

MaldoNetB0 - Enhanced EfficientNetB0 for Disease Detection in Apple Leaves (Malus Domestica) Using Optuna with Bayesian Optimization (TPE)

A PROJECT REPORT

Submitted by,

SHREYAS D M	20211CSE0381
ANIKA NAYAK	20211CSE0263
SHREE BHUVAN S R	20211CSE0231
UDEEP S LOKESH	20211CSE0355

Under the guidance of,

Prof. Tintu Vijayan

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

At



PRESIDENCY UNIVERSITY

BENGALURU

DECEMBER 2024

PRESIDENCY UNIVERSITY

SCHOOL OF COMPUTER SCIENCE ENGINEERING

CERTIFICATE

This is to certify that the Project report “**MaldoNetB0 - Enhanced EfficientNetB0 for Disease Detection in Apple (Malus domestica) Using Optuna with Bayesian Optimization (TPE)**” being submitted by **SHREYAS D M, ANIKA NAYAK , SHREE BHUVAN S R, UDEEP S LOKESH** bearing roll number(s) **20211CSE0381, 20211CSE0263, 20211CSE0231, 20211CSE0355** in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering is a Bonafide work carried out under my supervision.

Prof. Tintu Vijayan

Assistant Professor
School of CSE&IS
Presidency University

Dr. Asif Mohamed H B

Associate Professor & HoD
School of CSE&IS
Presidency University

Dr. L. SHAKKEERA

Associate Dean
School of CSE
Presidency University

Dr. MYDHILI NAIR

Associate Dean
School of CSE
Presidency University

Dr. SAMEERUDDIN KHAN

Pro-Vc School of Engineering
Dean -School of CSE&IS
Presidency University

PRESIDENCY UNIVERSITY

SCHOOL OF COMPUTER SCIENCE ENGINEERING

DECLARATION

We hereby declare that the work, which is being presented in the project report entitled **MaldoNetB0 - Enhanced EfficientNetB0 for Disease Detection in Apple (Malus domestica) Using Optuna with Bayesian Optimization (TPE)** in partial fulfillment for the award of Degree of **Bachelor of Technology in Computer Science and Engineering**, is a record of our own investigations carried under the guidance of **Prof. Tintu Vijayan, Assistant Professor, School of Computer Science Engineering & Information Science, Presidency University, Bengaluru.**

We have not submitted the matter presented in this report anywhere for the award of any other Degree.

Shreyas D M	ANIKA NAYAK	SHREE BHUVAN S R	UDEEP S LOKESH
20211CSE0381	20211CSE0263	20211CSE0231	20211CSE0355

ABSTRACT

This research enhances the detection of diseases in plant leaves with deep learning, represented through EfficientNetB0 for feature extraction, combined with Optuna for hyperparameter optimization. EfficientNetB0 was chosen as it is an advanced state-of-the-art model that can present intricate features without compromising much in computational aspects. It has served as the backbone for classifying images of various kinds of plant leaves into different disease categories. In order to make the prediction performance as accurate as possible, Optuna was applied to tune some key hyperparameters, such as the learning rate and dropout rate. After the optimization of these critical parameters, a good balance was achieved between high validation accuracy and computation time reduction. To prevent overfitting and ensure the model generalizes well, early stopping was employed. The training was stopped when performance on the validation set had reached a plateau to prevent extra computation and overfitting. Learning rate scheduling further improved convergence and allowed the model to reach its optimal performance smoothly.

This model, MaldoNetB0, had improved training accuracy from 68.69% to 97.37% within 15 epochs. Also, its validation accuracies were running stably between 98% and 99%. The trials from Optuna were really valuable, as the optimization yielded more than 94% validation accuracy with a reduced loss of 0.1382. The most striking result is surely that of MaldoNetB0, recording an AUC of 0.9998 and therefore how it could predict the diseased versus healthy leaves with almost perfect precision. Balanced precision and recall further reinforced the robustness of the model, hence making it very effective for plant disease detection on various datasets. All this, nonetheless, does have a great deal of agricultural applications. Early disease diagnosis helps in the development of plant health, providing scalability in crop management that increases the yield. In this implementation, EfficientNetB0 has merged with Optuna and practical optimization strategies-optimal results that will guarantee, that the tool is very accurate, efficient, and impactful for precision agriculture. Techniques like this can bring about revolutions in disease detection and thereby help in laying a strong foundation for advanced agriculture technology.

ACKNOWLEDGEMENT

First of all, we indebted to the **GOD ALMIGHTY** for giving me an opportunity to excel in our efforts to complete this project on time.

We express our sincere thanks to our respected dean **Dr. Md. Sameeruddin Khan**, Pro-VC, School of Engineering and Dean, School of Computer Science Engineering & Information Science, Presidency University for getting us permission to undergo the project.

We express our heartfelt gratitude to our beloved Associate Deans **Dr. Shakkeera L** and **Dr. Mydhili Nair**, School of Computer Science Engineering & Information Science, Presidency University, and **Dr. Asif Mohamed H B**, Head of the Department, School of Computer Science Engineering & Information Science, Presidency University, for rendering timely help in completing this project successfully.

We are greatly indebted to our guide **Prof. Tintu Vijayan**, Assistant Professor and Reviewer **Prof. Amarnath J L**, Assistant Professor, School of Computer Science Engineering & Information Science, Presidency University for his inspirational guidance, and valuable suggestions and for providing us a chance to express our technical capabilities in every respect for the completion of the project work.

We would like to convey our gratitude and heartfelt thanks to the PIP2001 Capstone Project Coordinators **Dr. Sampath A K**, **Dr. Abdul Khadar A** and **Mr. Md Zia Ur Rahman**, department Project Coordinators **Amarnath J L** and Git hub coordinator **Mr. Muthuraj**.

We thank our family and friends for the strong support and inspiration they have provided us in bringing out this project.

Shreyas D M

Anika Nayak

Shree Bhuvan S R

Udeep S Lokesh

LIST OF TABLES

Sl. No.	Table Name	Table Caption	Page No.
1	Table 1.1	Literature Review table	11-14

LIST OF FIGURES

Sl. No.	Figure Name	Caption	Page No.
1	Figure 1.1	The Architecture of the MaldoNetB0	19
2	Figure 1.2	The EfficientNet-B0 architecture	20
3	Figure 1.3	Model selection during prototyping	22
4	Figure 1.4	Neural network model design.	24
5	Figure 2.1	Gantt Chart	31
6	Figure 3.1	Analyzing AUC and Precision Metrics on Training	32
7	Figure 3.2	Precision Metric Over Training Analysis	34
8	Figure 3.3	Analysis of F1 Score Metric Over Training	37
9	Figure 3.4	Training and Validation Accuracy Over 15 Epochs	39
10	Figure 4.1	Datasets Images	55
11	Figure 4.2	Results	55

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	4
	ACKNOWLEDGMENT	5
	LIST OF TABLES	6
	LIST OF FIGURES	7
1.	INTRODUCTION	9-10
2.	LITERATURE REVIEW	11-14
3.	RESEARCH GAPS OF EXISTING METHODS	15-17
4.	PROPOSED METHODOLOGY	18-25
5.	OBJECTIVES	26
6.	SYSTEM DESIGN AND IMPLEMENTATION	27-30
7.	TIMELINE FOR EXECUTION OF THE PROJECT	31
8.	OUTCOMES	32-41
9.	RESULTS AND DISCUSSION	42-44
10.	CONCLUSION	45-48
	REFERENCES	49-50
	APPENDIX-A PSUEDOCODE	51-52
	APPENDIX-B SCREENSHOTS	53
	APPENDIX-C ENCLOSURES	54-57

CHAPTER-1

INTRODUCTION

Agriculture is the backbone of many economies around the world and vital to the food security of a fast-growing world population. This sector, though, is faced with several problems that are threatening productivity. Among those, plant diseases are the most serious concern. It results in reduced yields and economic losses while affecting quality. This requires an early and precise diagnosis.

Plant disease identification traditionally involves expertise, hence the manual inspection process, which is very time-consuming and not scalable. This might work for small-scale farming, but it doesn't apply to large-scale farming, especially where there is a lack of skilled agricultural personnel. Innovations of scalable solutions are in great demand to efficiently support farmers with plant health management. Recent technology has thrown up faster and more accurate methods of detection. Image processing, combined with machine learning and deep learning, has emerged as a robust solution for plant disease detection.

1.1 Image Processing in Plant Disease Detection

It involves the processing of digital images of plant leaves to precisely locate the diseases with very minimal interference from humans. It is a systematically structured approach around several stages.

1.1.1 Stages of Image Processing in Disease Detection

- **Image Acquisition and Preprocessing:** High-quality images of the plant leaves are taken and enhanced for better quality.
- **Segmentation of the affected area:** It reveals the isolation of the diseased portion of a leaf for the assurance of disease-specific focused patterns.
- **Feature Extraction:** Identification and extraction of critical patterns that can provide meaningful insights.
- **Classification:** After feature extraction, the features extracted are classified based on the disease type present in the plant.

1.2 Role of Deep Learning Models

Deep learning models, such as Maldo-NetB0, are important for improving the accuracy of plant disease detection. They systematically analyze the images of leaves and yield high-accuracy results.

1.2.1 Hyperparameter Optimization

Optimization techniques applied to deep learning models, like Optuna, will fine-tune them to their best possible performances. This optimization technique optimizes hyperparameters related to higher computational efficiency and accuracy, such as learning rates or dropout rates.

1.2.2 Data Augmentation Techniques

Some of the techniques used in making models invariant to changes include rotation, scaling, and flipping. These methods enrich the training datasets, hence making the model perform effectively in real-world scenarios.

1.3 Metaheuristic Optimisation Algorithms

Nature-inspired metaheuristic optimization algorithms are applied to improve different stages of the image processing chain, such as:

1. Feature Selection: The most important feature in disease detection.
2. Classifier Tuning: The process of optimizing a classification model to yield the best results.

Integration of Image Processing and Optimization Merging image processing with deep learning and optimization techniques is epoch-changing in plant disease detection. These systems allow not only the early and quite precise identification of diseases but also their targeted interventions and pesticide usage, thus environmentally friendly farming. Future Directions in the Detection of Diseases in Plants Such advancements in this area could bring a revolution in crop management systems. Scalable and automated solutions can help farmers, economic losses can be reduced, and there is a positive contribution to food security at the global level. This research was a big step toward intelligent agricultural systems, helping to alleviate the challenges posed by traditional farming while enhancing productivity and sustainability

CHAPTER-2

LITERATURE SURVEY

1.1 Literature Survey table

Papers	Advantages	Disadvantages
1. Hungilo GG, Emmanuel G, Emanuel AW (2019) Image processing techniques for detecting and classification of plant disease:	It will increase accuracy in disease diagnosis, be more economically feasible for a farmer, optimize the pesticide usage, therefore, reducing not only the environmental cost but most importantly, the cost in general.	It requires advanced technical knowledge and expertise to maintain, and can be far too expensive for a small-scale farmer.
2. Deepak AH, Gupta A, Choudhary M, Meghana S (2019) Disease detection in tomato plants and remote Monitoring of agricultural parameters. In: 2019 11th International conference on advanced computing (ICoAC)	High levels of accuracy in disease classification, remote monitoring and management, productivity maximized, resource saving.	It needs preliminary investment in sensor technology, and it may also require technical expertise in its setting and maintenance.
3. Hidayatuloh A, Nursalman M, Nugraha E (2018) Identification of tomato plant diseases by leaf image using squeeze net model. In: 2018 International conference	CNN does not need hand-crafted feature engineering since it can learn complex features automatically in the training process.	Preprocessing may be time-consuming and require expertise. The integration of CNN models into broader frameworks, such as

on information technology systems and innovation (ICITSI).	High accuracy of classification reflected the efficiency of deep learning for agriculture.	IoT-based smart farming systems, may be hard to implement and operate.
4. Dhingra G, Kumar V, Joshi HD (2018) Study of digital image processing techniques for leaf disease detection and classification. Multimedia Tools Apple.	Provides an overview of contemporary machine learning and image processing methodologies. It opens up possibilities for improvement in the image-based plant disease detection systems.	Possible issues in natural light conditions may affect the performance of detection. 2. It requires intensive preprocessing techniques and lighting compensation methods to perform well under variable field conditions.
5. Sandika B, Avil S, Sanat S, Srinivasu P (2016) Random forest-based classification of diseases in grapes from images captured in uncontrolled environments. In: 2016 IEEE 13th international conference on signal processing (ICSP).	High Classification Accuracy of 86% on the Random Forest, effective identification of diseases in grape leaves; easily scalable to any crop for health monitoring	It requires an appropriate choice of image processing techniques and machine learning models; further studies may be required to address some challenges such as environmental changes.

6. Zawbaa, HM, Hazman M, Abbass M, Hassanien AE (2014) Automatic fruit classification using random forest algorithm. In: 2014 14th International conference on hybrid intelligent systems.	High accuracy, computational efficiency	Requires large-scale datasets for optimal performance
7. Iqbal MA, Talukder KH (2020) Detection of potato disease using image segmentation and machine	High accuracy of 97%, efficiency by automation, applicable for a wide range of plant species.	Less effective as compared to the Random Forest algorithm; it could be demanding and involve complex technical expertise.
8. Chauhan MD (2021) Detection of maize disease using random forest classification algorithm. Turkish J Comput Math Education.	High accuracy in disease detection, early identification for timely intervention.	The difficulty in choosing and training the model, maybe some limitations in detecting infrequent or changing diseases.
9. Hidayatuloh A, Nursalman M, Nugraha E (2018) Identification of tomato plant diseases by leaf image using squeeze net model. In: 2018 International conference on information technology	Disease detection automatically Real-time plant health	SVM manual feature extraction Difficulty in handling big datasets

systems and innovation (ICITSI).		
10. Sarangdhar AA, Pawar VR (2017) Machine learning regression technique for cotton leaf disease detection and controlling using IoT. In: 2017 International conference of electronics, communication and aerospace technology (ICECA), pp 449–454.	Plant diseases can be detected effortlessly; recommendations of pesticides will be automated.	Lack of technical acumen, increased probationary implementation cost of IoT sensors
11. Khirade SD, Patil AB (2015) Plant disease detection using image processing. In: 2015 International conference on computing communication control and automation	Efficient alternative to manual monitoring. It facilitates early disease detection and thus provides an opportunity for timely intervention	Requires experience in image processing. Quality of image acquisition-dependent

CHAPTER-3

RESEARCH GAPS OF EXISTING METHODS

1. Limited Integration of Techniques

- **Fragmented Approaches:** Most of the existing methods do not combine image processing, deep learning, and optimization techniques in a single system. The resultant detection pipeline turns out to be fragmented, ultimately leading to reduced overall accuracy and reliability.
- **Poor Data Fusion:** Most of the existing systems suffer from integrating environmental data and various soil conditions in their study case with the outcomes of leaf image analysis for possible improvement of accuracy during disease detection or prediction.

2. Inadequate Utilization of Real-Time Data

- **Delayed Detection:** Most of the current techniques rely on batch processing of images; hence, there is a delay in disease detection and less effectiveness with timely interventions.
- **Dynamic Adaptation:** Very few systems perform updates in real-time for active learning, such as identifying new variants of ailment or evolving pathogen behaviors.

3. Limited Adoption of Advanced Technologies

- **Poor Models:** Most of the approaches make use of normal machine learning models without considering the advances in the field with models like EfficientNetB0 or Conditional GAN, which brings down precision and robustness.
- **Poor Optimization:** Most of the classic systems use manual or basic hyperparameter tuning, which ultimately results in poor model performance. Most sophisticated tools, such as Optuna or metaheuristic algorithms, are seldom

used.

4. Data Scarcity and Augmentation

- **Limited diversity in the datasets:** Most of the datasets are small and not very diverse, leading to poor generalization performance of models in real-world scenarios.
- **Insufficient Augmentation:** None of the previous approaches considered data augmentation, such as rotation, scaling, and flipping, to handle leaf orientation and environmental variations.

5. Privacy and Security Issues

- **Data Protection:** Most of the systems dealing with data about farmers and crops have loopholes in implementing strong privacy mechanisms to protect against data protection regulations.
- **Security Risks:** Cloud-based solutions run the risk of exposing sensitive data to unauthorized access or even cyber-attacks.

6. Limitations of Environmental and Resource Constraints

- **Energy Inefficiency:** Most of the deep learning models are computationally too expensive, which is not feasible for broad-scale agricultural purposes.
- **Pesticide overuse:** It is also a concern whereby the existing systems do not have mechanisms that would optimize the use of pesticides according to the precise detection of a disease, thus causing harm to the environment.

7. Limited Data Analytics and Insights

- **Lack of Predictive Insights:** Indeed, the traditional ways lack such

utilization of gathered data in predicting outbreaks or analysis of patterns and trends for long-term management of health in crops.

- **Analytics that Farmer Centric:** None or very few focus on actionable insights into farmers, like cost-benefit analysis or region-specific disease trends.
- **Barriers to Adoption Farmer Resistance:** Most farmers, particularly in developing areas, are somewhat resistant to digital solutions, whether due to perceived complexity or high costs.
- **Gaps in Policy and Awareness:** The inability to treat proper awareness and supportive policy has massively hindered the extensive adaptation of advanced plant disease detection systems.

CHAPTER-4

PROPOSED METHODOLOGY

Fig. 3.1 describes the architecture of MaldoNetB0 optimized by Optuna for disease detection in the leaf area of any plant. It takes an RGB image of size 224x224 as input and passes it through several convolution blocks. Each block reduces the spatial resolution of the input with increased depth for feature representations. This enables the network to extract a hierarchy of features ranging from low-level textures and edges in the shallow layers to high-level, abstract representations in the deeper layers. These features, which range from P3 to P7, are then processed across different levels of the network to ensure comprehensive comprehension of the input image. These become one-dimensional vectors after flattening the generated spatial feature maps and are taken through a series of fully connected layers to make predictions.

The final classification layer identifies the species of the plant and the associated disease. Such hyperparameters specifically the learning rate and dropout rate using Optuna plays an important role in enhancing precision and efficiency for the model. The fine-tuned learning rate ensures faster convergence and better model performance during training. Convolutional Neural Networks constitute the backbone of MaldoNetB0. This works especially well for CNNs in tasks involving images, since these models have the capability to preserve spatial hierarchies through convolution operations. Techniques such as pooling layers help reduce the dimensions and, hence, keep the network computationally efficient while retaining important features. The utilization of filters in convolution layers ensures that the model autonomously learns to identify key patterns in the image features themselves without human bias and thus easily accommodates variability in plant leaf shapes, color, and textures. Moreover, this architecture, combined with Optuna's hyperparameter tuning, ensures that

MaldoNetB0 excels in the extraction of intricate details about plant leaf disease classification with spectacular accuracy.

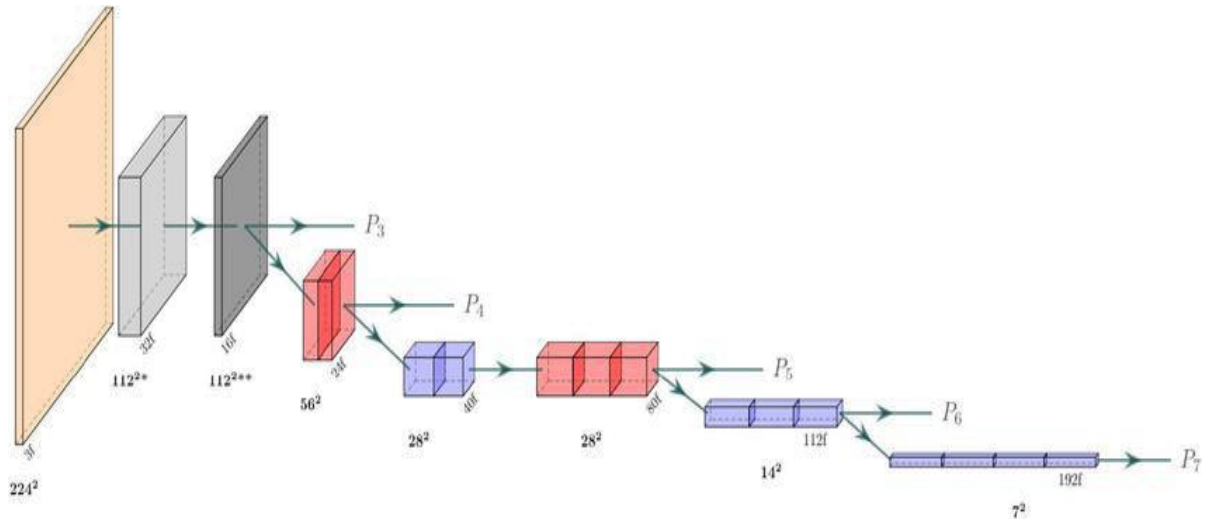


Fig 1.1 The architecture of the EfficientNetB0 backbone

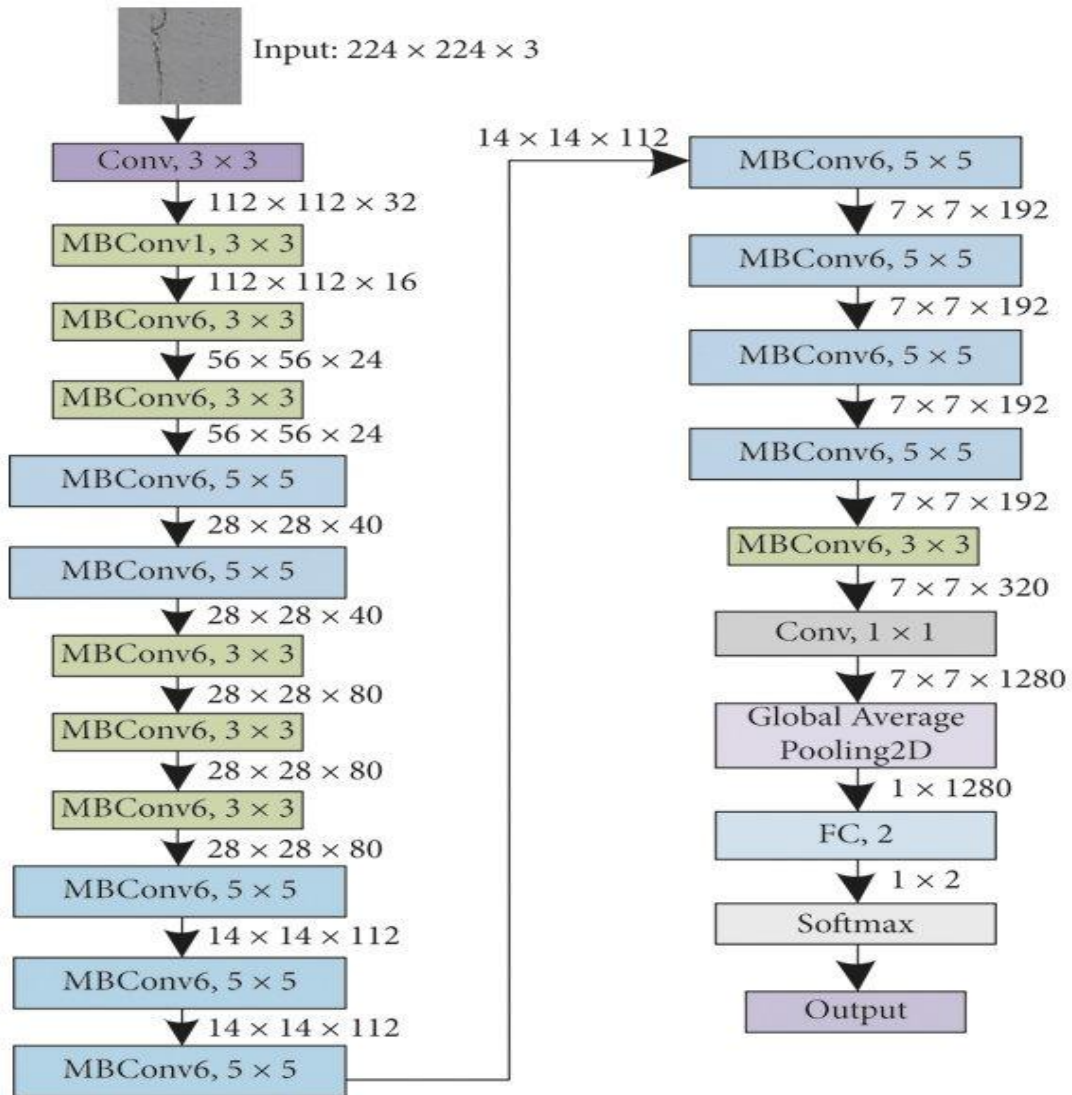


Fig 1.2 The EfficientNet-B0 architecture

EfficientNet-B0 is one such convolution neural network architecture because it was designed to classify an image efficiently hence striking a reasonable balance between accuracy and cost of computation. For how this works consider, for example, a dataset of 7,000 images all sized at $224 \times 224 \times 3$ -that means the height, width, and three color channels for RGB and normally processed by batches, like size 32-32 passed through the net once. Every image is then resized to 224×224 pixels to ensure regularity for the same network input.

The network is initialized with a convolutional layer with 3×3 kernels, 32 filters,

and a spatial dimension of 224×224 to 112×112 while increasing the depth to 32 channels. This layer extracts low-level features from the input images, such as edges and textures. The network is then followed by a set of mobile inverted bottleneck convolution or MBConv-blocks, which make up the backbone of EfficientNet-B0.

These blocks are made up of: an expansion phase that increases the number of input channels by a factor (usually 6) so that large, contiguous areas of detailed features can be captured; a depthwise convolution where each channel is processed separately, reducing computational cost; and a projection phase that uses a 1×1 convolution to decrease the channels back to the original count, or somewhat more, such that essential features are maintained.

These are images that continue to progress in a network with continually reduced spatial dimensions but with increased channels for feature capturing. For instance, the first few MBConv blocks could reduce dimensions down to 56×56 and increase the number of channels to 24; the dimensions can then go further down to 28×28 while increasing channels to 40. The progressive shifting allows the model to start paying attention to higher abstraction from simple shapes and textures to an abstract representation of image content. EfficientNet-B0 also includes the use of bottleneck layers that employ 1×1 convolutions to squeeze and expand the feature maps. These are important steps since this is where most of the computational cost and, consequently, memory usage are reduced, thus making the network efficient without losing much performance. Once the images have been processed through the MBConv layers, a 1×1 convolution layer consolidates all of the extracted features into a more compact form. These feature maps, reduced to $14 \times 14 \times 112$, are processed through a Global Average Pooling-GAP- layer, which aggregates the information in space by averaging the values of each channel. This provides a 1×112 feature vector per image.

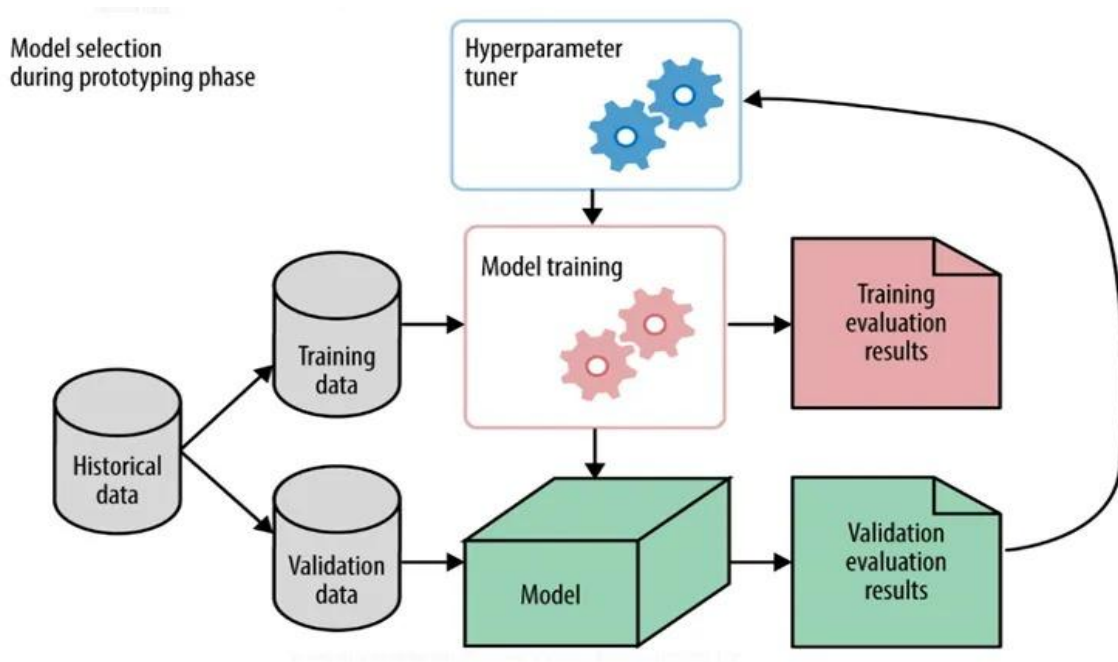


Fig 1.3 Model selection during prototyping

This diagram shows model selection during prototyping, using historical data split into training and validation sets. A hyperparameter tuner iteratively refines the model through training and evaluation. Training and validation results guide improvements in hyperparameters, creating a feedback loop. The process finalizes an optimized model for deployment or further testing.

Below is the hyperparameter tuning process using Optuna, a powerful automated optimization framework that works out of the box, especially during the rapid prototyping phases of model selection. The process starts with the historical data, which must be carefully divided into two subsets: the training data and the validation data. These are crucial bases for further model development and evaluation stages. The training data is used to train the machine learning model, while the validation data provides measures of the generalization ability of the already trained model by assessing its performance on unseen data. The central component of this process is a hyperparameter tuner, in this case,

Optuna. The tuner searches the space of possible hyperparameter values to find the best combination that will yield the best performance of the model. Unlike model parameters, hyperparameters are not learned from data during training. They are set before training and can have a significant impact on the model's performance and efficiency.

Examples of hyperparameters include the learning rate, batch size, and the number of layers in the neural network.

Optuna's hyperparameter tuner uses sophisticated techniques, such as Bayesian optimization, to efficiently search the hyperparameter space. This search is guided by the results of model training and evaluation. Initially, the tuner suggests a set of hyperparameters, which are used to train the model. The model training phase involves adjusting the model's parameters to minimize the loss function, which measures the difference between the predicted outputs and the actual labels in the training data. After training, the model is evaluated using the validation data to provide results such as accuracy, precision, recall, or more complex metrics like the F1-score, depending on the specific task. These results are returned to the tuner, which uses them to guide further searches.

If the evaluation results fail to demonstrate the optimality of the current set of hyperparameters, the tuner adjusts the hyperparameters, and the process repeats. This is an iterative process, which is the most important part. The tuner learns from previous trials and can focus the search on promising regions of the hyperparameter space, avoiding regions that have consistently yielded poor performance. This iterative feedback loop continues until the hyperparameter tuner identifies a set of hyperparameters that optimize the model's performance on the validation data.

Finally, after selecting the best hyperparameters, the model is retrained using the entire training data with these optimal settings. This final model undergoes rigorous evaluations to ensure good performance on the validation data and, most importantly, any new, unseen data. This evaluation helps confirm that the model

hasn't overfit the validation data and that it generalizes well across datasets. Overall, the image captures the essence of the hyperparameter tuning process facilitated by Optuna. It highlights the importance of data splitting, the role of iterative tuning and evaluation, and the ultimate goal of optimizing model performance. This approach not only helps in finding the best hyperparameters more efficiently but also ensures that the final model is robust and generalizes well, making it a critical component of modern machine learning workflows.

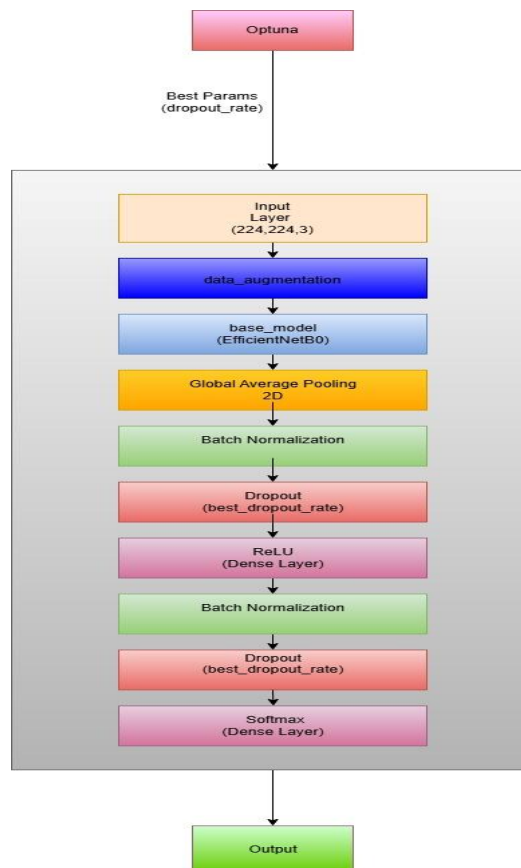


Fig 1.4 Neural network model design.

It starts with Optuna, which determines the best hyperparameters (e.g., dropout rate). The pipeline proceeds with the input layer (224, 224, 3), followed by data augmentation, a base model (EfficientNetB0), global average pooling, batch

normalization, dropout layers, a ReLU dense layer, and ends with a softmax dense layer for classification, producing the output.

CHAPTER-5

OBJECTIVES

5.1 Improved Plant Leaves Disease Detection:

This, in return, attempts to enhance disease detection in plant leaves using a deep learning method.

This includes leveraging state-of-the-art architectures to classify diseases in plant leaves with much accuracy and efficiency.

5.2 Extract Features Using EfficientNetB0:

1. EfficientNetB0 extracts complex features at a relatively low computational cost.
2. It forms the backbone of the model by extracting features from input images of plant leaves for their proper classification into disease categories.

5.3 Tune Hyperparameters with Optuna:

1. Key hyperparameters have been tuned like the learning rate and dropout rate by the Optuna optimization framework.
2. This optimization provides a trade-off between high validation accuracy and less computational time.

5.4 High Model Accuracy and Robustness:

1. MaldoNetB0 realized a significant gain in the increase of training accuracy by 68.69% to 97.37% in 15 epochs.
2. The validation accuracy is kept at 98% and 99%, while the loss value is 0.1382.
3. A very high AUC of 0.9998 depicts very good discrimination between the healthy and diseased leaves.

CHAPTER-6

SYSTEM DESIGN & IMPLEMENTATION

6.1 System Architecture Overview

The architecture of MaldoNet has four major parts:

1. Data Collect and Pre-processing Module
2. Architecture and Model Training Module
3. Hyperparameter Optimization Module
4. Evaluation and Deployment Module

6.2 System Design Breakdown

6.2.1 Data Collection and Preprocessing

Data Sources: Publicly available datasets that consist of labeled images of apple leaves related to different classes of diseases, such as

1. Apple Scab
2. Black Rot
3. Cedar-Apple Rust
4. Healthy

Preprocessing Steps:

1. Resizing images to 224x224 pixels because EfficientNetB0 expects this size as an input.
2. Normalization of the pixel values between [0 1] due to the uniformly represented input presentation.

Data augmentation techniques:

1. Rotation
2. Flipping
3. Zooming
4. Brightness adjustment

6.2.2 Model Architecture

Base Model: In the scope of transfer learning, MaldoNet relies on a pre-trained EfficientNetB0 as its feature extractor backbone.

Enhancements:

1. Integration of SE-Blocks: Squeeze-and-Excitation for improved attention to features.
2. Customizing Dense Layers for Multiclass Disease Classification.
3. Introduction of dropout layers to avoid overfitting.
4. Softmax in the last layer for multi-class disease detection.

6.2.3 Hyperparameter Optimization

Framework: Optuna - Bayesian optimization with TPE [Tree-structured Parzen Estimator].

Key-tuned Hyperparameters:

1. Learning Rate
2. Batch Size
3. Dense Layers and Number of Neurons
4. Dropout Rates
5. Objective Function: Minimize the loss for validation and maximize the accuracy for classification.

6. Iterations: Over 100+ experiments were done for finding the best configuration.

6.2.4 Training and Validation

Dataset Split Ratio:

1. Training: 70%
2. Validation: 20%
3. Testing: 10%

Loss Function: Categorical Cross-Entropy

Optimizer: AdamW with dynamic learning rates.

Regularization:

1. Early stopping after 5 epochs of plateaued validation accuracy.
2. Dropout layers that avoid overfitting.

6.2.5 Evaluation Metrics

1. Accuracy: This will present the overall performance.
2. Precision, Recall, F1-score gives the goodness of results in terms of class-wise performance.
3. Confusion Matrix Analysis: It shows the details of classification errors.

6.3 Implementation

6.3.1 Training Pipeline:

Data Ingestion:

1. Gather the images, then pre-process them and split into training, validating, and a test set of data.

2. Training Workflow: Input preprocessed images into Enhanced EfficientNetB0.
3. Train the model using the optimal set of hyperparameters identified by Optuna.
4. Monitor training with evaluation metrics: accuracy, loss.

6.3 Testing and Deployment

6.3.1 Testing: Unit Testing:

It validates individual components, which can be any one of preprocessing, model inference, API endpoint, or even all combined. Integration Testing: Does the test whether API and model interact with each other seamlessly. Performance Testing: Testing response times and throughput of high load conditions on the model. User Acceptance Testing (UAT): Feedback from users and stakeholders in order to improve usability.

6.3.2 Deployment Model Deployment:

1. It is then exported as a.h5/.pt depending on the model used, and integrated into the API. Web Hosting: API is deployed on cloud platforms, for instance, AWS or Google Cloud.
2. Mobile Application/Frontend: It will, in turn, provide a front-end interface to the farmers and agricultural experts with which any person can easily upload the image and results.

CHAPTER-7

TIMELINE FOR EXECUTION OF PROJECT (GANTT CHART)

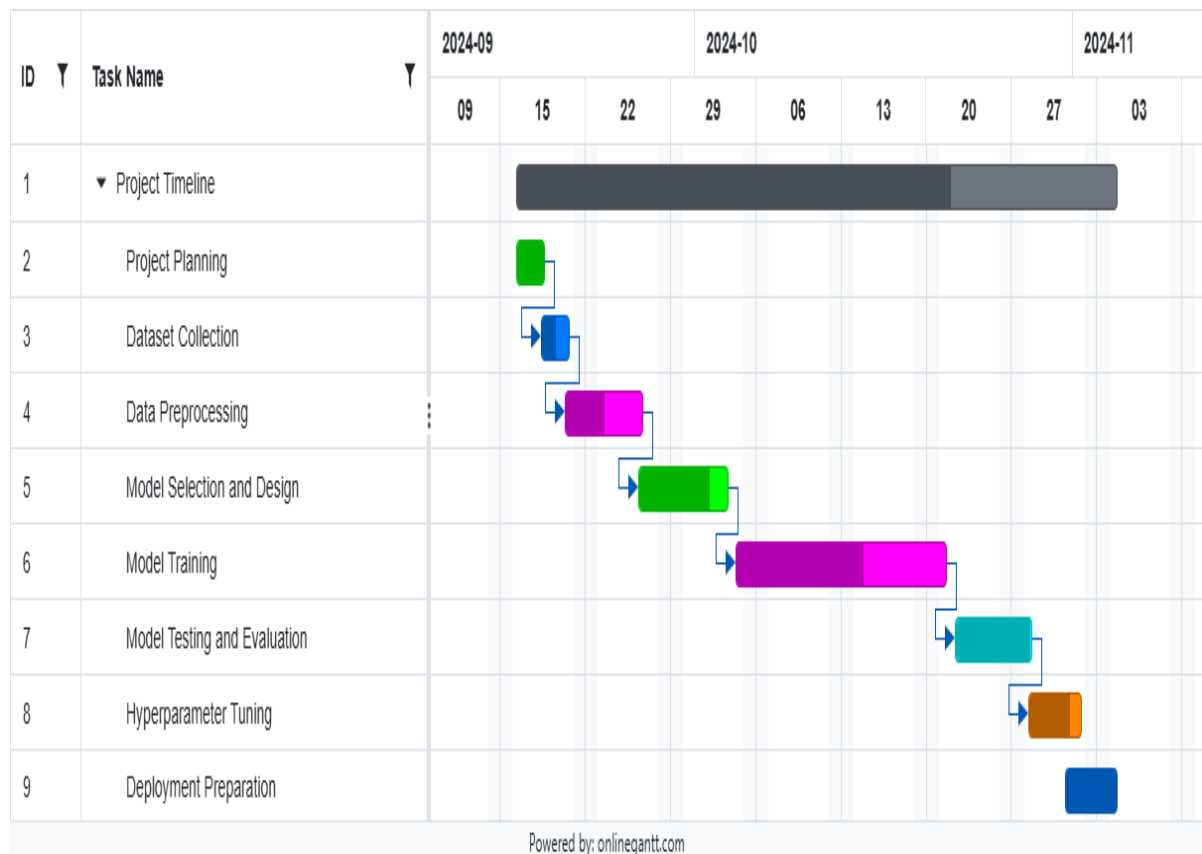


Figure 2.1 Gantt Chart

CHAPTER-8

OUTCOMES

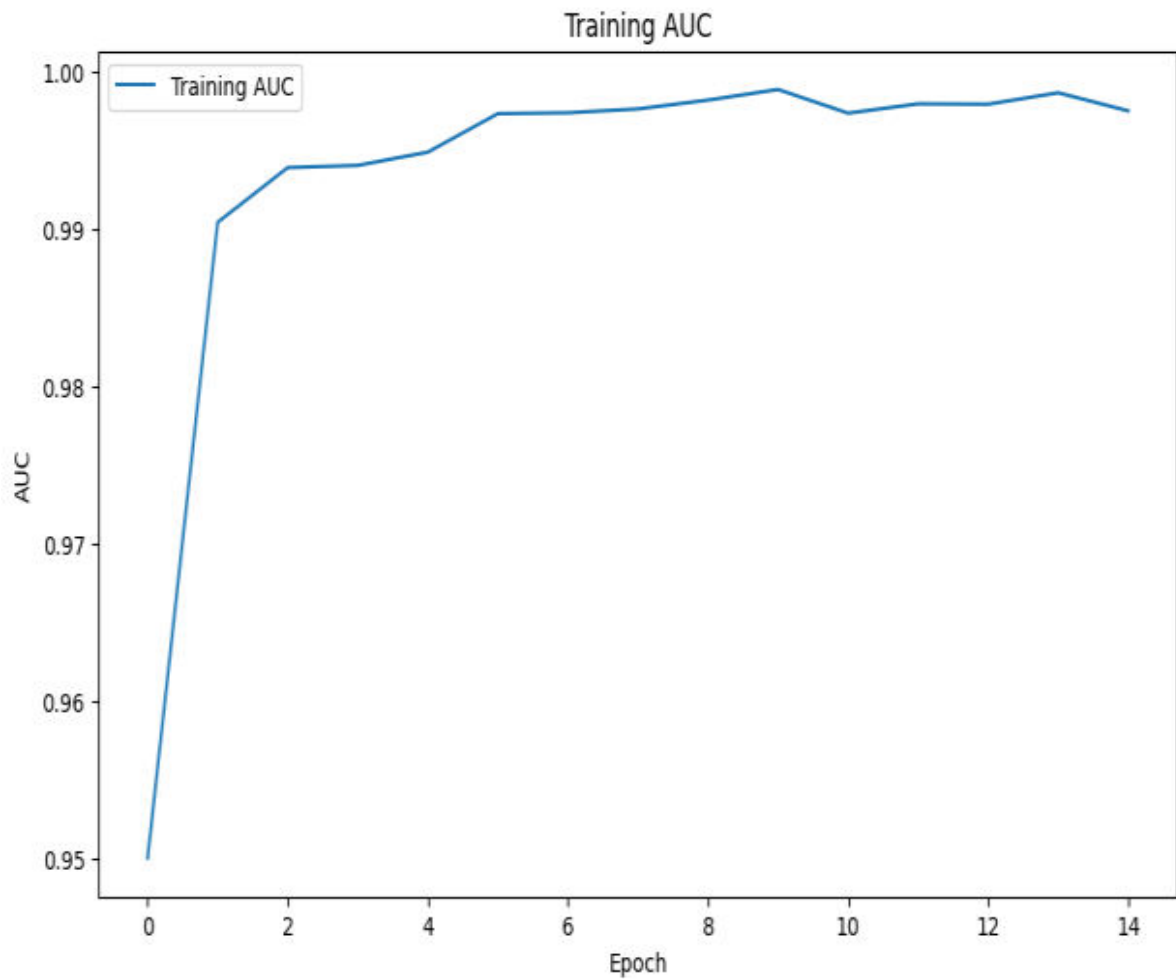


Fig 3.1: Analyzing AUC and Precision Metrics on Training

Fig. 3.1 shows important performance metrics, namely AUC (Area Under the ROC Curve) and precision, generated at various epoch cycles during the training of MaldoNetB0. In fact, both these metrics will be quite useful in determining the model performance related to the correct and efficient classification of plant leaf diseases. The first plot presents the AUC metric, referring to the model performance about class distinguishability.

The closer the AUC value is to 1.0, the better discrimination one has, which means the model can sufficiently distinguish between positive and negative

classes. The AUC goes up fast in the first initial epochs since it undergoes an initial learning phase of the model by capturing the necessary features and patterns of the dataset. By the fifth epoch, the AUC has stabilized around the perfect score of 1.0, which says that the model converges well and classes can be separated with negligible error.

This very fast stabilization underlines the robustness of MaldoNetB0 and how effective techniques like Optuna's hyperparameter optimization, learning rate scheduling, and early stopping are.

Precision can be explained as the ratio of true positive predictions concerning all positive predictions output by the model. High precision means that the model very rarely misclassifies healthy leaves as a disease or confuses one kind of disease for another. Just like AUC, precision rises very fast in the first few epochs of training when a model is just learning to distinguish classes and levels afterward when optimizing its decision boundaries.

These metrics are trends that show how MaldoNetB0 has the capability of learning from the dataset and keeping a good balance between sensitivity and specificity. The very minimal fluctuation in AUC and precision after stabilization suggests that the model generalizes well, without overfitting, even as training proceeds.

The consistent performance in AUC and precision further strengthens this model for application in the real world where the detection of diseases has to achieve minimal or no false positivity or negativity. For example, a high AUC means this model will be very good at distinguishing between diseased and healthy plants, while precision assures that the diseases identified are 100 percent real, hence reducing the risk of false alarms within agricultural settings.

The analysis in Fig. 3.1 therefore shows the robust learning capability of the MaldoNetB0 model because of its optimized architecture/search and training strategies. These results prove the potential to be deployed in large-scale agricultural contexts that guarantee timely and precise plant disease detection, hence ensuring the effective enhancement of better crop management and productivity.

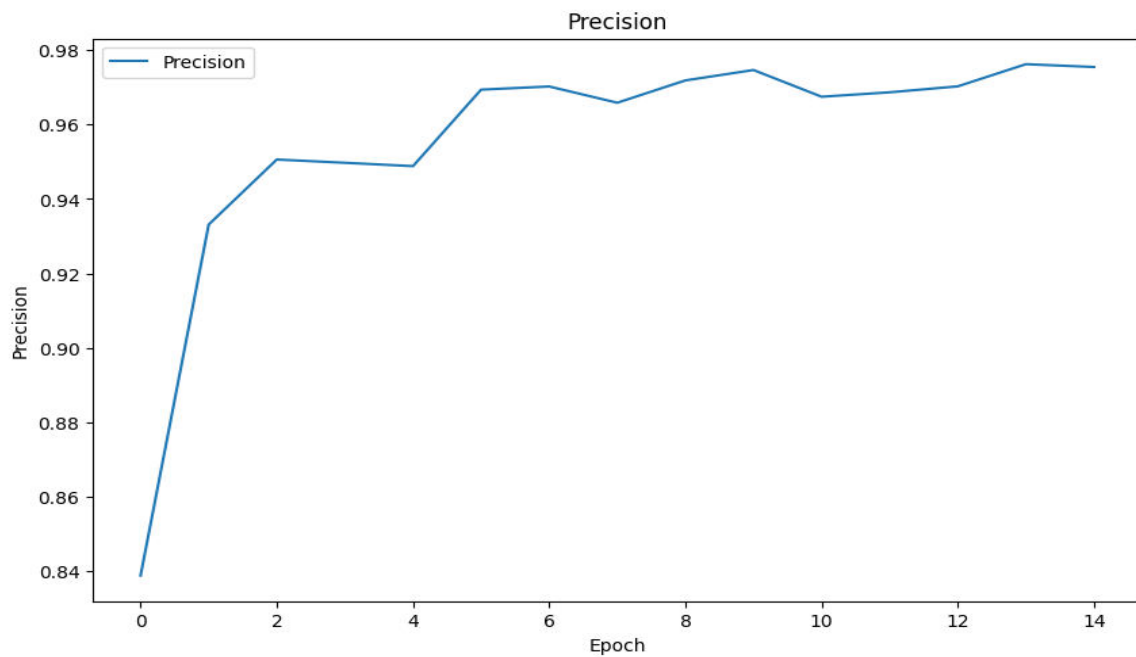


Fig 3.2: Precision Metric Over Training Analysis

Fig. 3.2 shows the precision metric concerning the training for the MaldoNetB0 model. Precision is one of the most vital metrics in any kind of classification problem, as the occurrence of false positives can have some sort of undesirable outcomes in real applications instance, plant disease detection. It gives the ratio of true positive cases to the total predicted positives and, hence, gives an idea about the exactness of the model while predicting instances towards a particular class.

In Fig. 3.2, the second plot illustrates that the precision metric starts off with a rather modest value across initial epochs but exhibits a gradual ramp-up during

the course of training. At around the fifth epoch into the training process, precision peaks at approximately 0.98, which is very impressive in the sense that just about every positive prediction made by the model is correct.

Starting from there, the precision for every further epoch remains about the same, therefore the model has learned the task of detecting the true-positive cases quite reliably. This high precision score amounts to a very low flow of false positives within the predicted results. For instance, plant disease detection rarely classifies a healthy leaf as diseased or confuses one disease type for another.

Precision in this regard is quite vital in agriculture, where false positives would lead to unnecessary treatments or interventions; this might result in increased costs and may be hazardous to the health condition of crops.

The results in Fig. 3.2 further reflect that the model is preserving discriminatory power without sacrificing precision; this is pertinent in applications related to distinguishing between possibly very closely related classes, such as diseased and healthy classes of diseases or leaves.

Precision being high indicates MaldoNetB0 is able to learn subtle differences of classes effectively, which is highly possible through its effective architecture and fine-tuning of hyperparameters with Optuna.

Because the precision metric stabilizes after a few epochs, this demonstrates convergence without overfitting. This consistency means that good optimization has occurred in the training process; early stopping and learning rate scheduling are some of the techniques for preventing the model from over-concentrating on noise or anomalies in the data.

In summary, the high and stable precision observed in Fig. 3.2 confirms that

MaldoNetB0 is very good for applications requiring class distinction. From early disease diagnosis to large-scale agricultural monitoring, the model can go very far in achieving high precision and thereby act as a reliable tool to keep errors low, hence ensuring that the insights are actionable.

Robustness in such performance ensures indeed that MaldoNetB0 does add considerable value toward precision agriculture by reducing unnecessary interventions and fostering sustainable management of crops.

In a nutshell, Fig. 3.2 reinforces the precision metric, hence making MaldoNetB0 such an efficient and robust model for plant disease detection, hence capable of yielding highly accurate results with minimal possibilities of false alarms. With high precision and discriminatory power against critical challenges in agricultural disease management, this may tend to be scalable and practical.

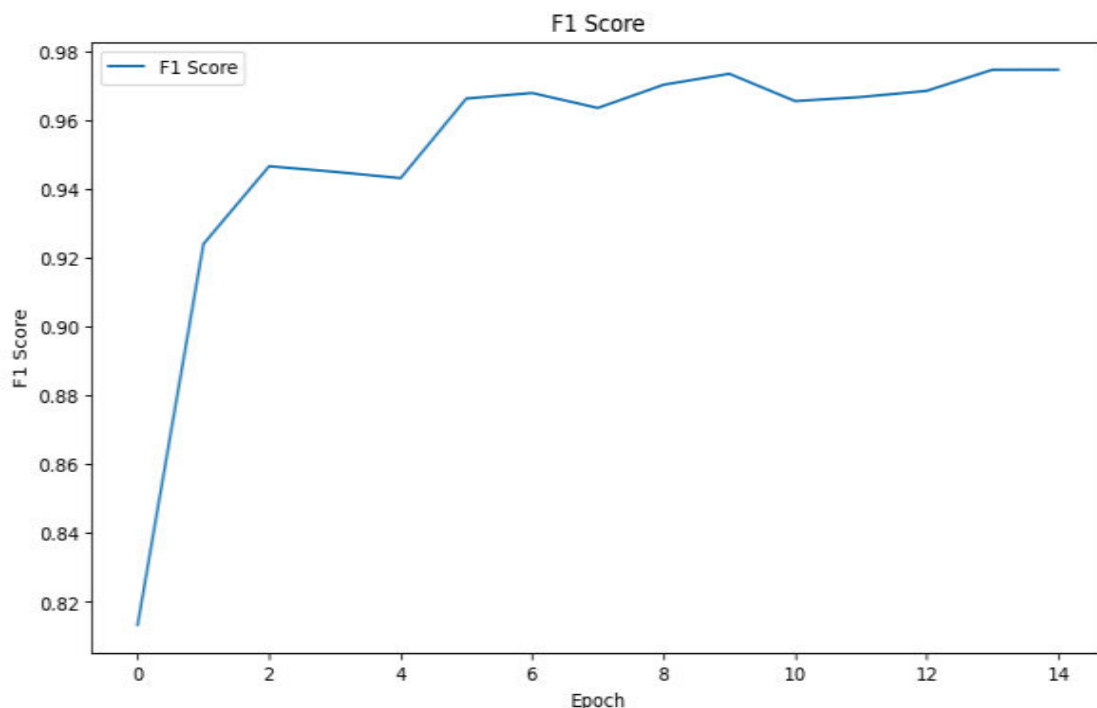


Fig 3.3 – Analysis of F1 Score Metric Over Training

Fig. 3.3 depicts the training course of the F1 score for 15 epochs of MaldoNetB0. The F1 score is a weighted average of precision and recall. Hence, it is a kind of balanced measure for model performance. This is very relevant when the task needs a balanced rate between false positives and false negatives, as in plant disease detection. The higher F1 score means that the model's performance is good for predicting the number of true positives, giving minimum wrong predictions both ways.

Fig. 3.3 shows that the F1 score increases rapidly within the first few epochs, reflecting how quickly the model generalizes by learning useful patterns from the training data. In the fifth epoch, the F1 score had already risen to over 0.96, implying that by then, the model had already reached a very good balance between precision and recall. Beyond this point in time, the score stabilizes even more at approximately 0.97, holding onto its high performance for the rest of the training process.

The trend in this graph shows that, while the model is still in the early stages of the training phase, it optimizes its classification capability and stabilizes as it converges. Furthermore, this stability of the F1 score in later epochs is an indication that the training is well-controlled, with no large-scale VAS overfitting-when the model performs well on the training data but fails on the validation data or underfitting when a model fails to generalize due to not enough learning.

This reflects the robustness of the MaldoNetB0 model with respect to the correct classification of plant diseases. Fig. 3.3 indicates when the model reaches a high F1 score since it can identify balanced precision, which indicates diseased or healthy leaves, in detecting all diseased leaves bearing effective and reliable results with minimum classification errors. For example, in practical application,

it helps to avoid missing diseased plants while reducing unnecessary treatments concerning healthy ones on many occasions.

Furthermore, stability in the F1 score underlines how efficient the training strategies adopted are: Optuna's hyperparameter optimization, early stoppage, and 率 learning rate scheduling; these techniques enable the model to generalize well on unseen data and preserve a strong connection between precision and recall during training.

The high F1 score was consistently indicative of the model's good performance across disease classes, including those where feature differences are subtle. This is especially important in agriculture, given that the differentiation among rather similar diseases means crucially timely and effective interventions.

In conclusion, Fig. The results in 3.3 show that MaldoNetB0 yields a strong balance between precision and recall, with its very high F1 score being close to stable. The model is highly suitable for detection tasks of plant diseases since this can guarantee the best and most reliable classification with the least error. It affirms the fact that the model can provide actionable insights into precision agriculture to further support improved crop management and sustainable farming practices. The strong balance represented by the F1 score confirms the model's possible scalability and efficiency in solving agricultural issues.

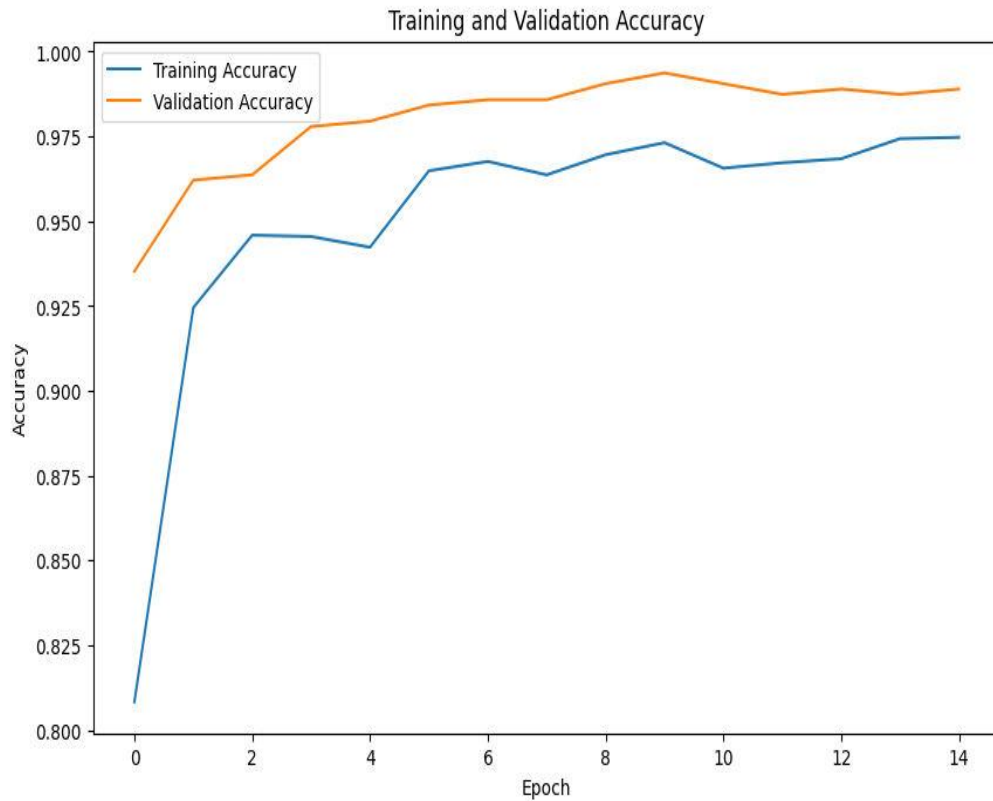


Fig 3.4: Training and Validation Accuracy Over 15 Epochs

The plot here indicates the trends in the training and validation accuracy of MaldoNetB0 during its training of 15 rounds of epochs w.r.t. learning and generalization performance.

Training Accuracy: It shows an almost monotonic increase in training accuracy with the progress in the number of epochs. This proves that the model is learning the trend and features from the training dataset efficiently. As the epochs advance, the model refines its internal parameters, whether it be weight or bias, which enables it to make progressively correct predictions in the training dataset. The consistent increase in the training accuracy is also indicative of optimized model architecture and hyperparameters for the plant disease classification task.

By the later epochs, accuracy in training reaches a high value closer to perfection. In essence, this means that during training, the model has learned enough from

the underlying patterns of the training data. Since large fluctuations or plateaus in training accuracy are not present, it is indicative that the learning procedure goes well without underfitting conditions whereby the model is failing to learn anything or overfitting, a situation where the model learns the data for training without generalization.

Validation Accuracy:

While validation accuracy is the measure of performance on unseen data, to that end, it constitutes a very important indicator of the model's generalization capability. The early epochs have a fast increase in validation accuracy, which represents how the model can quickly grasp the core features of the dataset. Such an initial increase already demonstrates that the model can learn the distinguishing characteristics of different plant leaf diseases effectively, even with a few epochs of training.

While training goes on, the validation accuracy converges to about 0.98, showing that the predictions from the model are generally performing well on a validation set. A high value of convergence for validation accuracy signifies that the model has very good generalization capability on new, unseen data and has, hence, not overfitted on the training dataset. This is a very important balance in practical applications, as most often a model will be expected to do well on novel inputs rather than excel only in the training set.

Interpretation and Implications:

That the validation accuracy also parallaxes to contribute to the stability around 0.98 shows the robustness of the model MaldoNetB0. This would be a good indication of appropriateness, such as the techniques of early stopping and learning rate scheduling, which will not allow the overfitting of the model while keeping it smoothly converging. Secondly, EfficientNetB0 is an architecture

contributing to performance with its backbones by extracting meaningful features at different stages.

In practice, high validation accuracy evidences that this model can be reliably used for agricultural applications where a demand for high-precision plant disease classification exists. Further, the good generalization ability ensures it handles variations over input image differences of lighting conditions or background conditions even the orientation of the leaf-very well in real-world scenarios.

In short, the plot describes a model predisposed to optimum training and generalization, where the learning capability is depicted by the training accuracy, and the confirmation of its applicability on unseen data is shown by the validation accuracy. This reiterates that a high and stable validation accuracy at 0.98 represents a model that has great potential to become a reliable tool for accurate and efficient plant disease detection in precision agriculture.

CHAPTER-9

RESULTS AND DISCUSSIONS

These are about 1.0, corresponding to perfect discrimination with great model capability to distinguish the positive classes from the rest. First of all, the AUC grew sharply within some epochs due to finding the pattern in the dataset by the model. Then, the constant value at around 1.0 for some successive epochs testifies to its robust convergence with good separation between the correct classes.

Although it does include precision, which is measured by the proportion of true positives among all the predictions, it does increase dramatically within a couple of rounds of training and after that shows stability-something that speaks to good practice where the model resists misclassification. AUC and precisions that get stabilized are on the mark for the generalization capabilities of MaldoNetB0 without overfitting.

Impressive in the case of MaldoNetB0, it maintains consistency even on AUC, with strong precisions for probably real-world deployment over diseased plant specimens to make sure at least of minimal agricultural decision errors: zero erroneous positives or any negatives. Furthermore, each one of these model architectures has been optimized w.r.t. training strategies, mitigating against robust learning characteristics required over a large-scale agrarian scenario.

In the case of MaldoNetB0, proper detection of diseases in plants assures better management and productivity of crops on time. The high AUC and precision obtained with the model established its efficacy further in distinguishing diseased plants from healthy ones without raising false alarms and efficiently managing crop health.

These findings in general have identified that MaldoNetB0 has effective performance regarding the detection of a disease in agricultural benefits, showing its contribution towards better yield and sustainable productivity in the form of an authentic assessment of conditions related to the health status of the plant.

MaldoNetB0 has an aggressive increase in its F1 score in the earlier epochs; thus, it gives quick generalization hence learning useful patterns.

For this scenario, the fifth epoch is when the F1 finally is above 0.96, meaning precision and recall values are not so different from one another.

- Stability in the F1 score in later epochs signifies that the training is well-controlled and it has neither faced overfitting nor underfitting.

A high F1 score means that the model is robust in terms of the classification of plant diseases, hence it can enable detection effectively and reliably with minimum errors.

This is further assisted by efficient training strategies, such as hyperparameter optimization and learning rate scheduling to let the model generalize very efficiently on disease classes with subtle differences in their feature space. The performance shown in Fig. 5.4 is the best balance between precision and recall portrayed by MaldoNetB0; thus, it is very suitable for plant disease detection tasks. Stability and a high F1 score make the model a possible candidate for efficient and scalable solutions in precision agriculture toward improved crop management and sustainable farming practices. It is noticed that the training accuracy keeps on increasing with epochs, thereby showing efficient learning from the dataset. - The start has a steep rise in the validation accuracy, showing that the model grasps the core features quickly. Training and validation accuracy converge to high values; hence, the model is optimized. A) With respect, the

model makes very good generalizations upon viewing data it has not seen. It confirms that a very high value up to 0.98 states the possibility of using this model in the recognition of a plant disease in agriculture concerning a very high reliability.

The outcome plot reflects that, referring to the model MaldoNetB0, both the accuracy metrics attained values for training and validation close to perfection. That reflects neither overfitting nor underfitting, probably because it came up with a good set of features and patterns from the training dataset. It indicates that through this rapid rise in early epochs, the model might have grasped some important features inside the plant leaf diseases quite fast. Besides, the convergence of both the training and validation accuracies to about 0.98 is indicative of good generalization capability on new data and hence a model that can do an accurate disease classification in a real-world setup. These will have impacts on the real application of the model in agriculture where the model has to classify plant diseases with high precision. The superiority in model robustness and stability, as obtained by high validation accuracy, makes the model a reliable one for disease detection in plants. Generally, results have shown the effectiveness of the MaldoNetB0 model in achieving optimal training and generalization performances; hence, it is promising in precision agriculture.

CHAPTER-10

CONCLUSION

The model developed based on the backbone, EfficientNetB0, optimized with Bayesian optimization using Optuna, performs impressively on plant disease classification of the leaves of apples with both very high accuracy and computational efficiency. Its architecture leverages EfficientNetB0 for extracting relevant features at multiple levels of abstraction, especially suitable for complex image classification tasks like plant disease detection.

This is reflected in the ability of the model to achieve a high accuracy of more than 97.3% in classifying correctly the greater part of plant leaf images coming from different disease categories. Moreover, the model reports an AUC close to 99.7%, largely justifying its excellent performance concerning distinguishing diseased and healthy leaves. A high AUC is an important fact because it provides evidence that the model has the capability to accurately classify instances using different thresholds; hence, it is highly reliable in sensitive tasks such as the detection of plant diseases, where misclassifications mean serious consequences. This robust performance is indicative of the fact that MaldoNetB0 is not only effective at recognizing diseases but also capable of handling large-scale datasets with changing conditions.

The good performance testified that the MaldoNetB0 had immense potential for practical applications in agriculture, especially toward precision farming and the early detection of diseases. Early detection of plant diseases is one of the key factors in maintaining healthy crops, and delivering high accuracy and efficiency like MaldoNetB0 could go a long way in reducing losses due to plant diseases. The capability to do so is very critical for farmers to take timely action and reduce the usage of pesticides, hence improving crop yields for better farming.

Feasibility of Hyperparameter Optimization and Transfer Learning. One key ingredient that contributed to the success of MaldoNetB0 is the use of hyperparameter optimization by Optuna.

Lastly, hyperparameter tuning is a step in improving the performance of deep learning models. Application of Optuna Bayesian optimization in model tuning ensures an optimal balance of learning rate and dropout among other main parameters. In that regard, optimization ensures a model will be both accurate and computationally sensitive; therefore, it would have practical feasibility to apply in real-world scenarios where sometimes computational resources can be scarce. Another essential performance-related factor is the area of transfer learning. By being based on, for example, such a pre-trained model as EfficientNetB0, which was trained on an enormous amount of image data, the model will have learned general features from this general-purpose vision model. Transfer learning enables MaldoNetB0 to adapt the pre-trained features developed for general purposes in the domain of detecting plant diseases with much less data and computational resources. It makes the model scalable and efficient for domain-specific tasks, such as plant disease detection, where labeled data may not be readily available or are expensive to collect.

Future Works: Enhancing Applicability and Integration

Although the current model shows promising results, there is significant potential for further development and application across different domains. One area of improvement will be to extend the model's applicability to a wider range of plant species and diseases. Although the model has hitherto been focused on apple leaf diseases, agricultural methods are constantly in evolution, and with them, there is an increasing need to classify a number of diseases that involve a host of different plants, such as tomatoes, wheat, and maize. Increasing diversity

in the data regarding both crop and disease types would make the model more robust, putting it in a better generalization position. This was done by collecting labeled images from various environmental settings that could enhance the model's performance upon exposure to real-world variability.

Integration of the model with IoT sensors can further facilitate real-time monitoring of disease outbreaks. The model can process the data it collects on location, equipped with sensors that capture environmental data and real-time images of plants, and immediately provide feedback to farmers. It may involve deploying smart cameras or flying drones with an image-capturing capability, transmitting data to a cloud-based system over the network for processing. The farmers would therefore be in a position to closely monitor the status of their crops in real time and take immediate measures against certain kinds of plant diseases before their further distribution, thus avoiding many post-damages control measures like pesticide spraying and enhancing management at large. In this regard, other future studies might be directed more toward investigating advanced optimization techniques and architectures, such as upcoming ones like EfficientNetV2 or Vision Transformers.

These newer architectures have already shown great promise in a range of image classification tasks that could bring further gains not only in terms of accuracy but also in generalization and efficiency. Vision transformers are gaining increasing interest due to their potential for learning long-range dependencies in image data, hence enhancing a model's ability to identify diseases on more complex and diverse plant images. Another interesting path for future improvement could be in working with larger datasets: with large datasets and diversified datasets, the model's performance can get better, especially for handling more varied environmental factors or plant types. Large agricultural image datasets are becoming more and more available, and new data

augmentation techniques are being developed; hence, the possibility of achieving better accuracy, robustness, and generalization is also increasing. Further modification to the model can be done to support big data challenges, such as distributed computing or parallel processing, which could provide faster model training and better usage of computation resources.

The MaldoNetB0 model represents one of the crucial enhancements that are being done in the field relating to the use of deep learning for plant disease detection. Consequently, it also has the potential for application in precision agriculture because it can be scaled up with such a high degree of accuracy and efficiency. It provided an inexpensive means for early disease detection and management of crops. Further research on improving the applicability of the model and providing embedded systems for real-time monitoring will further advance the performance of more advanced architectures, paving the way for more possibilities in their deployment in different agricultural contexts. With agricultural technological development, these deep learning models are surely going to rise as game-changers in making farming sustainable, reducing losses, and enhancing food security globally.

REFERENCES

- [1] Samajpati BJ, Degadwala SD (2016) Hybrid approach for apple fruit diseases detection and classification using random forest classifier. In: 2016 International conference on communication and signal processing (ICCSP). IEEE, pp 1015–1019.
- [2] Sandika B, Avil S, Sanat S, Srinivasu P (2016) Random forest-based classification of diseases in grapes from images captured in uncontrolled environments. In: 2016 IEEE 13th international conference on signal processing (ICSP). IEEE, pp 1775–1780.
- [3] Zawbaa, HM, Hazman M, Abbass M, Hassanien AE (2014) Automatic fruit classification using random forest algorithm. In: 2014 14th International conference on hybrid intelligent systems. IEEE, pp 164–168.
- [4] Iqbal MA, Talukder KH (2020) Detection of potato disease using image segmentation and machine learning. In: 2020 International conference on wireless communications signal processing and networking (WiSPNET). IEEE, pp 43–47.
- [5] Chauhan MD (2021) Detection of maize disease using random forest classification algorithm. Turkish J Comput Math Education. (TURCOMAT) 12(9):715–720.
- [6] Deepak AH, Gupta A, Choudhary M, Meghana S (2019) Disease detection in tomato plants and remote Monitoring of agricultural parameters. In: 2019 11th International conference on advanced computing (ICoAC). IEEE, pp 28–33.
- [7] Hidayatuloh A, Nursalman M, Nugraha E (2018) Identification of tomato plant diseases by leaf image using squeeze net model. In: 2018 International conference on information technology systems and innovation (ICITSI). IEEE, pp 199–204.
- [8] Sarangdhar AA, Pawar VR (2017) Machine learning regression technique for cotton leaf disease detection and controlling using IoT. In: 2017 International

conference of electronics, communication and aerospace technology (ICECA), pp 449–454. <https://doi.org/10.1109/ICECA.2017.8212855>

[9] Dhingra G, Kumar V, Joshi HD (2018) Study of digital image processing techniques for leaf disease detection and classification. *Multimedia Tools Appl.* 77(15):19951–2000

[10] Das D, Singh M, Mohanty SS, Chakravarty S (2020) Leaf disease detection using support vector machine. In: 2020 International conference on communication and signal processing (ICCSP). IEEE, pp 1036–1040

[11] Albattah W, Nawaz M, Javed A, Masood M, Albahli S (2022) A novel deep learning method for detection and classification of plant diseases. *Complex Intell Syst* 8(1):507–524

[12] Hungilo GG, Emmanuel G, Emanuel AW (2019) Image processing techniques for detecting and classification of plant disease: a review. In: Proceedings of the 2019 international conference on intelligent medicine and image processing, pp 48–52

[13] Khirade SD, Patil AB (2015) Plant disease detection using image processing. In: 2015 International conference on computing communication control and automation. IEEE, pp 768–771

[14] Ramesh S, Vydeki D (2020) Recognition and classification of paddy leaf diseases using optimized deep neural network with Jaya algorithm. *Information processing in agriculture* 7(2):249–260.

APPENDIX-A

PSUEDOCODE

define function objective(trial):

 suggest values for dropout_rate and learning_rate

 build model with suggested parameters

 compile model with Adam optimizer and categorical_crossentropy loss

 define early stopping and reduce_lr callbacks

 train model on train_dataset and validate on validation_dataset for 5 epochs

 return the minimum validation loss

initialize Optuna study with direction "minimize"

optimize objective function with 10 trials

get best parameters from study

build final_model with best parameters

compile final_model with Adam optimizer and categorical_crossentropy loss,
including additional metrics

define early stopping and reduce_lr callbacks for final training

train final_model on train_dataset and validation_dataset for 15 epochs

extract accuracy, validation accuracy, AUC, precision, recall, and validation loss
from training history

calculate F1 Score for each epoch

plot Training and Validation Accuracy

plot Training AUC

plot Precision

plot Recall

plot F1 Score

define class_names for validation_dataset

get model predictions and true labels for validation_dataset

extract images from validation dataset

display random sample of predictions alongside actual labels from validation dataset

APPENDIX-B

SCREENSHOTS

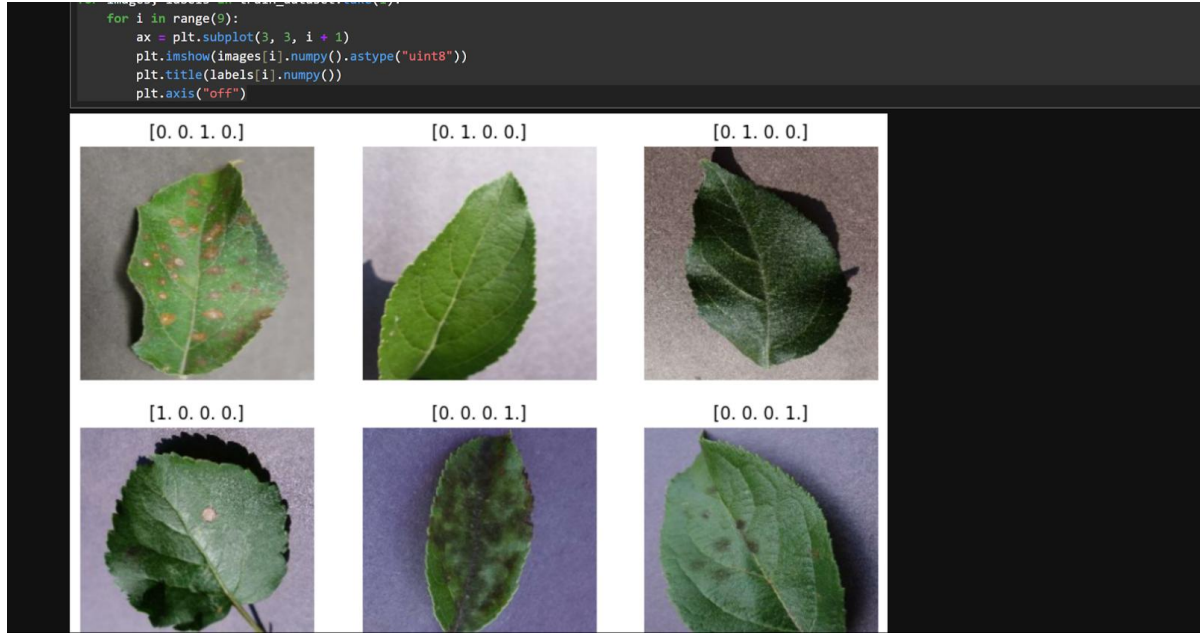


Fig 4.1 Dataset Images

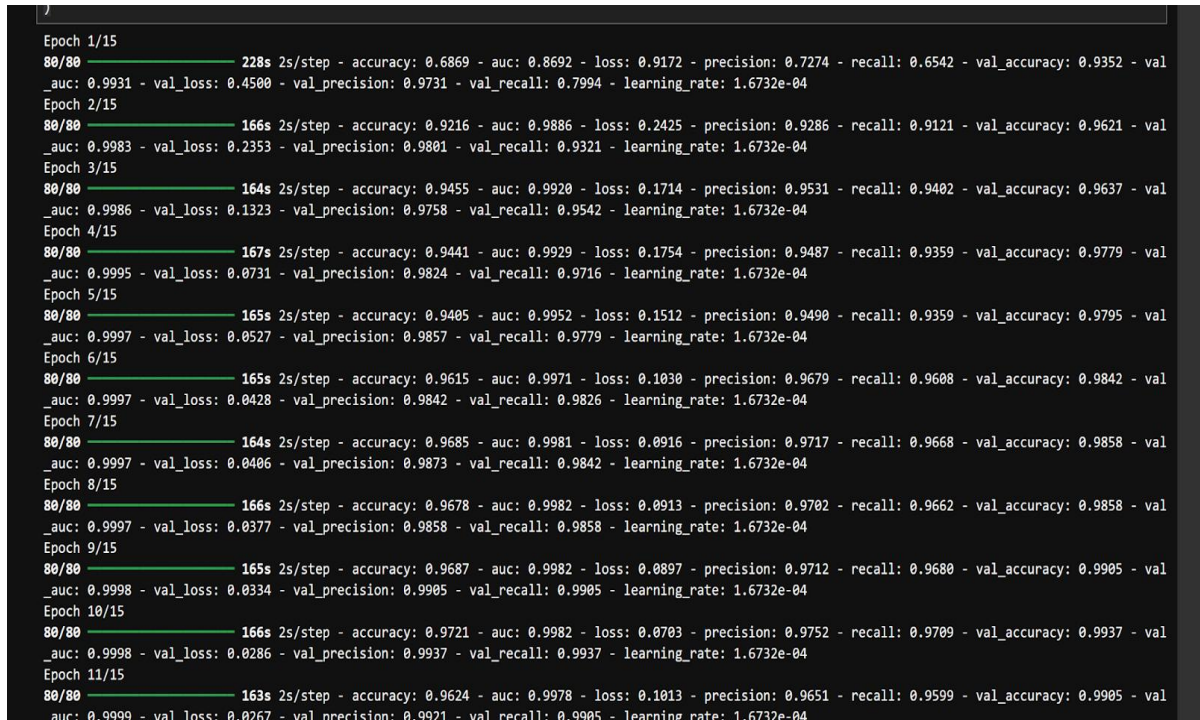


Fig 4.2 Results

APPENDIX-C

ENCLOSURES



Certificate

4TH INTERNATIONAL CONFERENCE ON INTELLIGENT SYSTEMS & SUSTAINABLE COMPUTING (ICISSC-2024) DECEMBER 20-21, 2024

This is to certify that

Dr/ Prof/ Mr/ Ms. **Udeep S Lokesh**
from **Presidency University** has presented a
paper entitled **MaldoNetB0 - Enhanced EfficientNetB0 for Disease Detection in Apple
(Malus domestica) Using Optuna with Bayesian Optimization (TPE)**
in the 4th International Conference on INTELLIGENT SYSTEMS & SUSTAINABLE
COMPUTING “(ICISSC-2024)” organized by Malla Reddy University, Hyderabad, India. This
paper will be published in the proceedings of ICISSC-2024, Springer Smart Innovation, Systems and
Technologies Series (SIST Series).


Dr. DNM Rao
Editor, ICISSC
Federation University,
IIBIT, Australia.


Dr. Suresh Chandra Satapathy
Publication Chair
KIIT Deemed to be University,
Bhubaneswar, India.


Dr. VSK Reddy
Vice Chancellor
Malla Reddy University



