Plant leaf disease classification

1. Introduction

Plant leaf diseases are one of the important reasons for the loss in the production and plant leaf disease identification is also very difficult in the agriculture field. The traditional method of identifying the diseases is by human observation which involves huge man power, inaccurate, time consuming and not applicable for larger fields[1]. On the other hand, using deep learning, a reliable prediction methodology is used for detecting various diseases of plant leaves caused by fungus, bacteria and virus[2]. However, disease prediction using classification algorithms appears to be a difficult task as the accuracy varies for different input data. In the current project, the classification is transfer learning with MobileNetV2 algorithm.

The current project explores how deep learning techniques can be utilised to automatically classify plant leaves as healthy, powdery or rust.

We use the ‘Plant Disease Recognition Dataset’ from Kaggle, and train a customised MobileNetV2 model to accurately identify various leaf diseases, for early detection of plant leaf diseases.

The prediction best model accuracy of the current model is 0.845

2. Related Work

Several studies have leveraged machine learning and deep learning approaches to address the problem of detecting plant leaf disease, primarily to improve speed, accuracy, and reliability over manual inspection methods.

1. L. Sherly Puspha Annabel et al. (2019) reviewed and summarised various techniques used for classifying and detecting various bacterial, fungal and viral plant leaf diseases. The classification techniques helped in automating the detection of plant leaf diseases and categorising them centered on their morphological features.
2. Yousef Methkal Abd Algani et al. (2023) Early identification of leaf diseases is essential to the agricultural industry. Here is the basic concept of plant leaf infection identification and plant leaf infection symptoms.For test real-time images for leaf disease identification the traditional method has been used. The proposed method can provide provision for farmers to detect and recognize plant leaf diseases.Here the ACO-CNN optimization approach is proposed for leaf disease detection. ACO (Ant Colony Optimization) used for the feature extraction and CNN classifier was used.
3. Chittabarni Sarkar et al (2023) have performed an in-depth study of this topic from 2010 to 2022 and made a workflow mechanism to help researchers in this field. Support vector machine (SVM), Random Forest, and multiple twin SVM (MTSVM) are popular ML models for predicting leaf disease, while convolutional neural networks (CNN), visual geometry group (VGG), ResNet (RNet), GoogLeNet, deep

CNN (DCNN), back propagation neural networks (BPNN), DenseNet (DNet), LeafNet (LN), and LeNet are common deep learning models used for detecting leaf disease.They concluded that models like CNN, VGG, and ResNet are highly capable at finding diseases in leaves. The performance of the algorithms is generally evaluated using F1 score, precision, accuracy and others.

3. Materials and Experimental Evaluation

3.1 Dataset

The dataset used here is “Plant DIsease Recognition Dataset” from Kaggle. This dataset contains three classes namely, "Healthy", "Powdery", "Rust" referring to plant conditions. There is a total of 1532 images divided into train, test, and validation sets. As this is a basic level experiment, the dataset is used as it is. Among the 1532 images, we have taken 1322 images for train and validation sets. The three classes are almost equally distributed. The “Healthy” class makes up to 35%, the “Powdery” class to 32% and the “Rust” class covers the remaining 33%. We used 80% of the data for training and 20% for Validation

3.2 Methodology

The experiment tests the hypothesis that deep learning models (CNNs), especially those utilising transfer learning with preprocessing methods, can accurately detect and classify plant leaves.

Evaluation Metrics:
The performance of the model is evaluated based on the following metrics:

1. Accuracy: Measured on both training and validation datasets.
2. Loss: Training and validation losses are tracked throughout the epochs.
3. Confusion Matrix: Used to evaluate the classification performance by analysing true positive (TP), true negative (TN), false positive (FP), and false negative (FN) rates.
4. Classification Report:
   a. Precision: It measures how many of a model's positive classifications are actually correct.
   b. Recall: It measures how well a machine learning model can identify all relevant cases in a data set.
   c. F1 Score : A performance metric that balances precision and recall.
   d. Support: It is the number of actual occurrences of a class in a dataset.
5. Best Model Accuracy: Captured based on validation accuracy with the lowest validation loss.

Experimental Methodology:
In this study, the methodology revolves around training and validating a Convolutional Neural Network (CNN) using the MobileNetV2 model pre-trained on the Plant Disease Recognition dataset. The experiment was conducted as follows:

1. Dataset:
   A set of 1,322 images from the Plant Disease Recognition Dataset is used, split into three classes: healthy, powdery and rust leaves. The dataset was divided into training (80%) and validation (20%) sets using the ImageDataGenerator.
2. Preprocessing:
   Images were preprocessed using the preprocess_input function from MobileNetV2, which normalised the images according to the model's input requirements.
3. Model Architecture:
   A pre-trained MobileNetV2 was utilised as the base model with non-trainable layers.
   A custom classifier head was added:
   - GlobalAveragePooling2D layer
   - Dense layer (100 units with ReLU activation)
   - Final softmax layer with three output nodes
   - Optimizer:
     The model was trained using the Adam optimizer with a learning rate of 0.0001.
4. Training and Validation:
   The model was trained for 30 epochs with a batch size of 32. Early stopping was implemented to prevent overfitting, restoring the best model weights based on validation loss.
5. Performance Metrics:
   - Accuracy and loss values were recorded at each epoch.
   - Confusion matrix and classification report were generated using validation data, and performance was measured based on precision, recall, and F1 score.
   - Best model accuracy was tracked based on the epoch with the lowest validation loss.

Training/Test Data:

- The training data comprised 80**%** of the 1322 images (approx. 1058 images), while the remaining 20**%** (approx. 264 images) was used for validation.

Performance Data:
The performance data collected during the training process includes:

1. Training and validation accuracy: Monitored throughout the epochs.
2. Training and validation loss: Used to evaluate overfitting or underfitting.
3. Confusion matrix: Shows the distribution of correct and incorrect classifications across both classes.
4. Best model accuracy: Captured when the validation loss reached its minimum value.

Performance analysis involved:

- Epoch-wise comparison: A comparison of model performance across different learning rates (ranging from 0.000001 to 0.05) and epoch counts (up to 50 epochs).
- Model evaluation: Based on the confusion matrix, the model’s ability to correctly classify healthy, powdery and rust leaf images was scrutinised. For instance, the model showed strong accuracy (F1 score: 83% after 30 epochs).

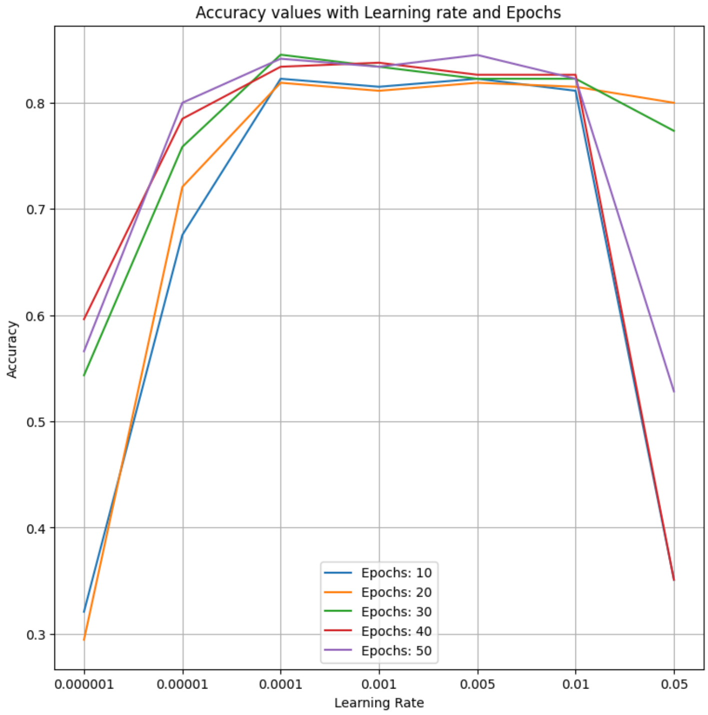

Figure 1 : Multiline Plot

3.3 Results

Confusion Matrix: The confusion matrix for the grayscale model shows excellent performance, with True Positives (TP) and True Negatives (TN) being dominant compared to False Positives (FP) and False Negatives (FN). The matrix provides the distribution of correct vs incorrect classifications for each class (healthy, powdery and rust).

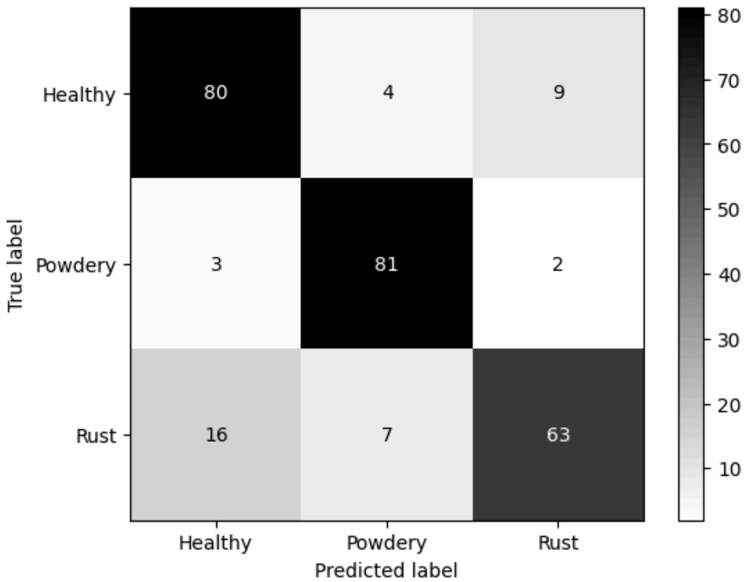

Figure 2: Confusion Matrix

- In the above matrix the total number of predictions are 265, out of which 99 times predicted Healthy, 92 times predicted as Powdery and 74 times as Rust.
- However, in reality, 93 leaves were Healthy, 86 were Powdery and 88 leaves were Rust.

Classification Report: The classification report provides precision, recall, and F1 scores for both classes:

- There are four ways to check if the predictions are right or wrong
  - TN / True Negative: the case was negative and predicted negative
  - TP / True Positive: the case was positive and predicted positive
  - FN / False Negative: the case was positive but predicted negative
  - FP / False Positive: the case was negative but predicted positive
- **Precision — *What percent of your predictions were correct?***
  - Precision is the ability of a classifier not to label an instance positive that is actually negative. For each class, it is defined as the ratio of true positives to the sum of a true positive and false positive.
  - Precision:- Accuracy of positive predictions.
  - Precision = TP/(TP + FP)
- **Recall — *What percent of the positive cases did you catch?***
  - Recall is the ability of a classifier to find all positive instances. For each class it is defined as the ratio of true positives to the sum of true positives and false negatives.
  - Recall:- Fraction of positives that were correctly identified.
  - Recall = TP/(TP+FN)
- **F1 score — *What percent of positive predictions were correct?***
  - The F1 score is a weighted harmonic mean of precision and recall such that the best score is 1.0 and the worst is 0.0. F1 scores are lower than accuracy measures as they embed precision and recall into their computation. As a rule of thumb, the weighted average of F1 should be used to compare classifier models, not global accuracy.
  - F1 Score = 2*(Recall * Precision) / (Recall + Precision)
- **Support**
  - Support is the number of actual occurrences of the class in the specified dataset. Imbalanced support in the training data may indicate structural weaknesses in the reported scores of the classifier and could indicate the

need for stratified sampling or rebalancing. Support doesn’t change between models but instead diagnoses the evaluation process.

```
Classification Report :
              precision    recall  f1-score   support

           0       0.81      0.86      0.83        93
           1       0.88      0.94      0.91        86
           2       0.85      0.73      0.79        86

    accuracy                           0.85       265
   macro avg       0.85      0.84      0.84       265
weighted avg       0.85      0.85      0.84       265
```

Figure 3: Classification Report

3.4 Discussion

The best model accuracy is 0.8453, the predictions accuracy is around 85%. But, as the confusion matrix shows, 20 Healthy leaves were predicted as Rust, the prediction needs to be improved.The number of images for the training set have to be enhanced so as to improve the results.

What conclusions do the results support about the strengths and weaknesses of your method compared to other methods? How can the results be explained in terms of the underlying properties of the algorithm and/or the data.

Prediction for healthy leaf

```
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 75ms/step
[[0.81363046 0.10061464 0.08575483]]
```

Prediction for powdery leaf

```
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 53ms/step
[[0.00697576 0.93410826 0.05891594]]
```

Prediction for rust leaf

```
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 51ms/step
[[0.01154236 0.05199663 0.93646103]]
```

4. Future Work

As discussed earlier, the dataset needs to be enhanced for getting better accuracy. The present work for leaf disease classification is more general covering different leaves. It can be implemented to a specific plant or group of plants.

5. Conclusion

This project presented a deep learning approach using MobileNetV2 to classify leaves as healthy, powdery and rust. Key findings include:

- The model achieved strong validation accuracy while avoiding overfitting through early stopping. The Adam optimizer with a learning rate of 0.0001 yielded the best results, though other optimizers offered insights into hyperparameter tuning.
- The use of precision, recall, and F1 scores provided a comprehensive evaluation of the model's ability to classify leaves, with best performance were shown in 30 epochs
- The successful application of transfer learning for a practical leaf disease problem, demonstrating that pre-trained models like MobileNetV2 can be effective leaf disease detection.

6.Reference

1. L. Sherly Puspha Annabel, T. Annapoorani and P. Deepalakshmi. Machine Learning for Plant Leaf Disease Detection and Classification – A Review. https://ieeexplore.ieee.org/document/8698004

2. Yousef Methkal Abd Algani, Orlando Juan Marquez Caro b, Liz Maribel Robladillo Bravo,Chamandeep Kaur, Mohammed Saleh Al Ansari, B. Kiran Bala - Leaf disease identification and classification using optimised deep learning. https://www.sciencedirect.com/science/article/pii/S266591742200277X

3. Chittabarni Sarkar, Deepak Gupta, Umesh Gupta, Barenya Bikash Hazarika. Leaf disease detection using machine learning and deep learning: Review and challenges. https://www.sciencedirect.com/science/article/abs/pii/S1568494623005525

4. Dataset Source: Plant Disease Recognition Dataset from Kaggle https://www.kaggle.com/datasets/rashikrahmanpritom/plant-disease-recognition-dataset