CSC2503 A2
Wuqi Li
1000292033
Part A Fundamental Matrix Estimation

1. According to the lecture slides, we have the equation of fundamental matrix.
   $F = \pi_L m \times \pi_L \pi_R^T (\pi_R \pi_R^T)^{-1}$
   π_L is the projection matrix for left image
   π_R is the projection matrix for right image
   m is from the null space of the π_R

   From the handout code, we know the M_{ext} and M_{int} for both right and left image.
   M_{ext} is a 3 x 4 matrix which can transform real world coordinate to camera center coordinate.
   M_{int} is a 3 x 3 matrix which can transform camera centered coordinate to image coordinate.

   => $M_{int} * M_{ext}$ is the projection matrix (π) which transform real world coordinate to image coordinate, therefore, we can get π_L and π_R.

   Cross product matrix is used to perform the cross product between a vector and a matrix. (From lecture slides)

   A x B = $[0, -a_3, a_2], [a_3, 0, -a_1], [a_2, a_1, 0]] * B$

   Therefore, we have everything we need to calculate Fundamental matrix. Plug in everything in the equation and the Fundamental matrix is calculated.

2. The error between F0 (ground-truth fundamental matrix) and F1(estimated fundamental matrix) is measured by the max perpendicular distance across all pairs epipolar lines generated by each of the matrix.

   The first step is calculating the epipolar lines. Left image is selected and each points in left image will be used to calculated the corresponding epipolar line the Right image.
   For each point in left image x_i, the epipolar lines are $x_i^T F0$ and $x_i^T F1$ since the camera for left image will be projected as null pointer in the second image.

   The next step is calculating the maximum perpendicular distance for each pair of epipolar lines.

Algorithm to find maximum distance between two lines

1) We need to find two boundary points for one epipolar line (generated from F0) using the function cropLineInBox().

2) Calculate the distance between the other line (generated from F1) and those two points, use the larger distance as the maximum distance between two lines.

Note for getting the maximum distance:

1. If two lines are parallel, any point in one line will have same distance two another line. => algorithm is correct.
2. If two lines are intersect at some point. Then the largest distance will appear at one of the boundary. => algorithm is correct since we use the max distance over all boundary points.

Final step, looping through all pairs of the epipolar lines and use the maximum distance as the error.

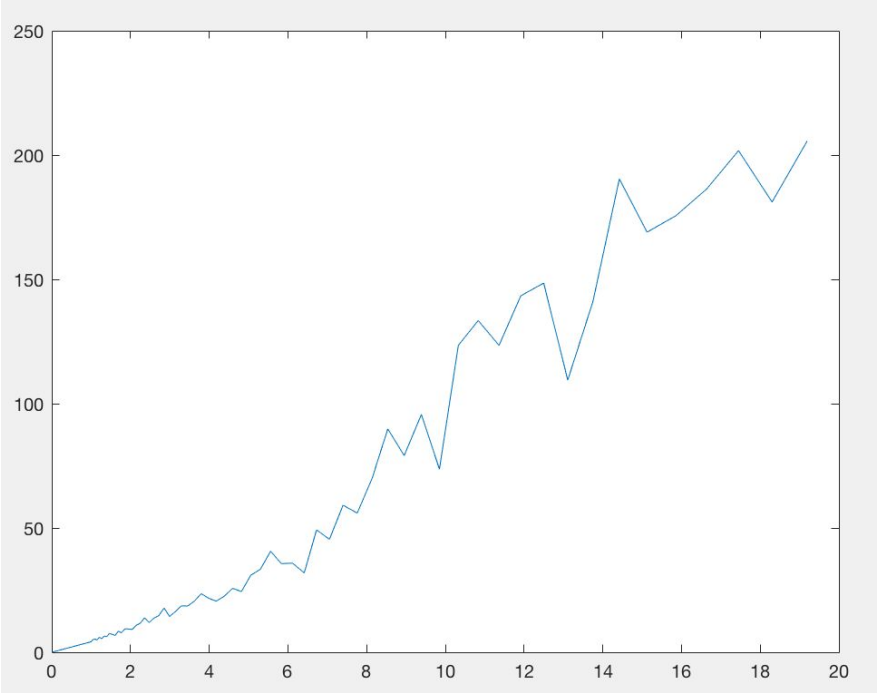
Note: according the dinoTest.m script, the points are draw in the region
(-150 < x < 150) and (-100 < y < 100). Therefore, The region for this part is same as the figure region.
Using this region. E = 2.3781e-12

3.

In order to try different sigma, a exponential growth function is used.
$\sigma = 1.2^n$, n is increased by 0.5 each loop.

For each sigma, we add Gaussian noise to each point with σ standard deviation. And calculate the estimate F1 using the Function linEstF() with the noisy points. After that, we can get the error from ground truth F0 and F1.

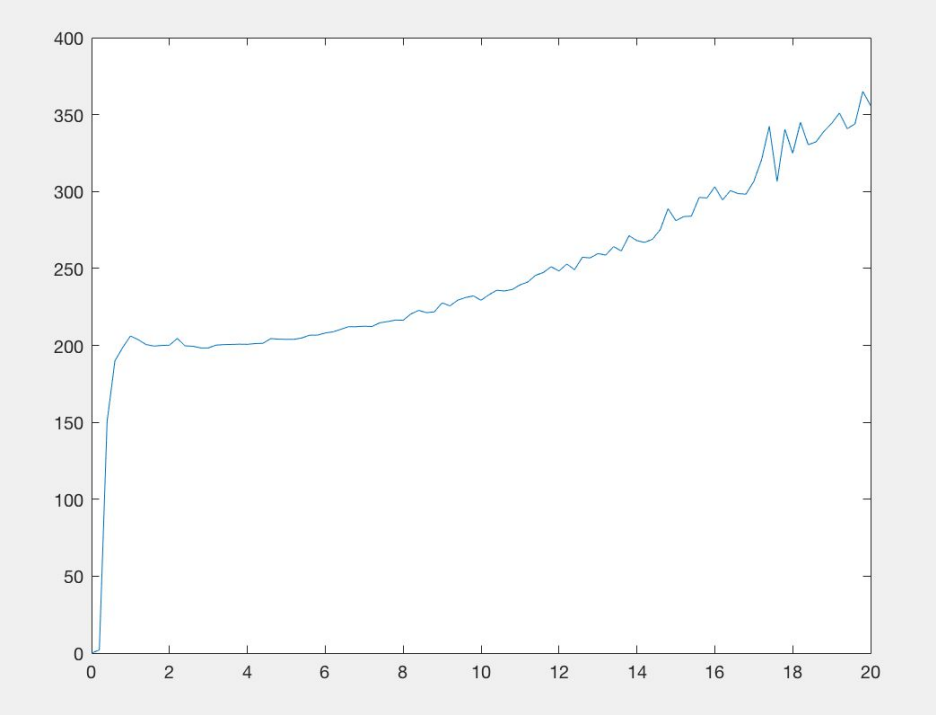
Previous step is repeated 10 times and a mean error is calculated for each sigma.



Plot for sigma and corresponding mean error

From the plot, we can see that the error is growing slowly with sigma less than 7. After that, error is growing rapidly. Therefore, the maximum sigma will be 7 in order to get a reasonable small error.

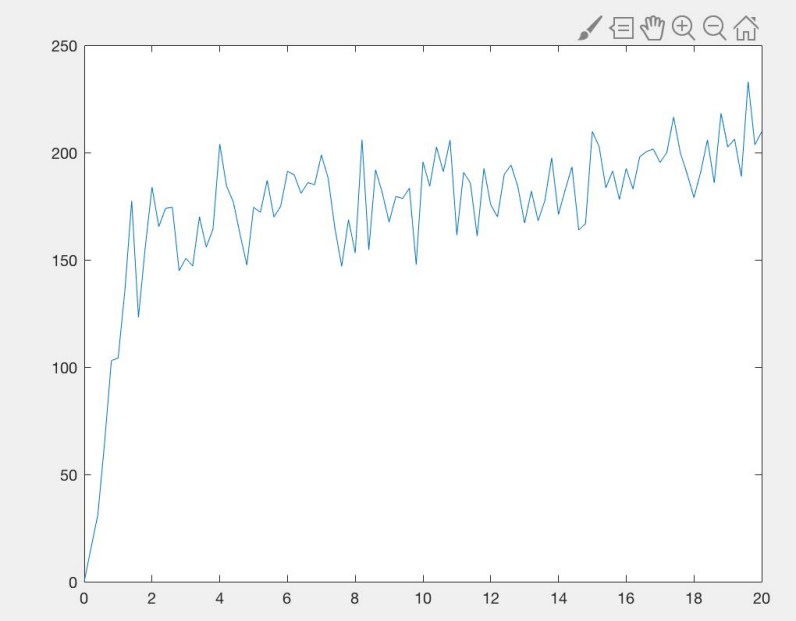
4.
By setting the NUM_RESCALE to 0, we get the following plot.



The plot for sigma and corresponding error

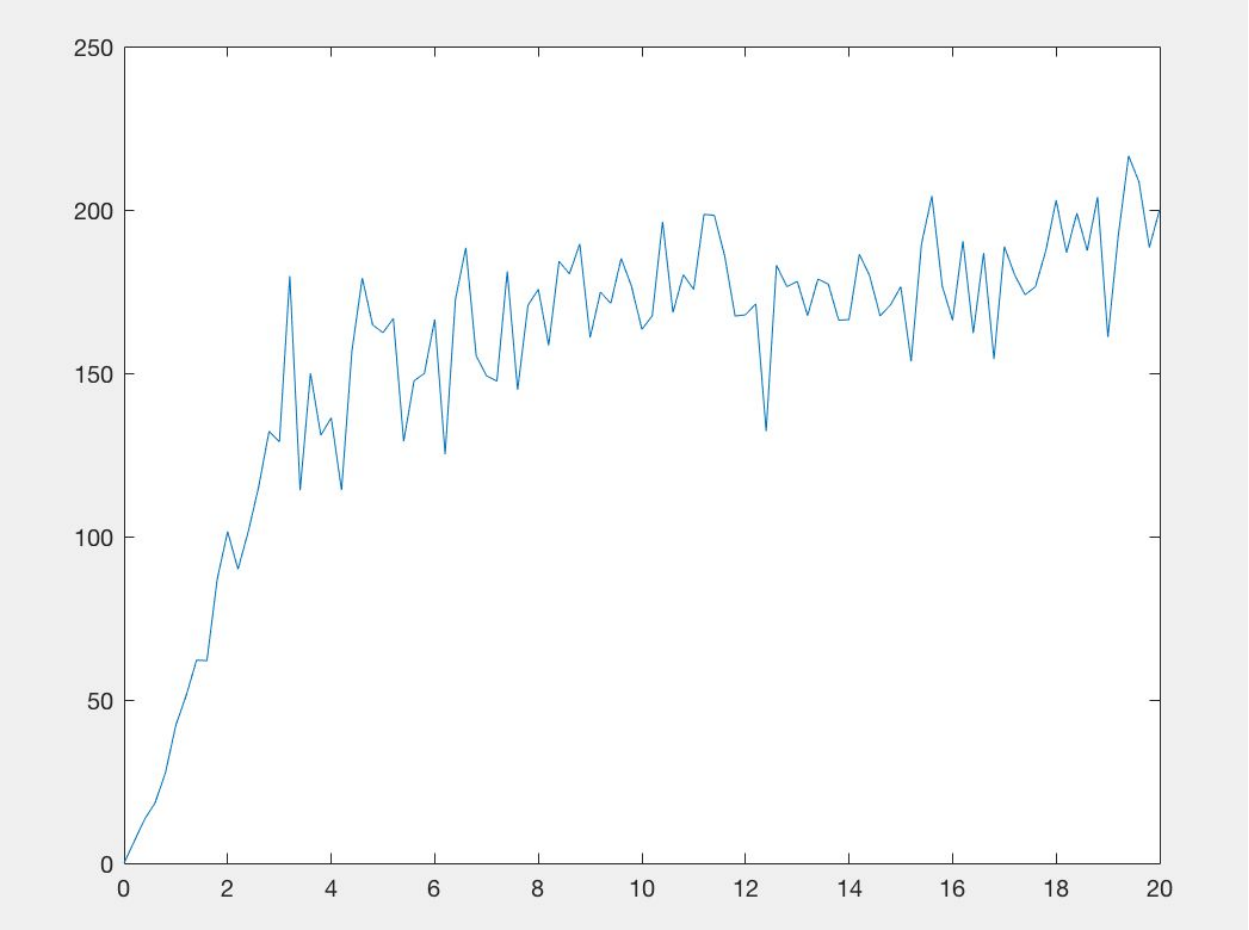
As seen from the figure, the error increase dramatically from a small value of sigma. Therefore, without the Hartley's normalization, the error will become very unstable.

5. The truth ground matrix is not changed. Since the sclZ only changed the magnitude of the homogenous vector. It will not change the relationship between one point to another point. Also, the fundamental matrix only represent a mapping from real world to the image coordinate. Therefore, sclZ will not affect the fundamental matrix.



The plot for sigma and corresponding error

Similar to part 4, the error grow rapidly from a very small value of sigma. For smaller sclZ, the noise will have larger impact on the epipolar lines, therefore, a larger error.



The plot for sigma and corresponding error

As seen from the plot, the Error grow very faster than the plot seen from part3. Therefore, we know that closer camera will result less tolerance to the noise.

Part B Homography Estimation

1. the min squared problem can be transferred as solving matrix.

$$\begin{bmatrix} q_{k1} \\ q_{k2} \\ 1 \end{bmatrix} \times \begin{bmatrix} h_1 h_2 h_3 \\ h_4 h_5 h_6 \\ h_7 h_8 h_9 \end{bmatrix} \begin{bmatrix} p_{k1} \\ p_{k2} \\ 1 \end{bmatrix} \Rightarrow \begin{bmatrix} q_{k1} \\ q_{k2} \\ 1 \end{bmatrix} \times \begin{bmatrix} p_{k1} * h_1 + p_{k2} * h_2 + h_3 \\ p_{k1} * h_4 + p_{k2} * h_5 + h_6 \\ p_{k1} * h_7 + p_{k2} * h_8 + h_9 \end{bmatrix}$$

Perform cross product

$$\begin{bmatrix} 0 & 0 & 0 & -p_{k1} & -p_{k2} & -1 & q_{k2} * p_{k1} & q_{k2} * p_{k2} & q_{k2} \\ p_{k1} & p_{k2} & 1 & 0 & 0 & 0 & -q_{k1} * p_{k1} & -q_{k1} * p_{k2} & -q_{k1} \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \\ h_6 \\ h_7 \\ h_8 \\ h_9 \end{bmatrix}$$

The last row is eliminated since it is the combination of the first and second row.

Therefore, for each point, we generate a matrix (left matrix) and append it at the end of matrix A. In the end, for n points, we got a (2n x 9) matrix for A.

Apply svd to solve for H. From the definition of SVD in lecture slides, we have $USV^T = A$. Similar as function linEstF, Sa = diag(S) and H0 is the last column of V.

After that, transform H0 to H. While doing Hartley's rescaling. Two Knum matrix K_q, K_p is calculated in order to normalize left points and right points. Therefore, we need to add those transformation back to H0 to calculate H.
$q_k = Hp_k$ => $q_{kt} = H0p_{kt}$
Also, we have $q_{kt} = q_k K_q$ and $p_{kt} = p_k K_p$
=> $H = K_q^{-1} H0 K_p$

After that, normalize H to have norm2 = 1.

2. Similar as calculating F. The process of RANSAC is similar as in the grappleFmatrix.m except several modification.

1) Error measurement. For each pair of point. Calculate the right point by multiplying H and left point. After that, calculate the distance between the real right point and the calculated point. The inlier points are the points with error smaller than (sigma * rho).
2) Since the H is calculated based on mapping (left -> right). The inverse of H is used when calculating the homography image of left image.