

Real time plants disease detection by Mudassar khan

Overview

This application is a Real time plants disease detection designed to detect diseases in plants. It can detect 30 different diseases. It utilizes YOLO model, the application is built using the **Flask** web framework, allowing users to interact with the model through a web interface.

Features

- Upload images for processing and detection.
- Start Web camera for real time detection.
- Display the detected results.

Model Used

The application uses the yolov11 model from ultralytics, which should be trained on your specific dataset for optimal performance.

Dataset Used

The dataset used for training is a Plantdoc dataset which has a 2569 labelled images of plant infected leaves. You can find the dataset here: [Plantdoc dataset](#)

Prerequisites

Before running the application, ensure you have the following installed:

- Python 3.7 or higher
- pip (Python package installer)

Setup Instructions

Create a Virtual Environment

It is recommended to create a virtual environment to manage dependencies for this application. You can create a virtual environment using the following commands:

```
# Navigate to your project directory
cd /path/to/your/project

# Create a virtual environment (you can name it 'venv' or any name you prefer)
python -m venv venv

# Activate the virtual environment
```

```
# On Windows
venv\Scripts\activate

# On macOS/Linux
source venv/bin/activate
```

Install Dependencies

Once the virtual environment is activated, you can install the required packages listed in the **requirements.txt** file. Run the following command:

```
pip install -r requirements.txt
```

This command will install all the necessary libraries for the application, including Flask, OpenCV, NumPy, and Pillow.

Run the Flask Application

After installing the dependencies, you can run the Flask application. Make sure you are still in the project directory during executing this command. If you are using VS code open the terminal and execute this command.

```
python application.py
```

This will start the Flask development server, and you should see output indicating that the server is running, typically on <http://127.0.0.1:5000/>.

Access the Application

Open your web browser and past this address into the web browser <http://127.0.0.1:5000/> to access the The frontend of this application. now there you can upload images