Plant Health Detection and ArUco Marker Detection

Aerial Robotics Vision Project

Smruti Ranjan Ghosh

1 Introduction

This project focuses on implementing computer vision techniques for aerial robotics applications. The objectives were:

- Detect yellow (infected) leaves in plant images.
- Detect and estimate pose of ArUco markers using a webcam.

These tasks simulate real-world drone applications such as crop health monitoring and autonomous navigation using visual landmarks.

2 Task 1: Yellow Leaf Detection

2.1 Objective

The goal was to detect infected (yellow) leaves from plant images and draw bounding boxes around them.

2.2 Conceptual Understanding

The approach is based on classical image processing:

- Conversion from BGR to HSV color space
- Mask generation using color thresholding
- Contour detection
- Bounding box localization

Why HSV Color Space?

HSV separates color information (Hue) from brightness (Value). This makes it more robust for color-based segmentation compared to RGB.

2.3 Workflow

1. Read input image
2. Convert image from BGR to HSV
3. Define yellow color range
4. Generate mask using `cv2.inRange()`
5. Detect contours using `cv2.findContours()`
6. Draw red bounding boxes around detected regions

2.4 Mask Generation

Mask generation creates a binary image:

- White pixels (255) represent yellow regions.
- Black pixels (0) represent background.

This isolates infected leaves for further processing.

2.5 Contour Detection

Contours represent boundaries of connected white regions in the mask. Using contour detection, bounding rectangles are drawn around infected leaves.

2.6 Results

The system successfully detected yellow leaves and ignored healthy green regions. Detection accuracy depended on lighting conditions and HSV threshold tuning.

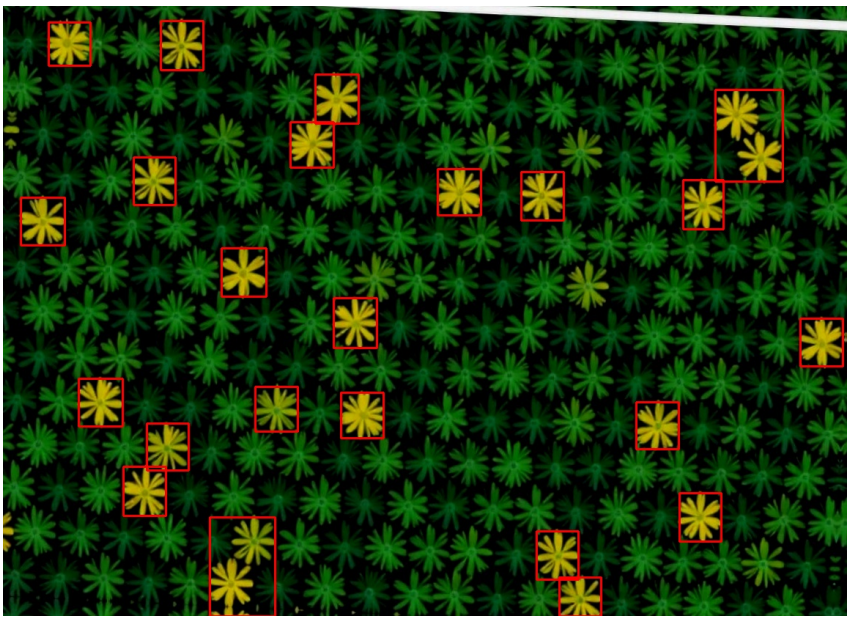

Figure 1: Detected Yellow Leaves with Bounding Boxes

2.7 Challenges Faced

- Tuning HSV threshold range.
- Removing small noisy contours.
- Handling varying lighting conditions.
- accessing the webcamera properly through codes.

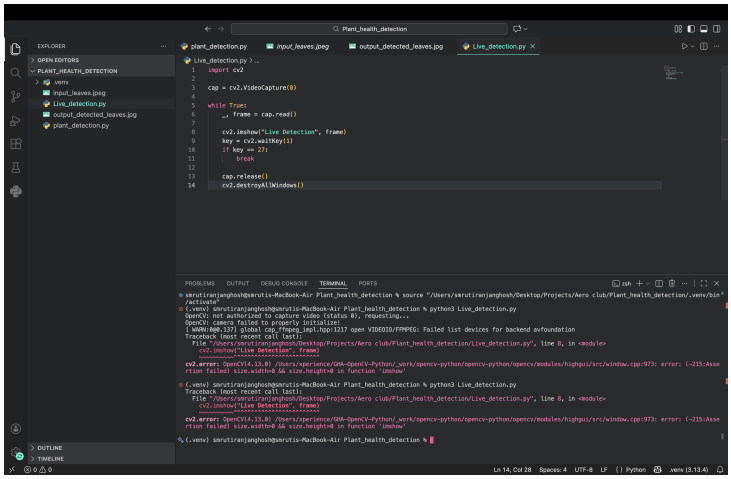

Figure 2: getting coding errors!

3 Task 2: ArUco Marker Detection

3.1 Objective

To detect ArUco markers in real-time using webcam feed and estimate their pose.

3.2 Understanding ArUco Markers

ArUco markers are square binary patterns used in robotics for:

- Localization
- Pose estimation
- Autonomous landing
- Navigation

Each marker has a unique ID encoded within a predefined dictionary.

3.3 Workflow

1. Capture frame from webcam
2. Convert frame to grayscale
3. Load predefined dictionary
4. Detect markers using `aruco.detectMarkers()`
5. Draw bounding boxes
6. Estimate pose using `estimatePoseSingleMarkers()`
7. Draw 3D axes (X, Y, Z)

3.4 Results

When the marker is visible:

- Bounding box is drawn.
- Marker ID is detected.
- 3D coordinate axes are displayed.

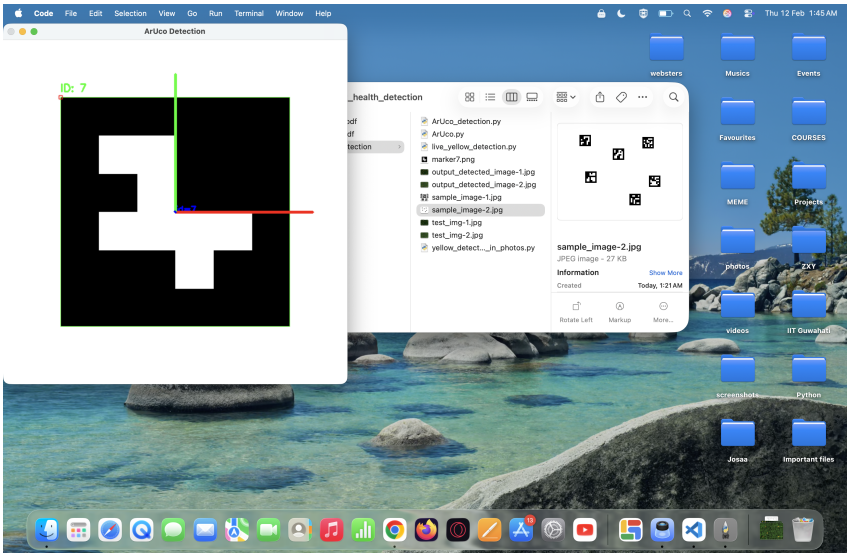

Figure 3: screenshot of ArUco markers getting detected

When the marker is not visible:

- No detection occurs (IDs = None).

3.5 Challenges Faced

- Marker detection failed under poor lighting.
- Dictionary mismatch caused detection errors.
- ESC key handling issues during debugging.
- Need for white border around generated markers.

4 Applications

4.1 Plant Health Detection

- Precision agriculture
- Crop disease monitoring
- Automated drone-based inspection

4.2 ArUco Marker Detection

- Autonomous drone landing pads
- Indoor robot navigation

5 Conclusion

This project demonstrates practical implementation of classical computer vision techniques for aerial robotics.

- Yellow leaf detection using HSV segmentation and contour analysis.
- Real-time ArUco marker detection with pose estimation.

These techniques form a foundation for advanced drone-based perception systems and can be extended using machine learning for improved robustness.