Project proposal

Artificial Intelligence

Submitted by:

Mahboob Rasheed (55434)

Submitted to:

Sir/Mr. Muhammad Usman Karim sahib

Faculty of Computing

RIPHAH International University Islamabad

Title

Plant Disease Detection System Using Deep Learning

1. Introduction

Agriculture is a backbone of many economies and plant diseases can significantly reduce crop yield and quality. Traditionally farmers rely on manual inspection, which is time consuming and often inaccurate. This project aims to build an AI-powered plant disease detection system that automatically classifies plant leaf images as healthy or diseased enabling farmers to take timely action.

2. Problem Statement

Farmers face difficulty in identifying plant diseases early leading to crop loss and reduced productivity. A quick accessible and automated system is needed to detect plant diseases accurately using image recognition techniques.

3. Objective

- To develop a machine learning model that can classify plant leaves into healthy or diseased categories.
- To create a user friendly interface where users can upload plant leaf images and get instant disease predictions.
- To help farmers and agricultural experts in taking early preventive measures.

4. Dataset

We will use the **Plant Village Dataset**

- Available on Kaggle Plant Village Dataset

- Contains 54,000+ images of plant leaves from 38 different classes (diseases + healthy).
- Covers common crops like tomato potato corn etc.

We may start with a subset (Tomato healthy vs diseased) for binary classification and later extend to multiple classes.

5. Proposed Methodology

Step 1: Data Collection & Preprocessing

- Download and clean the dataset.
- Resize all images to a fixed size
- Normalize pixel values (divide by 255).
- Split data into training (80%) validation (10%) and testing (10%) sets.
- Apply data augmentation (rotation flipping zoom) to make model robust.

Step 2: Model Design (Deep Learning - CNN)

We will use a Convolutional Neural Network (CNN) for image classification.
Example architecture

1. **Input Layer** (128×128×3 images)
2. **Conv2D + MaxPooling2D Layers** (feature extraction)
3. **Flatten Layer**
4. **Dense Layers** with ReLU activation
5. **Output Layer** with `softmax` (multi-class) or `sigmoid` (binary classification)

Step 3: Model Training

- Train for 20–30 epochs.
- Use Early Stopping to avoid overfitting.
- Monitor training and validation accuracy/loss.

Step 4: Model Evaluation

- Evaluate on test set using:
    - Accuracy
    - Confusion Matrix
    - Precision Recall F1-Score

Step 5: System Integration

- Build a simple Flask Web App**:**
  - Upload leaf image.
  - Model predicts class and displays result.
  - Show recommended treatment or preventive measure.

---

6. Tools and Technologies

Component	**Technology / Library**
Programming Language	Python
ML/DL Framework	TensorFlow / Keras
Data Handling	Pandas, NumPy
Visualization	Matplotlib, Seaborn
Model Deployment	Flask / Streamlit
Dataset Source	Kaggle (PlantVillage Dataset)

---

7. Expected Outcome

- A trained model that can classify plant leaves with **80–90% accuracy**.
- A user-friendly web interface that allows users to upload images and get instant results.
- Contribution to precision agriculture and reduced crop losses.