Computer Vision - HW4

1. Introduction

Structure from motion (SfM) is a photogrammetric range imaging technique for estimating three-dimensional structures from two-dimensional image sequences that may be coupled with local motion signals. In biological vision, SfM refers to the phenomenon by which humans can recover 3D structure from the projected 2D motion field of a moving object or scene.

In this homework, we have implemented the 3D reconstruction with two 2D images by applying SfM. To accomplish Sfm, we have to have an understanding of "Camera Calibration", "Feature Matching", "Epipolar Geometry", "Fundamental Matrix" and "Essential Matrix". In this assignment, we will focus on the last 3 techniques, since we have implemented the rest on HW1 and HW3.

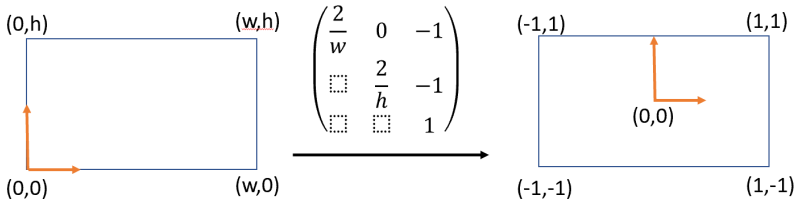
2. Implementation procedure

2.1. Use SIFT to obtain the correspondences (HW3)

We have used SIFT implemented by OpenCV to get the key points and descriptors and use the Sum of Squared Difference(SSD) and ratio threshold to find the good matching pairs as we have done in HW3.

2.2. Estimation of Fundamental Matrix

Before we start to use the algorithm to get fundamental matrix, we need to normalize the images to avoid magnitude difference by transforming the image to [-1,1] x [-1, 1] to do the normalization.



Next, we implemented the 8-point algorithm to calculate the fundamental matrix. Since the matching pairs might contain noise, we estimate the

fundamental matrix by applying the RANSAC algorithm, which selects the most inliners. With the condition $\boldsymbol{x'^{T}Fx = 0}$ we can first compute the initial F, $\boldsymbol{F = U * S * V^{T}}$ via SVD. Since F rank=2 and *det(F)=0*, we set S(2,2)=S[2] = 0 to compute $\boldsymbol{F' = U * S' * V'^{T}}$ to obtain the Fundamental matrix we want as follows.

$$F = U\begin{bmatrix}\sigma 1 & 0 & 0\\ 0 & \sigma 2 & 0\\ 0 & 0 & \sigma 3\end{bmatrix}\mathrm{V}^{\top}$$

$$F' = U\begin{bmatrix}\sigma 1 & 0 & 0\\ 0 & \sigma 2 & 0\\ 0 & 0 & 0\end{bmatrix}\mathrm{V}'^{\top}$$

Finally, we denormalize it $F = T'^{\top}\tilde{F}T$.

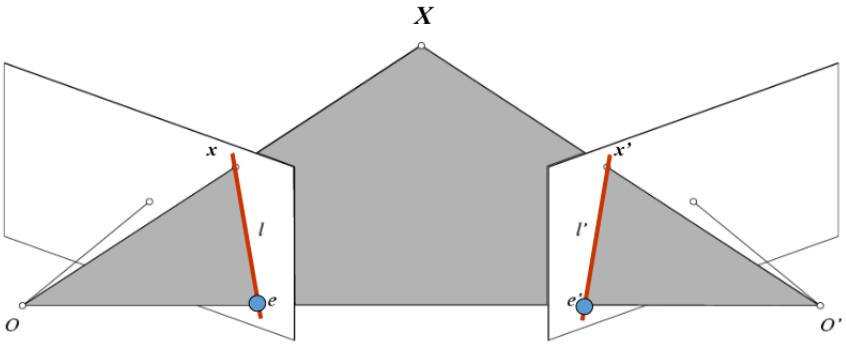
$$x'xf_{11} + x'yf_{12} + x'f_{13} + y'xf_{21} + y'yf_{22} + y'f_{23} + xf_{31} + yf_{32} + f_{33} = 0$$

$$\mathbf{Af} = \begin{bmatrix} x'_1x_1 & x'_1y_1 & x'_1 & y'_1x_1 & y'_1y_1 & y'_1 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x'_nx_n & x'_ny_n & x'_n & y'_nx_n & y'_ny_n & y'_n & x_n & y_n & 1 \end{bmatrix} \begin{bmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \\ f_{33} \end{bmatrix} = \mathbf{0}$$

2.3. Draw the epipolar lines

After we get the Fundamental matrix and the inlier matching pairs, we can get the epipolar lines, $\boldsymbol{l = F^{T}x \text{ and } l' = Fx'}$, which help us to understand the relative location between camera and object. Therefore, we simply multiply Fundamental matrix F and each

point in the other image to get the epipolar lines.



2.4. Estimation of Essential Matrix

In this step, we use fundamental matrix F and intrinsic matrix K to obtain our essential matrix. By the formula $\boldsymbol{E = K_1^T F K_2}$

As we have learned that the essential matrix can be parameterized by parameter R and t: $\boldsymbol{E=[t]_x R}$. Therefore, we use SVD to decompose E, $\boldsymbol{E = U * S * V^T}$, and get the S' and set m as (S(1,1)+S(2,2)/2) and find $\boldsymbol{E' = U * S' * V'^T}$.

$$E = U \begin{bmatrix} \sigma 1 & 0 & 0 \\ 0 & \sigma 2 & 0 \\ 0 & 0 & \sigma 3 \end{bmatrix} V^T$$

$$E' = U \begin{bmatrix} \frac{\sigma 1+\sigma 2}{2} & 0 & 0 \\ 0 & \frac{\sigma 1+\sigma 2}{2} & 0 \\ 0 & 0 & 0 \end{bmatrix} V'^T$$

Given first camera matrix $P_1 = [I|0]$

There are four possible solution for the second camera matrix P_2: [R1|t1] [R1|t2] [R2|t1] [R2|t2], which $\boldsymbol{R1 = UW^TV^T}$, $\boldsymbol{R2 = UWV^T}$, $\boldsymbol{t1 = u3}$, $\boldsymbol{t2 = -u3}$

$$\mathbf{P}_2 = [\mathbf{U}\mathbf{W}\mathbf{V}^T | +\mathbf{u}_3]$$
$$\mathbf{P}_2 = [\mathbf{U}\mathbf{W}\mathbf{V}^T | -\mathbf{u}_3]$$
$$\mathbf{P}_2 = [\mathbf{U}\mathbf{W}^T\mathbf{V}^T | +\mathbf{u}_3] \quad \mathbf{W} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$
$$\mathbf{P}_2 = [\mathbf{U}\mathbf{W}^T\mathbf{V}^T | -\mathbf{u}_3]$$

Then we can calculate the projection matrices K[R|t] of both cameras to project 2D points to 3D coordinate.

2.5. Applying triangulation to get 3D points

After obtaining the 2D coordinates and projection matrices of both cameras, we can reconstruct the 3D object by using the method Triangulation.

Let X be the 3D world coordinate, x and x' be the 2D coordinates of 2 images, P and P' are the projection matrices of two cameras. We know that

$$x=PX,\ \ x'=P'X$$

By substituting and transposition of term we get

$$x = w\begin{bmatrix}u\\v\\1\end{bmatrix} = PX = \begin{bmatrix}p_1{}^T X\\p_2{}^T X\\p_3{}^T X\end{bmatrix} = \begin{bmatrix}up_3{}^T X\\vp_3{}^T X\\p_3{}^T X\end{bmatrix}$$

$$x' = w\begin{bmatrix}u'\\v'\\1\end{bmatrix} = P'X = \begin{bmatrix}p'_1{}^T X\\p'_2{}^T X\\p'_3{}^T X\end{bmatrix} = \begin{bmatrix}up'_3{}^T X\\vp'_3{}^T X\\p'_3{}^T X\end{bmatrix}$$

$$\begin{bmatrix}up_3{}^T X\\vp_3{}^T X\\p_3{}^T X\end{bmatrix} - \begin{bmatrix}p_1{}^T X\\p_2{}^T X\\p_3{}^T X\end{bmatrix} = \begin{bmatrix}up_3{}^T - p_1{}^T\\vp_3{}^T - p_2{}^T\end{bmatrix} X = 0$$

$$\begin{bmatrix}u'p'_3{}^T X\\v'p'_3{}^T X\\p'_3{}^T X\end{bmatrix} - \begin{bmatrix}p'_1{}^T X\\p'_2{}^T X\\p'_3{}^T X\end{bmatrix} = \begin{bmatrix}u'p'_3{}^T - p'_1{}^T\\v'p'_3{}^T - p'_2{}^T\end{bmatrix} X = 0$$

so we can formulate our equation to an SVD problem

$$\mathbf{A}X = 0 \quad A = \begin{bmatrix} u\mathbf{p}_3^\top - \mathbf{p}_1^\top \\ v\mathbf{p}_3^\top - \mathbf{p}_2^\top \\ u'\mathbf{p}'^\top_3 - \mathbf{p}'^\top_1 \\ v'\mathbf{p}'^\top_3 - \mathbf{p}'^\top_2 \end{bmatrix}$$

2.6. Pick the correct solution of camera projection matrix

Since there are four possible projection matrices, we have to pick the solution that with most 3D points will be in front of both cameras. Therefore, we have to compute the forward vector to determine whether the points are in front cameras.

As we have got our rotation matrix R and transformation matrix t of both camera (P_1= [I|0], P_2: [R1|t1] [R1|t2] [R2|t1] [R2|t2], which ***R1= UW^TV^T, R2 = UWV^T, t1 = u3, t2 = -u3***).

Since the Camera Extrinsic [R|t] can transform camera coordinate to world coordinate as below

$$\begin{bmatrix} X_{cam} \\ Y_{cam} \\ Z_{cam} \end{bmatrix} = R \begin{bmatrix} X_{\text{world}} \\ Y_{\text{world}} \\ Z_{\text{world}} \end{bmatrix} + t \Leftrightarrow \begin{bmatrix} X_{\text{world}} \\ Y_{\text{world}} \\ Z_{\text{world}} \end{bmatrix} = R^{-1} \left(\begin{bmatrix} X_{\text{cam}} \\ Y_{\text{cam}} \\ Z_{\text{cam}} \end{bmatrix} - t \right) = R^\top \begin{bmatrix} X_{\text{cam}} \\ Y_{\text{cam}} \\ Z_{\text{cam}} \end{bmatrix} - R^\top t$$

We can compute the camera position of the world coordinate by R and t especially for four possible solutions of the second camera.

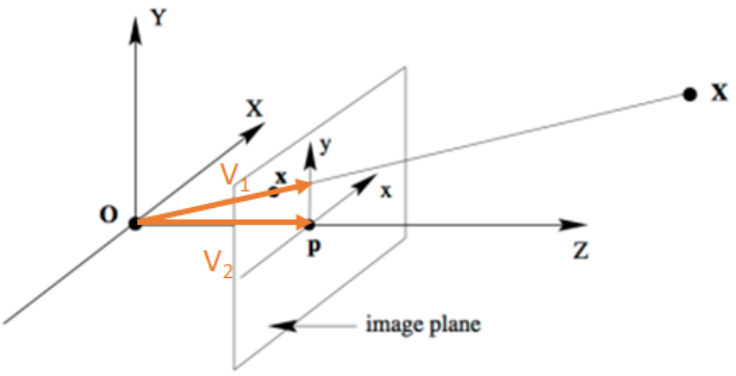
$$\begin{bmatrix} X_{\text{cam}} \\ Y_{\text{cam}} \\ Z_{\text{cam}} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \Leftrightarrow \mathbf{C} = \begin{bmatrix} X_{\text{world}} \\ Y_{\text{world}} \\ Z_{\text{world}} \end{bmatrix} = R^\top \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} - R^\top t = -R^\top t$$

Next, we can compute the view direction, which is the forward vector, by using the z axis identity matrix to compute the z axis

direction from camera coordinate to world coordinate as following

$$\begin{bmatrix}0\\0\\1\end{bmatrix}-\begin{bmatrix}0\\0\\0\end{bmatrix}\Leftrightarrow\left(R^{\top}\begin{bmatrix}0\\0\\1\end{bmatrix}-R^{\top}t\right)-(\mathbf{C})=\left(R(3,:)^{\top}-R^{\top}t\right)-\left(-R^{\top}t\right)=R(3,:)^{\top}$$

Furthermore, we assume two vectors V_1 and V_2. V_1 is the vector from world center to the 3D point (X-C)and V2 is the forward vector $R(3,:)^T$.



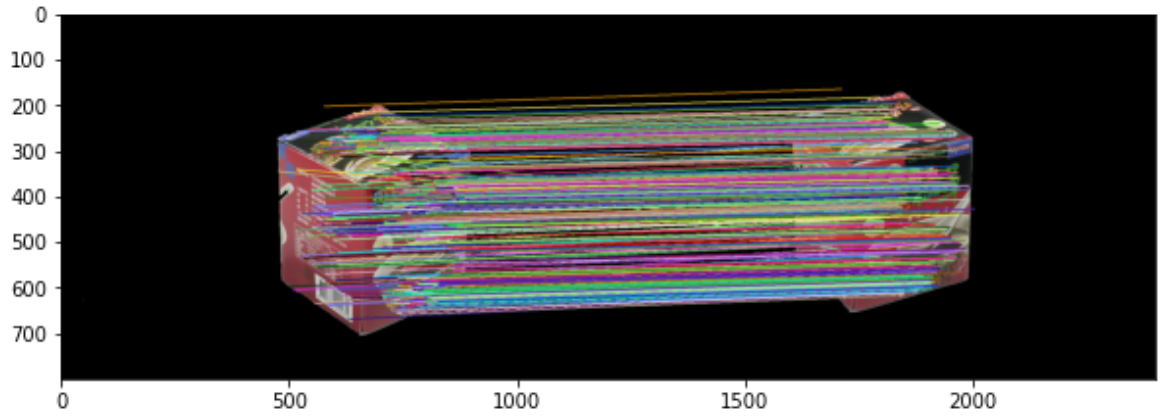
We only have to test whether $(X-\mathbf{C})\cdot R(3,:)^{\top}>0$ to know whether the 3D point is in the front of the camera.
Eventually, we find the solution with the most 3D points and reconstruct the 3D Model with these 3D points with MATLAB code which TA provided to us.

3. Experiment Result

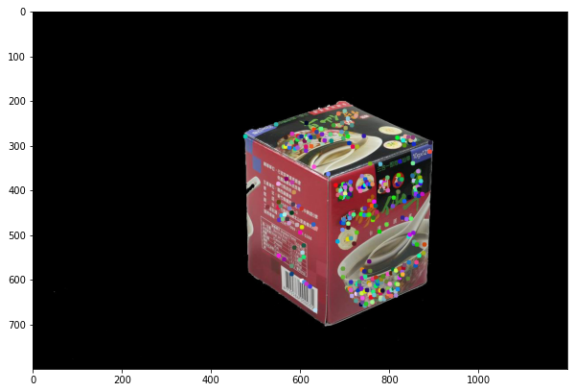
Image: Mesona
ratio_distance_threshold: 0.7
RANSAC: 0.01

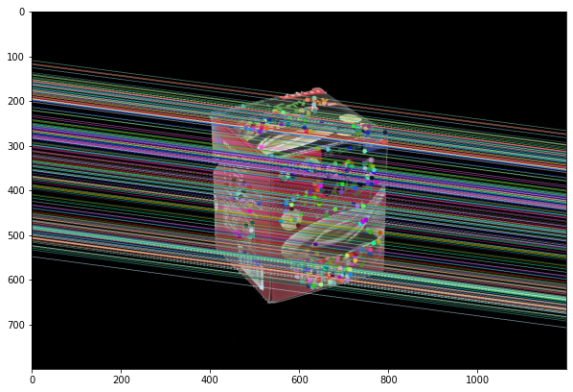
Mesona1.JPG	Mesona2.JPG
	

K1
[[1421.9, 0.5, 509.2] [0, 1421.9, 380.2] [0, 0, 1]]
K2
K2=K1
interest points and correspondence across images



interests point	epipolar lines





fundamental matrix
[[1.02439304e-07 -7.53093343e-08 -2.30490438e-03] [-7.48150540e-07 9.42614365e-08 1.76523681e-02] [1.22593653e-03 -1.68105234e-02 1.00000000e+00]]
essential matrix
[[0.20506415 -0.17779656 -3.18480639] [-1.49546802 0.36560146 24.15324789] [1.45142069 -24.36391018 0.47567255]]

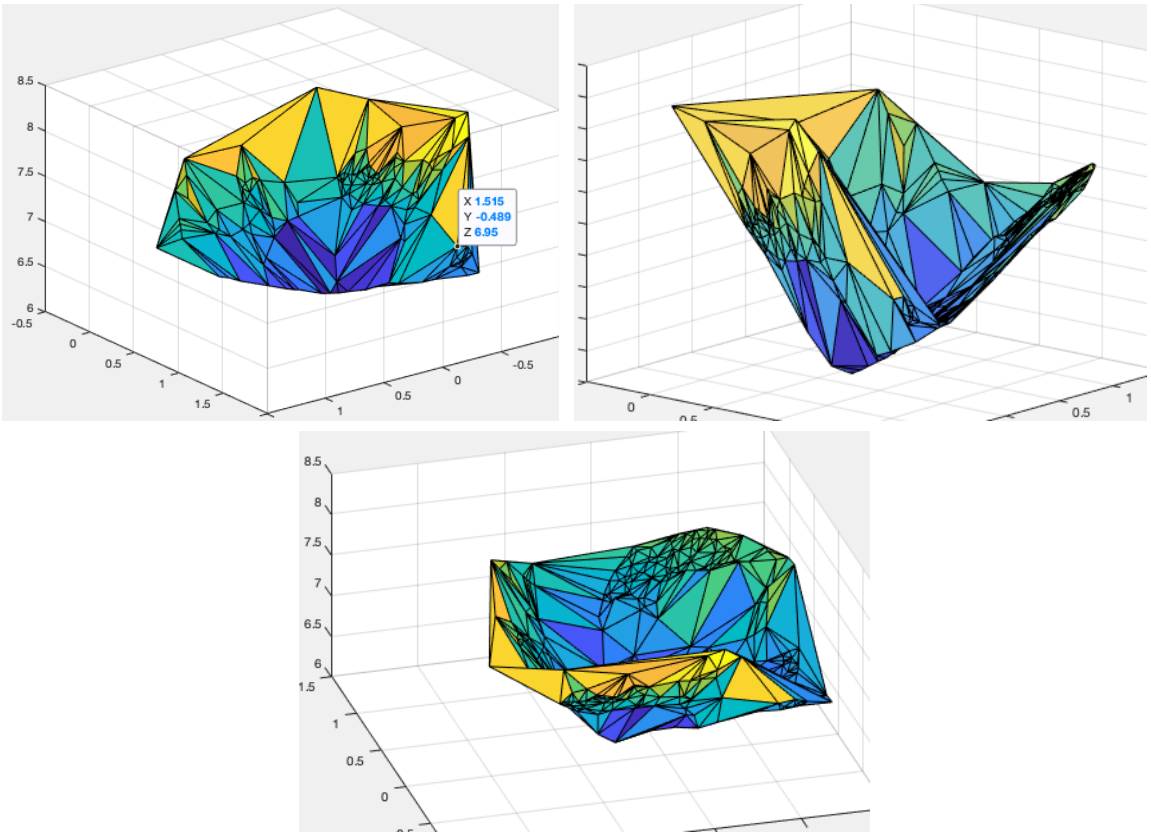
camera matrix
[[1.4219e+03 5.0000e-01 5.0920e+02 0.0000e+00] [0.0000e+00 1.4219e+03 3.8020e+02 0.0000e+00] [0.0000e+00 0.0000e+00 1.0000e+00 0.0000e+00]]
3D model


Image: Statue
ratio_distance_threshold: 0.5
RANSAC: 0.1

Statue1.bmp	Statue2.bmp

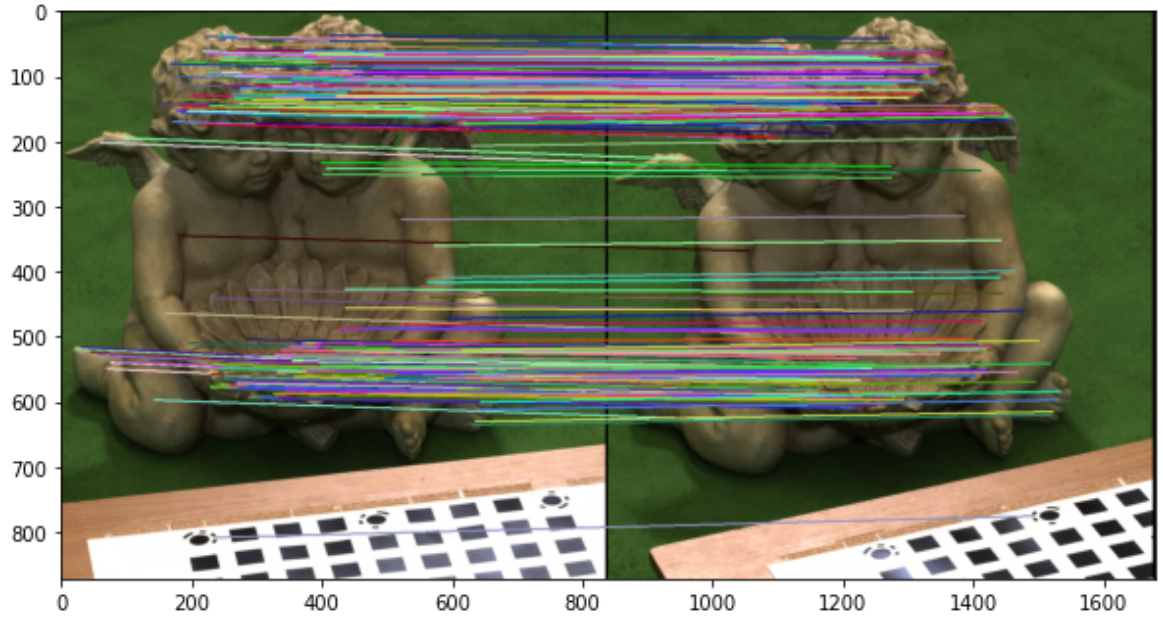
K1

[[5.42656689e+03 6.78017000e-01 3.30096680e+02]
 [0.00000000e+00 5.42313330e+03 6.48950012e+02]
 [0.00000000e+00 0.00000000e+00 1.00000000e+00]]

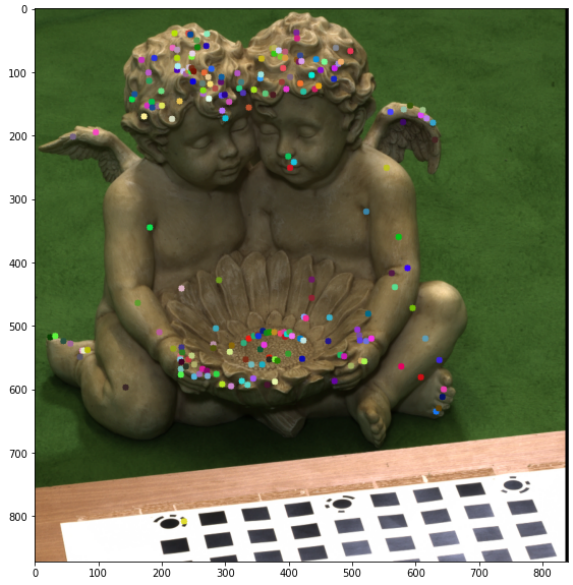
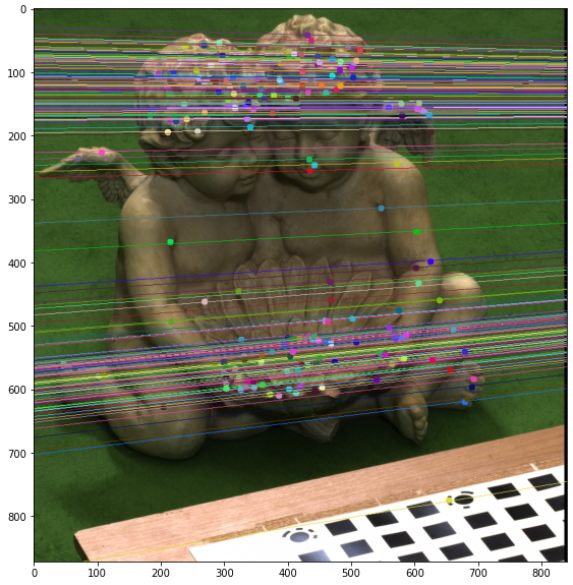
K2

[[5.42656689e+03 6.78017000e-01 3.87430023e+02]
 [0.00000000e+00 5.42313330e+03 6.20616699e+02]
 [0.00000000e+00 0.00000000e+00 1.00000000e+00]]

interest points and correspondence across images



Group 5

interests point	epipolar lines
	

fundamental matrix

[[5.35317406e-07 -6.61540173e-06 7.35171672e-04]
 [5.38536933e-06 4.76915401e-07 -2.89075949e-02]
 [-3.19435925e-03 2.95294543e-02 1.00000000e+00]]

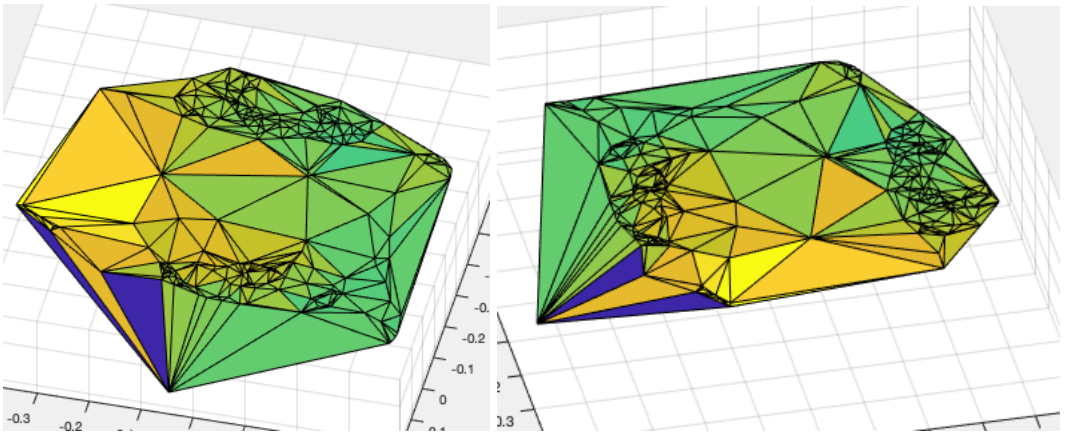
essential matrix

[[14.92569224 -183.06329272 -17.34758667]
 [169.10323638   16.3898401 -155.16684913]
 [3.48359456 139.30300989   -0.44598196]]

camera matrix

[[5.42656689e+03 6.78017000e-01 3.30096680e+02 0.00000000e+00]
 [0.00000000e+00 5.42313330e+03 6.48950012e+02 0.00000000e+00]
 [0.00000000e+00 0.00000000e+00 1.00000000e+00 0.00000000e+00]]

3D model



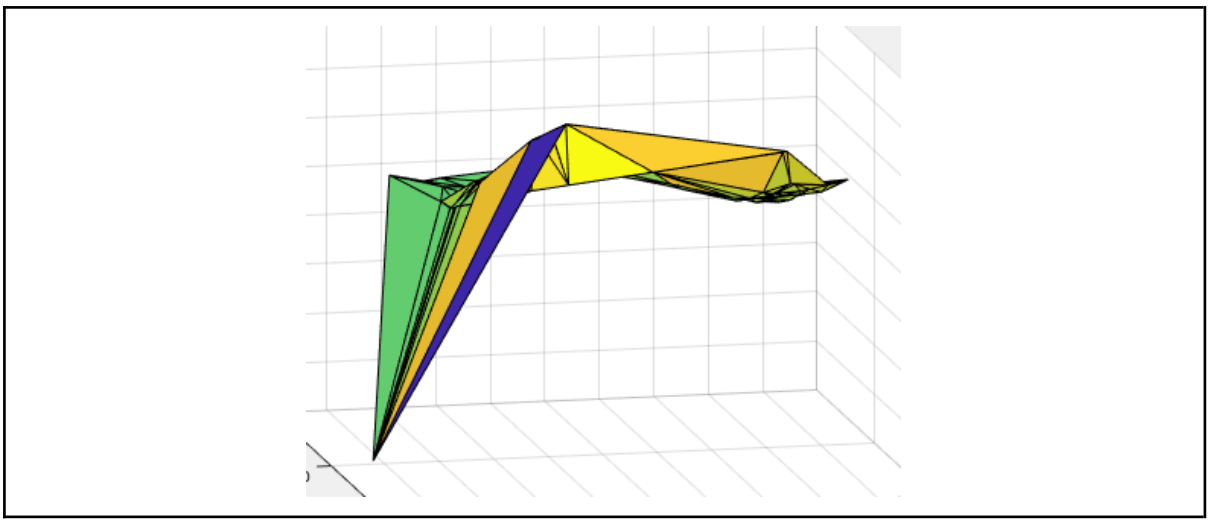
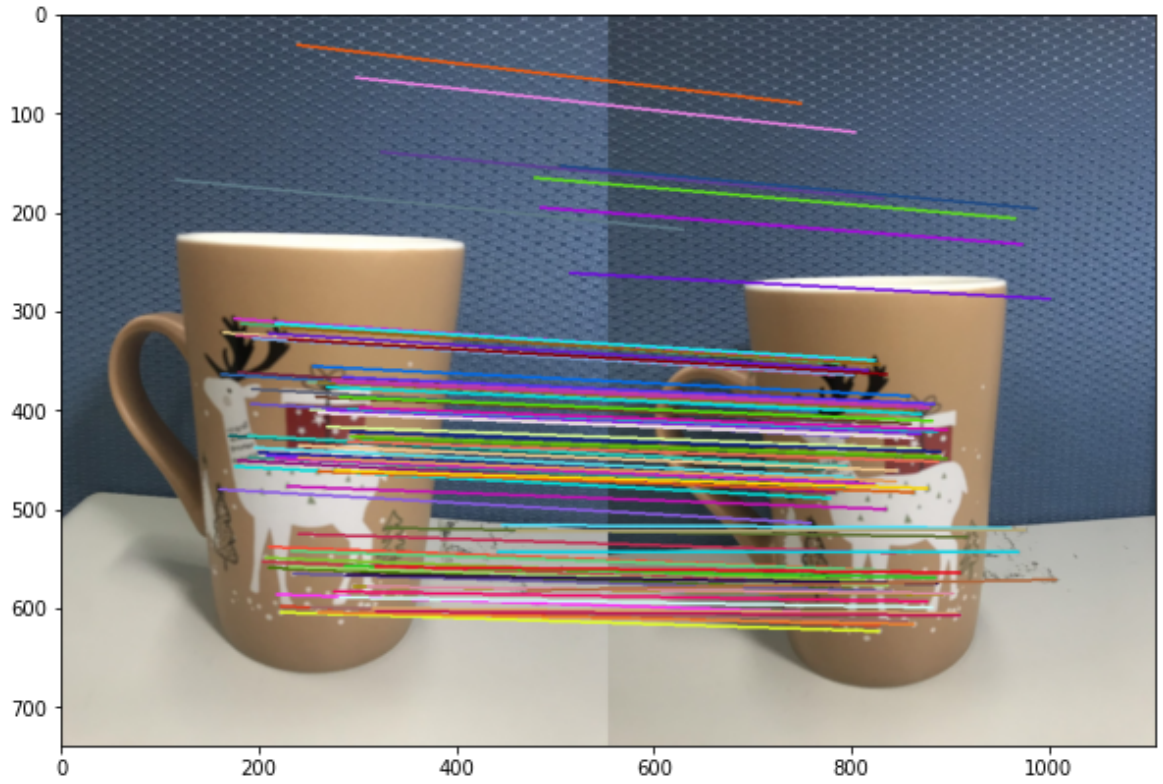


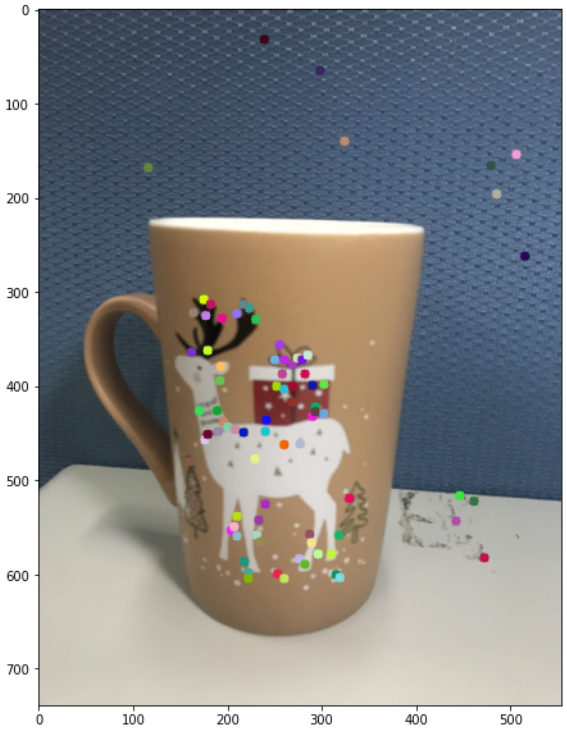
Image: Deer (our own data)
ratio_distance threshold: 0.5
ransac_threshold: 0.01

deer1.jpg	deer2.jpg
	
K1	
[[1.23451531e+03, -2.30410376e+00, 5.64016232e+02], [0.00000000e+00, 1.23420888e+03, 7.29026308e+02], [0.00000000e+00, 0.00000000e+00, 1.00000000e+00]]	
K2	
K2 = K1	

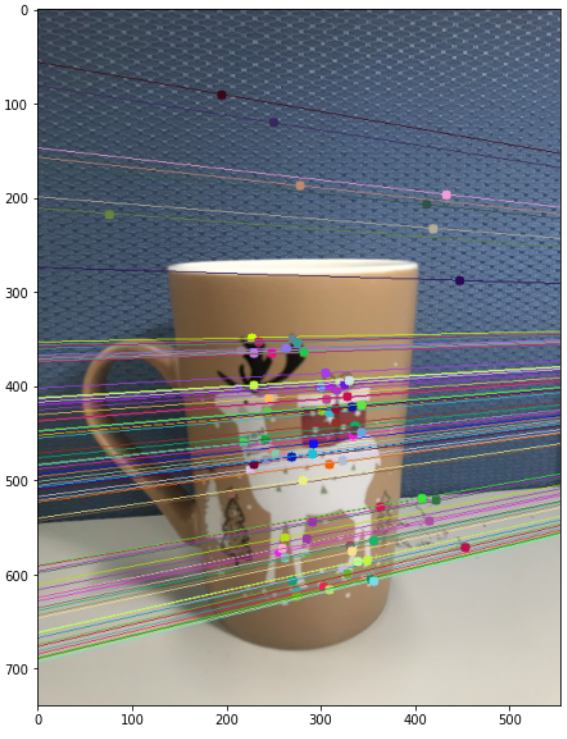
interest points and correspondence across images



interests point



epipolar lines



fundamental matrix

[[6.93565931e-07 -1.02436918e-05 2.73526959e-03]
 [5.14605131e-06 1.94504724e-06 -1.60354054e-02]
 [-2.71664604e-03 1.49905763e-02 1.00000000e+00]]

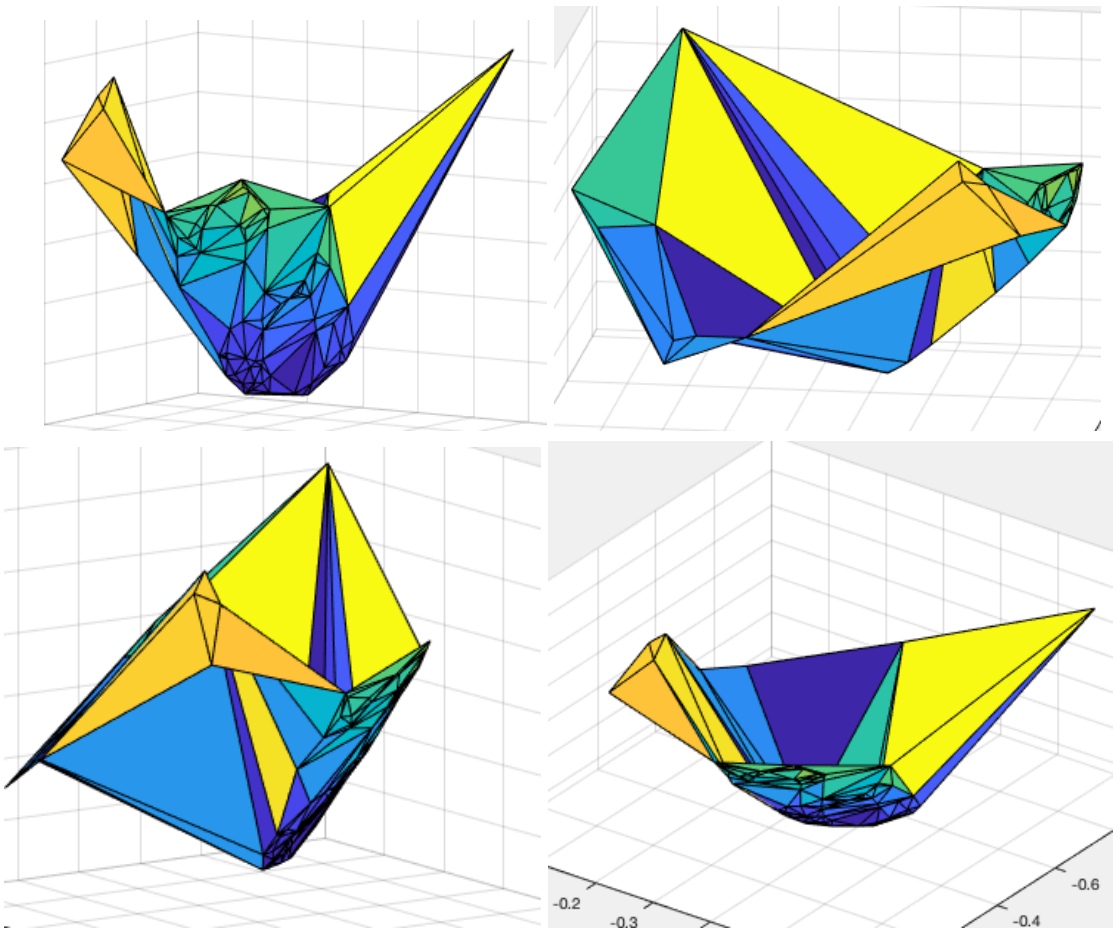
essential matrix

[[1.26115162 -14.22106807 -5.45652958]
 [8.62727912   3.08654816 -15.94257265]
 [1.86163303 12.07379259 -1.00966697]]

camera matrix

[[1.23451531e+03 -2.30410376e+00 5.64016232e+02 0.00000000e+00]
 [0.00000000e+00 1.23420888e+03 7.29026308e+02 0.00000000e+00]
 [0.00000000e+00 0.00000000e+00 1.00000000e+00 0.00000000e+00]]

3D models



4. Discussion

4.1. Two thresholds have to be set in the whole procedure. One is for ratio distance when searching the image correspondence with the

interest points found with the SIFT algorithm, the other is the threshold when choosing a good fundamental matrix with the RANSAC algorithm. We find out that both thresholds influence the final results. If we set a bad set of thresholds, the 3D model would be very strange. Therefore, we adjust the thresholds for each image set to get a 3D model which makes more sense.

4.2. It's hard to say if we are correct or not. We discuss with other groups, and realize that our results never are exactly the same. The parameters really matter, and we don't know what a good 3D model is.

4.3. We cannot recognize the 3D model clearly when comparing the 2 images it comes from. We think the reason is that the image information is still not enough to reconstruct the complete 3D model. If we use more images for one 3D model, it could be better reconstructed.

4.4. Some areas with no obvious texture make the reconstruction more difficult because the area is located few interest points. If the background has too many interest points and the main object has little, it's hard to get a perfect 3D model, such as the case performed on our own data. When we set the ratio distance higher, the points on the background get less, but it becomes hard to compute a good fundamental matrix.

5. Conclusion

In this homework, we implemented the task Structure from Motion. First, we implemented the 8-points algorithm and RANSAC to estimate the fundamental matrix. Second, we compute the essential matrix with the fundamental matrix. Third, by applying SVD on the essential matrix, we obtain four possible extrinsic matrix solutions of the second camera. Finally, we found the solution with the most 3D points in front of the camera to determine our result and draw the 3D model with MATLAB.

Work Assignment Plan with team members

We discussed and finished this assignment together.