AI Course

Capstone Project   
Final Report

For students (instructor review required)

ⓒ2023 SAMSUNG. All rights reserved.

Samsung Electronics Corporate Citizenship Office holds the copyright of this document.

This document is a literary property protected by copyright law so reprint and reproduction without permission are prohibited.

To use this document other than the curriculum of Samsung Innovation Campus, you must receive written consent from copyright holder.

|  |
| --- |
| Plant Diseases Detection |

30/10/2024

Machine Masters

Omar Alnosyan

Sarah Alaridi

Manal Alsuwat

Abdullah Baqais

Rahaf Alwaladi

Ameer Nawab

Content

1. Introduction

1.1. Background Information

1.2. Motivation and Objective

1.3. Members and Role Assignments

1.4. Schedule and Milestones

2. Project Execution

2.1. Data Acquisition

2.2. Training Methodology

2.3. Workflow

2.4. System Diagram

3. Results

3.1. Data Preprocessing

3.2. Exploratory Data Analysis (EDA)

3.3. Modeling

3.4. User Interface

3.5. Testing and Improvements

4. Projected Impact

4.1. Accomplishments and Benefits

4.2. Future Improvements

5. Team Member Review and Comment

6. Instructor Review and Comment

1. Introduction

1.1. Background Information

The dataset used in this project comes from a publicly available source on Kaggle, designed for plant disease detection. It contains over 87,000 RGB images of crop leaves, spread across 38 distinct classes. Each class represents a specific plant disease or the healthy state of the plant. The dataset includes images from various crops, such as apple, grape, corn, and potato, with all images standardized at a resolution of 256x256 pixels. Plant disease detection is critical in agriculture for minimizing crop loss and ensuring food security. Early diagnosis through technology can save both time and resources, compared to traditional, manual inspection methods. By using this dataset, the project aims to develop a machine learning model capable of identifying diseases accurately from images, providing a valuable tool for farmers and agricultural researchers.

1.2. Motivation and Objective

This project’s main motivation is to empower the agricultural sector by developing a reliable, accessible tool for early detection of plant diseases through leaf images. Early detection allows farmers to act swiftly, reducing the spread and severity of infections, which improves productivity and food security. By using Convolutional Neural Networks (CNNs) and transfer learning, our aim is to create an efficient, scalable model that can accurately classify plant diseases from images. This will enable faster diagnosis, helping farmers protect crops more effectively with minimal resources.

1.3. Members and Role Assignments

* **Manal:** Focused on exploratory data analysis (EDA), data preprocessing, and data modeling.
* **Rahaf:** Handled the data splitting and contributed to data modeling.
* **OMAR & Sarah:** Collaborated on model selection and evaluation.
* **Abdullah & Ameer:** Managed the project documentation and reporting.

1.4. Schedule and Milestones

The project was completed over a structured timeline, with clear milestones at each phase to ensure smooth progress. Below is a breakdown of the key tasks, their timeline, and the milestones achieved:

| **Task** | **Start date** | **End date** | **Responsibility** | **Deliverable** |
| --- | --- | --- | --- | --- |
|
|
| Data Collection | 2024-10-19 | 2024-10-20 | Manal | Collected the dataset and ready to be used. |
| Exploratory Data Analysis | 2024-10-20 | 2024-10-24 | Manal & Rahaf | Explore class distribution, check image quality, and visualize sample images. |
| Data Preprocessing | 2024-10-22 | 2024-10-24 | Manal & Rahaf | Preprocessed dataset with resized images and augmented data. |
| Data Splitting | 2024-10-24 | 2024-10-24 | Manal & Rahaf | Split dataset into training, validation, and test sets. |
| Model Training | 2024-10-24 | 2024-10-26 | Manal & Rahaf | Training models using CNN and transfer learning. |
| Model Selection | 2024-10-26 | 2024-10-28 | OMAR& Sarah | Select the best model based on performance metrics. |
| Model Evaluation | 2024-10-26 | 2024-10-28 | OMAR& Sarah | Evaluation report with performance metrics (accuracy, precision, recall, F1-score). |

￼

Project ExecutionProject Execution

2.1. Data Acquisition

The dataset for this project was acquired from a publicly available source on Kaggle. It contains approximately 87,000 RGB images of crop leaves, which are categorized into 38 different classes. These classes represent various diseases and healthy plant states. Each image is a size of 256x256 pixels, showing distinct visual differences between healthy and diseased leaves, making the dataset suitable for training machine learning models in plant disease detection. Initially, the dataset was provided with an 80/20 split between training and validation sets. However, the test set contained only 33 images, which was insufficient for robust model evaluation. Therefore, we modified the dataset by splitting the original training set into an 80/20 ratio to create new training and validation sets. The original validation data was then used as the test set to ensure a more balanced and effective evaluation of the model’s performance.

2.2. Training Methodology

Two different models were utilized for the plant disease detection task: a custom Convolutional Neural Network (CNN) and a transfer learning approach using the VGG16 model.

* **Custom CNN Model**

The architecture of the custom CNN consisted of multiple convolutional layers with ReLU activation functions to extract features from the images, followed by max-pooling layers to reduce dimensionality while retaining important information. The network ended with fully connected layers for classification, and the final layer used a softmax activation function to categorize the images into one of 38 classes. This model was trained using the Adam optimizer and the categorical cross-entropy loss function. It was run for several epochs to fine-tune the weights, using a validation set to monitor performance. Data augmentation techniques such as image rotation, flipping, and zooming were applied during training to improve the model’s generalization capabilities.

* **Transfer Learning with VGG16**

In addition to the custom CNN, a pre-trained VGG16 model was used as part of a transfer learning approach. VGG16, which was pre-trained on the ImageNet dataset, served as the base model, and additional custom layers were added on top to adapt it for the plant disease dataset. The added layers included fully connected layers, batch normalization for improved training stability, and dropout to prevent overfitting. The transfer learning model was also trained using the Adam optimizer with categorical cross-entropy loss. It was fine-tuned for 10 epochs with a batch size of 30, and data augmentation techniques were similarly applied to enhance the model's robustness. Early stopping was employed during training to avoid overfitting.

Both models were validated using a dedicated validation set, and their performances were compared to select the best approach for the plant disease detection task.

2.3. Workflow

The workflow for the plant disease detection project consisted of several key stages, each essential to building and evaluating the machine learning models:

1. **Dataset Acquisition:**  
   The dataset was sourced from Kaggle and contained around 87,000 images across 38 classes, representing both healthy and diseased plant leaves. These images were organized into training, validation, and test sets.
2. **Exploratory Data Analysis (EDA):**  
   EDA was conducted to understand the class distribution, image quality, and overall dataset structure. This step involved visualizing the number of images per class and identifying any potential imbalances that could affect model performance.
3. **Data Splitting:**  
   Due to the limitations in the original test set size, the data was restructured by splitting the training set into new training and validation sets. The original validation set was then used as the test set to provide a more balanced evaluation.
4. **Data Preprocessing:**  
   The images were resized to a uniform shape of 224x224 pixels, and data augmentation techniques such as rotation, flipping, and zooming were applied to increase variability in the dataset. These preprocessing steps helped the models generalize better to new data.
5. **Model Development:**  
   Two models were developed for this project:

* **Custom CNN Model:** Built specifically for this dataset with convolutional, max-pooling, and fully connected layers, trained to classify the 38 disease categories.
* **Transfer Learning VGG16 Model:** A VGG16 model pre-trained on ImageNet was adapted with additional layers to suit the plant disease classification task.

1. **Model Training:**  
   Both models were trained using the Adam optimizer and categorical cross-entropy loss function. The training process included monitoring the model performance on a validation set, with early stopping and model checkpoints to prevent overfitting.
2. **Model Evaluation:**  
   The models were evaluated on the test set using performance metrics like accuracy, precision, recall, and F1-score. These metrics helped identify the best-performing model, which achieved an accuracy of 83.2%.

2.4. System Design

1. **Data Input**: the dataset consists of around 87,000 images, and is loaded and organized into three subsets: training, validation, and test sets.
2. **Data Preprocessing:** images are resized to 224x224 pixels, and data augmentation techniques, such as rotation, flipping, and zooming, are applied to increase dataset diversity.
3. **Model Training:** two models were chosen for this task: a custom CNN model and a pre-trained VGG16 model with transfer learning. The training module takes the preprocessed data and trains both models, using the Adam optimizer and categorical cross-entropy loss.
4. **Model Evaluation:** the models are evaluated on a reserved test set. This module computes key performance metrics, such as accuracy, precision, recall, and F1-score, providing a comprehensive view of each model’s classification performance.

Results Results

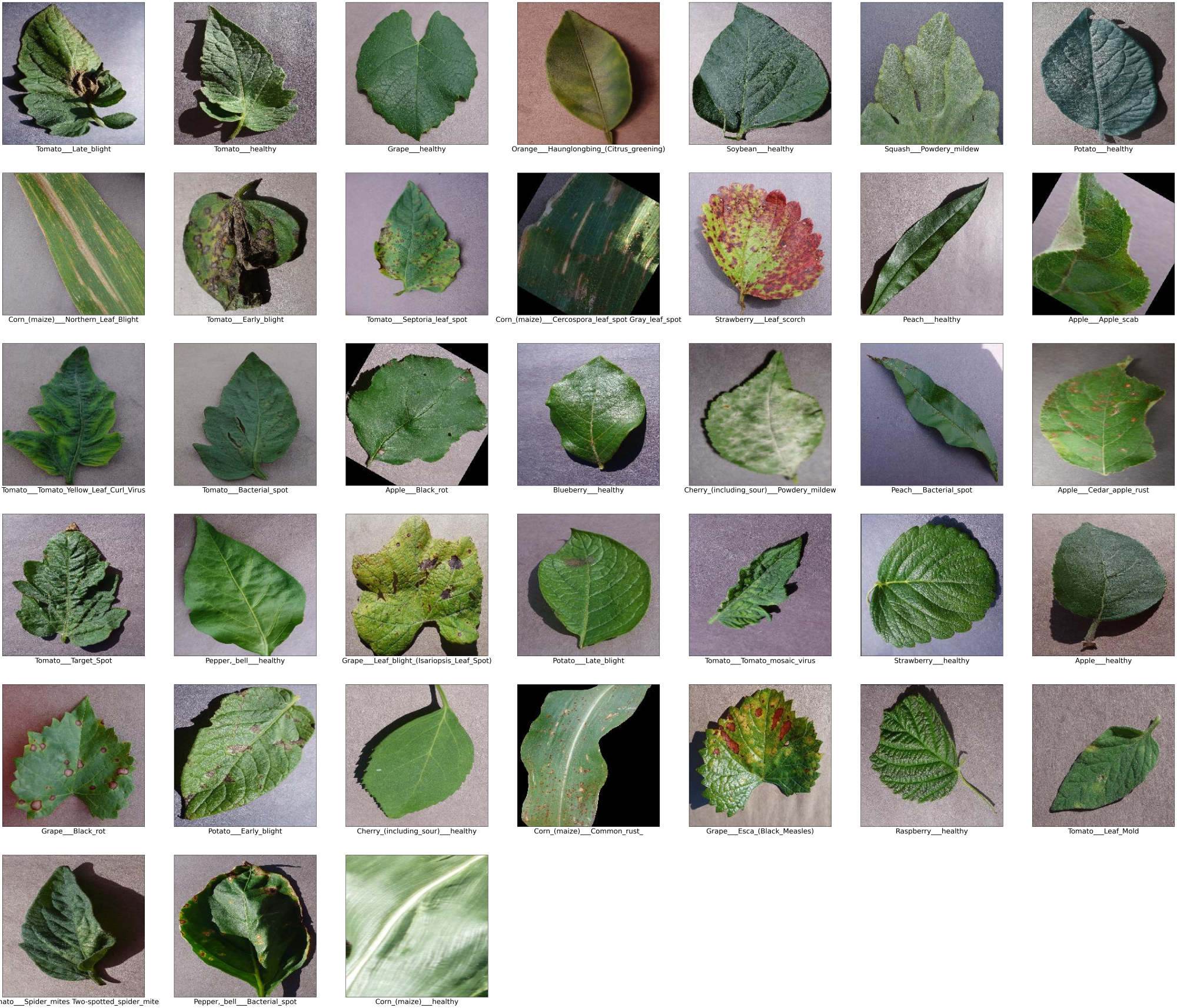
3.1. Data Preprocessing

Data preprocessing involved resizing all images to 224x224 pixels, normalizing pixel values, and applying various augmentation techniques. Augmentation methods such as rotation, flipping, and shifting were used to increase the dataset’s diversity, improving the model’s ability to generalize. The processed data was then split into training, validation, and test sets, with an 80-20 split for training and validation to ensure balanced learning and testing.

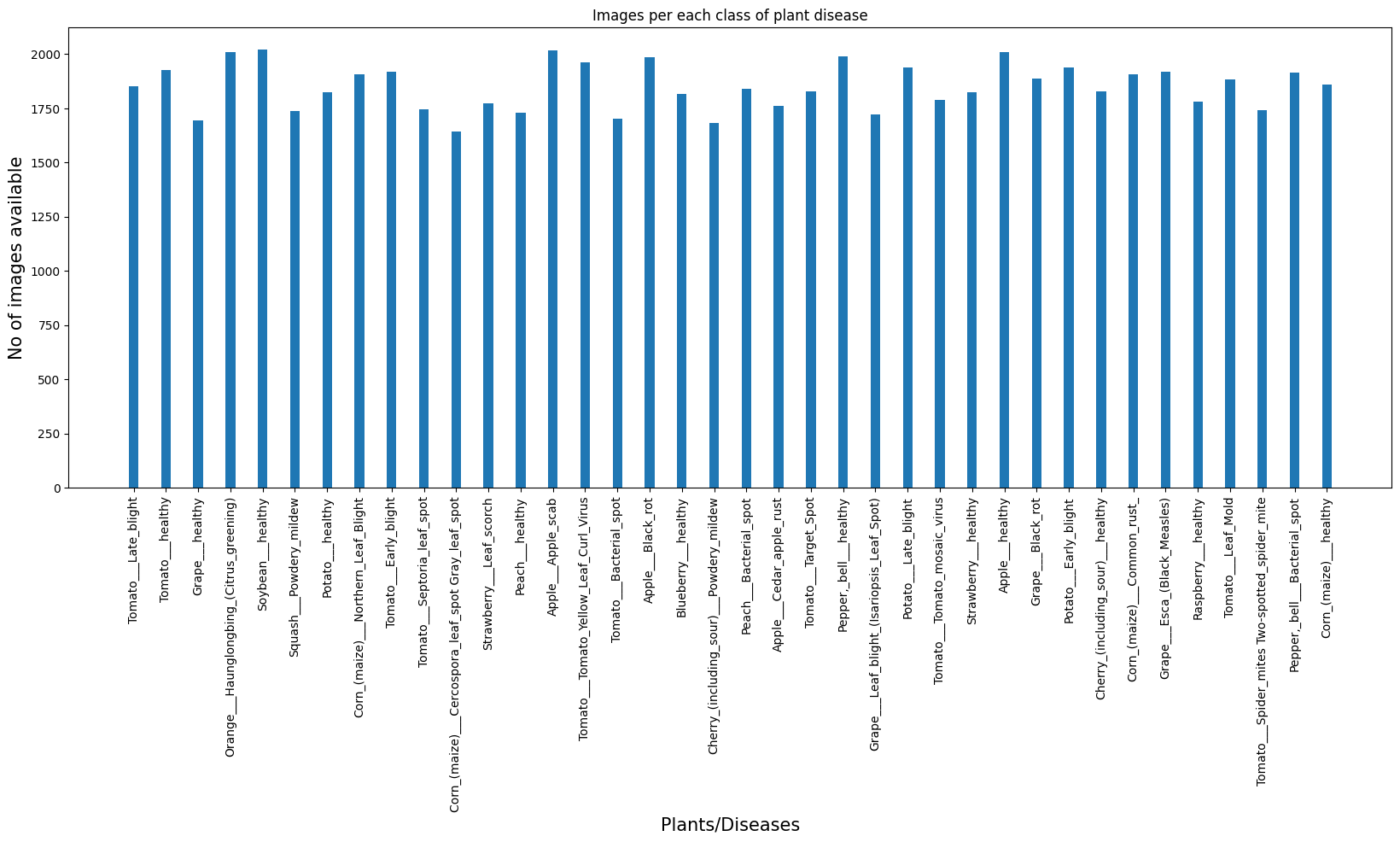
3.2. Exploratory Data Analysis (EDA)

The Exploratory Data Analysis (EDA) phase provided key insights into the dataset, which includes a total of 38 classes representing various plant diseases and healthy states. The dataset comprises 14 unique plant types, including crops like tomato, grape, orange, and corn, with diseases affecting 26 different conditions. The training subset contains 70,295 images, while the validation subset includes 17,572 images, allowing for comprehensive training and evaluation.

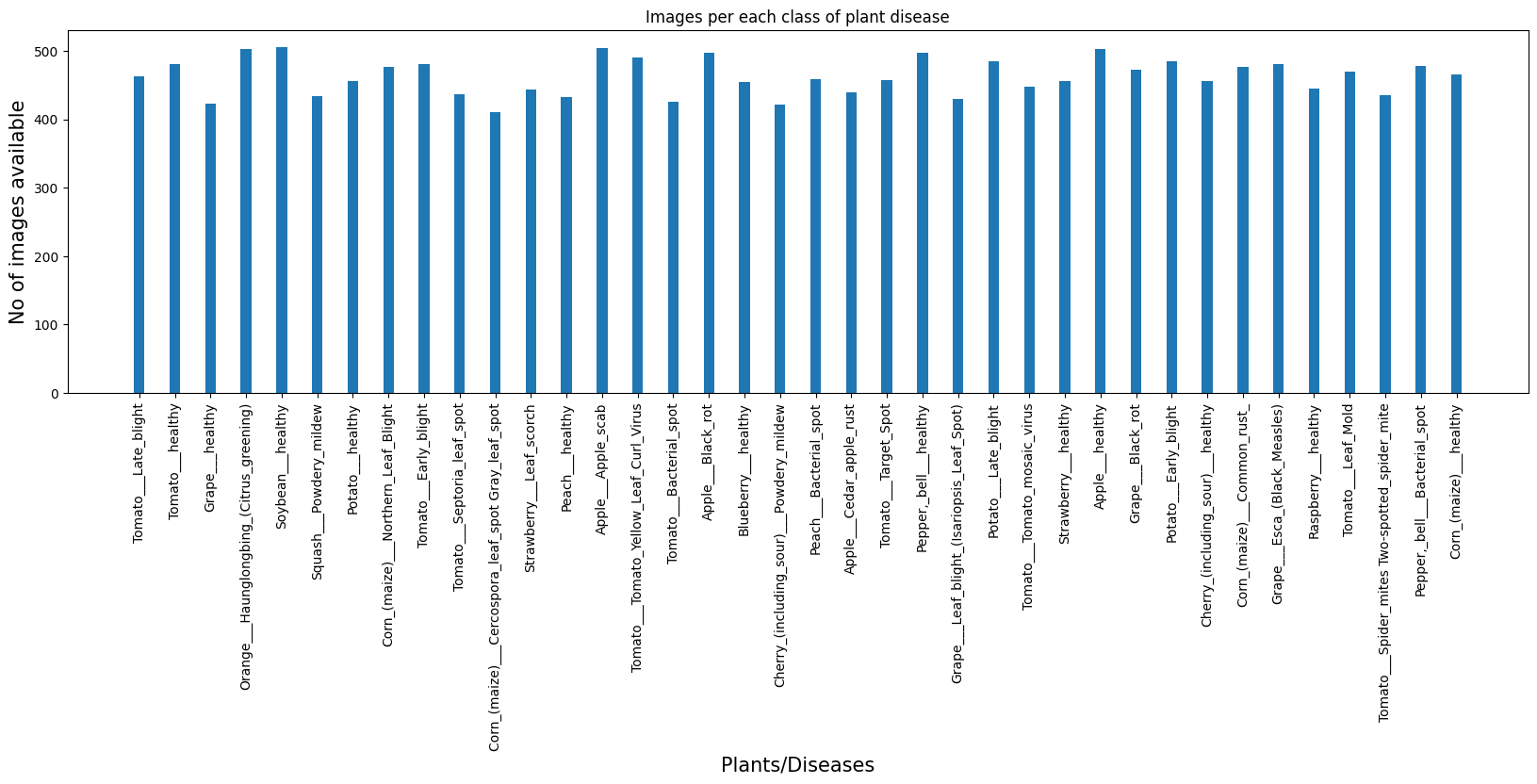
Class distribution analysis confirmed that the dataset is well-balanced across the 38 classes, with a similar number of images representing each disease and healthy condition.



**Distribution of Images per Class in the Training Subset for Plant Disease Detection**



**Distribution of Images per Class in the Validation Subset for Plant Disease Detection**



3.3. Modeling

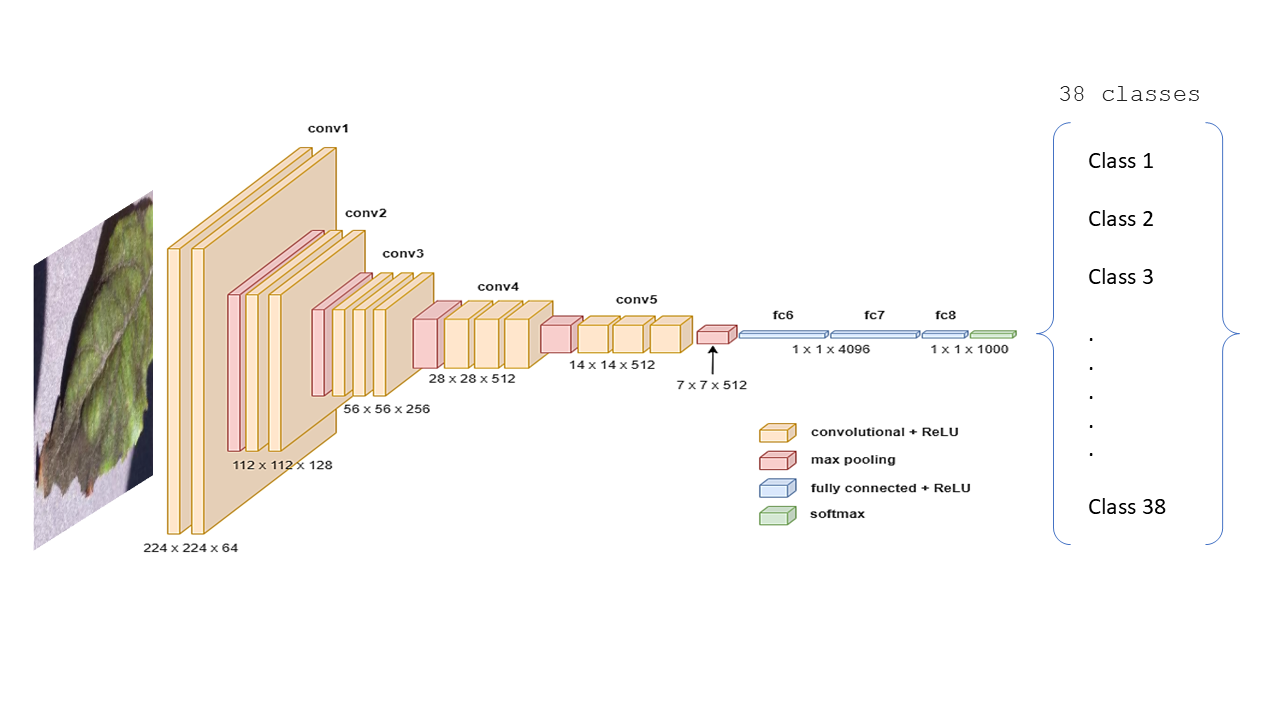
We utilized two models in our approach: one based on transfer learning and the other being our own custom Convolutional Neural Network (CNN) architecture.

**1. Transfer Learning Approach Using VGG16:**

The model employs transfer learning with the VGG16 architecture, pre-trained on ImageNet, to classify plant diseases. The architecture consists of the following key components:

1. **VGG16 Base**:
   1. Loaded without the top layer (include\_top=False) to retain pre-trained feature extraction capabilities.
   2. The input shape is set to (224, 224, 3) for compatibility with the leaf images.
2. **Custom Layers**:
   1. **Global Average Pooling**: Reduces the dimensions of feature maps, summarizing the information.
   2. **Fully Connected Layers**:
      1. Two dense layers with 1024 and 512 units using ReLU activation to refine features.
   3. **Batch Normalization**: Stabilizes learning and improves performance.
   4. **Dropout**: Randomly drops 20% of neurons during training to prevent overfitting.
   5. **Output Layer**: A dense layer with 38 units (one for each disease class) and softmax activation for multi-class classification.
3. **Compilation**:
   1. Compiled with the Adam optimizer and categorical\_crossentropy loss for efficient training on multi-class data.

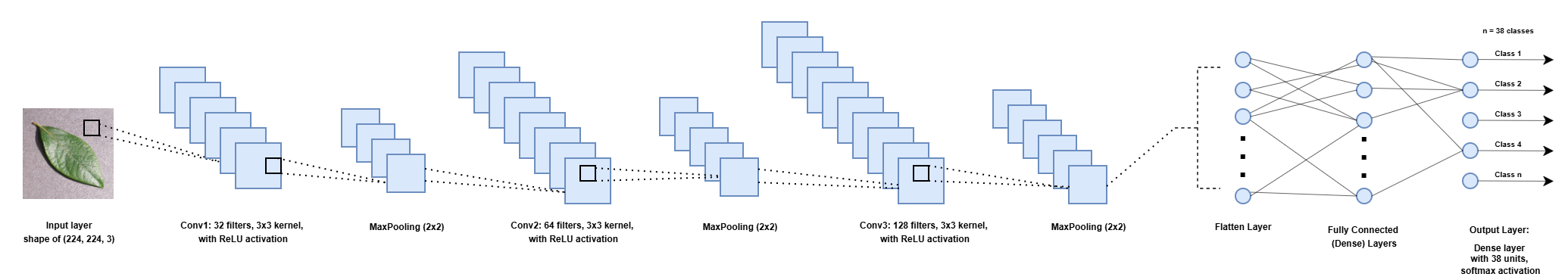
This architecture leverages VGG16’s strengths while customizing it for effective plant disease detection, balancing feature extraction and classification accuracy.



**2. Custom Convolutional Neural Network(CNN):**

The model is constructed using a sequential approach with the following layers designed for plant disease classification:

1. **Input Layer**:
   1. The model starts with an input layer accepting images of shape (224, 224, 3).
2. **Convolutional Layers**:
   1. **First Layer**: 32 filters of size 3x3 with ReLU activation, followed by max pooling to reduce dimensions.
   2. **Second Layer**: 64 filters of size 3x3 with ReLU activation, followed by max pooling.
   3. **Third Layer**: 128 filters of size 3x3 with ReLU activation, followed by max pooling.
3. **Flattening**:
   1. The output from the convolutional layers is flattened into a 1D array to prepare for the fully connected layers.
4. **Fully Connected Layers**:
   1. **First Dense Layer**: 1024 units with ReLU activation, followed by dropout (50%) to prevent overfitting.
   2. **Second Dense Layer**: 512 units with ReLU activation, followed by batch normalization and another dropout (50%).
5. **Output Layer**:
   1. A dense layer with 38 units (for the 38 disease classes) and softmax activation for multi-class classification.
6. **Compilation**:
   1. The model is compiled using the Adam optimizer and categorical cross-entropy loss for efficient training.



Both models were evaluated on a test set. The custom CNN model achieved an accuracy of 53.5%, while the VGG16 model with transfer learning demonstrated a significantly higher accuracy of 83.3%.

3.4. User Interface

For this project, a user interface was not developed. The primary focus was on building and evaluating machine learning models for plant disease detection. Future work could involve developing a web or mobile interface to enable users, such as farmers or agricultural specialists, to upload images and receive real-time disease diagnoses based on the trained model.

3.5. Testing and Improvements

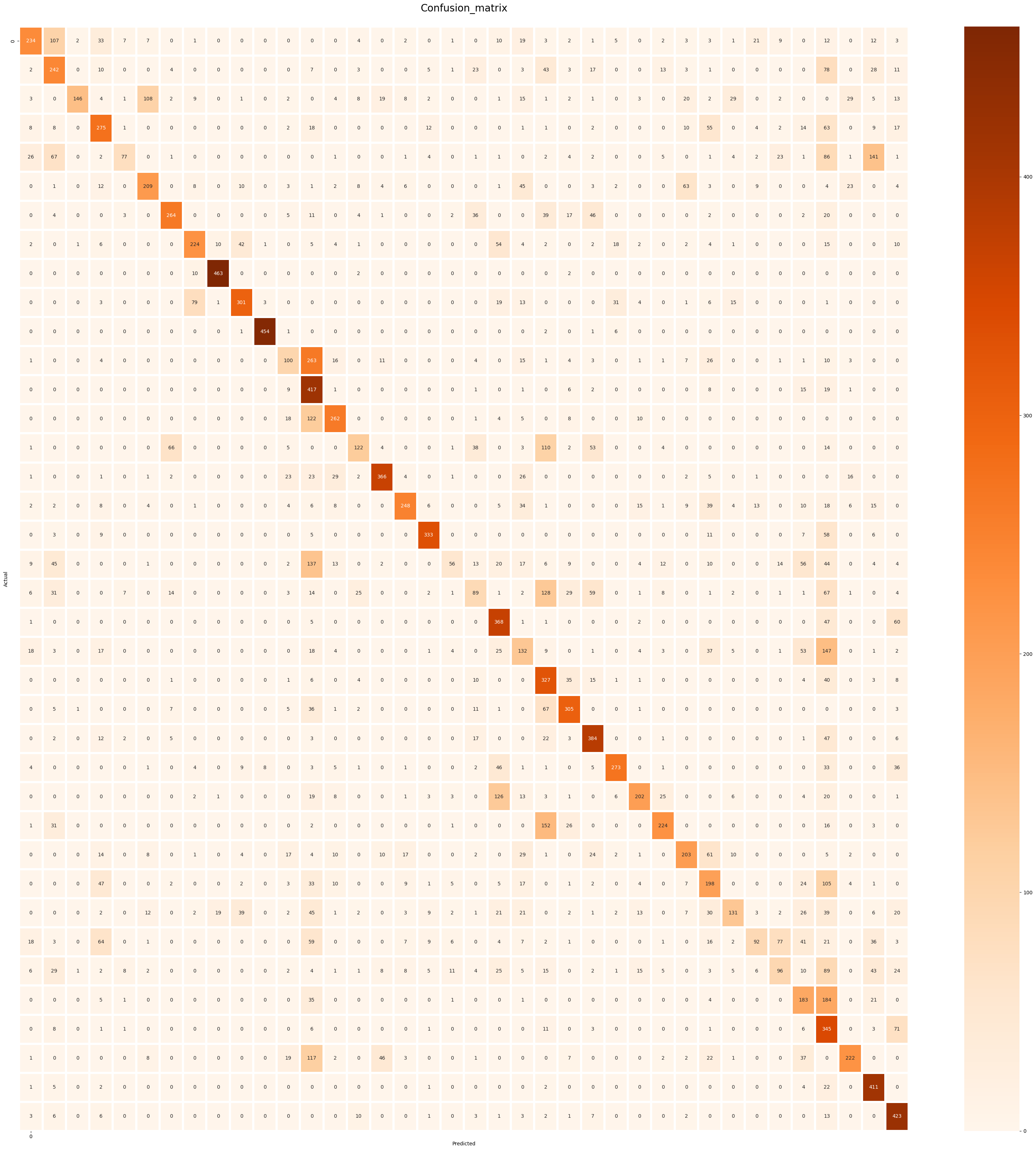
In the testing phase, the Convolutional Neural Network (CNN) achieved an accuracy of 0.535. In contrast, the model using transfer learning with the VGG16 architecture reached a much higher accuracy of 0.833. This improvement is due to VGG16's pre-trained weights from the ImageNet dataset, allowing it to capture more complex features in the leaf images. The transfer learning approach helps the model better distinguish between healthy and diseased leaves compared to the CNN trained from scratch.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Precision** | **Recall** | **F1-Score** | **Accuracy** |
| **Custom CNN** | 0.60 | 0.53 | 0.53 | 0.53 |
| **VGG16** | 0.84 | 0.83 | 0.83 | 0.83 |

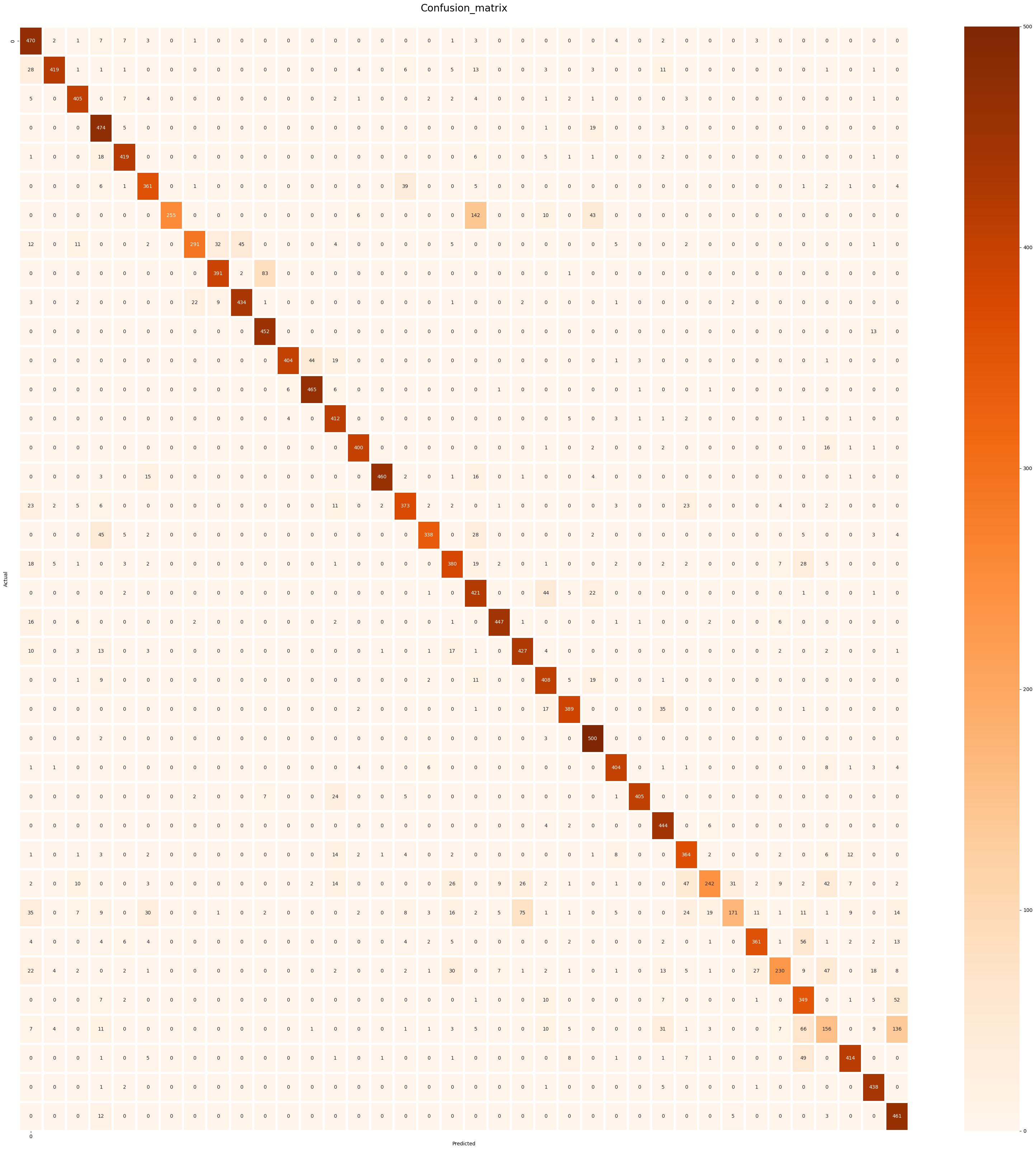
To improve the CNN's performance, future efforts could include hyperparameter tuning such as adjusting learning rates, batch sizes, and dropout rates, which could stabilize training and improve accuracy. Exploring other model architecture variations, like increasing the number of convolutional layers or experimenting with different filter sizes, might also help the CNN capture more complex features in the images, potentially boosting classification accuracy.

The images below display the confusion matrix, providing a visual summary of the model's classification performance across various plant disease categories.

**Custom CNN:**



**VGG16:**



The table below is an illustration of the model's predictions on plant leaf samples, showcasing its disease classification capabilities.

|  |  |
| --- | --- |
| **VGG16** | **Custom CNN** |
|  |  |

4. Projected Impact

4.1. Accomplishments and Benefits

This work has achieved significant milestones in developing a model for early disease detection in crops, aimed at bolstering agricultural productivity and reducing crop losses. Key accomplishments include:

* **Enhanced Model Accuracy**: The model successfully classifies a diverse range of plant diseases using the transfer learning with VGG16, achieving high accuracy on validation and test sets.
* **Increased Efficiency**: This model lowers the need for manual inspections by automating disease detection, allowing for faster response times and limiting disease spread.

These accomplishments collectively provide a strong solution that improves the precision and accessibility of crop disease diagnosis, allowing farmers and stakeholders to make proactive, educated decisions for improved agricultural results.

4.2. Future Improvements

While this project has achieved a strong foundation, further enhancements could improve its scalability, accuracy, and real-world applicability:

* Improving the architecture of the CNN by adding more convolutional and pooling layers or increasing the number of filters in each layer could allow the CNN to capture more complex features within the images. Additionally, experimenting with different kernel sizes and filter configurations could improve feature extraction.
* Extending the dataset to include more crop kinds and disease classes would improve the model's generalizability across various agricultural situations.
* Develop real-time image processing capabilities, enabling farmers to diagnostics via mobile applications.
* Implement a user interface to improve accessibility for farmers in diverse regions.
* Combine the model with IoT-based field data (e.g., temperature and humidity) to account for environmental conditions, improving disease forecasts and providing more comprehensive management recommendations.

These future improvements will help to scale the project into a practical, comprehensive tool for plant disease management.

5. Team Member Review and Comment

|  |
| --- |
| <ATTACH A TEAM PICTURE HERE> |

|  |  |
| --- | --- |
| NAME | REVIEW and COMMENT |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

6. Instructor Review and Comment

|  |  |  |
| --- | --- | --- |
| CATEGORY | SCORE | REVIEW and COMMENT |
| IDEA | \_\_/10 |  |
| APPLICATION | \_\_/30 |  |
| RESULT | \_\_/30 |  |
| PROJECT MANAGEMENT | \_\_/10 |  |
| PRESENTATION & REPORT | \_\_/20 |  |
| TOTAL | \_\_/100 |  |