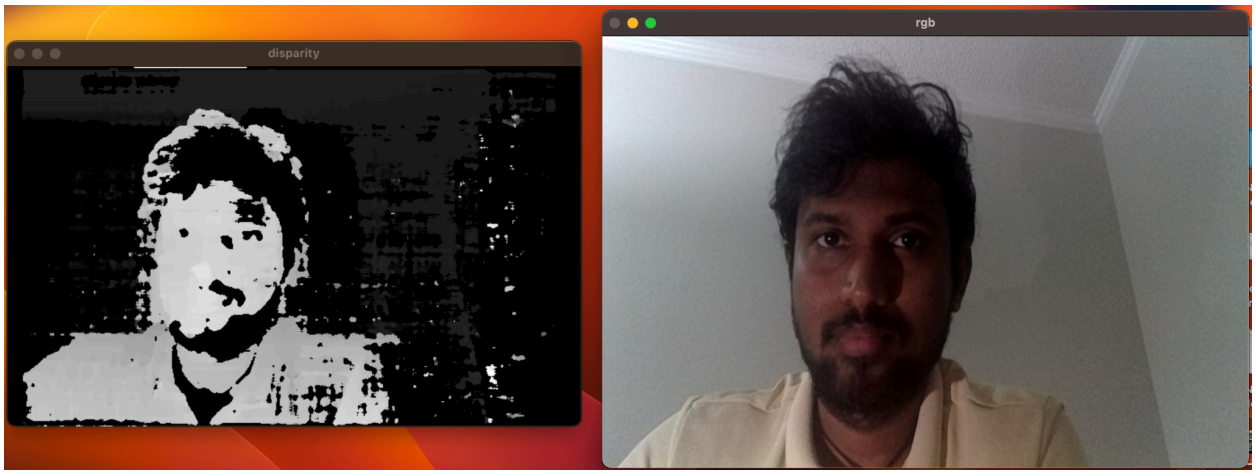
CSc 8830: CV Assignment-1 Solutions Report

(4). With the OAK-D camera, set up your application to show a RGB stream from the mono camera and a depth map stream from the stereo camera simultaneously. Make a note of what is the maximum frame rate and resolution achievable?



Maximum Frame Rate:92 FPS
Resolution: 1080 for RGB & 400 for depth map

1. **Report the calibration matrix for the camera chosen and verify (using an example) the same.**

   Calibration (Intrinsic) Matrix is:

   ```
   1.10337648962356047e+03 0.00000000000000000e+00 2.24967356582175028e+02
   0.00000000000000000e+00 1.10188906337563321e+03 2.85431664175799198e+02
   0.00000000000000000e+00 0.00000000000000000e+00 1.00000000000000000e+00
   ```

   Verification: (Verified edge of a square in chessboard pattern):

   ```
   Size of Square in mm: 32.32487685307927
   Calibration Error: 2.3248768530792674 mm
   ```

2. **Point the camera to a chessboard pattern or any known set of reference points that lie on the same plane. Capture a series of 10 images by changing the orientation of the camera in each iteration. Select any 1 image, and using the image formation pipeline equation, set up the linear equations in matrix form and solve for intrinsic and extrinsic parameters (extrinsic for that particular orientation). You will need to make measurements of the actual 3D world points, and mark pixel coordinates. Once you compute the Rotation matrix, you also need to compute the angles of rotation along each axis. Choose your order of rotation based on your experimentation setup.**

```
Intrinsic Camera Matrix:
[[210.45781727   0.         315.89660879]
 [  0.         210.13076857 238.12538675]
 [  0.           0.           1.        ]]

Extrinsic Rotation Matrix:
[[-0.99901378 -0.04423422 -0.00384739]
 [ 0.04406837 -0.9983854   0.0358409 ]
 [-0.00542657  0.03563601  0.9993501 ]]

Extrinsic Translation Vector:
[[130.86988591]
 [ 84.19519903]
 [169.59623503]]

Rotation Angles across X, Y, Z axes (degrees):
[  2.04225521   0.31092111 177.47421327]
```

3. **Write a script to find the real world dimensions (e.g. diameter of a ball, side length of a cube) of an object using perspective projection equations. Validate using an experiment where you image an object using your camera from a specific distance (choose any distance but ensure you are able to measure it accurately) between the object and camera.**

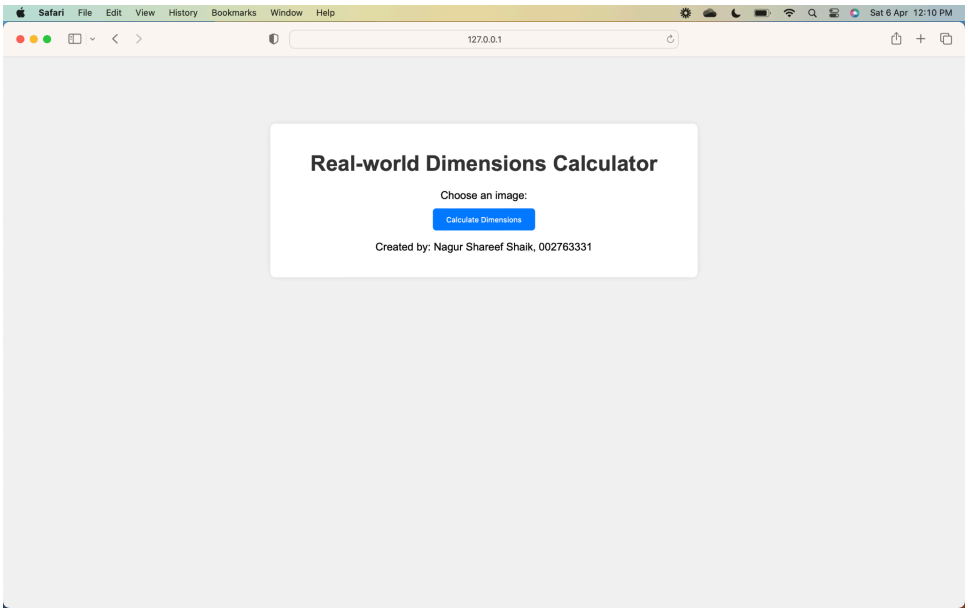
We choose to use circular objects to find real world dimensions

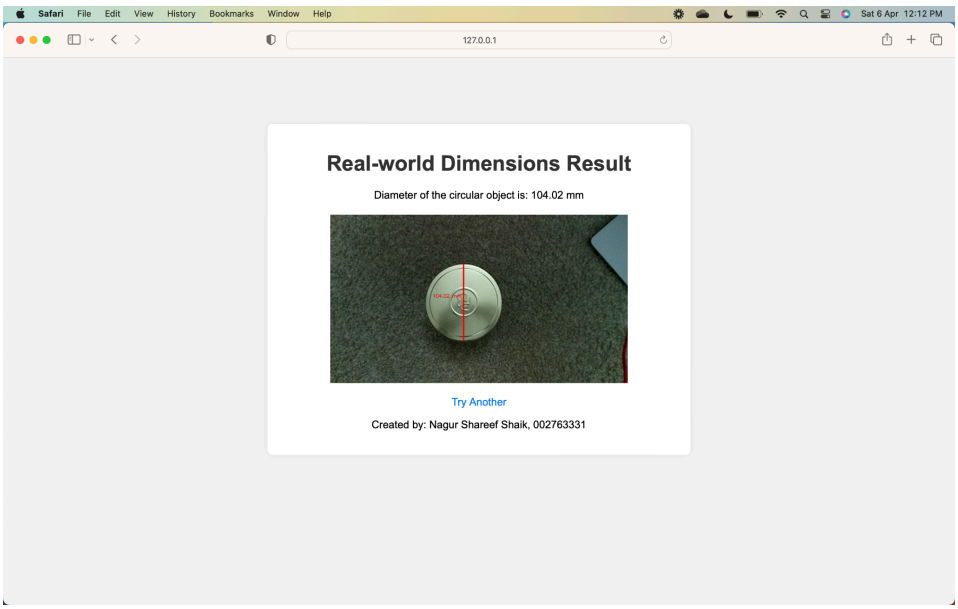
```
Camera Intrinsic Matrix:  [[210.45781726943, 0.0, 315.89660878657725], [0.0, 210.13076857289957, 238.1253867521593], [0.0, 0.0, 1.0]]
fx: 210.45781726943, fy: 210.13076857289957, Z: 300
Center (x, y): 877, 575; Width (w): 486; Height (h): 486

Real World Co-ordinates:
        1250.1317528309107
        820.917380027359
        1942.9071597588722
        1945.9311112648525

Diameter of circular object is: 104.03 mm
```

4. **Write an application – must run as a Web application on a browser and be OS agnostic – that implements the solution for problem (3) [An application that can compute real-world dimensions of an object in view]. Make justifiable assumptions (e.g. points of interest on the object can be found by clicking on the view or touching on the screen).**





Github Link:
https://github.com/ShaikNagurShareef/CSc8830-Computer-Vision/tree/main/Assignment-1