ETH ZURICH, D-INFK

Computer Vision Exercise 4: Model Fitting and MultiView

Xingchang Huang

November 9, 2017

1 LINE FITTING WITH RANSAC

For line fitting model, we should determine values of slope k and bias b from line $y = kx + b$. When there are many outliers that may have bad influence on line fitting, the key idea is to use RANSAC to robustly determine the line parameters and minimize the number of outliers at the same time. Therefore, the steps of this method is as follows. First find two points and determine a line and compute all distance from point to line using equation:

$$d_i = \frac{|kx_i - y_i + b|}{\sqrt{kx_i^2 + 1}} \tag{1.1}$$

where i is the index of points. If the distance from points to line is less than the threshold, they would be considerded as inliers. If the number of inliers in a iteration is bigger than before, the parameters k and b would be updated. If the ratio of inliers is small, then discard that line and continue the next iteration. The result of my line fitting model is shown in Figure 1.1.

2 FUNDAMENTAL MATRIX

Fundamental matrix is a constraint on two-view geometry, where $x'^T Fx = 0$. Here x and x' represent two points in two views respectively. Briefly, in order to get matrix F, we need to find at least 8 points for eight-point algorithm. For details, the points should be transformed in homogeneous coordinate and then normalized, ensuring that their root-mean-squared

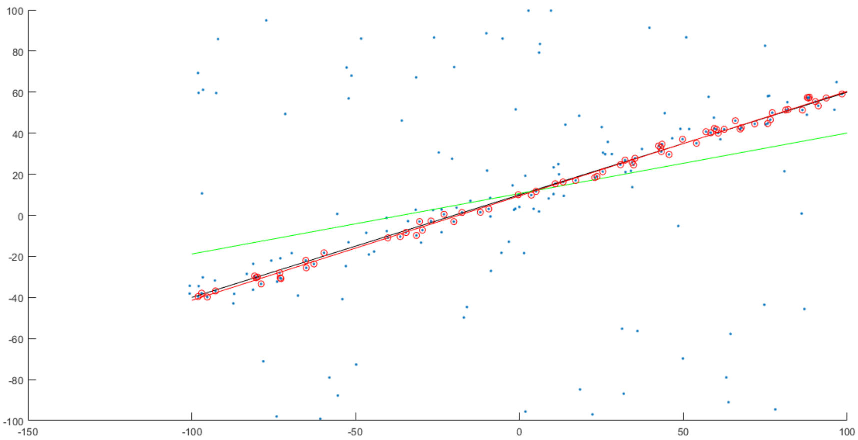

Figure 1.1: Line Fitting with RANSAC.

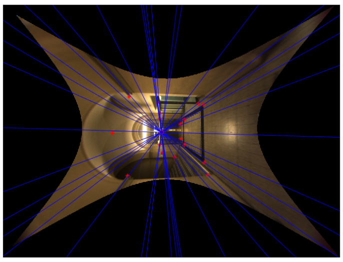

Figure 2.1: Epipolar Lines for estimating Fundamental Matrix: First View, with Epipole.

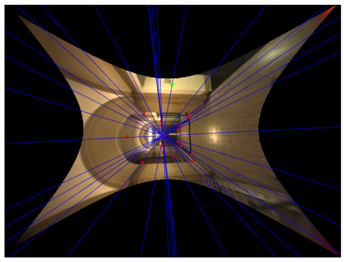

Figure 2.2: Epipolar Lines for estimating Fundamental Matrix: Second View, with Epipole.

distance is $\sqrt{2}$. Next is constructing a matrix using 8 points in order to solve the fundamental matrix. SVD should be used for newly constructed matrix and get the unconstraint fundamental matrix. After that, we should use SVD again and then zeros the third singular value. The re-normalized F should be $T_2^{'} F T_1$, where T_1 and T_2 are normalized matrix for x and xąő respectively. The results on two image pairs have been shown in Figure 2.1 - Figure 2.4. The comparison of F(unconstraint) and F is shown in Figure 3.5 and their values are quite similar. Also, their epipoles can be correctly shown in the right place.

3 Essential Matrix

When it comes to the Essential Matrix, the calibrated matrix would be give in advance, but the method used would be similar to the method used for getting fundamental matrix. The first step should be normalized the coordinate by intrinsic matrix K^{-1} for points from two

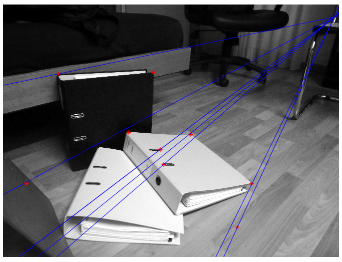

Figure 2.3: Epipolar Lines for estimating Fundamental Matrix: First View.

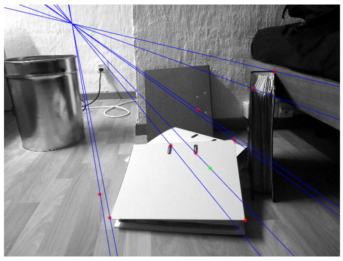

Figure 2.4: Epipolar Lines for estimating Fundamental Matrix: Second View.

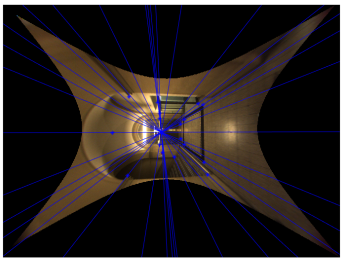

Figure 3.1: Epipolar Lines for estimating Essential Matrix: First View.

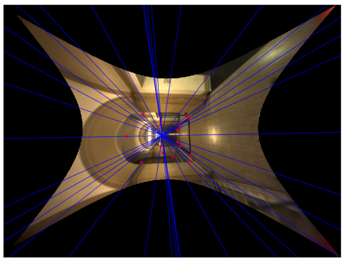

Figure 3.2: Epipolar Lines for estimating Essential Matrix: Second View.

views. Then find 8 normalized points for eight-point algorithm. The difference here is that the constraint of the first two singular values must be equal and the third one is zero. The results were shown in Figure 3.1 - Figure 3.4. Notice that the epipolar line may not cross all the points correctly due to the extra constraint. The comparison of E(unconstraint) and E is shown in Figure 3.6. The results of E and Eh are quite different but the reconstruction of F and Fh by E and Eh matrices are quite similar, which can be visualized in MatLab.

4 Camera Matrix

After getting the Essential Matrix E, we should decompose E into translation and rotation matrix and get four possible matrix P. Following the equation in the slides, we can get four possible solution and can be visualized using showCamera.m. Then we use linear triangulation to find the correct matrix P. The key idea is that the reconstructed point should be in front of both cameras, which means the z-values of all points should be greater than zeros. The results were shown in Figure 4.1 - Figure 4.4. The latter ones are the correct solution for four possible P matrix shown in 3D plot, where the blue line points to the front of cameras.

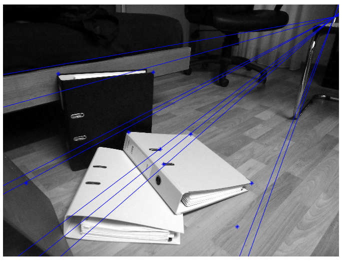

Figure 3.3: Epipolar Lines for estimating Essential Matrix: First View.

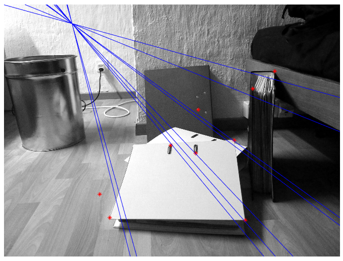

Figure 3.4: Epipolar Lines for estimating Essential Matrix: Second View.

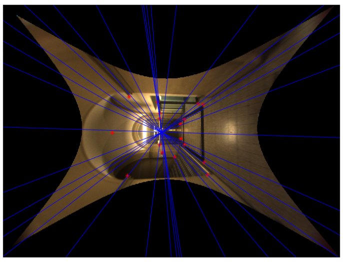

Figure 3.5: Result of F(unconstraint) matrix.

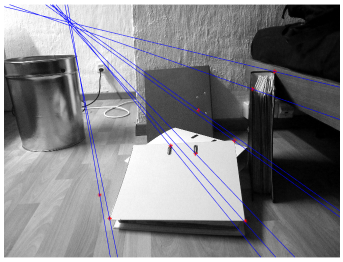

Figure 3.6: Result of E(unconstraint) matrix.

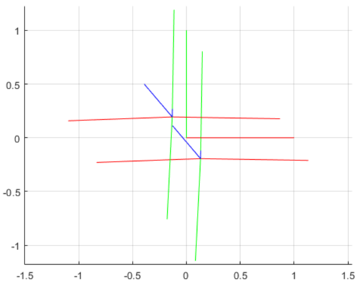

Figure 4.1: Four Possible Solution for Camera Matrix: First Image.

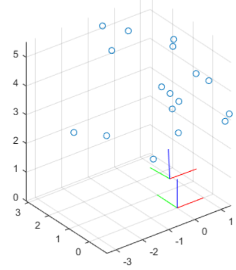

Figure 4.2: Correct Solution for Camera Matrix: First Image.

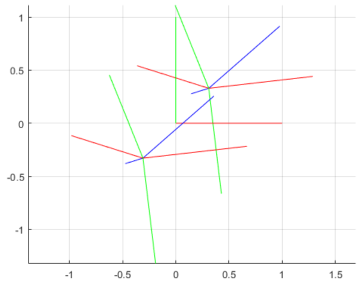

Figure 4.3: Four Possible Solution for Camera Matrix: Second Image.

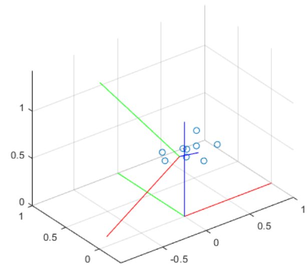

Figure 4.4: Correct Solution for Camera Matrix: Second Image.

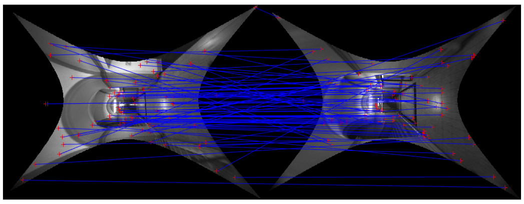

Figure 5.1: SIFT Feature Extraction and Matching.

5 Fundamental Matrix with RANSAC

5.1 Feature Extraction and Matching

The previous method for getting fundamental matrix is not so robust so here we want to use RANSAC to eliminate the influence of those outliers. I use the image pair of ladybug.

5.2 Simple RANSAC

As the methods used in line fitting RANSAC and eight-point algorithm, we combine these two methods in this task. First compute the fundamental matrix P and then get the number of inliers. If the current number of inliers is greater than best inliers number, then update the inliers, outliers and fundamental matrix. The error I used to determine whether a point is an inlier or outlier is residual error proposed in slides:

$$d = d(x', Fx) + d(x, F^T x') \tag{5.1}$$

The smaller d is, the more likely to be an inlier. I use the distance from point to line for computing the d. After 1000 iterations, the inlier ratio is 0.5435 with threshold equals 10.

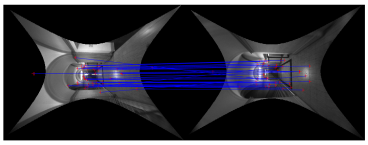

Figure 5.2: Inliers Matching, 1000 iterations, 10 threshold.

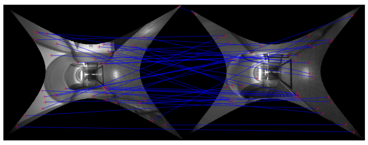

Figure 5.3: Outliers Matching, 1000 iterations, 10 threshold.

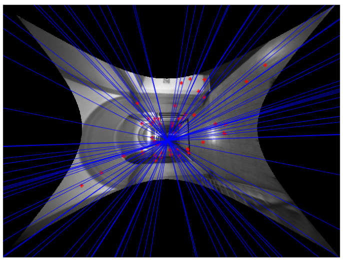

Figure 5.4: Epipolar Lines: First View, 1000 iterations, 10 threshold.

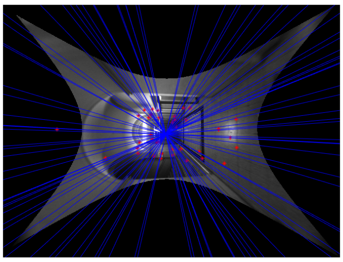

Figure 5.5: Epipolar Lines: Second View, 1000 iterations, 10 threshold.

5.3 Adaptive RANSAC

The difference of adaptive RANSAC I think is that it has a probability that the found matrix has no outliers. In this way, we can just calculate this probability p by $1-(1-r^N)^M$, where r is ratio of inliers, N is 8 for eight points and M is number of iterations. Here shows the results of RANSAC feature matching, inlier ratio, trials and epipolar lines. The number of iteration is 42753, which is far larger than 1000 and the inlier ratio is 0.3804, with threshold equals 2. Figure 5.6, 5.7 show the inliers and outliers respectively and Figure 5.4, 5.5 show the epipolar geometry of the computed F in the previous simple RANSAC.

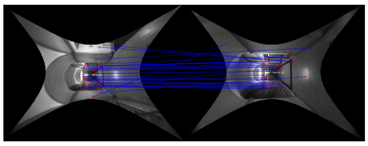

Figure 5.6: Inliers matching adaptively.

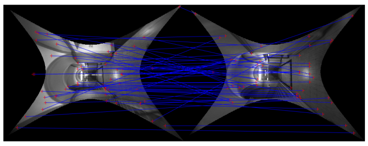

Figure 5.7: Outliers matching adaptively.