

Programming Assignment #2

Announcement: 01 February 2022

Submission Deadline: 01 March 2022

Description

The objective of this assignment is to gain an understanding of the methods discussed in the class relating to Hough transform, Harris corner detection, and feature descriptors/matching. You can use C, C++, or Python to implement the solutions.

Part 1: Hough transform (7.5 pts)

- A. Without using OpenCV's hough transform functions, write a program that implements Hough transform.
- B. Using OpenCV's imshow function, display the hough space
- C. Using OpenCV's line function, draw the detected lines on the original image

Part 2: Harris corner detection (7.5 pts)

- A. Without using OpenCV's goodFeaturesToTrack function, write a program that implements Harris corner detection.
- B. Using OpenCV's imshow function, display the results of the gradients (Ix, Iy, Ixy)
- C. Using OpenCV's imshow function, display the results of the corner strength or response (R)
- D. Using OpenCV's drawKeypoints and imshow functions, display the results of the final interest points.

Part 3: Feature descriptors/matching (5 pts)

- A. Without using OpenCV's matcher classes and functions, write a program that calculates SIFT-like descriptors and matches them using the ratio-test.
- B. Using OpenCV's drawKeypoints and imshow functions, display all the keypoints.
- C. Using OpenCV's drawMatches and imshow functions, display the matched keypoints.

Part 4: Pytorch

A. Using the Pytorch API, implement a class Sobel that creates and applies the Sobel operator. Use the Python Jupyter notebook provided [here](#).

Part 5: Optional for undergraduates/Compulsory for graduates (5 pts):

A. Make your feature descriptor contrast invariant.

B. Implement adaptive non-maximum suppression (MOPS paper). Input parameters: $N=500$, and $c=0.9$

Part 6: Extra credit (10 pts)

A. Make your feature detector scale invariant.

B. Implement a method that outperforms the SSD ratio-test for deciding if a feature is a valid match. Your method should outperform SSD on all provided images.

Submission (electronic submission through EAS only)

Please create a zip file containing your C/C++ or Python code (**2 source files** (1: hough, 2: Harris + matching)) and a readme text file (.txt). In the readme file document the features and functionality you have implemented, and anything else you want the grader to know i.e. control keys, keyboard/mouse shortcuts, etc if applicable.

Additional Information

- The source images for Hough transform can be downloaded [here](#).
- The source images for feature extraction and matching can be downloaded [here](#).