PROJECT ABSTRACT

Plant Leaf Disease Detection using CNN

CNN classifier that identifies plant leaf diseases from the 38-class PlantVillage dataset, delivered as a built-to-order student project with training notebook, weights, demo app, report, PPT and viva kit.

01. Title

Plant Leaf Disease Detection using CNN

02. Introduction / Background

Plant diseases reduce crop yields every season, and early visual diagnosis is still done by farmers and extension workers inspecting leaves by eye — a slow process that requires expertise most growers do not have at hand. Convolutional neural networks have proven strong at this exact task: learning lesion color, texture and shape cues directly from leaf photographs. The PlantVillage dataset (Hughes and Salathe, released through the crowdAI challenge) — approximately 54,305 leaf images across 14 crops and 38 healthy/diseased classes — is the standard teaching benchmark. This project trains a CNN on PlantVillage and wraps it in an upload-and-diagnose demo app with top-3 predictions.

03. Problem Statement

A farmer who spots suspicious leaf lesions today must wait for an expert visit or guess the treatment, and a wrong guess wastes pesticide and yield. The project addresses this with an image classifier: photograph a leaf, and the trained CNN returns the most likely crop-disease class with confidence scores and reference guidance text. The training experiment is fully logged, so the report presents the model's measured accuracy, per-class F1 and confusion matrix — and discusses exactly which diseases the model confuses.

04. Objectives

- Build a PlantVillage data pipeline with a stratified 80/10/10 split covering all 38 classes, plus augmentation (rotation, flip, brightness, zoom).
- Train a CNN (4-block custom architecture, with a MobileNetV2 transfer-learning variant documented) using Adam and early stopping, with a design target of approximately 90%+ test accuracy on the controlled-condition images.
- Log accuracy, per-class F1 and the confusion matrix every epoch; the notebook computes the final metrics on the held-out split during the buyer's build.

- Deliver a demo app (upload photo, top-3 prediction, attention-style visualization), saved weights, report, PPT and viva Q&A.

05. Proposed Methodology

The project follows the standard image-classification workflow. PlantVillage images, organized in crop___disease folders, are resized to 256x256, normalized and augmented to tolerate varied phone-camera conditions. The baseline model stacks four convolution blocks (32, 64, 128 and 256 filters) with batch normalization and max-pooling, followed by global average pooling and a 38-way softmax; a MobileNetV2 transfer-learning variant is documented as the stronger option. Training uses categorical cross-entropy with the Adam optimizer and early stopping on validation loss. Evaluation reports test accuracy, macro-F1 and the confusion matrix, with explicit discussion of confusions between visually similar diseases such as tomato early vs late blight.

06. System Architecture / Working

The system has three stages. (1) Data pipeline: PlantVillage images are loaded with stratified splitting and on-the-fly augmentation through a tf.data-style pipeline. (2) Model: the CNN is trained for up to 25 epochs with per-epoch logging of accuracy, loss and per-class F1; the best-validation checkpoint is kept as the shipped weights. (3) Demo application: a web app accepts a leaf photo, runs inference in under a second on CPU, and displays the top-3 predicted classes with confidence bars plus an attention-style heatmap that highlights the high-texture regions the network weighed most.

07. Hardware / Software Requirements

- Python 3.10, TensorFlow/Keras (training and inference)
- NumPy, pandas (data handling), scikit-learn (metrics, confusion matrix)
- Matplotlib, Seaborn (training curves, per-class analysis)
- Flask demo web app with image-upload UI
- PlantVillage dataset (approximately 54,305 images, 38 classes, 14 crops; public via Kaggle mirrors of the crowdAI release)
- CPU training is workable; a GPU shortens the multi-hour training run

08. Expected Outcomes

The student receives a trained 38-class leaf-disease classifier whose metrics are measured on the held-out split during their own build — the design target is approximately 90%+ test accuracy on PlantVillage's controlled-condition images. The confusion matrix is expected to show confusions between visually similar diseases, which the report analyzes openly. The demo app returns a top-3 diagnosis with confidence scores in under a second on CPU.

09. Applications

- Early-screening aid for farmers and agriculture students (prototype scope)
- Teaching material for CNN classification: augmentation, transfer learning, confusion-matrix analysis
- Reference build for other fine-grained agricultural vision tasks
- Extension-service demo tool for plant-pathology outreach

10. Future Scope

- Fine-tune on field photographs with cluttered backgrounds for real-world robustness
- Add severity estimation (percentage leaf area affected)
- Add treatment-guidance content per predicted disease
- Compress the model (quantization) for offline smartphone inference