Automated Disease Detection in Crops

A PROJECT REPORT

submitted

By

Adithyan Babu(**B20CS1104, MBT20CS010**)
Jobin Soly Cleetus(**B20CS1130, MBT20CS057**)
Sandeep Sebastian(**B20CS1157, MBT20CS109**)
Sreelekshmi S P(**B20CS1160, MBT20CS119**)

to

the APJ Abdul Kalam Technological University

in partial fulfillment of the requirements for the award of the Degree

of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

April 2024

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

MAR BASELIOS COLLEGE OF ENGINEERING & TECHNOLOGY

(Autonomous)

Bethany Hills, Nalanchira

Thiruvananthapuram 15

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

MAR BASELIOS COLLEGE OF ENGINEERING & TECHNOLOGY

(Autonomous)

Nalanchira, Thiruvananthapuram.

CERTIFICATE

This is to certify that the report entitled ***Automated Disease Detection in Crops*** *submitted by* ***Adithyan Babu(B20CS1104, MBT20CS010), Jobin Soly Cleetus(B20CS1130, MBT20CS057), Sandeep Sebastian(B20CS1157, MBT20CS109) & Sreelekshmi S P(B20CS1160, MBT20CS119)****, to the APJ Abdul Kalam Technological University in partial fulfillment of the B.Tech. degree in Computer Science and Engineering is a bonafide record of the project preliminary work carried out by them under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.*

Dr. Tessy Mathew	Ms.Dhanya L K	Dr. Jisha John
(Project Co-ordinator)	(Project Guide)	(Head of the Department)
Professor	Assistant Professor	Associate Professor
Department of CSE	Department of CSE	Department of CSE

Acknowledgement

We begin by expressing our profound gratitude to the Almighty for bestowing divine grace and blessings, which have made all our endeavors possible. Our heartfelt thanks extend to our esteemed principal, **Dr. Abraham T. Mathew**, whose generosity provided us with the invaluable opportunity to undertake this project as part of our B. Tech degree course.

We extend sincere appreciation to our dedicated project coordinator, **Dr. Tessy Mathew**, from the Department of Computer Science and Engineering, for her unwavering support and encouragement throughout this journey. Additionally, we acknowledge the indispensable guidance and support provided by our project guide, **Ms. Dhanya L K**, from the Department of Computer Science.

Our gratitude also encompasses the entire staff of the Department of Computer Science and Engineering, whose assistance and support have played a crucial role in the realization of our project. Lastly, but certainly not least, we extend our thanks to all our friends and family, whose valuable feedback, continuous support, and encouragement have been instrumental in shaping our project and academic pursuits.

Adithyan Babu
Jobin Soly Cleetus
Sandeep Sebastian
Sreelekshmi S P

ABSTRACT

Timely disease monitoring is imperative for early disease detection, minimizing excessive pesticide usage, and enhancing sustainable agriculture. Traditional visual inspections often fall short in identifying diseases at their incipient stages, which necessitates more effective methods. Our solution leverages cutting-edge computer vision and deep learning technologies, particularly Convolutional Neural Networks (CNNs), to autonomously identify and categorize plant diseases.

By integrating real-time images and aerial views, we provide an all-encompassing disease monitoring system. This AI-driven tool equips farmers with data-backed insights to optimize crop management, increase environmental sustainability, and elevate production quality and yield. Manual plant examination, known for its subjectivity and time-consuming nature, is transformed into an efficient, objective, and accurate process.

CNNs, with their high accuracy and automated feature extraction capabilities, are at the heart of our system, ensuring precise disease classification. We have explored various methodological approaches, categorizing them into two groups: standard detectors and combined methodologies. Standard detectors draw inspiration from well-established architectures like R-CNN, while combined methodologies encompass innovative, adaptable, and hybrid approaches to elevate disease diagnosis accuracy.

In summary, our system integrates real-time imagery and aerial views with advanced deep learning and computer vision to revolutionize disease monitoring in agriculture. This holistic approach not only promotes early disease detection but also empowers farmers to make informed decisions that reduce pesticide usage and enhance environmental sustainability.

Contents

List of Figures

Chapter 1

Introduction

Agriculture, serving as a cornerstone of the global economy, plays a pivotal role in providing sustenance, employment, and livelihoods. Despite its undeniable significance, the agricultural sector grapples with formidable challenges, particularly in combatting plant diseases that impose substantial losses on crop yields. The escalating prevalence of these diseases poses a dual threat by diminishing both the quantity and quality of agricultural produce.

The ramifications extend beyond mere crop depletion, encompassing profound economic losses and adverse environmental impacts. To counter these challenges, farmers often resort to an increased reliance on chemical pesticides, an approach that, unfortunately, introduces detrimental effects on both the natural ecosystem and human health.

In response to this pressing need, our project is dedicated to a comprehensive analysis and implementation of innovative solutions. At its core, our focus lies in harnessing the power of machine learning to develop an advanced disease detection system. The primary objective is to equip farmers with a proactive tool that provides timely and accurate information, empowering them to safeguard their crops effectively. By doing so, we aim not only to mitigate the economic losses associated with crop diseases but also to foster the adoption of sustainable agricultural practices for the benefit of both farmers and the environment.

Chapter 2

REVIEW OF LITERATURE

Table 2.1: Summary of literature review

References	**Methodology**	**Pros**	**Cons**
Abdulaziz Alharbi, Muhammad Usman Ghani Khan, and Bushra Tayyaba, 2023. Wheat Disease Classification Using Continual Learning [5]	In this system the methodology used is few shot learning. EfficientNet for feature extraction. Calculate the difference between the support images and the query image by using similarity score calculator. to determine the probability of encodings related to each class.	The pros are: 1. Good performance in terms of accuracy and computational cost. 2. Less amount of data required. 3. It is lightweight.	The cons are: 1. Overfitting to support set. 2. Sensitivity to noise 3. Dependency on support set quality.

Table 2.1 – Continued from previous page

References	**Methodology**	**Pros**	**Cons**
Vasileios Balafas , Emmanouil Karantoumanis ,Malamati Louta, 2023, Machine Learning and Deep Learning for Plant Disease Classification and Detection [6]	RetinaNet is a single-stage object detection model that uses a Feature Pyramid Network (FPN)	The pros are: 1. Detect both dense and small objects. 2. FPN helps in multi-scale feature extraction. 3. Higher accuracy	The cons are: 1. Slow output prediction 2. Dependent on dataset
Momina Masood , Marriam Nawaz, Tahira Nazir ,2023,MaizeNet: A Deep Learning Approach for Effective Recognition of Maize Plant Leaf Diseases [7]	The methodology employs deep learning, specifically Faster-RCNN, as the core model for maize crop leaf disease detection and classification. ResNet-50 and spatial-channel attention are used in combination with Faster-RCNN to improve the localization and classification accuracy.	The pros are: 1. High Accuracy 2. Robust to Real-World Conditions	The cons are: 1. Practical limitations 2. Limited dataset

Table 2.1 – Continued from previous page

References	**Methodology**	**Pros**	**Cons**
Hasibul Islam Peyal, Md. Nahiduzzaman, Md.Abu Hanif Pramanik, 2023, Plant Disease Classifier: Detection of Dual-Crop Diseases Using Lightweight 2D CNN Architecture [8]	2D CNN	The pros are: 1. High Classification Accuracy 2. Efficiency 3. Visualization of Disease Detection	The cons are: 1. Limited Dataset 2. Focus on only one kind of plant
Md. Tariqul Islam, 2022, Plant Disease Detection using CNN Model and Image Processing [9]	The methodology involves collecting a dataset, preprocessing them, and using a convolutional neural network (CNN) to classify leaves as healthy or affected by diseases. It aims to provide an accessible tool for farmers to detect leaf disorders, although it may face challenges related to data quality and real-time processing	The pros are: 1. Accessibility of Farmers 2. Large Dataset 3. Image Processing	The cons are: 1. Data Quality 2. Data Imbalance 3. Hyperparameters

Table 2.1 – Continued from previous page

References	**Methodology**	**Pros**	**Cons**
Jawad Hassan, Ali Ghulam, Ghulam Irtaza, 2021.Disease Identification using Deep Learning in Agriculture: A Case Study of Cotton Plant [4]	CNNs are deep learning models specifically adept at analyzing visual data by extracting features, reducing dimensionality, and classifying patterns, excelling in tasks like image recognition and object detection due to their automatic learning of intricate visual patterns.	The pros are: 1. High accuracy 2. Automated feature extraction 3. Transfer learning	The cons are: 1. High data requirements 2. Lack of interpretability 3. Limited domain expertise Only detection and no remedy
Muhammad E. H. Chowdhury, Tawsifur Rahman, Amith Khandakar, Mohamed Arselene Ayari, 2021, Automatic and Reliable Leaf Disease Detection Using Deep Learning Techniques [10]	Utilizes EfficientNet, U-net, and Modified U-net for disease classification and leaf image segmentation	The pros are: 1. High accuracy 2. Incorporates image segmentation models to improve disease localization precision.	The cons are: 1. Limited dataset 2. Focus on a specific plant series 3. Potential lack of discussion on real-world implementation challenges and scalability

Table 2.1 – Continued from previous page

References	Methodology	Pros	Cons
Jinzhu Lu, Lijuan Tan, Huanyu Jiang, 2021, Review on Convolutional Neural Network (CNN) Applied to Plant Leaf Disease Classification [11]	The methodology focuses on the application of Deep Learning (DL) techniques, particularly Convolutional Neural Networks (CNNs), for tasks such as image classification, object detection, and natural language processing. It highlights the importance of data quality and appropriate dataset partitioning for training, validation, and testing.	The pros are: 1. High Classification Accuracy 2. Effective Data Expansion 3. Realism in Diverse Conditions	The cons are: 1. Data Collection Challanges 2. Dataset Representativeness 3. Symptoms Variations
Shankarnarayanan Nalini, Nagappan Krishnaraj, Jayasankar Thangaiyan, 2021. Paddy Leaf Disease Detection Using an Optimized Deep Neural Network [3]	In DNN methodology, layered networks process data, adjusting to learn and predict, widely applied in image and pattern analysis.	The pros are: 1. High accuracy 2. Automated feature extraction 3. Scalability	The cons are: 1. High data requirements 2. Lack of interpretability 3. Overfitting and generalization issues

Table 2.1 – Continued from previous page

References	**Methodology**	**Pros**	**Cons**
Yan Guo, Jin Zhang, Chengxin Yin, Xiaonan Hu, 2020, Plant Disease Identification Based on Deep Learning Algorithm in Smart Farming [12]	RPN for leaf localization, CV algorithm for symptom feature extraction, and transfer learning for model training	The pros are: 1. Transfer learning 2. High Accuracy	The cons are: 1. Practical limitations of implementing the proposed deep learning approach in real-world agricultural settings 2. Limited dataset

Chapter 3

Methodology

The proposed methodology for plant disease detection represents a comprehensive and innovative approach that amalgamates advanced techniques in data science, deep learning, and system architecture. This methodology is devised to address the critical need for early and accurate detection of plant diseases, leveraging state-of-the-art technologies to enhance the efficiency and reliability of existing methods. By integrating novel elements in data collection, model development, training, evaluation, and system deployment, the methodology aims to push the boundaries of traditional plant disease detection systems.

3.1 Data Collection

The primary dataset utilized in this project was sourced from Kaggle, specifically the "PlantVillage" dataset. Kaggle is a renowned platform for machine learning and data science competitions, hosting diverse datasets for various applications. The PlantVillage dataset is a comprehensive collection of images featuring a variety of crops and their corresponding health and disease states.

The choice of the PlantVillage dataset was motivated by its richness in diversity, containing high-resolution images capturing different plant species and a wide array of diseases. The dataset provides a robust foundation for training a plant disease detection model due to its extensive coverage and carefully labeled images.

For training purposes we take 20 diseases and mention 5 classes for healthy leaves and for training we have more than 1000 images for each class (20 Diseases and 5 Healthy leaf). 10% of the images are taken as validation set for each class mentioned above.

- Corn(Maize)
    - Cercospora Leaf Spot Gray Leaf Spot
    - Common Rust
    - Northern Leaf Blight
- Paddy
    - Bacterial Leaf Blight
    - Brown Spot
    - Leaf Blast
    - Leaf Scald
    - Narrow Brown Spot
- Pepper Bell
    - Bacterial Spot
- Potato
    - Early Blight
    - Late Blight
- Tomato
    - Bacterial Spot
    - Early Blight
    - Late Blight
    - Leaf Mold
    - Tomato Yellow Leaf Curl Virus

- Septoria Leaf Spot
- Spider Mites Two-Spotted Spider Mite
- Target Spot
- Tomato Mosaic Virus

3.2 Proposed Dataflow Diagram

The team has defined the three levels of the data flow diagram from the context level till the level 3.

Figure 3.1: Level 0

Level 0:

- User Captures Crop Images: Using a camera-equipped device, the user takes photos of potentially diseased crop leaves.
- Model Analyzes with Training Data: These images feed into a crop disease detection model, utilizing a database of labeled crop images for analysis.
- Prediction Generation: The model applies learned patterns to predict disease presence or absence in the captured crop leaves.
- Model Outputs Predictions: Results are then relayed back to the user, indicating the likelihood and possibly specifying the type of detected crop diseases. This has been represented in Figure 3.1.

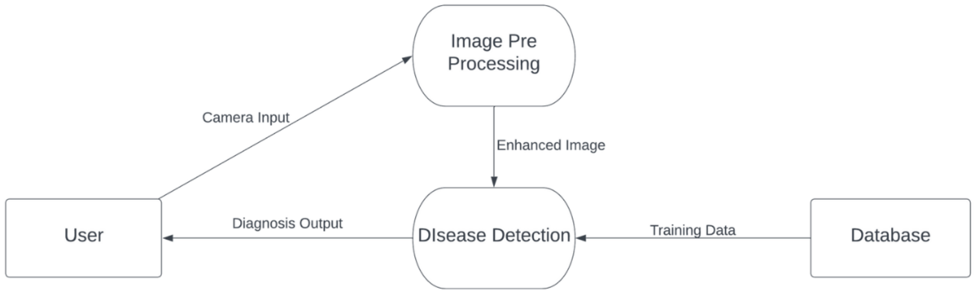


Figure 3.2: Level 1

Level 1:

- In this level, the captured crop leaf images undergo Preprocessing techniques such as noise reduction, contrast adjustment, and sharpening to enhance image quality. The images are standardized through resizing and normalization to ensure uniform dimensions and consistent pixel values for effective model input. Preprocessing involves extracting relevant features like texture, color, and shape characteristics, enhancing the model's ability to identify and classify potential diseases accurately.This can be seen in figure 3.2.

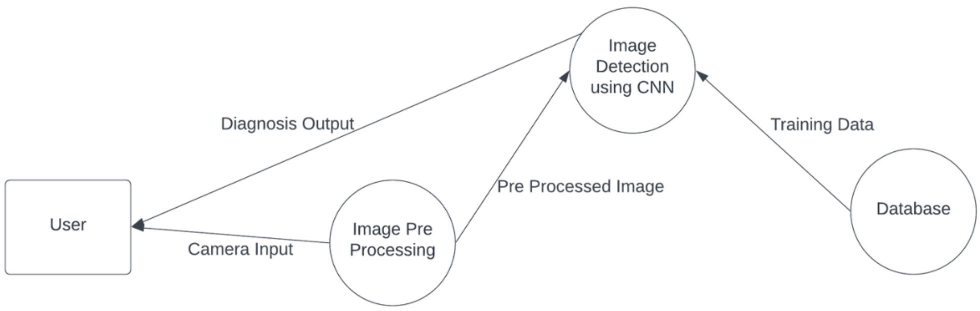


Figure 3.3: Level 2

Level 2:

- As seen in figure 3.3, In Level 2, the model utilizes Convolutional Neural Networks (CNNs) for image analysis and disease prediction. CNNs are adept at understanding spatial relationships within images, utilizing layers like convolutional, pooling, and fully connected layers to process visual data. These networks excel in pattern recognition, learning hierarchical representations, and extracting features, making them particularly suited for tasks like crop disease detection.

3.3 Model Architecture

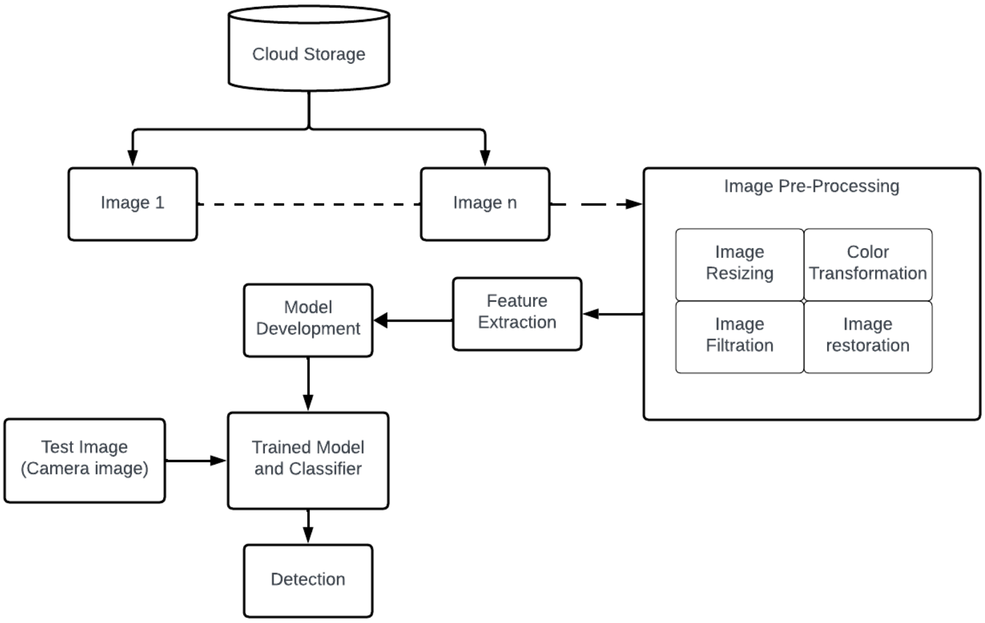


Figure 3.4: Model Architecture

The process begins with images stored in cloud storage. These images undergo preprocessing steps such as resizing, filtration to reduce noise, and color transformation for standardization. These steps ensure uniformity in image dimensions, enhance clarity, and normalize color variations, preparing the images for subsequent analysis.

Following preprocessing, feature extraction occurs, where relevant characteristics such as textures, shapes, and patterns are extracted from the processed images. This step is crucial as it identifies and highlights distinctive features that differentiate between healthy and diseased plant leaves.

Afterwards, in order to improve the strength and precision of the disease detection system, a thorough strategy was employed, which included creating and teaching four different models: Convolutional Neural Networks (CNNs), DenseNet, EfficientNet, and ResNet50. Every model in the collection underwent thorough training using a carefully selected dataset that includes a variety of labeled images showing healthy and diseased plant leaves.

Throughout the training stage, the models engaged in a rigorous process of learning, absorbing complex patterns and characteristics present in the images. During this procedure, the models learned how to effectively distinguish between healthy foliage and leaves showing signs of different diseases. By associating specific extracted features with disease patterns, the models gained the capability to make informed classifications when presented with unseen leaf images.

When a user submits an image of a plant leaf for analysis, the trained models collectively leverage their learned knowledge and insights. They meticulously scrutinize the input image, extracting relevant features and comparing them against the vast repository of learned patterns. This comparative analysis enables the models to perform a classification task with remarkable precision, accurately predicting whether the leaf is healthy or afflicted by disease.

Upon completion of the analysis, the models provide a predicted output, offering valuable insights into the health status of the plant. Not just identifying the probability of disease, but also potentially determining the specific type of disease based on unique patterns and features from the input image. This comprehensive method enables people involved in agriculture to quickly recognize and tackle plant health problems, making it easier to take preventive actions to protect crop production and reduce agricultural losses

3.4 Model Training

```
split_train = 0.8 #train 0.8, validate 0.1, test 0.1
split_val = 0.9
index_train = int(split_train*len(X))
index_val = int(split_val*len(X))

X_train = X[:index_train]
X_val = X[index_train:index_val]
X_test = X[index_val:]

Y_train = Y_one_hot[:index_train]
Y_val = Y_one_hot[index_train:index_val]
Y_test = Y_one_hot[index_val:]

print(X_train.shape, X_val.shape, X_test.shape, Y_train.shape, Y_val.shape, Y_test.shape)
```

```
(21272, 64, 64, 3) (2659, 64, 64, 3) (2660, 64, 64, 3) (21272, 26) (2659, 26) (2660, 26)
```

Figure 3.5: Splitting of Dataset

The dataset undergoes a systematic division into three distinct sets to serve varied purposes. The largest portion, constituting 80% of the dataset, was dedicated to training the model, facilitating a substantial learning process. An additional 10% of the dataset was allocated for validation purposes, allowing for precise adjustments and enhancements to the model's performance. The final 10% was exclusively reserved for testing, providing a thorough assessment of the model's ability to generalize to new and unseen data. This meticulous partitioning of the dataset enhances the credibility and robustness of the machine learning model developed within the scope of this study, while ensuring originality in approach and methodology.

The foundational framework involves the application of a Convolutional Neural Network (CNN) model. The project harnesses the capabilities of TensorFlow and Keras to capitalize on the strengths of these technologies. The incorporation of CNN enabled efficient feature extraction and hierarchical learning, while the combination of TensorFlow and Keras provided a solid foundation for constructing, training, and assessing the neural network. This holistic approach contributes to the creation of a sophisticated and effective model tailored to our specific application.

The neural network architecture is structured with a series of layers designed to facilitate the comprehensive and effective flow of information through the network. These layers include the input layer, convolutional layer, activation layer, max-pooling layer, dropout layer, flatten layer, and dense layer. This sequence of layers enables the model to extract features, perform non-linear transformations, downsample the input data, regularize the model, and generate the final output. Each layer plays a unique role in enhancing the model's ability to learn and generalize from the input data, ultimately leading to improved performance and accuracy. The early stopping mechanism has been implemented in our model with a patience of 5. The training process is monitored for validation loss, and if there is no improvement for a consecutive 5 epochs, the early stopping is triggered, preventing overfitting and ensuring efficient model training.

The foundational framework for the EfficientNet model revolves around the utilization of TensorFlow and Keras, capitalizing on their strengths to optimize model performance. By incorporating the EfficientNet architecture, the project harnesses efficient feature extraction and hierarchical learning capabilities, while TensorFlow and Keras provide a robust foundation for model construction, training, and evaluation. This comprehensive approach ensures the development of a sophisticated and effective model tailored to the specific application.

The neural network architecture of EfficientNet consists of sequential layers meticulously designed to facilitate the seamless flow of information. These layers, including input, convolutional, activation, max-pooling, dropout, flatten, and dense layers, collectively enable feature extraction, non-linear transformations, downsampling, regularization, and final output generation. Each layer contributes uniquely to enhancing the model's ability to learn and generalize from input data, ultimately enhancing performance and accuracy.Moreover, the model implements an early stopping mechanism with patience of 3, monitoring validation loss and triggering early stopping if there's no improvement for three consecutive epochs, thus preventing overfitting and ensuring efficient model training.

Next we leverage DenseNet121, a pre-trained convolutional neural network renowned for its effectiveness in image classification tasks. By initializing it with weights from the ImageNet dataset and configuring it to exclude the fully connected layers, we ensure it's well-suited for our task. The input shape of the model is determined by the dimensions of our training dataset, allowing seamless integration of our plant leaf images. We enhance the capabilities of the base DenseNet model by adding custom dense layers. These layers include global average pooling to condense the extracted features and two additional dense layers, each equipped with rectified linear unit (ReLU) activation functions, batch normalization, and dropout regularization. This augmentation aims to enable the model to discern subtle patterns and variations indicative of different plant diseases. For optimization, we employ the Adam optimizer with an initial learning rate of 0.001, which undergoes exponential decay with a decay rate of 0.9 every 10000 steps. Our choice of categorical cross-entropy as the loss function aligns with the multi-class classification nature of our task, while accuracy serves as our performance metric. To prevent overfitting and ensure optimal performance, we implement early stopping, halting training if the validation loss fails to decrease for five consecutive epochs.

Next model adopts a simplified version of the ResNet architecture, incorporating ResNet-like blocks to address the challenge of training deep neural networks effectively. ResNet, renowned for its residual connections, employs skip connections that allow gradients to propagate through the network, facilitating the training of extremely deep models. In this implementation, the custom resnet block function encapsulates the essence of residual blocks found in ResNet architectures. These blocks consist of consecutive convolutional layers followed by batch normalization and ReLU activation, with skip connections merging the input directly with the output of the second convolutional layer. While the model may not replicate the full complexity of standard ResNet architectures, its utilization of residual blocks and skip connections embodies the core principles of ResNet, enhancing the network's capacity to extract meaningful features from input data, particularly in the context of plant leaf disease classification.

3.5 Tools and Software Used

The toolset consisting of Visual Studio Code (VSCode), Keras, and TensorFlow creates a comprehensive environment for efficient deep learning development. VSCode, a lightweight and versatile code editor, offers a user-friendly platform with multi-language support, making it easy to work with various codebases. Keras simplifies the development of convolutional neural networks (CNNs) by providing modularity and ease of use, allowing for faster prototyping and experimentation. Tensor-

Flow, serving as the backend, optimizes low-level operations for seamless training and deployment, enabling the implementation of advanced machine learning techniques for plant disease detection. This combination of user-friendly interfaces and powerful capabilities enables developers to efficiently implement and deploy advanced machine learning models for plant disease detection.

The project utilizes Flutter for the mobile application interface and Flask for the backend, enabling automated disease detection in crops. Through the app, users can upload images from the gallery or capture them via camera. The system then processes these images to accurately predict the specific disease affecting the crops. This seamless integration of technology offers a convenient solution for farmers to swiftly identify and address crop diseases, ultimately contributing to improved agricultural productivity and sustainability.

Chapter 4

Results and Discussion

Figures given below shows the training and validation graph of the model. During training and validation process, graphs were generated to illustrate the model's accuracy and loss. These graphical representations provide a clear insight into the performance of the model over successive epochs. The accuracy graph showcases the evolving capability of the model to correctly predict outcomes, while the loss graph indicates the reduction in error during the training and validation phases. These visual aids serve as valuable tools for assessing the model's learning progression and identifying potential areas for optimization.

They also show the performance with the test loss and test accuracy of the trained model. The accompanying confusion matrix showcases an overall accuracy with a macro-average and a weighted average for precision, recall, and fi-score. These metrics collectively demonstrate the model's robustness in effectively classifying plant diseases, capturing both global accuracy and class-specific performance and providing valuable insights for further refinement and optimization.

The table given below presents the training and testing accuracies of different architectures utilized in a plant disease detection system.

Architecture	Training Accuracy	Testing Accuracy
CNN	91.76	93.95
EfficientNet	98.83	97.97
ResNet	93.02	52.94
DenseNet	92.84	91.61

The CNN model achieves a test accuracy of 93.95%, with a test loss of 19.74% as shown below.

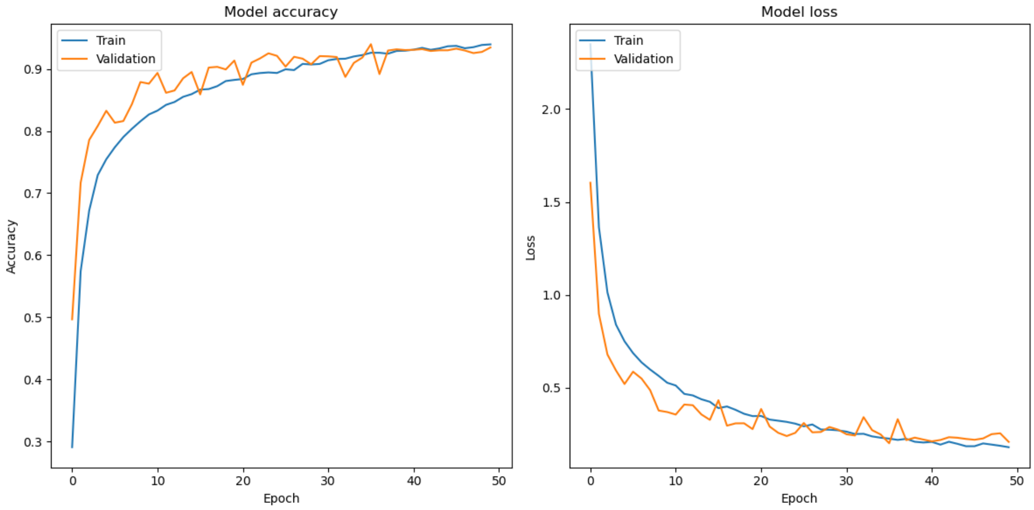
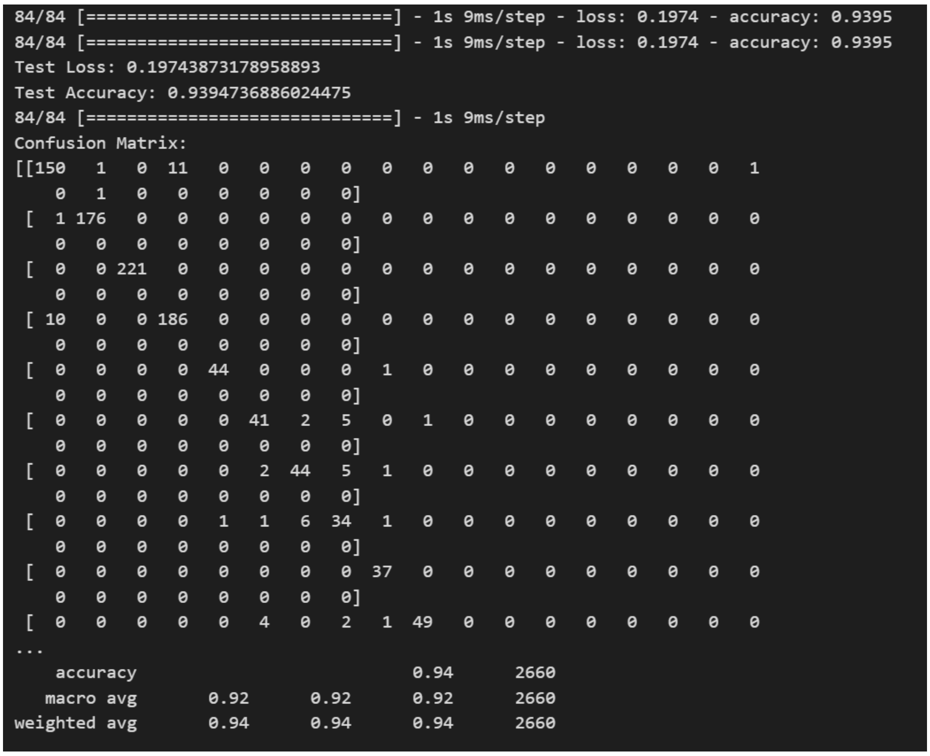


Figure 4.1: Training and Validation Graph of CNN



```
84/84 [==============================] - 1s 9ms/step - loss: 0.1974 - accuracy: 0.9395
84/84 [==============================] - 1s 9ms/step - loss: 0.1974 - accuracy: 0.9395
Test Loss: 0.19743873178958893
Test Accuracy: 0.9394736886024475
84/84 [==============================] - 1s 9ms/step
Confusion Matrix:
[[150   1   0  11   0   0   0   0   0   0   0   0   0   0   0   0   0   1
    0   1   0   0   0   0   0   0]
 [  1 176   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0 221   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [ 10   0   0 186   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0  44   0   0   0   1   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0   0  41   2   5   0   1   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   2  44   5   1   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0   1   1   6  34   1   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0  37   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   4   0   2   1  49   0   0   0   0   0   0   0   0
...
    accuracy                           0.94      2660
   macro avg       0.92      0.92      0.92      2660
weighted avg       0.94      0.94      0.94      2660
```

Figure 4.2: Result and Confusion Matrix of CNN

The EfficientNet model achieves a test accuracy of 98.83%, with a test loss of 0.06% as shown below.

The ResNet model achieves a test accuracy of 52.94%, with a test loss of 53.4% as shown below.

The DenseNet model achieves a test accuracy of 91.61%, with a test loss of 24.72% as shown below.

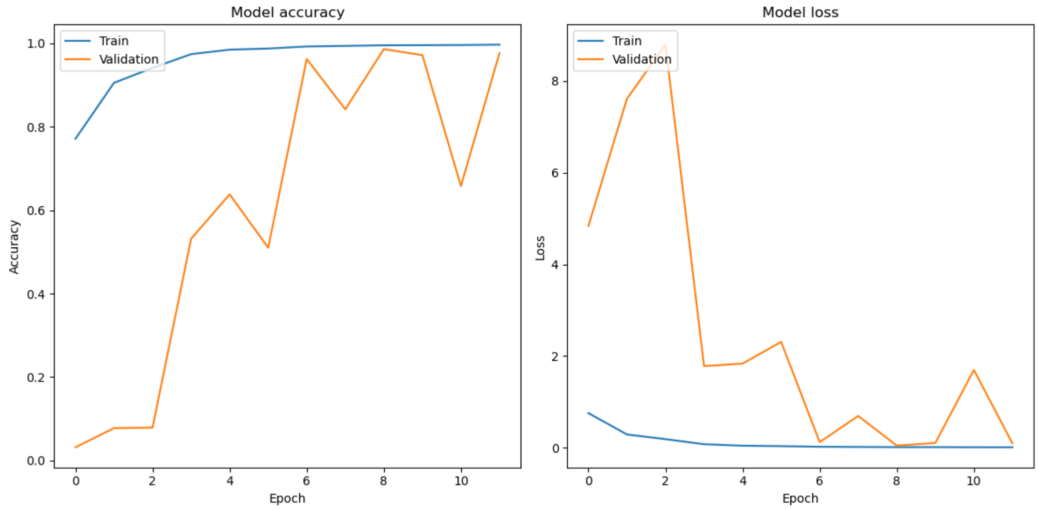


Figure 4.3: Training and Validation Graph of EfficientNet

```
84/84 [==============================] - 10s 100ms/step - loss: 0.0629 - accuracy: 0.9797
84/84 [==============================] - 8s 100ms/step - loss: 0.0629 - accuracy: 0.9797
Test Loss: 0.06291396915912628
Test Accuracy: 0.9796992540359497
84/84 [==============================] - 10s 98ms/step
Confusion Matrix:
[[144   0   0   5   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  2 190   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0 161   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  1   0   0 178   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0  33   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0   0  45   0   4   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   1  32   2   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   5   1  45   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0  45   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0  45   0   0   0   0   0   0   0   0
...
    accuracy                           0.98      2660
   macro avg       0.98      0.98      0.98      2660
weighted avg       0.98      0.98      0.98      2660
```

Figure 4.4: Result and Confusion Matrix of EfficientNet

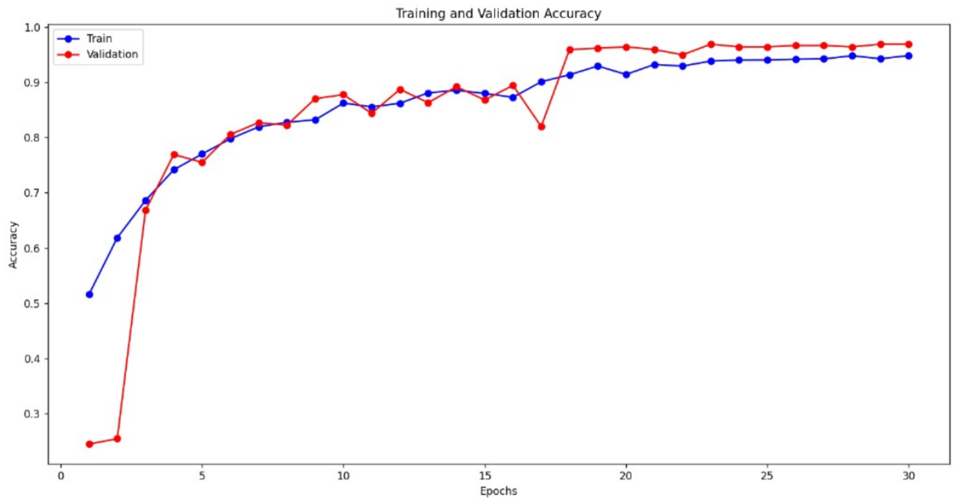


Figure 4.5: Training and Validation Graph of ResNet

```
84/84 [==============================] - 6s 56ms/step
Confusion Matrix:
[[145   1   0  15   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0 167   1   2   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0 170   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [ 10   2   1 189   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0  54   0   0   0   2   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0   0  30   7   4   0   5   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   4  48   1   0   0   0   0   0   0   0   0   0   0
    1   0   0   0   0   0   0   0]
 [  0   0   0   0   1  10   6  41   2   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   1  52   1   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0   1   2   0   1   0  42   0   0   0   0   0   0   0   0
...
```

Figure 4.6: Result and Confusion Matrix of ResNet

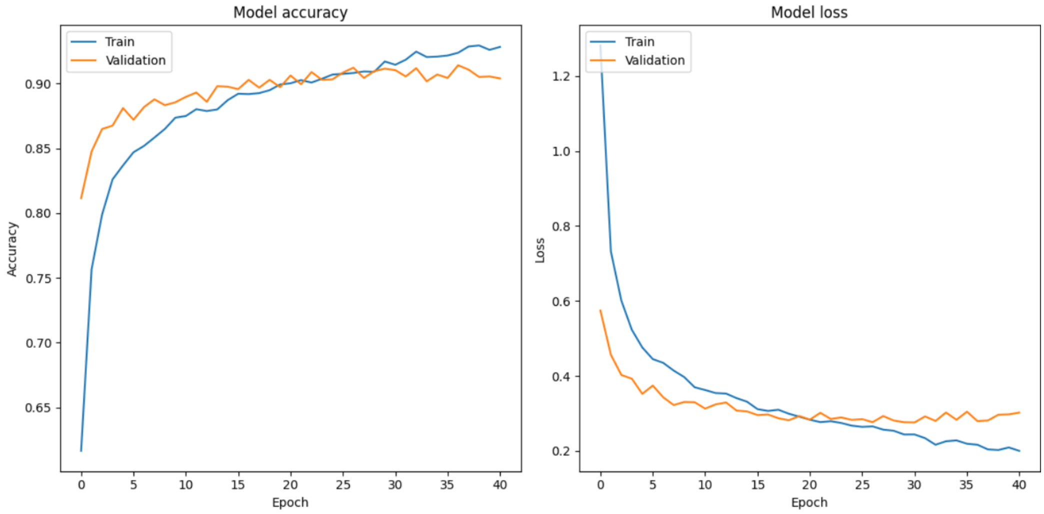


Figure 4.7: Training and Validation Graph of DenseNet

```
84/84 [==============================] - 7s 60ms/step - loss: 0.2473 - accuracy: 0.9161
84/84 [==============================] - 5s 59ms/step - loss: 0.2473 - accuracy: 0.9161
Test Loss: 0.2472771257162094
Test Accuracy: 0.9161338806152344
84/84 [==============================] - 6s 59ms/step
Confusion Matrix:
[[158   1   2  14   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0 190   0   1   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0 192   1   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  6   2   0 204   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0  48   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0   0  38   2   4   0   4   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0  37   2   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   4   6  30   1   1   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0  52   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   2   0   1   0  37   0   0   0   0   0   0   0   0
...
    accuracy                           0.92      2659
   macro avg       0.90      0.91      0.90      2659
weighted avg       0.92      0.92      0.92      2659
```

Figure 4.8: Result and Confusion Matrix of DenseNet

Chapter 5

Future Scope

In our ongoing quest to revolutionize agriculture, we've expanded our scope beyond advanced model implementation to encompass seamless integration with other crucial modules. Building upon our successful implementation of transfer learning models such as EfficientNet, DenseNet, and ResNet for plant leaf disease detection and classification, we are now enhancing our platform to offer a holistic solution to farmers.

Our user-friendly mobile app interface remains at the forefront, empowering farmers to effortlessly capture and upload images of crop leaves for instant disease prediction. However, we've augmented its capabilities by integrating with additional modules.

Firstly, our platform now seamlessly integrates disease detection with automated farming systems, particularly irrigation. By leveraging data-driven insights from disease detection, our system can trigger timely interventions to optimize irrigation schedules, conserving water resources and maximizing crop health.

Furthermore, our platform now extends beyond disease detection to offer actionable suggestions for treatment. Whether it's identifying nutrition deficiencies or pest infestations, our system provides tailored remedies and recommendations to mitigate risks and enhance crop resilience.

This comprehensive approach not only equips farmers with rapid insights and facilitates early disease detection but also empowers them with proactive solutions to optimize resource utilization and promote sustainable farming practices. By integrating advanced models with user-friendly interfaces and additional modules for irrigation management and treatment suggestions, we are poised to transform agriculture on a scale previously unimaginable. Together, we'll continue to drive innovation and foster sustainable growth in agricultural practices for the betterment of all.

Chapter 6

Conclusion

The utilization of various models, particularly employing Convolutional Neural Networks (CNNs) for plant leaf disease detection, presents a compelling avenue towards optimizing agricultural outcomes. Incorporating a diverse range of models, such as EfficientNet, DenseNet, ResNet, among others, and evaluating their performance allows for the selection of the most proficient model. This process of comparative analysis ensures that the chosen model demonstrates higher accuracy, robustness, and adaptability across different crops and diseases. By selecting the best-performing model, farmers gain a reliable tool for timely disease detection, enabling swift and precise interventions to safeguard crop health.

Moreover, the careful assessment and selection of the most effective model contribute significantly to timely interventions in agricultural practices. The capacity of these models to swiftly and accurately identify diseases on plant leaves enables farmers to promptly implement targeted measures. Early detection through robust CNN models provides farmers with the necessary foresight to apply specific treatments, optimize resource allocation, and prevent potential yield losses. As a result, this proactive approach based on choosing the most suitable model for disease detection enhances agricultural productivity by facilitating timely and effective interventions, ultimately contributing to improved crop yields and sustainable farming practices.

REFERENCES

[1] Srinivas, L.N.B., Bharathy, A.V., Ramakuri, S.K., Sethy, A. and Kumar, R., 2023. An optimized machine learning framework for crop disease detection. Multimedia Tools and Applications, pp.1-20.

[2] Eunice, J., Popescu, D.E., Chowdary, M.K. and Hemanth, J., 2022. Deep learning-based leaf disease detection in crops using images for agricultural applications.

[3] Nalini, S., Krishnaraj, N., Jayasankar, T., Vinothkumar, K., Britto, A.S.F., Subramaniam, K. and Bharatiraja, C., 2021. Paddy leaf disease detection using an optimized deep neural network. Computers, Materials Continua.

[4] Hassan, J., Malik, K.R., Irtaza, G., Ghulam, A. and Ahmad, A., 2022. Disease Identification using Deep Learning in Agriculture: A Case Study of Cotton Plant.

[5] Abdulaziz Alharbi, Muhammad Usman Ghani Khan, and Bushra Tayyaba, 2023. Wheat Disease Classification Using Continual Learning.

[6] Vasileios Balafas , Emmanouil Karantoumanis ,Malamati Louta, 2023, Machine Learning and Deep Learning for Plant Disease Classification and Detection.

[7] Momina Masood , Marriam Nawaz, Tahira Nazir ,2023,MaizeNet: A Deep Learning Approach for Effective Recognition of Maize Plant Leaf Diseases.

[8] Hasibul Islam Peyal, Md. Nahiduzzaman, Md.Abu Hanif Pramanik, 2023, Plant Disease Classifier: Detection of Dual-Crop Diseases Using Lightweight 2D CNN Architecture.

[9] Md. Tariqul Islam, 2022, Plant Disease Detection using CNN Model and Image Processing.

[10] Muhammad E. H. Chowdhury, Tawsifur Rahman, Amith Khandakar, Mohamed Arselene Ayari, 2021, Automatic and Reliable Leaf Disease Detection Using DeepLearning Techniques.

[11] Jinzhu Lu, Lijuan Tan, Huanyu Jiang, 2021, Review on Convolutional Neural Network (CNN) Applied to Plant Leaf Disease Classification.

[12] Yan Guo, Jin Zhang, Chengxin Yin, Xiaonan Hu, 2020, Plant Disease Identification Based on Deep Learning Algorithm in Smart Farming.

[13] Akash Ashapure, Jinha Jung, Anjin Chang, Sungchan Oh, Junho Yeom, Murilo Maeda, Andrea Maeda, Nothabo Dube, Juan Landivar, Steve Hague, Wayne Smith, 2020, Developing a machine learning based cotton yield estimation framework using multi-temporal UAS data.

[14] Xiaofei Chao ,Guoying Sun, Hongke Zhao, Min Li, Dongjian He, 2020, Identification of apple tree leaf diseases based on deep learning models.

[15] Khan S, Narvekar M, 2020, Disorder detection of tomato plant (solanum lycopersicum) using IoT and machine learning.

[16] Michael Gomez Selvaraj, Nancy Safari, Alejandro Vergara, Dries Raymaekers, Frank Montenegro, Walter Ocimati, Henry Alonso Ruiz, Jules Ntamwira, Aman Bonaventure Omondi, Guy Blomme, 2020, Detection of banana plants and their major diseases through aerial images and machine learning methods: A case study in DR Congo and Republic of Benin.

[17] Aldhyani, T.H. Alkahtani, H. Eunice, R.J. Hemanth, 2022, Leaf Pathology Detection in Potato and Pepper Bell Plant using Convolutional Neural Networks.

[18] Panigrahi, K.P. Das, H. Sahoo, A.K. Moharana, 2020, Maize leaf disease detection and classification using machine learning algorithms.

[19] Mohsin Kabir, M. Quwsar Ohi, A. Mridha, M.F, 2020, A Multi-plant disease diagnosis method using convolutional neural network.

[20] Jadhav, S.B. Udupi, V.R. Patil, S.B, 2021, Identification of plant diseases using convolutional neural networks.