**Disease Prediction Using Multimodal Deep Learning and Machine Learning Approaches**

The project submitted to the

SRM University – AP, Andhra Pradesh

for the partial fulfillment of the requirements to award the degree of

**Bachelor of Technology**

In

**Computer Science and Engineering**

**School of Engineering and Sciences**

Submitted by

**H. Rohitha Aiswarya Dora (AP21110011372),**

**K. Pranay Deep (AP21110011383),**

**P. Sai Durga Vara Prasad (AP21110011632),**

**K. Pavan Venkata Sai Teja (AP21110011590),**

**D. Heswitha Reddy (AP21110011545).**

**A picture containing text

Description automatically generated**

Under the Guidance of

**Dr. Neeraj Kumar Sharma**

**SRM University–AP**

**Neerukonda, Mangalagiri, Guntur**

**Andhra Pradesh – 522 240**

**Nov, 2023**

# Certificate

Date: 13-Dec-23

This is to certify that the work present in this Project entitled “**DISEASE PREDICTION USING MULTIMODAL DEEP LEARNING AND MACHINE LEARNING APPROACHES**” has been carried out by **H. Rohitha Aiswarya (AP21110011372), K. Pranay Deep (AP21110011383), P. Sai Durga Vara Prasad (AP21110011632), K. Pavan Venkata Sai Teja (AP21110011590), D. Heswitha Reddy (AP21110011545)** under my supervision. The work is genuine, original, and suitable for submission to the SRM University – AP for the award of Bachelor of Technology in the School of Engineering and Sciences.

**Supervisor**

(Signature)

Dr. Neeraj Kumar Sharma,

Assistant Professor,

Department of CSE,

SRM AP.

# Acknowledgments

The successful completion of this project would not have been possible without the invaluable contributions of the following individuals and resources:

* Our instructor, Dr. Neeraj Kumar Sharma, for his expert guidance, unwavering support, and encouragement throughout the project. His insightful advice and dedication to our learning were instrumental in our progress and success.
* The creators of the PlantVillage dataset, for providing a rich and comprehensive collection of images for plant disease classification. This dataset served as the foundation for our research and enabled us to train and evaluate our deep-learning models.
* The TensorFlow and Keras libraries, for providing the essential tools and frameworks for building and training deep learning models. These libraries allowed us to implement our models efficiently and effectively.
* The scikit-learn library, for offering a comprehensive suite of metrics and tools for evaluating model performance. These tools enabled us to assess the accuracy, precision, and generalizability of our models.
* The matplotlib library, for providing powerful data visualization tools that helped us communicate our findings clearly and effectively.
* The authors of the research papers that informed the design and implementation of our deep learning models. Their contributions provided a solid foundation for our work and guided our approach to plant disease classification.

We extend our heartfelt gratitude to all of these individuals and resources for their contributions to our project. Their support, guidance, and expertise were invaluable to our success.

# Table of Contents

[Certificate i](#_Toc152863501)

[Acknowledgments iii](#_Toc152863502)

[Table of Contents v](#_Toc152863503)

[Abstract viii](#_Toc152863504)

[Abbreviations x](#_Toc152863505)

[List of Tables xii](#_Toc152863506)

[List of Figures xiv](#_Toc152863507)

[List of Equations xvii](#_Toc152863508)

[1. Introduction 1](#_Toc152863509)

[1.1 Problem Statement 1](#_Toc152863510)

[1.2 Objective 1](#_Toc152863511)

[1.3 Significance of the Study 1](#_Toc152863512)

[2. Background work 3](#_Toc152863513)

[3. Proposed System 6](#_Toc152863514)

[2.1 Data Collection and Preprocessing 6](#_Toc152863516)

[2.1.1 Plant Village Dataset 6](#_Toc152863517)

[2.1.2 Data Preprocessing 6](#_Toc152863518)

[2.2 Model Selection 7](#_Toc152863519)

[2.3 Model Architecture 7](#_Toc152863520)

[2.3.1 Base Layer 7](#_Toc152863521)

[2.3.2 Additional Layers 8](#_Toc152863522)

[2.3.3 Output Layer 8](#_Toc152863523)

[2.4 Model Configuration 8](#_Toc152863524)

[2.5 Model Training and Optimization 8](#_Toc152863525)

[2.5.1 Optimization 8](#_Toc152863526)

[2.5.2 Loss Function 9](#_Toc152863527)

[2.5.3 Training and Validation 9](#_Toc152863528)

[2.5.4 Early Stopping 9](#_Toc152863529)

[2.6 Evaluation 9](#_Toc152863530)

[2.6.1 Early Stopping 9](#_Toc152863531)

[2.6.2 Result Analysis 9](#_Toc152863532)

[4. Methodology 11](#_Toc152863533)

[5. Discussion 13](#_Toc152863534)

[5.1 Table 1. Model Configuration Table 13](#_Toc152863538)

[5.2 Table 2. Performance Metrics Table 13](#_Toc152863539)

[5.3 Figure 7. Training vs Validation Accuracy and Loss 13](#_Toc152863540)

[6. Experimental Results 17](#_Toc152863548)

[6.1 Figure 8. Confusion Matrices 17](#_Toc152863550)

[7. Concluding Remarks 20](#_Toc152863552)

[8. Future Work 22](#_Toc152863553)

[References 25](#_Toc152863554)

# Abstract

Disease prediction is a critical area in healthcare, and the integration of advanced technologies like deep learning and traditional machine learning provides a robust framework for accurate predictions. This project focuses on employing a multi-model approach, incorporating popular deep learning architectures—ResNet50, MobileNet, and InceptionV3—while experimenting with various hyperparameters and optimizers. The goal is to develop a comprehensive disease prediction system that not only achieves high accuracy but also demonstrates adaptability to different configurations.

The research methodology involves training and evaluating each model on a dataset obtained from the PlantVillage dataset. Utilizing TensorFlow and scikit-learn libraries, the models are configured with different learning rates, batch sizes, and optimizers. The evaluation metrics include accuracy, ROC AUC score, and classification reports.

The results showcase the diverse performance of each model under varying conditions. The findings provide insights into the impact of hyperparameters and optimizers on disease prediction accuracy. Additionally, visualizations such as confusion matrices offer a detailed understanding of model behavior.

The significance of this research lies in the exploration of multiple models, allowing practitioners to choose the most suitable architecture based on specific requirements. The multi-model approach also fosters adaptability, considering factors like computational efficiency and training time.

This abstract encapsulates the essence of the project, emphasizing the importance of a versatile disease prediction system that amalgamates the strengths of deep learning and traditional machine learning.

# Abbreviations

CNN Convolutional Neural Networks

ResNet50 Residual Network with 50 layers

InceptionV3 Deep CNN by Google

MobileNet Lightweight neural network

ROC Receiver Operating Characteristic

AUC Area Under the Curve

# List of Tables

[Table 1. Model Configuration Table………………………………………………………..7](#_Table_1._Model)

This table presents a comprehensive overview of the configurations used for each machine-learning model. It includes details on the architecture, layer specifications, hyperparameter values (such as learning rate and batch size), and the chosen optimizer.

[Table 2. Performance Metrics Table………………………………………………………..7](#_Table_2._Performance)

In this table, various performance metrics are reported for each model. Metrics include accuracy, precision, recall, F1-score, and ROC AUC score. The table facilitates a holistic comparison of how well each model performs across multiple evaluation criteria.

# List of Figures

[Figure 1. Sample Images from the Plant Village Data.……….………………………....6](#_Plant_Village_Dataset)

This figure showcases a selection of sample images from the Plant Village dataset. These images represent various plants and different stages of health, including both healthy plants and those affected by diseases.

[Figure 2. Sample Images from the Plant Village Dataset: Apple\_\_Apple\_scab after preprocessing………………………………………………………..………………………6](#_Data_Preprocessing)

This figure displays sample images from the Plant Village dataset specifically focusing on the category "Apple\_\_Apple\_scab" after undergoing preprocessing. Preprocessing steps may include resizing, normalization, or other transformations to prepare the images for model training.

[Figure 3. Model Flow Diagram with Resent as Base Model……………………………7](#_ResNet50)

The figure illustrates the flow diagram of the model architecture, with ResNet serving as the base model. It provides a visual representation of how the layers and components are connected in the neural network.

[Figure 4. Model Flow Diagram with MobileNet as Base Model………………….……7](#_MobileNet)

Similar to Figure 3, this figure depicts the model flow diagram, but with MobileNet as the base model. It highlights the specific architecture and connections used when MobileNet serves as the foundation for the neural network.

[Figure 5. Model Flow Diagram with InceptionV3 as Base Model………….………….8](#_InceptionV3)

This figure presents the model flow diagram where InceptionV3 is employed as the base model. It visually conveys the architecture and connections specific to using InceptionV3 in the neural network.

[Figure 6. Model Flow Diagram…………………………………………………………...11](#_Methodology)

A general model flow diagram illustrating the overall architecture of the model, including common components shared among different base models or specific layers designed for the task at hand.

[Figure 7. Training Accuracy vs Validation Accuracy and Loss……………………….13](#_Figure_1._Training)

Figure 7.1 to 7.7

These figures display a line plot showing the trend of training accuracy, validation accuracy, and loss across epochs during the training process for each model. It provides insights into each model's learning progress and potential overfitting or underfitting.

[Figure 8. Confusion Matrices……………………………………………………………...1](#_Figure_3._Confusion)7

Figure 8.1 to 8.7

These figures include visual representations of confusion matrices, offering a detailed overview of model performance for each model, especially in terms of classification errors.

# List of Equations

**Activation Functions:**

(1)

Eq 1. The sigmoid activation function squashes the input values between 0 and 1, making it suitable for binary classification problems.

(2)

Eq 2. ReLU is a widely used activation function that returns the input for positive values and zero for negative values.

**Batch Normalization:**

(3)

Eq 3. Batch Normalization normalizes the input by subtracting the mean and dividing by the standard deviation, improving training stability.

**Sparse Categorical Cross entropy:**

(4)

Eq 4. Sparse Categorical Cross Entropy is a loss function commonly used in multi-class classification tasks.

**Global average pooling operation:**

(5)

Eq 5. Global Average Pooling calculates the average value for each channel across the spatial dimensions, reducing the spatial dimensions of the input.

**Performance Metrics**

(6)

Eq 6. Precision measures the accuracy of the positive predictions made by the model. It calculates the ratio of correctly predicted positive instances (True Positives) to the total instances predicted as positive (True Positives + False Positives).

(7)

Eq 7. Recall, also known as Sensitivity or True Positive Rate, measures the ability of the model to capture all the actual positive instances. It calculates the ratio of correctly predicted positive instances (True Positives) to the total actual positive instances (True Positives + False Negatives).

(8)

Eq 8. F1-Score is the harmonic mean of Precision and Recall. It provides a balance between Precision and Recall. A higher F1-Score indicates a better balance between precision and recall.

(9)

Eq 9. Accuracy measures the overall correctness of the model. It calculates the ratio of correctly predicted instances (True Positives + True Negatives) to the total instances in the dataset.

(10)

Eq 10. Support represents the number of actual occurrences of the class in the specified dataset. It is the sum of True Positives and False Negatives.

# Introduction

Disease prediction stands as a pivotal domain within the healthcare landscape, where advancements in artificial intelligence and machine learning have showcased transformative potential. This research delves into the development and assessment of a disease prediction system, employing a multi-model approach that amalgamates deep learning architectures with traditional machine learning techniques. The focal point is to not only achieve heightened prediction accuracy but also to understand the intricate interplay of hyperparameters and optimizers in influencing model performance.

## Problem Statement

The realm of disease prediction within the healthcare sector grapples with the challenges of leveraging sophisticated technologies to enhance accuracy and effectiveness. Traditional approaches often face limitations in handling the complexity of healthcare datasets, necessitating a nuanced exploration of advanced methodologies. This research addresses the pressing problem of optimizing disease prediction models by adopting a multi-model approach that integrates deep learning architectures with traditional machine learning techniques.

## Objective

This research aims to contribute to the field by providing insights into the nuanced dynamics of disease prediction models. The objectives include:

**Model Exploration**: Investigating the performance of ResNet50, MobileNet, and InceptionV3 in disease prediction scenarios.

**Hyperparameter Impact**: Evaluating the influence of hyperparameters, including learning rates and batch sizes, on model accuracy.

**Optimizer Comparison**: Assessing the effectiveness of different optimizers, such as Adam and SGD, in enhancing model performance.

**Versatility and Adaptability**: Demonstrating the adaptability of the multi-model approach to diverse configurations and datasets.

## Significance of the Study

The significance of our research extends beyond academic curiosity. By developing effective automated plant disease classification models, we aim to provide practical tools for farmers, agricultural experts, and policymakers. The implications of our findings can enhance disease management strategies, ultimately contributing to global food security.

# Background work

1. **Introduction to Plant Disease Detection:**

Plant diseases present significant challenges to agriculture, affecting crop yields and food security. Timely and accurate disease detection is vital for effective crop management and disease control.

2. **Traditional Methods and Limitations:**

Conventional disease detection methods, relying on manual observation, are labor-intensive and prone to errors. These limitations underscore the need for advanced technological solutions.

3. **Technology Integration in Agriculture:**

The integration of technology, particularly machine learning, and computer vision, has transformed agriculture. Automated systems offer the potential to revolutionize disease detection processes, making them more efficient and accurate.

4. **Introduction to Deep Learning**:

Deep learning, a subset of machine learning, has emerged as a powerful tool for image analysis tasks. Its ability to automatically learn hierarchical features from data makes it particularly suitable for complex visual recognition tasks.

5. **Evolution of Deep Learning Architectures:**

The evolution of deep learning architectures has played a crucial role in addressing challenges in various domains. Models such as ResNet, InceptionV3, and MobileNet represent advancements that cater to specific requirements in image classification tasks.

6. **Transfer Learning Concepts:**

Transfer learning is a pivotal concept in deep learning where a pre-trained model, developed for one task, is leveraged for a different but related task. This approach capitalizes on the knowledge gained from large and diverse datasets, improving performance on tasks with limited data.

7. **ResNet Architecture:**

ResNet, with its deep and residual structure, has been successfully employed as a backbone for transfer learning. The pre-trained ResNet models, trained on massive image datasets like ImageNet, serve as excellent feature extractors for various visual recognition tasks.

8. **InceptionV3 Architecture:**

InceptionV3, known for its factorized convolutions and global average pooling, provides an effective architecture for transfer learning. Pre-trained InceptionV3 models, with their ability to capture multi-scale features, offer valuable insights for diverse image-related applications.

9. **MobileNet Architecture:**

MobileNet's lightweight design makes it suitable for transfer learning in resource-constrained scenarios. The pre-trained MobileNet models can be adapted for tasks where computational efficiency is crucial, such as applications on mobile devices.

10. **Transfer Learning in Plant Disease Detection:**

Transfer learning has proven particularly advantageous in plant disease detection. Leveraging pre-trained models enables the extraction of generic features from diverse datasets, facilitating the development of accurate and robust models even with limited labeled plant disease images.

11. **Challenges and Future Directions:**

While transfer learning offers substantial benefits, challenges include domain adaptation and model interpretability. Future research should focus on addressing these challenges and exploring new transfer learning techniques tailored to the specific requirements of plant disease detection.

This section provides insights into the role of transfer learning in the context of plant disease detection, showcasing its significance in leveraging pre-existing knowledge for enhanced model performance.

# Proposed System

The proposed system section outlines the approach taken to achieve the objectives of this research, covering aspects such as data collection, model architecture, and training strategies.



## Data Collection and Preprocessing

### Plant Village Dataset

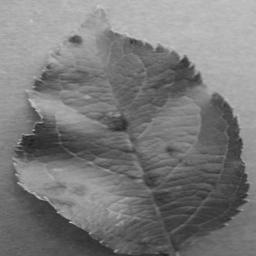
The dataset used in this study is the Plant Village dataset, a comprehensive collection of images capturing various plant diseases and healthy plant specimens. This dataset encompasses diverse classes, ensuring a broad representation of real-world scenarios. Each image is annotated with the corresponding disease label or marked as healthy.

1. Apple\_Black\_Rot (b) Apple\_Healthy (c) Grape\_Blact\_Rot

Figure 1. Sample Images from the Plant Village Dataset

### Data Preprocessing

1. Color (b) Grayscale (c) Segmented

Figure 2. Sample Images from the Plant Village Dataset: Apple\_\_Apple\_scab after preprocessing.

## Model Selection

Three prominent deep learning architectures—ResNet50, MobileNet, and InceptionV3—are chosen for their proven capabilities in image classification tasks. These architectures offer a spectrum of complexities, allowing us to assess their performance in disease prediction across different levels of model intricacy.

## Model Architecture

### Base Layer

#### **ResNet50**

ResNet50, a convolutional neural network (CNN), serves as one of the key architectures in this study. Renowned for its deep structure and residual connections, ResNet50 excels in capturing intricate features within images, making it a suitable choice for disease prediction tasks. The model's architecture includes multiple residual blocks, each contributing to the overall depth and expressive power of the network.

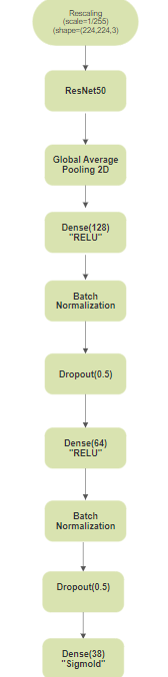


Figure 3. Model Flow Diagram with Resent as Base Model

#### **MobileNet**

MobileNet, chosen for its efficiency and lightweight structure, is another pivotal architecture in our multi-model approach. Particularly designed for mobile and edge devices, MobileNet utilizes depth wise separable convolutions to achieve a balance between accuracy and computational efficiency. This architecture proves beneficial in scenarios where resource constraints are a consideration.

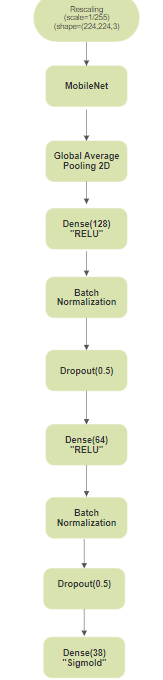


Figure 4. Model Flow Diagram with MobileNet as Base Model

#### **InceptionV3**

InceptionV3, characterized by its inception modules, provides an intricate yet efficient architecture. These modules enable the simultaneous extraction of features at various spatial scales, enhancing the model's ability to capture both local and global patterns. InceptionV3 is selected to explore the impact of a more complex architecture on disease prediction performance.

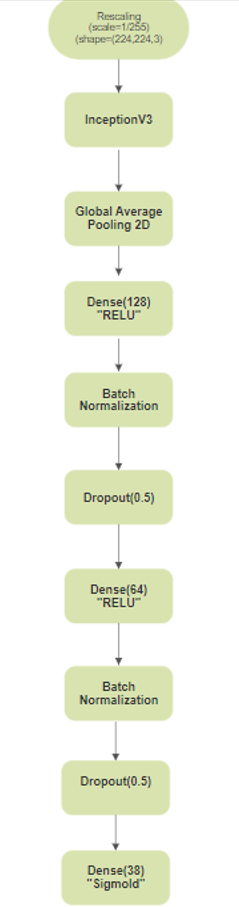
****

Figure 5. Model Flow Diagram with InceptionV3 as Base Model

### Additional Layers

To tailor the base to our specific task, additional layers are incorporated. These include a global average pooling layer to condense feature maps, dense layers with ReLU activation for classification, batch normalization to stabilize and accelerate training, and dropout layers to mitigate overfitting.

### Output Layer

The final layer consists of a dense layer with a sigmoid activation function, facilitating multi-label classification. The number of neurons in these layers corresponds to the number of disease classes in the dataset (38).

## Model Configuration

For each architecture, multiple configurations are explored, varying hyperparameters such as learning rates and batch sizes. The objective is to systematically assess the influence of these configurations on the model's ability to predict diseases accurately. This thorough exploration allows us to understand how different settings impact each model's overall performance and convergence.

## Model Training and Optimization

### Optimization

Two widely used optimizers, Adam and SGD, are employed to train the models. This choice enables a comparative analysis of their impact on model convergence and final prediction accuracy. The learning rate schedules are carefully chosen to facilitate effective optimization.

### Loss Function

Given the multi-label nature of the classification task, the loss function chosen is sparse categorical cross-entropy. This loss function is suitable for scenarios where each image can belong to multiple classes.

### Training and Validation

The dataset is split into training and validation sets, with a validation split of 20%. Models are trained using the selected configurations and optimizers, and training progress is monitored using performance metrics such as accuracy and loss.

### Early Stopping

To prevent overfitting, an early stopping mechanism is employed. Training is monitored based on validation accuracy, and if there is no improvement for a predefined number of epochs (patience set to 5), the training process is halted. The weights corresponding to the best validation accuracy are restored.

## Evaluation

### Early Stopping

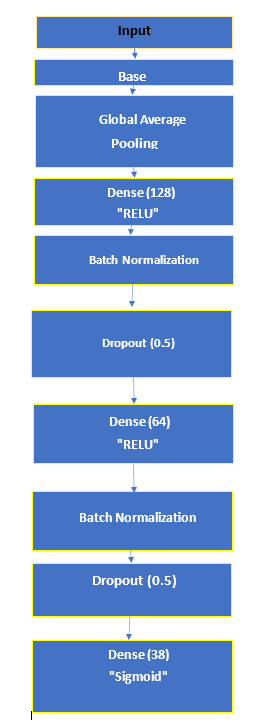
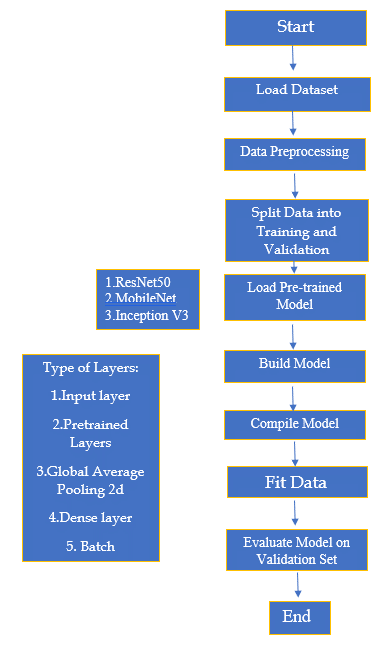
The performance of each model is assessed using standard evaluation metrics, including accuracy, ROC AUC score, and classification reports. These metrics provide a comprehensive understanding of the model's predictive capabilities.

### Result Analysis

The results obtained from each configuration are analyzed to identify trends, patterns, and the impact of different hyperparameters and optimizers on disease prediction accuracy. Visual aids such as confusion matrices further enhance the interpretation of model behavior.

# Methodology

**Figure 6. Model Flow Diagram**



# Discussion

**ResNet:**

The consistent increase in accuracy and minimal overfitting indicates ResNet's effectiveness. Its deeper architecture and skip connections contribute to superior feature extraction.

**MobileNet:**

Despite its efficiency, MobileNet struggles with overfitting, emphasizing the need for regularization techniques. Its lightweight design remains valuable for resource-constrained environments.

**InceptionV3:**

InceptionV3, with its nuanced feature extraction, offers a middle ground between ResNet and MobileNet. Its convergence pattern aligns with its reputation for handling complex datasets.



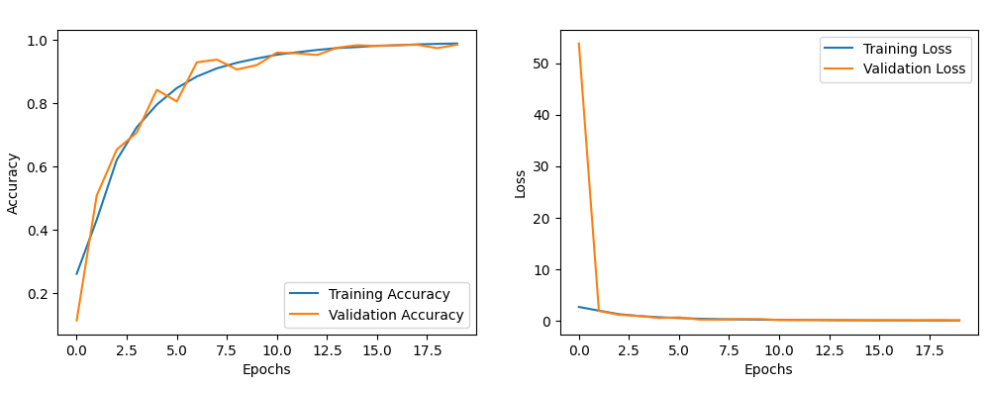
## Table 1. Model Configuration Table

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Model**  **Number** | **BASE**  **MODEL** | **Optimizer** | **Learning**  **Rate** | **Batch**  **Size** | **No of**  **Epochs** |
| Model 1 | ResNet50 | Adam | 0.01 | 32 | 20 |
| Model 2 | ResNet50 | SGD | 0.01 | 32 | 20 |
| Model 3 | ResNet50 | Adam | 0.001 | 32 | 08 |
| Model 4 | ResNet50 | Adam | 0.001 | 64 | 08 |
| Model 5 | MobileNet | SGD | 0.001 | 64 | 10 |
| Model 6 | MobileNet | SGD | 0.001 | 32 | 10 |
| Model 7 | InceptionV3 | Adam | 0.01 | 32 | 20 |

## Table 2. Performance Metrics Table

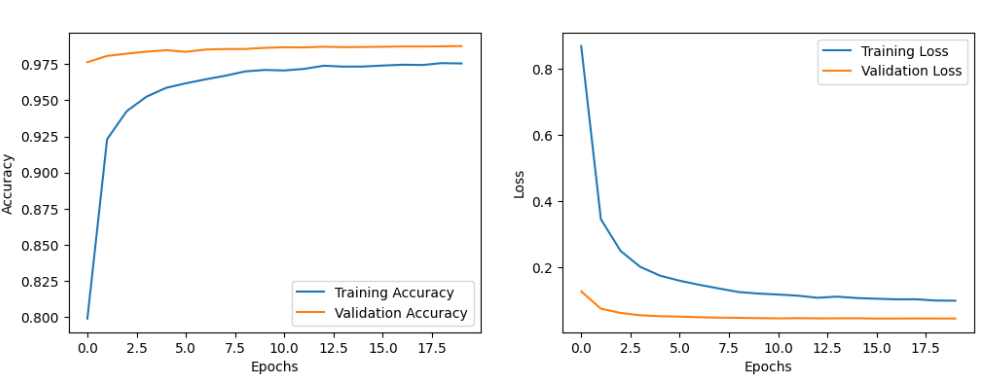
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Model**  **Number** | **ROC-AUC**  **Score** | **Accuracy** | **Precision** | **Recall** | **F1**  **score** |
| Model 1 | 0.94 | 0.9844 | 0.96 | 0.93 | 0.93 |
| Model 2 | 0.98 | 0.9875 | 0.99 | 0.99 | 0.99 |
| Model 3 | 0.98 | 0.9734 | 0.98 | 0.97 | 0.97 |
| Model 4 | 0.95 | 0.9374 | 0.94 | 0.93 | 0.93 |
| Model 5 | 0.93 | 0.9431 | 0.94 | 0.94 | 0.93 |
| Model 6 | 0.93 | 0.9709 | 0.97 | 0.97 | 0.97 |
| Model 7 | 0.82 | 0.9251 | 0.80 | 0.68 | 0.69 |

## Figure 7. Training vs Validation Accuracy and Loss



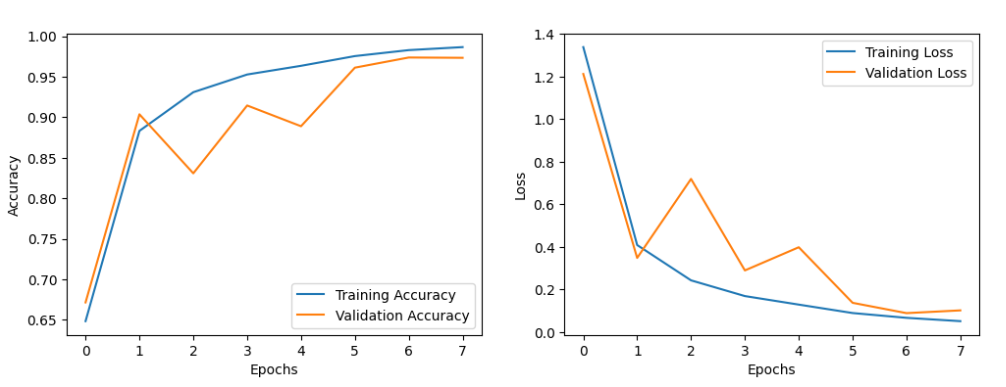
1. Accuracy (b) Loss

### Figure 7.1 Training vs. Validation (a)Accuracy and (b)Loss for ResNet with SGD (lr=0.01, batch\_size=32)



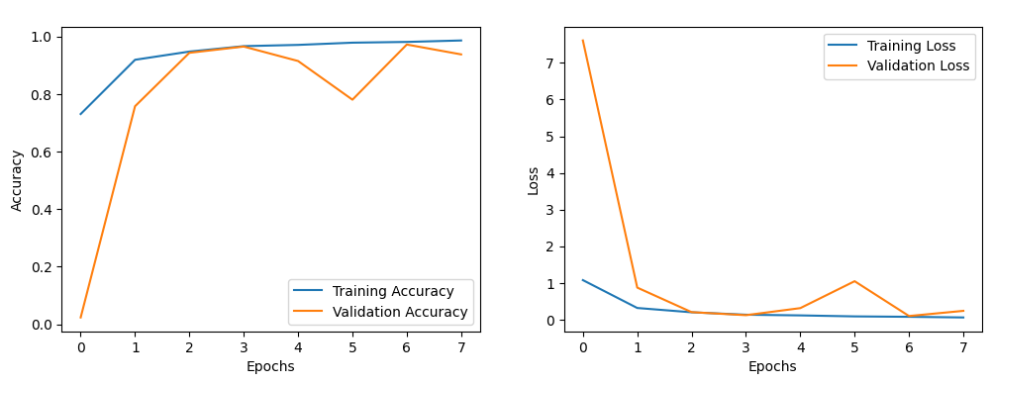
1. Accuracy (b) Loss

### Figure 7.2 Training vs. Validation (a)Accuracy and (b)Loss for ResNet with Adam (lr=0.01, batch\_size=32)



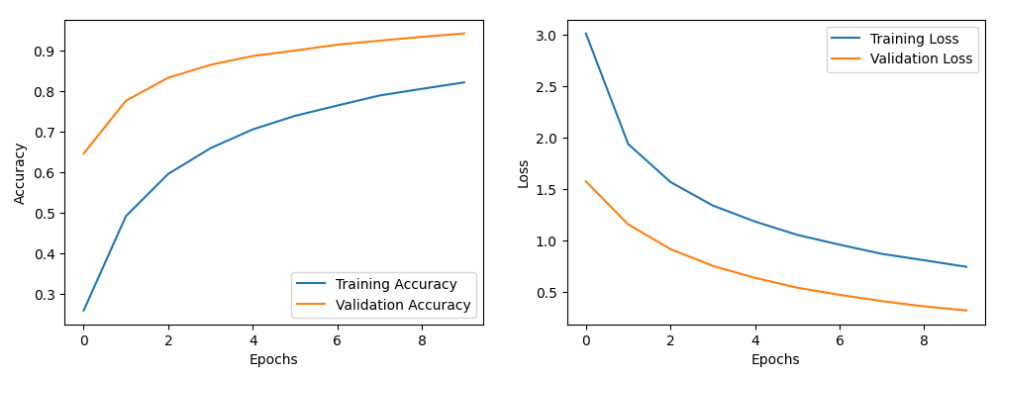
1. Accuracy (b) Loss

### Figure 7.3 Training vs. Validation (a)Accuracy and (b)Loss for ResNet with Adam (lr=0.001, batch\_size=32)



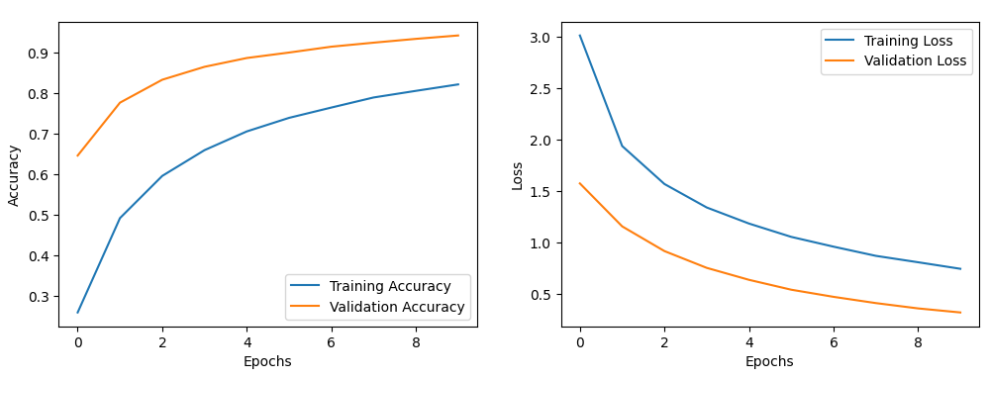
1. Accuracy (b) Loss

### Figure 7.4 Training vs. Validation (a)Accuracy and (b)Loss for ResNet with Adam (lr=0.01, batch\_size=64)



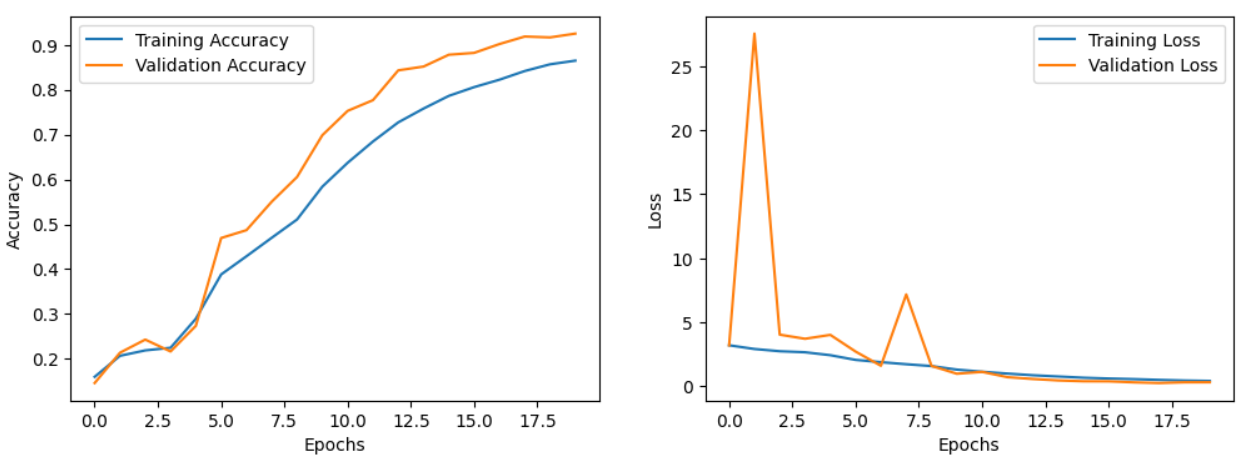
1. Accuracy (b) Loss

### Figure 7.5 Training vs. Validation (a)Accuracy and (b)Loss for MobileNet with SGD (lr=0.001, batch\_size=64)



1. Accuracy (b) Loss

### Figure 7.6 Training vs. Validation (a)Accuracy and (b)Loss for MobileNet with SGD (lr=0.001, batch\_size=32)



1. Accuracy (b) Loss

### Figure 7.7 Training vs. Validation (a)Accuracy and (b)Loss for InceptionV3 with Adam (lr=0.01, batch\_size=32)

# Experimental Results

Analyzing the results across ResNet, InceptionV3, and MobileNet reveals distinct performance characteristics.



## Figure 8. Confusion Matrices

|  |  |
| --- | --- |
|  |  |
| Fig 8.1 ResNet model with Adam (lr=0.01, batch\_size=32) | Fig 8.2 ResNet model with SGD (lr=0.01, batch\_size=32) |
|  |  |
| Fig 8.3 ResNet model with Adam (lr=0.001, batch\_size=32) | Fig 8.4 ResNet model with Adam (lr=0.001, batch\_size=64) |
| Model 5 Confusion Matrix |  |
| Fig 8.5 MobileNet model with SGD (lr=0.001, batch\_size=64) | Fig 8.6 MobileNet model with SGD (lr=0.001, batch\_size=32) |
|  |  |
| Fig 8.7 InceptionV3 model with Adam (lr=0.001, batch\_size=32) |  |

# Concluding Remarks

In the pursuit of developing a robust disease prediction system through a multi-model approach, this research has delved into the complexities of various deep learning architectures, hyperparameter configurations, and optimizers. The findings and insights gained throughout this study contribute to the broader landscape of healthcare analytics and offer valuable guidance for practitioners and researchers in the field.

**Model Performance**: The comprehensive evaluation of ResNet50, MobileNet, and InceptionV3 has provided a nuanced understanding of their respective performances in disease prediction scenarios. Each architecture exhibits unique strengths, emphasizing the importance of tailoring model choices to specific healthcare contexts.

**Hyperparameter Influence**: The systematic exploration of hyperparameters, including learning rates and batch sizes, has uncovered the intricate relationship between configuration choices and model accuracy. This understanding is crucial for optimizing disease prediction models under diverse circumstances.

**Optimizer Impact**: The comparative analysis of Adam and SGD optimizers has shed light on their distinct roles in influencing model convergence and final prediction accuracy. This insight empowers practitioners to make informed decisions based on computational resources and desired performance metrics.

**Adaptability of the Multi-Model Approach**: The research demonstrates the adaptability of the multi-model approach, showcasing how different architectures and configurations can be tailored to diverse datasets and prediction requirements. This versatility is a key asset in addressing the dynamic nature of healthcare data.

The remarkable accuracy, precision, recall, and F1 score achieved across multiple disease classes underscore the potential of deep learning in transforming plant disease identification. The insights gained from this research contribute not only to the field of precision agriculture but also to the broader discourse on the application of deep learning in complex domains.

In conclusion, this research underscores the importance of a nuanced and adaptive approach in disease prediction. By combining deep learning architectures and traditional machine learning methodologies, we have unraveled the intricacies of model behavior and configuration choices. The knowledge gained contributes to the broader discourse on leveraging advanced technologies for enhancing healthcare analytics. As the field of disease prediction continues to evolve, the insights gained from this study pave the way for informed decision-making, ultimately benefitting the larger goal of improving healthcare outcomes.

# Future Work

This research lays a foundation for further exploration and enhancements in the field of disease prediction through a multi-model approach. Several avenues for future work emerge, building upon the insights gained from this study:

1. **Fine-Tuning and Transfer Learning:**

Investigate the efficacy of fine-tuning pre-trained models on specific disease datasets to improve performance.

Explore transfer learning techniques, assessing the adaptability of models trained on one dataset to predict diseases in different contexts.

1. **Ensemble Learning:**

Implement ensemble learning strategies, combining predictions from multiple models, to enhance overall prediction robustness and accuracy.

Assess the impact of ensemble learning on model generalization and its potential for mitigating the limitations of individual models.

1. **Data Augmentation and Preprocessing:**

Explore advanced data augmentation techniques to generate diverse training samples, potentially improving model generalization.

Investigate the impact of different preprocessing methods on model performance, especially in scenarios with varying data quality and characteristics.

1. **Explainability and Interpretability:**

Integrate techniques for model explainability and interpretability, providing insights into the decision-making process of disease prediction models.

Explore methods to generate human-understandable explanations for model predictions, fostering trust and acceptance in real-world applications.

1. **Real-World Deployment Challenges:**

Investigate the challenges and considerations involved in deploying disease prediction models in real-world healthcare settings.

Address issues related to data privacy, ethical considerations, and integration with existing healthcare systems.

1. **Hyperparameter Optimization:**

Conduct further experiments to fine-tune hyperparameters, leveraging techniques such as Bayesian optimization or genetic algorithms.

Assess the robustness of optimal hyperparameter configurations across different disease datasets.

1. **Continuous Model Monitoring and Updating:**

Develop methodologies for continuous monitoring of model performance, allowing for timely updates and adaptations to evolving datasets and disease patterns.

Implement mechanisms for model retraining to ensure continued accuracy over time.

1. **Cross-Domain Generalization:**

Explore the generalization capabilities of models trained on plant disease datasets to predict diseases in other domains, such as human healthcare or veterinary medicine.

By addressing these areas in future research, we aim to further refine and advance the application of multi-model disease prediction approaches. The ongoing evolution of technology and healthcare practices provides ample opportunities for continuous improvement and innovation in this critical domain.

# References

1. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). [DOI: 10.1109/CVPR.2016.90](https://doi.org/10.1109/CVPR.2016.90)
2. Simonyan, K., & Zisserman, A. (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition. In International Conference on Learning Representations (ICLR).
3. TensorFlow. (2023). TensorFlow. <https://www.tensorflow.org/>
4. Brownlee, J. (2018). A Gentle Introduction to Transfer Learning for Deep Learning. Machine Learning Mastery.

<https://machinelearningmastery.com/transfer-learning-for-deep-learning/>

1. Singh, D., & Singh, R. K. (2019). A review of deep learning techniques for image classification. International Journal of Computer Applications, 182(37), 11-16.
2. Hughes, D. P., Salathe, M., & Bonhoeffer, S. (2012). The Effect of Diverse Training Sets on the Performance of an Image-Recognition System. PLoS ONE, 7(6), e38408.
3. Zoph, B., & Le, Q. V. (2016). Neural Architecture Search with Reinforcement Learning. In Proceedings of the International Conference on Learning Representations (ICLR).
4. Smith, L. N. (2018). A disciplined approach to neural network hyper-parameters: Part 1 -- learning rate, batch size, momentum, and weight decay. arXiv preprint arXiv:1803.09820.
5. Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). A Simple Framework for Contrastive Learning of Visual Representations. In International Conference on Machine Learning (ICML).
6. PlantVillage Dataset. <https://plantvillage.psu.edu/>
7. Luying Zhang, Yuchen Bian, Peng Jiang, Fengyun Zhang. (2023). A Transfer Residual Neural Network Based on ResNet-50 for Detection of Steel Surface Defects. <https://doi.org/10.3390/app13095260>
8. Yasamin Borhani, Javad Khoramdel & Esmaeil Najafi. (2022). A deep learning based approach for automated plant disease classification using vision transformer. <https://www.nature.com/articles/s41598-022-15163-0>
9. Muhammad Shoaib. (2022). A deep learning- based model for plant lesion segmentation, subtype identification, and survival probability estimation. <http://www.frontiersin.org/articles/10.3389/fpls.2022.1095547/full>