EE243: Advanced Computer Vision
Assignment #6

Ali Jahanshahi (862029266)

June 2019

1 Homography Estimation

In this question, we are asked to estimate the Homography matrix (H) between pairs of provided images. We used the `detectCheckerboardPoints` function in Matlab to extract the corner points. In order implementation, we also made sure that there is a one-to-one correspondence between the corner points in the two images (line 28 in the code).
Equation 1 shows the homography matrix for $Image1$ and $Image9$ pair.

$$\begin{vmatrix} -0.0012 & -0.0016 & -0.2764 \\ -0.0003 & -0.0057 & 0.9610 \\ 0.0000 & -0.0000 & -0.0017 \end{vmatrix} \tag{1}$$

In order to see if the result is correct we calculated Equation 2. It shows that mean square error between $P9$ and $HP1$ which is equal to 0.2970, which is much less than 1.

$$MSE(P9, HP1) = 0.2970 \tag{2}$$

2 Structure from Motion (SFM)

In this problem we are asked to implement the factorization based Structure from Motion (SFM) method. We used the starter code `StructureFromMotionExample.m`, which is a modified version of SFM example in matlab. We wrote function `get3DPoints` that takes in a matrix of dimension $2F \times P$, F is the number of

frames ($F = 2$ here) and P is the number of matched points. The output of this function is a matrix of $3D$ points of dimension $P \times 3$.
Figure 1 shows original images, and Figure 2 shows the undistorted images.

Figure 1: Original images.

Figure 2: Undistorted images.

In order to apply the `get3DPoints` on the images, we concatenated the tracked points and inputted them to `get3DPoints` function. As we can see, the value passed to the function is not the original W matrix from [1], and we do further processings in `get3DPoints` function. The following steps are performed in $get3DPoints$ function according to [1]:

- Mean subtraction

- Stacking
- Calculating singular values, and Right singular vectors
- Calculating 3D points

Figure 3 shows the 150 strongest corners of Figure 1.

Figure 3: 150 strongest corners.

Figure 4 shows the tracked features.

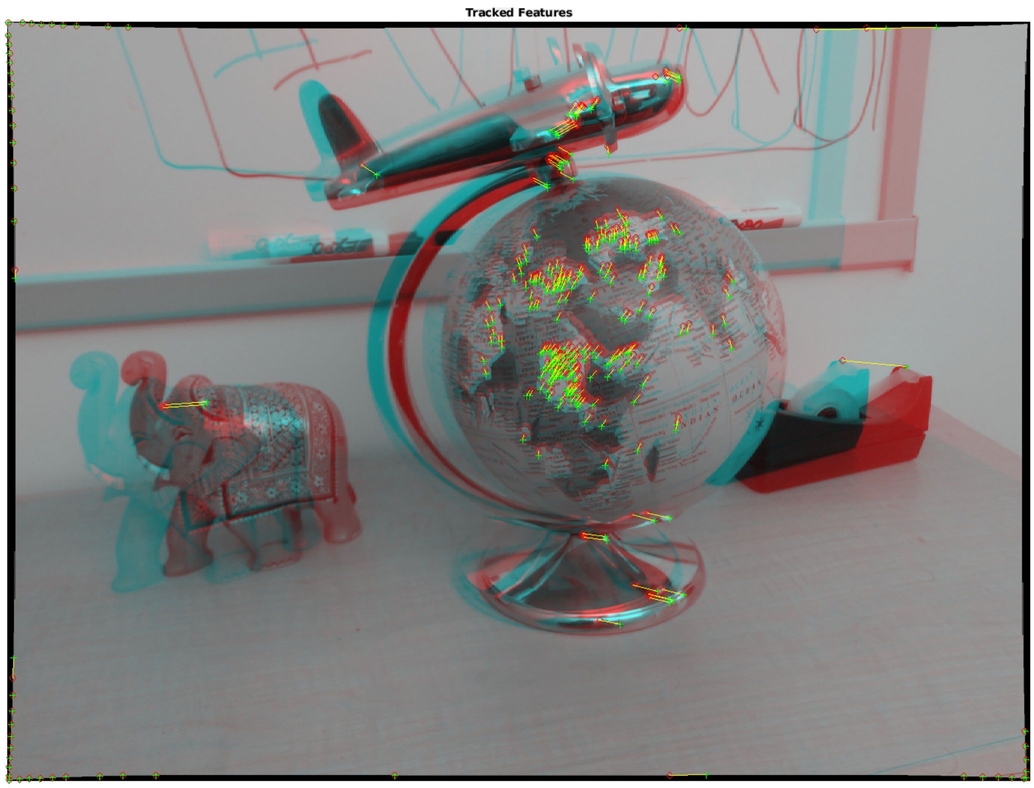

Figure 4: Tracked features.

Figure 6 shows the epipolar inliers.

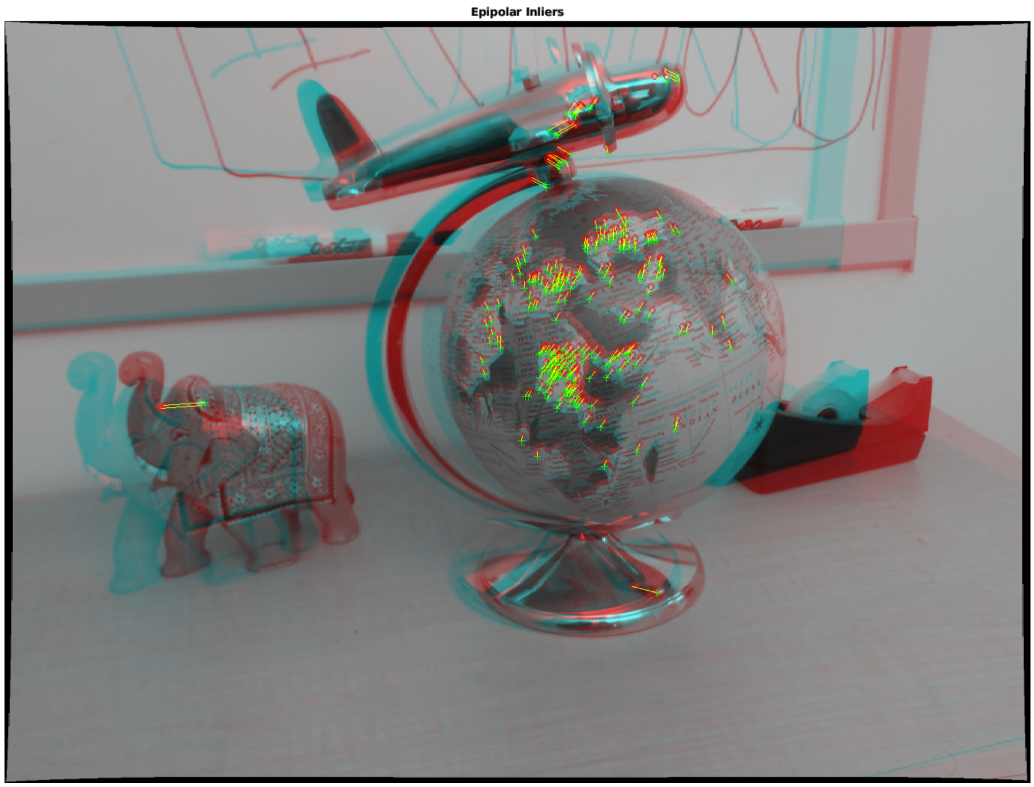

Figure 5: Epipolar inliers.

Figure 6 shows the up to scale reconstruction of the scene.

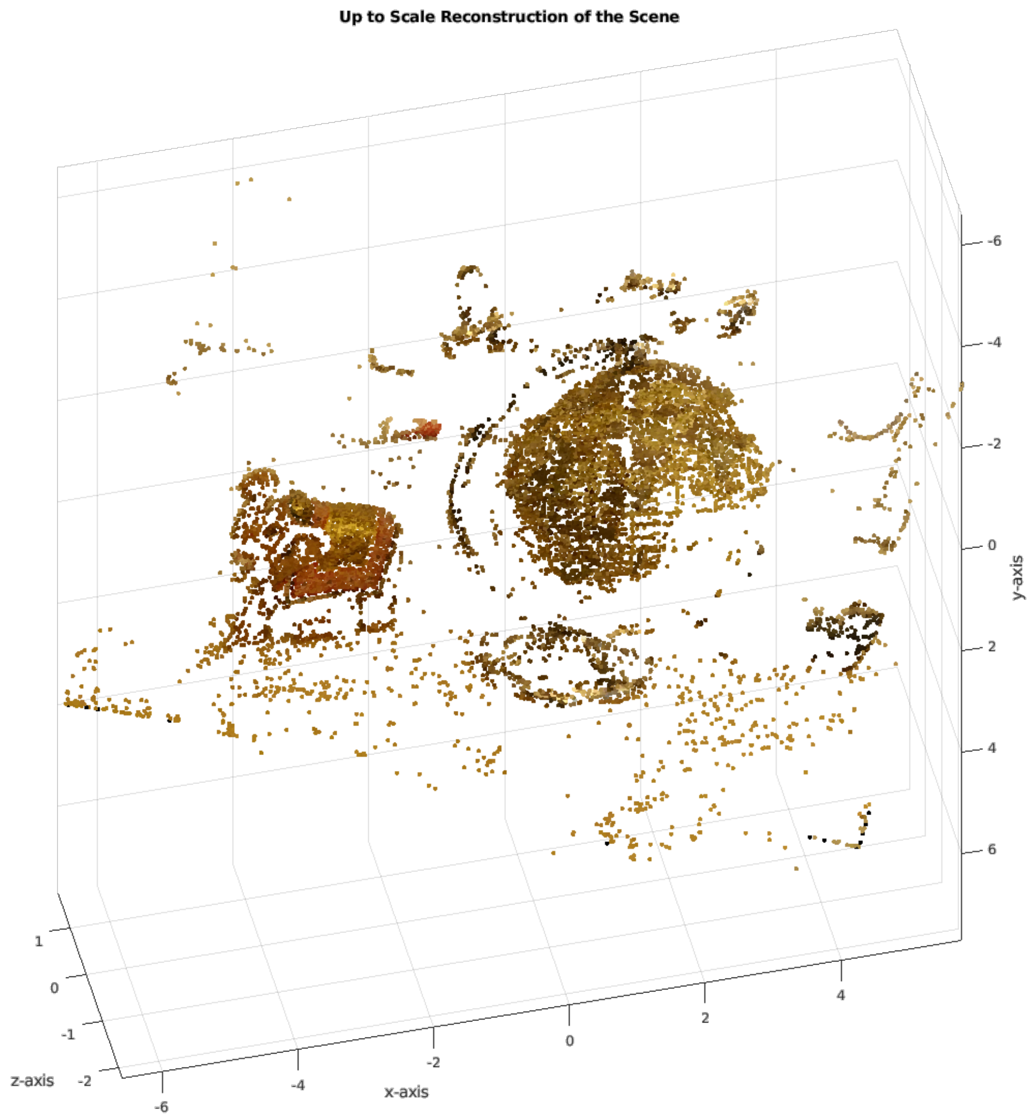

Figure 6: Up to scale reconstruction of the scene.

3 Fundamental Matrix Estimation

In this problem, we used the stereo image pairs $viprectification\_deskLeft.png$ and $viprectification\_deskRight.png$, available in Matlab. We load the images and detect the key-points along with their descriptors using $SIFT$. Figure 7 shows the matching points, which are 41 points. Using the matched points, we employed the 8-point algorithm to obtain the fundamental matrix. According to Equation 3, the number of choosing 8-point subsets is very large. We chose 100000 random subsets of 8-point to estimate a set of fundamental matrices. Then, for each fundamental matrix, we computed the mean error on the rest of matched points using the Sampson distance. Equation 4 shows the fundamental matrix that produced the least error and also report the mean error, which is 1.6510845e-04.

$$\# \; of \; 8 - point \; subsets = {}_{41}\mathrm{C}_8 = \binom{41}{8} = 95548245 \tag{3}$$

$$F_{min_{error}} = \begin{vmatrix} -0.0000005 & -0.0000037 & 0.0077564 \\ 0.0000037 & 0.0000010 & -0.0107442 \\ -0.0075330 & 0.0104630 & -0.002508 \end{vmatrix} \tag{4}$$

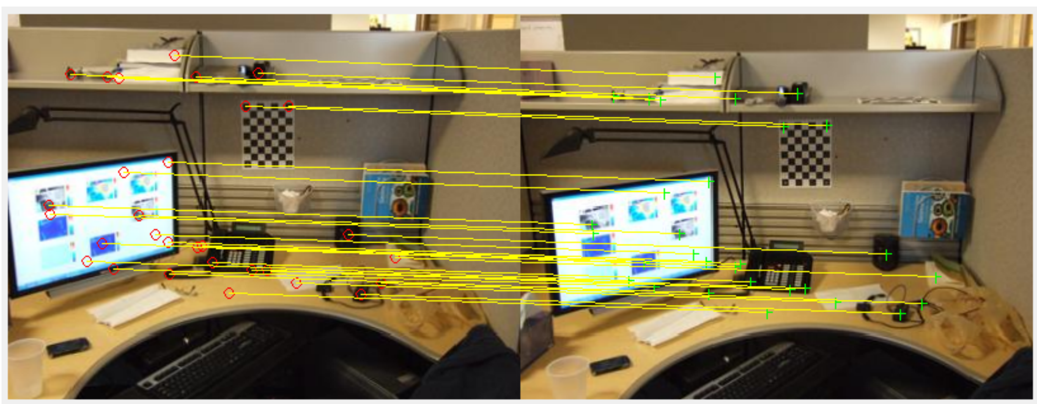

Figure 7: Matching points.

References

[1] Takeo Kanade and Daniel D Morris. Factorization methods for structure from motion. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 356(1740):1153–1173, 1998.