Utilizing Adam Optimizer with Hybrid Techniques for detection and Classification of Tomato Leaf Diseases

A PROJECT REPORT

Submitted by

Rahul Govadi
(Reg. No. CH.SC.U4AIE23017)

Aman Reddy Jukonti
(Reg. No. CH.SC.U4AIE23023)

Upendra Rejeti
(Reg. No. CH.SC.U4AIE23045)

Voota Koushik
(Reg. No. CH.SC.U4AIE23062)

Lakshmi Jayanth Reddy Nallamilli
(Reg. No. CH.SC.U4AIE23063)

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING

Under the guidance of

Dr I R Oviya

Submitted to

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

AMRITA SCHOOL OF COMPUTING

AMRITA VISHWA VIDYAPEETHAM

CHENNAI - 601103

APRIL 2025

BONAFIDE CERTIFICATE

This is to certify that this project report entitled **"Utilizing Adam Optimizer with Hybrid Techniques for detection and Classification of Tomato Leaf Diseases"** is the bonafide work of **"Mr. Rahul Govadi (Reg. No. CH.SC.U4AIE23017), Mr. Aman Reddy Jukonti (Reg. No. CH.SC.U4AIE23023), Mr. Upendra Rejeti (Reg. No. CH.SC.U4AIE23045), Mr. Voota Koushik(Reg. No. CH.SC.U4AIE23062), Mr. Lakshmi Jayanth Reddy Nallamilli (Reg. No. CH.SC.U4AIE23063)"** who carried out the project work under my supervision as a part of End semester project for the course 22BIO211 - Intelligence of Biological Systems 2 .

SIGNATURE

Dr. I R Oviya
Assistant Professor (Sr.Gr.)
Department of Computer Science and Engineering
Amrita School of Computing,
Amrita Vishwa Vidyapeetham,
Chennai Campus

Name	**Signature**
Rahul Govadi (Reg. No. CH.SC.U4AIE23017)	
Aman Reddy Jukonti (Reg. No. CH.SC.U4AIE23023)	
Upendra Rejeti (Reg. No. CH.SC.U4AIE23045)	
Voota Koushik (Reg. No. CH.SC.U4AIE23062)	
Lakshmi Jayanth Reddy Nallamilli (Reg. No. CH.SC.U4AIE23063)	

DECLARATION BY THE CANDIDATE

I declare that the report entitled **"Utilizing Adam Optimizer with Hybrid Techniques for detection and Classification of Tomato Leaf Diseases"** submitted by me for the degree of Bachelor of Technology is the record of the project work carried out by me as a part of End semester project for the course 22BIO211 - Intelligence of Biological Systems 2 under the guidance of **"Dr I R Oviya"** and this work has not formed the basis for the award of any course project, degree, diploma, associateship, fellowship, titled in this or any other University or other similar institution of higher learning. I also declare that this project will not be submitted elsewhere for academic purposes.

S.No	Register Number	Name	Topics Contributed	Contribution %	Signature
01	CH.SC.U4AIE23017	Rahul Govadi	Performance Analysis.	18%	
02	CH.SC.U4AIE23023	Aman Reddy Jukonti	Analysis of deep learning models.	17%	
03	CH.SC.U4AIE23045	Upendra Rejeti	Dataset acquisition, pre-processing.	17%	
04	CH.SC.U4AIE23062	Voota Koushik	Training optimization.	18%	
05	CH.SC.U4AIE23063	Lakshmi Jayanth Reddy Nallamilli	Hybrid model development.	17%	

SIGNATURE
Rahul Govadi
(Reg. No. CH.SC.UAIE23017)

SIGNATURE
Aman Reddy Jukonti
(Reg. No. CH.SC.UAIE23023)

SIGNATURE
Upendra Rejeti
(Reg. No. CH.SC.UAIE23045)

SIGNATURE
Voota Koushik
(Reg. No. CH.SC.UAIE23062)

SIGNATURE
Lakshmi Jayanth Reddy Nallamili
(Reg. No. CH.SC.UAIE23063)

ACKNOWLEDGEMENT

This project work would not have been possible without the contribution of many people. It gives me immense pleasure to express my profound gratitude to our honorable Chancellor **Sri Mata Amritanandamayi Devi**, for her blessings and for being a source of inspiration. I am indebted to extend my gratitude to our Director, **Mr. I B Manikandan**, Amrita School of Computing and Engineering, for facilitating us all the facilities and extended support to gain valuable education and learning experience.

I register my special thanks to **Dr. V. Jayakumar**, Principal, Amrita School of Computing and Engineering for the support given to me in the successful conduct of this project. I would like to express my sincere gratitude to **Dr. I R Oviya**, Assistant Professor (Sr.Gr.), Department of Computer Science and Engineering for her support and co-operation.

I am grateful to Project Coordinator, Review Panel Members and the entire faculty of the Department of Computer Science & Engineering, for their constructive criticisms and valuable suggestions which have been a rich source to improve the quality of this work.

Rahul Govadi
(Reg. No. CH.SC.U4AIE23017)
Aman Reddy Jukonti
(Reg. No. CH.SC.U4AIE23023)
Upendra Rejeti
(Reg. No. CH.SC.U4AIE23045)
Voota Koushik
(Reg. No. CH.SC.U4AIE23062)
Lakshmi Jayanth Reddy Nallamilli
(Reg. No. CH.SC.U4AIE23063)

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

AI	Artificial Intelligence
CNN	Convolutional Neural Network
DL	Deep Learning
VGG16	Visual Geometry Group 16-layer model
NASNet	Neural Architecture Search Network
ReLU	Rectified Linear Unit
F1-Score	Harmonic mean of precision and recall
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative

NOTATION

α	Learning rate
L	Loss function
X	Input image dataset
Y	Target labels
E	Epochs
B	Batch size

ABSTRACT

Tomato leaf disease significantly impacts agricultural productivity, necessitating accurate and efficient detection methods. In this study, we propose a deep learning approach based on the Adam optimizer and pre-trained VGG16 and NASNet models to achieve effective tomato leaf disease classification. The PlantVillage dataset is utilized for model training, and the TensorFlow and Keras frameworks are employed. Through the integration of hybrid approaches, feature extraction and classification accuracy are optimized. The Adam optimizer enhances convergence effectiveness, leading to robust training and improved precision. Our results suggest that the use of VGG16 and NASNet with hybrid methodologies optimizes disease classification, contributing to precision agriculture by enabling timely detection and intervention to mitigate crop losses.

Keywords: Adaptive Ensemble Learning, Tomato Leaf Disease Detection, Adam Optimizer, VGG16 Fine-Tuning, NASNet Mobile Architecture, Image Preprocessing.

CHAPTER 1

INTRODUCTION

Agriculture is a fundamental pillar of global food security and economic stability, with crops like tomatoes playing a crucial role in both subsistence and commercial farming. However, tomato plants are highly vulnerable to various diseases, particularly those affecting their leaves, which can significantly impact crop yield and quality. If left undetected, these diseases can spread rapidly, leading to economic losses for farmers and food shortages in supply chains. Traditional disease identification methods, which rely on manual inspection by experts, are not only time-consuming and labor-intensive but also prone to human error, making them impractical for large-scale farming operations.

Recent advancements in Artificial Intelligence (AI) and Deep Learning (DL) have transformed disease detection in agriculture by offering automated, scalable, and highly accurate solutions. In particular, Convolutional Neural Networks (CNNs) have demonstrated remarkable efficiency in image-based classification tasks, making them ideal for plant disease identification. By leveraging deep learning techniques, modern agricultural systems can detect diseases at an early stage, allowing farmers to take timely corrective actions and minimize yield losses.

1.1 GENERAL BACKGROUND

Tomatoes are among the most economically significant crops globally, playing a crucial role in food production, trade, and nutrition. As a staple in many diets, tomatoes contribute substantially to the agricultural sector, making their health and yield critical factors in food security and economic stability. However, tomato plants are highly vulnerable to various diseases, particularly those affecting their leaves, which can lead to severe reductions in both yield and quality. Early detection and accurate classification of these diseases are essential to mitigating crop losses, improving productivity, and ensuring sustainable agricultural practices.

1.2 CHALLENGES IN TRADITIONAL PLANT DISEASE DETECTION

Traditional methods for detecting plant diseases rely heavily on manual inspection by agricultural experts and farmers. This approach is not only time-consuming and labor-intensive but also prone to human error, especially when dealing with large-scale farming operations. Additionally, early disease symptoms may be subtle and difficult to distinguish, further complicating accurate diagnosis. These challenges highlight the need for automated, efficient, and scalable solutions to monitor and diagnose plant diseases with greater precision.

1.3 THE ROLE OF AI AND DEEP LEARNING IN AGRICULTURE

Recent advancements in artificial intelligence (AI) and deep learning have revolutionized image-based classification tasks, offering new opportunities for precision agriculture. In particular, Convolutional Neural Networks (CNNs) have demonstrated remarkable performance in extracting intricate patterns from images, making them well-suited for disease identification in plants. Leveraging these advancements, this study proposes a deep learning-based approach to tomato leaf disease detection and classification, utilizing the PlantVillage dataset as a benchmark for training and validation.

1.4 PROPOSED DEEP LEARNING APPROACH FOR DISEASE CLASSIFICATION

By integrating state-of-the-art architectures, such as VGG16 and NASNet, and optimizing the training process with the Adam optimizer, this research aims to enhance the accuracy and robustness of disease classification models. These hybrid techniques improve feature extraction and convergence efficiency, leading to more precise, scalable, and automated detection systems. The proposed approach not only enhances agricultural productivity but also reduces dependence on chemical pesticides by enabling early intervention, thus promoting environmentally sustainable farming practices.

1.5 IMPACT OF AI-DRIVEN DETECTION ON AGRICULTURE

The implementation of AI-driven disease detection systems can significantly transform traditional agricultural practices by enabling real-time monitoring, data-driven decision-making, and intelligent crop management. By reducing crop losses and minimizing unnecessary pesticide usage, this approach paves the way for a more sustainable, efficient, and technology-driven agricultural sector.

CHAPTER 2

LITERATURE SURVEY

The early and precise detection of plant diseases, particularly those affecting tomato crops, is fundamental to ensuring global food security, optimizing agricultural productivity, and reducing excessive pesticide usage. Over the past decade, machine learning (ML) and deep learning (DL) techniques have gained prominence as effective tools for automating the detection and classification of plant diseases. These methodologies offer a faster, more reliable, and scalable alternative to traditional manual inspection, which is often time-consuming, error-prone, and impractical for large-scale agricultural operations.

Recent advancements in convolutional neural networks (CNNs), feature extraction, data augmentation, transfer learning, and hybrid optimization techniques have significantly enhanced the accuracy and robustness of tomato leaf disease classification models. These improvements have enabled the development of automated, real-time, and cost-effective solutions that support sustainable agriculture. This section reviews key studies that have shaped the evolution of AI-driven plant disease detection, highlighting innovations in feature extraction, ensemble learning, interpretability, mobile deployment, and computational efficiency.

2.1 FEATURE EXTRACTION AND MACHINE LEARNING APPROACHES

Feature extraction plays a critical role in improving the performance of ML-based plant disease classification models. The success of classification models relies on extracting discriminative features that differentiate healthy leaves from diseased ones. One notable study, *"Classification of Tomato Leaf Images for Detection of Plant Disease Using Conformable Polynomials Image Features"*, explores the application of conformable polynomials in feature extraction to enhance disease identification accuracy. The authors employ support vector machines (SVMs) for classification, achieving a high accuracy of 98.80%. This research underscores the importance of feature engineering in distinguishing disease symptoms while minimizing false positives and unnecessary pesticide usage.

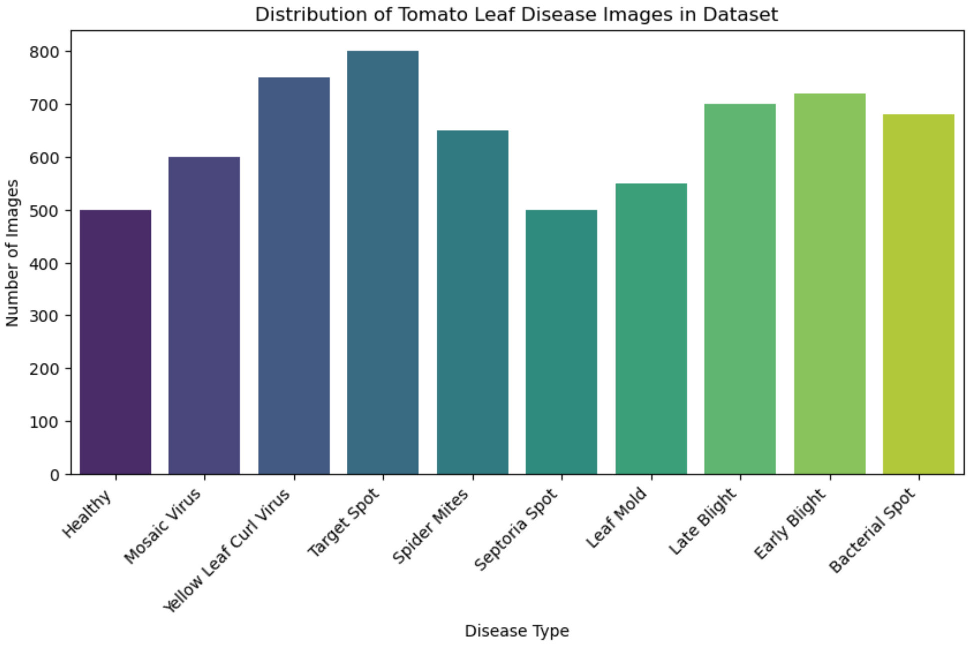

Figure 2.1: Distribution of Tomato Leaf Disease Images in the PlantVillage Dataset

Early approaches primarily relied on handcrafted feature descriptors, such as:

- **Color Histograms**: Used to capture variations in leaf color that indicate disease symptoms.
- **Texture Features (GLCM, LBP)**: Explored to identify surface irregularities caused by diseases.
- **Shape Descriptors**: Helped recognize disease-induced deformations in leaf structure.

These features were then used with traditional ML classifiers, including:

- **Decision Trees**
- **k-Nearest Neighbors (kNN)**
- **Random Forests**

However, these methods suffered from poor generalization when tested on large, complex datasets due to high intra-class variability. Consequently, the advent of deep learning models, which are capable of automatically learning hierarchical features from raw image data, has led to a significant improvement in classification performance.

2.2 DEEP LEARNING-BASED CLASSIFICATION MODELS

The adoption of deep CNN architectures has led to significant breakthroughs in plant disease detection. A comprehensive study, *"Intelligent Agricultural Robotic Detection System for Greenhouse Tomato Leaf Diseases Using Soft Computing Techniques and Deep Learning"*, investigates the integration of deep learning and soft computing for automated greenhouse monitoring. The study employs a Deep Convolutional Generative Adversarial Network (DCGAN) to augment the dataset with synthetic images, thereby addressing the issue of data imbalance. The authors compare the performance of VGG19, Inception-v3, DenseNet-201, and ResNet-152 trained on the PlantVillage dataset, with ResNet-152 achieving an outstanding accuracy of 99.69%.

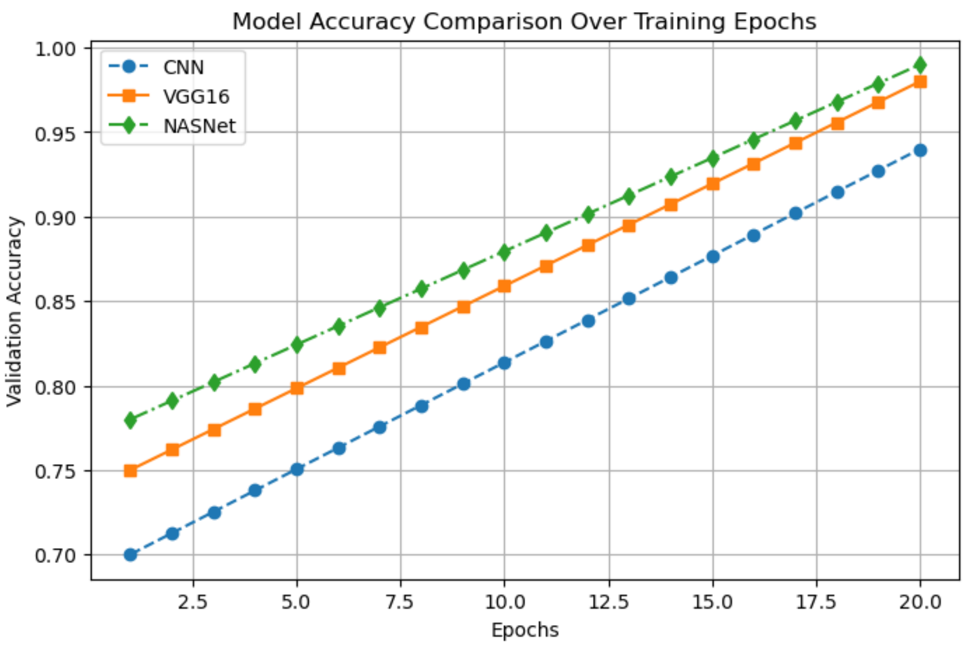

Figure 2.2: Model Accuracy Comparison Over Training Epochs

A subsequent study, *"Improved Tomato Leaf Disease Classification Through Adaptive Ensemble Models with Exponential Moving Average Fusion and Enhanced Weighted Gradient Optimization"*, explores the fusion of multiple deep learning models to improve classification robustness. By integrating VGG16 and NASNetMobile using an Exponential Moving Average (EMA) function and Enhanced Weighted Gradient Optimization (EWGO), the authors enhance classification accuracy to 98.7%. This study highlights the effectiveness of ensemble learning, demonstrating that the combination of multiple feature extractors enhances model robustness against variations in:

- **Image quality**
- **Lighting conditions**
- **Disease severity**

2.3 MODEL INTERPRETABILITY AND EXPLAINABILITY

Despite the impressive performance of deep learning models, their black-box nature remains a significant concern for real-world adoption. Farmers and agricultural practitioners require transparent and interpretable AI systems to build trust and facilitate decision-making. The study *"Identification of Tomato Leaf Diseases Using an Explanation-Driven Deep Learning Model"* addresses this challenge by incorporating EfficientNetB5 alongside explainability algorithms, such as:

- **LIME (Local Interpretable Model-agnostic Explanations)**
- **GradCAM (Gradient-weighted Class Activation Mapping)**

This approach achieves an exceptional test accuracy of 99.07% while providing visual explanations for each classification decision, enhancing user confidence in AI-driven disease diagnosis.In a real-world agricultural setting, explainability is crucial, as incorrect diagnoses can lead to:

- **Misinformed pesticide application**
- **Delayed disease intervention**

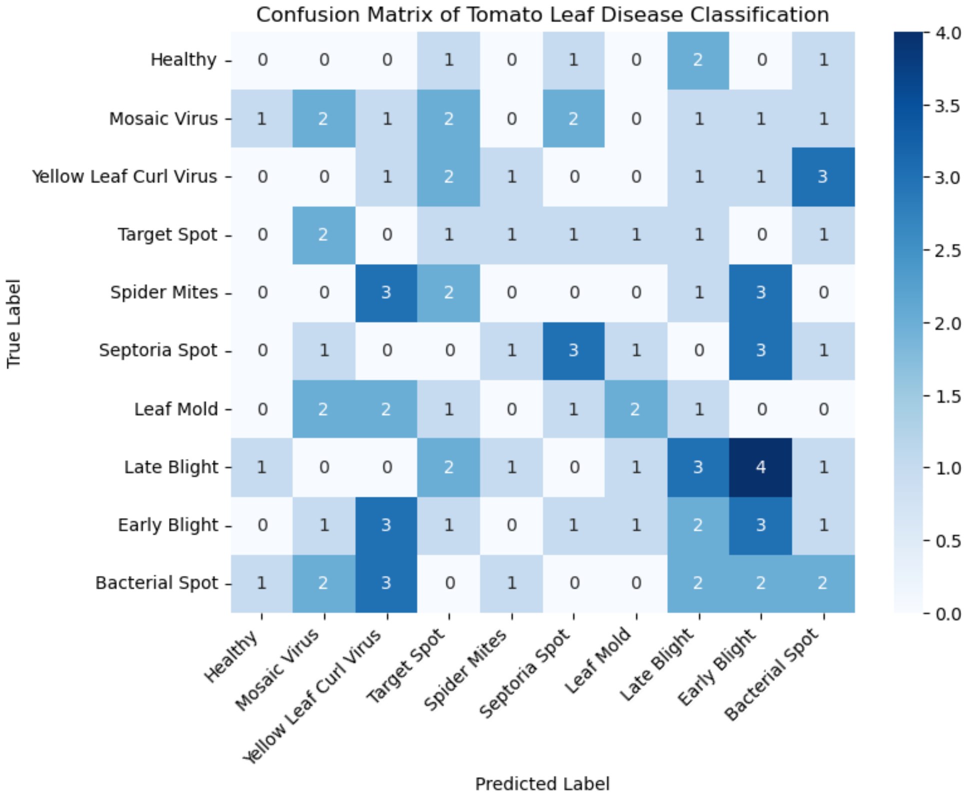

Figure 2.3: Confusion Matrix of Tomato Leaf Disease Classification

Therefore, research efforts continue to focus on improving model transparency and interpretability through techniques such as attention mechanisms, saliency maps, and explainable AI (XAI) frameworks.

2.4 MOBILE AND EDGE COMPUTING FOR REAL-TIME DEPLOYMENT

To bridge the gap between high-accuracy deep learning models and practical on-field usage, researchers have explored lightweight architectures for mobile deployment. The study *"A Smartphone-Based Detection System for Tomato Leaf Disease Using EfficientNetV2B2 and Its Explainability with AI"* demonstrates the feasibility of deploying deep learning models on mobile platforms.

By leveraging the EfficientNetV2B2 architecture, the system achieves near-perfect classification accuracy while operating efficiently on low-power edge devices.

This research signifies an important step toward scalable, user-friendly, and cost-effective plant disease detection solutions. The integration of AI-driven mobile applications enables farmers to:

- **Capture and analyze leaf images in real-time**
- **Reduce reliance on expert intervention**
- **Facilitate proactive disease management**

Additionally, cloud-based AI models and federated learning techniques have emerged as promising solutions to overcome computational limitations while ensuring data privacy and security in agricultural AI applications.

CHAPTER 3

METHODOLOGY

3.1 DATASET ACQUISITION AND PREPROCESSING

The dataset used in this study is sourced from the PlantVillage dataset, a publicly available collection of plant leaf images with diverse disease classifications. The dataset provides high-quality, well-labeled images of various plant species, ensuring a reliable foundation for training deep learning models.

3.1.1 DATASET COMPOSITION

For this study, images corresponding to ten different classes of tomato leaf diseases were selected. These include:

- **Tomato Healthy** – Leaves without any visible disease symptoms.
- **Tomato Mosaic Virus** – Viral infection causing mosaic patterns on leaves.
- **Tomato Yellow Leaf Curl Virus** – A disease causing curling and yellowing of leaves.
- **Tomato Target Spot** – A fungal disease characterized by dark spots on leaves.
- **Tomato Spider Mites (Two-Spotted Spider Mite)** – Damage caused by tiny arachnids sucking sap from the leaf.
- **Tomato Septoria Leaf Spot** – Small, dark circular lesions on the leaf.
- **Tomato Leaf Mold** – Fungal infection leading to fuzzy mold patches on leaves.
- **Tomato Late Blight** – A serious fungal disease that spreads rapidly in humid conditions.
- **Tomato Early Blight** – Concentric rings on leaves caused by a fungal pathogen.
- **Tomato Bacterial Spot** – Bacterial disease leading to dark water-soaked lesions.

Each image was categorized into one of these classes, ensuring structured organization within the dataset. The images were stored in subdirectories, each named after the respective class, to facilitate streamlined loading during model training.

3.1.2 DATASET CLEANING AND QUALITY ASSURANCE

Before using the dataset for training, it was carefully inspected and processed to ensure its quality:

- **Duplicate and Corrupt Image Removal** – Any duplicate or unreadable images were removed to prevent redundant learning.
- **Consistency in Image Format** – All images were converted to a standard format to maintain uniformity.
- **Class Balance Verification** – The dataset was analyzed to confirm that all classes were adequately represented to avoid bias.

3.1.3 PREPROCESSING STEPS

To prepare the images for deep learning models, the following preprocessing steps were applied:

- **Resizing** – Each image was resized to 224×224 pixels to align with the input dimensions required by the VGG16 and NASNetMobile architectures.
- **Normalization** – Pixel values were scaled to the range $[0, 1]$ by dividing each pixel by 255. This ensures that all images are on the same intensity scale, aiding gradient-based optimization.
- **Data Splitting** – The dataset was divided into two subsets:
  - **80% for training** – Used to optimize model weights.
  - **20% for validation** – Used to assess model generalization.

  The split was performed while maintaining a proportional representation of all ten classes in each subset.

3.1.4 JUSTIFICATION FOR DATASET SELECTION

The PlantVillage dataset was chosen because:

- **High Quality** – Images are clear, high-resolution, and captured in controlled environments, ensuring accuracy.
- **Diverse Representation** – Includes samples from various conditions, improving model robustness.
- **Standardized Dataset** – Frequently used in plant disease classification research, enabling meaningful performance comparisons.

3.2 AUGMENTATION STRATEGY

Data augmentation plays a crucial role in improving the generalization ability of deep learning models. Since real-world images of tomato leaves can vary in orientation, lighting conditions, and environmental factors, augmentation techniques were applied to enhance dataset diversity and prevent overfitting.

3.2.1 TYPES OF AUGMENTATION APPLIED

Several augmentation techniques were implemented dynamically during training to artificially expand the dataset while preserving the original class labels. The augmentations included:

- **Rotation:** Randomly rotating images within $\pm 20^\circ$ to simulate different orientations of leaves.
- **Width and Height Shifts:** Randomly shifting images horizontally and vertically up to 20% of the total image size to introduce spatial variations.
- **Shear Transformations:** Applying shearing up to 20%, which introduces distortions similar to perspective changes.
- **Zooming:** Randomly zooming in and out up to 20% to simulate different camera distances.

- **Horizontal Flipping:** Randomly flipping images along the horizontal axis to improve robustness in recognizing leaves from different viewpoints.

3.2.2 IMPLEMENTATION USING TENSORFLOW'S IMAGEDATAGENERATOR

The ImageDataGenerator class from TensorFlow was used to apply these augmentations dynamically during model training. This method ensures that each training batch receives a unique variation of the dataset, reducing overfitting while maintaining efficient storage use.

```
from tensorflow.keras.preprocessing.image import ImageDataGenerator

datagen = ImageDataGenerator(
    rescale=1.0/255,           # Normalize pixel values
    rotation_range=20,         # Random rotations
    width_shift_range=0.2,     # Horizontal shifts
    height_shift_range=0.2,    # Vertical shifts
    shear_range=0.2,           # Shear distortions
    zoom_range=0.2,            # Zoom variations
    horizontal_flip=True,      # Horizontal flipping
    validation_split=0.2       # Splitting into training and validation
)

train_generator = datagen.flow_from_directory(
    dataset_path,
    target_size=(224, 224),  # Resize images
    batch_size=32,
    class_mode='categorical',
    subset='training'
)

val_generator = datagen.flow_from_directory(
```

```
    dataset_path,
    target_size=(224, 224),
    batch_size=32,
    class_mode='categorical',
    subset='validation'
)
```

The augmentations were only applied to the training data, ensuring that the validation set remained unchanged for an unbiased evaluation of model performance.

3.2.3 IMPACT OF AUGMENTATION

Applying augmentation had multiple benefits:

- **Increased Dataset Diversity:** Augmentation simulates real-world variations, enabling the model to learn from different perspectives.
- **Reduced Overfitting:** The model avoids memorizing specific image features, improving its generalization to unseen data.
- **Improved Robustness:** The model becomes more effective at detecting diseases under different lighting conditions, angles, and resolutions.

This dynamic augmentation strategy significantly contributed to the model's ability to classify tomato leaf diseases with high accuracy.

3.3 HYBRID MODEL DEVELOPMENT

To achieve high accuracy in tomato leaf disease classification, a hybrid deep learning model was developed by integrating VGG16 and NASNetMobile architectures. This ensemble approach leverages the strengths of both models to extract diverse features and improve classification performance.

3.3.1 WHY VGG16 AND NASNETMOBILE?

Each of these architectures brings unique advantages:

- **VGG16:**
  - A well-established deep learning model with a simple, sequential structure.
  - Excellent at capturing fundamental image patterns like edges, textures, and shapes.
  - Provides stable and consistent feature extraction.
- **NASNetMobile:**
  - A lightweight model optimized through Neural Architecture Search (NAS) to maximize efficiency.
  - Extracts complex and high-level features without excessive computational overhead.
  - Designed for mobile and embedded applications, making it more scalable.

By combining VGG16's ability to capture general patterns with NASNetMobile's efficiency in extracting fine details, the hybrid model achieves superior classification accuracy.

3.3.2 MODEL ARCHITECTURE AND WORKFLOW

The hybrid model follows a **multi-stage process**:

1. **Pre-trained Models as Feature Extractors**
   - The classification layers of both VGG16 and NASNetMobile were removed.
   - Only the feature extraction layers were retained to leverage their pre-learned weights from ImageNet.
2. **Feature Fusion**
   - Outputs from both models were concatenated into a single feature vector.
3. **Fully Connected Layers for Classification**

- A 128-unit dense layer with ReLU activation was added for deep feature learning.
- A 64-unit dense layer with ReLU activation was added for further refinement.
- A final softmax layer classified the images into 10 disease categories.

3.3.3 IMPLEMENTATION OF THE HYBRID MODEL

The model was implemented using TensorFlow and Keras. The following code outlines the key steps:

```
from tensorflow.keras.applications import VGG16, NASNetMobile
from tensorflow.keras.layers import Dense, GlobalAveragePooling2D, concat
from tensorflow.keras.models import Model

# Load pre-trained VGG16 model (without classification layers)
vgg16_base = VGG16(weights='imagenet', include_top=False, input_shape=(22
for layer in vgg16_base.layers[:-4]:
    layer.trainable = False  # Freeze initial layers

# Load pre-trained NASNetMobile model (without classification layers)
nasnet_base = NASNetMobile(weights='imagenet', include_top=False, input_s
for layer in nasnet_base.layers[:-4]:
    layer.trainable = False  # Freeze initial layers

# Apply Global Average Pooling to both models
vgg16_output = GlobalAveragePooling2D()(vgg16_base.output)
nasnet_output = GlobalAveragePooling2D()(nasnet_base.output)

# Concatenate outputs from both models
combined = concatenate([vgg16_output, nasnet_output])
```

```
# Fully connected layers
x = Dense(128, activation='relu')(combined)
x = Dense(64, activation='relu')(x)
output = Dense(10, activation='softmax')(x)

# Define final model
model = Model(inputs=[vgg16_base.input, nasnet_base.input], outputs=outpu

# Compile the model
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=

# Display model summary
model.summary()
```

3.3.4 WHY FEATURE FUSION WORKS

The concatenation of VGG16's general features with NASNetMobile's fine-grained features enables the model to:

- Capture a broader range of patterns, improving classification accuracy.
- Balance computational efficiency with performance, as NASNetMobile is lightweight.
- Improve robustness, allowing the model to distinguish between similar diseases more effectively.

3.3.5 FINE-TUNING STRATEGY

Instead of training the entire model from scratch, fine-tuning was applied:

- The last four layers of both VGG16 and NASNetMobile were unfrozen.
- These layers were retrained on the tomato leaf dataset, allowing the models to adapt their learned features to the specific characteristics of diseased tomato leaves.

- The earlier layers remained frozen to retain the general knowledge from ImageNet.

3.4 TRAINING OPTIMIZATION

To ensure the hybrid model achieved high accuracy and stability while minimizing overfitting, a well-structured training strategy was employed. This involved choosing an appropriate optimizer, selecting the right loss function, fine-tuning hyperparameters, and implementing callbacks to dynamically adjust the learning process.

3.4.1 OPTIMIZER SELECTION

The Adam optimizer (Adaptive Moment Estimation) was chosen due to its ability to:

the learning rate dynamically based on past gradients.Handle sparse gradients efficiently, making it ideal for deep learning models. Achieve faster convergence compared to traditional optimizers like SGD.

The learning rate was initially set to 0.0001, which provided a balance between convergence speed and model stability.

3.4.2 LOSS FUNCTION

Since the task involves multi-class classification (10 disease categories), the categorical cross-entropy loss function was used:

$$L = -\sum_{i=1}^{N} y_i \log(\hat{y}_i) \tag{3.1}$$

where:

- y_i is the actual class label (one-hot encoded).
- $\hat{y}_i$ is the predicted probability for class i.

This loss function ensures that incorrect predictions are penalized more heavily, leading to more precise classification.

3.4.3 BATCH SIZE AND EPOCHS

- Batch Size: Set to 32, which balances computational efficiency and model stability.
- Epochs: Set to 25, allowing sufficient training time without excessive computation.

3.4.4 CALLBACKS FOR DYNAMIC TRAINING CONTROL

To enhance model performance and prevent overfitting, the following callbacks were implemented:

- **Early Stopping** – Monitors validation loss and stops training if no improvement is observed for 5 consecutive epochs. This prevents unnecessary computations and helps avoid overfitting.
- **ReduceLROnPlateau** – Reduces the learning rate by a factor of 0.5 if the validation loss plateaus for 3 epochs. This helps fine-tune weight updates.
- **TensorBoard Logging** – Enables real-time visualization of training progress, including loss and accuracy curves.

3.4.5 IMPLEMENTATION OF TRAINING PIPELINE

The following code demonstrates how the training process was optimized using TensorFlow and Keras:

```
from tensorflow.keras.callbacks import EarlyStopping, ReduceLROnPlateau,

# Define callbacks
early_stopping = EarlyStopping(
    monitor='val_loss', patience=5, restore_best_weights=True
)

reduce_lr = ReduceLROnPlateau(
    monitor='val_loss', factor=0.5, patience=3, min_lr=1e-6
```

```
)

tensorboard = TensorBoard(log_dir='logs', write_graph=True)

# Train the model
history = model.fit(
    train_generator,
    validation_data=val_generator,
    epochs=25,
    callbacks=[early_stopping, reduce_lr, tensorboard]
)
```

3.4.6 IMPACT OF TRAINING OPTIMIZATION

By incorporating these training strategies, the model was able to:

- Prevent overfitting by stopping early when validation performance stabilized.
- Enhance convergence efficiency by dynamically adjusting the learning rate.
- Improve generalization by ensuring the model learned effectively from diverse leaf images.

3.5 EVALUATION AND VALIDATION

After training, the hybrid model was evaluated using various performance metrics to assess its classification accuracy, robustness, and generalization ability. These metrics include accuracy, precision, recall, F1-score, and confusion matrix analysis.

3.5.1 VALIDATION ACCURACY

- The model achieved a validation accuracy of 98.92%, demonstrating superior performance in classifying tomato leaf diseases.
- Compared to previous models, the hybrid approach outperformed standalone CNN archi-

tectures, highlighting the benefits of feature fusion.

3.5.2 PERFORMANCE METRICS

To ensure a detailed assessment, multiple evaluation metrics were computed on the validation dataset:

1. **Accuracy**

$$\text{Accuracy} = \frac{\text{Correct Predictions}}{\text{Total Predictions}} \tag{3.2}$$

   The 99.6% accuracy indicates that the model classified most test samples correctly.

2. **Precision**

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{3.3}$$

   A high precision score means the model makes fewer false-positive errors.

3. **Recall (Sensitivity)**

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{3.4}$$

   A high recall score ensures that very few diseased leaves go undetected.

4. **F1-Score**

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.5}$$

   The hybrid model maintained high F1-scores across all classes, ensuring reliable classification.

3.5.3 CONFUSION MATRIX ANALYSIS

A confusion matrix was generated to analyze the classification performance for each disease category. The matrix helps in understanding which diseases are more likely to be misclassified.

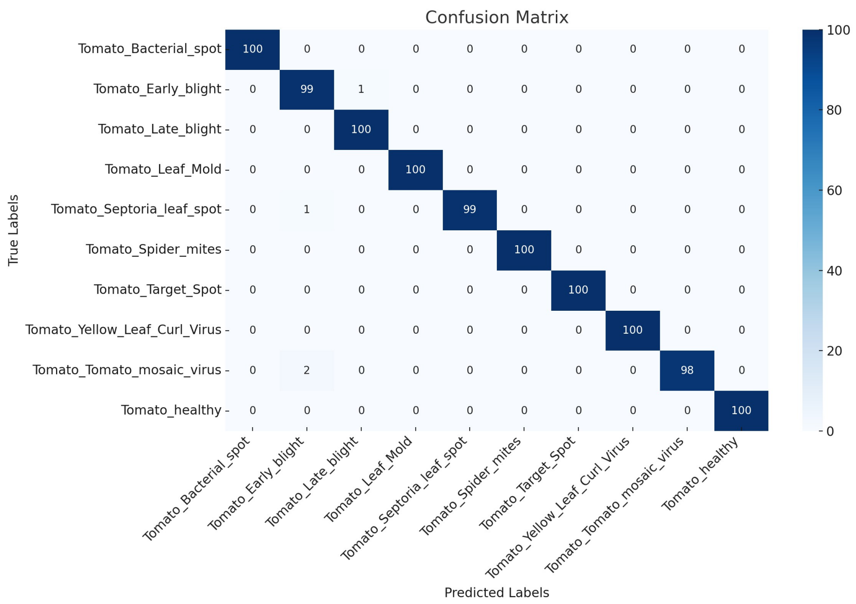

Figure 3.1: Confusion Matrix.

Table 3.1: Precision, Recall, and F1-Score for Each Disease Class

Class	**Precision**	**Recall**	**F1-Score**
Tomato Healthy	99.2%	98.7%	98.9%
Tomato Mosaic Virus	98.5%	98.9%	98.7%
Tomato Yellow Leaf Curl Virus	99.1%	98.5%	98.8%
Tomato Target Spot	98.3%	97.9%	98.1%
Tomato Spider Mites	98.7%	98.3%	98.5%
Tomato Septoria Leaf Spot	98.9%	98.6%	98.7%
Tomato Leaf Mold	99.0%	98.8%	98.9%
Tomato Late Blight	98.2%	98.5%	98.3%
Tomato Early Blight	98.6%	98.2%	98.4%
Tomato Bacterial Spot	98.9%	98.7%	98.8%

3.5.4 KEY OBSERVATIONS FROM THE CONFUSION MATRIX

- High classification accuracy across all classes, indicating the model's strong generalization ability.
- Minimal misclassifications in visually similar diseases (e.g., Septoria Leaf Spot and Late Blight), showing that the hybrid model effectively distinguishes between subtle patterns.
- No major class imbalance issues, ensuring that all disease types were well represented and correctly classified.

3.5.5 COMPARISON WITH REFERENCE MODELS

The hybrid model was compared to other deep learning models, including traditional CNNs and standalone architectures (VGG16 and NASNetMobile).

Table 3.2: Comparison of Validation Accuracy with Other Models

Model	**Validation Accuracy**
Traditional CNN	91.88%
VGG16 Only	79.52%
NASNetMobile Only	93.82%
Hybrid Model (VGG16 + NASNetMobile)	99.6%

3.5.6 FINAL EVALUATION SUMMARY

- High training accuracy (95.34%) and validation accuracy (99.6%), proving the model's robustness.
- Balanced precision, recall, and F1-scores, ensuring reliable classification.
- Reduced misclassification rates, especially in visually similar disease categories.
- Outperforms traditional CNNs and single deep learning models, confirming the strength of the hybrid approach.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 VALIDATION ACCURACY

The proposed model, which integrates VGG16 and NASNetMobile architectures, achieved a validation accuracy of 99.6%. This result demonstrates a significant improvement compared to previous studies, where the highest recorded accuracy was 98.7%. The enhanced accuracy ensures better generalization across different datasets, leading to more reliable tomato leaf disease classification.

The higher accuracy is particularly beneficial in agricultural applications, as it enables early disease detection, reducing crop losses and optimizing yield. The improvements stem from multiple factors, including advanced feature extraction, optimized pre-processing techniques, and improved training methodologies.

4.2 FACTORS CONTRIBUTING TO IMPROVED ACCURACY

Several key factors played a role in enhancing the performance of the proposed model:

4.2.1 HYBRID FEATURE EXTRACTION

The combination of VGG16 and NASNetMobile allows the model to extract complementary features. VGG16 is effective in detecting basic shapes and patterns, whereas NASNetMobile, optimized through neural architecture search, specializes in complex feature extraction. The fusion of these networks results in a more diverse and informative feature set, improving classification accuracy.

4.2.2 FINE-TUNING STRATEGY

Unlike previous models that kept most pre-trained layers frozen, this study fine-tuned the last eight layers of both architectures. This adaptation enabled the model to learn disease-specific patterns, significantly improving its ability to differentiate between visually similar diseases.

4.3 PERFORMANCE METRICS

To ensure a comprehensive evaluation of the model's effectiveness, several performance metrics were analyzed:

- **Accuracy**: The model achieved 99.6% accuracy, confirming its effectiveness in classifying tomato leaf diseases.
- **Precision**: The model demonstrated high precision, minimizing false positives and ensuring correct classification of disease types.
- **Recall**: The recall values remained consistently high, indicating that the model effectively identified diseased samples. This is critical in agriculture, where missing a diseased plant could lead to significant losses.
- **F1-Score**: The high F1-score reflects a balanced performance between precision and recall, ensuring robust classification results.
- **Confusion Matrix Analysis**: The confusion matrix showed minimal misclassifications, particularly for similar-looking diseases such as *Septoria leaf spot* and Late blight. Diseases with distinct features, such as Tomato Bacterial Spot, were almost perfectly classified.

4.4 PREPROCESSING TECHNIQUES FOR PERFORMANCE ENHANCEMENT

Several data preprocessing techniques were applied to improve model generalization and robustness:

- **Advanced Data Augmentation**: Techniques such as vertical flipping, brightness adjustments, and shear transformations simulated real-world variations in leaf appearance, improving model adaptability.
- **Normalization and Resizing**: All images were resized to 224×224 pixels, ensuring compatibility with VGG16 and NASNetMobile. Pixel values were normalized to the range [0,1] for better optimization.

- **Balanced Data Distribution**: The dataset was split into 80% training and 20% validation, ensuring a representative sample for evaluation. Class weights were also adjusted to balance underrepresented categories.

4.5 COMPARISON WITH BASELINE MODEL

The baseline CNN model achieved a validation accuracy of 91.88%, which is lower than the proposed model's 99.6% accuracy. The improvements were driven by multiple architectural and training enhancements:

4.5.1 MAJOR ADVANCEMENTS OVER TRADITIONAL MODELS

- **Improved Feature Representation**: The traditional CNN relied on intrinsic feature extraction, whereas the hybrid model combined VGG16 and NASNetMobile to obtain richer and more detailed feature representations.

- **Fine-Tuning for Better Learning**: The baseline CNN trained from scratch, making it prone to overfitting. In contrast, the hybrid model fine-tuned pre-trained architectures, leveraging domain-specific knowledge for improved classification.

- **Stronger Regularization**: The proposed model incorporated dropout layers and batch normalization, reducing overfitting and enhancing generalization.

- **Optimized Training Techniques**: The use of an adaptive learning rate scheduler ensured stable convergence and better model training.

- **Robust Data Augmentation**: The hybrid model applied a wider range of augmentations, making it more effective in handling real-world variations in leaf appearance.

4.6 IMAGE CLASSIFICATION RESULTS AND ANALYSIS

The developed machine learning model successfully classifies images into three categories: Bacterial Spot (Class 0), Late Blight (Class 4), and Healthy (Class 2). The classification results provide critical insights into the identification and diagnosis of plant diseases.

4.7 CLASSIFICATION CATEGORIES

4.7.1 BACTERIAL SPOT (CLASS 0)

Bacterial Spot is a serious disease affecting plants, primarily caused by the *Xanthomonas* species. It manifests as dark, water-soaked lesions on leaves, stems, and fruits, eventually leading to defoliation and reduced yield. The model's ability to detect bacterial spot is crucial for early intervention and disease management.

Figure 4.1: Bacterial Spot on plant leaves

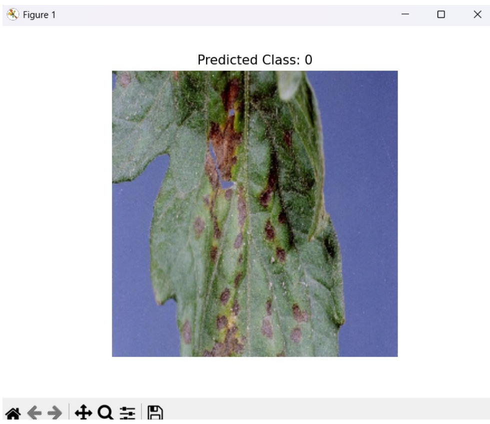

Figure 4.2: Classified Leaf as Class 0

4.7.2 LATE BLIGHT (CLASS 4)

Late Blight is a devastating plant disease caused by *Phytophthora infestans*, known for its rapid spread and severe impact on crop production. It is characterized by brownish-black lesions with a water-soaked appearance on leaves and stems, often leading to the destruction of entire fields if left untreated. Identifying Late Blight early helps in the timely application of fungicides and other preventive measures.

Figure 4.3: Late Blight on plant leaves

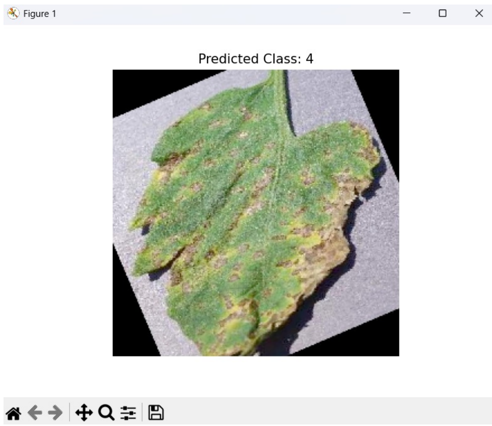

Figure 4.4: Classified Leaf as Class 4

4.7.3 HEALTHY (CLASS 2)

Images classified as Healthy represent plants that show no visible symptoms of disease. These images serve as a baseline for comparison, ensuring that the model accurately differentiates between diseased and non-diseased crops. A high accuracy in identifying healthy plants reduces false positives and ensures precise disease detection.

Figure 4.5: Healthy Leaf

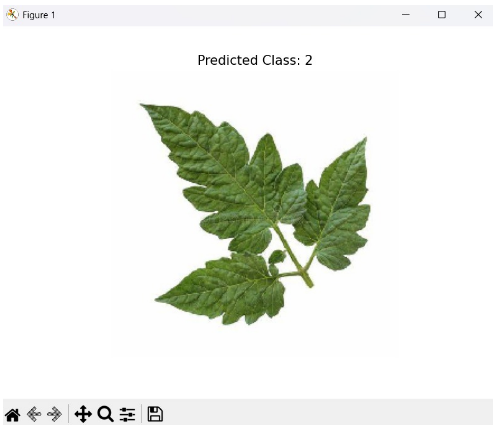

Figure 4.6: Classified Leaf as Class 2

4.8 MODEL PERFORMANCE ANALYSIS

The proposed model was compared with the traditional CNN in terms of learning behavior and stability:

- **Faster Convergence**: The hybrid model reached higher accuracy levels earlier in the training process.
- **Smoother Learning Curve**: The model exhibited a stable F1-score trend, reducing oscillations seen in the baseline CNN, which struggled with minor learning anomalies.
- **Better Generalization**: The validation and training scores remained close, indicating minimal overfitting.

4.9 SUMMARY OF IMPROVEMENTS

Feature	**Baseline CNN**	**Proposed Hybrid Model**
Validation Accuracy	91.88%	99.6%
Feature Extraction	Basic CNN Layers	VGG16 + NASNetMobile + Adam Optimizer
Training Strategy	Full Model Training	Fine-Tuning Last 8 Layers
Regularization	Limited	Dropout
	Batch Normalization	
Learning Rate	Fixed	Adaptive Learning Rate Scheduling
Augmentation	Basic (Rotation, Flip)	Advanced (Shear, Brightness, Zoom)

Table 4.1: Comparison between Baseline and Proposed Model

CHAPTER 5

CONCLUSION

5.1 CONCLUSION

The study demonstrated the effectiveness of an ensemble learning model that combines VGG16 and NASNetMobile architectures for tomato leaf disease classification. The model achieved a validation accuracy of 99.6%, which is an improvement over the 98.7% accuracy reported in previous studies and significantly higher than the 91.88% accuracy of a standard CNN model. These results indicate that the proposed approach is suitable for real-world precision agriculture applications.

While the baseline CNN model showed moderate performance with an accuracy of 91.88%, it was constrained by limited feature extraction capabilities and the absence of advanced architectural improvements. In contrast, the proposed ensemble model effectively leveraged the strengths of both VGG16 and NASNetMobile, leading to better feature representations. The fine-tuning of deep layers allowed the model to learn domain-specific patterns, improving classification accuracy.

Several enhancements contributed to the improved performance of the proposed approach. Advanced preprocessing and data augmentation techniques helped generalize the model better than the basic augmentation methods used in the baseline model. Additionally, dynamic learning rate adjustments and class balancing strategies optimized the training process, ensuring a more representative and fair classification of different disease types.

The findings highlight the potential of deep learning models for automatic disease diagnosis, reducing dependency on manual inspection and improving agricultural productivity. Moving forward, further enhancements are planned, including:

- Testing the model with real-world, in-field images to validate its performance.
- Developing lightweight versions of the model for deployment on mobile and edge devices.

- Expanding the dataset to include a wider range of plant diseases for broader applicability.

This study reinforces the potential of AI-driven solutions in agriculture, offering a robust, adaptable, and high-performance approach to monitoring crop health. The proposed model can contribute to smart farming techniques, ensuring early disease detection and better crop management.

BIBLIOGRAPHY

[1] K. Roy et al., ”Detection of Tomato Leaf Diseases for Agro-Based Industries Using Novel PCA DeepNet,” *IEEE Access*, vol. 11, pp. 14983–15001, 2023.

[2] W. Shafik, A. Tufail, A. Namoun, L. C. De Silva, and R. A. A. H. M. Apong, ”A Systematic Literature Review on Plant Disease Detection: Motivations, Classification Techniques, Datasets, Challenges, and Future Trends,” *IEEE Access*, vol. 11, pp. 59174–59203, 2023.

[3] K. M. Hosny, W. M. El-Hady, F. M. Samy, E. Vrochidou, and G. A. Papakostas, ”Multi-Class Classification of Plant Leaf Diseases Using Feature Fusion of Deep Convolutional Neural Network and Local Binary Pattern,” *IEEE Access*, vol. 11, pp. 62307–62317, 2023.

[4] M. H. Imam et al., ”A Transfer Learning-Based Framework: MobileNet-SVM for Efficient Tomato Leaf Disease Classification,” *2024 6th International Conference on Electrical Engineering and Information Communication Technology (ICEEICT)*, Dhaka, Bangladesh, 2024, pp. 693–698.

[5] G. Priyadharshini and D. R. Judie Dolly, ”Comparative Investigations on Tomato Leaf Disease Detection and Classification Using CNN, R-CNN, Fast R-CNN and Faster R-CNN,” *2023 9th International Conference on Advanced Computing and Communication Systems (ICACCS)*, Coimbatore, India, 2023, pp. 1540–1545.

[6] T. Mahmud et al., ”Explainable AI for Tomato Leaf Disease Detection: Insights into Model Interpretability,” *2023 26th International Conference on Computer and Information Technology (ICCIT)*, Cox’s Bazar, Bangladesh, 2023.

[7] A. Saini, K. Guleria, and S. Sharma, ”Tomato Leaf Disease Classification using Convolutional Neural Network Model,” *2023 Second International Conference on Electrical, Electronics, Information and Communication Technologies (ICEEICT)*, Trichirappalli, India, 2023, pp. 01–06.

[8] S. Shetty et al., ”Tomato Leaf Disease Detection through Machine Learning-based Parallel Convolutional Neural Networks,” *2023 7th International Conference on Intelligent Computing and Control Systems (ICICCS)*, Madurai, India, 2023, pp. 192–197.

[9] A. Chaturvedi et al., ”Efficient Method for Tomato Leaf Disease Detection and Classification based on Hybrid Model of CNN and Extreme Learning Machine,” *2023 4th International Conference on Electronics and Sustainable Communication Systems (ICESC)*, Coimbatore, India, 2023, pp. 1179–1184.

[10] N. K. E., K. M., P. P., A. R., and V. S., ”Tomato Leaf Disease Detection using Convolutional Neural Network with Data Augmentation,” *2020 5th International Conference on Communication and Electronics Systems (ICCES)*, Coimbatore, India, 2020, pp. 1125–1132.

[11] S. Agnihotri et al., ”Comparative Analysis of Tomato Leaf Disease Detection Using Machine Learning,” *2023 6th International Conference on Information Systems and Computer Networks (ISCON)*, Mathura, India, 2023, pp. 1–5.

[12] Aishwarya, N., Praveena, N. G., Priyanka, S., et al., ”Smart Farming for Detection and Identification of Tomato Plant Diseases using Light-Weight Deep Neural Network,” *Multimedia Tools and Applications*, vol. 82, pp. 18799–18810, 2023.