**ROB 498/599 – 3D Robot Perception, (write up).**

Problem 1

* Projection Matrix =

[[ 0.45827554 0.0040258 -0.54105993 0.17834548]

[-0.29474237 -0.05085589 -0.05237592 -0.04426782]

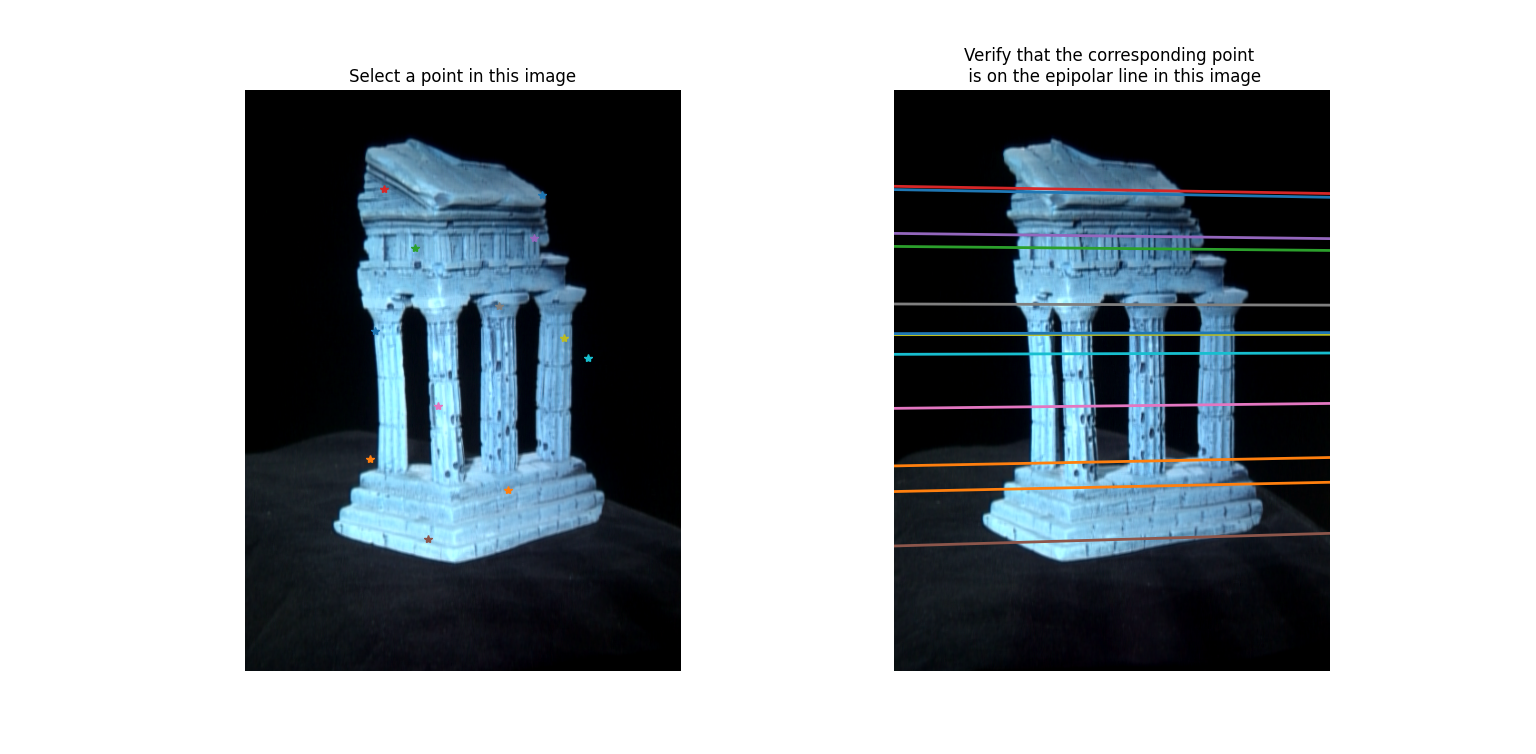
[-0.01395746 -0.0545847 0.10900958 0.5968205 ]]

* Reprojection Error =

0.6602427157295764

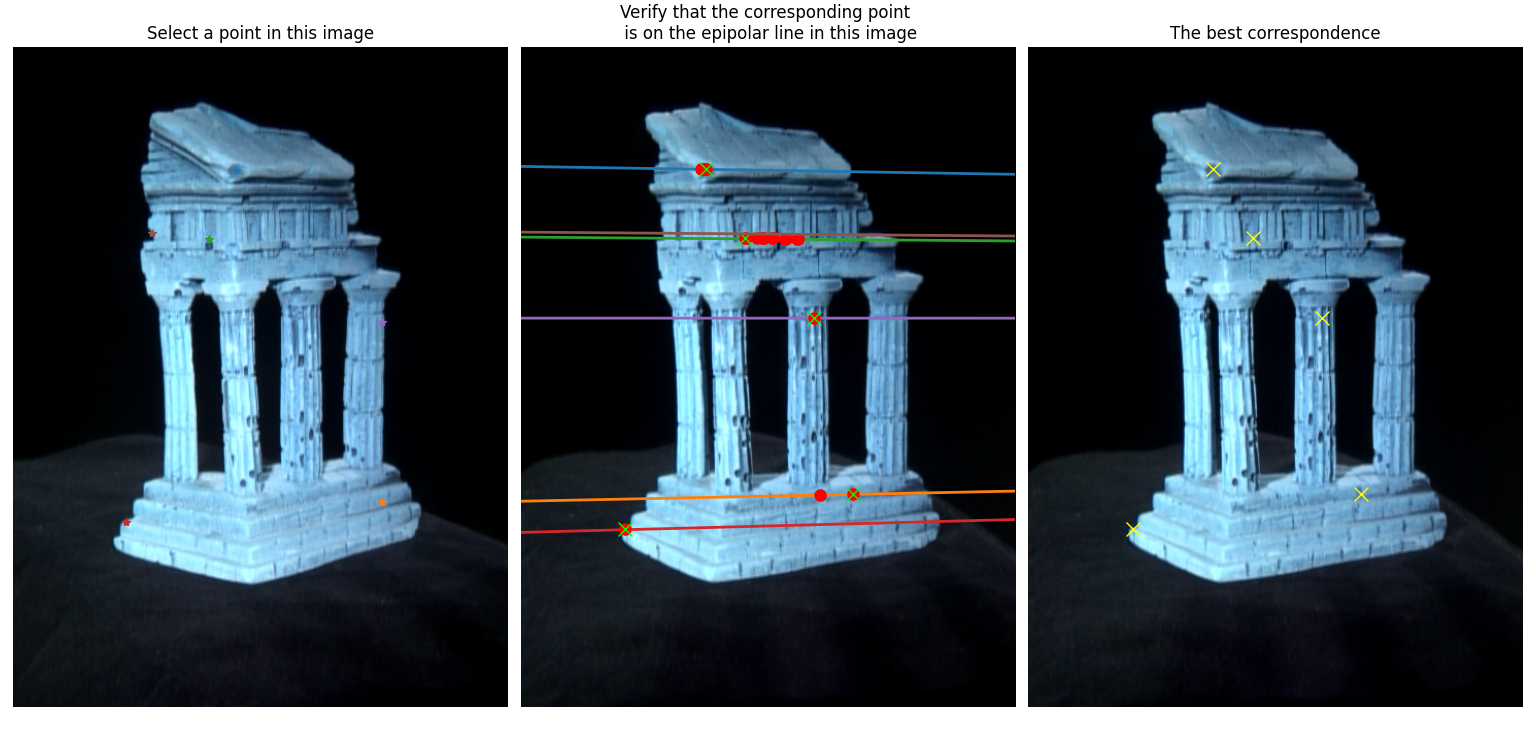
Problem 2

The screenshot of the Gui is attached with the zip folder while also inserting the image here.



Problem 3

The screenshot of the Gui is attached with the zip folder while also inserting the image here. I have also attached the image of pts 1 and their corresponding poins side by side for visualization. It isn’t perfect but it works.





Problem 4

The essential matrix is calculated with respect to camera 1 so using the formula ). The resulting matrix is as follows.

Essential matrix =

[[ 0.0298144 -0.30320729 0.0028928 ]

[-0.13677206 -0.00304365 -1.67587386]

[-0.04061437 1.65542993 -0.00192902]]

Problem 5

For triangulation, we need to compute the projection matrices of both the cameras. We can compute those because we have the intrinsic and the extrinsic as well. For the intrinsic, we have direct data files.

For projection matrix of camera 1, We have assumed the camera 1 as the reference camera, so the projection matrix is just the 3x3 identity matrix with [0 0 0].T translation as a 3x4 matrix.

For projection matrix for camera 2, we decompose the essential matrix and make 4 combinations with 2 R and 1 T. After finding the most points in front of the camera (I calculated point clouds for all of them and then manually saw which combination have positive x and y the most. We can directly check in the matrix as well but I didn’t think of that at first).

We then multiply the intrinsic and extrinsic matrices to get the projection matrix for both cameras. After the calculation for z is finished using the Ax = 0 SVD decomposition we get the 3d points.

The projection error of the calculated 3D points are written below

* Projection error for camera 1 = 183.72904179043667
* Projection error for camera 2 = 2.3723247840618282

I haven’t used the triangulation function from the cv2 because I don’t know why but my computer wasn’t processing the output of that function.

Problem 6

I have attached the point cloud picture here and with the submission scripts as well. Basic 3D visualization of matplotlib.

