CS543/ECE549 Assignment 5

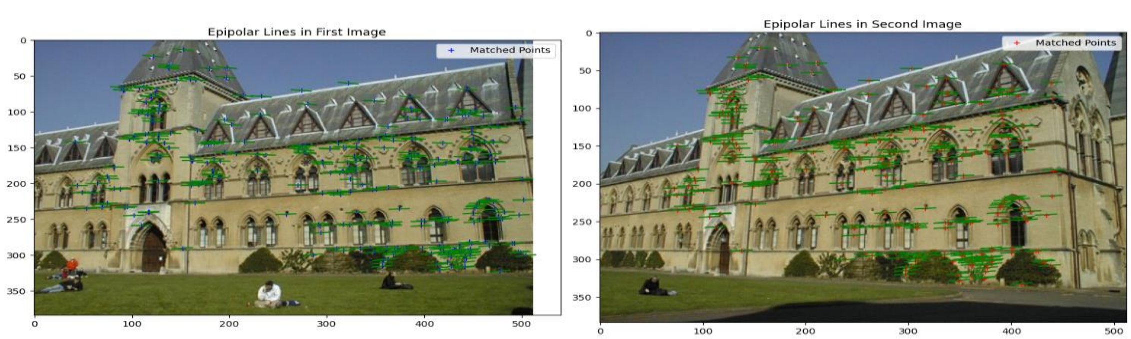
Your Name: Sai Rohit Muralikrishnan
Your NetId: srm17

Part 1: Fundamental Matrix Estimation, Camera Calibration, Triangulation

1. For the **lab** and **library** image pairs, display your result (points and epipolar lines) and report your residual for both unnormalized and normalized fundamental matrix estimation.

Image pair 1:

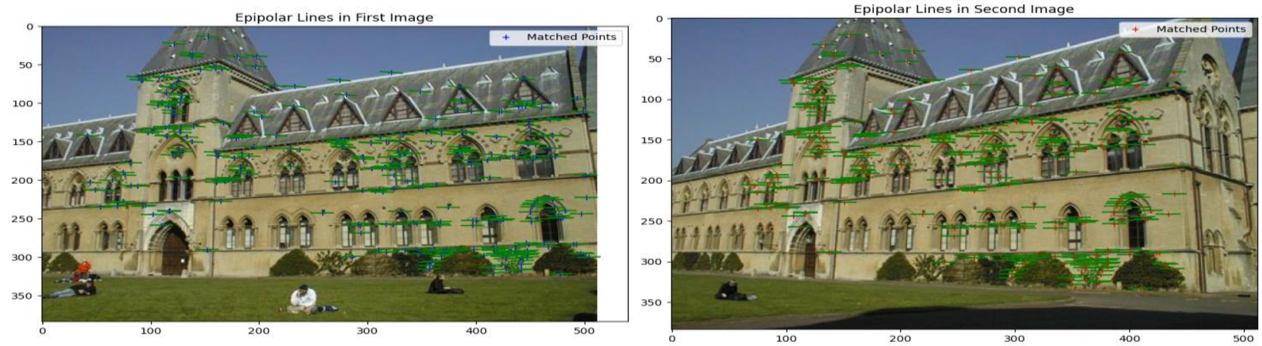
Un – normalized:



Fundamental matrix:

```
[[-1.52512494e-10  1.57466615e-09 -8.10620100e-06]
 [-3.32100917e-09  3.07050444e-11  4.71669930e-04]
 [ 5.69875548e-05 -3.73802735e-04 -1.03111520e+00]]
```

Normalized:

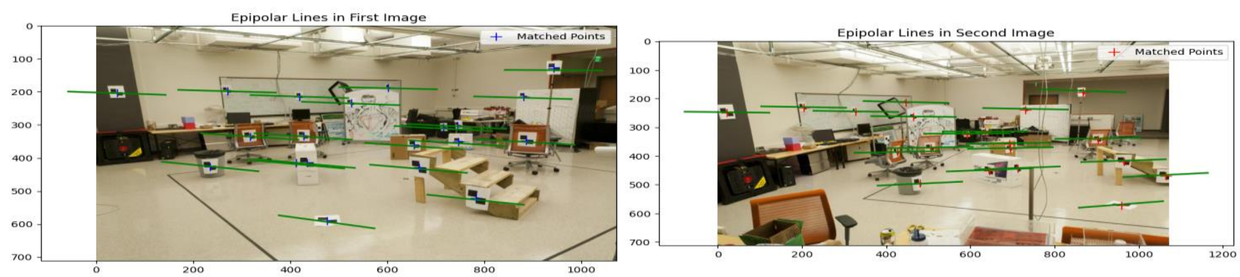


Fundamental matrix:

```
[[-3.66106207e-08  7.80422368e-07 -1.18097936e-04]
 [-4.73752936e-06 -4.86861294e-08  8.79648624e-03]
 [ 1.13070861e-03 -7.90017674e-03 -2.13969289e-01]]
```

Image pair 2:

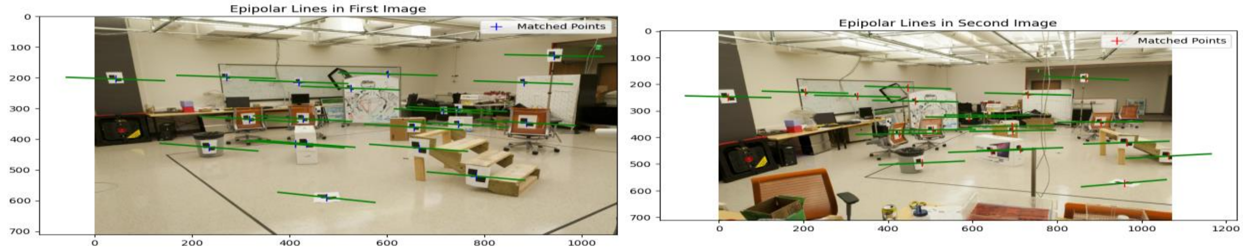
Un – normalized:



Fundamental matrix:

```
[[-1.92012586e-11  2.82994808e-10 -1.14625257e-05]
 [ 3.16356461e-10  4.34399510e-11  1.03803645e-04]
 [-5.48182167e-06 -1.56972242e-04  1.02461732e+00]]
```

Normalized:



Fundamental matrix:

```
[[-1.35187922e-07  1.85402375e-06 -4.63400019e-04]
 [ 1.28179467e-06 -3.15513089e-07  3.72713590e-03]
 [-2.71248547e-05 -5.12276341e-03  1.19199548e-01]]
```

2. For the **lab** image pair, show your estimated 3x4 camera projection matrices. Report the residual between the projected and observed 2D points.

Image 1:

```
Projection matrix P0:
[[-3.09963996e-03 -1.46204548e-04  4.48497465e-04  9.78930678e-01]
 [-3.07018252e-04 -6.37193664e-04  2.77356178e-03  2.04144405e-01]
 [-1.67933533e-06 -2.74767684e-06  6.83964827e-07  1.32882928e-03]]
```

```
Camera 0: residual = 13.545832894859183 , mse = 0.3930761430020345
```

Image 2:

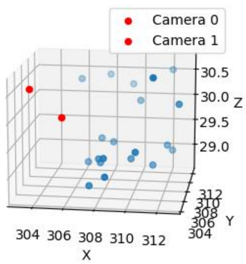
```
Projection matrix P1:
[[ 6.93154686e-03 -4.01684470e-03 -1.32602928e-03 -8.26700554e-01]
 [ 1.54768732e-03  1.02452760e-03 -7.27440714e-03 -5.62523256e-01]
 [ 7.60946050e-06  3.70953989e-06 -1.90203244e-06 -3.38807712e-03]]
```

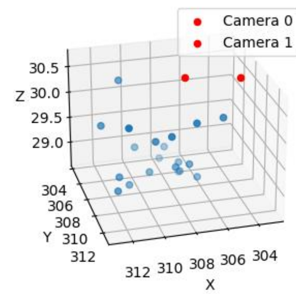
```
Camera 1: residual = 15.544953460643853 , mse = 0.3728337540009026
```

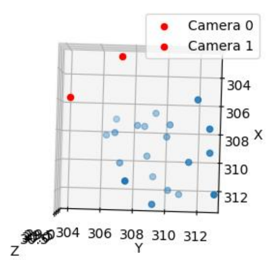
3-4. For the **lab** and **library** image pairs, visualize 3D camera centers and triangulated 3D points.

Lab image pairs:

Camera Center 0: [[7.28863053 -21.52118112 17.73503585]]
]
Camera Center 1: [[6.89405488 -15.39232716 23.41498687]]

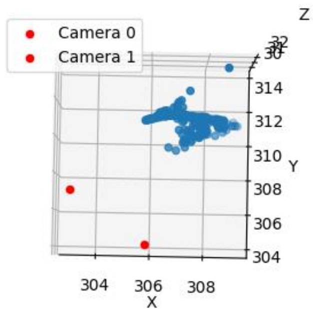


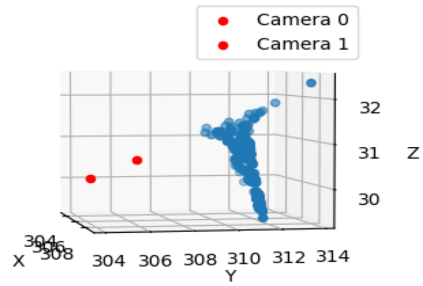


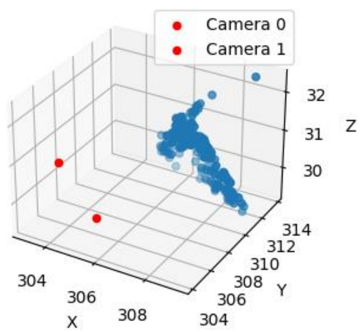


Library image pairs:

Camera Center 0: [305.83276769 304.20103826 30.13699243   1.   ]
Camera Center 1: [303.10003925 307.18428016 30.42166874   1.   ]







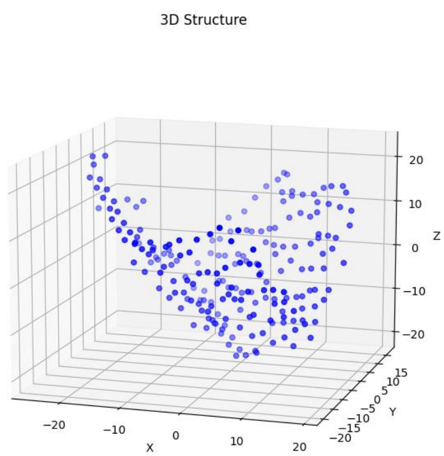
5. For the **house** and **gaudi** image pairs, display your result and report your number of inliers and average inlier residual for normalized estimation without ground truth matches.

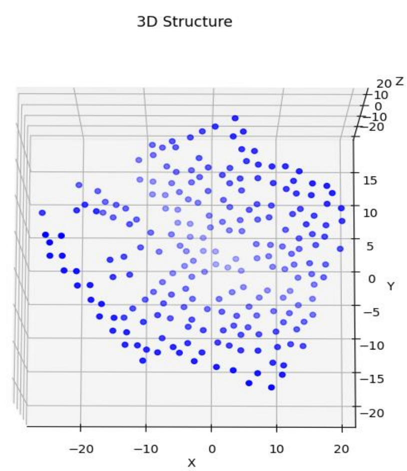
Part 1 Extra Credit (optional)

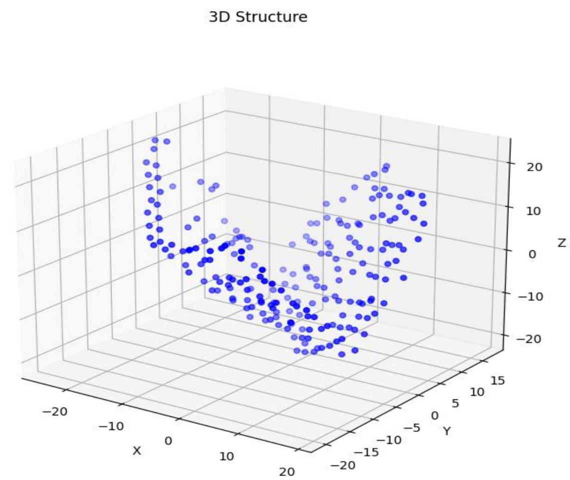
Part 2: Affine factorization

A: Display the 3D structure (you may want to include snapshots from several viewpoints to show the structure clearly). Report the Q matrix you found to eliminate the affine ambiguity. Discuss whether or not the reconstruction has an ambiguity.

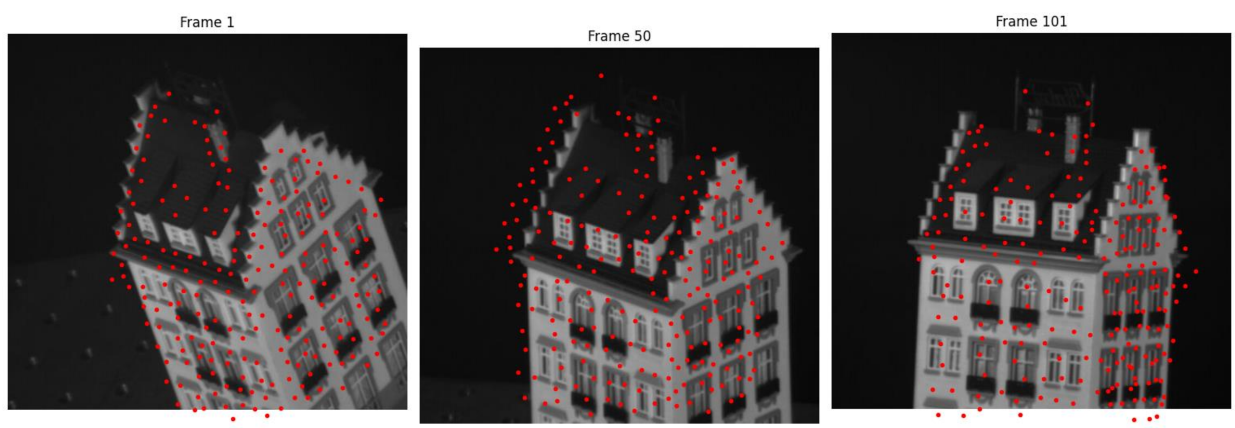
```
Q Matrix:
[[0.80930751 0.         0.        ]
 [0.00377312 0.86633144 0.        ]
 [0.04112798 0.01759367 0.27851645]]
```



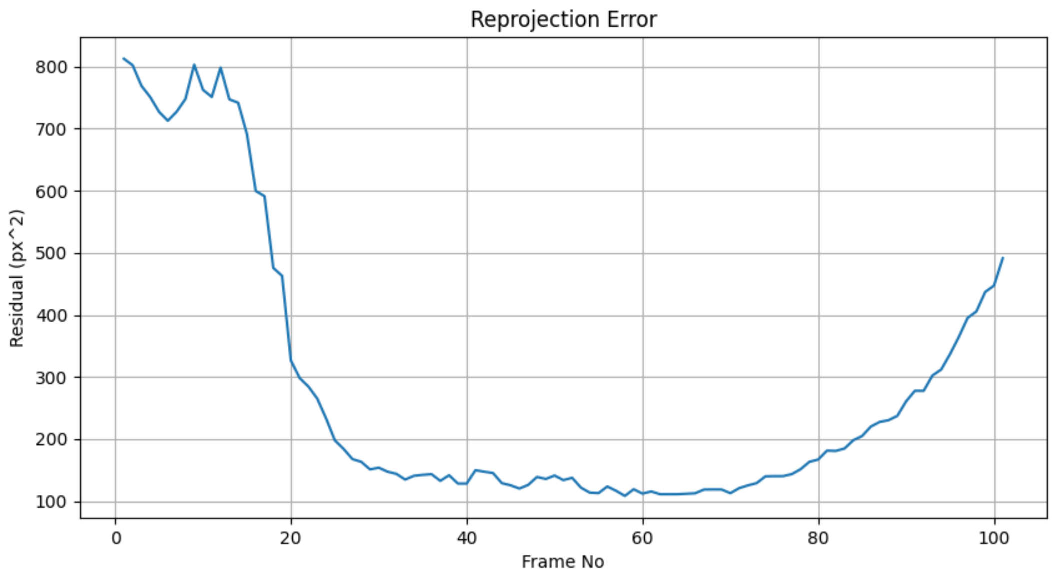




B: Display three frames with both the observed feature points and the estimated projected 3D points overlayed.



C: Report your total residual (sum of squared Euclidean distances, in pixels, between the observed and the reprojected features) over all the frames, and plot the per-frame residual as a function of the frame number.



Part 2 Extra Credit (optional)

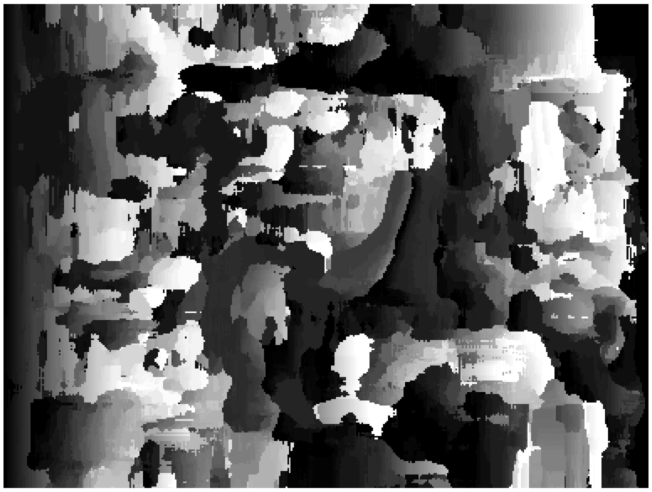
Part 3: Binocular stereo

A: Display best output disparity maps for both pairs.


B: Study of implementation parameters:

1. Search window size: show disparity maps for several window sizes and discuss which window size works the best (or what are the tradeoffs between using different window sizes). How does the running time depend on window size?

Window size = 5
Runtime = 27.69s



Window size = 15
Runtime = 24.11s

Window size = 25
Runtime = 32.51s

When using a smaller window size, the disparity map retains more fine details and object boundaries appear sharper, but the results are often noisier and less stable. In contrast, larger window sizes smooth out noise and provide more stable matches, but at the cost of losing detail and producing less accurate depth boundaries.

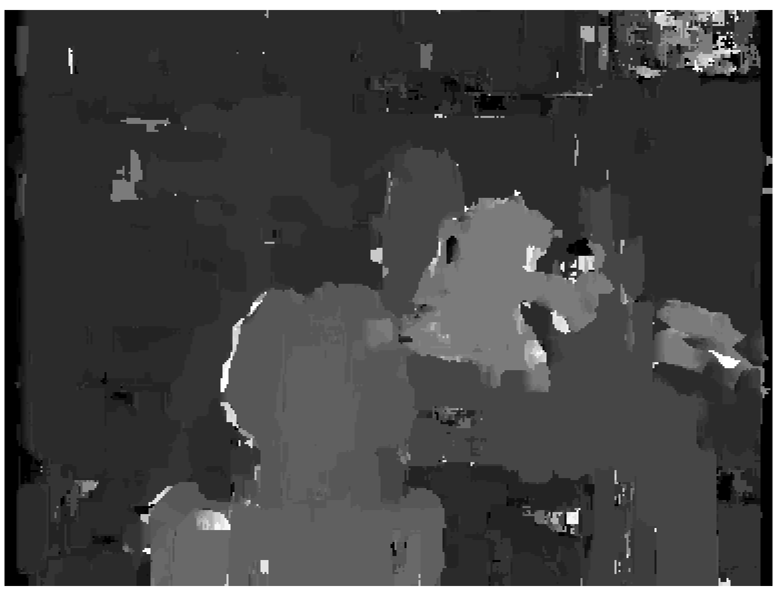
In terms of running time, larger windows require more computations per pixel since there are more pixels to compare, leading to slower processing. Smaller windows reduce the computational load per pixel, but since the results can be noisier, additional post-processing may be required, potentially affecting overall runtime.

2.) Disparity range: what is the range of the scanline in the second image that should be traversed in order to find a match for a given location in the first image? Examine the stereo pair to determine what is the maximum disparity value that makes sense, where to start the search on the scanline, and which direction to search in. Report which settings you ended up using.

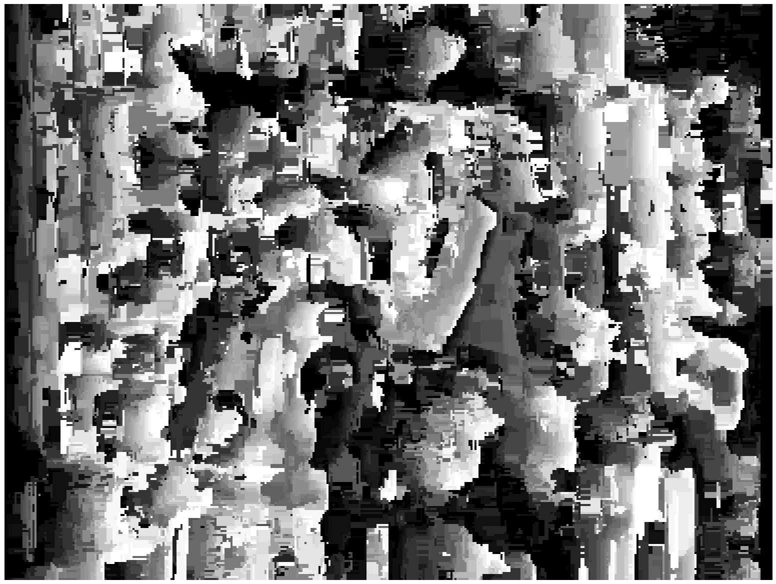
The maximum disparity should be chosen based on the camera's baseline and the closest objects in the scene. Typically, values around 15–20 pixels might suffice for moderate scenes, but using a higher value (like 30–40) ensures capturing objects that are very close. For the given dataset, I ended up using a maximum disparity of 26 pixels. The search was performed from left to right on the scanline of the right image.

2. Matching function: try sum of squared differences (SSD), sum of absolute differences (SAD), and normalized correlation. Show the output disparity maps for each. Discuss whether there is any difference between using these functions, both in terms of quality of the results and in terms of running time.

SSD



NCC



SAD

Among SSD, SAD, and NCC, SSD produced the best disparity maps with sharper edges and better accuracy. SAD performed worse with larger window sizes due to increased blending, while NCC, though robust to intensity variations, was the slowest. SSD balanced quality and runtime efficiency effectively, making it the most practical choice. NCC’s higher computation time limited its usefulness despite decent results.

C: Discuss the shortcomings of your algorithm. Where do the estimated disparity maps look good, and where do they look bad? What would be required to produce better results? Also discuss the running time of your approach and what might be needed to make stereo run faster.

The algorithm struggles with accuracy at object edges because its fixed-size window mixes pixels from the foreground and background, making edges blurry. It also has trouble in areas without much texture or with repeating patterns, leading to noisy and unreliable results. While it works well on surfaces with good texture that are parallel to the camera and close by, it performs poorly in areas with sudden depth changes or few distinct features.

To improve results, techniques like using adjustable window sizes, optimizing globally with methods like dynamic programming, handling occlusions, and applying post-processing filters can help. For faster processing, approaches like using integral images, dynamic programming, multi-scale search (pyramid search), or GPU acceleration can speed up computations

Part 3 Extra Credit (optional)