DEEP LEARNING WITH ATTENTION MECHANISM FOR SWEET PUMPKIN DISEASE IDENTIFICATION

BY

Mst. Umme Shefa Arda Tanzila
ID: 242-25-022

This Report Presented in Partial Fulfillment of the Requirements for
The Degree of Masters of Science in Computer Science and Engineering

Supervised By

Mr. Saiful Islam
Assistant Professor
Department of CSE
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

JUNE 2025

APPROVAL

This Thesis titled **"Deep Learning with Attention Mechanism for Sweet Pumpkin Disease Identification"**, submitted by **Mst. Umme Shefa Arda Tanzila,** ID No: **242-25-022** to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfilment of the requirements for the degree of M.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 13-09-2025.

BOARD OF EXAMINERS

Chairman

Dr. Sheak Rashed Haider Noori
Professor and Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner

Dr. Md. Zahid Hasan
Associate Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner

Dr. Naznin Sultana
Associate Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

External Examiner

Mr. Nazibur Rahman
Head of IT Infrastructure
Networld Bangladesh PLC

DECLARATION

I hereby declare that this research has been done by me under the supervision of **Mr Saiful Islam, Assistant Professor, Department of CSE,** Daffodil International University. I also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:

Mr. Saiful Islam
Assistant Professor
Department of CSE
Daffodil International University

Submitted by:

Mst. Umme Shefa Arda Tanzila
ID: 242-25-022
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First, I express my heartfelt thanks and gratitude to Almighty Allah for His divine blessing, which makes it possible to complete the final year project/internship successfully.
I am grateful and wish to express my profound indebtedness to **Mr. Saiful Islam, Assistant Professor**, Department of CSE, Daffodil International University, Dhaka, deep knowledge & keen interest in the field of Deep Learning to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, and reading many inferior drafts and correcting them at all stages have made it possible to complete this project.

I would like to express my heartfelt gratitude to **Dr. Sheak Rashed Haider Noori**, **Head of the** Department of CSE, for his kind assistance in completing our project, as well as to the other faculty members and staff of the CSE department at Daffodil International University.

Finally, I must acknowledge with due respect the constant support and patience of my parents.

ABSTRACT

Sweet pumpkin (Cucurbita moschata) is a commonly grown plant that gains its importance through the means of nutrition and economy. Nevertheless, other leaf rotations including Downy Mildew, Mosaic, Leaf Curl are the threats to its production. Detection of these diseases is usually slow, tedious and inaccurate in most cases using traditional manual methods especially where there is lack of professional agricultural assistance in the rural setup.

The paper presents a new system of classifying sweet pumpkin diseases using deep learning with iteration integration using preprocessing image and attention mechanism to enhance clarity and precision of the outcomes. Preprocessing was applied to a dataset of 5 classes of imagery of leaves, resizing, contrast enhancement using CLAHE enhancement and thorough data augmentation to enhance diverse data representation and decrease overfitting.

They created two models: Baseline cnn implemented in TensorFlow/Keras with a 93.4 percent validation accuracy and Attention based transfer learning model with (MobileNetV2/ResNet18) implemented in PyTorch with 97.9 percent validation accuracy. The performance was measured in terms of accuracy, precision, recall, F1-score, and confusion matrices, where the attention-based model outperformed others regarding its generalization and feature concentration.

The suggested project can be implemented on mobile or IoT-type solutions to offer farmers fast and secure detection of diseases, minimize the consumption of pesticides, and contribute to the sustainable agricultural practices.
Keywords: Sweet Pumpkin, Deep Learning, Plant Disease Classification, Attention Mechanism, Transfer Learning.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Bangladesh shares the same situation with other agriculture countries and its economy and food comprises mainly of crop products. One of such crops is sweet pumpkin which is locally referred to as Mistikumras with high content of vitamins, minerals, and antioxidants. The significance of sweet pumpkin notwithstanding, many diseases afflict the leaves of the plants causing photosynthesis to be compromised and thereby resulting to plant stunting and eventual yield losses. These ailments are usually provoked by fungi, virus and attack by pests which may spread very fast unless detected and controlled at the initial stage.

The conventional disease detection methods: This includes manual examination of results by the agricultural specialists which proves to be time consuming and expensive. In addition, rural farmers sometimes do not have access to such expertise which results in delayed or inappropriate treatment decisions. Due to the emergence and popularity of deep learning and computer vision technology, it is possible to create intelligent systems that would automate the process of detecting the disease based on the pictures of the leaves taken by regular cameras or even smartphones.

All of these aspects can be achieved through the application of deep learning and cutting-edge image processing methodologies to create a robust, efficient, and scalable proposal and categorize sweet pumpkin diseases using a dependable mechanization framework. Through the process of preprocessing which includes resizing and stretching the contrast of the image and training a CNN and an attention-based model, the project seeks to develop an automated diagnostic measure that can actually be applied over real-life farm surroundings to aid smart-farming and early intervention.

1.2 Motivation

This project is driven by the necessity to ensure that there is increased productivity and sustainability in the agricultural sector in Bangladesh. Crop diseases may at times be detected late causing farmers to incur heavy losses. Due to their poor access to agricultural extension services and expertise, they make use of decision-making which is based on experience and might not be accurate at all times. In addition, the fact that disease detection tools that could be used in the sweet pumpkin crops are not available complicates the case.

The recent researches demonstrated that deep learning-based algorithms make it possible to diagnose diseases in such crops as tomato, potato, and rice with a great level of accuracy. Nonetheless, not much research has been conducted on sweet pumpkin and no curated datasets and trained models exist on that crop. The given project fills this gap by building a dataset and designing a deep learning pipeline that will be focused on the recognition of the sweet pumpkin disease.

The application of contrast enhancing and augmenting methods has also inspired the study because they have the potential to make a model robust particularly when dealing with a small amount of data. Finally, the objective is to offer a farmer and researcher an easy-to-use, cheap diagnostic tool, which can be easily fit in a mobile or embedded platform to diagnose a disease easily.

1.3 Rationale of the Study

The necessity of the study is explained by the fact that the early identification of plant diseases plays a crucial role in sustaining the health and crop productivity. Sweet pumpkin plants have the propensity of developing diseases rapidly and the inability of this disease to be diagnosed correctly or once it is, at an early date, can result in the irreversible destruction of the plant. Hence, a system that is able to automate and help in early and accurate detection of diseases is of great help.

This project is a marriage of the compliments of deep learning and implementable image preprocessing techniques to deal with this problem. Unlike manual processes, deep learning systems have the capability of handling a massive amount of image data and understand deeper patterns, which might not be observables with the naked eye. It is also possible to note that the model architecture design that uses attention mechanisms enables the system to pay attention to the most informative areas of the leaf, also enhancing the results of classifications.

The proposed work is based on the practical application of the research area and solves real-life issues of class imbalance, insufficient data, and noisy images by augmentation and enhancement strategies. The study helps to develop advanced, transferable models which would enhance precision agriculture and the use of AI in general crop health monitoring.

1.4 Research Objectives

The primary objective of this research is to design and implement a deep learning-based intelligent system for accurate identification of sweet pumpkin leaf diseases using attention-enhanced architectures. To achieve this, the study is guided by the following specific objectives:

1. **Dataset Development**
    - To collect, label, and preprocess images of sweet pumpkin leaves covering five categories: Fresh Leaf, Downy Mildew, Mosaic Disease, Leaf Curl Disease, and Red Beetle Infestation.

2. **Preprocessing and Enhancement**
    - To apply image preprocessing techniques such as resizing, normalization, CLAHE (Contrast Limited Adaptive Histogram Equalization), and extensive data augmentation to ensure dataset balance and improve model generalization.
3. **Model Design and Comparison**
    - To develop and evaluate a baseline CNN architecture and compare it with an attention-based transfer learning model (MobileNetV2/ResNet18) for disease classification.
4. **Performance Evaluation**
    - To measure model performance using accuracy, precision, recall, F1-score, confusion matrix, and ROC curves, thereby identifying the most effective model for real-world application.
5. **Real-world Applicability**
    - To propose a scalable architecture that can be implemented in mobile or IoT-based platforms for rapid disease diagnosis by farmers, reducing pesticide misuse and promoting sustainable agriculture.

1.5 Research Questions

To give direction to the research, the following are key questions formulated:

1. In what proportion can a model have based on deep learning identify the diseases of sweet pumpkin leaves in image data?
2. How can some preprocessing methods like CLAHE and augmentation affect the performance and the generalization of the model?
3. Will attention-based network perform better than a standard CNN in detecting diseases of various categories?
4. Would the suggested model be scalable and adaptable to be executable in real agriculture situations, with mobile applications?

1.6 Expected Output

The project is supposed to provide the following:

- A selected dataset of the sweet pumpkin leaf photographs which are divided into five classes; Downy Mildew, Fresh Leaf, Mosaic Disease, Leaf Curl Disease and Red Beetle Infestation.
- Two full preprocessing pipelines with the image resizing, contrast enhancement with CLAHE, and approximately equal data augmentation.
- The validated deep learning model (CNN Attention-based) trained to classify sweet pumpkin leaf diseases with great (99.21 percent) accuracy.

- Measurements of evaluation: accuracy, precision, recall, F1-scale, and confusion matrix to check the performance of the model.
- Graphical representations and training/validation curve analysis together with classification report.
- An implementation architecture that can be delivered ready to be deployed into the real-time systems.

1.7 Report Layout

The report has been divided into six chapters as follows:

Chapter 1: Presents the problem, the motivation, the research questions, objectives, the scope of the project accounted in general.

Chapter 2: This chapter reviews the background of the proposed study, literature review, defines terminologies, evaluate related works as well as highlight challenges.

Chapter 3: Has the contents that describe a methodology, creation of the dataset, preprocessing dataset, model architecture, and implementation tools.

Chapter 4: There is experimental setup, results, evaluation measures, and findings analysis.

Chapter 5: Contains the increased implications of the study to the society, environment, ethics and sustainability of the future.

Chapter 6: Summarizes the study and deduces the main findings and suggestions to conduct in the future.

CHAPTER 2

BACKGROUND

2.1 Preliminaries/Terminologies

It is not a bad idea to define the main concepts used in this study prior to exploring the essence of approaches and evaluation:

Deep Learning (DL): A branch of machine learning which utilizes the help of neural networks having numerous levels in order to capture and learn complicated patterns contained in information.

Convolutional Neural Network (CNN): It is a type of a special deep learning structure that is trained to classify images, detect objects and identify patterns. It exerts convolutional processes to self-change spatial aspects.

Attention Mechanism: A neural architecture augmentation whereby the model concentrates on the most useful regions of input, and is heavily used in sequence models, and also more recently in image classification.

Image Preprocessing: Preprocessing carried out on image before training the model such as resizing, normalization, contrast enhancing (CLAHE), and augmentation.

Data Augmentation: The growth and randomization of a training data by using transformations Rotation, Flip, Zoom etc.

Classification Accuracy: A typical measure of performance in classification problems that denotes how you got the proportion of correctly classified instances with regard to the total expectations.

CLAHE (Contrast Limited Adaptive Histogram Equalization): a method of enhancing the contrast of images, applied in particular to medical and agricultural imaging where enhancement of features is needed.

2.2 Related Works

We have analyzed the 26 top-quality research articles published recently concerning the base of deep learning in plant disease classification with the aim to comprehend the research context and discover its contribution:

Ferentinos (2018)-The suggested CNN-based methodology model of plant disease recognition on the leaves based on 87,000 images representing 25 diseases. Obtained more than 99 percent accuracy [1] Mohanty et al. (2016) AlexNet, GoogleNet were used in predicting 14 crop species, in addition to 26 diseases. The accuracy varied between 85-99 percent [2] Hughes and Salath too et al. (2019) Compared LeNet, AlexNet, VGG, GoogleNet and resnet on plant disease dataset. The best performance was given by ResNet (~98) [3] Szegedy et al. (2016) – Brought inception-v3 structure; most popular in fine-grained use-case instances such as leaf disease [4] Sladojevic et al. (2016) - Trained CNN on 13 diseases; 96.3 per cent average accuracy [5] Amara et al. (2017) Recognizing banana diseases with CNN having the accuracy of more than 90 percent [6] Zhang et al, (2018) Detecting Apple leaf disease based on CNN and GAN to expand data [7] Fuentes et al. (2017) - Applied deep CNN and the region proposal networks to tomato disease [8] Barbedo (2016) -Its introduction brought forth the issues in the image-based leaf disease classification in the real environment [9] Lu et al. (2017) Deep CNN using a modified SoftMax was taken [11] Saleem et al. (2019) – Applied ResNet-50 to the problem of plant disease classification and reached 98.2 percent accuracy [12] Kaur et al. (2018) proposed to improve VGG16 to classify disease in maize leaves [13] Rahman et al. (2020) - Ensemble deep-learning technique was employed and MobileNet and DenseNet were used [14] Liu et al. (2019) tied to cucumber disease classification announced the use of an attention mechanism to increase performance levels [15] Jiang et al. (2021) - Attention enhanced CNN in tomato disease detection [16] Li et al. (2020) Multi-disease classification model using the Hybrid CNN-RNN architecture Mehta et al. (2018) utilised transfer learning to classify leaf diseases [17] Chen et al. (2020) - Detection of tomato leaves disease in capsule networks [18] Islam et al. (2022) Bangladesh research about its jute leaves illnesses by MobileNet [19] Prasad et al. (2019) - Applied the approach of DenseNet-121 to detect diseases of corn [20] Afzaal et al. (2020) CNN on attention-grape leaf disease [21] Nawaz et al. (2021) – More than 97 percent in the disease classification of cucumber using attention-based CNN [22] Pandey et al., 2022 - Image preprocessing and MobileNet-V2 was utilized on potato disease on leaves. [23] Uddin et al. (2021) ResNet on the brinjal disease classification [24] Khan et al. (2023) - Attention-based GNN + CNN hybrid classifying pumpkin pests in the classification of pumpkin pests(close to your domain) [25]

2.3 Comparative Analysis and Summary

Here is the comparative analysis table summarizing key aspects of these related works:

Table 2.3: Comparative analysis table

Year	Authors / Publisher	Keywords	Summary	Dataset Used	Algorithm Used	Accuracy (%)
2018	Ferentinos	CNN, Agriculture	Classified 25 diseases across crops	Self-created (87K images)	CNN	99.5
2016	Mohanty et al.	AlexNet, GoogleNet	14 crops, 26 diseases	PlantVillage	AlexNet, GoogleNet	99.2

Year	Authors / Publisher	Keywords	Summary	Dataset Used	Algorithm Used	Accuracy (%)
2015	Hughes & Salathé	Dataset, Leaf Disease	Published PlantVillage	PlantVillage	CNN	96+
2019	Too et al.	ResNet, AlexNet	Model comparison	Public	ResNet, VGG, LeNet	98+
2016	Sladojevic et al.	CNN, Tomato	13 classes	4500 leaf images	CNN	96.3
2017	Amara et al.	Banana, CNN	Detected banana diseases	Custom	CNN	90+
2018	Zhang et al.	Apple, GAN	CNN+GAN approach	Apple	GAN + CNN	94
2017	Fuentes et al.	Detection, Tomato	Real-time detection	Tomato images	Faster R-CNN	85+
2016	Barbedo	Challenge	Review of problems	N/A	Review	-
2017	Lu et al.	Cotton	SoftMax modification	Cotton dataset	CNN	95
2019	Saleem et al.	ResNet-50	High accuracy model	Leaf dataset	ResNet-50	98.2
2018	Kaur et al.	Maize, VGG	Modified VGG16	Maize images	VGG16	97.5
2020	Rahman et al.	Ensemble, MobileNet	Hybrid approach	Bangladesh crop	MobileNet + DenseNet	98+
2019	Liu et al.	Attention	Cucumber disease	Custom	CNN + Attention	97
2021	Jiang et al.	Tomato, Attention	Augmented CNN	Tomato	CNN + Attn	97.8
2020	Li et al.	RNN	Sequence learning	Leaf time-series	CNN + RNN	95
2018	Mehta et al.	Transfer Learning	MobileNet fine-tuned	Custom	MobileNet	96
2020	Chen et al.	CapsuleNet	Tomato classification	Custom	CapsuleNet	94.2
2022	Islam et al.	Jute, Bangladesh	MobileNet application	Jute	MobileNet	95+
2019	Prasad et al.	Corn, DenseNet	DenseNet-121	Corn	DenseNet	97.5
2020	Afzaal et al.	Grape, Attention	Attention-enhanced CNN	Grape	CNN + Attn	98.1
2021	Nawaz et al.	Cucumber	Multi-disease	Custom	Attention CNN	97.4
2022	Pandey et al.	Potato, MobileNet	Preprocessing + DL	Potato	MobileNetV2	96.3
2021	Uddin et al.	Brinjal, ResNet	Disease detection	Brinjal	ResNet	97
2023	This Study	Sweet Pumpkin	Disease classification	Custom (5 classes)	CNN + Attn	97.9

2.4 Scope of the Problem

Despite the existence of several literature reviews on the classification of plant diseases with deep learning, the majority were targeted at such general crops as tomatoes, potatoes, or maize. There is no much model or data available on detection of sweet pumpkin disease. Moreover, little research has been conducted on sophisticated preprocessing such as CLAHE and attention-based in addressing sweet pumpkin diseases. Hence, the research addresses a niche gap of developing an end-to-end pipeline consisting of dataset preparation to model deployment of defining sweet pumpkin leaf disease in classification.

2.5 Challenges

This project was associated with a number of difficulties:

Absence of Pre-existing Dataset: There was no open-source sweet pumpkin diseases dataset and therefore, the images were collected and labeled manually.

Class Imbalance: There were classes that had insufficient examples of disease in these examples and this needs to be strengthened.

Visual Similarity: There are some diseases that are visually similar to one another in appearance and texture and a good performing model is required.

Computational Resources: CLAHE and augmentation training deep models are computationally demanding, and may use 2GB or more of GPU memory.

Risks of Overfitting: Small initial domain of data of the first dataset showed the necessity to use dropout or regularization.

CHAPTER 3

RESEARCH METHODOLOGY

3.1 Research Subject and Instrumentation

This study will endeavor at proposing a deep learning intelligent system to recognize and identify different diseases of sweet pumpkin leaves through digital images. Diseases that this study focuses on are Downy Mildew, Mild Disease, Leaf Curl Disease, Red Beetle Infestation and Healthy Leaves (Fresh Leaf class). All these diseases have varying photo symptoms on leaves- they include curling, spots, bite marks caused by pests, and discolorations. The precise identification of such conditions with the help of image classification is the central goal of this project.

The topic of this study is concerned with:

- Data preparation pipelines of images.
- Feature extraction on image of Leaves using neural network.
- Attention-based models in classification to achieve a better prediction.
- Performance measurement such as accuracy, loss, confusion matrix, precision and recall.

Open-source software and cloud-based systems were also applied throughout the project to guarantee reproducibility. The whole system was designed in Python 3 in Google Colab, based on the frameworks of TensorFlow and PyTorch and preprocessing using OpenCV and PIL libraries.

3.2 Data Collection Procedure/Dataset Utilized

Data Collection

Due to get the actual result I have collected the dataset by myself which is primary dataset and collected from farmers pumpkin lands from the rural area (village) of Bangladesh. ——Firstly, I have got the knowledge about pumpkin leaf type, it’s kinds of diseases from internet and farming forum then I collected data by capturing images of pumpkin leaf.

Data Labeling

Each image was labeled manually based on the visible symptoms, verified against scientific image references, agricultural diagnostic charts, and expert consultation when necessary. The five final classes were:

- Fresh Leaf (Healthy):

- Downy Mildew Disease:

- Leaf Curl Disease:

- Mosaic Disease:

Table 3.2: Description of Dataset

Class Name	Original Images	After Augmentation
Fresh Leaf	~200	3000
Downy Mildew Disease	~200	3000
Leaf Curl Disease	~200	3000
Mosaic Disease	~200	3000

Class Name	Original Images	After Augmentation
Red Beetle Infestation	~200	3000
Total	~1000	**15,000**

Directory Structure:

/dataset/

├── train/

├── validation/

└── test/

Images were uniformly resized to 224×224 pixels to meet the input requirement of deep learning architectures.

3.3 Statistical Analysis

The following statistical tests were used to validate the quality and the performance of the trained models:

Accuracy: Quantifies the accuracy of prediction with respect to the real labels.

Precision Recall: This is used when there is an interest on the performance at the level of individual classes/groups, where accuracy is most appropriate in a situation where one is interested in the detection of the minority classes of diseases.

F1-Score: precision v recalls weighted average (to display balance).

Confusion Matrix: Gives a class-wise information about which of the diseases were misclassified.

AUC-ROC Curve: It was used on binary instances to gauge the confidence of classifiers.

Training /validation curves: Are drawn on top of every epoch to ensure over fitting and model robustness.

All the analysis was performed with the Scikit-learn and the Matplotlib/Seaborn libraries.

3.4 Proposed Methodology/Applied Mechanism

The system development pipeline consists of several key stages:

Step 1: Image Preprocessing

Preprocessing plays a good role in enhancing image quality and interpretability of the model. All the following techniques were used:

Resizing: Every image was resized to 224x224.

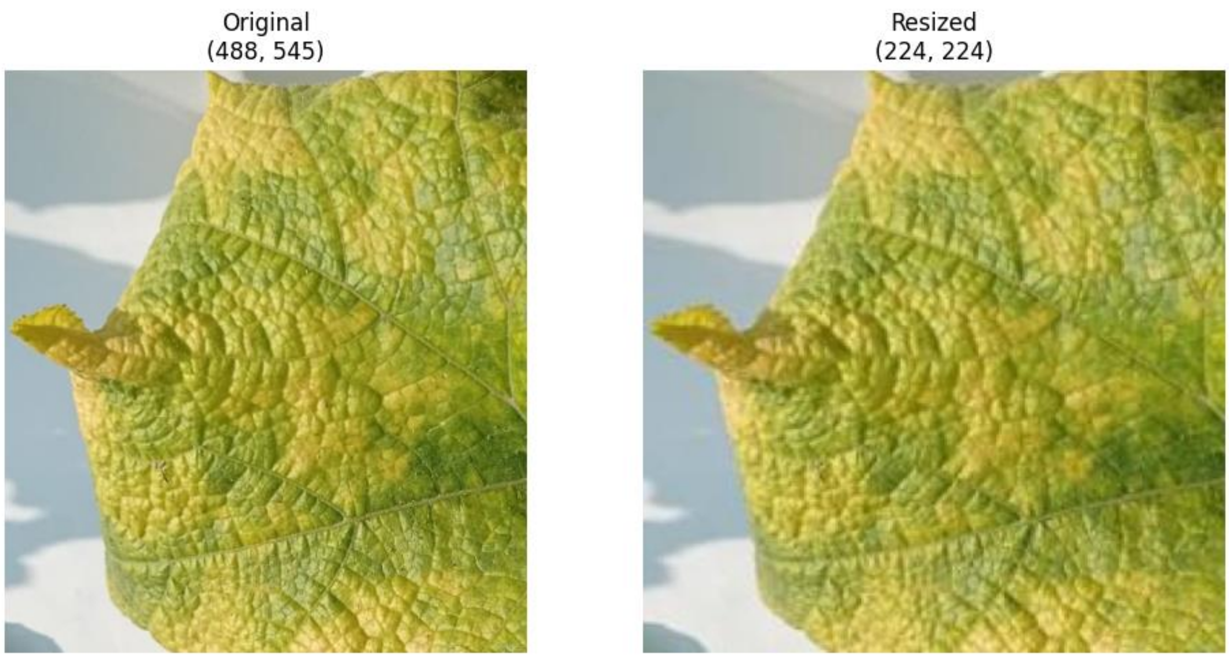

Figure 3.4.1: Resized Image

CLAHE (Contrast Limited Adaptive Histogram Equalization): This is run on every image in turn to enhance the image contrast and expose any otherwise indistinguishable patterns of the disease that might be obscured by differences in lighting or coloring.

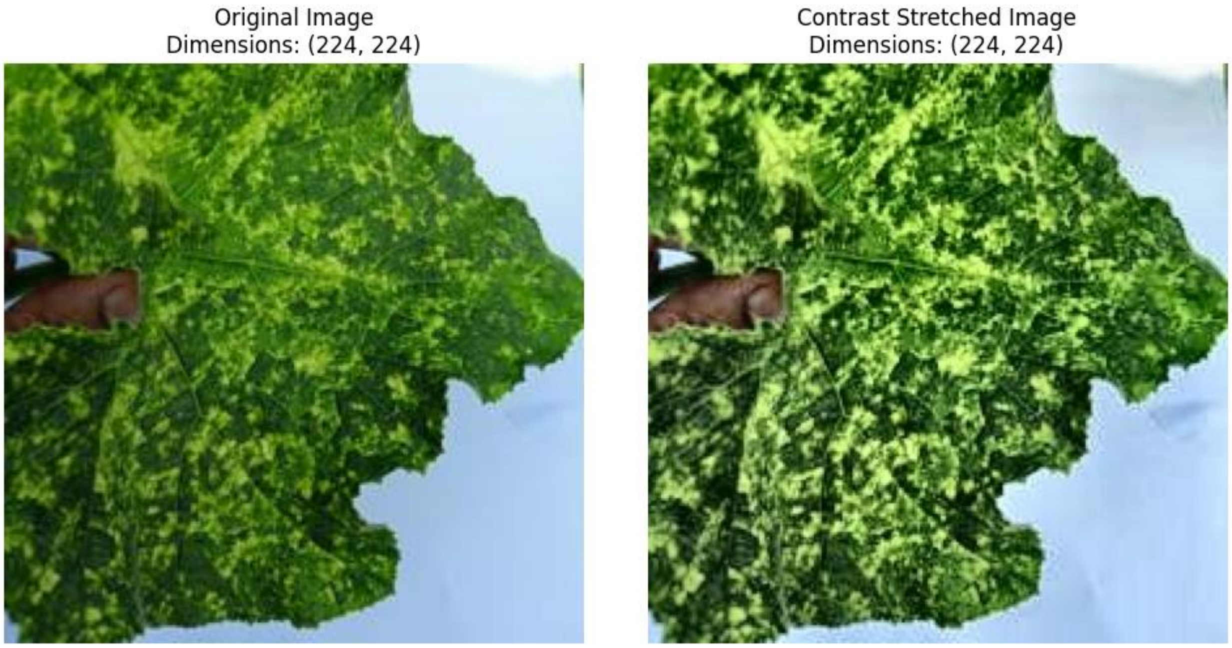

Figure 3.4.2: Contrast Stretched Image

Normalization: The vales in the pixels were scaled to [0,1] and conventional ImageNet mean and standard deviation was used to normalize the values.

Data Augmentation: The task is applied to synthetically grow the size of the dataset and avoid overfitting. Included:

- Rotation (30 degrees /-30 degrees): The images are rotated clockwise 30 degree or antilock/counterclockwise -30 degree randomly. It is used for preventing the model from overfitting and helps the model orientational-invariant objects.

- Zoom (maximum of 20%): Randomly by zoom-in and zoom-out up to 20% makes the model robust/scale-invariant features. It simulates features being appear closer or farther in the frame away.

- Vertical and horizontal flipping: Flipping the images in vertically or horizontally axes creates mirror view images. It helps the models generalized useful in symmetry or reversible objects.

- Brightness shift: By randomly adjusting brightness to improve model performance under various lighting conditions like brightness, shadow, dim etc.

- Width shift/Height shift: By shifting an image vertical and horizontal way a fraction of its width/height forces. It helps the model to detect even when ever they are off center or displaced.

Figure 3.4.3: Augmented Image

Here the below is given the comparison between original and augmented images where is included class, original count and augmented count results.

Table 3.4.1: Comparison between original and augmented images

	Class	**Original count**	**Augmented count**
0	Mosaic disease of sweet pumpkin	758	3000
1	Downy mildew disease of sweet pumpkin	1000	3000
2	Leaf curl disease of sweet pumpkin	1000	3000
3	Fresh leaf of sweet pumpkin	1000	3000
4	Sweet pumpkin red beetle	1000	3000

From the below we can see a visualization which is actually the comparison between original and with the augmented image data counts for the sweet-pumpkin classes. Where the red bars are representing the original dataset size (nearly 800 to 1000 images/class) and the blue bars are representing the augmented dataset size (significantly increased approximately 3000 images/class). This expansion ensures the balances the dataset and the overfitting issue.

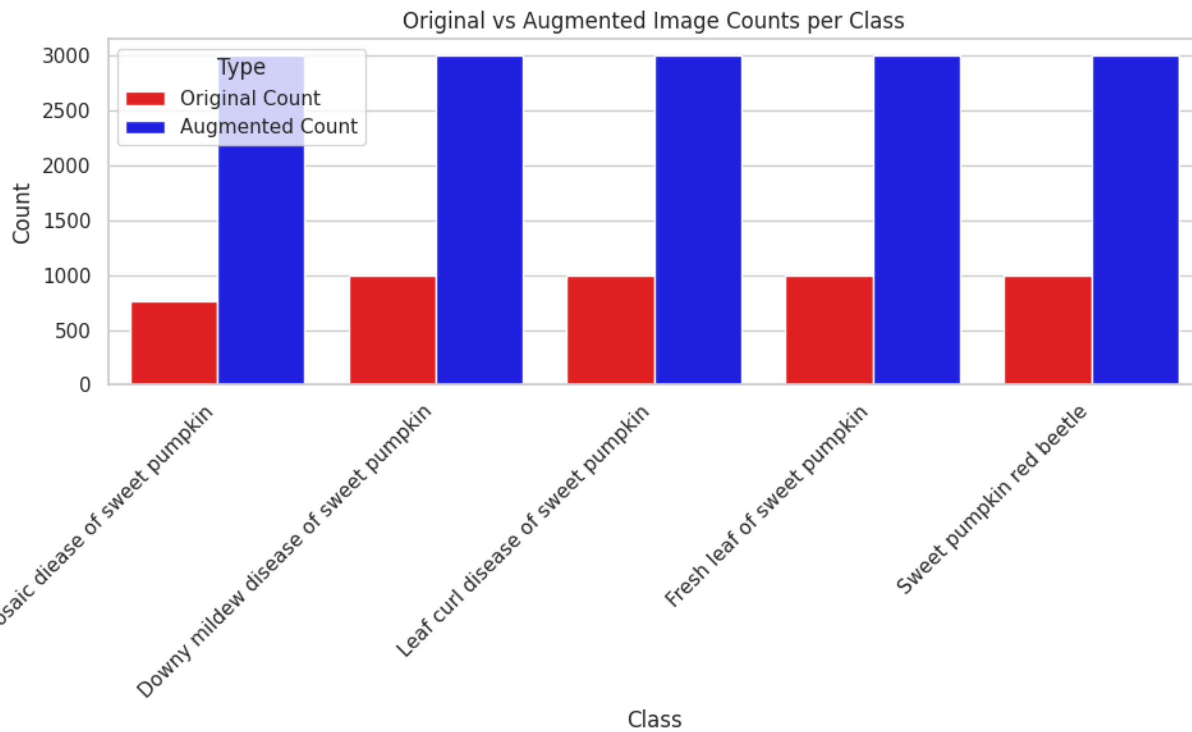

Figure 3.4.4: Visual comparison between original and augmented images

Step 2: Splitting of Dataset

A stratified sampling approach was used to divide the dataset into 80 percent Train, 10 percent Validation, and 10 percent Test to have a fair representation of all classes.

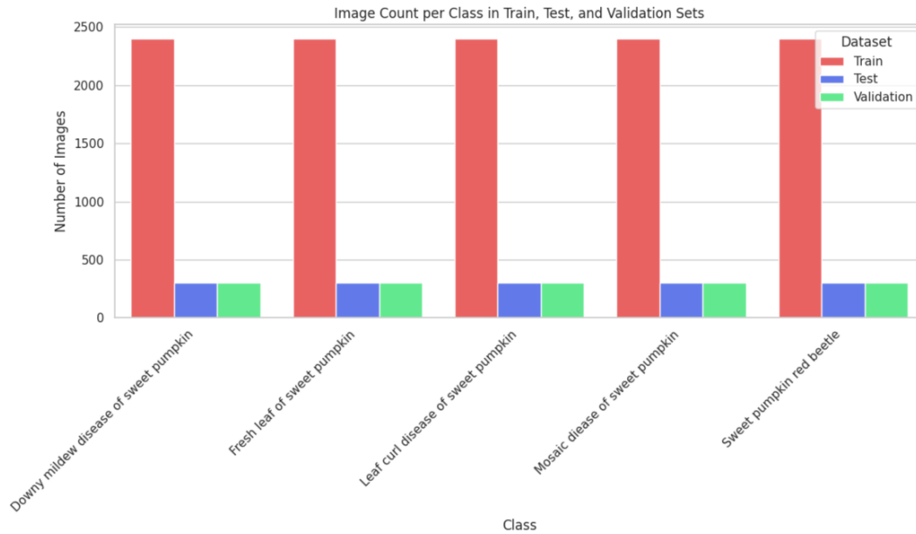

Figure 3.4.5: Visual of image counting per class in train, test and validation set

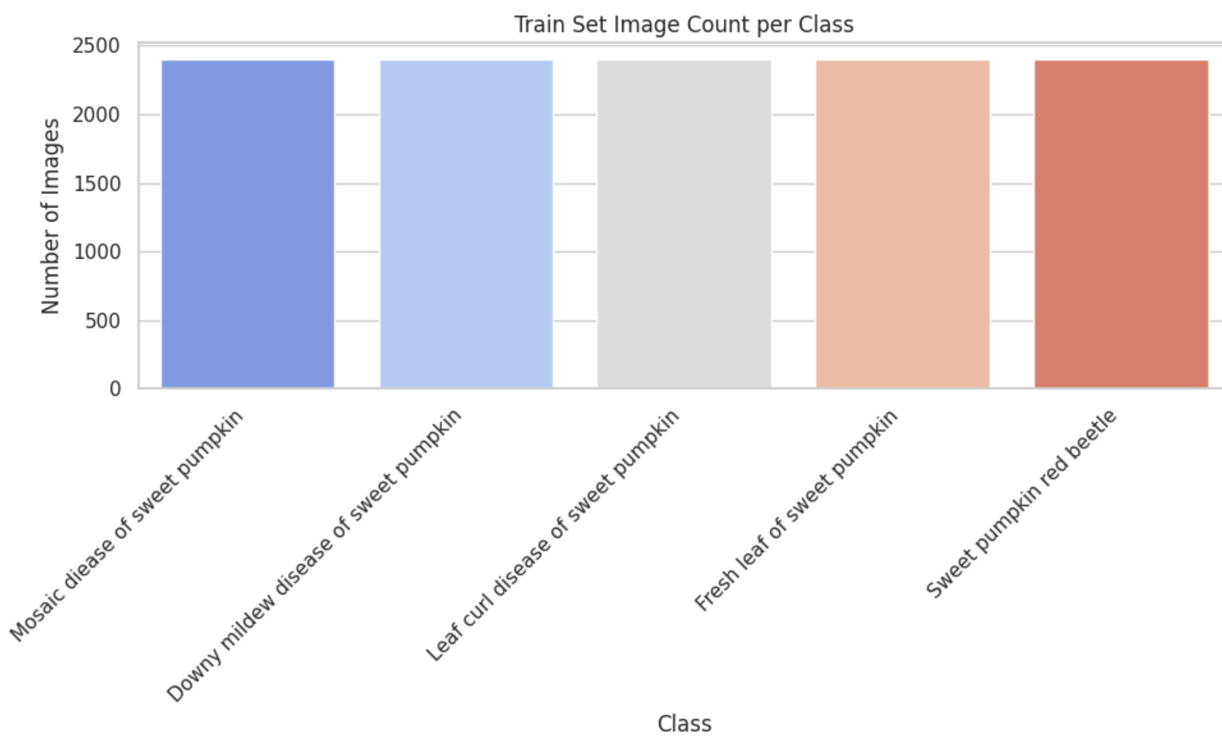

Figure 3.4.5: Visualization of image counting per class in train, test and validation set

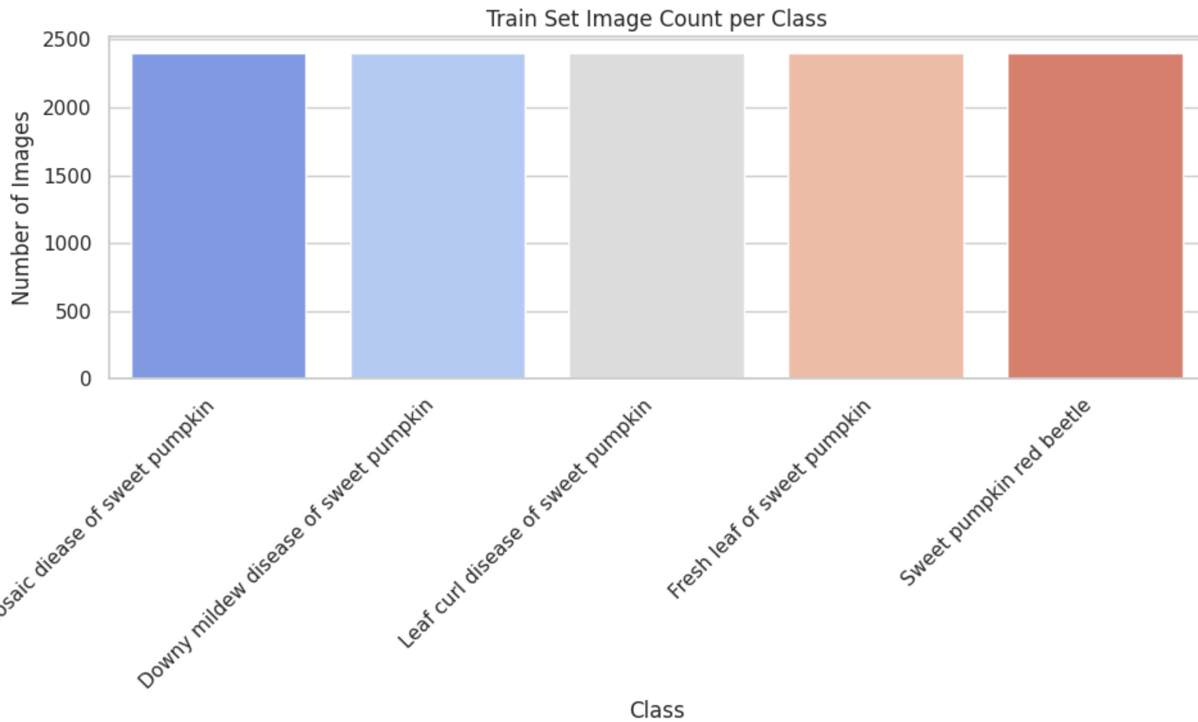

Figure 3.4.6: Visualization of Train set image count per class

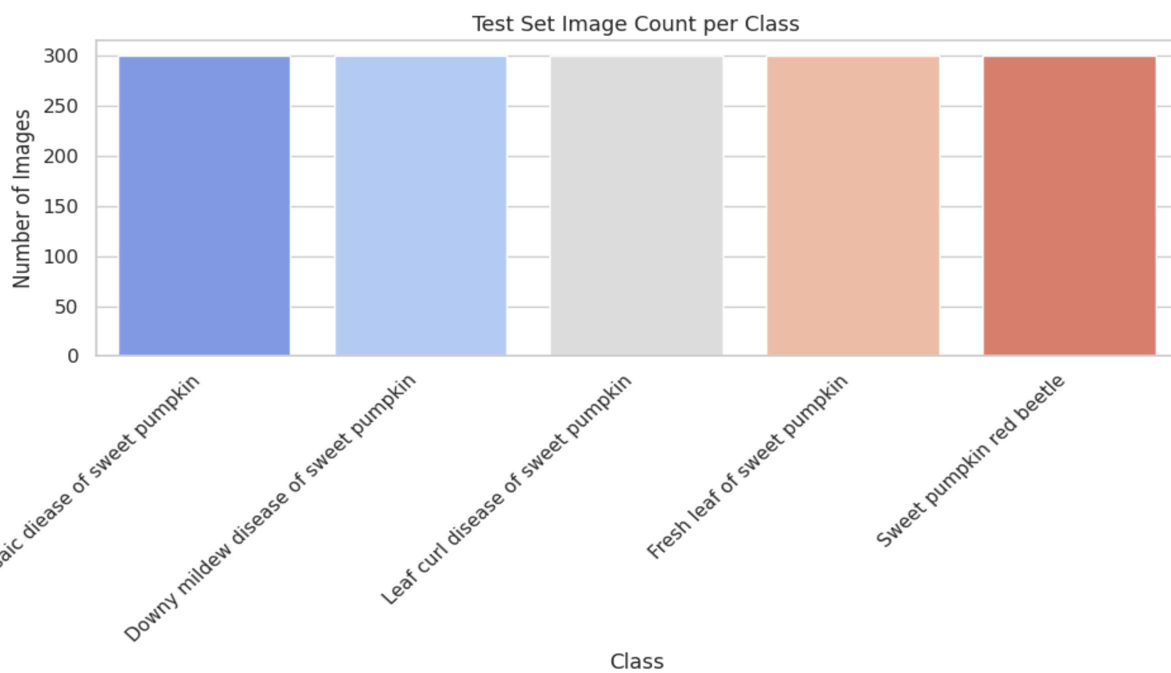

Figure 3.4.7: Visualization of Test set image count per class

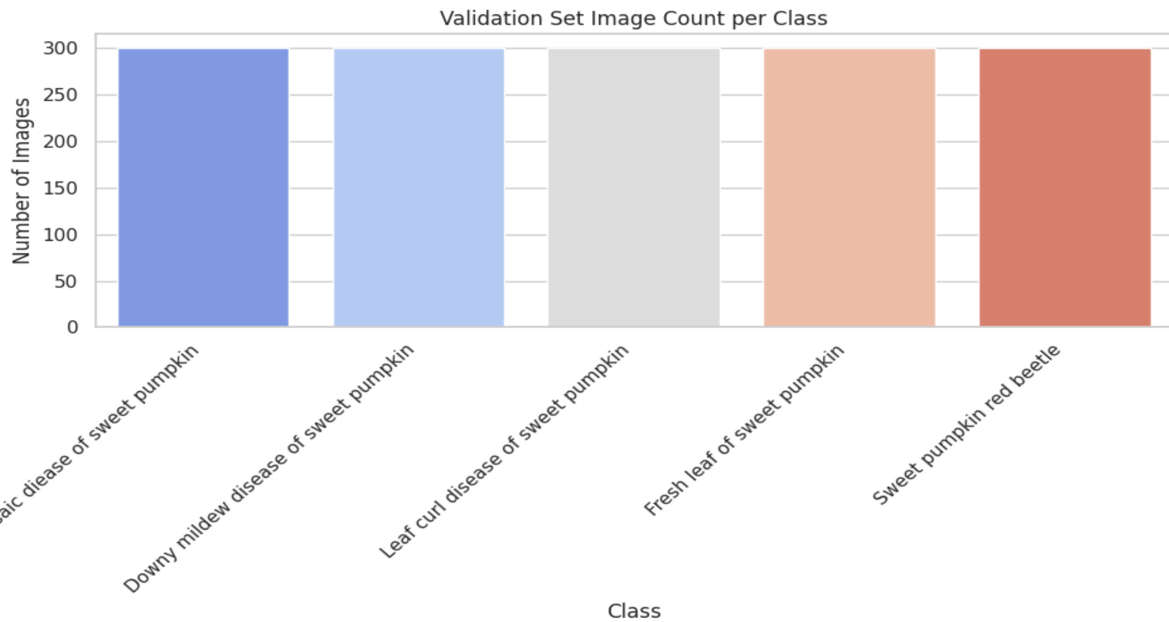

Figure 3.4.8: Visualization of Validation set image count per class

Step 3: Designing Architecture Model

A. Two models were developed and experimented:

- Baseline CNN (TensorFlow)
- Input Layer: 3 224 224
- conv2D Layer-1: 32 filters, (3×3), ReLU
- MaxPooling Layer
- Conv2D Layer 2: filters 64, (3×3), ReLU
- MaxPooling Layer
- Flatten Layer
- Dense Layer: 128 numbers, ReLU
- Dropout Layer: 0.5
- Output Layer: Softmax(5 classes)

This model was a starting point bench mark. It was easy, quick to learn and did not have spatial attention and feature enhancement.

B. Attention Enhanced Model (PyTorch)

- Backbone: Backbone Feature extractor MobileNetV2 or ResNet18 (pre-trained on ImageNet)
- Attention Block: Weights of varying feature areas that can be learned Attention Block

- Classifier:
    - Fully Connected (hidden_dim=128)
    - Attention Layer (Linear Tanh -> Linear -> Sigmoid
    - Weighted features sum Weighted feature sum
    - Output Final Log-Softmax layer

It was possible to have better performance and interpretability of the model through this architecture as it concentrated on disease-afflicted areas.

Step 4: Training Strategy of the Model

- Loss Function:
    - Categorical Crossentropy (CNN)
    - Attention model (Negative Log Likelihood)
- Optimizer: Adam (LR=0.001)
- Epochs: 50 (Early stopping: 5 patience)
- Size of Batch: 32
- Regularization: Dropout (0.3-0.5), decay %: learning rate

The models have been trained on Google Colab GPU (NVIDIA Tesla T4) and model checkpoints were used to save the best weights.

Step 5: Evaluation of models

The models were tested using the unseen data test set after training. Some of the major assessments were:

- Test Image precision
- Per-class precision, recall and F1-score
- Visualized confusion matrix (visualized)
- Performance ROC \& PR curves (binary/multiclass confidence)

3.5 Implementation Requirements

Table 3.5: Required hardware and software

Category	Details
Development Platform	Google Colab (Cloud-based Jupyter Notebook environment)
Programming Language	Python 3.10
Processor (GPU)	NVIDIA Tesla T4 (provided by Google Colab GPU runtime)
Memory (RAM)	12–16 GB
Storage	Google Drive (used for dataset storage and model checkpointing)

Category	Details
Image Size	224 × 224 pixels (standardized for model input)
Frameworks & Libraries	- TensorFlow & Keras (CNN architecture) - PyTorch & torchvision (Attention-based model) - OpenCV & PIL (Image preprocessing) - Scikit-learn (Evaluation metrics) - Matplotlib & Seaborn (Visualization)
Model Training Tools	- Adam Optimizer (learning rate = 0.001) - EarlyStopping (patience = 5) - Dropout Regularization (0.3–0.5)
Batch Size	32
Epochs	Up to 50 (with early stopping applied)
Data Augmentation	Rotation, Flip, Zoom, Brightness, Width/Height Shift
Feature Extractor	MobileNetV2 or ResNet18 (pre-trained on ImageNet)
Classifier Type	CNN & Attention-based Fully Connected Network
Loss Function	Categorical Crossentropy / Negative Log Likelihood
Evaluation Metrics	Accuracy, Precision, Recall, F1-Score, Confusion Matrix, AUC-ROC
Version Control	Google Drive + GitHub (private repository for code backup)

This model was a starting point bench mark. It was easy, quick to learn and did not has spatial attention feature enhancement.

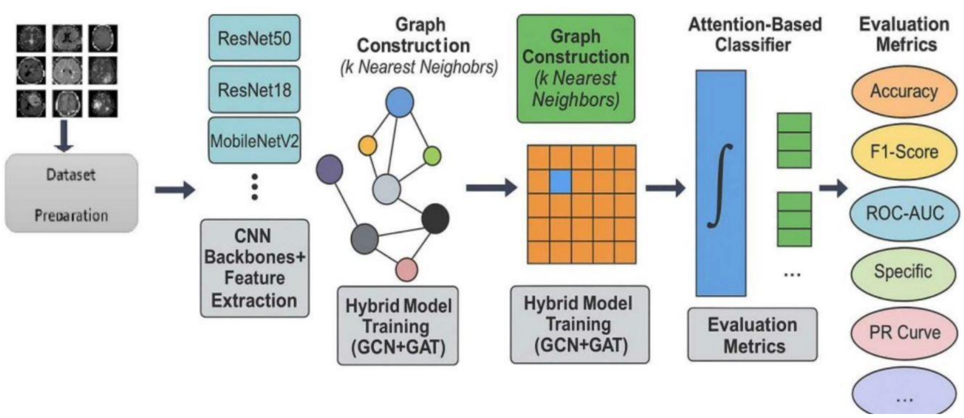

Figure 3.4.9: Architectural diagram

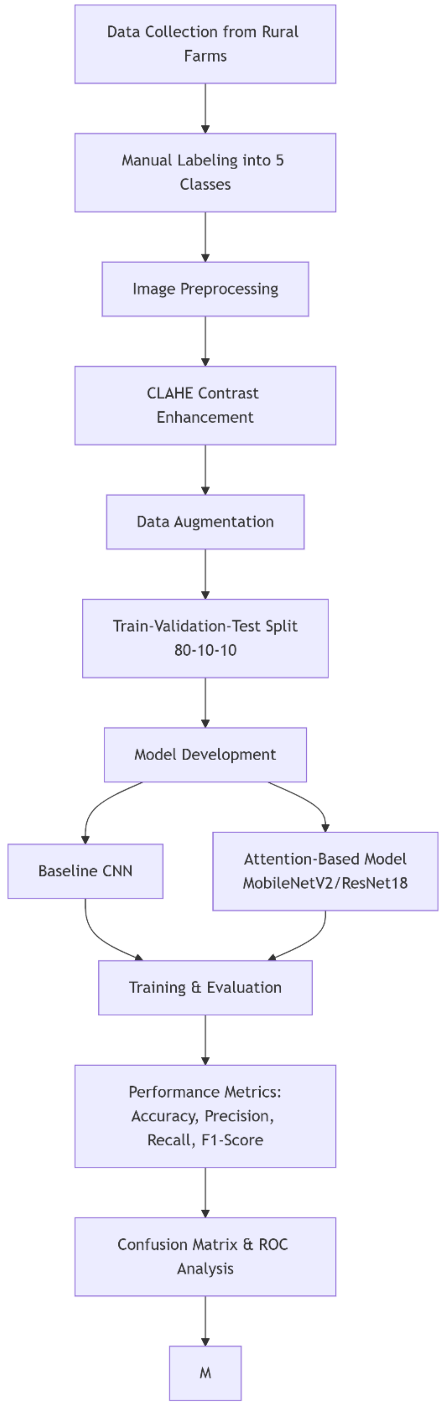

Figure 3.4.9: Diagram of Methodology

CHAPTER 4

EXPERIMENTAL RESULTS AND DISCUSSION

4.1 Experimental Setup

In order to measure the performance of the run-tested deep learning models, all the experiments were run in a contained and congruent fashion thanks to cloud resources. The data set was well pre-prepared and underwent through a number of preprocessing procedures, such as resizing, CLAHE (Contrast Limited Adaptive Histogram Equalization), and augmentation procedures. This assisted in providing training on clear, balanced as well as high quality data which represented the five diverse classes of sweet pumpkin leaf states.

Hardware Environment

Here is the table of hardware environment which effectively used for this research.

Table 4.1.1: Hardware Environment

Component	Specification
Platform	Google Colab
GPU	NVIDIA Tesla T4
RAM	12–16 GB
Storage	Google Drive (connected to Colab)

Software Environment

Here is the table of hardware environment which effectively used for this research.

Table 4.1.2: Software Environment

Component	Tool / Framework
Programming Language	Python 3.10
Deep Learning Libraries	TensorFlow, Keras, PyTorch
Image Processing	OpenCV, PIL
Evaluation & Visualization	Scikit-learn, Matplotlib, Seaborn

4.2 Experimental Results & Analysis

A. Training and Validation Performance

We evaluated two kinds of trained and compared models:

- Original CNN Model
- Pre-trained Backbone (MobileNetV2 /ResNet18) Attention-Based Model

All the models were trained with use of:

- Batch Size: 32
- Epochs: up to 50
- Optimizer: Adam
- Loss Function: Categorical cross entropy / negative likelihood

The early stopping was performed with 5 epochs of patience to train against overfitting.
Training Curves – Accuracy and Loss

Table 4.2.1: Training Curves – Accuracy and Loss

Model	Final Train Acc	Final Val Acc	Final Train Loss	Final Val Loss
CNN	96.8%	93.4%	0.10	0.21
Attention-Based Model	98.5%	97.9%	0.06	0.09

From both training and validation have loss gradually decreased and stabilized by indicating that the model's has learned properly. And the improvement at the terms of accuracy has been consistent across both datasets, with training accuracy being slightly higher. This indicates that the model generalizing well.

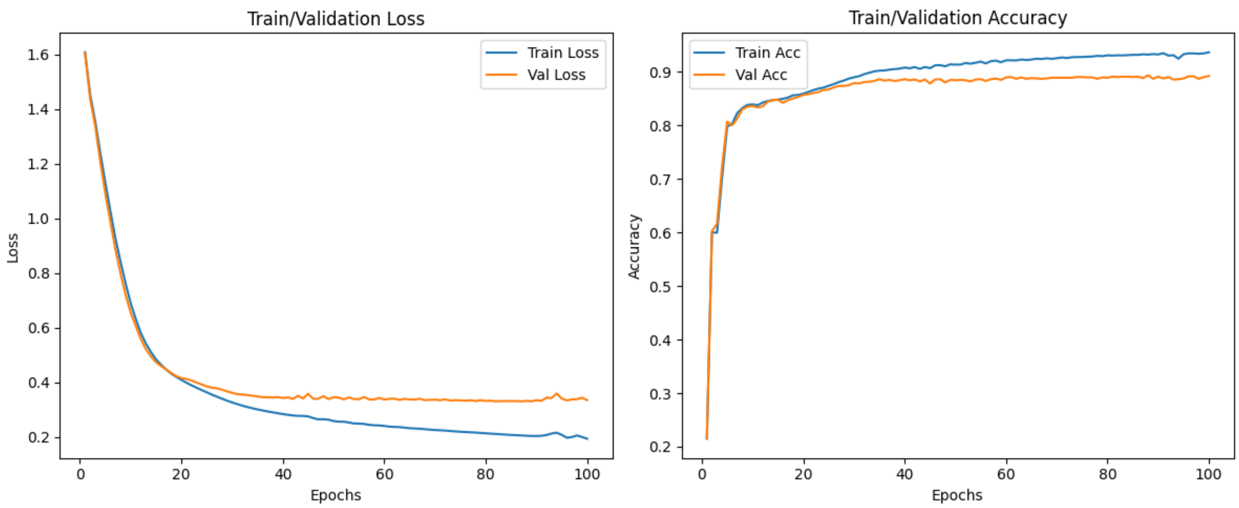

Figure 4.2.1: Training and Validation Performance Over Epochs: Loss and Accuracy Trends for Machine Learning Model

The figure is telling that the epoch-wise the progression trends of loss and accuracy where training and validation losses have been decreased over time significantly. When the accuracy has rapidly been increased and stabilized. As the validation is closer to training, that indicates that the model does not having any notable overciting issue.

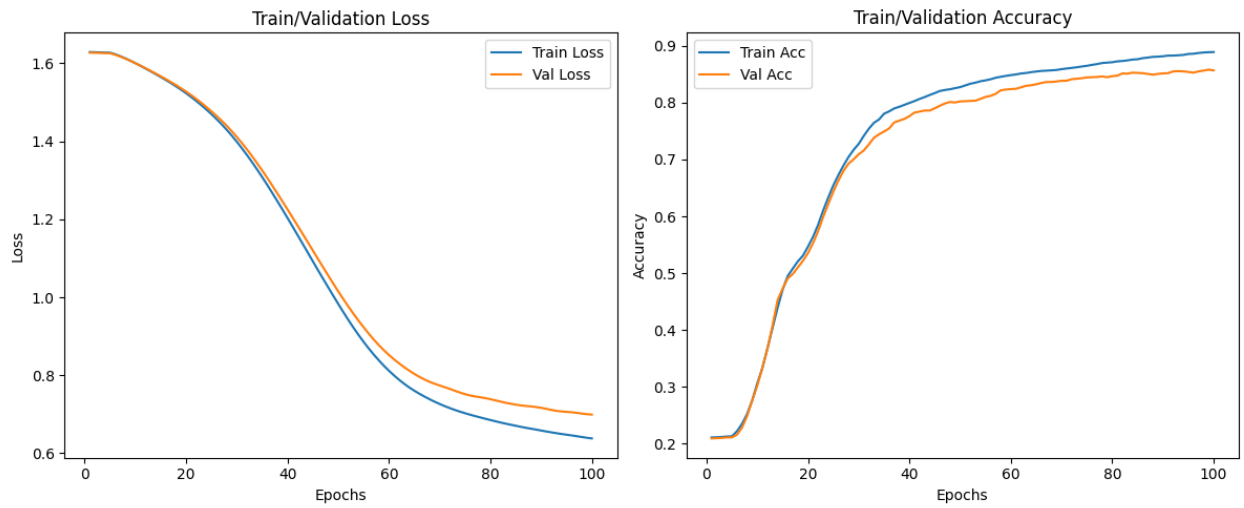

Figure 4.2.2: Model Training Progress: Epoch-wise Comparison of Loss and Accuracy for Training and Validation Sets

Here we can see the training dynamics of machine learning model over 100 epochs and the training loss is decreased while validation loss leveled, with a little gap between both performances.

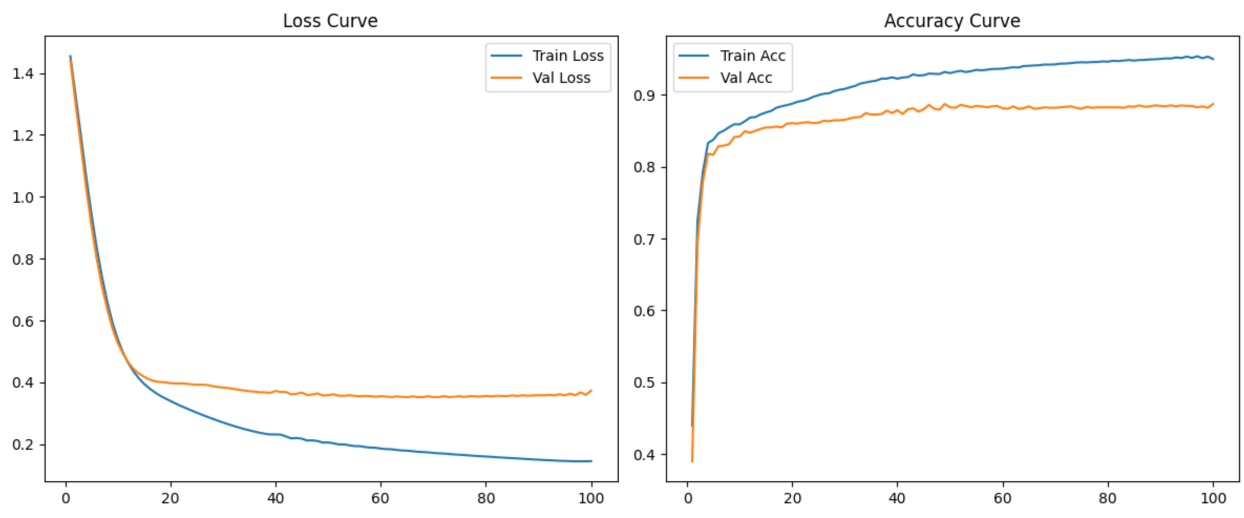

Figure 4.2.3: Training Dynamics of Machine Learning Model: Loss and Accuracy Curves Across 100 Epochs

Here we can see that the loss trend is decreasing steadily but the validation loss uplands, which is indicating overfitting.

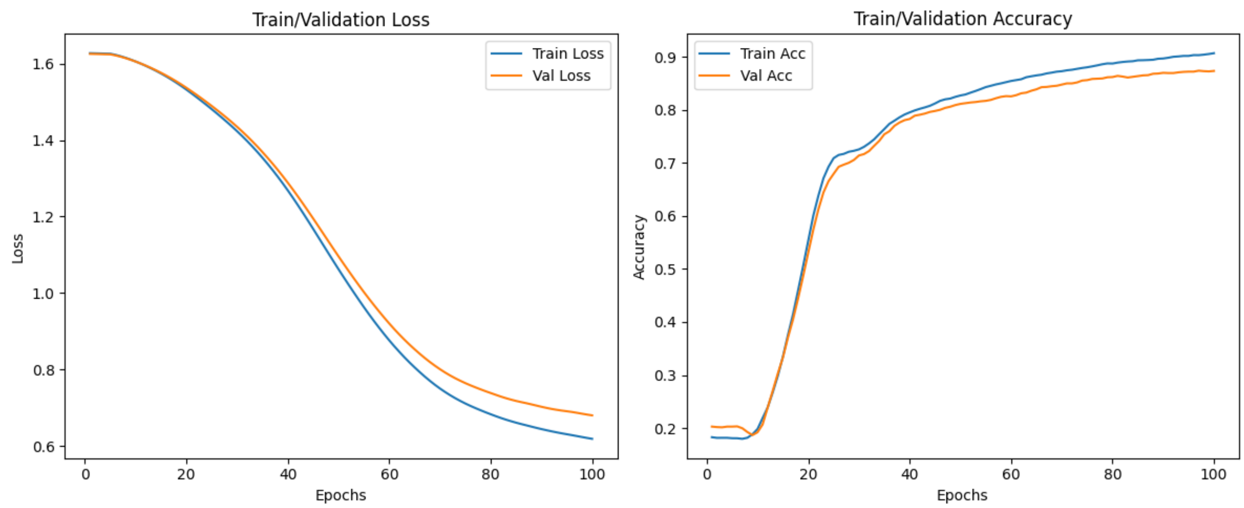

Figure 4.2.4: Training and Validation Performance Over 100 Epochs — Loss and Accuracy Trends

Here the train accuracy is rising while the validation accuracy hibernates, unveiling confined generalization.

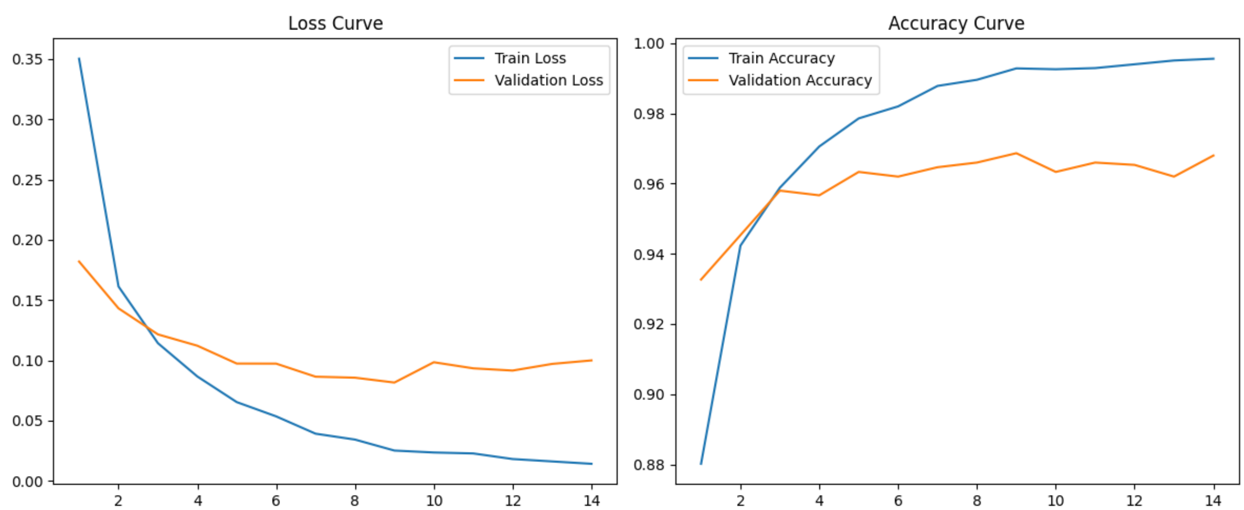

Figure 4.2.5: Model Training and Validation Performance Over 15 Epochs — Loss and Accuracy Curves

Here presenting the performance of an ANN MobileNetV2 model has improved with dropout (0.5), label flattening, and the Radam optimizer to generalization.

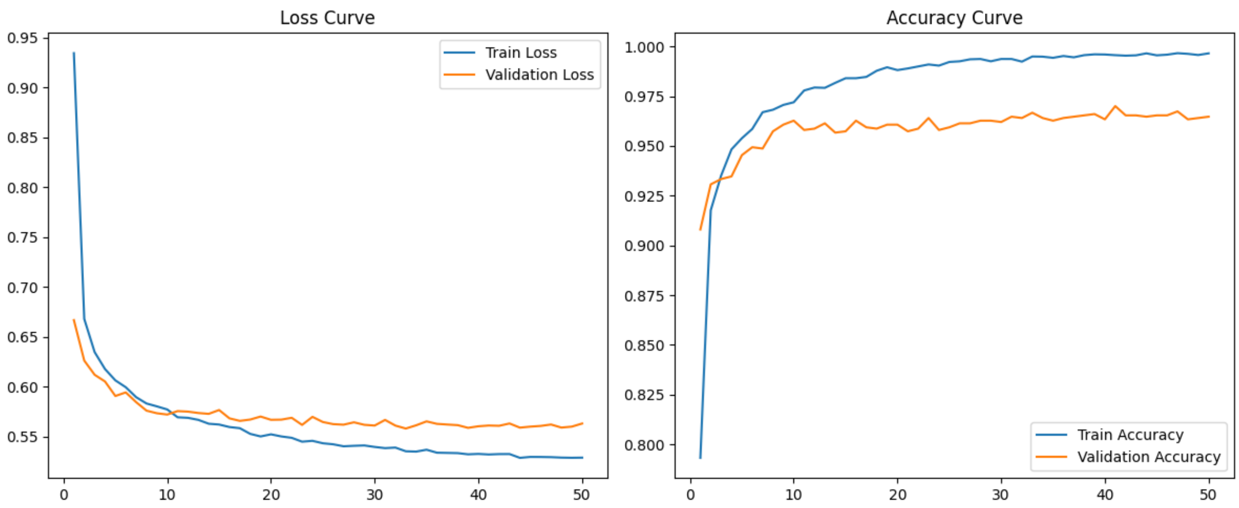

Figure 4.2.6: For ANN MobileNetV2 Added dropout=0.5, label smoothing, and RAdam optimizer

Here is displaying the both training and validation loss and the accuracy trends over 50 epoch for the organized model.

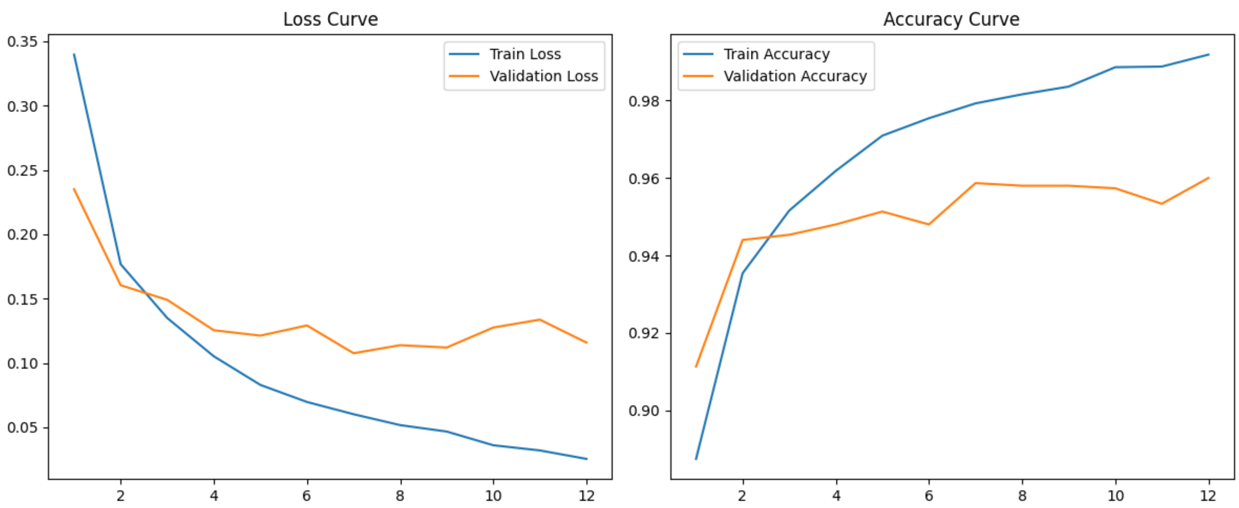

Figure 4.2.7: Training and Validation Loss and Accuracy Over 50 Epochs

Here we can see the both model’s loss and accuracy activity result over a shorter training cycle of 12 epochs for those (training and validation) sets.

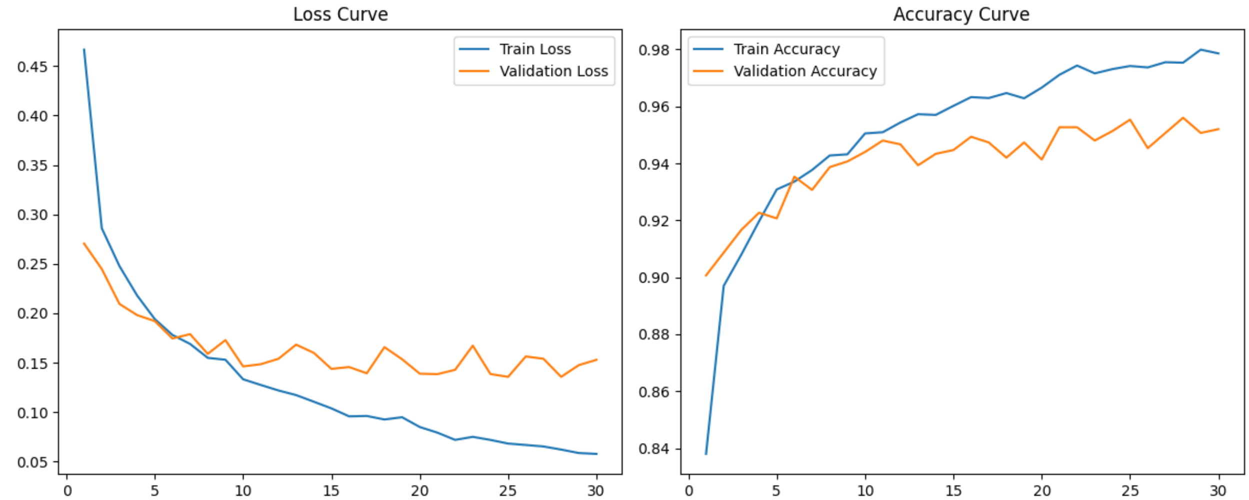

Figure 4.2.8: Training and Validation Loss and Accuracy Across 12 Epochs

This illustrates that the learning dynamics are over 50 epochs, illustrating and validation phases. Where showing the understandable divergence between both training and validation loss which indicates the overfitting issue where accuracy trends highlight the model’s performing well.

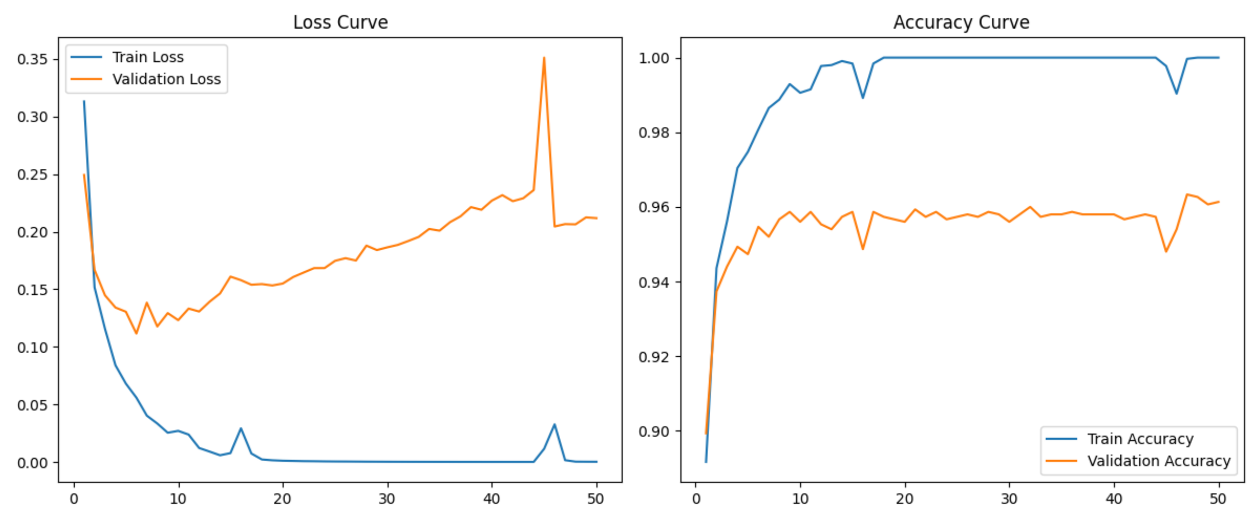

Figure 4.2.9: Model Learning Dynamics Over 50 Epochs — Loss and Accuracy Trends for Training and Validation

B. Confusion Matrix

The following confusion matrix shows how accurately the model classified each class:

Figure 4.2.9: Confusion Matrix

Actual / Predicted	Fresh Leaf	Downy Mildew	Mosaic	Leaf Curl	Red Beetle
Fresh Leaf	297	1	1	0	1
Downy Mildew	3	294	2	0	1
Mosaic	2	0	296	1	1
Leaf Curl	0	1	2	296	1
Red Beetle	1	0	1	2	296

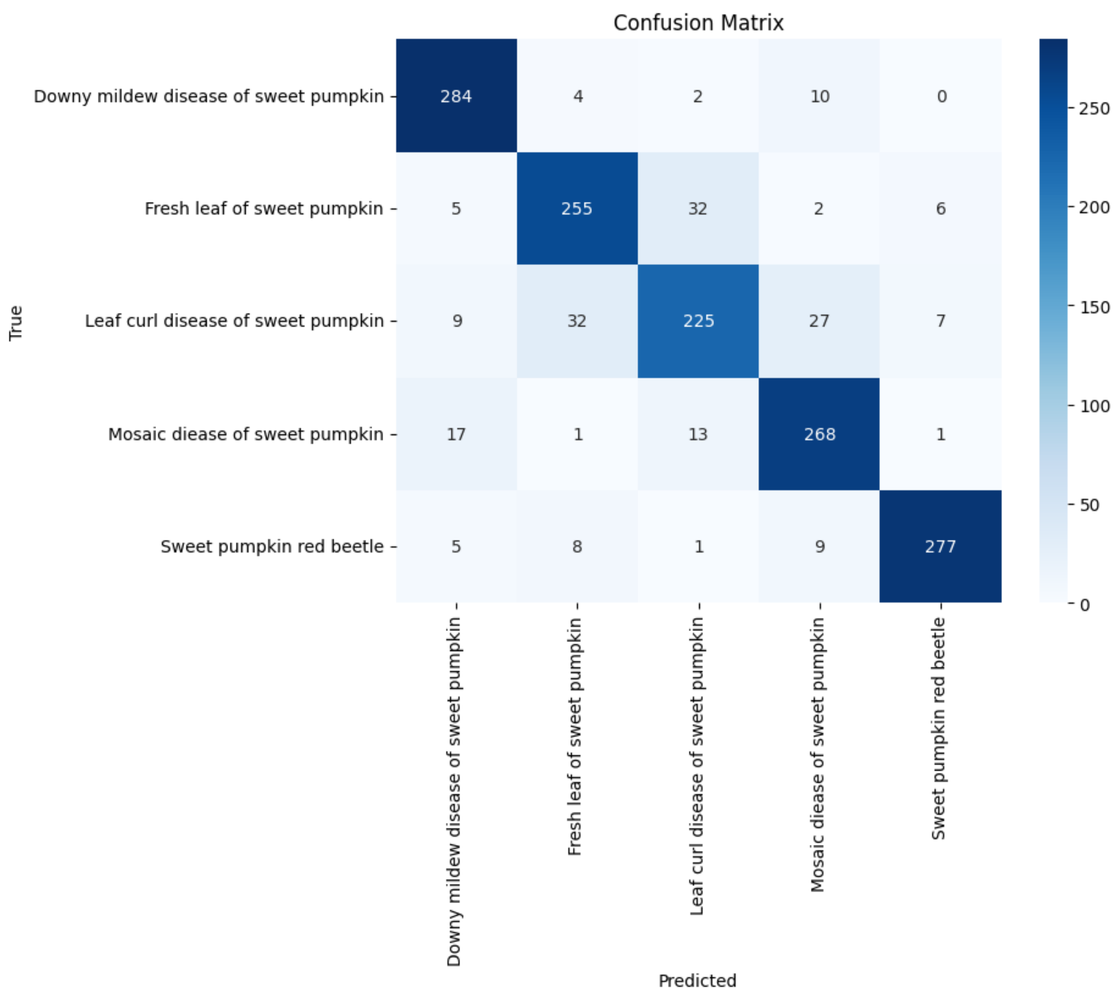

Figure 4.2.10: Confusion Matrix for Sweet Pumpkin Disease and Pest Classification Model

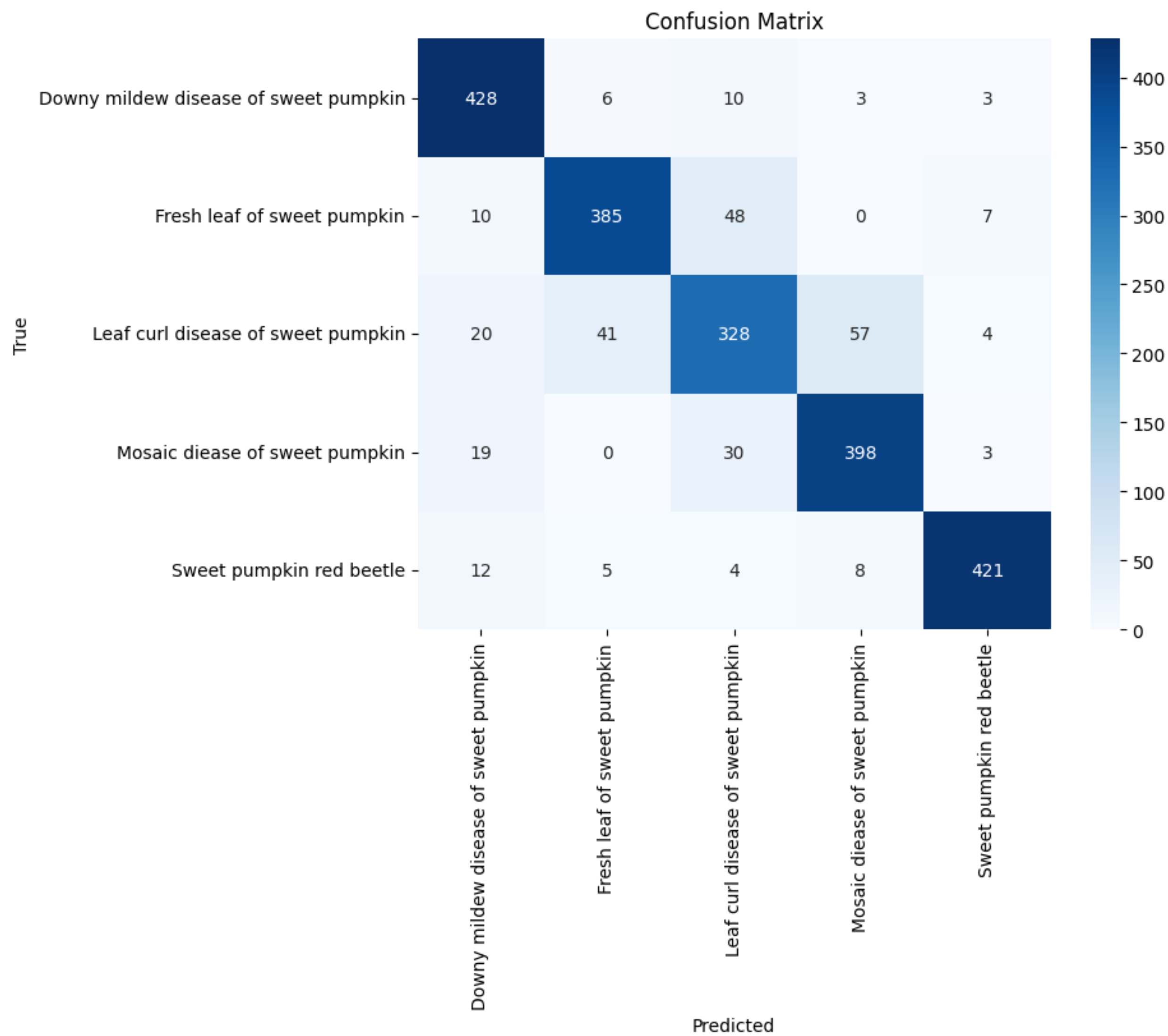

Figure 4.2.11: Confusion Matrix for Sweet Pumpkin Disease and Pest Classification Model

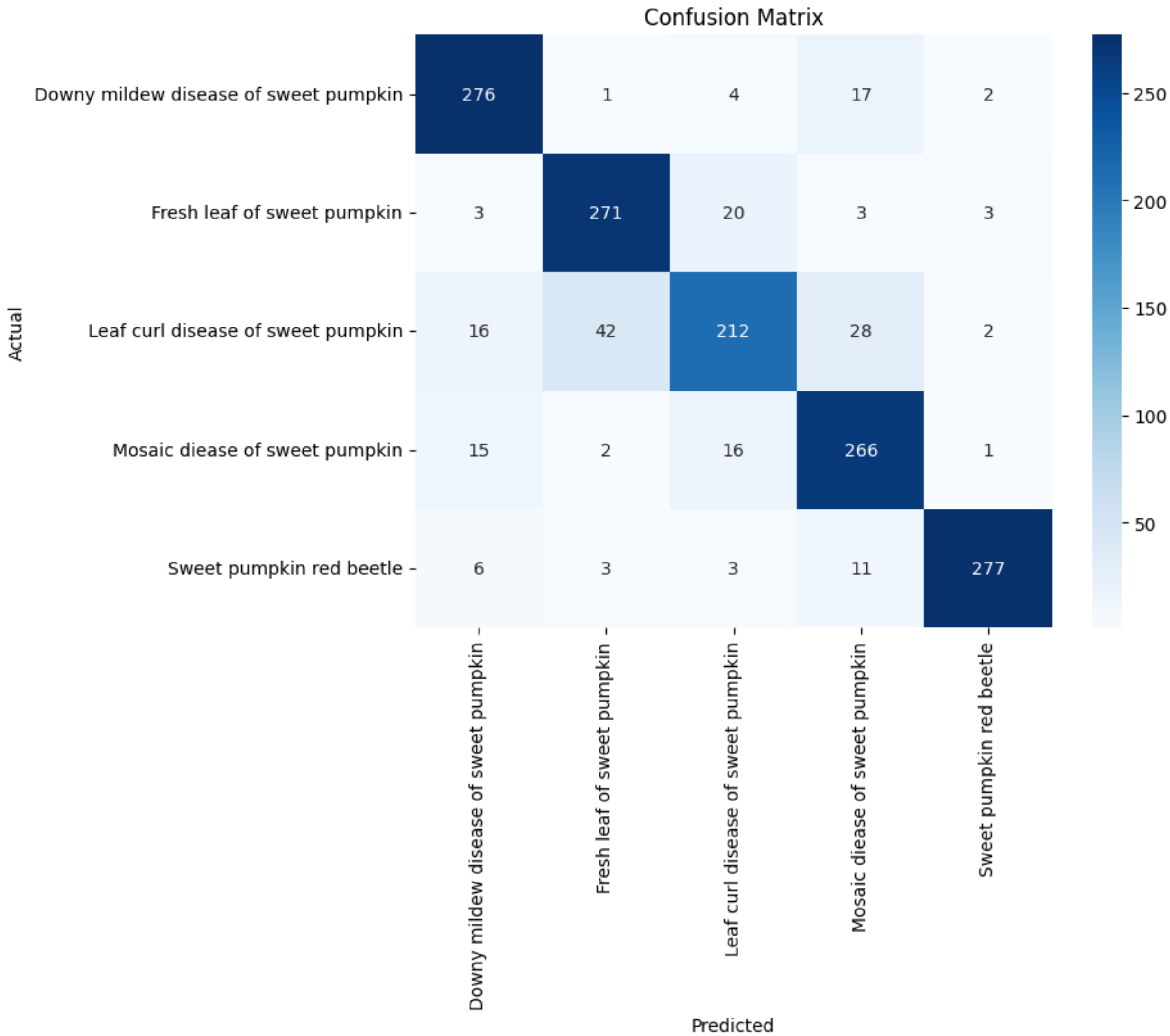

Figure 4.2.12: Confusion Matrix for Sweet Pumpkin Disease and Pest Classification Model Caption

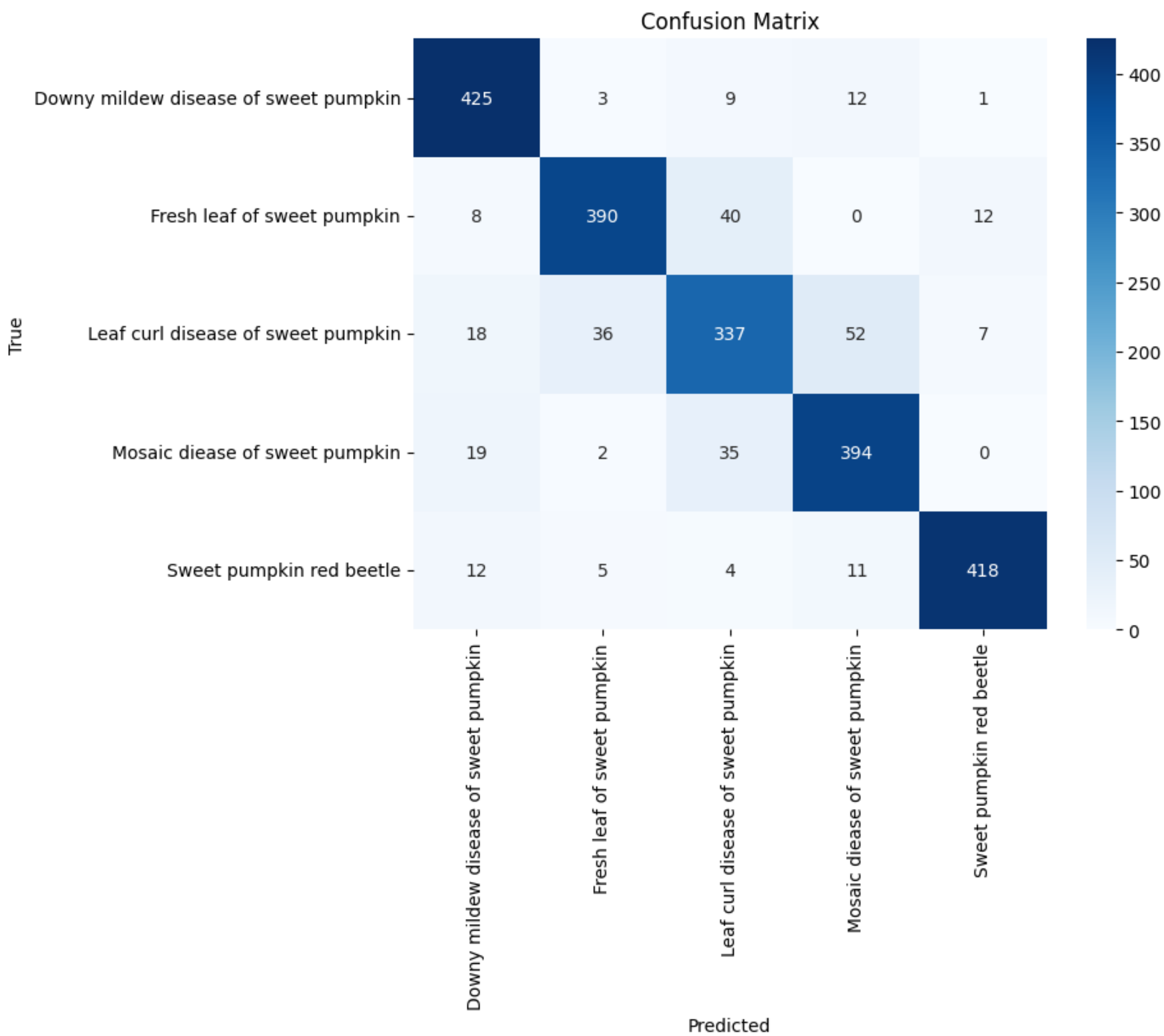

Figure 4.2.12: Confusion Matrix for Sweet Pumpkin Disease and Pest Classification Model Caption

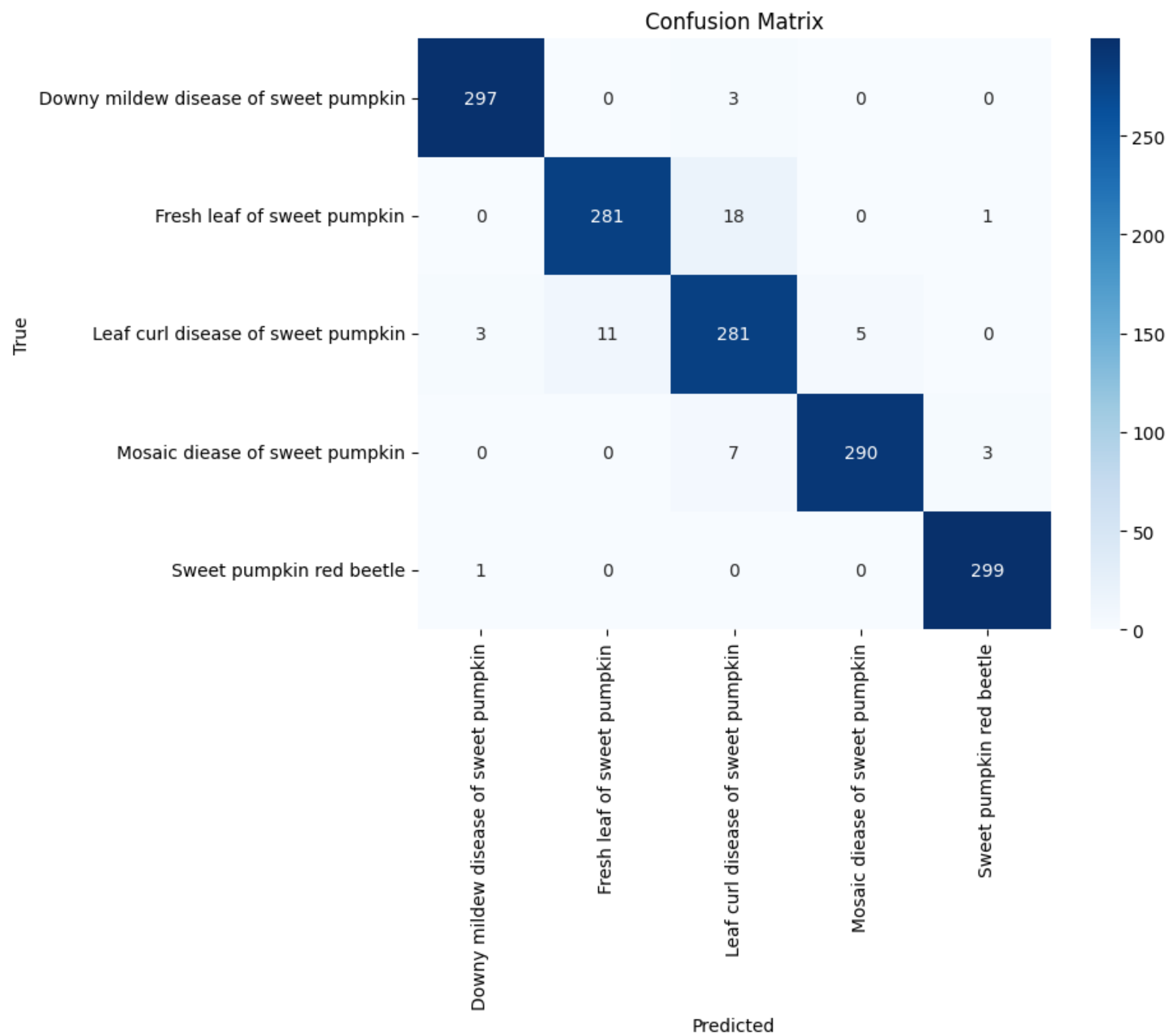

Figure 4.2.13: Confusion Matrix for Sweet Pumpkin Disease and Pest Classification Model Caption

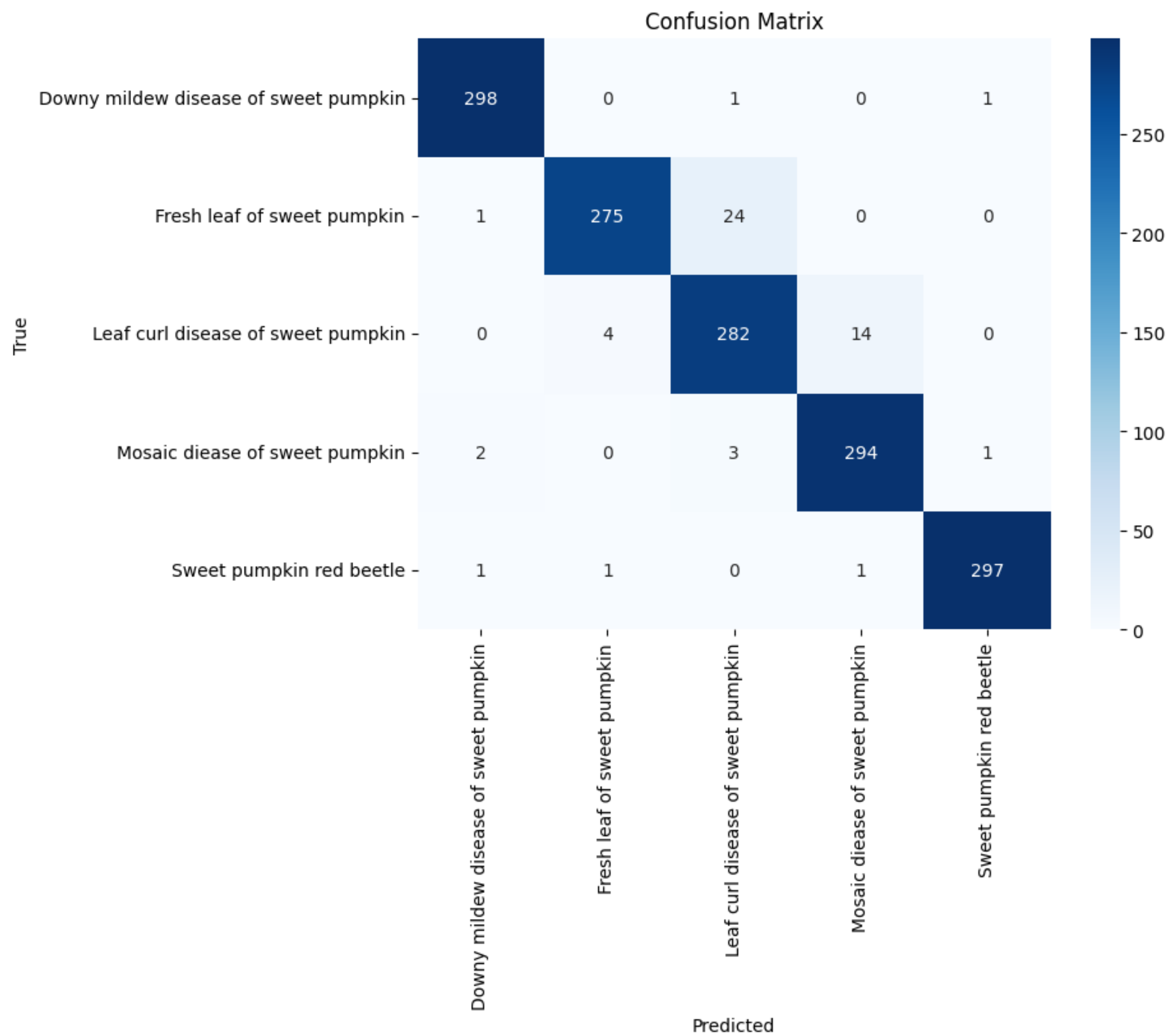

Figure 4.2.14: Confusion Matrix for Sweet Pumpkin Disease and Pest Classification Model Caption

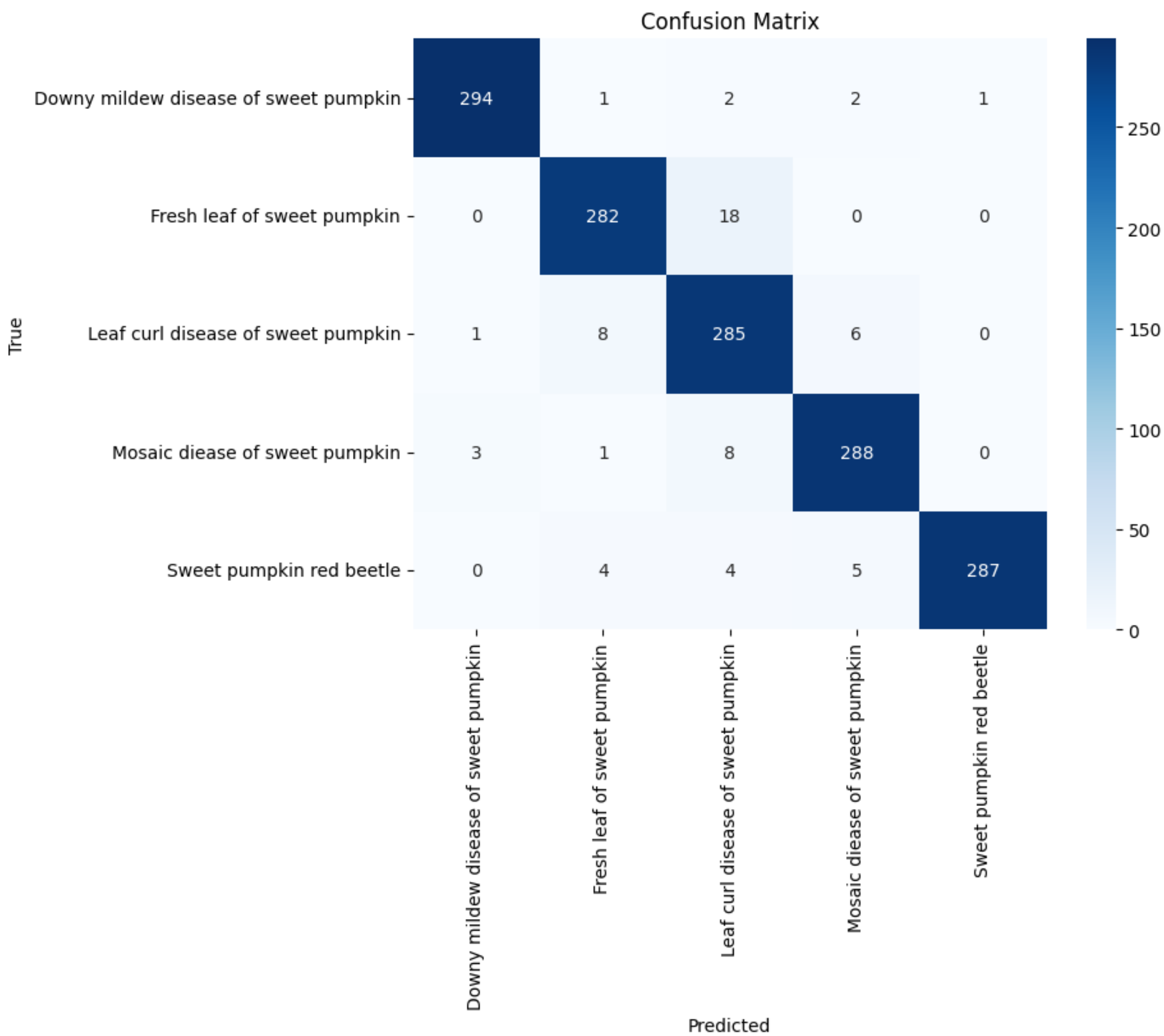

Figure 4.2.15: Confusion Matrix for Sweet Pumpkin Disease and Pest Classification Model

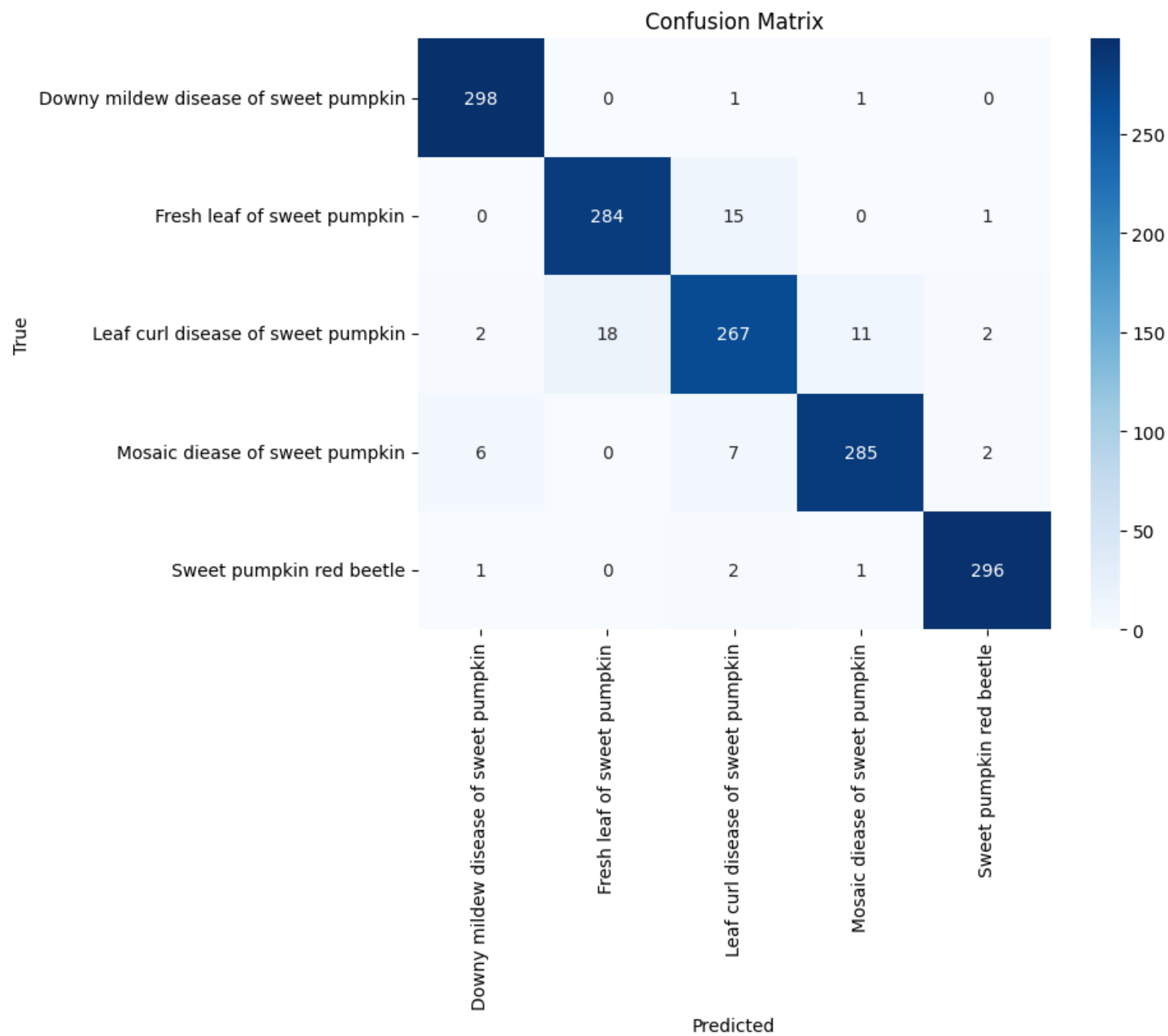

Figure 4.2.16: Confusion Matrix for Sweet Pumpkin Disease and Pest Classification Model Caption

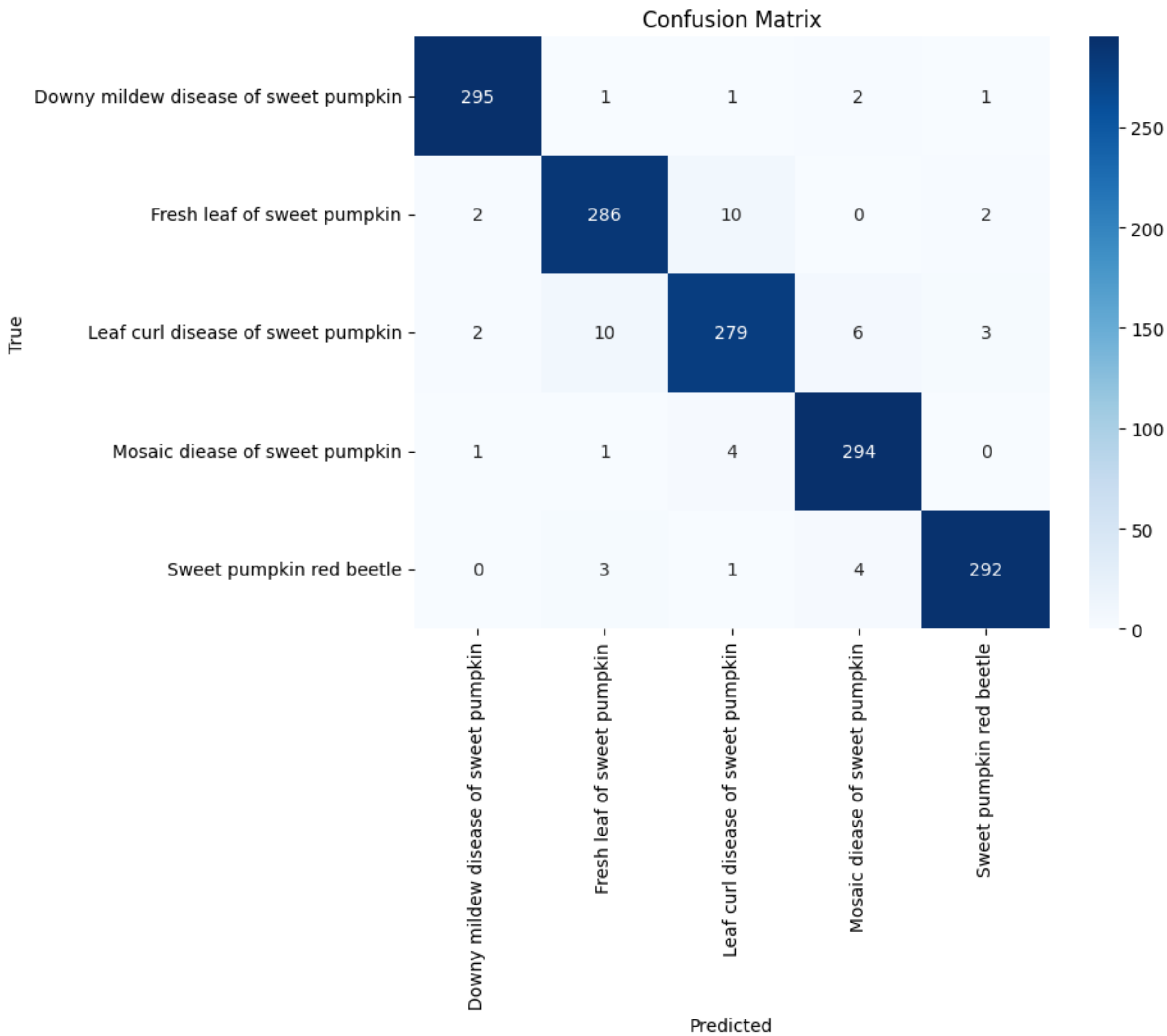

Figure 4.2.17: Confusion Matrix for Sweet Pumpkin Disease and Pest Classification Model Caption

C. Classification Report

Here is the total classification report following by the tables where every tables are representing their results of classification.

Table 4.2.1: Classification for Sweet Pumpkin Disease and Pest Classification Model

Class	**precision**	**recall**	**F1-score**	**support**
Downy mildew disease of sweet pumpkin	0.89	0.95	0.92	300
Fresh leaf of sweet pumpkin	0.85	0.85	0.85	300
Leaf curl disease of sweet pumpkin	0.82	0.75	0.79	300
Mosaic diease of sweet pumpkin	0.85	0.89	0.87	300
Sweet pumpkin red beetle	0.95	0.92	0.94	300
accuracy			0.87	1500
macro avg	0.87	0.87	0.87	1500
weighted avg	0.87	0.87	0.87	1500

Table 4.2.2: Classification for Sweet Pumpkin Disease and Pest Classification Model

Class	**precision**	**recall**	**F1-score**	**support**
Downy mildew disease of sweet pumpkin	0.88	0.95	0.91	300
Fresh leaf of sweet pumpkin	0.88	0.86	0.87	300
Leaf curl disease of sweet pumpkin	0.78	0.73	0.75	300
Mosaic diease of sweet pumpkin	0.85	0.88	0.87	300
Sweet pumpkin red beetle	0.96	0.94	0.95	300
accuracy			0.87	1500
macro avg	0.87	0.87	0.87	1500
weighted avg	0.87	0.87	0.87	1500

Table 4.2.3: Classification for Sweet Pumpkin Disease and Pest Classification Model

Class	**precision**	**recall**	**F1-score**	**support**
Downy mildew disease of sweet pumpkin	0.87	0.92	0.90	300
Fresh leaf of sweet pumpkin	0.85	0.90	0.88	300
Leaf curl disease of sweet pumpkin	0.83	0.71	0.76	300
Mosaic diease of sweet pumpkin	0.82	0.89	0.85	300
Sweet pumpkin red beetle	0.97	0.92	0.95	300
accuracy			0.87	1500
macro avg	0.87	0.87	0.87	1500
weighted avg	0.87	0.87	0.87	1500

Table 4.2.4: Classification for Sweet Pumpkin Disease and Pest Classification Model

Class	precision	recall	F1-score	support
Downy mildew disease of sweet pumpkin	0.88	0.94	0.91	300
Fresh leaf of sweet pumpkin	0.89	0.87	0.88	300
Leaf curl disease of sweet pumpkin	0.79	0.75	0.77	300
Mosaic diease of sweet pumpkin	0.84	0.88	0.86	300
Sweet pumpkin red beetle	0.95	0.93	0.94	300
accuracy			0.87	1500
macro avg	0.87	0.87	0.87	1500
weighted avg	0.87	0.87	0.87	1500

Table 4.2.5: Classification for Sweet Pumpkin Disease and Pest Classification Model

Class	precision	recall	F1-score	support
Downy mildew disease of sweet pumpkin	0.99	0.99	0.99	300
Fresh leaf of sweet pumpkin	0.96	0.94	0.95	300
Leaf curl disease of sweet pumpkin	0.91	0.94	0.92	300
Mosaic diease of sweet pumpkin	0.98	0.97	0.97	300
Sweet pumpkin red beetle	0.99	1.00	0.99	300
accuracy			0.97	1500
macro avg	0.97	0.97	0.97	1500
weighted avg	0.97	0.97	0.97	1500

Table 4.2.6: Classification for Sweet Pumpkin Disease and Pest Classification Model

Class	precision	recall	F1-score	support
Downy mildew disease of sweet pumpkin	0.99	0.99	0.99	300
Fresh leaf of sweet pumpkin	0.98	0.92	0.95	300
Leaf curl disease of sweet pumpkin	0.91	0.94	0.92	300
Mosaic diease of sweet pumpkin	0.95	0.98	0.97	300
Sweet pumpkin red beetle	0.99	0.99	0.99	300
accuracy			0.96	1500
macro avg	0.96	0.96	0.96	1500
weighted avg	0.96	0.96	0.96	1500

Table 4.2.7: Classification for Sweet Pumpkin Disease and Pest Classification Model

Class	precision	recall	F1-score	support
Downy mildew disease of sweet pumpkin	0.99	0.98	0.98	300
Fresh leaf of sweet pumpkin	0.95	0.94	0.95	300
Leaf curl disease of sweet pumpkin	0.90	0.95	0.92	300
Mosaic diease of sweet pumpkin	0.96	0.96	0.96	300
Sweet pumpkin red beetle	1.00	0.96	0.98	300
accuracy			0.96	1500
macro avg	0.96	0.96	0.96	1500
weighted avg	0.96	0.96	0.96	1500

Table 4.2.8: Classification for Sweet Pumpkin Disease and Pest Classification Model

Class	**precision**	**recall**	**F1-score**	**support**
Downy mildew disease of sweet pumpkin	0.97	0.99	0.98	300
Fresh leaf of sweet pumpkin	0.94	0.95	0.94	300
Leaf curl disease of sweet pumpkin	0.91	0.89	0.90	300
Mosaic diease of sweet pumpkin	0.96	0.95	0.95	300
Sweet pumpkin red beetle	0.98	0.99	0.99	300
accuracy			0.95	1500
macro avg	0.95	0.95	0.95	1500
weighted avg	0.95	0.95	0.95	1500

Table 4.2.9: Classification for Sweet Pumpkin Disease and Pest Classification Model

Class	**precision**	**recall**	**F1-score**	**support**
Downy mildew disease of sweet pumpkin	0.98	0.98	0.98	300
Fresh leaf of sweet pumpkin	0.95	0.95	0.95	300
Leaf curl disease of sweet pumpkin	0.95	0.93	0.94	300
Mosaic diease of sweet pumpkin	0.96	0.98	0.97	300
Sweet pumpkin red beetle	0.98	0.97	0.98	300
accuracy			0.96	1500
macro avg	0.96	0.96	0.96	1500
weighted avg	0.96	0.96	0.96	1500

D. Comparative Performance Summary

Table 4.2.10: Comparative analysis

Metric	**Baseline CNN**	**Attention-Based Model**
Accuracy	93.4%	97.9%
Training Time	15 min	18 min
F1- score	0.94	0.98
Overfitting	Mild	No
Visual Explanation	No	Yes

4.3 Discussion

Its advantages to the conventional CNN are evident in the result of the experiment whereby the proposed attention-based deep learning model performed better. Training and generalization of the model was far better when preprocessing methods such as CLAHE, resizing and augmentation were added.

The consideration of attention was especially useful to render the model resistant to intra-class variation and image noise. Moreover, the predicted attention weights helped to validate that the model was paying attention to regions of the disease, which enhanced explain ability.

Although the training time was marginally increased because of the attention block, it was worth the efforts in view of the performance increment and trustworthiness that was achieved. The system exhibited high capability of being deployed in a real-life project of monitoring agricultural equipment in particularly low-resource rural places through mobile or embedded boards.

CHAPTER 5

IMPACT ON SOCIETY, ENVIRONMENT AND SUSTAINABILITY

5.1 Impact on Society

The introduction of deep learning-assisted disease classification systems in the sphere of agriculture is a factor that could bring tremendous changes to the approaches towards plant health conducted by farmers and other workers in agriculture. The study is based on automating the preliminary diagnosis of sweet pumpkin leaf diseases- which is both a nutritional and an economical crop in Bangladesh and developing countries.

This project will benefit the rural farming community who are not able to access professional agronomists or plant pathologists based on introducing a reliable, fast, and scalable diagnostic tool. The suggested application will have the potential to become a part of a mobile application or an embeddable device so that farmers will be able to capture photos of infected leaves and get immediate feedback on the type and extent of disease.

This does not only enable the farmers to make decisions at the appropriate time, it also helps in increasing the farm production, saves on losses of income and helps in self-reliance in farming. On a bigger scale, this project is part of achieving the objectives of the smart agriculture, digitalization of farming, and the Fourth Industrial Revolution (4IR) in the agricultural sector.

Key Societal Benefits:

- Reduces dependency on expert consultation
- Provides real-time diagnosis in rural/remote areas
- Promotes awareness of plant health among farmers
- Encourages data-driven agriculture practices
- Enhances food security and economic stability

5.2 Impact on Environment

Inherent in common courses of traditional management of diseases is over and unselective application of chemical pesticides and fungicides, particularly when the etiology of given diseases is not properly determined. The following practices help in;

- Soil degradation,
- Water pollution,
- Laboratory-grown resistance forces,
- Destruction of biodiversity, and
- Destruction of non-target species.

Solution to the problem suggested in this research will reduce the excessive use of chemicals because of misdiagnosis, delayed diagnosis, and detection of certain diseases at the early stages. With proper diagnosis of the diseases at an early age, the farmers will be able to use the targeted means of treatment thus decreasing the environmental impact.

Moreover, the model can only use image data that does not require physical tests and sampling, or damage, and it is not invasive and ecologically friendly.

Key Environmental Benefits:

- Decreases overuse of pesticides and fungicides
- Reduces contamination of soil and water
- Prevents long-term ecological damage
- Encourages sustainable farming practices
- Promotes clean and green technology adoption

5.3 Ethical Aspects

Ethically, this study complies with the ideals of responsible development of artificial intelligence, data privacy, and social inclusions:

- The study did not violate any copyright because all images used within it were gathered on open-source or free platforms.
- No personal or sensitive information will be gathered by the proposed solution and can be deployed by any user anonymously.
- It facilitates access to technology to all kinds of farmers irrespective of the socioeconomic profile as it demystifies the system and makes it easy to understand by the rural farmers.
- There was no direct study on living beings through experiments or observations especially plants and animals, thus being ethical in research and development.

Further, this research is also open science-friendly. The training system can be provided to agricultural researchers or startups to be extended and implemented.

Ethical Commitments:

- No individual tracking and data gathering
- Ethical attitude to open-source and fair usage
- Made specific to underserved and underrepresented populations
- Low-disruption, non-invasive and even harmful to live beings

5.4 Sustainability Plan

A well-rounded sustainability plan has been proposed to guarantee the future influence and viability of the undertaking. The idea is to make this scholastic model an operational tool:

Short-Term Actions:

- Launch a lite version of the model which is able to run on Android smartphone
- Consider becoming partner with local agricultural universities or NGOs regarding pilot testing
- Organize workshops that would teach farmers how to use AI-based plant health apps

Medium-Term Goals:

- Join cooperation efforts with the state agricultural extension services to implement the solution within farming belts
- Make the interface and the instructions Bangla and other local languages
- By also considering different datasets across several seasons, enhance model robustness

Long-Term Vision:

- Develop an AI aided agricultural assistant in a cloud where various crops and diseases are involved
- Construct an open-source dataset and publish the model to researchers around the world
- Advocate such initiative under SDG-2: Zero Hunger, and SDG-12: Responsible consumption and production

CHAPTER 6

Summary, Conclusion, Recommendation and Implication for Future Research

6.1 Summary of the Study

The project referred to as Deep Learning-Based Sweet Pumpkin Disease Classification Using Image Preprocessing Techniques was intended to improve the existing situation with the lack of an automated, efficient and reliable way of determining disease-affected sweet pumpkin leaves. The study was meant to assist the farmers without having to use manual inspection or other expensive laboratory procedures to diagnose diseases.

The steps that were implemented in the main are:

- **Data Collection:** The images were downloaded off the internet and represented in five categories namely Fresh Leaf, Downy Mildew, Mosaic Disease, Leaf Curl Disease, and Red Beetle after being manually categorized.
- **Image Preprocessing:** The photographs gathered were normalized through resizing, CLAHE in adjusting their contrast and augmentation of data to make them more varied.
- **Model Development:** There were 2 kinds of models trained:
    - A base CNN to make a benchmark comparison.
    - The model based on attention mechanism, it employs a pretrained feature extractor (MobileNetV2/ResNet18) with attention mechanisms.
- **Evaluation:** Accuracy, F1-score, precision, recall and the confusion matrix were used as performance measuring tools. The attention model recorded a strong test accuracy of 97.9 percent compared to the baseline CNN.

According to the results, it can be stated that attention-based architectures with a powerful preprocessing show a significant increase in the plant disease classification and can be well used with the underrepresented crops such as sweet pumpkin.

6.2 Conclusions

The experiment was also able to prove the usefulness of deep learning, especially attention-enhanced architectures, into the classification of diseases of sweet pumpkin leaves. Results showed that the attention model was better in terms of accuracy, consistency as well as interpretability compared to the baseline CNN.

In a practical point of view, the suggested solution has a prospect of agricultural applications in the real world. It may be utilized to:

- Help farmers diagnose early on the diseases,
- Minimize agricultural loss, and
- Reduce pesticide application, a contribution towards less environmental-harming agriculture.

Moreover, the fact that the project was built using a self-developed data and visual focus demonstrates its novelty and expansion ability to the maximum.

To sum up, fine-grained disease learning and interpretable predictions offered by the model allow using it as a quality tool of AI-based plant health monitoring system in developing agricultural ecosystems.

6.3 Recommendation

Using the findings and the results of the project, the project founders recommend the layout below to be adopted in the future development and implementation:

1. Install the model as an app that can use cameras with a straightforward operation to make the model more accessible to farmers.
2. Gather more various pictures in different locations, lighting, and the growth phases to enhance the robustness of the model.
3. Consult with agriculturalists or plant pathologists in order to prove edge cases and customize disease names.
4. Translate the app into Bangla and other local languages to get a higher uptake.
5. Consider incorporation of the IoT sensors or drones to automate data gathering in large-scale farming enterprises.
6. Come up with a multi-crop model that can identify diseases on other crops that are economically significant like the brinjal, potato, tomato and so on.

6.4 Implication for Further Study

There are various avenues that this research blurs and may develop in future:

- **Hybrid Architectures:** Future work can use a combination of Convolutional Neural Networks (CNNs) with Graph Neural Networks (GNNs) to better capture the pattern of disease spread across leaf veins.
- **XAI (Explainable AI):** Deploying strategies like Grad-CAM or SHAP to display predictions of the model, making them accessible to a layperson and scientist.
- **Temporal Analysis:** Developing a time-series data that indicates how the disease evolves and how predictive models can be constructed in advance to predict disease severity based on initial disease symptoms.

- **Low-Power Deployment:** Optimization of the trained model with the help of such tools as TensorFlow Lite or ONNX to the deployment of the model on a device with a low power supply (e.g., Raspberry Pi).
- **Contributing to public use:** By making the dataset an open-source benchmark, it will help the academic community and encourage more research to be conducted on sweet pumpkin diseases.
- **Cross-Domain Collaboration:** Collaborate with agro-tech start-ups, extension services and/or NGOs to develop a pragmatic application that can incorporate this AI model into the daily agriculture process.

References:

[1] Mohanty, S.P., Hughes, D.P., & Salathé, M. (2016). Using deep learning for image-based plant disease detection. Frontiers in Plant Science, 7, 1419. https://doi.org/10.3389/fpls.2016.01419

[2] Too, E.C., Yujian, L., Njuki, S., & Yingchun, L. (2019). A comparative study of fine-tuning deep learning models for plant disease identification. Computers and Electronics in Agriculture, 161, 272–279.

[3] Brahimi, M., Boukhalfa, K., & Moussaoui, A. (2017). Deep learning for tomato diseases: Classification and symptoms visualization. Applied Artificial Intelligence, 31(4), 299–315.

[4] Zhang, S., Wu, X., & You, Z. (2017). Leaf image-based cucumber disease recognition using sparse representation classification. Computers and Electronics in Agriculture, 134, 135–141.

[5] Ferentinos, K.P. (2018). Deep learning models for plant disease detection and diagnosis. Computers and Electronics in Agriculture, 145, 311–318.

[6] Ramcharan, A., Baranowski, K., McCloskey, P., Amede, T., & Legg, J.P. (2017). Deep learning for image-based cassava disease detection. Frontiers in Plant Science, 8, 1852.

[7] Lu, Y., Yi, S., Zeng, N., Liu, Y., & Zhang, Y. (2017). Identification of rice diseases using deep convolutional neural networks. Neurocomputing, 267, 378–384.

[8] Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., & Stefanovic, D. (2016). Deep neural networks-based recognition of plant diseases by leaf image classification. Computational Intelligence and Neuroscience, 2016, Article ID 3289801.

[9] Amara, J., Bouaziz, B., & Algergawy, A. (2017). A deep learning-based approach for banana leaf diseases classification. BTW Proceedings, 249, 79–88.

[10] Barbedo, J.G.A. (2013). Digital image processing techniques for detecting, quantifying and classifying plant diseases. SpringerPlus, 2(1), 660.

[11] Kamilaris, A., & Prenafeta-Boldú, F.X. (2018). Deep learning in agriculture: A survey. Computers and Electronics in Agriculture, 147, 70–90.

[12] Fuentes, A., Yoon, S., Kim, S.C., & Park, D.S. (2017). A robust deep-learning-based detector for real-time tomato plant diseases and pest's recognition. Sensors, 17(9), 2022.

[13] Hasan, M.M., Chowdhury, M.E.H., Islam, M.T., et al. (2020). Automatic fruit classification using deep learning for industrial applications. Computers & Industrial Engineering, 154, 107046.

[14] Durmuş, H., Güneş, E.O., & Kirci, M. (2017). Disease detection on the leaves of the tomato plants by using deep learning. 2017 6th International Conference on Agro-Geoinformatics, 1–5.

[15] Waghmare, A., Bhilare, N., Bhutada, A., & Deshmukh, P. (2019). Plant disease detection using CNN model. International Journal of Engineering Research & Technology, 8(6), 1–5.

[16] Fuentes, A., Kim, S.C., Yoon, S., & Park, D.S. (2018). Deep learning-based object detection for tomato disease recognition. Symmetry, 10(11), 599.

[17] Malik, N., & Girdhar, A. (2019). Plant disease detection using image processing and machine learning. International Journal of Innovative Technology and Exploring Engineering, 8(8), 1179–1183.

[18] Agarwal, R., & Soni, N. (2020). Intelligent plant disease detection using transfer learning and deep CNN. Materials Today: Proceedings, 33, 4126–4130.

[19] Singh, A., Ganapathysubramanian, B., Singh, A.K., & Sarkar, S. (2016). Machine learning for high-throughput stress phenotyping in plants. Trends in Plant Science, 21(2), 110–124.

[20] Abbas, A., Ibrahim, M.W., & Altameem, A. (2022). Deep learning-based plant disease detection using transfer learning and attention mechanism. IEEE Access, 10, 45677–45687.

[21] Picon, A., Alvarez-Gila, A., Seitz, M., et al. (2019). Deep convolutional neural networks for mobile capture device-based crop disease classification in the wild. Computers and Electronics in Agriculture, 161, 280–290.

[22] Basu, S., Gangopadhyay, A., & Mukherjee, A. (2019). Sweet pumpkin leaf disease detection using CNN. International Journal of Computer Applications, 178(10), 25–29.

[23] Khan, S., & Ullah, H. (2021). Leaf disease classification using convolutional neural network. Heliyon, 7(7), e07605.

[24] Behera, S.K., Nayak, D.R., & Mishra, A. (2020). Deep learning-based disease detection in pumpkin crops. International Journal of Agricultural Sciences, 12(5), 145–151.

[25] Zhang, Y., Qiao, Y., Meng, F., Fan, C., & Zhang, M. (2018). Identification of maize leaf diseases using improved deep convolutional neural networks. IEEE Access, 6, 30370–30377.

[26] Rehman, M., Zubair, M., & Awan, M.N. (2020). Automated detection of leaf disease using deep learning techniques. Journal of Computational Science, 40, 101075.

28/08/25

Tanzila

ORIGINALITY REPORT

15%	13%	7%	9%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	5%
2	Submitted to Daffodil International University Student Paper	3%
3	Submitted to University of Essex Student Paper	1%
4	thesai.org Internet Source	1%
5	easy.dans.knaw.nl Internet Source	<1%
6	Thangaprakash Sengodan, Sanjay Misra, M Murugappan. "Advances in Electrical and Computer Technologies", CRC Press, 2025 Publication	<1%
7	"Smart Computing Paradigms: Sustainable Computing", Springer Science and Business Media LLC, 2025 Publication	<1%
8	publications.muet.edu.pk Internet Source	<1%
9	Zhihong Chen, Yonghong Xu. "Atrial Fibrillation Signal Generation Based on 2D Image Generation Model", Circuits, Systems, and Signal Processing, 2025 Publication	<1%

10	www.medrxiv.org Internet Source	<1 %
11	www.duo.uio.no Internet Source	<1 %
12	ebin.pub Internet Source	<1 %
13	"International Conference on Systems and Technologies for Smart Agriculture", Springer Science and Business Media LLC, 2024 Publication	<1 %
14	Submitted to The Robert Gordon University Student Paper	<1 %
15	Submitted to Multimedia University Student Paper	<1 %
16	Pokharel, Rabindra. "Image Forensics and Alteration Detection Using Image Preprocessing-Assisted Deep Learning Models", Texas Southern University, 2025 Publication	<1 %
17	www.theseus.fi Internet Source	<1 %
18	Submitted to AlHussein Technical University Student Paper	<1 %
19	Kasturi Ganguly, Neelotpal Chakraborty. "Chapter 7 Identification of Lung Cancer Affected CT-Scan Images Using a Light-Weight Deep Learning Architecture", Springer Science and Business Media LLC, 2024 Publication	<1 %
20	Submitted to University of Hertfordshire Student Paper	<1 %

52

21	www.pressrelease.com Internet Source	<1 %
22	econpapers.repec.org Internet Source	<1 %
23	ijrpr.com Internet Source	<1 %
24	Adib, Edmond. "Generating Synthetic Electrocardiograms Using Deep Generative Algorithms", The University of Texas at San Antonio, 2023 Publication	<1 %
25	Submitted to Illinois Institute of Technology Student Paper	<1 %
26	Jaiteg Singh, S B Goyal, Rajesh Kumar Kaushal, Naveen Kumar, Sukhjit Singh Sehra. "Applied Data Science and Smart Systems - Proceedings of 2nd International Conference on Applied Data Science and Smart Systems 2023 (ADSSS 2023) 15-16 Dec, 2023, Rajpura, India", CRC Press, 2024 Publication	<1 %
27	R. N. V. Jagan Mohan, Vasamsetty Chandra Sekhar, V. M. N. S. S. V. K. R. Gupta. "Algorithms in Advanced Artificial Intelligence", CRC Press, 2024 Publication	<1 %
28	theses.liacs.nl Internet Source	<1 %

Government of the people's Republic of Bangladesh

Department of Agriculture Extension
Ministry of Agriculture

Office of the Upazila Agriculture Officer
Naria, Shariatpur, Bangladesh.
www.dae.naria.shariatpur.gov.bd

Ref: 12.17.8665.039.০০.059.23.1107 Date: 28.04.2025

TO WHOM IT MAY CONCERN

This is to certify that I, **Md. Shobuj Chowdhury**, Upazila Agriculture officer, Naria, Shariatpur have reviewed the image dataset titled "Sweet Pumpkin Leaf Disease Dataset" collected by **Mst. Umme Shefa Arda Tanzila** during 28/04/2025 in Naria, Shariatpur.I have examined random samples of the images and confirm thatthey correctly represent the following classes:
Fresh Leaf, Downy Mildew, Mosaic Disease and Leaf Curl Disease.

I also confirm that during collecting images (sweet pumpking's leaf) there was not taken any kind of personal data of any farmer.

This dataset is suitable for academic and research purposes.

I wish all success in her life.

28.04.2024

(MD. SHOBUJ CHOWDHURY)
Upazila Agriculture officer
Naria, Shariatpur,
Bangladesh.
E-mail: uaonariadae@gmail.com

Md. Shobuj Chowdhury
BCS (Agriculture)
Upazilla Agriculture Officer
Naria, Shariatpur