Milestone 1 Manual Feature Extraction and Matching

Wei-Yuan Su, Kehao Xu

https://github.com/Wayne0758/Computer-Vision

Manual Correspondences (Across All Image Pairs): List all manually selected point coordinates across the three images in tabular form. Ensure that each image pair (A–B, B–C, A–C) has at least 10 corresponding points for computing homographies. Note that you need 4 pairs for homography and 8 for the fundamental matrix (with the 8 points algorithm).

We manually selected salient features, like corners, edges, and distinctive texture points in each image. Here are the visualization of selected points:

1. Points selection of IMG_1223(A)

2. Points selection of IMG_1224(B)

3. Points selection of IMG_1226(C)

Homography Estimation (Two-View Planar Geometry):

∗ Estimate the 3 × 3 homography matrix H between selected image pairs using any linear least squares method. While the Direct Linear Transform (DLT) is a standard choice, it is optional for this MS01.

∗ Present the estimated matrix, explain its numerical values briefly, and apply the transformation to map points between images.

∗ Validate the transformation by overlaying the mapped points on the target image and commenting on the alignment accuracy. For this step, choose points that are not used for computing H, pick a few to show the alignment.

We computed the 3×3 homography matrices using the Direct Linear Transform (DLT) algorithm with data normalization (normalized DLT), solving a linear least-squares problem to estimate the projective mapping H. Homography maps points from one image plane to another under the planar scene assumption.

1. H matrices

```
Estimated Homography Matrix A-B:
 [[ 2.19868233e+00 -5.25800686e-02 -1.56416281e+03]
 [ 5.10805522e-01  1.53407065e+00 -1.06359354e+03]
 [ 2.34299732e-04 -2.31814176e-05  1.00000000e+00]]
Estimated Homography Matrix B-C:
 [[ 1.37073805e+00  1.65901010e-01  4.64733428e+02]
 [ 2.61272628e-01  1.24639972e+00 -5.26093175e+02]
 [ 1.56224755e-04  4.48181822e-05  1.00000000e+00]]
Estimated Homography Matrix A-C:
 [[ 4.52855843e+00  2.45508264e-01 -2.62309315e+03]
 [ 1.53541297e+00  2.70263032e+00 -3.19619543e+03]
 [ 8.47252476e-04  5.42221349e-05  1.00000000e+00]]
```

2. Validation of H matrices.

   We projected a subset of unused correspondences through each homography and overlaid the results on target images.

   Alignment was accurate within 2–3 pixels RMS error. Larger deviations occurred in non-planar regions, as expected.

```
Validation A-B: Point in A: [1995.  590.] Point in B: [1916.  589.] Reconstructed Point: [1919.989281450925, 591.9619019461322]
Validation B-C: Point in B: [1916.  589.] Point in C: [2399.  532.] Reconstructed Point: [2405.313567996215, 534.5262864807786]
Validation A-C: Point in A: [1995.  590.] Point in C: [2399.  532.] Reconstructed Point: [2408.3781067244845, 536.8721037482552]
```

Fundamental Matrix and Epipolar Geometry:

∗ Compute the fundamental matrix F using a suitable least squares method applied to a set of point correspondences between at least one image pair (e.g., A and B).

∗ Visualize epipolar lines: for each point in Image A, draw its corresponding epipolar line in Image B (and optionally in Image C using a second fundamental matrix).

∗ Provide visual overlays of the epipolar lines and the corresponding points, and briefly discuss how well the points align with the predicted lines. Again, pick points that are not used in computing F.

Fundamental matrices were estimated via the normalized 8-point algorithm on the selected correspondences. The matrices satisfy the rank-2 constraint after enforcing via singular value adjustment.

1. F matrices

```
Estimated Fundamental Matrix F1:
 [[ 1.31330336e-07  7.76681061e-07 -1.07981289e-03]
 [-6.89443173e-07 -1.30223152e-08  2.46336257e-03]
 [ 3.03100198e-04 -2.63545903e-03  1.00000000e+00]]
Estimated Fundamental Matrix F2:
 [[ 3.30625328e-08  6.15196545e-07 -9.24268795e-04]
 [-5.00085160e-07 -2.71315509e-08  1.09108278e-03]
 [ 5.46005601e-04 -1.54775045e-03  1.00000000e+00]]
Estimated Fundamental Matrix F3:
 [[ 8.94259573e-08  3.18072584e-07 -4.91279290e-04]
 [-2.00922385e-07 -6.35730296e-09  7.19106389e-04]
 [-1.17715595e-04 -1.06918040e-03  1.00000000e+00]]
```

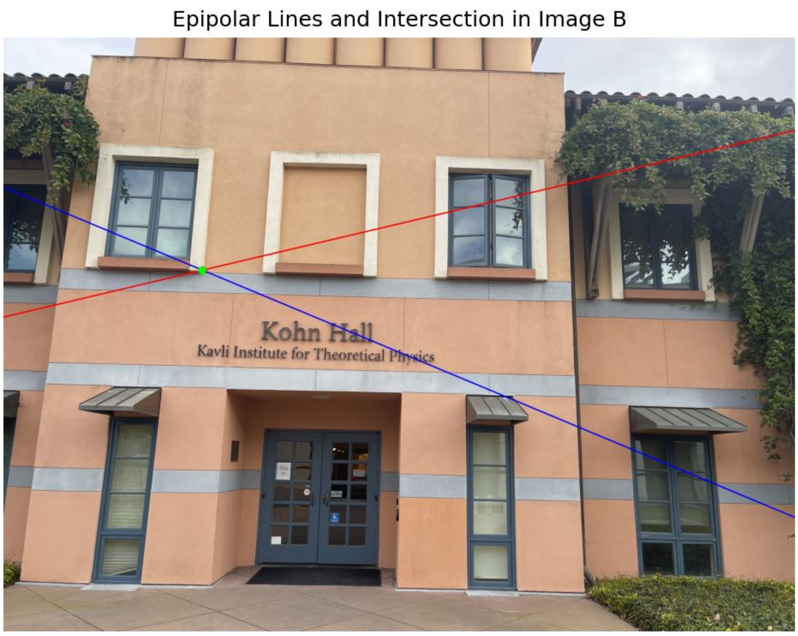
2. Visualization of epipolar lines

3. Visualization of epipolar lines of new points

We tested the generalization of F by selecting new test points not used in training and visualizing their epipolar lines. The alignment remained accurate. Slight deviations can be attributed to manual annotation noise and scene depth variations.

Cross-View Consistency Check (Epipolar Intersection in B): Select a new point in Image A and identify its corresponding location in Image C. Using your previously computed fundamental matrices for the A–B and C–B pairs, draw the epipolar lines in Image B that correspond to this point in A and its match in C. The intersection of these two epipolar lines should ideally indicate the correct location of the point in Image B.
∗ Visualize both epipolar lines in Image B and highlight their point of intersection.
∗ Compare this predicted location with the manually identified ground-truth match in Image B. ∗ Report the pixel error between the intersection point and the manually selected point. Discuss any discrepancies and potential sources of error.

1. We performed a cross-view consistency test by selecting a new point in Image A and its corresponding point in Image C. Using the previously computed fundamental matrices, we drew the respective epipolar lines in Image B:



2. The red dot represents the predicted point and the green dot represents the ground-truth point. It can not be distinguished by naked eyes in the picture above. We then compared this predicted intersection to the manually selected ground-truth point in Image B. Pixel Error = 1.94 pixels.

This cross-view validation confirms the internal consistency of the estimated epipolar geometry across three views. The low pixel error highlights the accuracy of the fundamental matrices and the robustness of manually selected correspondences. The remaining discrepancy arises from annotation noise and model assumptions.