Computer Vision - 16720
Homework 4

[redacted]
Andrew id: [redacted]

Start date: Mar 10th
Due date: Mar 30th

Q1.1:

Fundamental matrix is 3*3 and from its property we know that:

$$x_2^T F x_1 = 0$$

For the two coordinates' origin, we could derive

$$\begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = 0$$

$$\begin{bmatrix} f_{31} & f_{32} & f_{33} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = 0$$

Thus we could derive that $f_{33} = 0$

Q1.2:

In calibrated case, assume the translation matrix is $\begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}$. And since the translation is parallel to the x-axis so $t_2 = 0, t_3 = 0$.
From the property of essential matrix (E) we know that: $\boldsymbol{x_2^T E x_1} = 0$ and $\boldsymbol{l_2} = \boldsymbol{E x_1}$
where $\boldsymbol{E} = \bar{T} R$ and $\bar{T}$ is the cross product matrix of translation matrix. So

$$\bar{T} = \begin{bmatrix} 0 & -t_3 & t_2 \\ t_3 & 0 & -t_1 \\ -t_2 & t_1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -t_1 \\ 0 & t_1 & 0 \end{bmatrix}$$

and since it's pure translation so the rotation matrix R is

$$\boldsymbol{R} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\boldsymbol{E} = \bar{T} R = \boldsymbol{E} = \bar{T} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -t_1 \\ 0 & t_1 & 0 \end{bmatrix}$$

$$\boldsymbol{l_2} = \boldsymbol{E x_1} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -t_1 \\ 0 & t_1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ -t_1 \\ y_1 \end{bmatrix}$$

So the epipolar line for C_2 is $-t_1 y_2 + t_1 y_1 = 0$, which is parallel to x axis and vice versa for C_1. It also holds in uncalibrated case.
So epipolar lines in the two cameras are also parallel to the x-axis as well.

Q1.3:

For different frame, assume x denotes points in image and capital X denotes points in real world.

$$\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = K \cdot [R_i | t_i] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \rightarrow\rightarrow \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = K \cdot (R_i \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + t_i)$$

$$\begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = K \cdot (R_1 \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + t_1) \&\& \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = (K^{-1} \begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix} - t_2) \cdot R_2^{-1}$$

$$\begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = K \cdot (R_1 \cdot ((K^{-1} \begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix} - t_2) \cdot R_2^{-1}) + t_1)$$

$$\begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = K R_1 R_2^{-1} K^{-1} \begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix} - K R_1 R_2^{-1} t_2 + K t_1$$

So the R_{rel} and t_{rel} are:

$$R_{rel} = K R_1 R_2^{-1} K^{-1}$$

$$t_{rel} = -K R_1 R_2^{-1} t_2 + K t_1$$

So the essential matrix E and fundamental matrix F are :

$$E = \bar{t}_{rel} R_{rel}$$

$$F = K^{-T} E K^{-1} = K^{-T} \bar{t}_{rel} R_{rel} K^{-1}$$

Q1.4:

Assume the intrinsic matrix of camera is known as $\boldsymbol{K}$, the point in real world is $\boldsymbol{A}$, its reflection point is $\boldsymbol{A'}$, their corresponding point in image are $\boldsymbol{a}$ and $\boldsymbol{a'}$:

$A = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$ and $A' = RefA$, and $a = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} A$ and $\lambda_1 a = KA, \lambda_2 a' = KA'$

Since it's reflection between $\boldsymbol{A}$ and $\boldsymbol{A'}$, so their's only a translation between these two points, assume the translation matrix is T, rotation matrix is R and:

$$R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, T = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}$$

$$A' = RA + T \rightarrow\rightarrow \lambda_2 K^{-1} a' = R\lambda_1 K^{-1} a + T \rightarrow\rightarrow \lambda_2 K^{-1} \bar{T} a' = K^{-1} \bar{T} R \lambda_1 a + T \bar{T}$$

where $\bar{T}$ means take cross product with T and $T\bar{T}$ is equal to 0 since $sin(0) = 0$.
Then take dot product of both side with a':

$$\lambda_2 K^{-T} a'^T \bar{T} a' K^{-1} = K^{-T} a'^T \bar{T} R \lambda_1 a K^{-1}$$

where $\lambda_2 K^{-T} a'^T \bar{T} a' K^{-1} = 0$ because the volume is zero, So:

$$K_{-T} a'^T \bar{T} R \lambda_1 a K^{-1} = 0 \rightarrow\rightarrow a'^T F a = 0$$

where $\bar{T} = \begin{bmatrix} 0 & -t_3 & t_2 \\ t_3 & 0 & -t_1 \\ -t_2 & t_1 & 0 \end{bmatrix}$ is a skew-symmetric matrix since $\bar{T}^T = -\bar{T}$, so:

$F = K^{-T} \bar{T} R K^{-1}$ where K is the intrinsic matrix of camera and R is I_3, so they will not affect the skew-symmetric property of matrix. So:

$$F^T = -F$$

Which means the fundamental matrix F is skew-symmetric.

Q2.1: Eight point Algorithm

The fundamental matrix F is:

$$F = \begin{bmatrix} -0.0000 & 0.0000 & -0.0021 \\ 0.0000 & -0.0000 & -0.0000 \\ 0.0020 & -0.0000 & 0.0083 \end{bmatrix}$$

And the image of some example output of displayEpipolarF is as below.

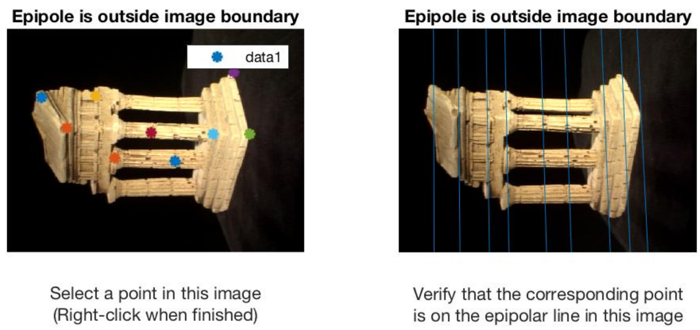

Figure 1: displayEpipolarF.m creates a GUI for visualizing epipolar lines

Q2.2: Seven point Algorithm

I manually selected 7 points in each of the image and stored pts1 and pts2 in $q2\_2.mat$. Three fundamental matrices I got are as below. From the images below we can see F_1 is the best one.

$$F_1 = \begin{bmatrix} 0.0000 & -0.0000 & -0.0009 \\ -0.0000 & 0.0000 & 0.0001 \\ 0.0008 & 0.0001 & -0.0269 \end{bmatrix}$$

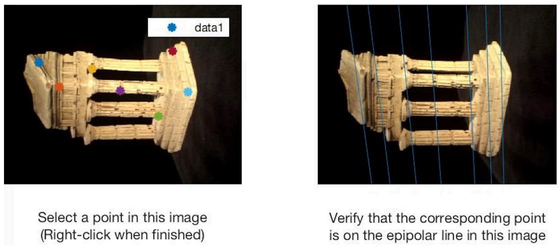

Figure 2: F_1 and output of displayEpipolarF with F_1

$$F_2 = \begin{bmatrix} 0.0000 & 0.0001 & -0.0520 \\ -0.0003 & 0.0005 & -0.3204 \\ 0.0818 & 0.1276 & 13.7267 \end{bmatrix}$$

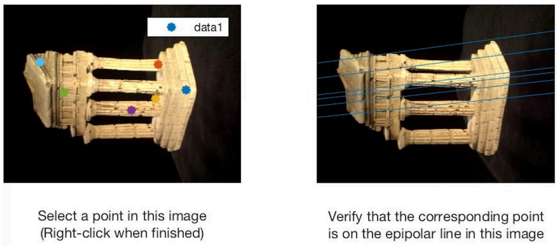

Figure 3: F_2 and output of displayEpipolarF with F_2

$$F_3 = \begin{bmatrix} 0.0000 & 0.0000 & -0.0064 \\ -0.0001 & 0.0000 & 0.0039 \\ 0.0161 & 0.0014 & -1.3638 \end{bmatrix}$$

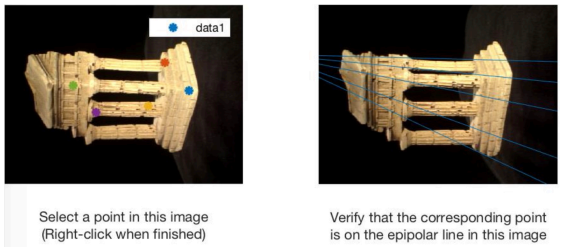

Figure 4: F_3 and output of displayEpipolarF with F_3

Q2x: RANSAC

In my code 7 pair of points were randomly selected to compute the F. With the output of sevenpoint algorithm, F is a 3*1 cell of 3 matrices. For each of the 3 matrices, I compute the epipolar line by points in im1 like the way computing correspondence, assume the epipolar line is $E = \begin{bmatrix} a & b & c \end{bmatrix}^T$, then for points in im2, $ax_2 + by_2 + c = 0$. So the error matrix I used is a matrix of value of this equation for each points. Since this equation equals to zero so the values in error matrix should be small. The tolerance used is 1 and for any specific value in error matrix smaller than this tolerance was considered to be an inlier. Since there are about 75% percent inliers in the given points, I set the number of iteration to be 50 in order to find the most inliers and avoid adding too much computation. The images of RANSAC and eightpoint algorithm are as below and this image of RANSAC is yield with 103 inliers.

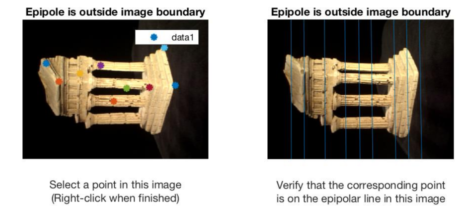

Figure 5: displayEpipolarF.m creates a GUI for visualizing epipolar lines of RANSAC

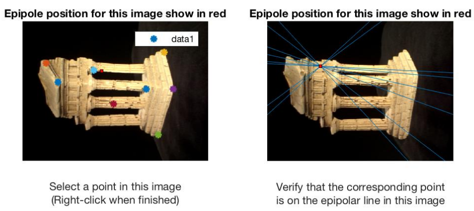

Figure 6: displayEpipolarF.m creates a GUI for visualizing epipolar lines for 8 points algorithm

Q2.3: Computing the essential matrix E

My outcome of essential matrix E is :

$$E = \begin{bmatrix} -0.0071 & 0.3496 & -3.1562 \\ 0.4764 & -0.0058 & 0.0837 \\ 3.1655 & 0.0026 & 0.0012 \end{bmatrix}$$

Q2.4: Triangulation

Please see the code submitted on Blackboard.

Q2.5: Finding correct M_2

With four matrices of M_2 we could got four matrices of 3D points P, each row of P are corresponding to the X, Y and Z coordinates of 3D points. Only the third M_2 lead to a matrix P which all points have positive Z value. So this is the correct M_2 matrix and this M_2 matrix is:

$$M_2 = \begin{bmatrix} 0.9994 & 0.0330 & 0.0043 & -0.0265 \\ -0.0330 & 0.9659 & 0.2567 & -1.0000 \\ 0.0043 & -0.2567 & 0.9665 & 0.1504 \end{bmatrix}$$

Q2.6: Epipolar Correspondence

In this algorithm, I chose the size of window to be 21*21. I've tested through window length of 5 to 30 with a step of 5 and it turned out with window size equal to 21*21 and sigma equal to 5 in gaussian filter rendered the best results. Besides, I used the square root of SSD between patches to evaluate the similarity. The higher the similarity between two patches, the smaller the square root of SSD will be. Below is an image of the outcome of the function epipolarMatchGUI. From the image we can see corresponding points were detected correctly in image2.

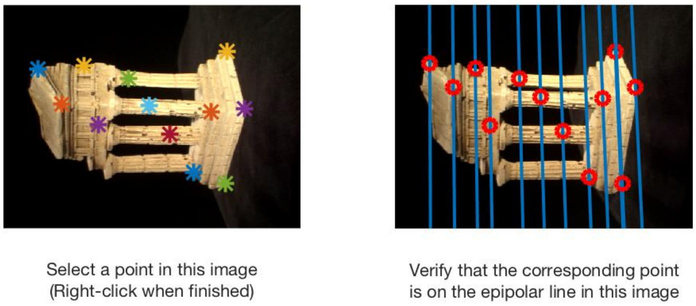

Figure 7: epipolarMatchGUI shows the corresponding point found by calling epipolarCorrespondence

Q2.7: Visualization

The output of visualization are as below.

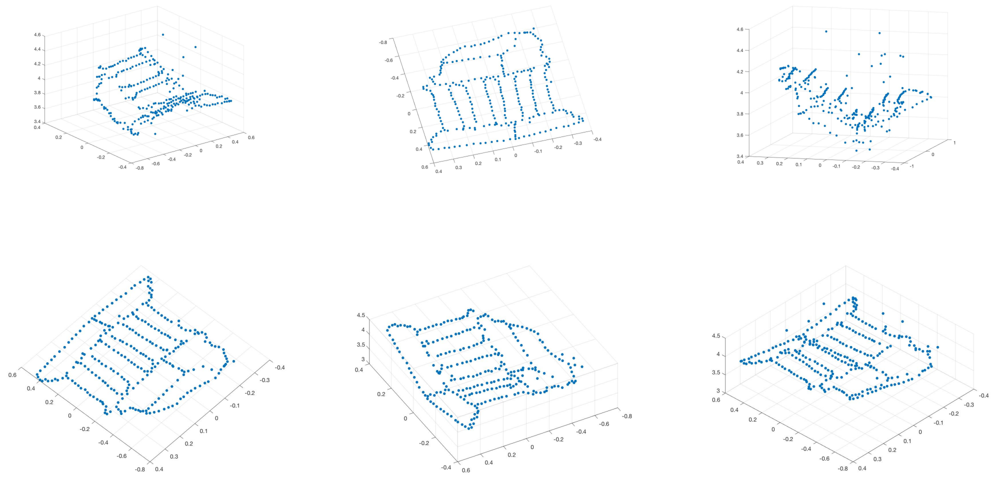

Figure 9: Visualization by point cloud

Q3: Awesome Visualization

I used a software named Pix4D to do the visualization. I also used more images in "Temple" dataset to reconstruct the 3D model. More at: Pix4D and Temple dataset
Here are some screen shot of the video. The video is available at: Video

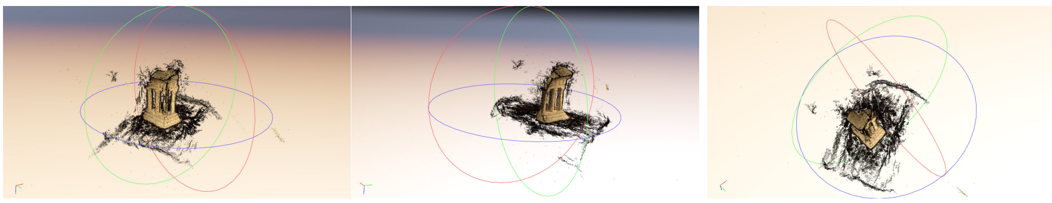

Figure 10: Visualization by Pix4D