

2 Theory

Q2.1 Triangulation

Using Figure 3 in the assignment guideline, we first compute d by sine law.

$$\frac{b}{\sin \gamma} = \frac{d}{\sin \alpha} \implies d = \frac{b \sin \alpha}{\sin(\alpha + \beta)}$$

Then, we can compute $(x, z) = (d \sin \beta, d \cos \beta) = \left(\frac{b \sin \alpha \sin \beta}{\sin(\alpha + \beta)}, \frac{b \sin \alpha \cos \beta}{\sin(\alpha + \beta)} \right)$.

Q2.2 Fundamental Matrix

The epipolar constraint of this setup can be written as

$$\begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \mathbf{F} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = 0 \implies \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{F}_{11} & \mathbf{F}_{12} & \mathbf{F}_{13} \\ \mathbf{F}_{21} & \mathbf{F}_{22} & \mathbf{F}_{23} \\ \mathbf{F}_{31} & \mathbf{F}_{32} & \mathbf{F}_{33} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = 0 \implies \mathbf{F}_{33} = 0.$$

3 Programming

Q3.1 Sparse Reconstruction

Q3.1.1 Implement the eight point algorithm

$$\mathbf{F} = \begin{bmatrix} -0.0000 & 0.0000 & -0.0000 \\ 0.0000 & -0.0000 & -0.0015 \\ -0.0000 & 0.0015 & 0.0064 \end{bmatrix}$$

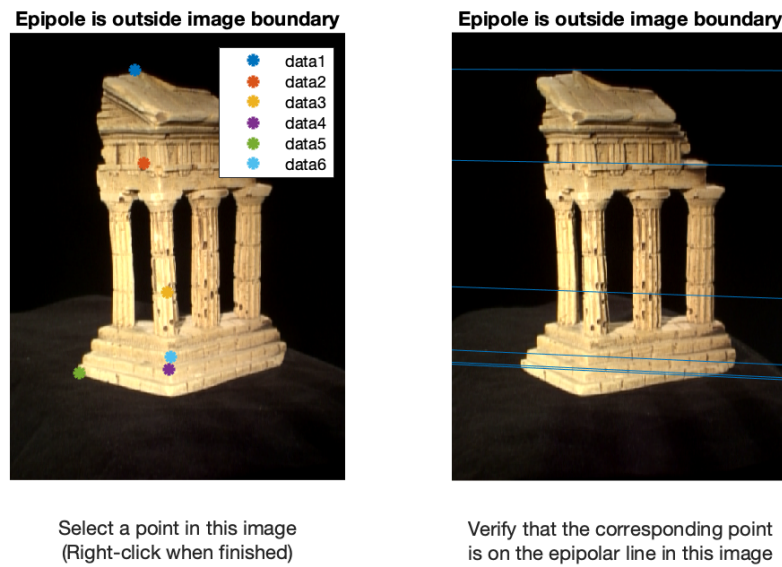


Figure 1: Visualization of some epipolar lines

Q3.1.2 Find the epipolar correspondences

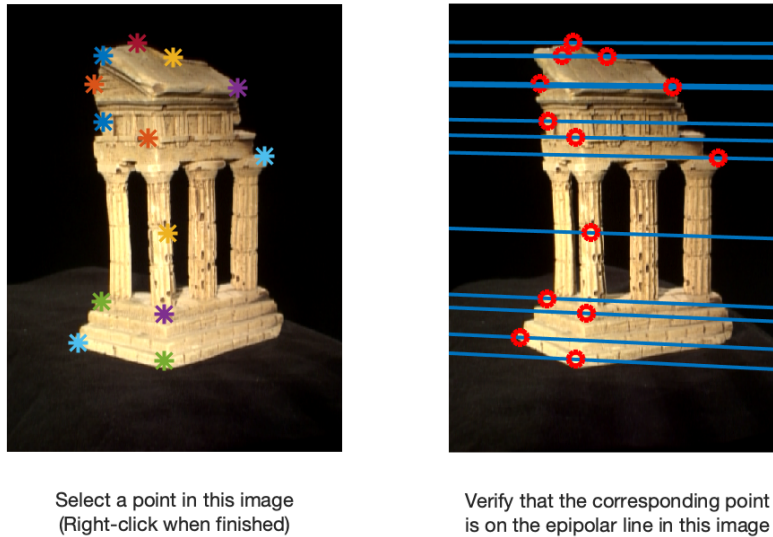


Figure 2: Screenshot of `epipolarMatchGui`

The similarity metric used is the reciprocal of the Manhattan distance between a target window of image 1 and a candidate window of image 3, with window size of 7. In most cases our matching algorithm succeeds, but it may fail when there are highly similar but unmatched windows along the epipolar line.

Q3.1.3 Write a function to compute the essential matrix

$$\mathbf{E} = \begin{bmatrix} -0.0025 & 0.4070 & 0.0476 \\ 0.1863 & 0.0127 & -2.2833 \\ 0.0076 & 2.3114 & 0.0026 \end{bmatrix}$$

Q3.1.4 Implement triangulation

The correct extrinsic matrix is determined by first computing all 4 sets of 3D points with the 4 candidate extrinsic matrices, then for each set we count how many points have a positive third (depth) coordinate - that is $[0, 288, 249, 39]$, finally the candidate with the highest count is the correct extrinsic matrix - that is index 2.

The re-projection error for `pts1` is 0.5664, for `pts2` is 0.5711.

Q3.1.5 Write a test script that uses `templeCoords`

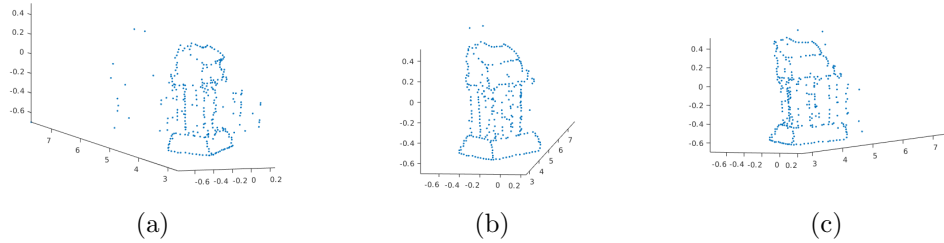


Figure 3: Three images of final reconstruction from different angles

Q3.1 Dense Reconstruction

Q3.2.1 Image Rectification

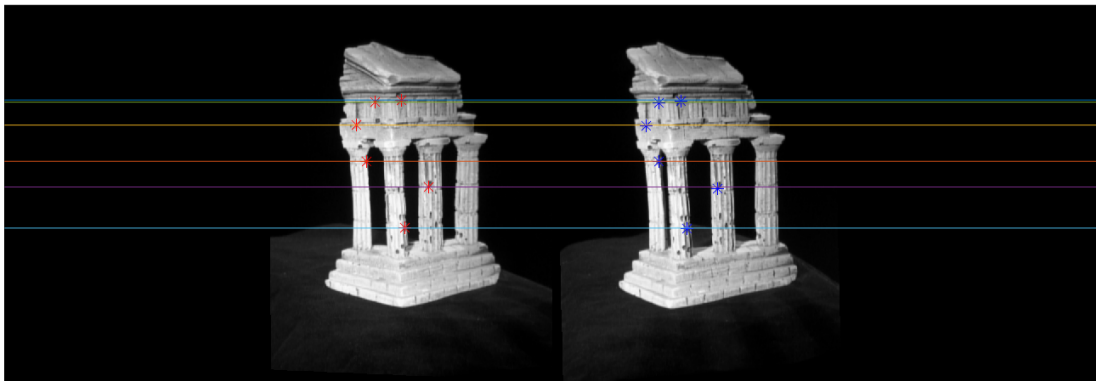


Figure 4: Rectified stereo images.

Q3.2.3 Depth map (extra credit)

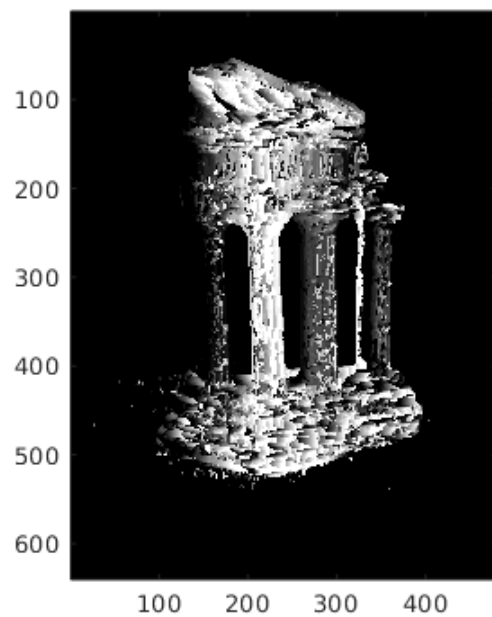


Figure 5: Disparity map.

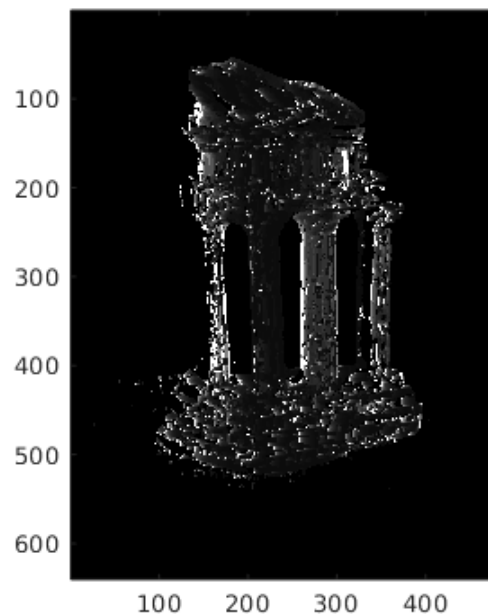


Figure 6: Depth map.

Q3.3 Post Estimation (Extra Credit)

Q3.3.1 Estimate camera matrix P

Reprojected Error with clean 2D points is 0.0000

Pose Error with clean 2D points is 0.0000

Reprojected Error with noisy 2D points is 4.5648

Pose Error with noisy 2D points is 0.7531

Q3.3.2 Estimate intrinsic/extrinsic parameters

Intrinsic Error with clean 2D points is 0.0000

Rotation Error with clean 2D points is 0.0000

Translation Error with clean 2D points is 0.0000

Intrinsic Error with noisy 2D points is 1.6370

Rotation Error with noisy 2D points is 1.9826

Translation Error with noisy 2D points is 4.5096

Q3.3.3 Project a CAD model to the image



Figure 7: Image annotated with 2D points (green) and projected 3D points (black).

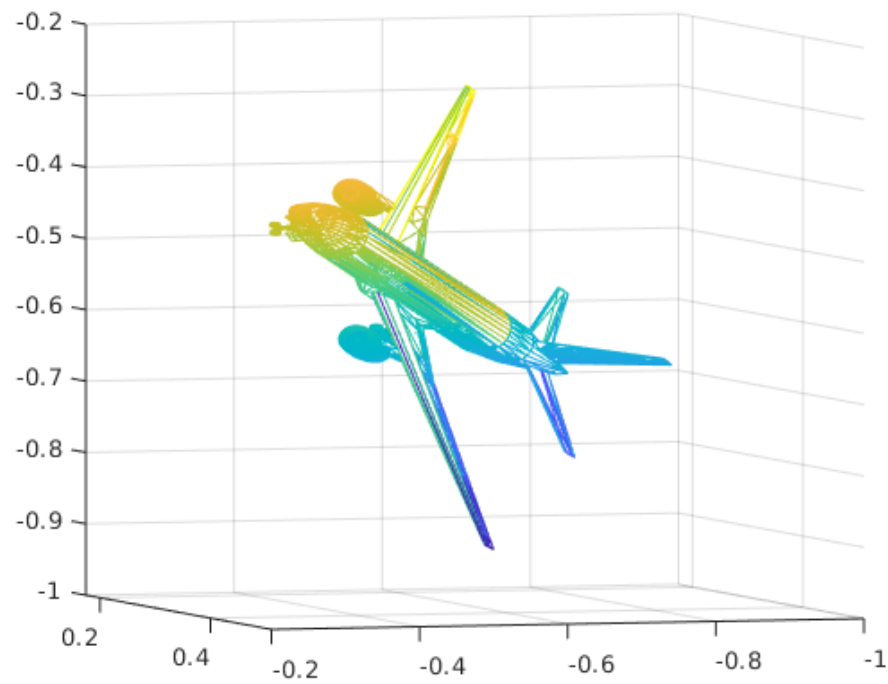


Figure 8: CAD model rotated by \mathbf{R} .



Figure 9: Image overlapped with projected CAD model.