16-385 Computer Vision, Spring 2025

Programming Assignment 3
3D Reconstruction

Belle Connaught (bconnaug)

Due Date: Mon March 17, 2025 23:59 (ET)

2 Theory Questions

Q2.1 Essential matrices and epipolar lines

As we discussed in class, two cameras are said to form a *rectified* pair if their camera coordinate systems differ only by a translation of their origins (the camera centers) along a direction that is parallel to either the x or y axis of the coordinate systems.

1. Derive an expression for the essential material $\mathbf{E}$ of the rectified pair.

   A rectified camera pair has camera coordinate systems that differ only by a translation along one of the coordinate axes (either x or y). Given two cameras with projection matrices:

   $$P_1 = [I|0],$$
   $$P_2 = [I|\boldsymbol{t}]$$

   where $\boldsymbol{t} = (t_x, 0, 0)^T$ for translation along the x-axis (or $\boldsymbol{t} = (0, t_y, 0)^T$ for translation along the y-axis), the essential matrix is defined as:

   $$E = [\boldsymbol{t}]_\times R$$

   Since rectified cameras have no rotation ($R = I$), we obtain:

   $$E = [\boldsymbol{t}]_\times$$

   where $[\boldsymbol{t}]_\times$ is the skew-symmetric matrix of the translation vector:

   $$[\boldsymbol{t}]_\times = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix}$$

For a rectified pair with translation along the x-axis ($\boldsymbol{t} = (t_x, 0, 0)^T$), this simplifies to:

$$E = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -t_x \\ 0 & t_x & 0 \end{bmatrix}$$

Similarly, for translation along the y-axis ($\boldsymbol{t} = (0, t_y, 0)^T$), we obtain:

$$E = \begin{bmatrix} 0 & 0 & t_y \\ 0 & 0 & 0 \\ -t_y & 0 & 0 \end{bmatrix}$$

2. Prove that the epipolar lines of a rectified pair are parallel to the exis of translation.

The epipolar line in the second image corresponding to a point $\boldsymbol{x}_1$ in the first image is given by:

$$\boldsymbol{l}_2 = E\boldsymbol{x}_1$$

For translation along the x-axis ($\boldsymbol{t} = (t_x, 0, 0)^T$), we substitute E:

$$\boldsymbol{l}_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -t_x \\ 0 & t_x & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ -t_x \\ t_x y_1 \end{bmatrix}$$

Since the epipolar line equation is of the form $ax + by + c = 0$, this simplifies to:

$$-t_x y + t_x y_1 = 0 \quad \Rightarrow \quad y = y_1$$

which shows that the epipolar line is horizontal (parallel to the x-axis). Similarly, for translation along the y-axis, the epipolar lines are vertical (parallel to the y-axis).

Thus, the epipolar lines of a rectified camera pair are always parallel to the axis of translation.

Q2.2 Fun with fundamental matrices

1. Write an expression for x_3 in terms of x_1, x_2, F_{23} and F_{23}.

The fundamental matrix F_{ij} satisfies the epipolar constraint for corresponding points:

$$x_j^T F_{ij} x_i = 0$$

Therefore, we have the following (meaning that the point $\mathbf{x}_3$ must lie on the epipolar lines in I_3, given by $F_{13}x_1$ and $F_{23}x_2$):

$$x_3^T F_{13} x_1 = 0$$
$$x_3^T F_{23} x_2 = 0$$

Said lines can be defined as:

$$i_{3,1} = F_{13} x_1$$
$$i_{3,2} = F_{23} x_2$$

$\mathbf{x}_3$ must lie at the intersection of both of these equations, which can be represented as the cross product of the two (i.e. $i_{3,1} \times i_{3,2}$). Therefore, we have:

$$\boxed{x_3 = F_{13}x_1 \times F_{23}x_2}$$

2. Describe a degenerate configuration of three cameras for which the point cannot be uniquely determined by this expression.

   A degenerate configuration of three cameras for which x_3 cannot be uniquely determined by this expression occurs when cameras one and two are parallel to each other, since their epipolar lines would not intersect.

3 Sparse Reconstruction

3.1 Implement the Eight Point Algorithm

Recovered F:

$$F = \begin{bmatrix} 6.79315012 * 10^{-10} & -5.84895627 * 10^{-8} & 2.06431754 * 10^{-6} \\ -4.07094625 * 10^{-8} & 5.37087279 * 10^{-10} & 5.89662500 * 10^{-4} \\ 8.13063418 * 10^{-6} & -5.67222237 * 10^{-4} & -2.33090044 * 10^{-3} \end{bmatrix}$$

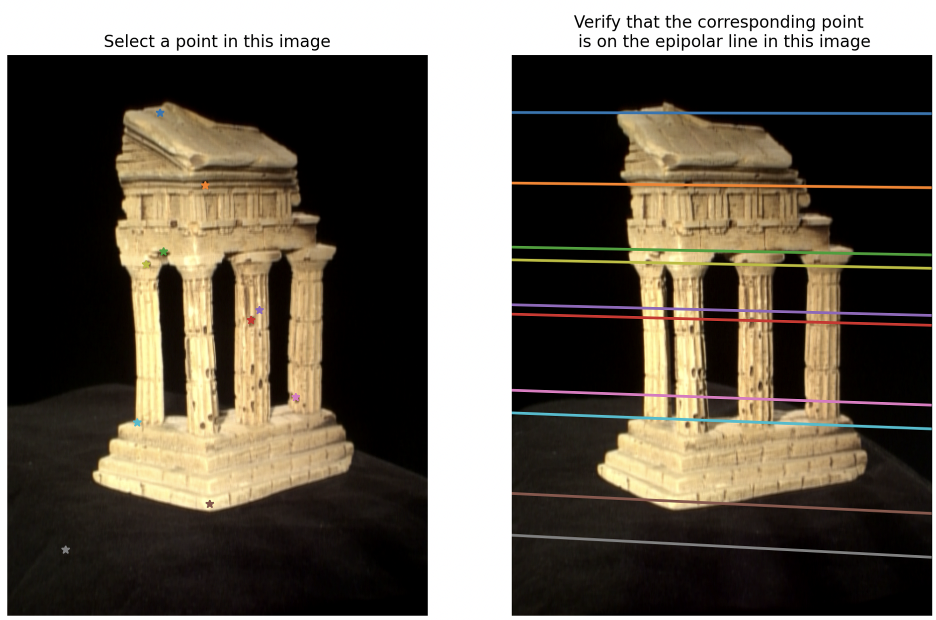

Figure 1: Epipolar lines visualization from `displayEpipolarF`

3.2 Find Epipolar Correspondences

I used Euclidean distance as my similarity metric. For each point in image 1, I extracted a small patch (5×5 pixels) and compared it with candidate patches along the epipolar line in image 2. The patch with the smallest Euclidean distance was chosen as the best match.

My algorithm works well for distinct features such as corners and high-contrast edges but struggles in smooth regions and repetitive patterns. In featureless areas, many patches look alike, leading to incorrect matches. Similarly, in repetitive textures (e.g., building windows), the function often selects the wrong match because multiple candidates have similar scores. The method also fails when perspective distortion is significant since simple window comparisons do not account for warping.

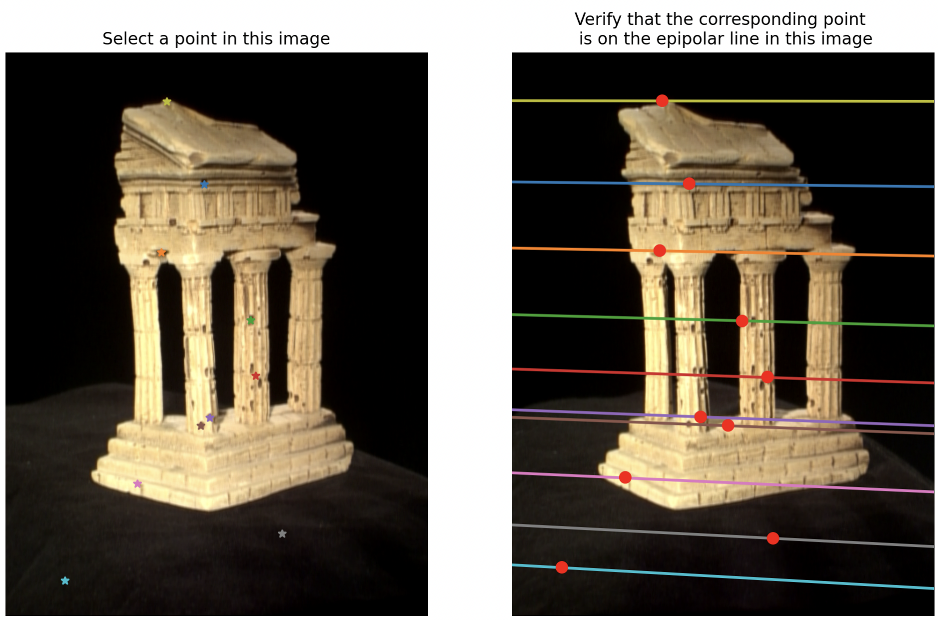

Figure 2: Epipolar Match visualization from `epipolarMatchGUI`

3.3 Write a function to compute the essential matrix

Essential Matrix:

$$E = \begin{bmatrix} 0.00157032 & -0.13569452 & -0.01850271 \\ -0.09444507 & 0.00125054 & 0.88118864 \\ -0.00260587 & -0.89230392 & -0.00101771 \end{bmatrix}$$

3.4 Implement Triangulation

To determine the correct extrinsic matrix M2 for triangulation, I evaluated the four matrices generated by the camera2 function. The correct matrix is the one that results in the most 3D points with positive depth (i.e., the points are in front of both cameras). This ensures that the points are valid and correspond to a physically feasible scene.

$\boxed{\text{Reprojection error} = 2.2802589480326256}$

3.5 Write a Test Script that uses `data/templecoords.npz`

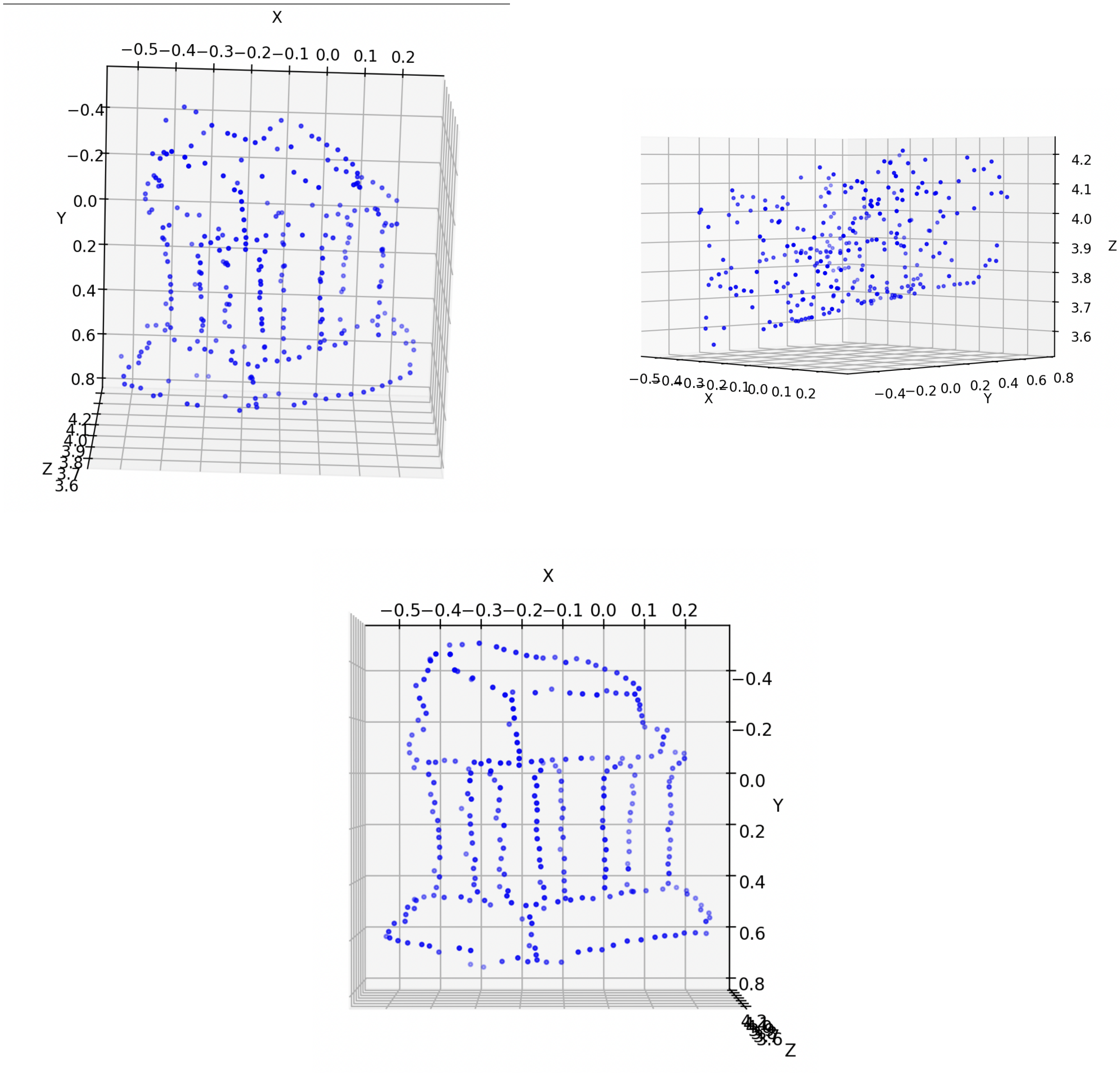

Figure 3: Sample Reconstructions

4 Dense Reconstruction

Q4.1 Image Rectification

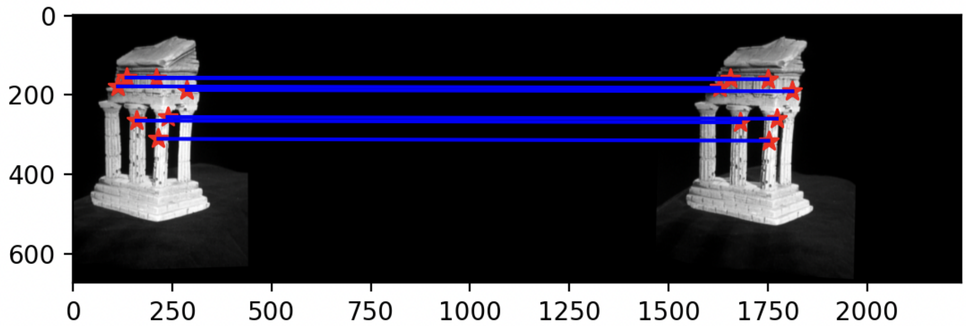

Figure 4: Result of `test_rectify.py`

Q4.3 Depth Map

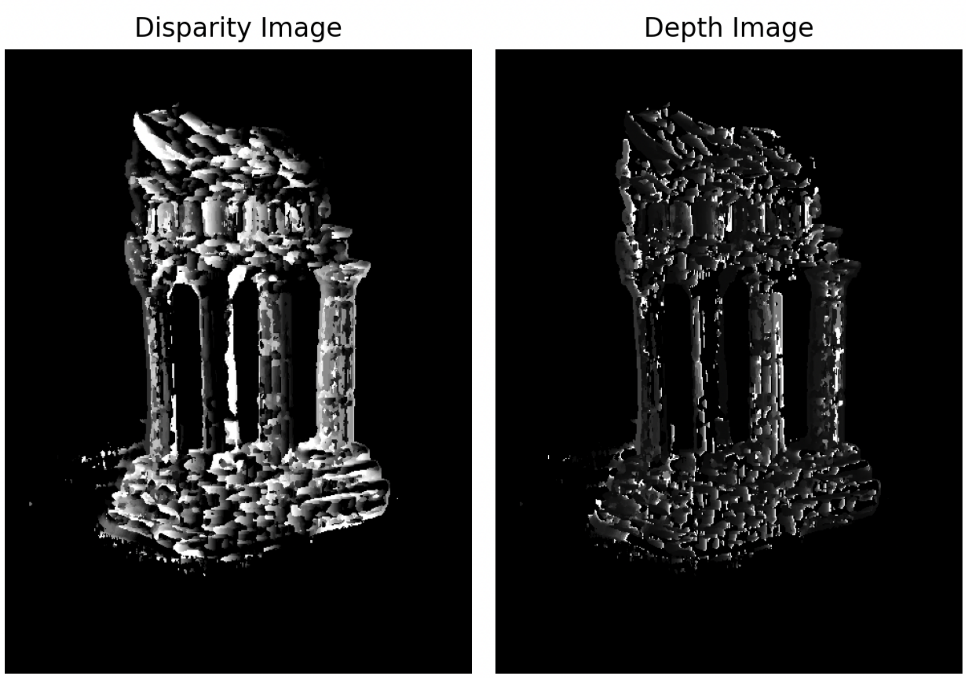

Figure 5: Disparity Map