1. A brief description of the programs you implement.

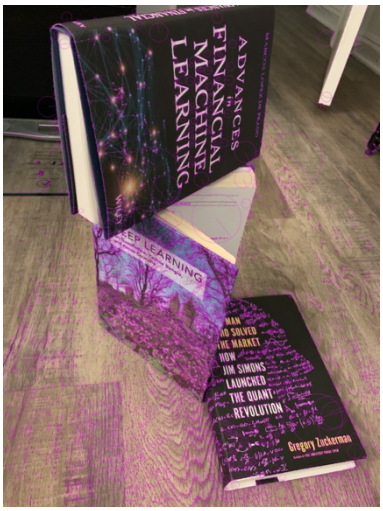
1. We iterate through each combination of source image and destination image
2. Using SIFT operation find the key points and descriptors
3. Draw the keypoints on the image (use the arg cv.DRAW_MATCHES_FLAGS_DRAW_RICH_KEYPOINTS to get the directions)
4. Use the bruteforce knn matches
5. Filter out the good matches using the ratio test (0.75)
6. Sort the good matches matches found based on the distance
6. Draw the top 20 matches on the images
7. Find homography matrix and mask using findhomography function with RANSAC
8. Find the number of inliers from the mask obtained.
9. Find distances between dest keypoints and projected points after applying the homography matrix operation for projection, sort the points according the differences (error)
10. Draw the top 10 matches with min error.

2. Show the results of intermediate steps as listed in the descriptions above.

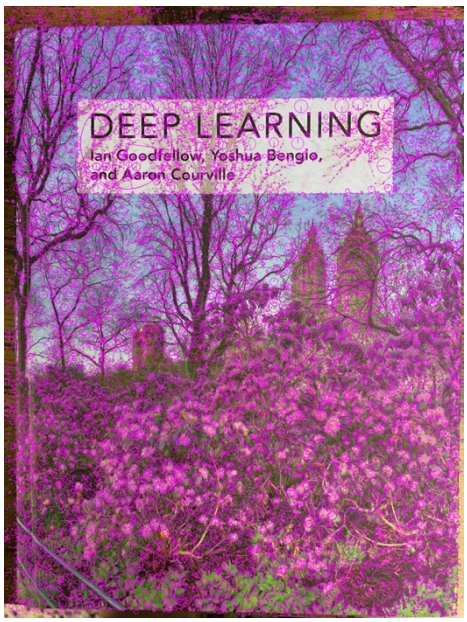
a. SIFT features: show the detected features overlaid on the images (*both the locations and directions, not 128-d vectors*). Also give out *the number of features detected* in each image.

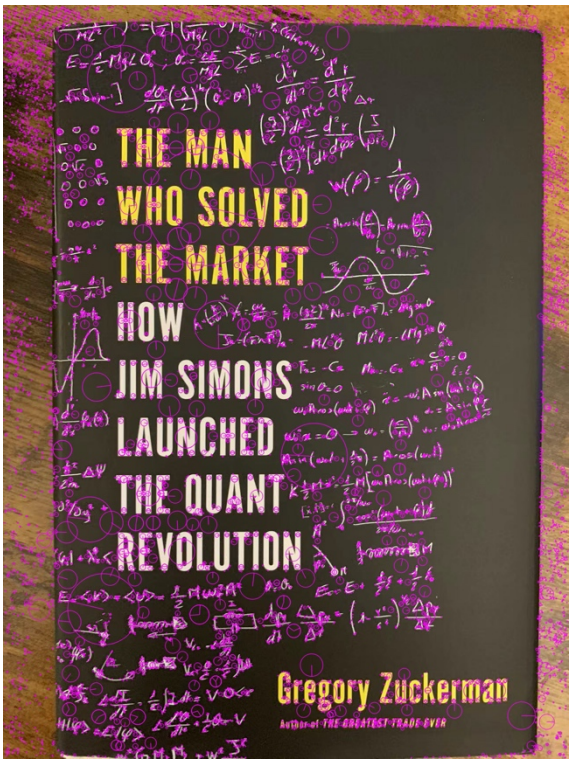
No of Key Points found using SIFT in image src_0.jpg 5259
No of Key Points found using SIFT in image src_1.jpg 42457
No of Key Points found using SIFT in image src_2.jpg 13004

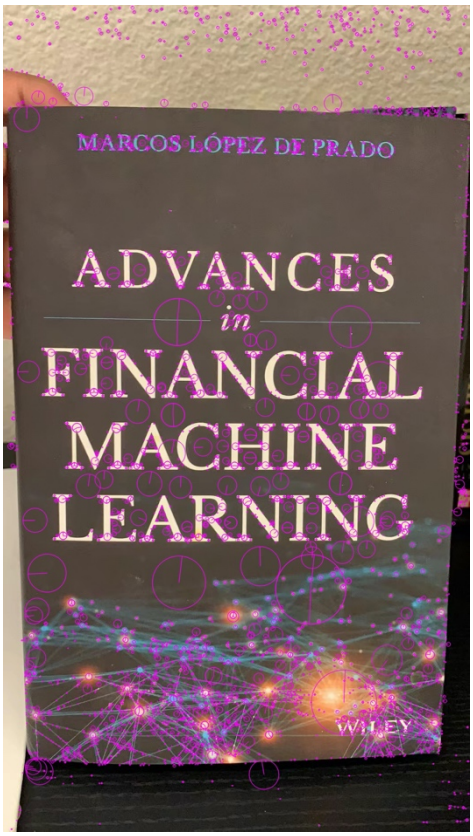
No of Key Points found using SIFT in image dst_0.jpg 49979
No of Key Points found using SIFT in image dst_1.jpg 10625


ADVANCES
FINANCIAL
MACHINE
LEARNING
DEEP LEARNING
Gregory Zuckerman


ADVANCES
in
FINANCIAL
DEEP LEARNING
THE QUANT
REVOLUTION


DEEP LEARNING
Ian Goodfellow, Yoshua Bengio,
and Aaron Courville


THE MAN
WHO SOLVED
THE MARKET
HOW
JIM SIMONS
LAUNCHED
THE QUANT
REVOLUTION
Gregory Zuckerman


MARCOS LÓPEZ DE PRADO
ADVANCES
in
FINANCIAL
MACHINE
LEARNING
WILEY

b. Graphically show the top-20 scoring matches found by the matcher before the RANSAC operation. Provide statistics of how many matches are found for each image pair.

Number of good matches found for src_0.jpg dst_0.jpg 219

Number of good matches found for src_0.jpg dst_1.jpg 529

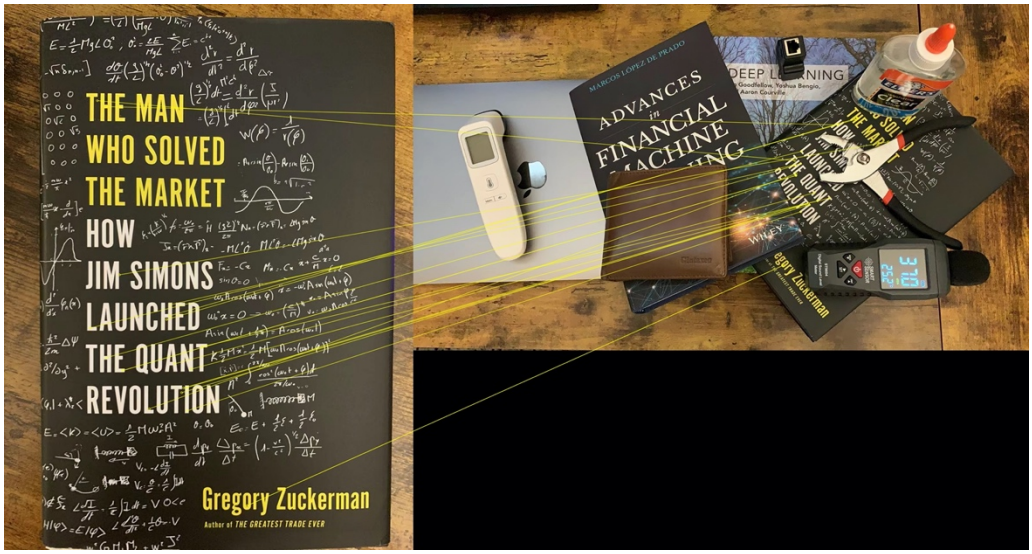
Number of good matches found for src_1.jpg dst_0.jpg 155

Number of good matches found for src_1.jpg dst_1.jpg 606

Number of good matches found for src_2.jpg dst_0.jpg 1241

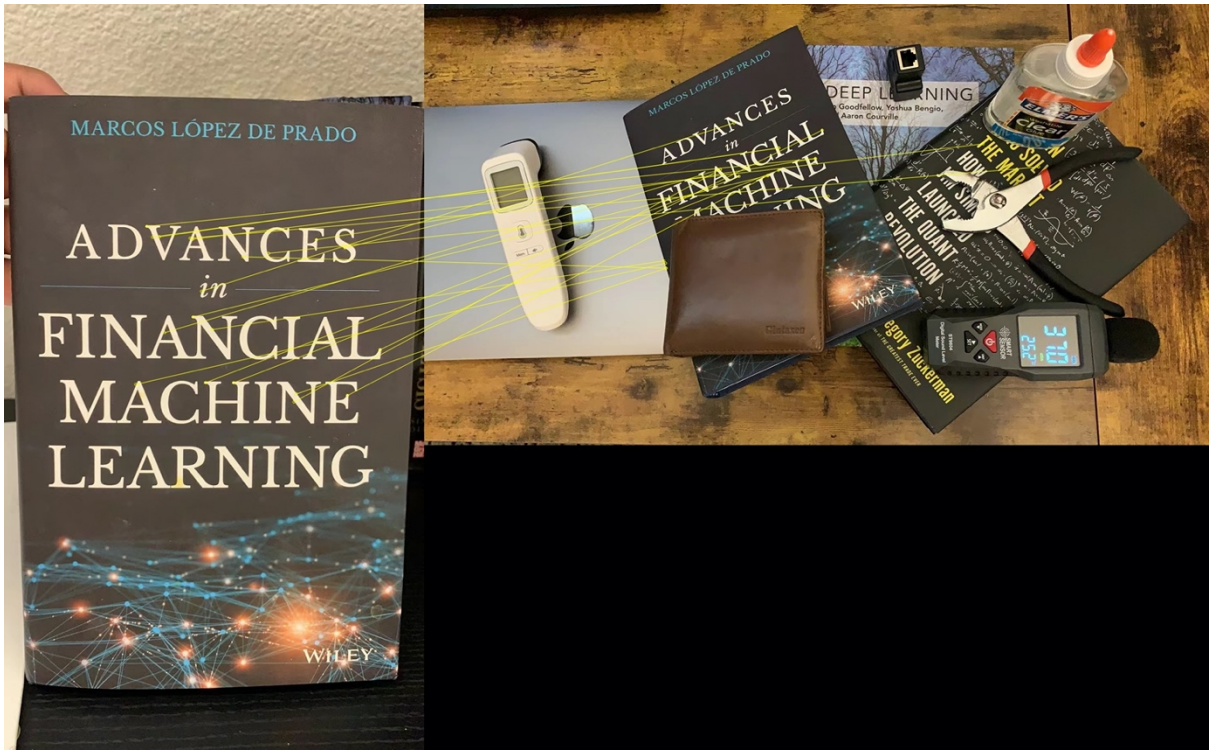
Number of good matches found for src_2.jpg dst_1.jpg 1047


DEEP LEARNING
Ian Goodfellow, Yoshua Bengio, and Aaron Courville
ADVANCES in FINANCIAL MACHINE LEARNING
DEEP LEARNING
HOW JIM SIMONS LAUNCHED THE QUANT REVOLUTION
Gregory Zuckerman


THE MAN WHO SOLVED THE MARKET
HOW JIM SIMONS LAUNCHED THE QUANT REVOLUTION
Gregory Zuckerman
Author of THE GREATEST TRADE EVER
MARCOS LÓPEZ DE PRADO
ADVANCES in FINANCIAL


THE MAN
WHO SOLVED
THE MARKET
HOW
JIM SIMONS
LAUNCHED
THE QUANT
REVOLUTION
Gregory Zuckerman
ADVANCES IN FINANCIAL MACHINE LEARNING
MARCOS LOPEZ DE PRADO
WILEY
EEP LEARNING
MAN
O SOLVED
HE MARKET
HOW
JIM SIMONS
LAUNCHED
THE QUANT
REVOLUTION
Gregory Zuckerman


MARCOS LÓPEZ DE PRADO
ADVANCES in FINANCIAL MACHINE LEARNING
WILEY
DEEP LEARNING
THE QUANT REVOLUTION
Gregory Zuckerman


MARCOS LÓPEZ DE PRADO
ADVANCES in FINANCIAL MACHINE LEARNING
WILEY
DEEP LEARNING
THE QUANT REVOLUTION

c. Show total number of inliner matches after homography estimations. Also show top-10 matches that have the minimum error between the projected source keypoint and the destination keypoint. (Hint: check the mask value returned by the function estimating the homography)

No of inliers found for image combination src_0.jpg dst_0.jpg 48

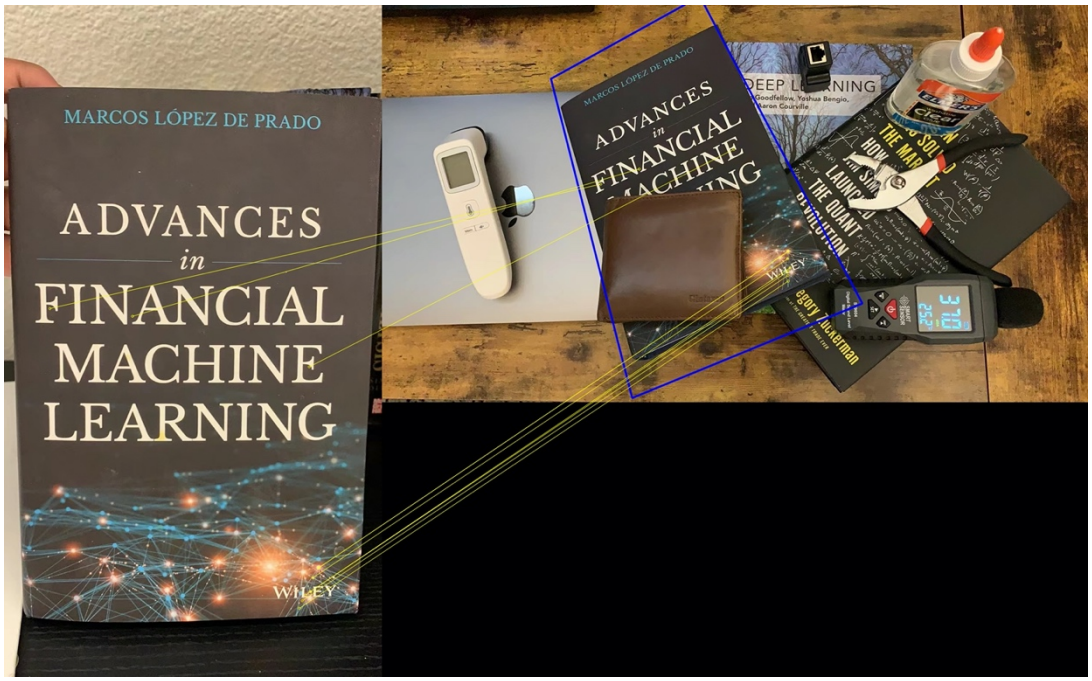
No of inliers found for image combination src_0.jpg dst_1.jpg 243

No of inliers found for image combination src_1.jpg dst_0.jpg 30

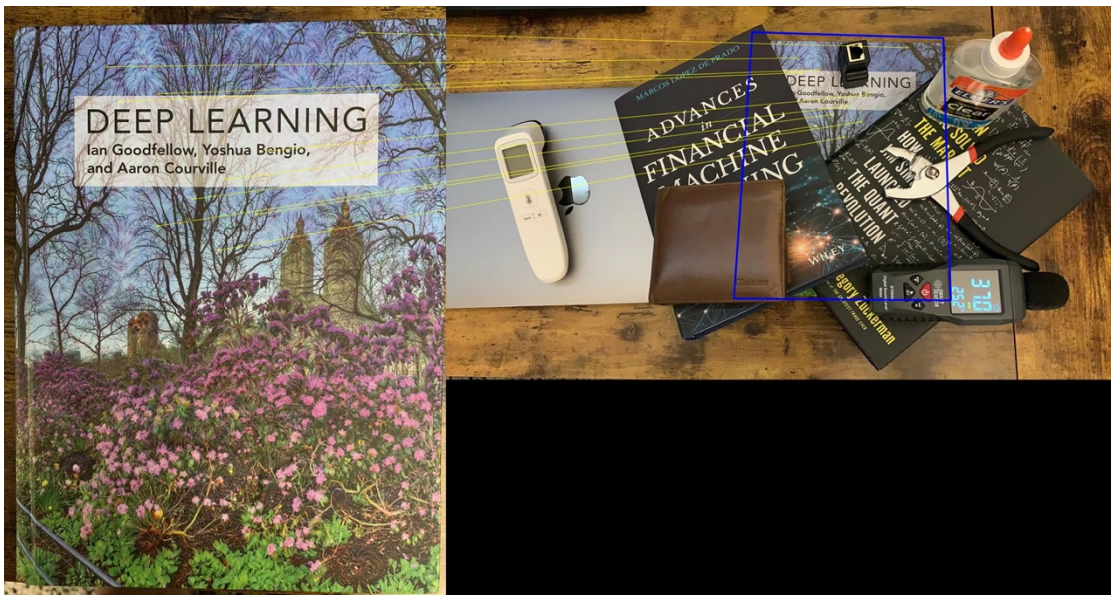
No of inliers found for image combination src_1.jpg dst_1.jpg 427

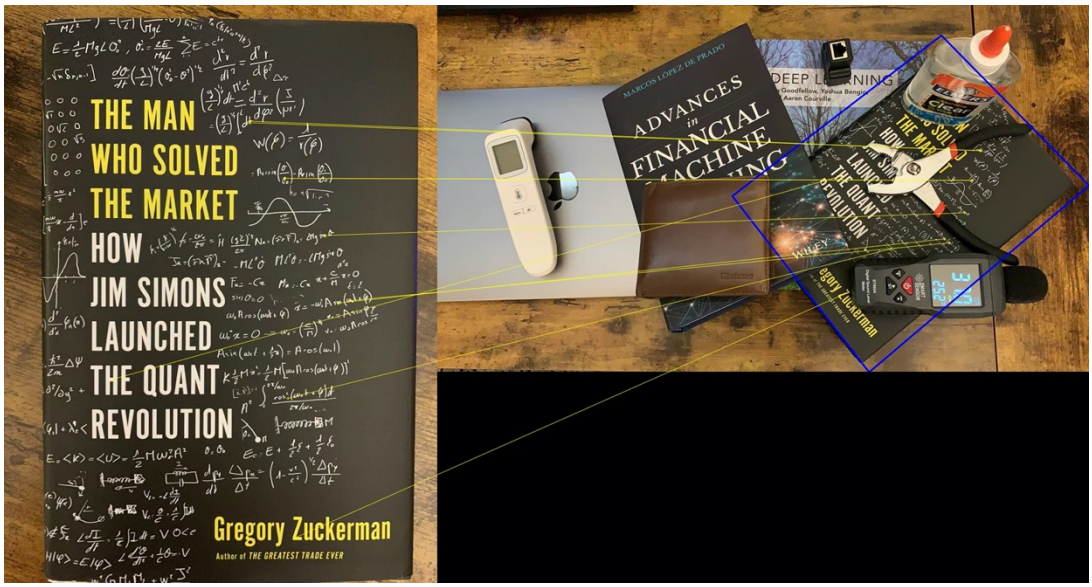
No of inliers found for image combination src_2.jpg dst_0.jpg 635

No of inliers found for image combination src_2.jpg dst_1.jpg 402


MARCOS LÓPEZ DE PRADO
ADVANCES
in
FINANCIAL
MACHINE
LEARNING
WILEY


DEEP LEARNING
Ian Goodfellow, Yoshua Bengio,
and Aaron Courville
ADVANCES in FINANCIAL MACHINE LEARNING
MARCOS LÓPEZ DE PRADO
WILEY
THE QUANT REVOLUTION
Gregory Zuckerman


DEEP LEARNING
Ian Goodfellow, Yoshua Bengio,
and Aaron Courville
ADVANCES in FINANCIAL MACHINE LEARNING
MARCOS LÓPEZ DE PRADO
WILEY
THE QUANT REVOLUTION
Gregory Zuckerman


THE MAN
WHO SOLVED
THE MARKET
HOW
JIM SIMONS
LAUNCHED
THE QUANT
REVOLUTION
Gregory Zuckerman
ADVANCES
FINANCIAL
MACHINE


THE MAN
WHO SOLVED
THE MARKET
HOW
JIM SIMONS
LAUNCHED
THE QUANT
REVOLUTION
Gregory Zuckerman
ADVANCES
FINANCIAL
MACHINE
LEARNING

d. Output the computed homography matrix. (The built-in homography finder also applies a non-linear optimization step at the end; you can ignore or disable this step it if you wish.)

Homography Matrix for src_0.jpg dst_0.jpg
[[3.50941630e-01 -1.20062098e+00 2.32703002e+03]
[1.37673968e+00 7.79759844e-02 -5.42872173e+01]
[1.98417755e-04 -2.74249648e-04 1.00000000e+00]]

Homography Matrix for src_0.jpg dst_1.jpg
[[3.45488480e-01 4.16982644e-03 4.59782936e+02]
[-2.17602511e-01 2.92198177e-01 2.00664455e+02]
[-5.64259651e-05 -1.71364131e-04 1.00000000e+00]]

Homography Matrix for src_1.jpg dst_0.jpg
[[7.84292397e-01 2.00187628e-01 6.79394639e+02]
[4.20863442e-01 1.23376587e+00 1.33159558e+03]
[-7.94313094e-05 2.03188167e-04 1.00000000e+00]]

Homography Matrix for src_1.jpg dst_1.jpg
[[4.52379694e-01 -8.93854911e-02 8.84559291e+02]
[1.46736281e-02 3.93965833e-01 7.54379109e+01]
[1.34278389e-05 -6.97134468e-05 1.00000000e+00]]

Homography Matrix for src_2.jpg dst_0.jpg
[[8.19927361e-01 -1.36280812e-01 1.41908453e+03]
[4.10725717e-02 1.67992565e-01 2.36921943e+03]
[4.29394197e-05 -1.80772576e-04 1.00000000e+00]]

Homography Matrix for src_2.jpg dst_1.jpg
[[2.11957612e-01 -3.92439093e-01 1.48597026e+03]
[3.10456637e-01 2.58301888e-01 8.99926830e+01]
[-6.69603689e-05 -3.47699306e-05 1.00000000e+00]]

3. An qualitative analysis of your test results: how well does the method work? Does it work equally well on the different examples? If not, why might the performance be better in one case then the other? Note that the main goal is to locate the objects in the given images. We could transform the entire object image and overlay on the target image, using the computed homography, but this is not asked for. It will suffice to make your judgment based on results of feature matching (after homography computation).

The algorithm works well to find the object in all cases. Since sufficient number of features get matched to locate the object.

However if we notice how all feature keypoints are being matched. There are more outliers in some combinations of images.

With respect to the orientation of the query image:
In cases where the face of the book in the query image is not fully visible in the train image. i.e the book is slightly moved in the 3D space about x ,y or z axis (roll,pitch,yaw) the feature keypoints get mismatched.

With respect to occlusion in the train image:
In cases where the keypoints are occluded as well we can see that there are mismatches