Data Preparation/Feature Engineering

1. Overview

In this project, data preparation and feature engineering play a crucial role in building a reliable deep learning model for leaf disease detection. High-quality input data is essential to enable convolutional neural networks (CNNs) to learn meaningful patterns rather than noise.
Following the principles outlined in Chapter 10 of ISLP, we focus on organizing, cleaning, augmenting, and transforming the image data to maximize model performance and generalization. Through thoughtful preparation, we ensure that the model can accurately distinguish between healthy leaves and different types of diseases based on visual characteristics.

2. Data Collection

The main sources of data used to train the model are obtained from Mendeley. The data sets are called JMuBEN and JMuBEN2. As a back up three more data sets are planned to be used. The data sets are all obtained from Kaggle, and are called Ethiopian Coffee Leaf Dataset, Coffee-Leaf-Diseases Dataset, and Miner and Rust XML Dataset. These data sets will be used in case the main data set results in unsatisfactory results.

3. Data Cleaning

The original dataset was downloaded from Mendeley Data and organized into five separate folders, each corresponding to a class label (Healthy, Miner, Leaf Rust, Cercospora, Phoma).
During the data cleaning phase, we ensured the integrity and usability of the dataset:

Missing Data:
There were no missing images across the five classes.

Corrupted Images:
We checked all files for corruptions (e.g., unreadable files or improper formats). Only one corrupted image was found and fixed.

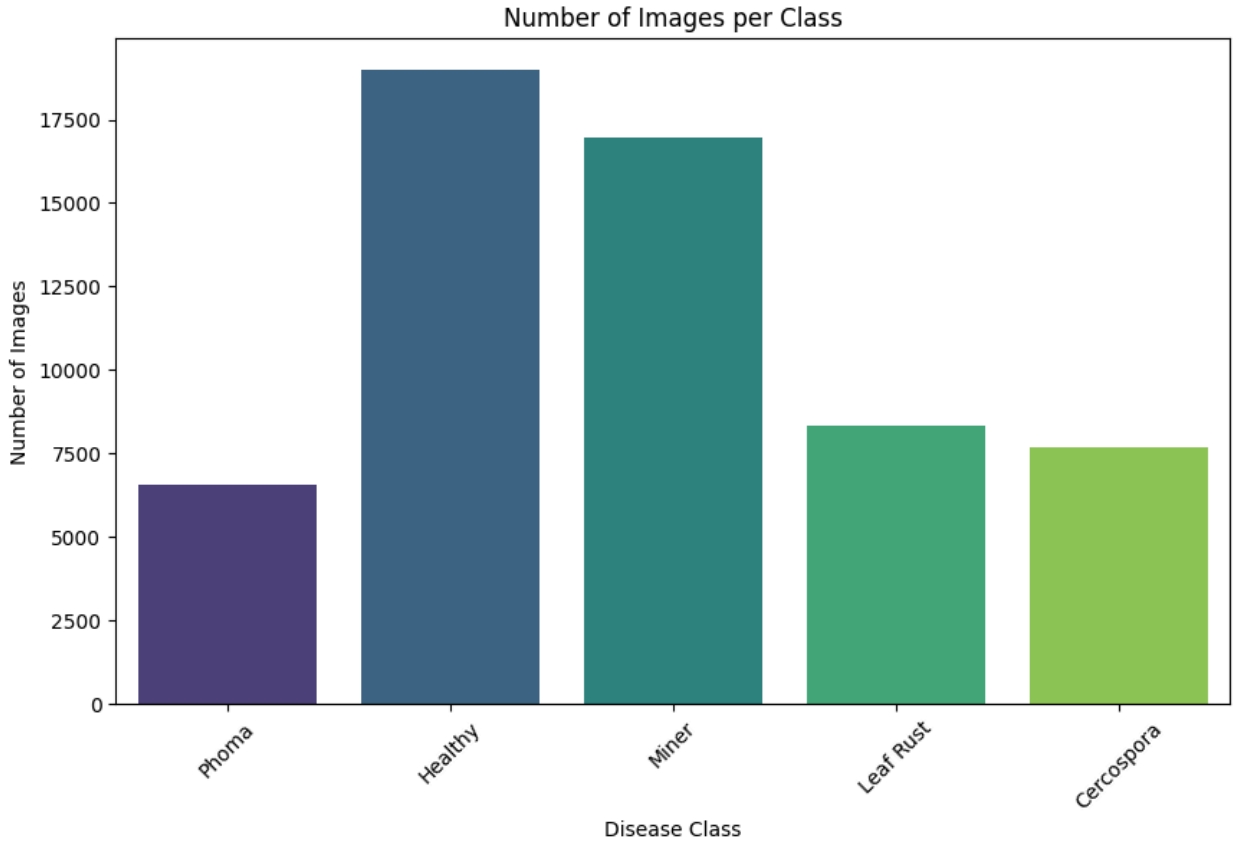
Duplicate Images:
A visual and automated check (using image hashing) was done to identify potential duplicate images. No duplicate was found.

Label Consistency:
Since the folder names correspond directly to the class labels, no relabeling was necessary.

This cleaning process ensured that the CNN model would train on a reliable and consistent set of images.

4. Exploratory Data Analysis (EDA)

Bar Plot: Image counts by class. It shows that there is imbalance in our dataset with there being a lot of healthy leaf images.



Sample Images Grid: 5 random images from each disease class.

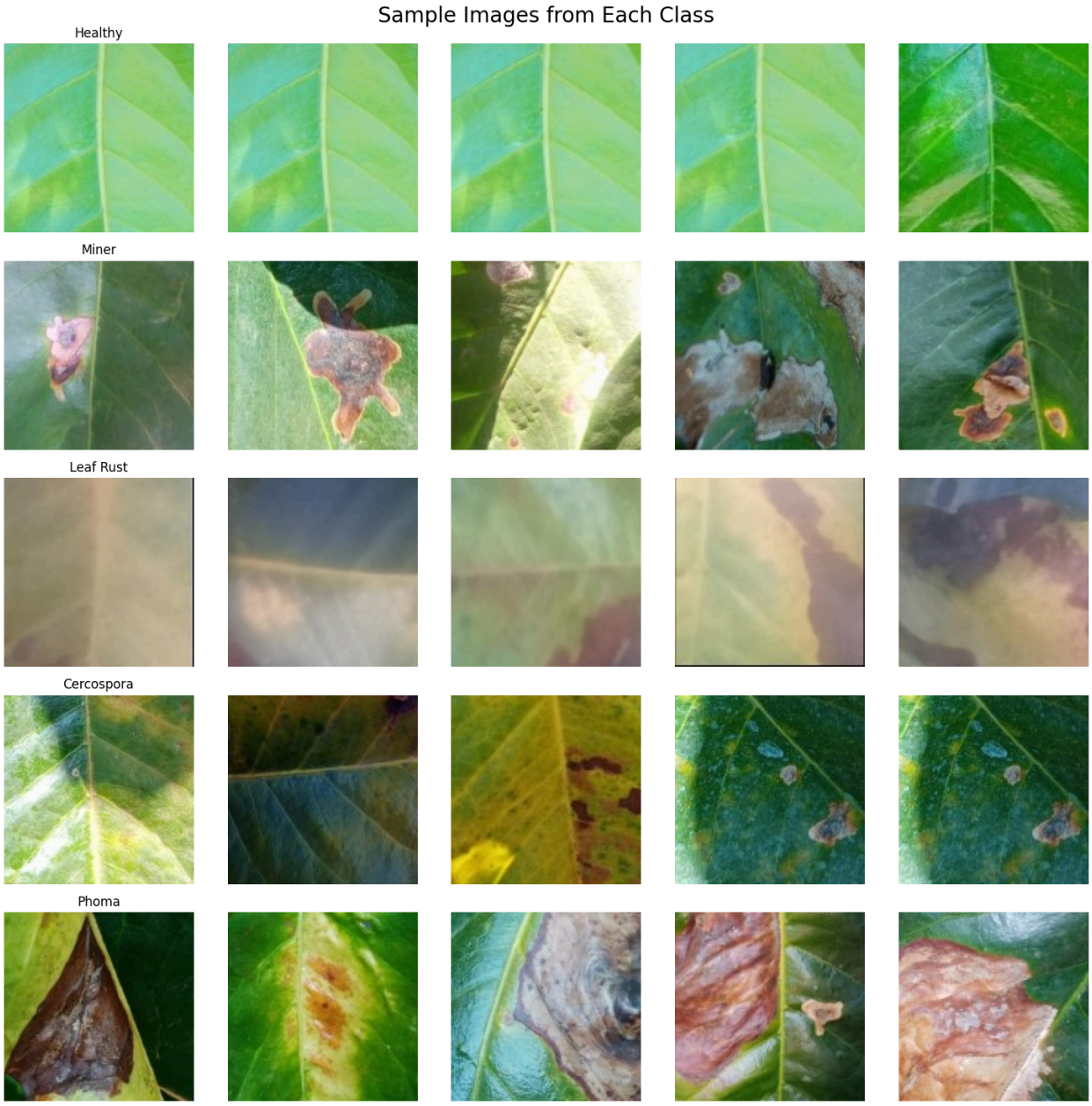
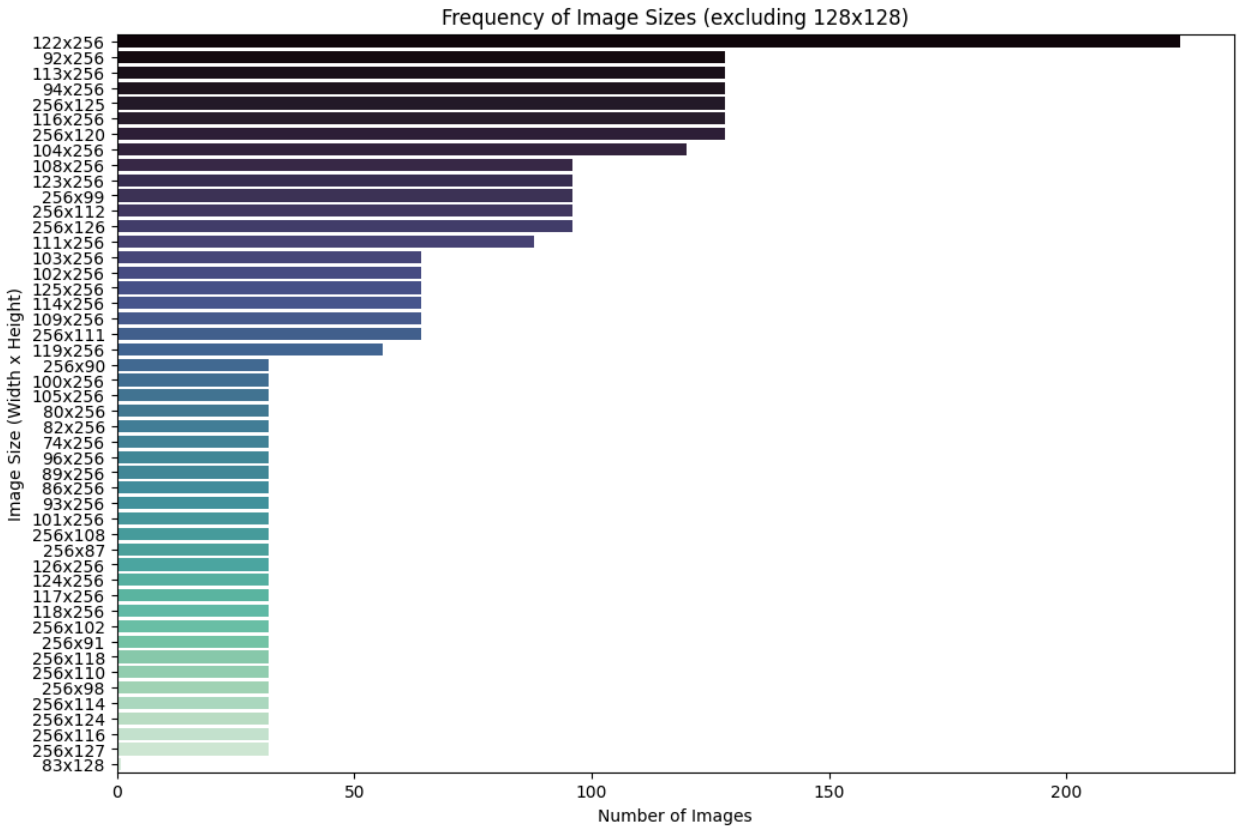


Image size variation:

In the image data set there are 58549 total images. Out of the total, 55596 images have the size (128,128). The rest have the following distribution.



Overall, standardizing images to a common input size (e.g., 224×224 or 256×256) is essential for the model training

5. Feature Engineering

In this project, feature engineering focused on preparing the image data for effective training with a convolutional neural network (CNN).
Following best practices, the following steps were applied:

Image Resizing:
All images were resized to a standard dimension of 224×224 pixels to ensure consistent input size for the CNN.

Normalization:
Pixel values were rescaled from the original range of [0, 255] to [0, 1] by dividing by 255. This normalization step helps the model converge faster and improves training stability.

Data Augmentation:
Additional data augmentation was not performed manually, as the Mendeley Coffee Leaf Disease dataset already includes a variety of augmented samples such as rotated, flipped, and zoomed images

6. Data Transformation

To prepare images for CNN model input, the following transformations were applied:

Image Resizing:
All images were resized to 224×224 pixels to ensure uniform input dimensions.

Normalization:
Pixel values were rescaled from the [0, 255] range to the [0, 1] range to improve model convergence and stability.

Model Exploration

1. Model Selection

Given the nature of the project, which involves classifying images of coffee leaves based on disease symptoms, we selected a Convolutional Neural Network (CNN) model.

CNNs are specifically designed for image-related tasks, leveraging convolutional layers to automatically extract spatial features at multiple levels of abstraction. Following the guidance from Chapter 10 of ISLP, CNNs were chosen due to their proven success in handling large-scale image classification problems such as CIFAR-100 and MNIST datasets.

Strengths of CNNs:

Automatically learn hierarchical features (edges, textures, disease spots)
Require minimal manual feature engineering
Achieve high accuracy when trained on sufficient data
Robust to slight variations (e.g., different lighting, rotation)

Weaknesses of CNNs:
Require significant computational resources (especially deeper networks)
Can overfit if not enough data or regularization
Sensitive to imbalanced datasets without correction

Given the availability of a reasonably large dataset (~58,000 images after cleaning) and the careful handling of class imbalance, CNNs are an appropriate model choice for this leaf disease detection project.

2. Model Training

CNN Architecture:

A custom Convolutional Neural Network (CNN) was designed for this task, consisting of the following layers:

- Multiple convolutional layers with ReLU activation
- Max-pooling layers for downsampling
- Fully connected dense layers at the end
- Softmax activation for multi-class classification

This architecture was inspired by the simple CNN designs discussed in Chapter 10 of ISLP, adapted to handle the complexity of coffee leaf images.

Handling Class Imbalance:

To address the class imbalance observed during exploratory data analysis, class weights were calculated based on the frequency of each class. These weights were incorporated into the loss function during training, ensuring that the model treated minority classes with greater importance, and preventing it from being biased toward the dominant "Healthy" class.

Training Hyperparameters:

- Optimizer: Adam
- Learning rate: 0.001
- Batch size: 32
- Number of epochs: 30 (with early stopping based on validation loss)
- Loss Function: Categorical Cross-Entropy
- Metrics: Accuracy, Precision, Recall
- Validation Strategy:
A validation split of 20% was used to evaluate the model performance during training, ensuring that the model generalizes well to unseen data.
- Early Stopping:
Training was monitored using early stopping with patience set to 5 epochs, to prevent overfitting by halting training when validation loss stopped improving.

3. Model Evaluation

The CNN model was evaluated using standard classification metrics, including accuracy, precision, and recall.
Performance was monitored both during training (using training/validation loss and accuracy curves) and after training (using a confusion matrix).
These evaluations helped identify potential overfitting or bias toward specific classes, ensuring a robust and generalizable model as encouraged by ISLP principles.

4. Code Implementation

Here is a link to the Jupyter notebook used in the model training with all the outputs cleared.

https://colab.research.google.com/drive/1Tfz7zX9O9N_pduO7TX5DP3ZzVbyIkA8D?usp=sharing

Note: Model training is still in progress and developments and improvements are still being done.