

# Detecting Plant Diseases: A Comprehensive Approach to Houseplant Health

Bill Song - b26song@uwaterloo.ca  
Raymond Liu - r354liu@uwaterloo.ca

October 8, 2024

## 1 Overview

House plants have become an integrated part of many households, providing many benefits such as enhancing indoor air quality and lowering stress. However, house plants are vulnerable to many diseases, and many people lack the knowledge to identify the health issues of their plants. This project addresses this issue by designing a machine learning model that analyzes photos of house plants and identifies the specific type of disease affecting this house plant (or that it is healthy). While deep learning has been widely used for plant disease detection, most existing efforts focus on agricultural crops. This project, however, is specifically tailored to house plants, addressing a gap in current research. Due to the large number of plant diseases, we plan to focus on three common diseases for now: anthracnose, powdery mildew, sooty mold, and potentially extend our disease classes as the project goes on.

## 2 Related Works

- Performance of deep learning vs machine learning in plant leaf disease detection [1]: this paper compares different types machine learning and deep learning approaches to classifying citrus diseases. It examines four types of diseases for citrus: black spot, melanose, canker, and greening. The ML and DL methods being applied are SGD, RF, SVM, and Inception-v3, VGG-16, VGG-19. They achieved an accuracy ranging from 76% to close to 90%, with VGG-19 having the best performance.
- Houseplant leaf classification system based on deep learning algorithms [2]: This paper focuses on house plants classification. It uses a dataset of 2500 images, where 250 images for each class. The images are pre-processed into 224x224x3 and the dataset were divided into 8:1:1 for training, validating, and testing. They also augmented the dataset by rotating, flipping, and shifting. Their choice of model to fine-tune is ResNet-50. The authors trained their dataset on ResNet in different stages, where at

stage  $i$ , they freeze the  $i$ -th block of the model and train the remaining part. Their metrics were based on the confusion matrix, and included accuracy, recall, precision, and F1-score. They were able to achieve close to perfect classification result (98.6%, 99%, etc).

- Disease detection on the leaves of the tomato plants by using deep learning [3]: This paper aims to detect disease on tomato plants using Deep learning. In particular, they chose AlexNet and SqueezeNet as their model choice, aiming to achieve real-time classification. It uses the PlantVillage dataset, and apparently just apply the trained AlexNet model on the tomato section of the dataset to achieve a 97% accuracy.

**Summary:** None of the related works directly address the task of classifying house plant diseases. One paper surveys various machine learning and deep learning approaches for citrus disease classification, but it is focused on crops. Another paper focuses on classifying house plant types, not diseases, and achieves good results using ResNet-50 with advanced training techniques. The final paper examines the accuracy of deep learning models in detecting tomato plant diseases, using the older AlexNet model from 2017. However, no specific training techniques were discussed in that paper.

## 3 Execution Plan

### 3.1 Data Collection

The initial plan is to collect 1000 images, consisting of 250 anthracnose images, 250 powdery mildew images, 250 sooty mold images, and 250 healthy house plant images. The dataset will be collected from available free sources such as Kaggle and Imgur. If the dataset is insufficient, a web scraping tool will be implemented to gather images from popular search engines such as Bing and Google.

### 3.2 Data Preprocessing

The collected images will then be preprocessed to maintain consistency across the dataset. Steps will include resizing the images to a standard size suitable for the model, as well as sorting and labeling the collected datasets into their respective categories (diseased vs. healthy). Moreover, image transformations (e.g., rotations, flips, and brightness adjustments) will be employed to augment the dataset and improve model accuracy and robustness during training.

### 3.3 Model Training

The dataset will be split into training, validation, and testing datasets following a 6:2:2 ratio. We are going to implement the ResNet50 model for detecting house plant diseases. ResNet50 can handle complex image classification tasks while maintaining high accuracy through deep

feature extraction, it does not need bounding boxes on the dataset. The model will be fine-tuned using transfer learning and adjusting pre-trained weights from ImageNet to detect specific plant diseases. The model's hyperparameters, such as learning rate and batch size, will be carefully optimized during the training process. The network architecture will retain ResNet50's core layers while adjusting the output layer to match the four classification categories (three diseases + healthy).

### **3.4 Model Evaluation**

The model's performance will be evaluated using the validation set during training and further evaluated on the testing set after all training iterations are completed. We will adopt classification evaluation metrics such as confusion matrix, accuracy, precision, recall, and F1 score. These evaluation metrics are good indicators to tell whether the model effectively distinguishes between each diseases and healthy plants.

## **4 Evaluation Plan**

### **4.1 Confusion Matrix**

A confusion matrix will be constructed to analyze the model's performance across all four classes (three diseases + healthy). This will allow us to visualize the number of true positives, false positives, true negatives, and false negatives for each category.

### **4.2 Test Dataset Accuracy**

The model's accuracy such as precision, recall, and F1 score will be calculated on the test dataset to quantify how well it classifies unseen data. In addition, ROC curves will be plotted for each class, with the area under the curve (AUC) used to assess the ability of the model to distinguish between classes.

### **4.3 Manual Testing with Live Photos**

Finally, the model will undergo manual testing using live plant photos to verify its real-world performance, ensuring that it generalizes well outside the dataset.

## **5 Expected Outcomes / Contribution Objectives**

### **5.1 Accuracy**

The resulting confusion matrices, ROC curves and AUC scores should provide us with detailed quantitative information to verify whether the model can reliably distinguish between different plant diseases so that the false positive and/or false negative rates are minimized.

Based on these metrics, the deep learning model will allow us to group plant images into four choices – anthracnose, powdery mildew, sooty mould, and a healthy plant. If the fine-tuned ResNet50 model is well-trained and the dataset is well-designed and prepared, the classification accuracy on the test dataset should be high. Ideally, it should be greater than 90 percent.

## References

- [1] R. Sujatha, J. Chatterjee, NZ Jhanjhi, S. Brohi *Performance of deep learning vs machine learning in plant leaf disease detection*. Microprocessors and Microsystems, 2020.
- [2] H. Hama, T. Absulsamad, S. Omer *Houseplant leaf classification system based on deep learning algorithms*. Journal of Electrical System and Information Technology, vol. 11, 2024.
- [3] H. Durmus, E. Gunes, M. Kirci *Disease detection on the leaves of the tomato plants by using deep learning*. IEEE, 2017.
- [4] N. Gupta *Evaluation metrics for multi-class classification*. Kaggle.  
Available: <https://www.kaggle.com/code/nkitgupta/evaluation-metrics-for-multi-class-classification>
- [5] E. Rex *PlantVillage Dataset*. Kaggle.  
Available: <https://www.kaggle.com/datasets/emmarex/plantdisease>

## Links

- <https://link.springer.com/article/10.1186/s43067-024-00141-5>
- <https://www.sciencedirect.com/science/article/pii/S0141933120307626>
- <https://ieeexplore.ieee.org/abstract/document/8047016>