Introduction to Computer Vision (ECSE 415)
Assignment 2

Due: October 20^{th}, 11:59PM

Please submit your assignment solutions electronically via the myCourses assignment dropbox. Attempt all parts of this assignment. The assignment will be graded out of total of **84 points**. Students are expected to write their own code. (Academic integrity guidelines can be found at `https://www.mcgill.ca/students/srr/academicrights/integrity`). Assignments received up to 24 hours late will be penalized by 30%. Assignments received more than 24 hours late will not be graded. Note that you can use any of the OpenCV functions shown during tutorial sessions for this assignment, unless stated otherwise. Follow these instructions for the submission:

1. Answer each question in separate Jupyter notebook.
2. Comment your code appropriately.
3. Do not forget to run Markdown cells.
4. Assume image folders are kept in a same directory as the codes. Make sure that the submitted code is running without error. Add a README file if required.
5. If external libraries were used in your code, please specify their names and versions in the README file.
6. Do not submit input/output images.
7. Answers to reasoning questions should be comprehensive but concise.
8. Submissions that do not follow the format will be penalized 10%.

1 Invariance of SIFT Features

You are given a reference image of a book as shown in Figure 1. Verify the invariance of SIFT features under changes in image scale and rotation.

1.1 Invariance Under Changes in Scale

1. Compute **SIFT** keypoints for the reference image. **(2 points)**
2. Scale reference image using scaling factors of (0.2, 0.5, 0.8, 1.25, 2, 5). **(2 points)**
3. Compute **SIFT** keypoints for the transformed images. **(2 points)**
4. Match all keypoints of the reference image to the transformed images using a brute-force method. **(2 points)**
5. Sort matching keypoints according to the matching distance. **(2 points)**
6. Display top ten matched keypoints for each pair of reference image and a transformed image. **(2 points)**
7. Plot the matching distance for top 100 matched keypoints. Plot indices of keypoints on x-axis and corresponding matching distance on y-axis. **(4 points)**
8. Discuss the trend in the plotted results. What is the effect of increasing the scale on the matching distance? Reason the cause. **(3 points)**

1.2 Invariance under Rotation

1. Compute **SIFT** keypoints for the reference image. **(2 points)**
2. Rotate reference image at the angle of (10,30,90,150,170,180) degrees. **(2 points)**

Figure 1: Reference image of a book.

3. Compute **SIFT** keypoints for the transformed images. **(2 points)**

4. Match all keypoints of the reference image to the transformed images using a brute-force method. **(2 points)**

5. Sort matching keypoints according to the matching distance. **(2 points)**

6. Display top ten matching keypoints for each pair of reference image and a transformed image. **(2 points)**

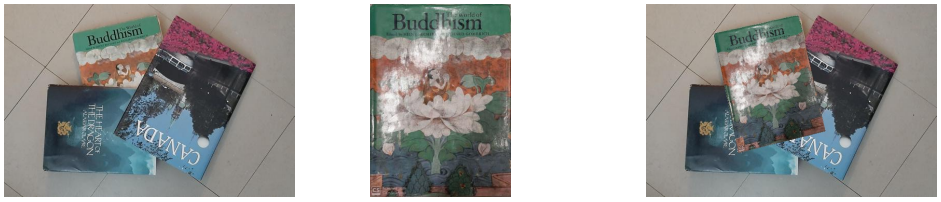
7. Plot the matching distance for top 100 matched keypoints. Plot indices of keypoints on x-axis and corresponding matching distance on y-axis. **(4 points)**

8. Discuss the trend in the plotted results. What is the effect of increasing the angle of rotation on the matching distance? Reason the cause.**(3 points)**

2 Matching using SIFT - Book Reveal

You are given an image of the reference book taken under different acquisition conditions: (a) under occlusions (Figure 2(a)) and (2) under different lighting conditions (Figure 2(b)). The task is to transform the image of the book in Figure 2(a) and align and merge it with the occluded view in (Figure 2(b) to generate an image with an unoccluded view of the book (See Figure 2(c)). To achieve this objective, please perform following steps:

- Find SIFT keypoints in given input images. **(2 points)**
- Match keypoints of reference image to the keypoints of the occluded image using brute-force method. **(2 points)**
- Sort matching keypoints according to the matching distance. **(2 points)**
- Display top ten matching keypoints. **(2 points)**



(a) (b) (c)

Figure 2: Input and desired output for image manipulation task (a) image of an occluded book (book_occlusion.jpg) (b) reference image of a book in different lighting condition (book_crop.jpg) (c) desired output.

- Compute a homography to align the images using RANSAC method and apply the transformation on the reference image. **(6 points)**
- Paste transformed reference image on the occluded view to generate unoccluded view as shown in Figure 2(c). **(2 points)**

3 Object Detection

In this question you will learn to detect cars in a given scene using Histogram of Gradient (HoG) features. You are given a small training set of 14 images of cars and a test image containing 3 cars (see Figure 3). Use following instructions to perform the detection task.

- Training
  1. Resize the training images to 128×128. **(1 points)**
  2. Compute HoG features using cell size of 4×4 pixels, block size of 2×2 cells and 9 orientation bins **(5 points)**
     (Suggestion: Make a function which takes list of images as arguments and delivers list of HoG features as output. The same function can be used during testing.)
  3. Calculate and store the mean feature map across training images. **(1 points)**
  4. Repeat above steps for training images flipped around vertical axis. **(2 points)**
  5. Display 9 orientation channels of the mean feature maps for the first block. **(3 points)**
- Testing
  1. Extract overlapping windows from the test image. Details on the size of the window and the stride are given below. **(3 points)**
  2. Resize windows to 128×128 and Compute HoG features similar to what was done during training. **(5 points)**

Figure 3: Test image for car detection

3. Compute the euclidean distance between the feature map of each window and the mean feature map of training images. Repeat the same with the mean feature map of flipped training images. **(2 points)**
4. Threshold the two distances to detect cars in the test image. Display detected window. **(3 points)**
5. Experiment with size of the window, stride and the detection threshold to achieve accuracy. **(5 points)**