1 RANSAC Line Fitting

As we can observe in Figure 1 the real model is being approximated well by RANSAC unlike the green line which was computed using least squares. This is because the least squares line fitting algorithm minimizes the error over all points making it susceptible to outliers. This is not the case for RANSAC as it uses the count of inliers as a guideline for fitting the line. This is reflected in the error output, the real model as well as RANSAC have almost the same error whereas least squares has a lower error, this is in line with the previous statement that least squares minimizes the distance to all points and hence gives the minimum error achievable with a linear function.

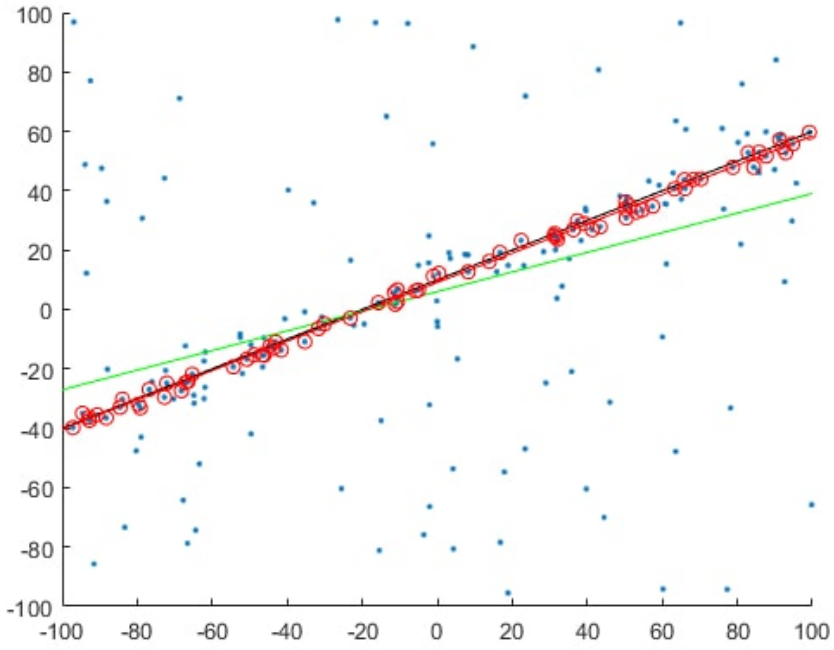


Figure 1: green line represents least square, black line represents real model, red line represents ransac.

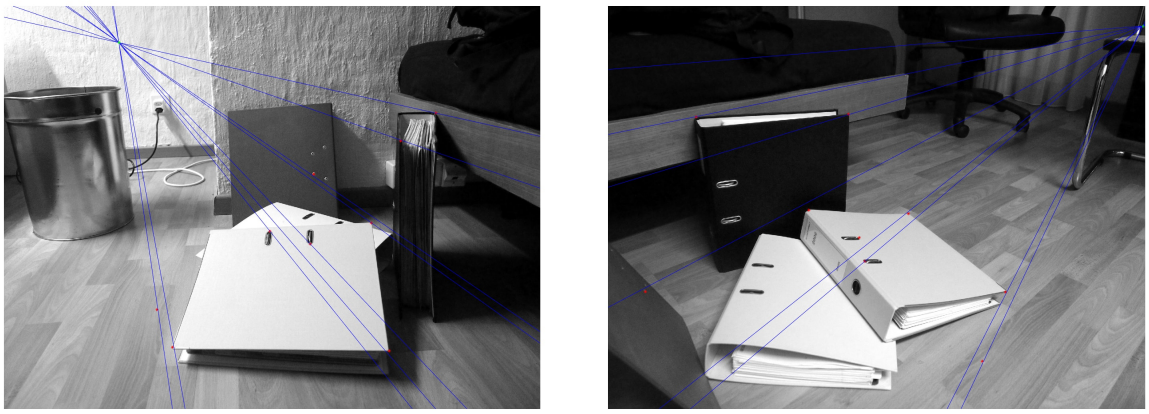
2 Fundamental Matrix

As we can observe in Figure 2 for the case where we constrain the fundamental matrix by setting the last singular value to zero, all epipolar lines pass through a single point, the epipole. This epipole represents the projection of the other cameras position into the current image. If we however remove the constraint we can clearly see (Figure 3) that the epipolar lines do not all intersect at the same location.

3 Essential Matrix

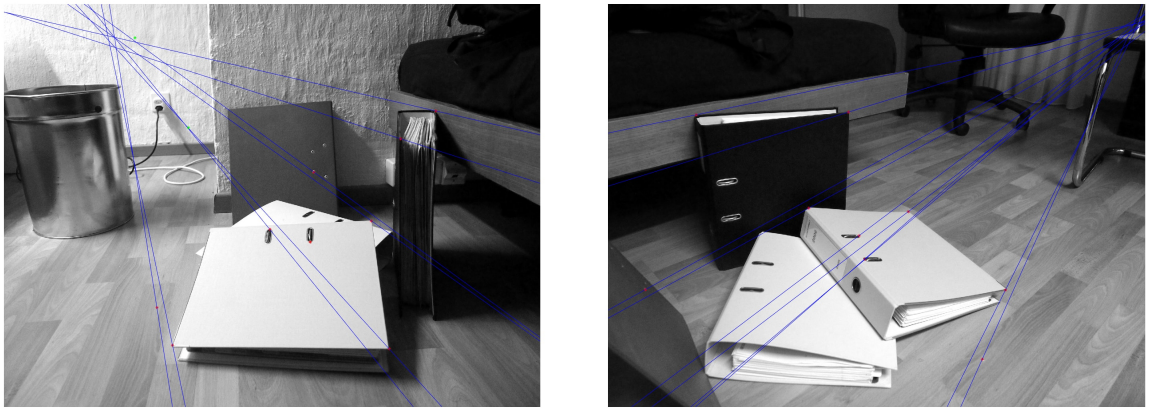
By constraining the last singular value in the essential matrix to zero, all the epipolar lines pass through the epipole. However through the additional constraint on the first two singular values of the essential matrix the epipolar lines cannot go through every clicked point as can be seen in Figure 5. In the following we also list the two matrices corresponding to the essential and fundamental matrix taken from the ladybug image.

$$K^{-T}E_H K^{-1} = \begin{bmatrix} -0.0000 & -0.0002 & 0.0774 \\ 0.0002 & -0.0000 & -0.0951 \\ -0.0733 & 0.0967 & -1.9919 \end{bmatrix} \quad F_H = \begin{bmatrix} -0.0000 & -0.0000 & 0.0006 \\ 0.0000 & -0.0000 & -0.0007 \\ -0.0006 & 0.0007 & -1.8336 \end{bmatrix}$$



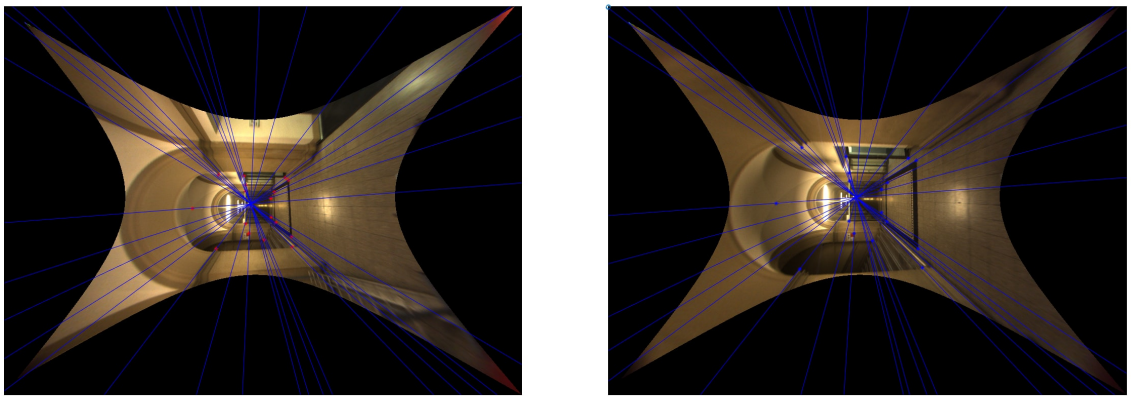
(a) (b)

Figure 2: constrained fundamental matrix



(a) (b)

Figure 3: unconstrained fundamental matrix

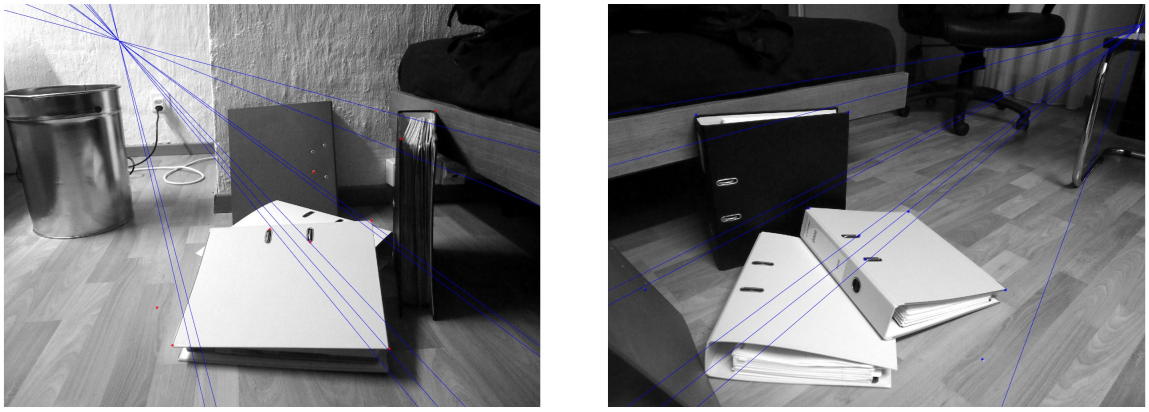


(a) (b)

Figure 4: epipolar lines using essential matrix

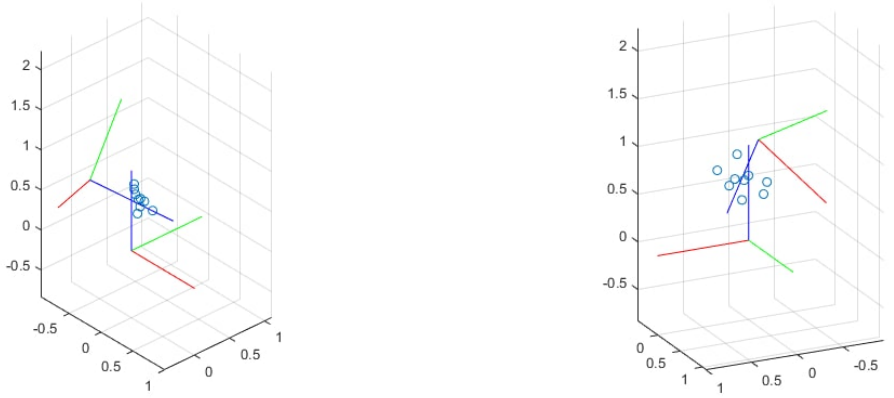
4 Camera Matrix

As we can observe in Figure 6 the points lie along the z-axis in the xy-plane of both cameras which are given by the camera matrices P_0 and P_s.



(a) (b)

Figure 5: epipolar lines using essential matrix



(a) (b)

Figure 6: camera matrix from two different view points

5 Feature Matching Extraction

The extracted features and the corresponding matches from SIFT can be seen in Figure 7.

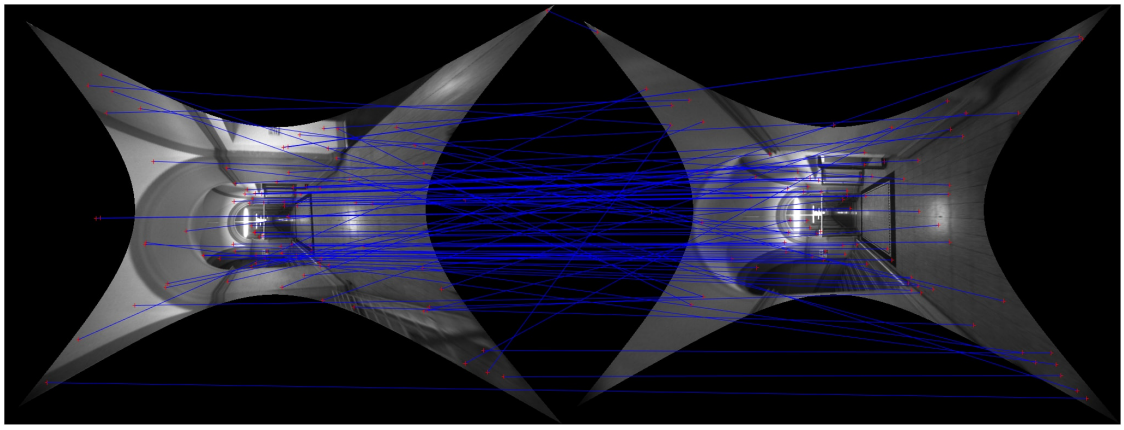


Figure 7: Ladybug image, feature matching from SIFT

6 8-Point RANSAC

For the standard RANSAC algorithm with an error threshold of 2 pixels, running for 1000 iterations the inlier count is around 20 with an inlier ratio of around 20%. If we change the error threshold from 2 to 5 pixels we have an inlier count of around 40 with an inlier ratio of around 40%.

Since RANSAC is random in nature it can end up with the wrong decision as can be seen in Figure 8, however, out of many runs the correct decision is made more often than not. This issue is reduced by running the algorithm for m iterations where $m = log(1-p)/log(1-r^n)$ and p is the desired confidence and n the number of samples. As we can observe in Figure 9 where the threshold was increased to 5 pixels more features are matched since there are more inliers that now lie within the 5 pixel error margin.

For the adaptive RANSAC we use the equation described above to compute the number of iterations it would take to reach the correct answer with 99% probability. For 8 samples and a desired confidence of 0.99 with an error threshold of 10 pixels we need roughly $m = 1500$ iterations, the result can be seen in Figure 10. For all figures the inliers are circled in green.

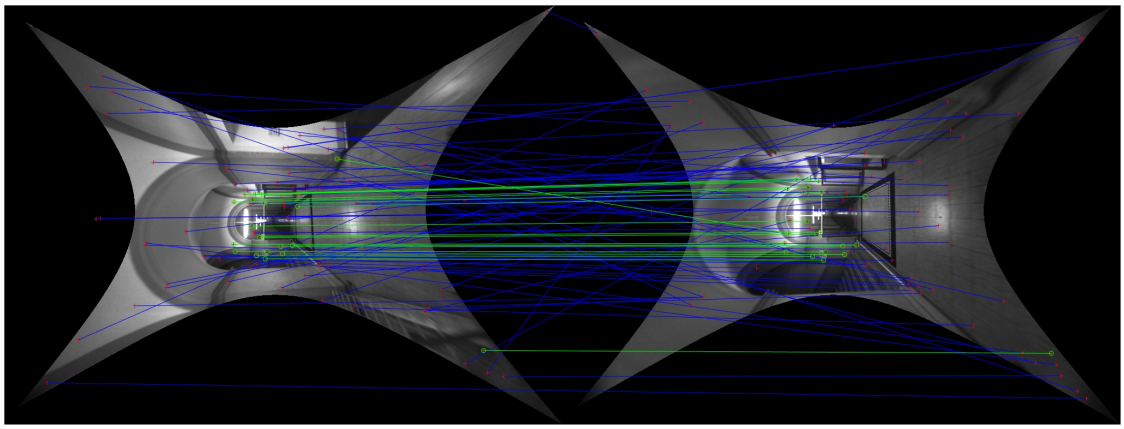


Figure 8: 8-Point standard RANSAC with 2 pixel threshold

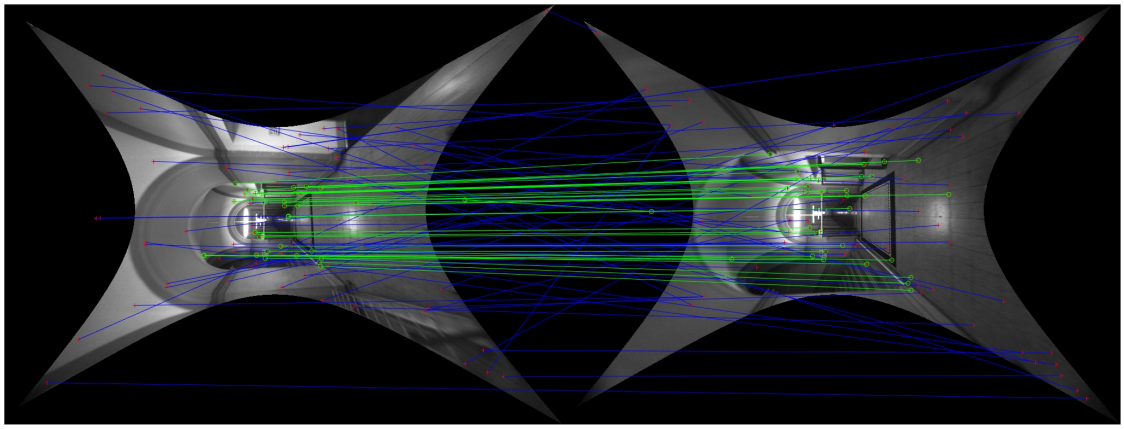


Figure 9: 8-Point standard RANSAC with 5 pixel threshold

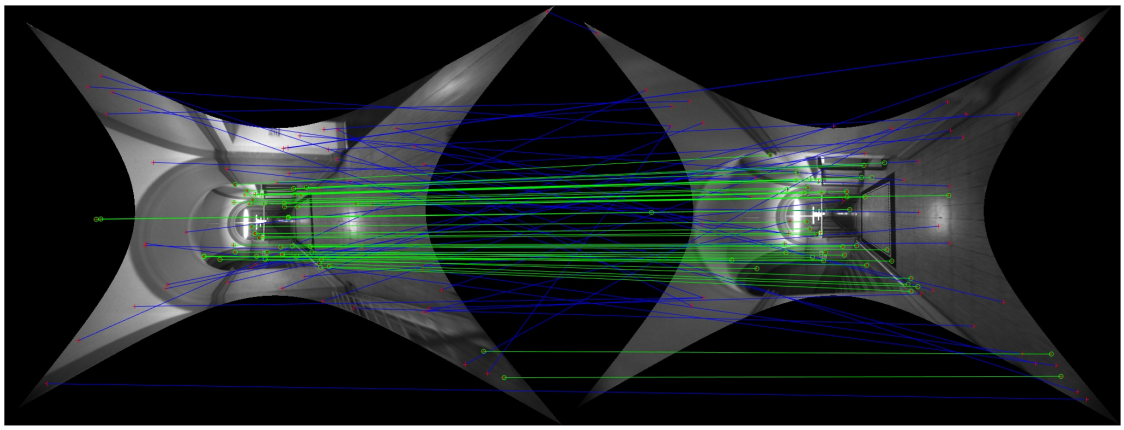


Figure 10: 8-Point adaptive RANSAC with 10 pixel threshold & m = 1459 iterations.