VIETNAM ACADEMY OF SCIENCE AND TECHNOLOGY

UNIVERSITY OF SCIENCE AND TECHNOLOGY OF

HANOI

Final Project Report

Machine Learning and Data Mining II

ICT3.011 – B2 DS

Image Classification

Topic: Plant Disease Classification

Hanoi, June 2024

Table of Contents

I. Introduction

1. Motivation

From our daily meals to essential materials, agriculture forms the backbone of our society. But plant disease is a hidden persistent enemy lurking in the fields. These diseases cause massive crop losses, endanger our food supply, and hurt economies.

Traditionally, spotting these diseases relies on experts visually inspecting plants, but this method is slow, inaccurate at times, and often unavailable to small farmers, especially in remote areas.

To address this challenge, this project explores how machine learning (ML) and data mining can be used to automatically classify plant diseases. This is a promising area of data science with the potential to revolutionize agriculture by enabling faster and more accurate disease detection, leading to a more secure agricultural future.

2. Overview

In this project, we investigated how supervised learning algorithms can be leveraged to extract meaningful features from plant images and classify them into different disease categories. The exploration will encompass various data mining techniques for preparing and analyzing datasets of labeled plant images.

We will evaluate the performance of two different Machine Learning models on the classification task: MobileNetV2 and SVM. This analysis will provide insights into the strengths and weaknesses of these algorithms for plant disease detection. Finally, the report will discuss the practical implications of this research and potential future directions for this evolving field within data science.

II. Data Analysis

- Data source: A public dataset called “Plant disease recognition dataset” on the Kaggle website.
- Dataset description: The dataset has three labels for plant conditions: Healthy, Powdery, and Rust (which are “labels”). It has a total of 1532 leaf photos, separated into three sets: training, testing, and validation (which are “categories”).
- Statistics analysis of the data

	Train	Validation	Test
Healthy	458	20	50
Rust	434	20	50
Powdery	430	20	50
Total	**1322**	**60**	**150**

Table 1: Number of instances in the data

Figure 1.1: Distribution by category

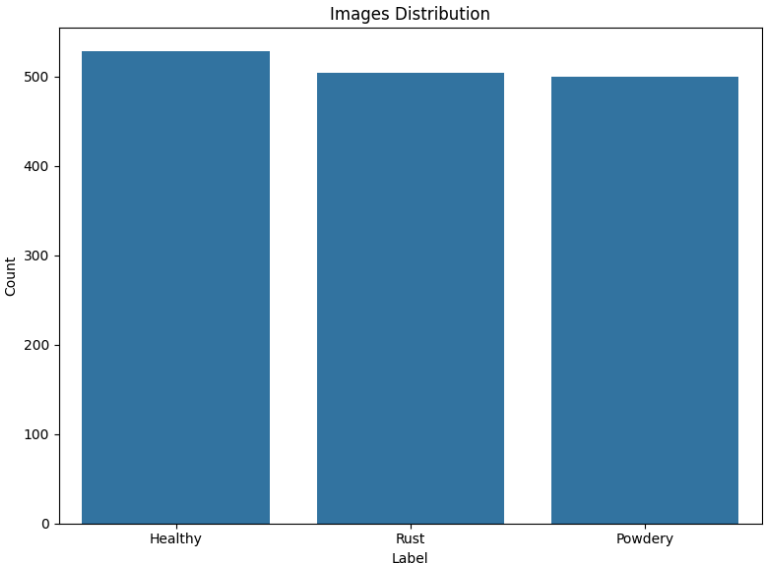

Figure 1.2: Distribution by label

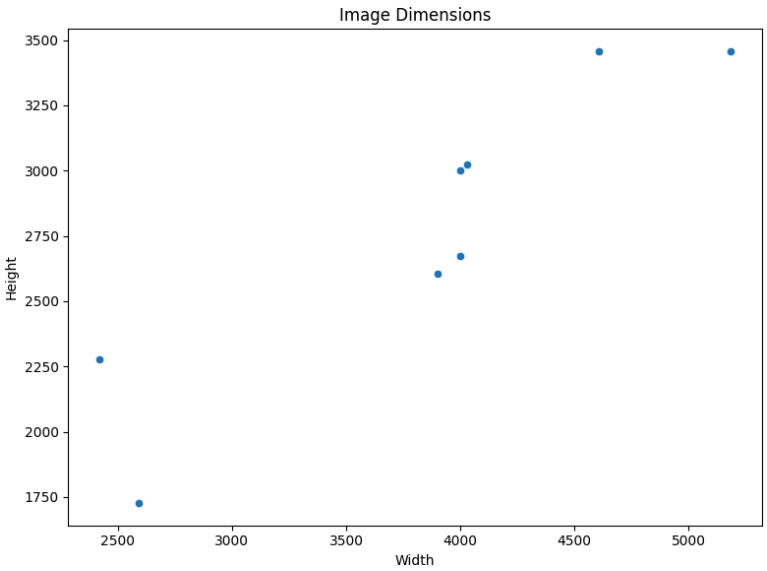

Figure 1.3: Distribution of image dimensions

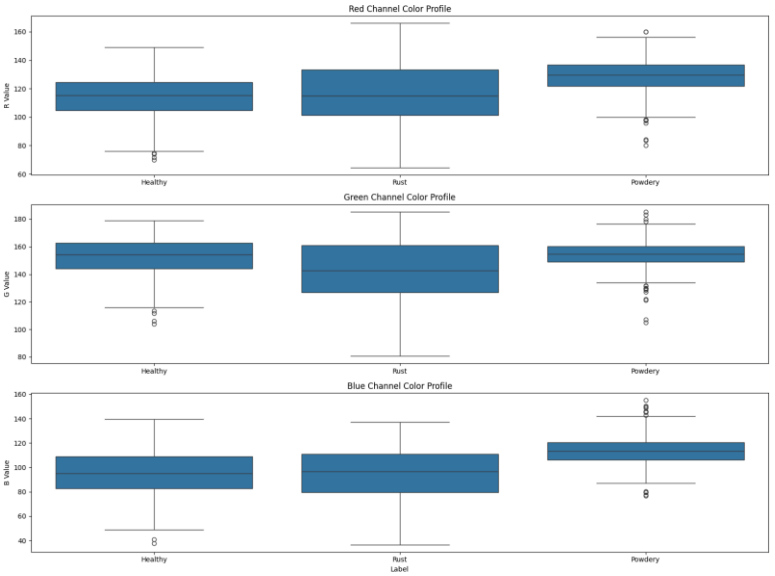

Figure 1.4: Distribution of color profiles (R, G, B)

Figure 1: Data analysis of the images in the dataset

The data shows a balanced distribution across disease labels (Fig 1.2), while the converse is true for categories (Fig 1.1). All photos are RBG (Fig 1.4) with close dimensions (Fig 1.3), simplifying processing and analyzing. Besides, “Powdery” leaves in Fig 1.4 show more outliers in color channels, which could be a potential disease indicator.

III. Project process

This section will detail the stages of building the models. Our project has 5 parts: Problem Definition, Model Selection, Data Preprocessing, Model Training, and Model Evaluation.

1. Problem Definition

- Objective: Classify a disease based on a plant image.
- Target accuracy: Achieve an accuracy of at least 90%.

2. Model Selection

- Models chosen: MobileNetV2 and SVM.
- Reason for selection:
    - MobileNetV2: It is a lightweight and efficient model pre-trained on ImageNet, which provides a strong starting point with rich feature extraction capabilities while being computationally less demanding.
    - SVM: It is effective in high dimensions, resistant to overfitting, and capable of handling non-linear data via the kernel trick; therefore, it can efficiently classify high-dimensional features extracted by a pre-trained neural network, potentially achieving high accuracy.

3. Data Preprocessing

Firstly, we iterate through each dataset type and its subdirectories corresponding to labels. Then we open the image and resize it to 128x128 pixels. Finally, we applied specific image filtering. In this project, we use distinct preprocessing methods for the two models to optimize their effectiveness.

Let's take a look at the difference between the methods.

- For MobileNetV2:
    - Convert to RGB if not already.
    - Convert to a numpy array.
    - Save the pre-processed image to the output directory.
- For SVM
    - Convert a PIL Image or NumPy array to a PyTorch tensor and scale pixel values to the range [0, 1].
    - Normalization: Images were normalized using the mean and standard deviation values of [0.485, 0.456, 0.406] and [0.229, 0.224, 0.225], respectively.

4. Model Training

4.1. MobileNetV2

Model Architecture

To leverage pre-trained knowledge, we used MobileNetV2, a model trained on a massive image dataset (ImageNet). This efficient model utilizes depth-wise separable convolutions for faster processing. We then customized MobileNetV2 by removing its top layers and adding new ones specific to our plant disease classification task.

- Global Average Pooling Layer: Added to reduce the dimensionality of the output.
- Dense Layer: Introduced with 128 units and ReLU activation to add non-linearity.
- Output Layer: A final Dense layer with softmax activation to output class probabilities corresponding to the number of categories in the dataset.

Training Process

The training process was conducted in two main phases: Initial Training and Fine-Tuning. We detailed these phases below.

- **Initial Training**
    - Frozen for stability: MobileNetV2's pre-trained layers were frozen, preserving their general image recognition knowledge gained from ImageNet.
    - Focused training: Only the newly added layers were trainable, allowing the model to learn the intricacies of our plant disease data.
    - Standard training techniques: This phase involved training for 10 epochs using the Adam optimizer with the categorical cross-entropy loss function.
- **Fine-Tuning**
    - Unfreezing layers: The last 20 layers of the MobileNetV2 base model were unfrozen for fine-tuning, allowing them to adapt to plant disease data's specific features.
    - Gradual adaptation: The lower learning rate prevented significant changes to the pre-trained layers, safeguarding their valuable image recognition expertise.

Model Saving and History

The final trained model, which was saved in a specified format, and its training history (including performance metrics on both training and validation data) were saved for future use and analysis. This data allows us to assess the training process, identify potential issues, and compare future models.

4.2. SVM

Model Architecture

- Base model: We utilized a pre-trained ResNet-18 model as the foundation. Its extensive training on the diverse ImageNet dataset provided a strong base for feature extraction relevant to image recognition. The architecture of ResNet-18 includes 18 layers with residual connections, which help mitigate the vanishing gradient problem and improve model performance.

- Feature Extraction: We removed the final fully connected layer of the ResNet-18, designed for general image classification on ImageNet. Instead, we leveraged the rich feature representations learned by the convolutional layers in the model for our specific classification task.

Training Process

The training was divided into two major sections: Initial Training and Hyperparameter Tuning.

- **Initial Training:**
    - Feature Extraction: The pre-trained ResNet-18 model was used to extract features from the images.
    - SVM Classifier: An SVM classifier was trained using these features.
- **Hyperparameter Tuning:**
    - A grid search was performed to identify the optimal hyperparameters for the SVM classifier.
    - Parameters tuned included cost (C), gamma (influencing the influence of training points), and kernel (defining how data points are mapped in higher dimensions).

Model Saving and History

- Saving the best model The best SVM model was saved using 'joblib' library for future use.
- Training History Documentation, including accuracy and classification reports, was documented to facilitate analysis and future comparisons.

5. Model Evaluation

- After training, the models' performances were evaluated using the test set.
- The predictions on the test set were compared with the true labels to generate a classification report, which included precision, recall, and F1-score for each class.
- This provided a comprehensive view of the models' accuracy and their ability to generalize to unseen data.

IV. Experiments

We used the two models to test their abilities to classify plant diseases. The models performed well on the plant disease classification task since they can identify plant diseases correctly.

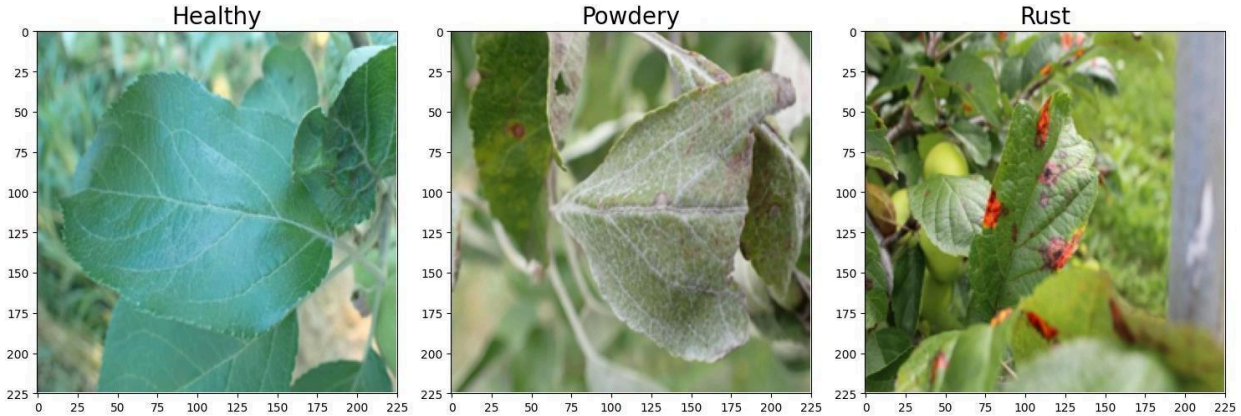

Figure 2: Testing results of MobileNetV2

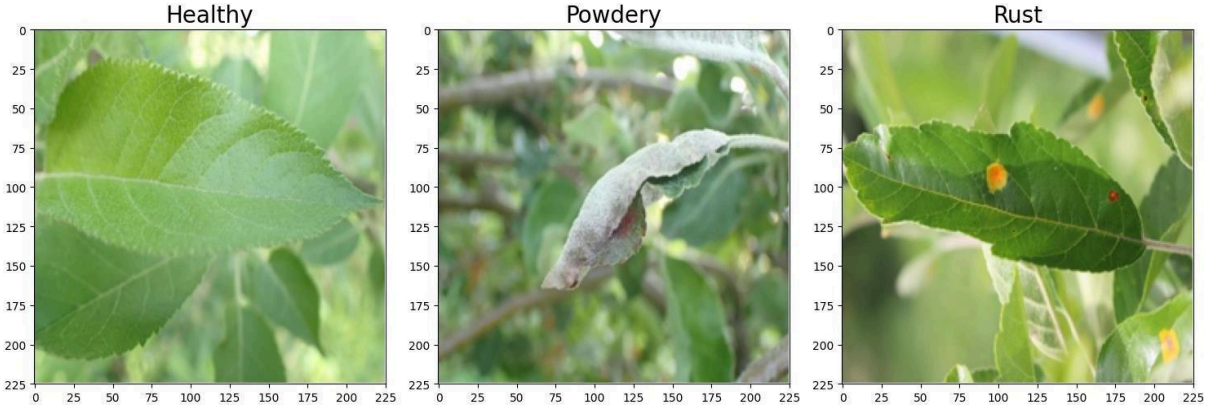

Figure 3: Testing results of SVM

V. Evaluation

1. MobileNetV2

	precision	recall	f1-score	support
Healthy	0.86	1.00	0.93	50
Powdery	1.00	0.90	0.95	50
Rust	1.00	0.94	0.97	50
accuracy			0.95	150
macro avg	0.95	0.95	0.95	150
weighted avg	0.95	0.95	0.95	150

Table 2: Classification report on MobilenetV2 model's running

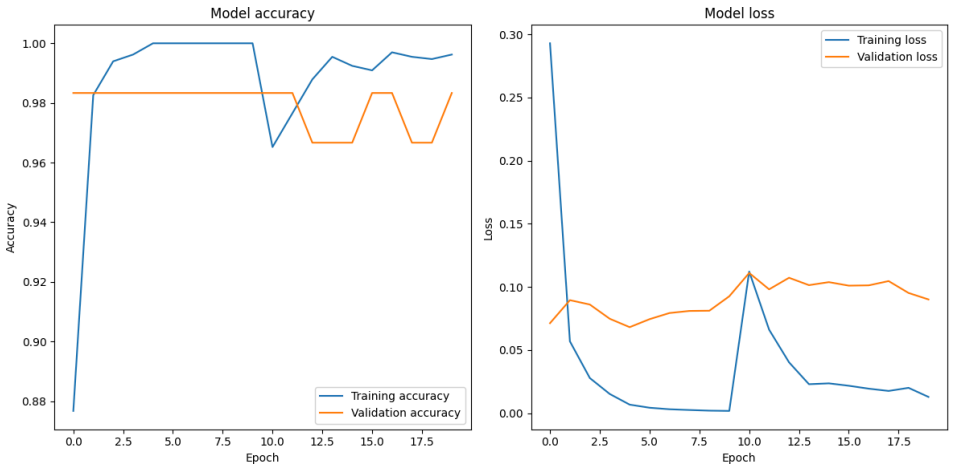

Figure 4: Training and validation accuracy and loss visualization

2. SVM

	precision	recall	f1-score	support
Healthy	0.89	0.96	0.92	50
Powdery	1.00	0.90	0.95	50
Rust	0.94	0.96	0.95	50
accuracy			0.94	150
macro avg	0.94	0.94	0.90	150
weighted avg	0.94	0.94	0.94	150

Table 3: Classification report on SVM model's running

VI. Comparison and Conclusion

We compared the performance metrics of two models. The observed result shows that both models have accuracy at a high level, with around 95% of MobileNetV2 and around 94% of SVM. The MobileNetV2 is a little bit more effective than the SVM. These results prove that ML models can be applied to detect plant diseases, contributing to agriculture and improving human life.

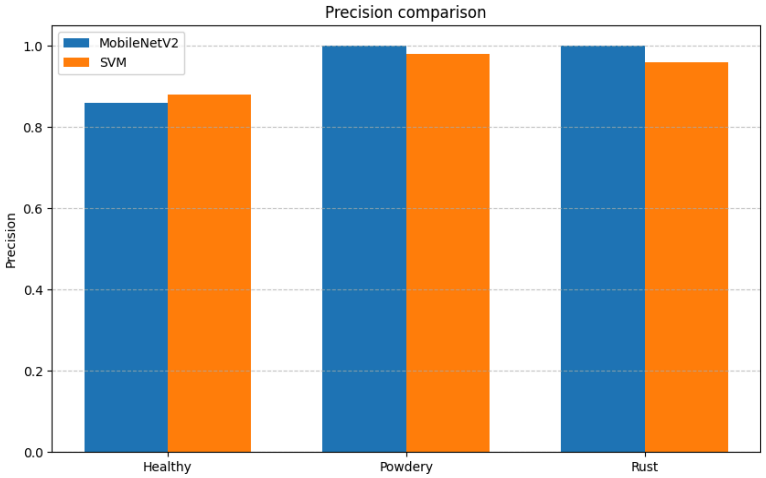

Figure 5.1: Precision

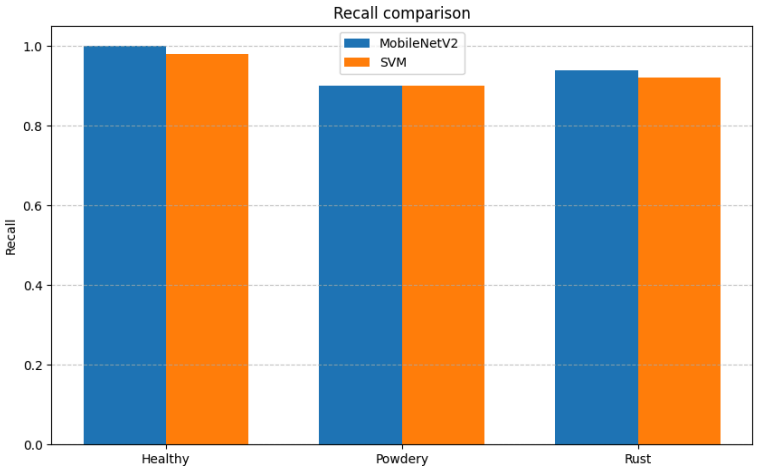

Figure 5.2: Recall

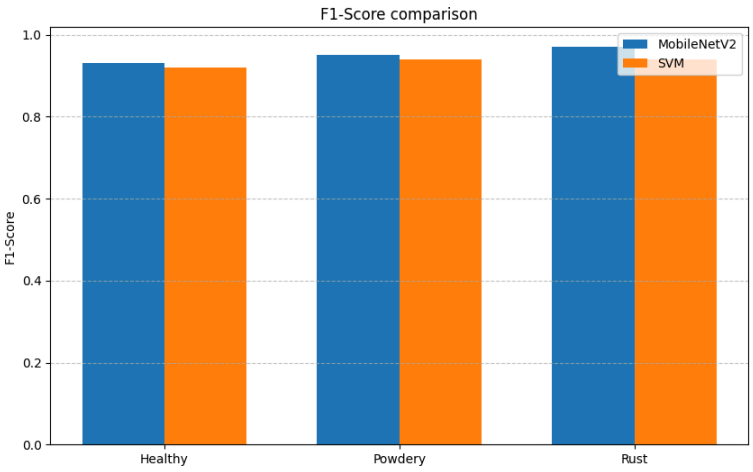

Figure 5.3: F1-score

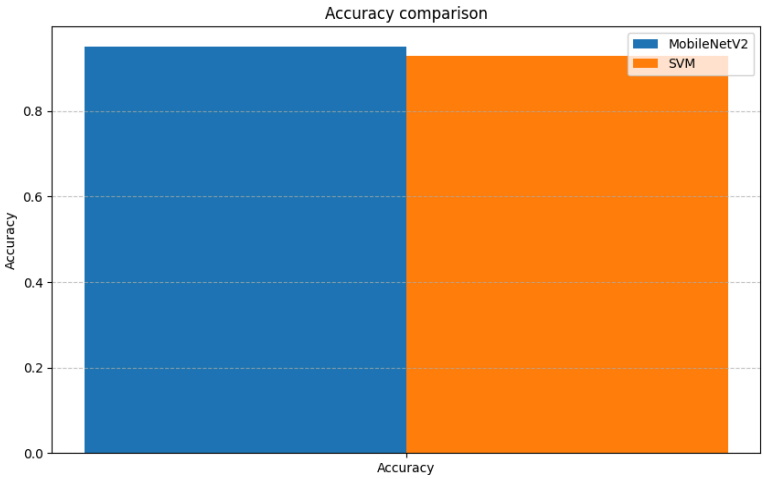

Figure 5.4: Accuracy

Figure 5: Comparisons of the MobileNetV2 and SVM models' performance metrics

VII. Reference list

www.kaggle.com. (n.d.). *Plant disease recognition dataset*. [online] Available at: https://www.kaggle.com/datasets/rashikrahmanpritom/plant-disease-recognition-dataset/ [Accessed 30 May 2024].