Junseo Kim

Assignment 4

Summer 2023

The objective of this course is to learn how to distinguish inliers from out-liers in statistical inference. For more information, refer to 'Ch 5.3 Handling Outliers' of your textbook.

Given the source and destination 2d points below, compute a homography that maps the source points to the destination. a minimum of 10 inliers needed tobe detected by the RANSAC algorithm. The re-projection error of the inliers need to be less than 0.005.

Report your normalized homography transformation. Write your code in Python, do not use OpenCV. Plot a scatter plot, showing inliers (marked by o)and outliers (marked by x), where inliers of source and destination are connected via a line. Include the plot in your report. You should only use the following modules:
import numpy as npimport
random
import matplotlib.pyplot as plt

In the first part, homography should be defined based on the source points and destination points.

Assuming the source points are in (x,y) form and the destination points are in (u,v), the following equation is considered for the homography:

$$A = \begin{bmatrix} x & y & 1 & 0 & 0 & 0 & -u*x & -u*y & -u \\ 0 & 0 & 0 & x & y & 1 & -v*x & -v*y & -v \end{bmatrix}$$

Once matrix A is constructed, Singular Value Decomposition (SVD) was used for the system. This will generate matrix V.
The last row of the matrix V will give the solution of the system. This is reshaped into 3x3 matrix for the final homography matrix.

RANSAC function

1. Get four random points

As homography is defined by 8 parameters, with each point pair providing 2 equations, we need at least 4 point pairs to solve for 8 unknowns.

2. Compute H using DLT

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = H \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Assuming H is the homography matrix, (x,y) are the coordinates for source points and (x', y') are the coordinates for projected points, projected points are converted to inhomogeneous coordinates by

dividing by w’

$$\begin{bmatrix} x'' \\ y'' \end{bmatrix} = \begin{bmatrix} \frac{x'}{w'} \\ \frac{y'}{w'} \end{bmatrix}$$

3. Inlier detection

Euclidean distance between the projected points and destination points are calculated. As the threshold is set as 0.005 from the requirement, if the error is less than the threshold, the point pair will be considered as an inlier.

$$error = \sqrt{(u - x'')^2 + (v - y'')^2}$$

4. Update

If the number of inliers for the model is greater than the number of inliers for the best model, the function will update the best model and the maximum number of inliers.

```
Homography Transformation:
[[0.71377842 0.42900386 0.12835417]
 [0.28547466 0.17162952 0.42849705]
 [0.15697198 0.01441864 1.        ]]
```

Figure 1: Best model using RANSAC

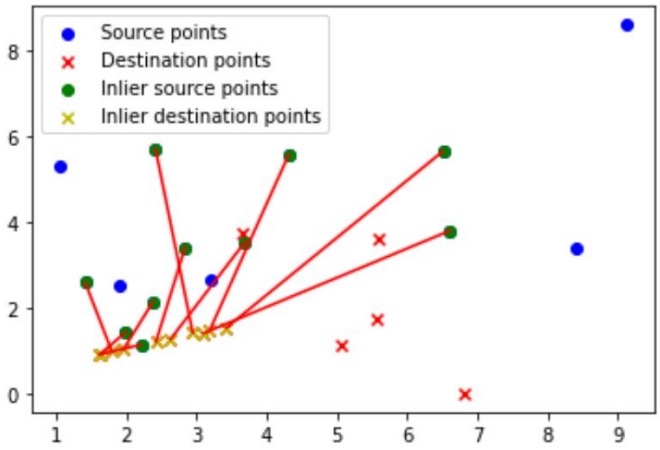

Figure 2: RANSAC example

From figure 2, 10 inliers are detected with 15 source points and destination points

Detailed code can be found from
https://github.com/JunseoKim19/State_estimation/blob/main/RANSAC.py