16-385 Computer Vision, Fall 2019

Homework 5
3D Reconstruction

Due Date: Wed November 20, 2019 23:59

1 Overview

2 Theory

2.1 Triangulation

We can divide b as b_1 and b_2. Each is like below figure.
The height of triangle can be expressed by α, β and b_1 and b_2. . It is part of z.

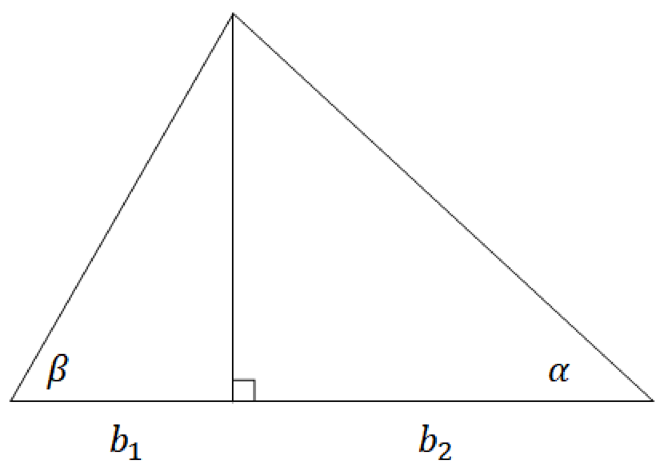

Figure 1: The figure of triangle

$$h = b_1 tan\beta = b_2 tan\alpha$$

$$\begin{aligned} b &= b_1 + b_2 \\ &= \frac{h}{tan\beta} + \frac{h}{tan\alpha} \\ &= h \times \frac{tan\alpha + tan\beta}{tan\alpha \times tan\beta} \end{aligned}$$

$$h = b \times \tfrac{tan\alpha \times tan\beta}{tan\alpha + tan\beta}$$

The part of x is expressed by b_1.

$$\begin{aligned} x &= b_1 \\ &= \frac{h}{tan\beta} \\ &= \frac{b \times tan\alpha}{tan\alpha + tan\beta} \end{aligned}$$

So the (x,z) is like below.

$$(x,z) = (\tfrac{b \times tan\alpha}{tan\alpha + tan\beta}, b \times \tfrac{tan\alpha \times tan\beta}{tan\alpha + tan\beta})$$

2.2 Essential Matrix

R as the rotation matrix and t as the translation vector. In this case, we can denote $R = I$ because it is rectified images.

$$\begin{aligned} t &= \begin{bmatrix} T & 0 & 0 \end{bmatrix} \\ R &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

$$\begin{aligned} E &= t \times R \\ &= t \times I \\ &= \begin{bmatrix} T & 0 & 0 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -T \\ 0 & T & 0 \end{bmatrix} \end{aligned}$$

By Longuet-Higgins equation, we can denote $x'^T E x = 0$. In this case, x' is $x - t$.

$$\begin{aligned} x'^T E x &= [x'y'1] \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -T \\ 0 & T & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \\ &= -Ty' + Ty \\ &= 0 \end{aligned}$$

So, $y = y'$. x and x' have same y value.
The epipolar line of a rectified pair are parallel to the axis of translation vector t.
An expression for the essential matrix **E** of the rectified pair is $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -T \\ 0 & T & 0 \end{bmatrix}$.

2.3 Fundamental Matrix

By the Longuet-Higgins equation, we can denote $x_1^T F x_2 = 0$. In this case, the camera 1's point is x_1 and the camera 2's point is x_2. We can define $x_1 = (0, 0, z_1)$, $x_2 = (0, 0, z_2)$ and $F = \begin{bmatrix} F_{11} & F_{12} & F_{13} \\ F_{21} & F_{22} & F_{23} \\ F_{31} & F_{32} & F_{33} \end{bmatrix}$.

Apply this to equation, then

$$\begin{aligned} x_1^T F x_2 &= \begin{bmatrix} 0 & 0 & z_1 \end{bmatrix} \begin{bmatrix} F_{11} & F_{12} & F_{13} \\ F_{21} & F_{22} & F_{23} \\ F_{31} & F_{32} & F_{33} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ z_2 \end{bmatrix} \\ &= z_1 F_{33} z_2 \\ &= 0 \end{aligned}$$

So, we can know $F_{33} = 0$.

3 Programming

3.1 Sparse Reconstruction

3.1.1 Implement the eight point algorithm

$$\text{The } F \text{ result is } F = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0.0011 \\ 0 & 0.0011 & 0.0044 \end{bmatrix}$$

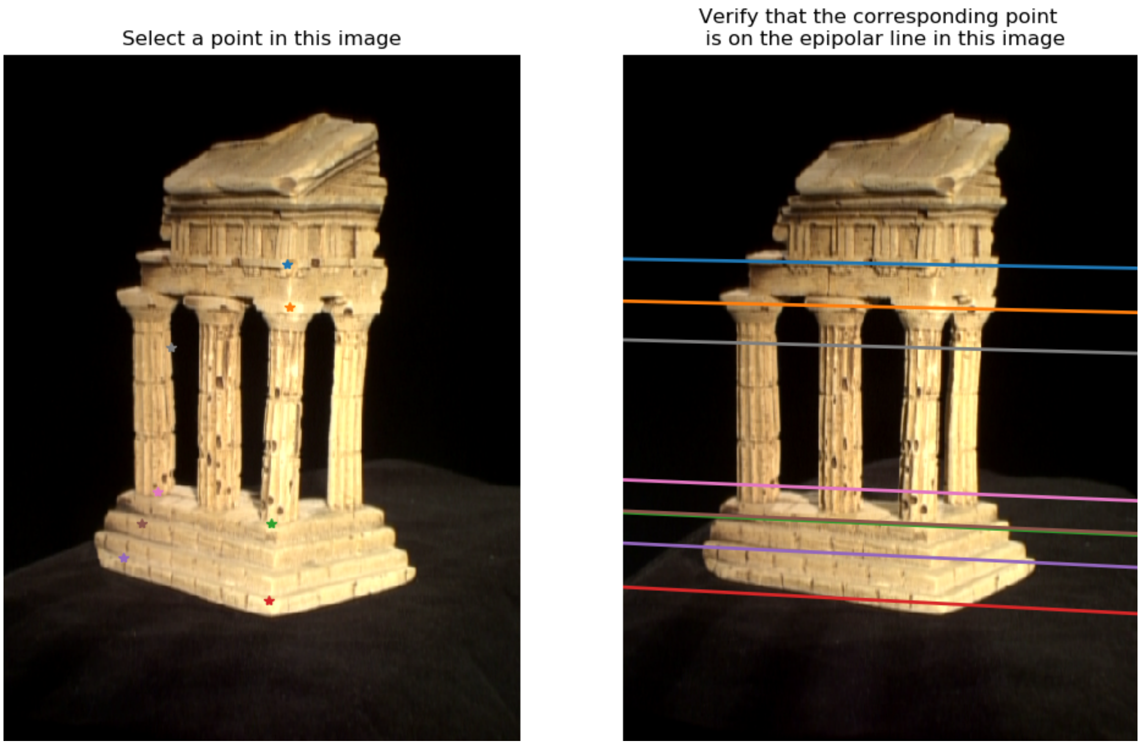

Figure 2: The result of epipolar lines

3.1.2 Find epipolar correspondences

The similarity metric which is used is checking the difference between patches. First, we select the big

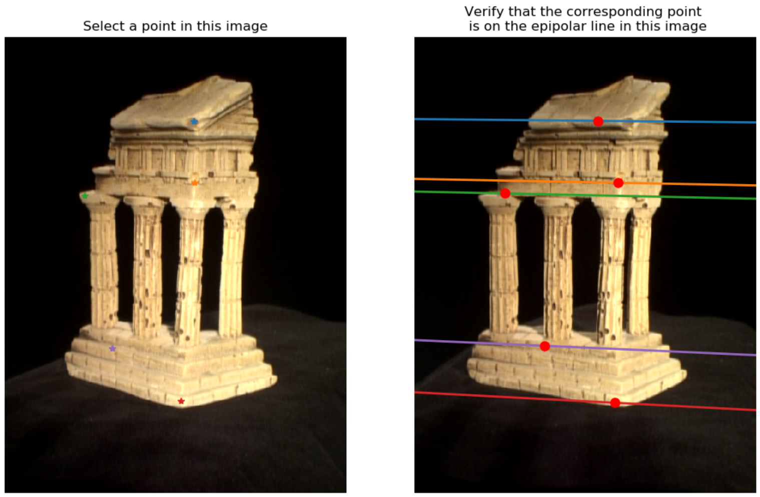

Figure 3: The result of Epipolar Match visualization

window based on image 1's points. The big window is selected like below image. The point is one point of image 1 and window width of left part is 70 and right part is 30. That value is selected empirically. Inside

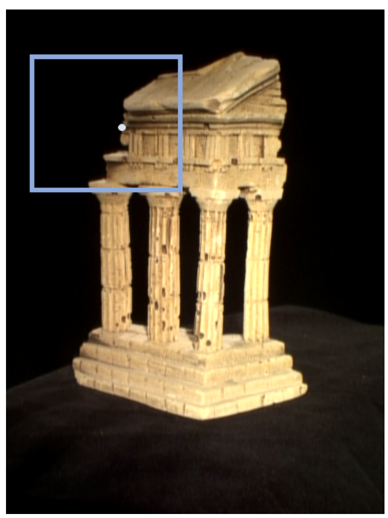

Figure 4: The big window of image 1

this window, we select little window. This window size is 30. By sliding this window, as like as convolution, we select patch at image 1. Also, we select corresponding patch at image 2. And get difference between two patches. Through this differences, we choose the index of minimum difference.
If matching algorithm consistently fails, the probability of the corresponding line is calculated wrong is high. If the corresponding line is calculated wrong, the patch is selected by wrong way. So the result will be failed.

3.1.3 Write a function to compute the essential

$$\text{The } E \text{ result is } E = \begin{bmatrix} -0.0037 & 0.2807 & 0.0403 \\ 0.1577 & 0.0001 & -1.6965 \\ 0.0003 & 1.7177 & 0.002 \end{bmatrix}$$

3.1.4 Implement triangulation

I determine which extrinsic matrix is correct by all of z value is greater or equal to 0. It means the distance between camera and object is positive. So we need to have positive distance, I choose that method.
In this case, the reprojection error is 0.3301. However, the error is similar at 4 cases of the result of camera function. In my opinion, it is nature result, because just we change the camera's location but the distance is same.

3.1.5 Write a test script that uses temple coords npz data

The xy plane of the reconstruction result is like below.

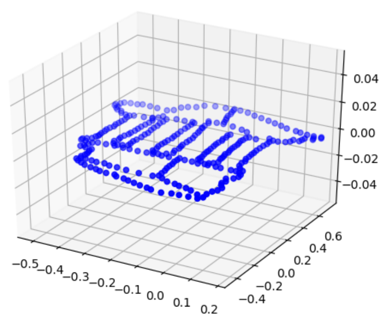

Figure 5: XY plane of reconstruction result

And the 3 different angle result of the reconstruction result is like below.

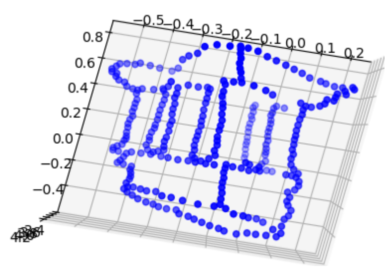

Figure 6: The first angle of reconstruction result

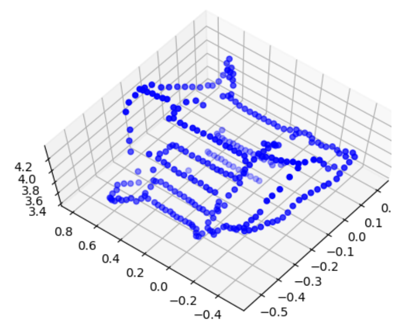

Figure 7: The second angle of reconstruction result

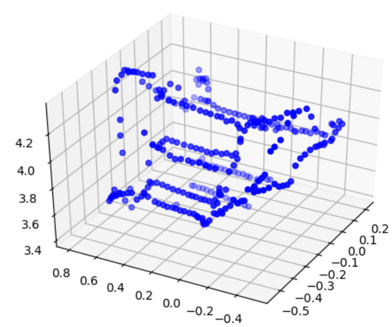

Figure 8: The third angle of reconstruction result

3.2 Dense Reconstruction

3.2.1 Image Rectification

The result of image rectification is like below.

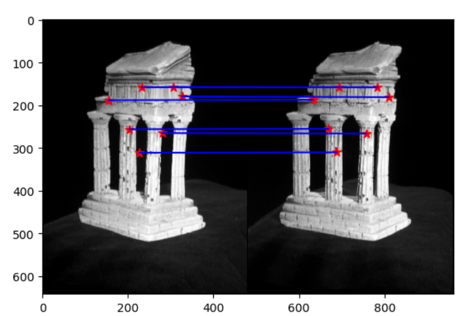

Figure 9: The screenshot of the result of rectification

3.2.2 Dense window matching to find per pixel disparity

3.2.3 Depth map

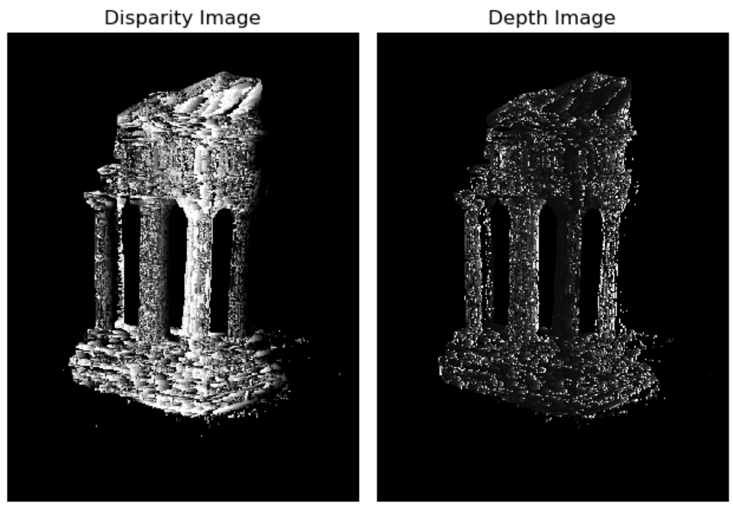

Figure 10: The image of the disparity and the depth maps