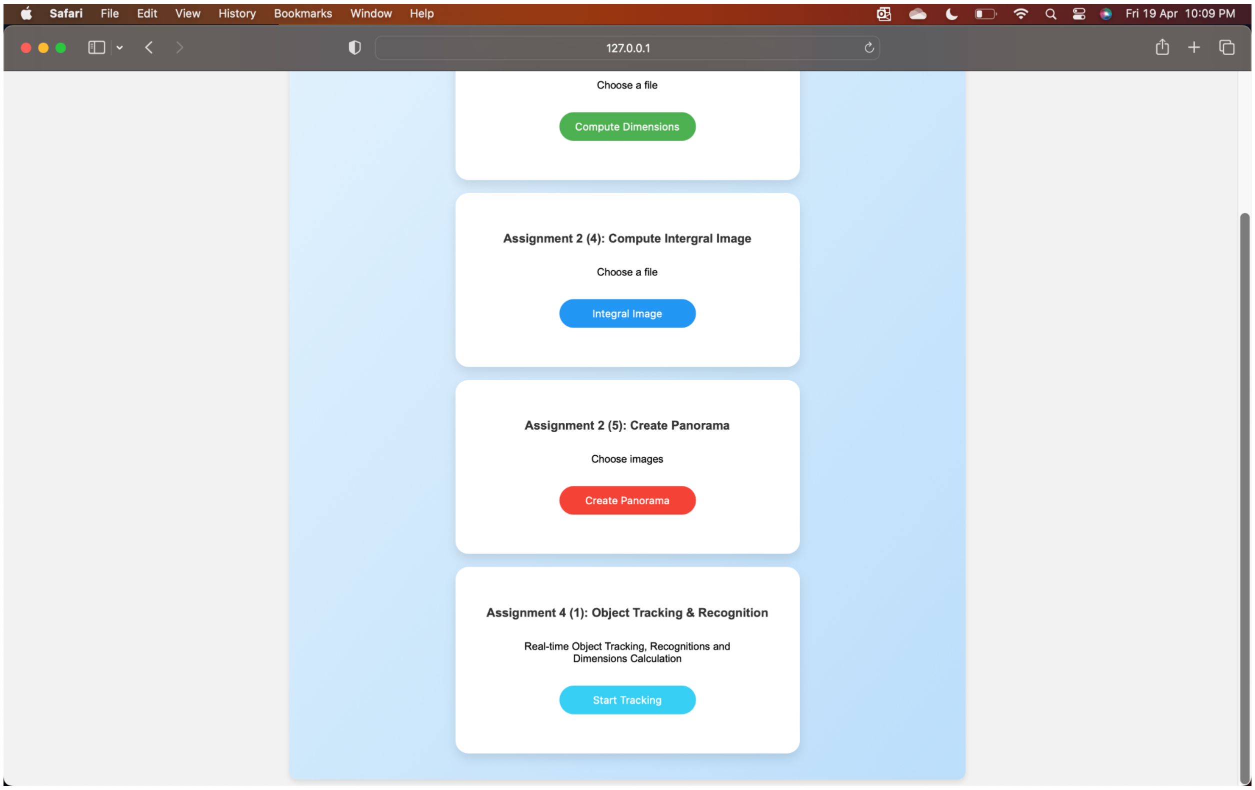
CSc 8830: Computer Vision

Assignment-4 Report

1. Implement an application (must run on web or as an app on mobile device) using the stereo camera where it will recognize, track and estimate dimensions (at least 2D) of any object within 3m distance and inside field-of-view to the camera. You can use barcodes or text recognition tools for identification. However, the entire object must be tracked (not just the barcode or text). Machine/Deep learning tools are NOT allowed.

1. **Stereo Vision Setup:**
   - The application sets up stereo depth processing using DepthAI library, which provides depth information from stereo cameras.
   - It configures two mono cameras (left and right) and a color camera for capturing RGB images.
   - Stereo depth estimation is performed using the stereo pair to generate a depth map.
2. **Object Tracking:**
   - The application implements an object tracker (ObjTracker) that detects and tracks objects using background subtraction and contour detection.
   - Detected objects are tracked based on their centroids, and bounding boxes are drawn around them in the stereo vision feed.
3. **QR Code Detection:**
   - QR codes are detected using OpenCV's QR code detector (cv2.QRCodeDetector()).
   - Detected QR codes are outlined, and their decoded information is displayed on the stereo vision feed.
4. **Dimension Estimation:**
   - The application utilizes a class (ObjDimensionMarker) to estimate the dimensions of objects within the camera's field of view.
   - It marks the dimensions of objects by identifying their contours, calculating the dimensions based on known object widths, and drawing lines and text on the RGB feed.



Choose a file

Compute Dimensions

Assignment 2 (4): Compute Intergral Image

Choose a file

Integral Image

Assignment 2 (5): Create Panorama

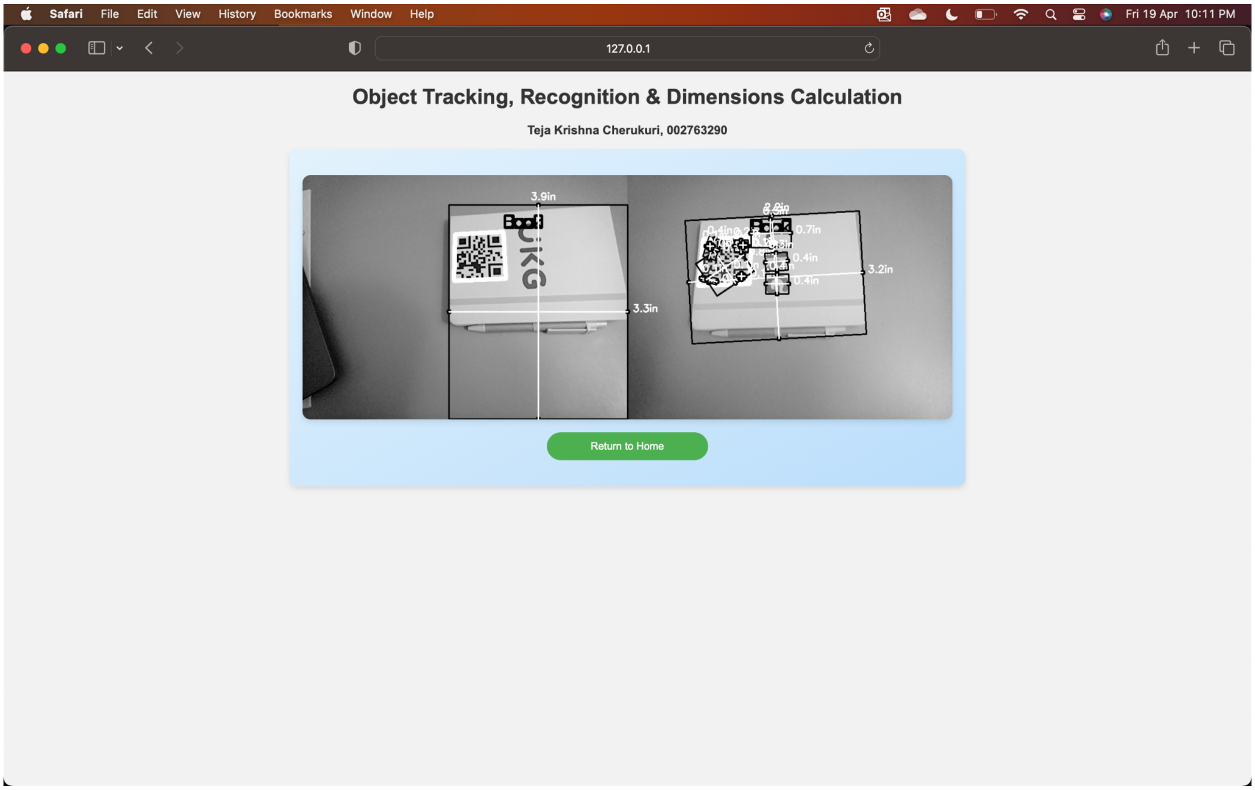
Choose images

Create Panorama

Assignment 4 (1): Object Tracking & Recognition

Real-time Object Tracking, Recognitions and Dimensions Calculation

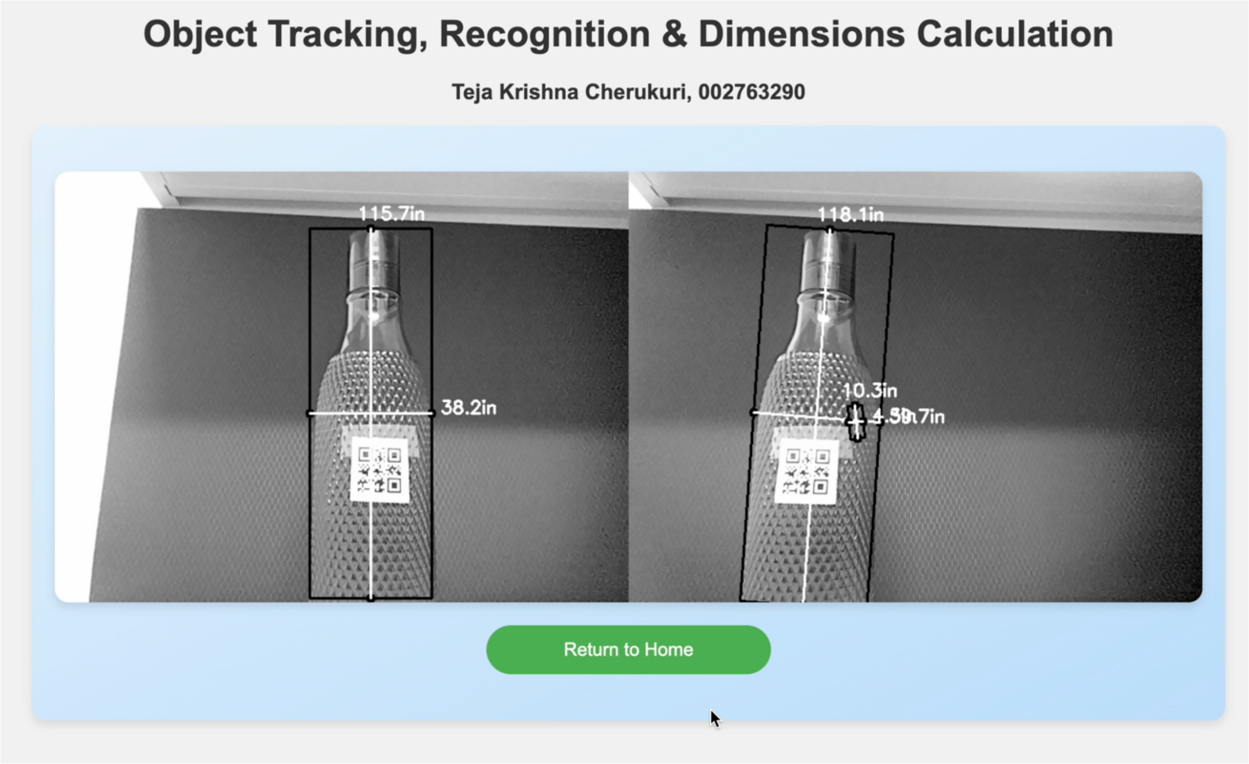
Start Tracking



Object Tracking, Recognition & Dimensions Calculation

Teja Krishna Cherukuri, 002763290

Return to Home



Object Tracking, Recognition & Dimensions Calculation

Teja Krishna Cherukuri, 002763290

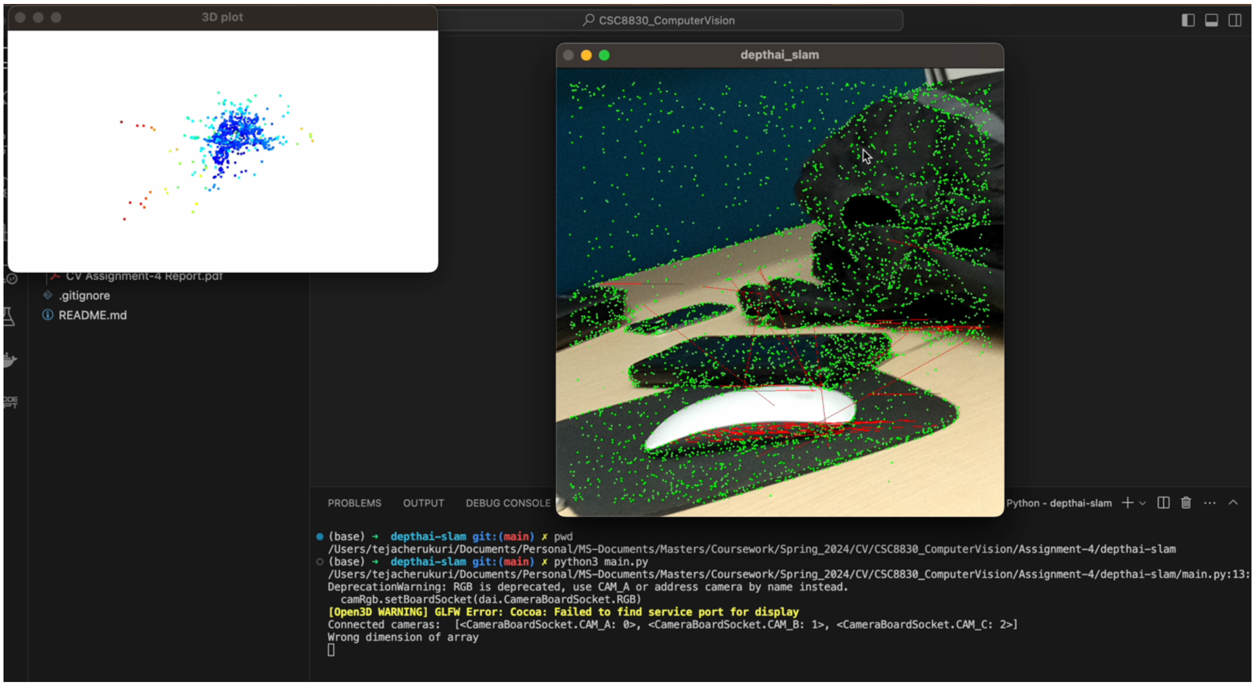
2. Use the DepthAI SDK or use ORB3-Visual SLAM (https://github.com/UZ-SLAMLab/ORB_SLAM3) to execute the scripts on your depth camera and run experiments in two different locations. Provide snapshots of your SLAM output and what limitations/corner cases do you observe.

Approach:

For this, I have used depthai-slam repository present at the below link.
https://github.com/bharath5673/depthai-slam

Observations:

- SLAM algorithms typically assume static environments, where the scene does not change significantly over time.
- In dynamic environments where objects move or change appearance, SLAM systems may struggle to accurately map the environment or track the robot's position.
- This can result in errors in both localization and mapping, especially if moving objects are mistaken for static landmarks.



```
● (base) → depthai-slam git:(main) ✗ pwd
/Users/tejacherukuri/Documents/Personal/MS-Documents/Masters/Coursework/Spring_2024/CV/CSC8830_ComputerVision/Assignment-4/depthai-slam
○ (base) → depthai-slam git:(main) ✗ python3 main.py
/Users/tejacherukuri/Documents/Personal/MS-Documents/Masters/Coursework/Spring_2024/CV/CSC8830_ComputerVision/Assignment-4/depthai-slam/main.py:13:
DeprecationWarning: RGB is deprecated, use CAM_A or address camera by name instead.
  camRgb.setBoardSocket(dai.CameraBoardSocket.RGB)
[Open3D WARNING] GLFW Error: Cocoa: Failed to find service port for display
Connected cameras:  [<CameraBoardSocket.CAM_A: 0>, <CameraBoardSocket.CAM_B: 1>, <CameraBoardSocket.CAM_C: 2>]
Wrong dimension of array
```

Repo:
https://github.com/TejaCherukuri/CSC8830_ComputerVision/tree/main/Assignment-4