3D RECONSTRUCTION

PART 1:

EIGHT POINT ALGORITHM:

Made pts1 and pts2 homogenous (Nx2 to Nx3 matrix)

Scaled the points by dividing x and y coordinates by M(max of height and width of the image) using a transformation matrix T.

T=[[1/M, 0, 0],

[0 , 1/M,0],

[0, 0, 1]]

So each row in pts1 and pts2 matrices become [x/M,y/M,1]

Each correspondence $P1_i$=(x1,y1,1) and $P2_i$=(x2,y2,1) satisfies the epipolar constraint:

$P1_i^T * F * P2_i = 0$

So overall the constraint is written as A*F=0

Where A is an Nx9 matrix where each row is equal to:

[x1*x2, y1*x2, x2, x1*y2, y1*y2,x1,y1,1]

Then I applied _singularize function from helper.py

In which singular value decomposition is done on matrix A such that A=U*S*V

U(NxN matrix)

S(Nx9 matrix)

V(9x9 matrix)

When linalg.svd function from numpy is used, we get U.shape=(N,N), S.shape=(9,), V.shape=(9,9)

S contains the singular values in descending order. By setting the smallest singular value(last element of S) as zero we enforce the rank 2 constraint.

And we rebuild the matrix as U*S'*V and apply svd again.

In the new V matrix that we get, the singular vector corresponding to smallest singular value is given by the last row and this is the solution for elements of F matrix. Then we reshape it into a 3x3 matrix.

The last step is to unscale the F matrix that I got using the formula:

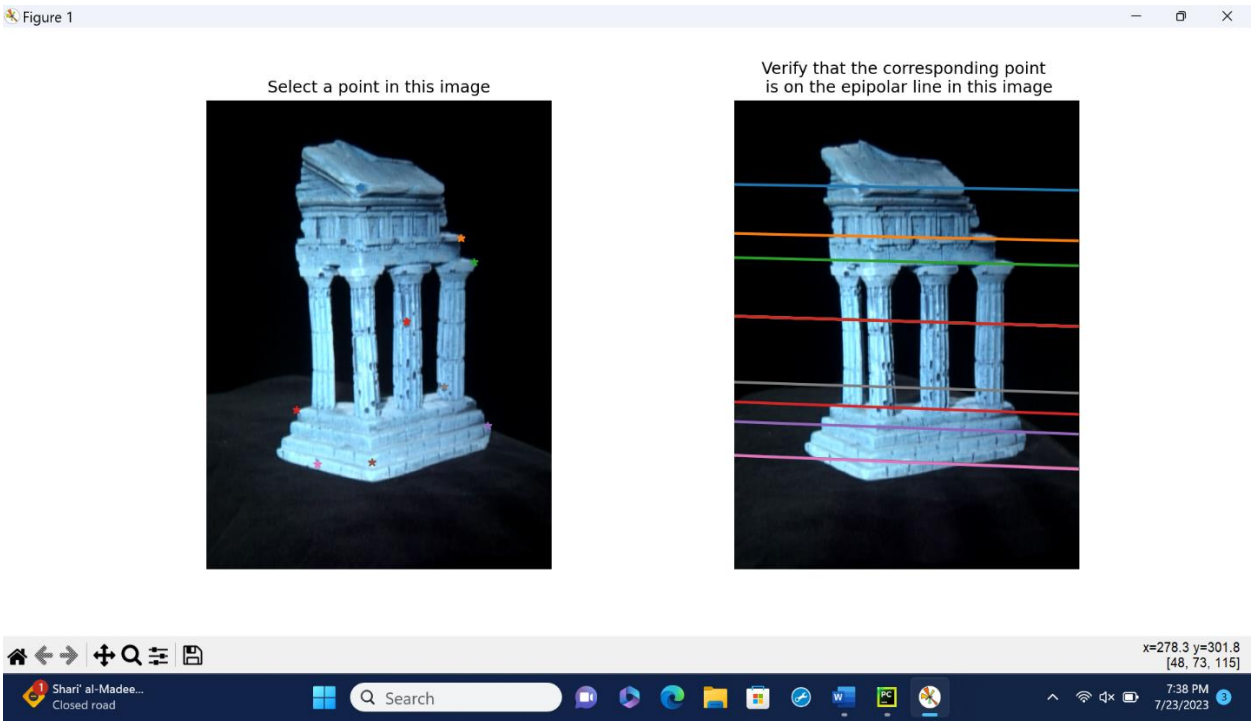
$F'=T^T*F*T$ where T is the transformation matrix used earlier to scale the data.

Derivation:

We replace P1 and P2 in the epipolar constraint equation $P1_i^T*F*P2_i=0$ with scaled P1 and P2

Thus $P1'^T*F'*P2'=(T*P1)^T*F*(T*P2)$

From this we get $F'=T^T*F*T$. This F' matrix is the fundamental matrix.



FINDING EPIPOLAR CORRESPONDANCES:

The epipolar line for each point is calculated as follows:

$L_i=F*P_i$

$L_i=[[a_i],[b_i],[c_i]]$ where a,b and c are cooefficients of the epipolar line equation.

Where $P_i=[[x],[y],[1]]$

HOW TO FIND CANDIDATES POINTS:

The two end points of the epipolar line is computed as follows:

Case 1: If b is not equal to zero,it means the line covers whole of the x axis of the image.

So we get y values for each end points by looping by substituting x=0 and x=width in the equation of epipolar line:

y=int(-(ax+c)/b)

In all other cases:

We get values of x coordinate of end points by substituting y=0 and y=height in the equation of epipolar line:

x=int(-(by+c)/a)

xs,ys=end point1

xe,ye=end point2

now we use a for loop to loop from xs+ width//2 to xe+1- width//2, compute y coordinate for each point by substituting in the equation of epipolar line and adding points from y-1 to y+2 for each x coordinate to account for any error that may occur due to rounding off the y coordinate from the equation as follows:

y=int(-(ax+c)/b)

the range of x is taken by leaving a space of int(w/2) on each side of the image as looping from x=0 to x=width would cause error in making windows of width w as there are no pixels towards one side of each end points.

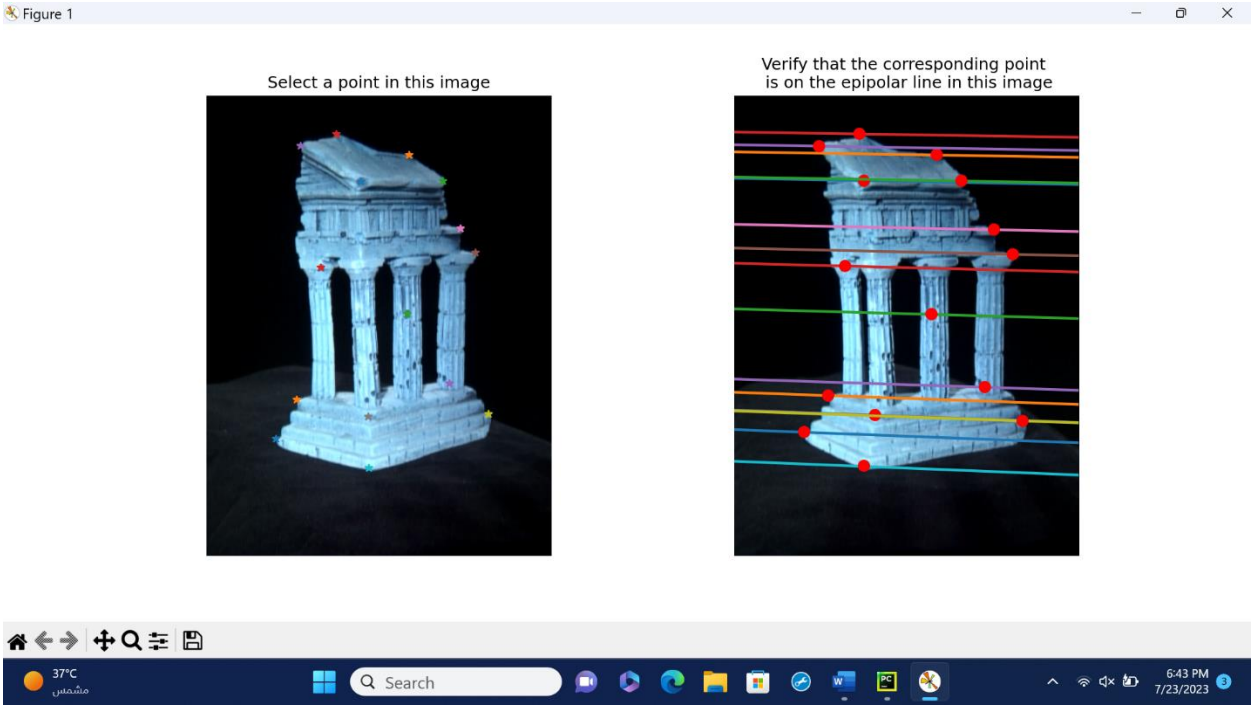
CHECKING SIMILARITY:

After this is done we get an array of x,y coordinates of candidates and we find the best candidate using the window method.

The idea is to take a small window around the desired point and compare it with the window of same size around the candidate points i.e., the points on the epipolar line.

The cost is computed by using Euclidian distance as the metric.

The square root of sum of square of difference in intensity of each pixel in the respective windows is the cost and the cost is minimized to find the right candidate for a given point.



Taking the window size as small might give wrong epipolar correspondence when the image has mulpiple similar looking areas in a small range.

But taking a large window size would increase the computational cost as the area to check similarity for each point on epipolar line is larger.

ESSENTIAL MATRIX

With our computed fundamental matrix, we can find the essential matrix using the formula:

$E=K_2^T*F*K_1$

Essential matrix makes the computation scale invariant.

It is used when the pair of corresponding points are normalized or in the simplified or canonical case:

P1=K1*pc1

P2=K2*pc2

Where pc1 and pc2 are points 1 and 2 in normalized form.

Substituting this in the epipolar equation.

$P2^T*F*p1=Pc2^T*E*pc1$

By substituting the above relations we get the relation between E and F.