Automated Disease Detection in Crops

Group Members

- Adithyan Babu B20CS1104
- Jobin Soly Cleetus B20CS1130
- Sandeep Sebastian B20CS1157
- Sreelekshmi S P B20CS1160

Project Guide

- Ms. Dhanya L K
  Assistant Professor

Contents

- Introduction
- Motivation
- Literature Review
- Limitations of Existing approach
- Methodology
- Results
- Conclusion
- Future Scope

Introduction

- Agriculture faces significant challenges due to plant diseases, leading to economic losses and environmental impacts.
- Our project aims to address this by leveraging deep learning to develop a proactive disease detection system for farmers.

Motivation

- In India, where agriculture plays a pivotal role, modernizing crop management practices is crucial for improved yields and food security.
- By utilizing technology for timely disease identification, we aim to enhance crop health and contribute to sustainable agricultural practices.

LITERATURE REVIEW

Reference	Methodology	Pros	Cons
Abdulaziz Alharbi, Muhammad Usman Ghani Khan, and Bushra Tayyaba, 2023. Wheat Disease Classification Using Continual Learning. [5]	In this system **EfficientNet** is used for feature extraction.	• Good performance in terms of accuracy and computational cost. • Less amount of data required. • It is lightweight.	• Overfitting to support set. • Sensitivity to noise • Dependency on support set quality.
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)	• Detect both dense and small objects. • FPN helps in multi-scale feature extraction. • Higher accuracy	• Slow output prediction • 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 **ResNet-50**, as the core model for maize crop leaf disease detection and classification.	• High Accuracy • Robust to Real-World Conditions	• Practical limitations • Limited dataset

LITERATURE REVIEW

Reference	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**	• High Classification Accuracy • Efficiency • Visualization of Disease Detection	• Limited Dataset • 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 **CNN** to classify leaves as healthy or affected by diseases.	• Accessibility of Farmers • Large Dataset • Image Processing	• Data Quality • Data Imbalance • Hyperparameters
Jawad Hassan, Ali Ghulam, Ghulam Irtaza, 2021.Disease Identification using Deep Learning in Agriculture: A Case Study of Cotton Plant. [4]	**CNN** 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.	• High accuracy • Automated feature extraction • Transfer learning	• High data requirements • Lack of interpretability • Limited domain expertise • Only detection and no remedy

LITERATURE REVIEW

Reference	Methodology	Pros	Cons
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** for disease classification and leaf image segmentation	• High accuracy • Incorporates image segmentation models to improve disease localization precision.	• Limited dataset • Focus on a specific plant series • Potential lack of discussion on real-world implementation challenges and scalability
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 **Convolutional Neural Networks (CNNs)**, for tasks such as image classification, object detection, and natural language processing.	• High Classification Accuracy • Effective Data Expansion • Realism in Diverse Conditions	• Data Collection Challanges • Dataset Representativeness • 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.	• High accuracy • Automated feature extraction • Scalability	• High data requirements • Lack of interpretability • Overfitting and generalization issues

LITERATURE REVIEW

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

Limitations of existing approaches

- Many current methods for disease detection in crops rely heavily on manual inspection by farmers or agricultural experts.
- Lack of emphasis on early detection leading to missed opportunities for timely intervention, allowing diseases to spread unchecked and causing significant damage to crops.
- Traditional disease detection methods often lack scalability, making it difficult to monitor crops across large agricultural areas effectively.
- Many existing disease monitoring systems cannot provide real-time data on disease outbreaks.

Methodology

1. **Data Collection**
   - The primary dataset utilized in this project was sourced from Kaggle, specifically the ”PlantVillage” dataset along with other datasets.
   - For training purposes we take 20 diseases and mention 5 classes for healthy leaves and for training we have more than 500 images for each class (20 Diseases and 5 Healthy leaf).

Methodology

2. Tools Used

- The toolset consisting of Visual Studio Code (VSCode), Keras, and TensorFlow creates a comprehensive environment for efficient deep learning development.
- Flutter for the mobile application interface and Flask for the backend.

Methodology

3. Model Training

- The dataset is divided into training, validation, and testing sets for robust model development.
- Utilization of Convolutional Neural Networks (CNNs) and advanced architectures for effective disease detection in plant leaves.

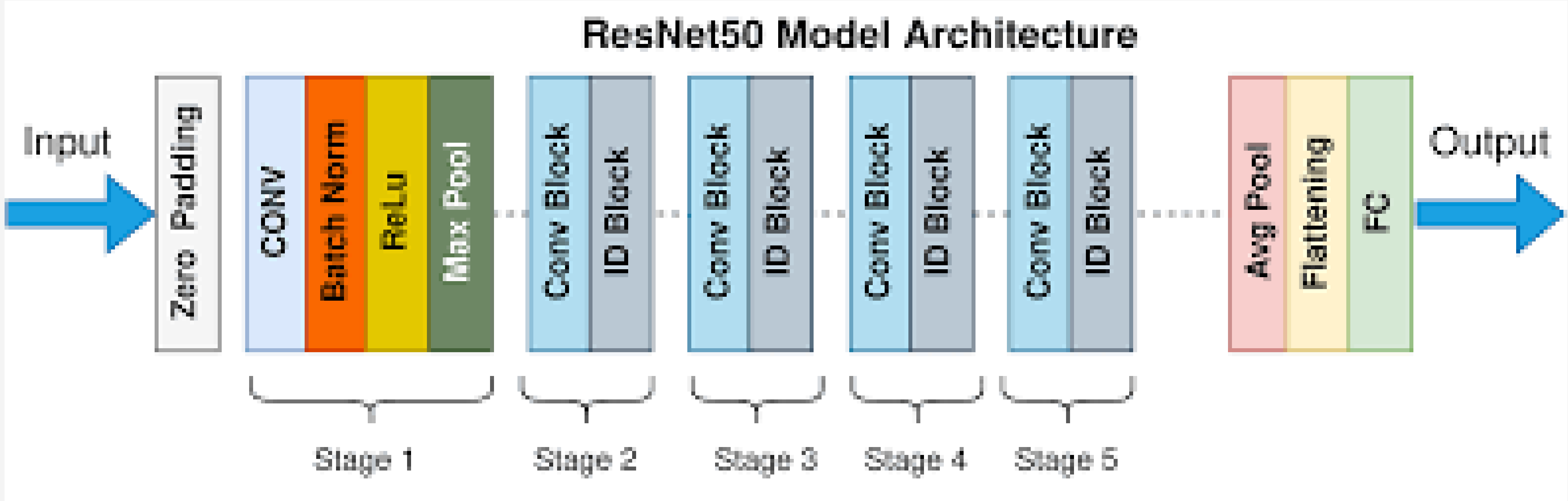
Convolutional Neural Network (CNN)

- CNN architecture facilitates efficient feature extraction and hierarchical learning.
- Sequential layers enable the comprehensive flow of information, including feature extraction, regularization, etc.
- An early stopping mechanism has been implemented to prevent overfitting and ensure efficient model training.

ResNet

- Implementation of simplified ResNet architecture with ResNet-like blocks for effective training.
- Utilization of residual connections and skip connections for gradient propagation and feature extraction.
- Model enhances capacity to extract meaningful features from input data, particularly in plant leaf disease classification.

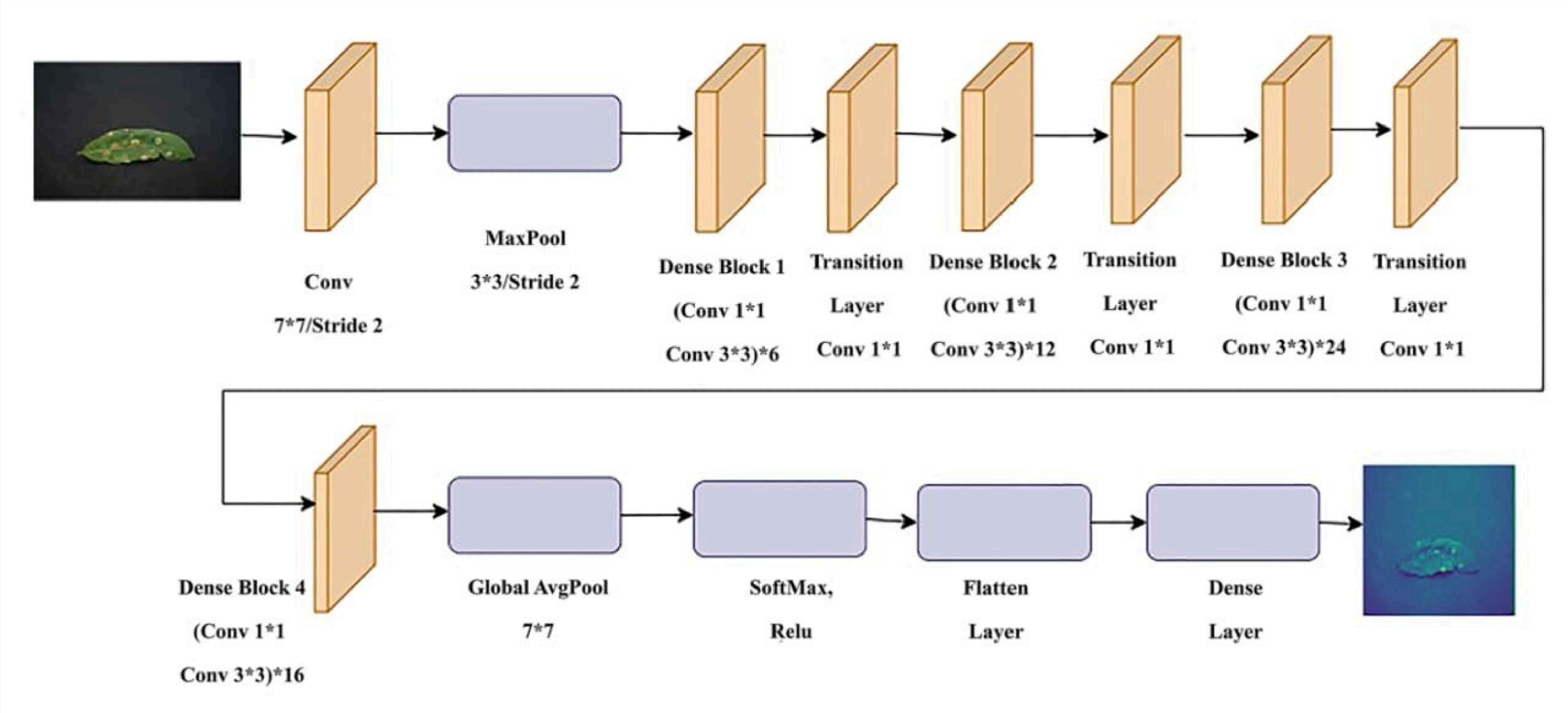
ResNet


ResNet50 Model Architecture
Input
Zero Padding
CONV
Batch Norm
ReLu
Max Pool
Conv Block
ID Block
Conv Block
ID Block
Conv Block
ID Block
Conv Block
ID Block
Avg Pool
Flattening
FC
Output
Stage 1
Stage 2
Stage 3
Stage 4
Stage 5

DenseNet121

- DenseNet121 initialized with weights from ImageNet and configured for our task.
- Addition of custom dense layers enhances model capabilities in discerning subtle patterns.
- Optimization with Adam optimizer and early stopping mechanism for optimal performance.

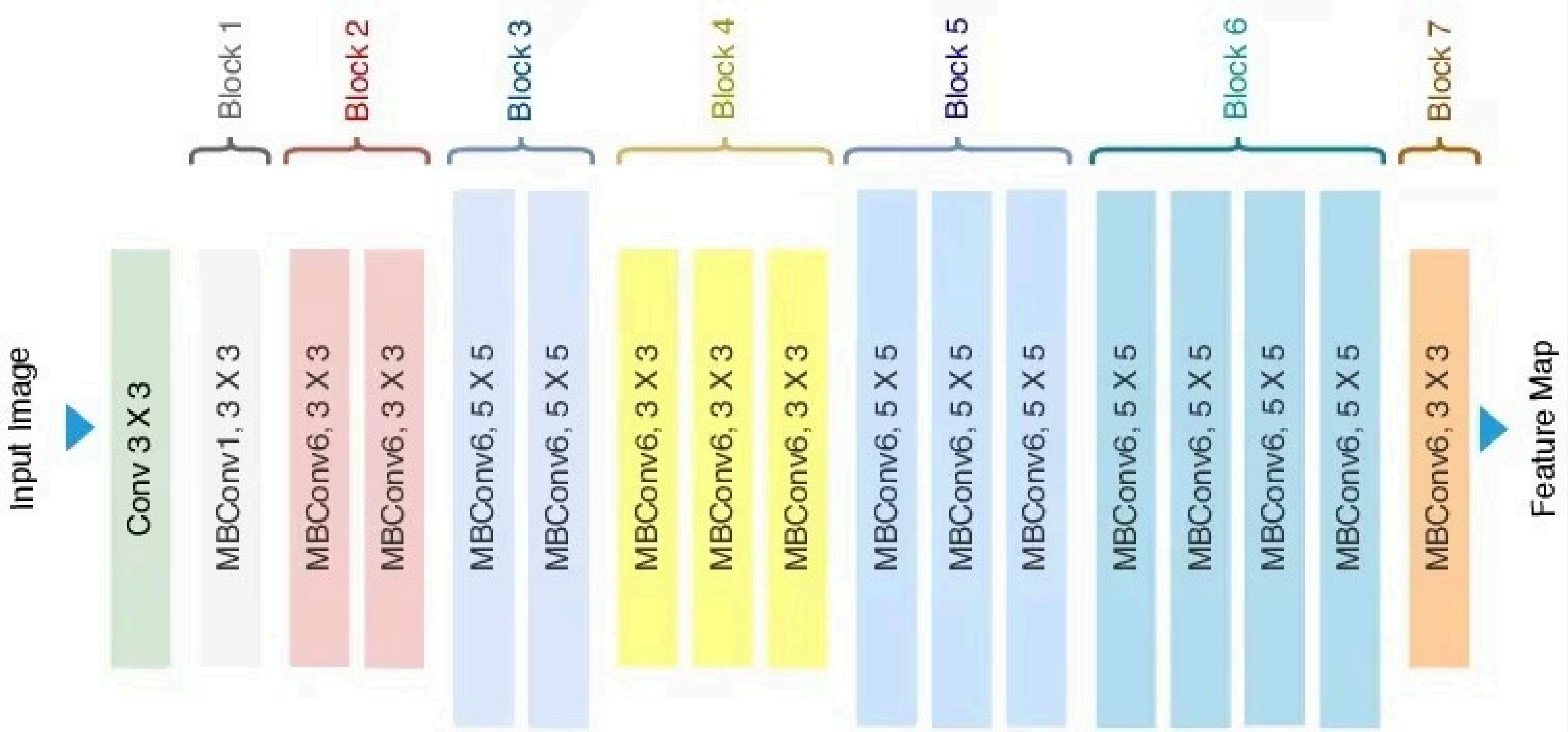
DenseNet121



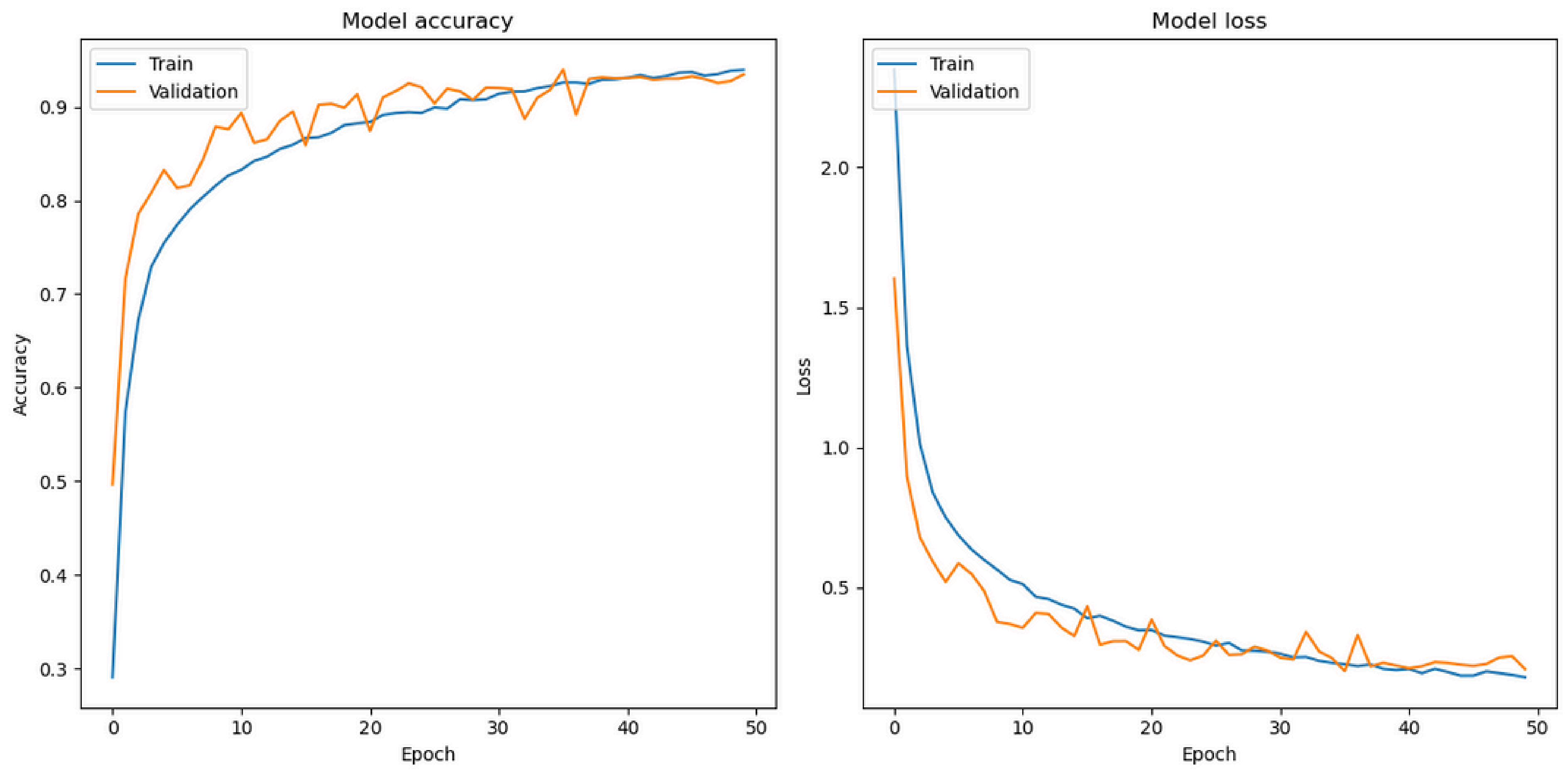
EfficientNet

- Utilization of EfficientNet architecture for optimized extraction of characteristics and learning in a structured system.
- Additional dense layers are added to enhance feature representation and facilitate multi-class classification.
- Configured with the Adam optimizer and categorical cross-entropy loss function.
- Utilizes callbacks such as ReduceLROnPlateau and EarlyStopping to dynamically adjust the learning rate and prevent overfitting during training.

EfficientNet

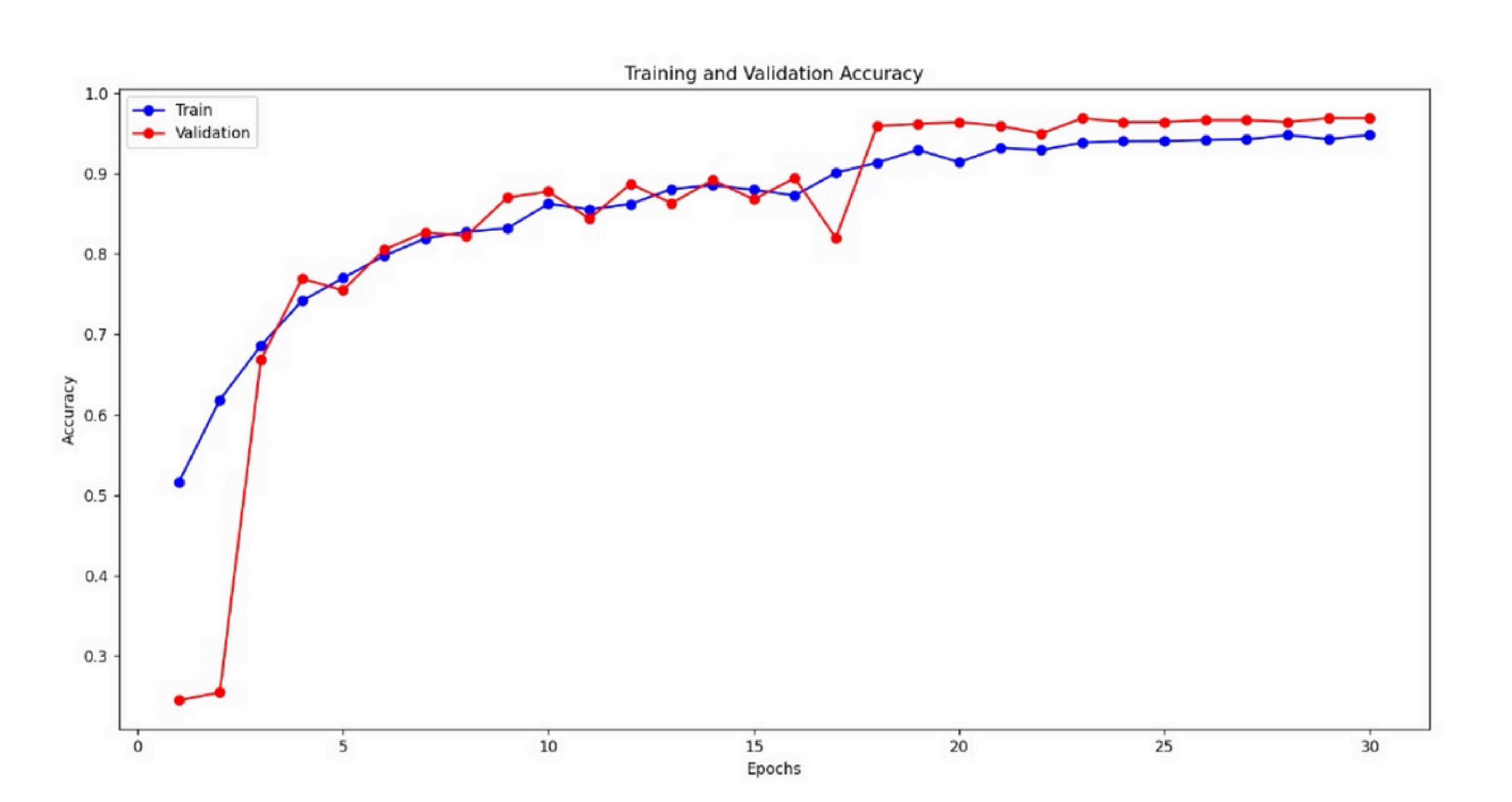


Result (CNN)



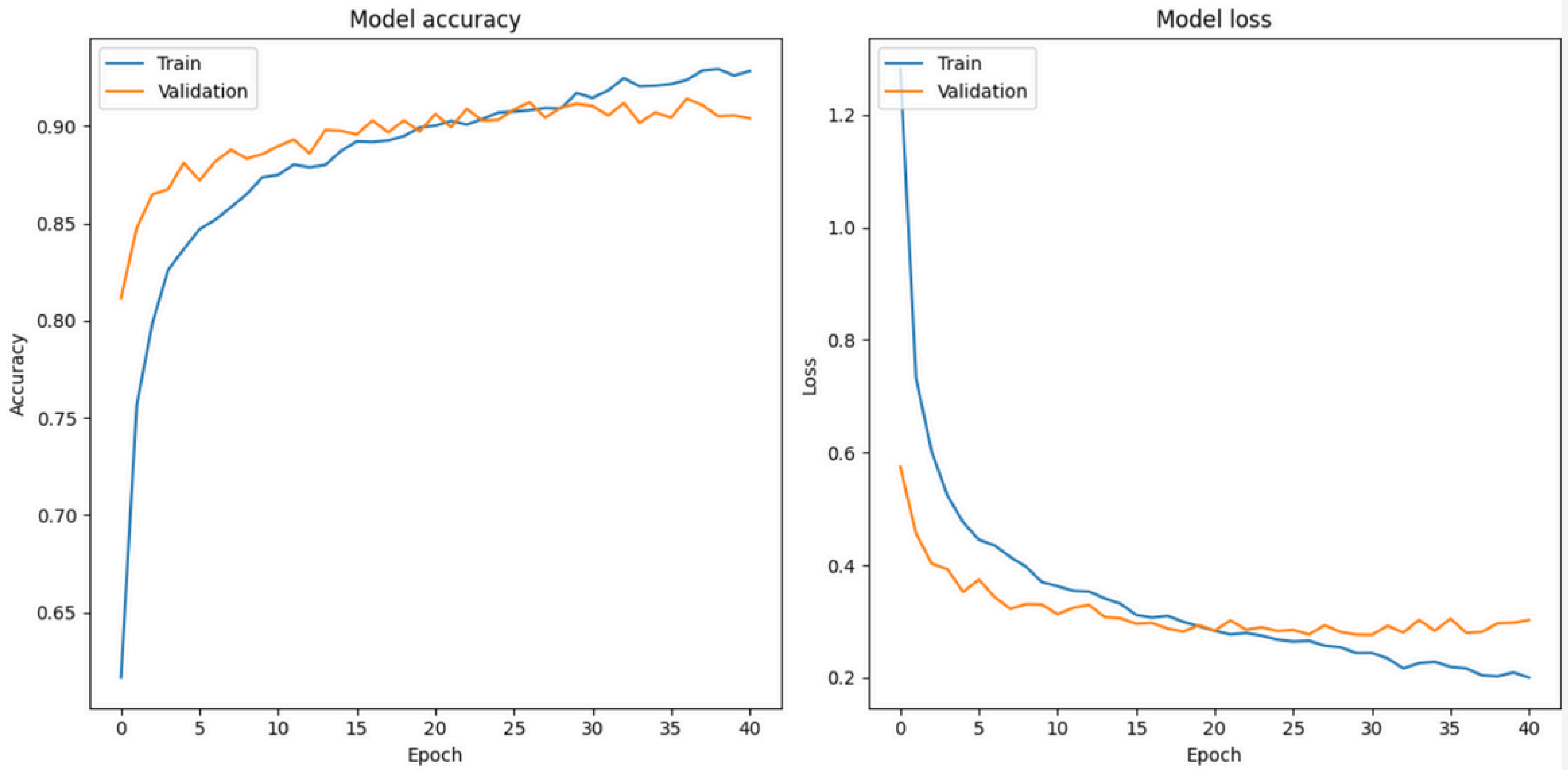

```
84/84 [==============================] - 1s 9ms/step - loss: 0.1974 - accuracy: 0.9395
84/84 [==============================] - 1s 9ms/step - loss: 0.1974 - accuracy: 0.9395
Test Loss: 0.19743873178958893
Test Accuracy: 0.9394736886024475
84/84 [==============================] - 1s 9ms/step
Confusion 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
```

Result (ResNet)



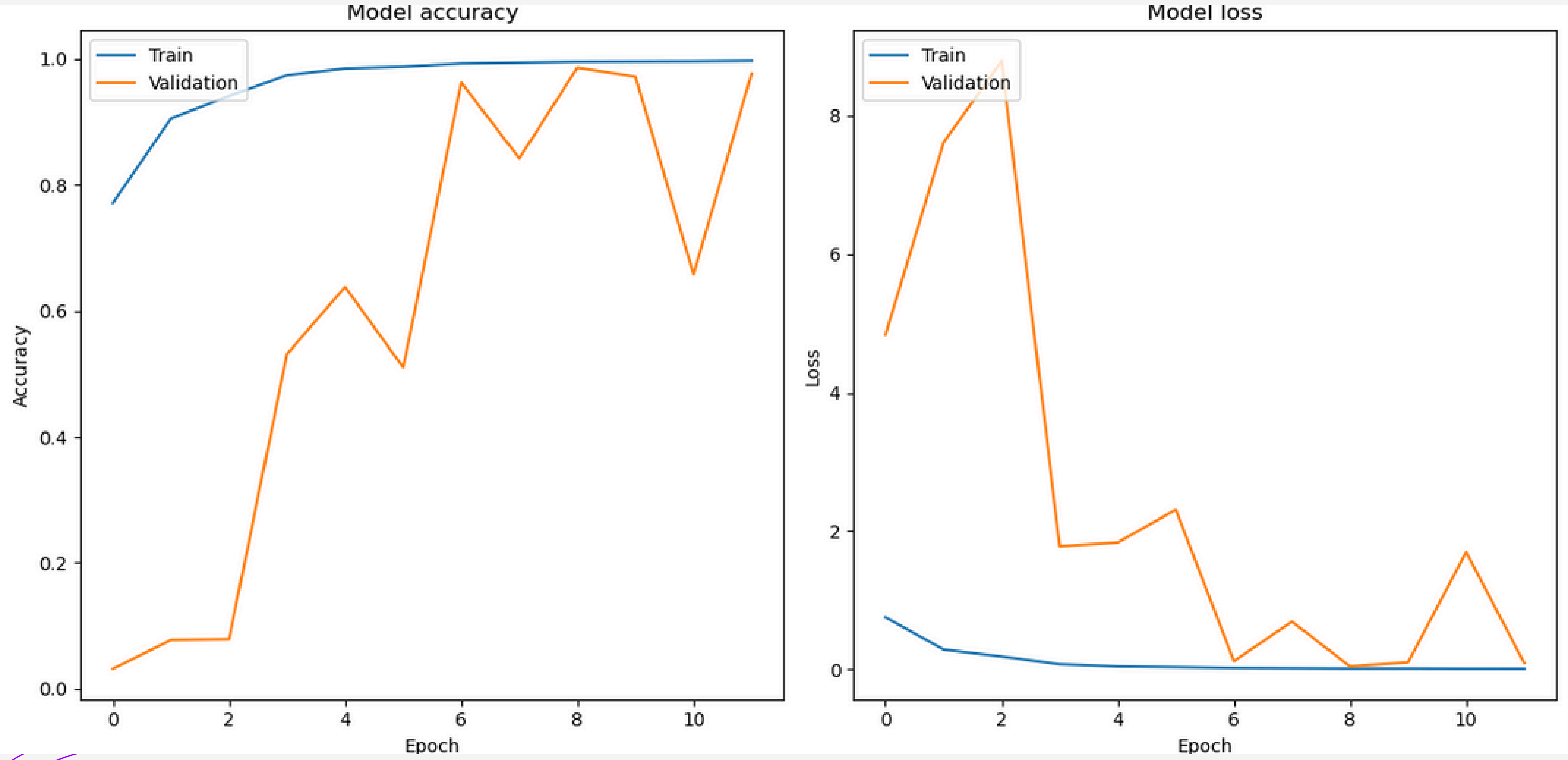

```
84/84 [==============================] - 6s 56ms/step
Confusion Matrix:
[[145   1   0  15   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0 167   1   2   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0]
 [  0   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
...
```

Result (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
```

Result (EfficientNet)



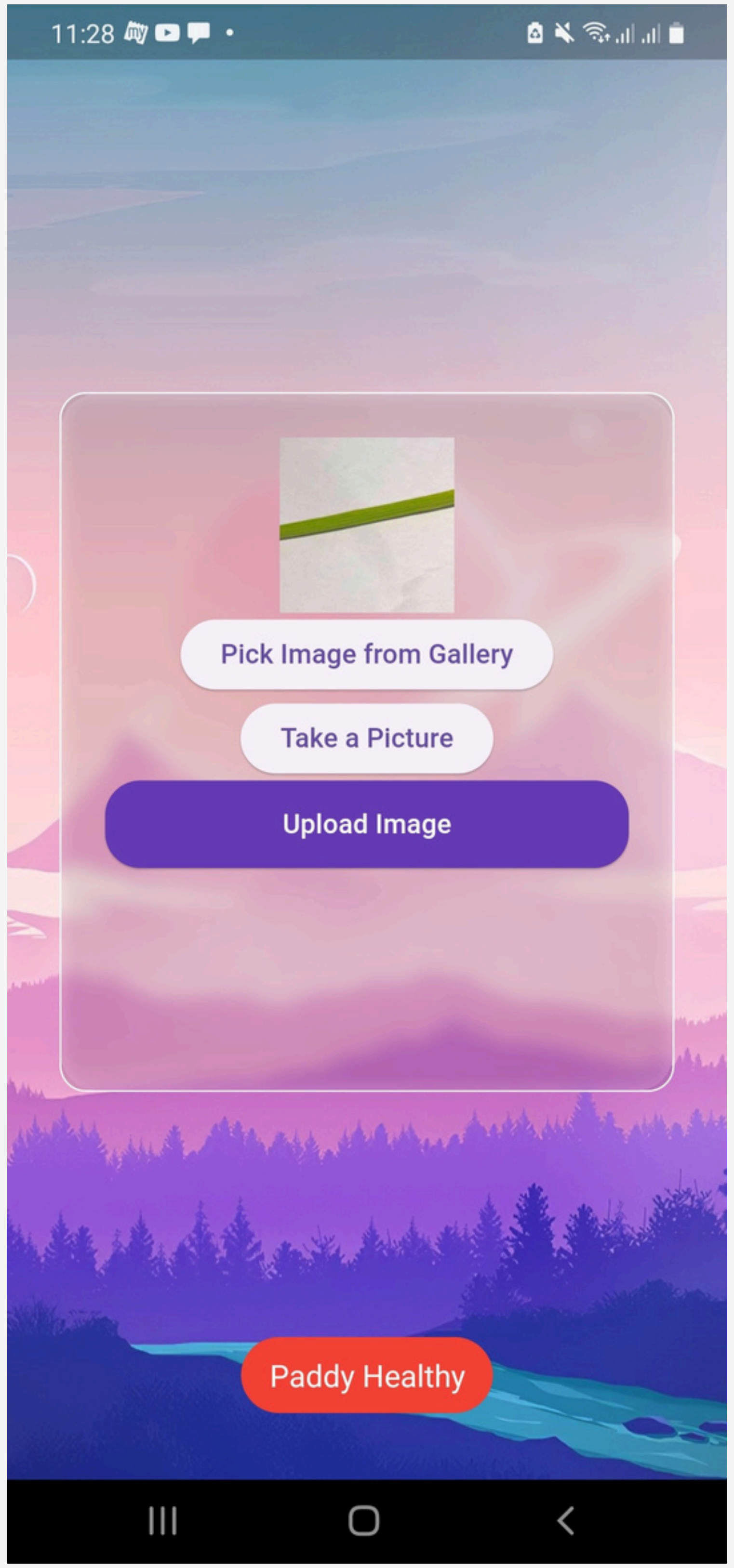

```
84/84 [==============================] - 10s 100ms/step - loss: 0.0629 - accuracy: 0.9797
84/84 [==============================] - 8s 100ms/step - loss: 0.0629 - accuracy: 0.9797
Test Loss: 0.06291396915912628
Test Accuracy: 0.9796992540359497
84/84 [==============================] - 10s 98ms/step
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
```

Performance Comparision

Architecture	Training Accuracy	Testing Accuracy
CNN	91.76	93.95
EfficientNet	**98.83**	**97.97**
DenseNet	92.84	91.61
ResNet	52.94	35.28

App

- **User-Friendly Interface**: The app features a simple and intuitive interface allowing users to easily capture or upload images from the gallery or camera for disease detection, enhancing accessibility for farmers with varying levels of technical expertise.
- **Real-Time Prediction**: Utilizing Flask and Flutter, the app delivers real-time predictions of crop diseases, providing instant feedback to users on the health status of their crops and enabling timely interventions to prevent further damage.
- **Seamless Integration:** Through seamless integration between the Flask backend and Flutter frontend, the app ensures smooth communication and data transfer, facilitating efficient processing of image data and accurate disease classification without significant latency.
- **Scalability and Customization**: Built with scalability in mind, the app can easily accommodate future updates and enhancements, such as incorporating additional disease detection models or integrating with external agricultural modules for comprehensive crop management solutions.



Conclusion

- Utilizing models like EfficientNet, DenseNet, and ResNet for plant leaf disease detection offers farmers a reliable tool for timely interventions.
- By selecting the most proficient model through comparative analysis, higher accuracy and adaptability across different crops and diseases are ensured.
- The chosen model enables swift disease detection, empowering farmers to implement targeted measures, optimize resource allocation, and prevent yield losses. This proactive approach enhances agricultural productivity, leading to improved crop yields and sustainable farming practices.

Future Scope

- Future advancements in automated plant disease detection systems involving detailed descriptions of detected diseases to farmers.
- Additionally, recommending specific cures and treatments based on the detected disease can help in decision-making for farmers.
- Integration with other agricultural modules, such as pesticide recommendation systems and irrigation management platforms

Reference

[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.

Reference

[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.

Reference

[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.

Reference

[16] 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.

Reference

- Md. Tariqul Islam, 2022, Plant Disease Detection using CNN Model and Image Processing.
- Muhammad E. H. Chowdhury, Tawsifur Rahman, Amith Khandakar, Mohamed Arselene Ayari, 2021, Automatic and Reliable Leaf Disease Detection Using Deep Learning Techniques
- Jinzhu Lu, Lijuan Tan, Huanyu Jiang, 2021, Review on Convolutional Neural Network (CNN) Applied to Plant Leaf Disease Classification.
- Yan Guo, Jin Zhang, Chengxin Yin, Xiaonan Hu, 2020,Plant Disease Identification Based on Deep Learning Algorithm in Smart Farming.

THANK YOU