CENG 391 – Introduction to Image Understanding Homework 4

January 11, 2021

Due Date: January 22, 2021

Download and extract the contents of `ceng391_hw_04.tar.gz`. Submit your solution by e-mail by creating a tar file of the modified contents named as `ceng391_hw04_{STUDENT_ID}.tar.gz`.

Exercise 1 Image Processing and Feature Detection (40 points)

Please do the following exercises by a single Python script named as `src/detect_and_match.py`. You may use OpenCV for feature detection and descriptor computation.

a. Detect SIFT interest points on the six images of the Golden Gate Bridge that are in the folder `data`.

b. Draw the SIFT interest points on each image and store the resulting images in the same folder with names as `sift_keypoints_i.png`, where `i` is the image number.

c. Calculate SIFT descriptor matches between consecutive pairs of images by brute force matching, for example between `goldengate-00.png` and `goldengate-01.png`, between `goldengate-01.png` and `goldengate-02.png`, and so on.

d. Draw these tentative correspondences on a match image and save the resulting images in the same folder with names as `tentative_correspondences_i-j.png`, where `i` and `j` are image numbers.

e. Save the SIFT interest points, descriptors, and tentative correspondences as text files in the same folder with names as `sift_i.txt` and `tentative_correspondences_i-j.txt`.

Exercise 2   RANSAC (40 points)

Please do the following exercises by a single Python script named as `src/ransac.py`. You may use OpenCV for homography computation with RANSAC.

a. Read the keypoints and tentative correspondences for each image pair and match them by RANSAC.

b. You may use RANSAC from OpenCV, implement RANSAC yourself for 10 bonus points.

c. Save the resulting homography matrices in files within the folder `data` with names such as `h_i-j.txt`, where `i` and `j` are image numbers.

d. Do not forget about normalization and the final estimation over all inliers. You may optionally perform guided matching.

e. Draw and save the resulting final inlier correspondences in files in the `data` folder with names as `inliers_i-j.png` and `inliers_i-j.txt`.

Exercise 3   Basic Stitching (20 points)

Please do the following exercises by a single Python script named as `src/stitch.py`. You may use OpenCV function `warp_perspective` for image warping.

a. Stitch all the images by calculating a homography matrix from each image to one of the center images `goldengate-02.png` or `goldengate-03.png` and warping the images to this coordinate system.

b. Save the resulting image in the folder `data` named as `panorama.png`.

c. To blend multiple images just overwrite or average intensities of overlapping pixels.