Plant Health Analysis README

This Python script analyzes images of plants to detect leaf damage using color thresholding techniques. It identifies yellow/brown and white patches on the leaves, which may indicate health issues. The script utilizes OpenCV for image processing and Matplotlib for visualization.

Requirements

To run this script, you need to have the following Python packages installed:

- OpenCV (opencv-python)
- NumPy (numpy)
- Matplotlib (matplotlib)

You can install these packages using pip:

```
pip install opencv-python numpy matplotlib
```

Directory Structure

The script expects a directory named input in the same location as the script. This directory should contain the images to be analyzed. The images should be named in the format DayXX_T_06.jpg, where XX is a two-digit day number (e.g., Day01_T_06.jpg).

```
/project_directory
│
├── input
│   ├── Day01_T_06.jpg
│   ├── Day02_T_06.jpg
│   ├── ...
│   └── Day14_T_06.jpg
│
└── analyze_plant_health.ipynb
```

Usage

Place your images in the input directory.

Run the script

The script will process each image from Day 1 to Day 14 and display the results.

Output

For each image processed, the script will:

- Display the original image.
- Show a mask indicating the detected damage areas.
- Show the original image with red bounding boxes around the detected damage areas.

Additionally, the script will print a message indicating whether damage was detected in each image:

- "No Damage Detected" if the damage percentage is less than 0.001%.
- "Leaf Damage Detected" if damage is found.

Code Explanation

- Image Reading: The script reads images from the specified directory using OpenCV.
- Color Conversion: It converts the image from BGR to RGB and then to HSV color space for better color segmentation.
- Color Thresholding: Two masks are created to detect yellow/brown and white patches:
- Yellow/Brown patches are detected using specified HSV thresholds.
- White patches are detected using different HSV thresholds.
- Contour Detection: The script finds contours in the combined mask of detected areas.
- Bounding Boxes: For each detected contour, a red bounding box is drawn on the original image.
- Damage Calculation: The script calculates the total area of detected damage and computes the damage percentage relative to the total image area.

Visualization:

- The results are displayed using Matplotlib.

Notes

Ensure that the images are well-lit and clear for better detection accuracy.

The color thresholds may need to be adjusted based on the specific conditions and types of plants being analyzed.

The script currently processes images for 14 days. You can modify the range in the loop if you have more or fewer images.