Plant Disease Detection from Leaf Images

Abstract

Agricultural productivity is heavily impacted by plant diseases, leading to substantial economic losses and reduced crop quality. Traditional disease detection methods rely on manual inspection, which is time-consuming and requires expert knowledge. With advancements in deep learning, automated plant disease detection has become feasible. This project implements a **Convolutional Neural Network (CNN)** model to classify plant leaf images into 15 categories, covering both healthy and diseased plants. Using the **PlantVillage dataset**, the system achieved an accuracy of approximately **89%** on the validation dataset. The model can predict plant diseases from unseen images, assisting farmers and agricultural researchers in early disease detection and prevention strategies.

Introduction

The agricultural sector is vital for global food security, and plant diseases pose a serious threat to crop yield and quality. Early detection of plant diseases can significantly reduce losses and improve productivity. However, manual disease detection is labor-intensive and often inaccurate due to human error. **Machine learning and deep learning** provide an efficient and scalable solution for disease detection by analyzing images of plant leaves. This project utilizes a **pre-trained CNN model** trained on the PlantVillage dataset. The model processes plant leaf images and predicts the respective disease class. By automating this process, the project aims to provide a **cost-effective, accurate, and fast disease detection method**.

Tools Used

Programming Language: Python 3.x
Libraries and Frameworks: TensorFlow, Keras, NumPy, OpenCV, scikit-learn, Matplotlib
Model: Pre-trained CNN model (PlantVillage.h5)
Development Environment: Jupyter Notebook / Python IDE
Dataset: PlantVillage dataset containing labeled images of healthy and diseased plant leaves.

Steps Involved in Building the Project

1. Dataset Preparation: The PlantVillage dataset is organized into class-wise folders. The dataset is split into training, validation, and testing sets using an 80-20 split ratio.
2. Data Preprocessing: Images are resized to 224×224×3 and normalized to have pixel values between 0 and 1. For training data, augmentation techniques such as rotation, zoom, and horizontal flipping are applied to improve generalization.
3. Model Loading: The pre-trained CNN model (PlantVillage.h5) is loaded and compiled using the Adam optimizer with a learning rate of 0.0001. The model uses categorical cross-entropy as the loss function.
4. Model Evaluation: The trained model is evaluated on the validation set, achieving approximately 89% validation accuracy and a loss of 0.37.
5. Prediction on New Images: A custom function takes an image input, preprocesses it, and predicts the disease class. The predicted class label is mapped to the corresponding disease name.

6. Visualization: Graphs of training and validation accuracy/loss are plotted to analyze the model's performance.
7. Saving and Loading the Model: The trained model is saved in .h5 format and reloaded for predictions without retraining.

Results and Discussion

The CNN model performed well on the validation dataset, achieving high accuracy across most plant disease classes. However, some classes with fewer samples had slightly lower performance. Misclassified images were analyzed to identify possible causes such as image quality, similarity between disease symptoms, and insufficient data for certain classes.

The use of data augmentation improved the generalization ability of the model. The confusion matrix highlighted that the model performed better on classes with larger datasets. The approach can be enhanced further by fine-tuning with transfer learning using models like VGG16, ResNet50, or EfficientNet for improved accuracy.

Conclusion

The project successfully implemented a deep learning-based plant disease detection system using a pre-trained CNN model. It provides a **fast, reliable, and cost-effective method** for plant disease classification. This solution can be integrated into mobile or web applications for real-time disease detection, reducing dependency on manual inspection. Future improvements can include extending the dataset, applying advanced transfer learning techniques, and deploying the model in user-friendly applications to benefit farmers and researchers worldwide.

Future Scope

Integration with mobile apps for real-time disease detection.
Expanding the dataset to include more plant species and diseases.
Applying advanced deep learning models for higher accuracy.
Developing a recommendation system for treatment based on detected disease.

By leveraging AI, this project contributes to **precision agriculture**, improving productivity and reducing crop losses due to plant diseases.