropout rate to 0.2, meaning that we randomly drop out two neurons.

***Flatten Layer***

This layer receives the output from the MaxPooling layer-2 as input. This layer's responsibility is to transform the pooled feature matrix M output into a feature vector or column. The feature vector can be obtained from the feature matrix M by restructuring the function.

*Classification Layer*

The probabilities of different categories for citrus leaf images are calculated through the dense layer comprising different neurons by using the SoftMax function. The following equation obtains the resultant output.

$$z = \sum_{i}^{l} wixi + b \qquad (3)$$

Whereas x represents the input vector, w is a weight vector, and b represents the bias. Z is the activation function that generates the maximum probability value as the classification output.

**Applying a Citrus Leaf Disease Classification Model:**

This section elaborates on the process of citrus leaf disease recognition and classification using the proposed model. The suggested study's workflow model to identify the seven different forms of leaf diseases is shown in Figure 2. Bacterial spot, Black spot, Canker, Citrus Powdery Mildew, Greening, Melanose, and healthy images were identified and classified using the said approach.

*Data Acquisition of the Input Image*

The first step is to collect the input images dataset. These images are converted into a pixel with a dimensions matrix (256,256,3), with height, width, and RGB color combination ratio, respectively. These images are rescaled and normalized for further processing.

*Data Augmentation*

The goal of data augmentation is to supplement existing data to improve model performance and overcome the overfitting problem. The image dataset size can be increased for better training by applying data augmentation hyperparameters.

*Convolutional Layer-1*

This layer is responsible for high-level feature extraction. The input images are transformed into feature maps using the filter and padding processes. The process is described in section 4.4.

*MaxPooling Layer-1*

The MaxPooling layer performs the down-sampling operation on the feature matrix obtained from the convolution layer. It results in dimensionality reduction of the feature matrix.

*Convolution Layer-2*

To extract the feature matrix, this layer operates identically to the first convolution layer. It gets input from the previous MaxPooling layer-1. The pooling layer condenses the image feature matrix.

*MaxPooling Layer-2*

The second MaxPooling layer's goal is to decrease the dimension size to make features more easily discernible.

***Dropout Layer***

The dropout layer drops the identical neuron with similar weights to reduce the burden in the training phase.

***Flatten Layer***

The flattened layer performs the job of converting the pooled feature matrix into a single-dimensional feature vector.

***Classification Layer***

The classification layer calculates the probabilities of the seven distinct citrus leaf diseases. The output probability ranges between 0 and 1. We obtain the following probabilities of each citrus leaf disease by applying the SoftMax function with the score: Bacterial spot (0.40), Black spot (0.3), Canker (0.51), Citrus Powdery (0.45), Mildew (0.60), Greening (0.27), Melanose (0.87), and healthy (0.29). Using the above probability computation, the Melanose gets the higher probability, i.e., 0.87, so the Melanose picture is the resultant output.

## Result and Discussion

This part assesses and displays the results and conclusions attained via various experiments and empirical settings to validate our suggested study. The Citrus detection function is shown in Algorithm 1. Establishing Parameters for the CNN Model for Citrus Disease Recognition and Classification is shown in Table 3.

**Table 3**

*Parameters used for DCNN*

| Parameter | Value |
|---|---|
| Image Dimensions | 256 x 256 |
| # of layers in CNN | 2..4 |
| # of MaxPool Layers | 2..3 |
| # of Filters | 64,32,16 |
| Dimension of Filters | 2,3 |
| Activation Functions | SoftMax, ReLU |
| # of epochs | 10..50 |
| Batch size | 64 |

**Algorithm 1** CNN Workflow Model

**Input:** train
**Output:** test
**Data:** Training/Testing set *x*
**Function** CNN Workflow Model (*train, test*)**:**
**Import** Image Data Generator from tensorflow.keras.preprocessing.image
**Import** ImageDataGenerator from tensorflow.keras.preprocessing.image
**Define** data_augmentation as a sequence of RandomContrast and RandomZoom layers
**Define** model as a Sequential model
**Add** data_augmentation to the model
**Add** a Conv2D layer with 64 filters, 3x3 kernel, relu activation, and input shape of(256,256,3) to model
**Add** a MaxPool2D layer with 2x2 pool size to the model
**Add** a Conv2D layer with 32 filters, 3x3 kernel, and relu activation to the model
**Add** a MaxPool2D layer with 2x2 pool size to the model
**Add** a Flatten layer to the model

**Add** a Dropout layer with rate 0.2 to the model
**Add** a Dense layer with 7 units and softmax activation to the model
**Compile** model with loss of 'categorical_crossentropy', optimizer of 'adam', and metricof 'accuracy'
**Define** train_dataset as the flow of images from the directory **Citrus_Dataset**/Processed_output/train/' with target size of (256,256), batchsize of 64, and class mode of 'categorical'
**Define** validation_dataset as the flow of images from the directory **Define**'Citrus_Dataset/Processed_output/val/' with target size of
**Define** Fit the model with train_dataset, validation_data as
**Add** validation_dataset, steps_per_epoch as 3, and 30 epochs
**CNNWorkflowModel**

To gauge the effectiveness of the suggested task, we used various ratios of training and testing samples. This study used a 75-25 split ratio for evaluation purposes. These variations in the train-test samples provide a way to check the accuracy of the model, because as the training sample grows, its efficiency also increases. The precision (Buckland & Gey, 1994), refers to the number of accurately detected positive occurrences and measures the accuracy ofpositive instances, and can be calculated using equation (1) whereas in memory the TP reflects the actual positive cases that are precisely identified, and FN indicates the false negatives or the number of positive cases that are incorrectly named as negatives.

$$Precision = \frac{TP+TN}{TP} \quad (4)$$

The recall (Gillund & Shiffrin, 1984) represents the ratio of negative instances and refers to the number of negative instances that were wrongly marked as positive. The TN represents the quantities ofcases or actual negatives that are negative and identified as true, and FP represents the quantities that were actual negatives, but were wrongly marked as positive. It is represented in equation (5).

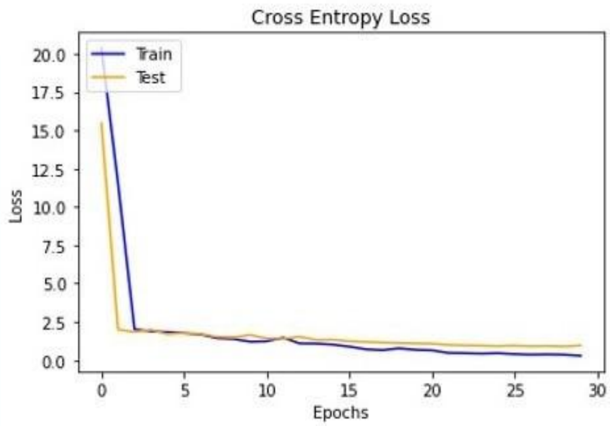
$$Recall = \frac{TN+FP}{TN} \quad (5)$$

The accuracy (Swets, 1988), represents the joint distribution of precision and recall. The accuracyreturns the overall performance of the model. It is represented in an equation.

$$Accuracy = \frac{TN+FP+TP+TN}{TN+TP} \quad (6)$$

We set different experiment settings for DCNN, with various epoch settings. In the training phase for 30 epochs, we achieved a 93.76% accuracy score, and for 50 epochs, we got a 95.80% accuracy score. and for 100 epochs, we got 97.66% highest accuracy score. The proposed study achieves an accuracy of 97.66% score, with an increase of 12% accuracy score with comparison to (Liu, Xiang, Qin, Ma, Zhang, & Xiong, 2021) and an increase of 4.16% with comparison to (Elaraby, Hamdy, Alanazi, 2022), respectively. Results are shown in Fig. 4.

**Figure 5**
*On the left: For cross cross-entropy loss graph, it can be seen that with the increasing number of epochs, the training/testing error decreases. On the Right: For the accuracy graph, it is clear that the accuracy of the DL model increases with more iterations.*



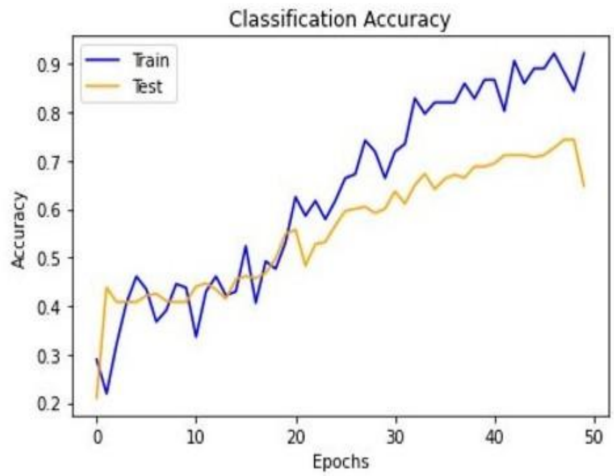




**Table 4**
*DCNN Score with different Settings.*

| Epochs | Recall | Precision | F-measure | Accuracy |
|---|---|---|---|---|
| DCNN (30) | 91.74 | 90.14 | 92.15 | 93.76 % |
| DCNN (50) | 93.74 | 94.50 | 95.00 | 95.80 % |
| DCNN (100) | 95.76 | 95.99 | 96.56 | 97.66 % |

**Table 5**
*Comparative Analysis with other Studies.*

| Baseline Study | Technique | Accuracy Score |
|---|---|---|
| (Liu, Xiang, Qin, Ma, Zhang, & Xiong, 2021) | Image recognition of citrus diseases using CNN | 87.28% |
| (Elaraby, Hamdy, Alanazi, 2022) | Classification of citrus diseases using CNN + AlexNet | 93.5 % |
| (Luaibi, Salman, & Miry, 2021) | Detection of citrus leaf diseases using CNN | 91.75 % |
| **Proposed Study** | Citrus leaves disease detection and classification using a DCNN approach with data augmentation | 97.66% |

**Threats to Validity**

The following shortcomings are observed for the suggested study:

1. Dataset Size: Due to the limited dataset size, a diverse dataset will further improve the accuracy of the system.
2. Limited classes: The model may not generalize well to other classes if the number of classes is limited in the training dataset.
3. Limited images: The model may not generalize well if the number of images per class is limited in the training dataset.
4. Environmental conditions: the model performance may vary depending on the conditions under which the images are taken (e.g., lighting, temperature, humidity).

## Conclusion

Deep learning methods such as convolutional neural networks (CNNs) are highly effective in detecting plant diseases with high accuracy rates. The suggested DCNN model is capable of classifying diseases that affect citrus leaves. It distinguishes the healthy leaf images from the infected leaf images. The proposed model performs data acquisition, pre-processing, and application of the DCNN model to the acquired dataset into multiple distinct classes. The DCNN module is composed of multiple convolutional and max pooling layers. While the second convolutional layer concentrates on high-level characteristics, the first convolutional layer collects low-level features. Coupled with MaxPool layers to downsample the feature matrix generated by the convolution layer. The dropout layer randomly drops the identical weights of neurons to tackle the overfitting issue. Our proposed model classifies citrus leaf diseases into Bacterial spot, Black spot, Canker, Citrus Powdery Mildew, Greening, Melanose, and healthy images with an accuracy of 97.66%, outperforming the state-of-the-art deep learning models. The proposed model can be further extended to mobile applications (IoT), which facilitates farmers with prompt and timely messaging to adopt precautionary measures to take necessary actions to save the crops. Real-time image capturing (sensory data) can be used for image extraction. To incorporate further hybrid combinations of deep learning approaches, like combining CNN with LSTM or Bi-LSTM. The proposed work can be extended further from leaf disease detection to other parts of plants, such as stems and flowers.

## Declarations

*Ethical Approval and Consent to Participate:* This study strictly adhered to the Declaration of Helsinki and relevant national and institutional ethical guidelines. Informed consent was not required, as secondary data available on websites was obtained for analysis. All procedures performed in this study were by the ethical standards of the Helsinki Declaration.

*Consent for Publication:* We, both authors, give our consent for publication.

*Availability of Data and Materials:* Data will be made available upon request from the corresponding author.

*Competing Interest:* The authors declare that they have no competing interests.

*Funding:* Not Applicable

*Authors' Contribution:* All authors have equal contributions.

*Acknowledgement:* The authors are highly thankful to colleagues and fellows for their cooperation in conducting this study.

## References

Albawi, S., Mohammed, T.A., & Al-Zawi, S. (2017). *Understanding of a convolutional neural network.* In the 2017 International Conference on Engineering and Technology (ICET), pp. 1-6.

Ali, H., Lali, M., Nawaz, M.Z., Sharif, M., & Saleem, B. (2017). Symptom-based automated detection of citrus diseases using color histogram and textural descriptors. *Computers and Electronics in Agriculture*, 138; 92-104.

Andrianto, H., Faizal, A., & Armandika, F. (2020). *Smartphone application for deep learning-based rice plant disease detection*. In 2020 International Conference on Information Technology Systems and Innovation (ICITSI), pp. 387-392.

Atila, U., Ucar, M., Akyol, K., & Ucar, U. (2021). Plant leaf disease classification using an EfficientNet deep learning model. *Ecological Informatics*, 61; 101182.

Baldi, P., & Sadowski, P. J. (2013). Understanding dropout. *Advances in Neural Information Processing Systems*, 26; 2013.

Barbedo, J. G. A., Koenigkan, L. V., Halfeld-Vieira, B. A. *et al.* (2018). Annotated plant pathology databases for image-based detection and recognition of diseases. *IEEE Latin America Transactions*, 16; 1749-1757.

Bharate, A. A., & Shirdhonkar, M. A review on plant disease detection using image processing. in the 2017 International Conference on Intelligent Sustainable Systems (ICISS), pp. 103-109.

Buckland, M., & Gey, F. (1994). The relationship between recall and precision. *Journal of the American Society for Information Science*, 45; 12-19.

Catara, A., & Polizzi, G. (1999). Dry root rot of citrus: symptoms, causes, and susceptibility of rootstocks. *Rivisto di Fruticolture*, 6; 38-41.

Chen, J., Chen, J., Zhang, D., Sun, Y., Nanehkaran, Y.A. (2020). Using deep transfer learning for image-based plant disease identification. *Computers and Electronics in Agriculture*, 173; 105393.

Coulibaly, S., Kamsu-Foguem, D., Kamissoko, D., & Traore, D. (2019). Deep neural networks with transfer learning in millet crop images. *Computers in Industry*, 108; 115-120.

Dai, Q., Guo, Y., Li, Z., Song, S., Lyu, S., Sun, D. et al. (2023). Citrus disease image generation and classification based on improved FastGAN and EfficientNet-B5. *Agronomy*, 13; 988.

Dandurand, L., & Menge, J. (1992). Influence of Fusarium Solani on citrus root rot caused by Phytophthora parasitica and Phytophthora Citrophthora. *Plant and Soil*, 144; 13-21, 1992.

Dhanesha, D., & Shrinivasa, N. C. (2018). Segmentation of arecanut bunches using the HSV color model in the 2018 International Conference on Electrical, *Electronics, Communication, Computer, and Optimization Techniques*, pp. 37-41.

Dhanesha, R., Naika, C. S., & Kantharaj, Y. (2019). *Segmentation of arecanut bunches using the YCgCr color model.* In 2019 1st International Conference on Advances in Information Technology (ICAIT), pp. 50-53.

Diah, N. M., Jahim, M. L. M., Roslan, N. A. M., Ibrahim, Z., & Abdullah, A. (2016). *Development of a mobile application for plant disease recognition using a convolutional neural network method*. In 2021 6th IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE), pp. 1-6.

Elaraby, A., Hamdy, W., & Alanazi, S. (2022). Classification of citrus diseases using an optimization deep learning approach. *Computational Intelligence and Neuroscience*, 2022.

Elhassouny, A., & Smarandache, F. (2019). Smart mobile applications to recognize tomato leaf diseases using convolutional neural networks. *Collected Papers*, p. 431.

Falaschetti, L., Manoni, L., Di Leo, D., Pau, D., Tomaselli, V., & Turchetti, C. (2022). A CNN-based image detector for plant leaf disease classification. *HardwareX,* 12; e00363.

Ferentinos, K. P. (2018). Deep learning models for plant disease detection and diagnosis. *Computers and Electronics in Agriculture*, 145; 311-318.

Gayathri, S., Wise, D. J. W., Shamini, P. B., & Muthukumaran, N. (2020). *Image analysis and detection of tea leaf disease using deep learning.* In 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), pp. 398-403.

Ghosal, S., & Sarkar, K. (2020). *Rice leaf diseases classification using CNN with transfer learning*. In 2020 IEEE Calcutta Conference (CALCON), pp. 230-236.

Gillund. G., & Shiffrin, R.M. (1984). A retrieval model for both recognition and recall. *Psychological Review*, 91; 1.

Giusti, A., Cireşan, D. C., Masci, J., Gambardella, L. M., & Schmidhuber, J. (2013). *Fast image scanning with deep max-pooling convolutional neural networks*. In 2013 IEEE International Conference on Image Processing, pp. 4034-4038.

Hu, G., Yin, C., Wan, M., Zhang, Y., & Fang, Y. (2020). Recognition of diseased Pinus trees in UAV images using deep learning and AdaBoost classifier. *Biosystems Engineering*, 194; 138-151.

Hughes, D., & Salathe, M. (2015). An open-access repository of images on plant health to enable the development of mobile disease diagnostics. *arXiv preprint arXiv*:1511.08060.

Jin, J., Dundar, D., & Culurciello, E. (2014). Flattened convolutional neural networks for feedforward acceleration. *arXiv preprint arXiv*:1412.5474, 2014.

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

Khattak, A., Asghar, M. U., Batool, U., Asghar, M. Z., Ullah, H., Al-Rakhami, M. *et al*. (2021). Automatic detection of citrus fruit and leaves diseases using a deep neural network model. *IEEE Access*, 9; 112942-112954.

Kour, V. P., & Arora, S. Particle swarm optimization-based support vector machine (P-SVM) for the segmentation and classification of plants. *IEEE Access*, 7; 29374-29385.

Kumar, M., Gupta, P., Madhav, P. (2020). *Disease detection in coffee plants using a convolutional neural network.* In 2020, the 5th International Conference on Communication and Electronics Systems (ICCES), pp. 755-760.

Kumar, N., Belhumeur, P. N., Biswas, A., Jacobs, D. W., Kress, W. J., Lopez, I. C. *et al*. (2012). *Leafsnap: A computer vision system for automatic plant species identification*. In Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, Proceedings, Part II 12, pp. 502-516.

Li, Y., & Yang, J. (2020). Few-shot cotton pest recognition and terminal realization. *Computers and Electronics in Agriculture*, 169; 105240.

Liang, X., Wang, X., Lei, Z., Liao, S., & Li, S. Z. (2017). *Soft margin SoftMax for deep classification*. In International Conference on Neural Information Processing, pp. 413-421.

Liaqat, M. I., Hassan, M. A., Shoaib, M., Khurshid, S. K., & Shamseldin, M.A. (2022). Sentiment analysis techniques, challenges, and opportunities: Urdu language-based analytical study. *Peer J Computer Science*, 8; e1032.

Liu, Z., Xiang, X., Qin, J., Ma, Y., Zhang, Q., & Xiong, N. N. (2021). Image recognition of citrus diseases based on deep learning. *CMC-Computers Materials & Continua*, 66; 457-466.

Luaibi, A. R., Salman, T. M., & Miry, A. H. (2021). Detection of citrus leaf diseases using a deep learning technique. *International Journal of Electrical and Computer Engineering*, 11; 1719.

Maria, S. K., Taki, S. S., Mia, M. J., Biswas, A. A., Majumder, A., & Hasan, F. (2022). *Cauliflower disease recognition using machine learning and transfer learning*. In Smart Systems: Innovations in Computing: Proceedings of SSIC 2021, pp. 359-375.

Mendonça, L., Zambolim, L., & Badel, J. (2017). Bacterial citrus diseases: major threats and recent progress. *Journal of Bacteriology & Mycology: Open Access*, 5; 340-350.

Mohanty, S. P., Hughes, D. P., & Salathe, M. (2016). Using deep learning for image-based plant disease detection. *Frontiers in Plant Science*, 7; 1419.

Munisami, T., Ramsurn, M., Kishnah, S., & Pudaruth, S. (2015). Plant leaf recognition using shape features and colour histogram with K-nearest neighbor classifiers. *Procedia Computer Science*, 58; 740-747.

Naqvi, S. A. H., Wang, J., Malik, M. T., Umar, U.-U.-D., Hasnain, A., Sohail, M. A. *et al*. (2022). Citrus canker—distribution, taxonomy, epidemiology, disease cycle, pathogen biology, detection, and management: A critical review and future research agenda. *Agronomy*, 12; 1075.

Nie, X., Wang, L., Ding, H., & Xu, M. (2019). Strawberry verticillium wilt detection network based on multi-task learning and attention. *IEEE Access*, 7; 170003-170011.

Pandian, J. A., Kumar, V. D., Geman, O., Hnatiuc, M., Arif, M., & Kanchanadevi, K. (2022). Plant disease detection using a deep convolutional neural network. *Applied Sciences*, 12; 6982.

Qassim, H., Verma, A., Feinzimer, D. (2018). *Compressed residual-VGG16 CNN model for big data places image recognition*. In 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC), pp. 169-175.

Rahman, C. R., Arko, P. S., Ali, M. E., Khan, M. A. I., Apon, S. H., Nowrin, F. et al. (2020). Identification and recognition of rice diseases and pests using convolutional neural networks. *Biosystems Engineering*, 194; 112-120.

Rao, E. G., Anitha, G., & Kumar, G. K. (2021). *Plant disease detection using convolutional neural networks*. In 2021, the 5th International Conference on Trends in Electronics and Informatics (ICOEI), pp. 1473-1476.

Rauf, H. T., Saleem, B. A., Lali, M. I. U., Khan, M. A., Sharif, M., & Bukhari, S. A. C. (2019). A citrus fruit and leaves dataset for the detection and classification of citrus diseases through machine learning. *Data in Brief*, 26; 104340.

Rumy, S. S. H., Hossain, M. I. A., Jahan, F., Tanvin, T. (2021). *An IoT-based system with edge intelligence for rice leaf disease detection using machine learning*. In 2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), pp. 1-6.

Sagar, A., & Jacob, D. (2021). On using transfer learning for plant disease detection. B*ioRxiv*. Google Scholar.

Saheb, S. K., Narayanan, B., & Rao, T. V. N. (2022). Review of machine learning-based disease diagnosis and severity estimation of COVID-19. *Journal of Algebraic Statistics*, 13; 2022.

Salamon, J., & Bello, J. P. (2017). Deep convolutional neural networks and data augmentation for environmental sound classification. *IEEE Signal Processing Letters*, 24; 279-283.

Saleem, M. H., Khanchi, S., Potgieter, J., & Arif, K. M. (2020). Image-based plant disease identification by deep learning meta-architecture. *Plants*, 9; 1451.

Sharma, M., Nath, K., Sharma, R.K., Kumar, C. J., & Chaudhary, A. (2022). Ensemble averaging of transfer learning models for identification of nutritional deficiency in rice plants. *Electronics*, 11; 148.

Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data*, 6; 1-48.

Siddique, M. I., & Garnevska, E. (2018). Citrus value chain (s): A survey of the Pakistan citrus industry. Agric. *Value Chain*, 37; 37-58.

Singh, H., Rani, R., & Mahajan, S. (2020). *Detection and classification of citrus leaf disease using hybrid features*. In Soft Computing: Theories and Applications: Proceedings of SoCTA 2018, pp. 737-745.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15;1929-1958.

Sultana, F., Sufian, A., & Dutta, P. (2018). *Advancements in image classification using a convolutional neural network*. In 2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), pp. 122-129.

Swets, J. A. (2021). Measuring the accuracy of diagnostic systems. *Science*, 240; 1285-1293.

Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., & Liu, C. (2018). *A survey on deep transfer learning.* In Artificial Neural Networks and Machine Learning–ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4-7, 2018, Proceedings, Part III 27, pp. 270-279.

Tan, M., Pang, R., & Le, Q.V. (2020). *Efficientdet: Scalable and efficient object detection*. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 10781-10790.

Taylor, L., & Nitschke, G. (2018). *Improving deep learning with generic data augmentation*. In 2018 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 1542-1547.

Thapa, R., Zhang, K., Snavely, N., Belongie, S., & Khan, A. (2020). The Plant Pathology Challenge 2020 data set to classify foliar disease of apples. *Applications in Plant Sciences*, 8; e11390.

Turkoglu, T., Yanikoğlu, B., & Hanbay, D. (2022). PlantDiseaseNet: Convolutional neural network ensemble for plant disease and pest detection. *Signal, Image and Video Processing*, 16; 301-309.

Udawant, P., & Srinath, P. (2021). Leaf diagnosis using transfer learning. In Applied Information Processing Systems: *Proceedings of ICCET 2021*, pp. 235-246.

Valdoria, J. C., Caballeo, A. R., Fernandez, B. I. D., & Condino, J. M. M. (2019). *iDahon: An Android-based terrestrial plant disease detection mobile application through digital image processing using a deep learning neural network algorithm*. in 2019 4th International Conference on Information Technology (InCIT), pp. 94-98.

Vallabhajosyula, S., Sistla, V., & Kolli, V.V.K. (2022). Transfer learning-based deep ensemble neural network for plant leaf disease detection. *Journal of Plant Diseases and Protection*, 129; 545-558.

Varshney, D., Babukhanwala, B., Khan, J., Saxena, D., Singh, A. K. (2022). *Plant disease detection using machine learning techniques*. In 2022 3rd International Conference for Emerging Technology (INCET), pp. 1-5.

Wang, C., Du, P., Wu, H., Li, J., Zhao, C., & Zhu, H. (2021). A cucumber leaf disease severity classification method based on the fusion of DeepLabV3+ and U-Net. *Computers and Electronics in Agriculture*, 189; 106373.

Wang, Y., Wang, C., Zhang, H., Dong, Y., & Wei, S. (2019). Automatic ship detection based on RetinaNet using multi-resolution Gaofen-3 imagery. *Remote Sensing*, 11; 531.

Wojtowicz, A., Piekarczyk, J., Czernecki, B., & Ratajkiewicz, H. (2021). A random forest model for the classification of wheat and rye leaf rust symptoms based on pure spectra at the leaf scale. *Journal of Photochemistry and Photobiology B: Biology*, 223; 112278.

Yadav, P. K., Burks, T., Frederick, Q., Qin, J., Kim, M., Ritenour, M.A. (2022). Citrus disease detection using convolution neural network generated features and Softmax classifier on hyperspectral image data. *Frontiers in Plant Science*, 13; 1043712.

Yang, C., Teng, Z., Dong, C., Lin, Y., Chen, R., & Wang, J. (2022). In-field citrus disease classification via convolutional neural network from smartphone images. *Agriculture*, 12; 1487.

Yang, R., Liao, T., Zhao, P., Zhou, W., He, M., & Li, L. (2022). Identification of citrus diseases based on AMSR and MF-RANet. *Plant Methods*, 18;1-21.

Yu, H., Cheng, X., Li, Z., Cai, Q., & Bi, C. (2022). Disease recognition of apple leaf using a lightweight multi-scale network with ECANet. *CMES-Computer Modeling in Engineering & Sciences*, 132.

**Note:** **Open Access Digital Management and Governance Review** is under the process of recognition by the Higher Education Commission Pakistan in the Y category.

**Disclaimer/ Publisher's Note:** The statements, opinions, and data contained in all publications in this journal are solely those of the individual author(s) and not of the MDPIP and/ or the editor(s). MDPIP and editor(s) disclaim responsibility for any injury to the people or property resulting from any ideas, methods, instructions, or products referred to in the content.