Introduction to Computer Vision (ECSE 415)
Assignment 3

Due: November 5^{th}, 11:59PM

Please submit your assignment solutions electronically via the mycourses assignment dropbox. 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.

There are five questions in this assignment. The questions permit you to explore segmentation and stereo vision algorithms. Attempt all parts of this assignment. The assignment will be graded out of total of **75 points**. The first question requires that you to implement several algorithms from scratch. You can use library-implemented functions for the rest of the assignment. Segmentation part has optional bonus question. If you score below 17.5 points in question 1.1, only then question 1.2 will be graded and the final grade for segmentation question would be max{grade for Q-1.1, grade for Q-1.2}.

1 Segmentation

Use image 'home.jpg' for all three questions (Figure 1).

Figure 1: Image be used for all questions.

1.1 K-means clustering and Expectation Maximization (35 points)

- Implement K-means clustering algorithm using only the numpy library. You can use opencv and matplotlib libraries only to read and display images but not for clustering. Apply k-means to the provided image with K=2 and K=3. Display the resulting segmented images for 10 iterations in both cases. (10 points)
- Implement the EM algorithm using only the numpy library. You can use opencv and matplotlib libraries only to read and display images. Apply it to the given image with 2 and 3 Gaussian components. Display the resulting segmented images for 10 iterations in both cases. (20 points)
- Under what data distribution conditions would K-means and EM give the same solution, provided similar initializations? (5 points)

1.2 Normalized graph-cut and Mean-Shift segmentation (BONUS 20 points)

You can use functions from skimage libraries for this question.

- Segment the given image using normalized graph-cuts. Vary the following parameters (try several values of each parameter): compactness and n_segments (*slic* function), thresh (*cut_normalized* function). Display segmentation results for several parameters and state their effect on the output. (10 points)
- Segment the given image using mean-shift. Vary the following parameters (try several values of each parameter): ratio, kernel_size, max_dist. Display segmentation results for several parameters and state their effect on the output. (10 points)

2 Stereo Vision

Use a stereo image-pair of your choice from the Middlebury dataset: `http://vision.middlebury.edu/stereo/data/` for the following questions. The dataset provides depth images in .pfm format. A function to read such file in provided along with this assignment.

2.1 Depth Estimation from Stereo Images (13 points)

- Estimate depth using the stereo image pair. Show the results of varying the following parameters: blockSize $\in [5, 10, 15]$, numDisparities $\in [16, 32, 48]$. You will have a total of 9 depth-maps, each corresponding to one combination of blocksize and numDisparities. (5 points)

- For each depth-map, compute the pixel-wise squared difference error with respect to the ground-truth. (2 points)
- Comment on the error-maps for each combination of parameters. Specifically, mention the effects of varying the parameters on the error in the region of high/low texture, edge, large/small depths. (6 points)

2.2 Epipolar Geometry (27 points)

- Compute matching SIFT keypoints from a stereo image pair. (5 points)
- Compute and display the epipolar lines for both images. (5 points)
- Pick any one keypoint in the right image which doesn't have a correct match in the left image, and is not on the corresponding epipolar line. Extract a patch of size (5×5) around this keypoint in the right image. (2 points)
- Match the extracted patch using the sum of squared difference (SSD) with every patch along the corresponding epipolar line in the left image. (7 points)
- Plot SSD values (refer Lecture 14 slide 5). Find the matching point with minimum SSD value. Display found matching points in both the images. (4 points)
- Did you find exactly one matching point with lowest SSD or multiple? Is the found matching point correct one? Explain why. (2 points)
- Suppose you found multiple matches in the previous question. What constraint(s) could be imposed on the matching algorithm to find a single match? (1 point)
- Suppose you found exactly one match with the lowest SSD, but it is incorrect one. What constraint(s) could be imposed to find a correct match? (1 point)

3 Submission Instructions

1. Title four jupyter notebooks as (i) Kmeans_EM (ii) GraphCut_MeanShift (iii) Depth_Estimation (iv) Epipolar_Geometry.
2. Comment your code appropriately.
3. Do not forget to run Markdown cells.
4. Do not submit 'home.jpg'. Assume 'home.jpg' is kept in a same directory as the codes.
5. **Submit stereo image-pair.**

6. Make sure that the submitted code is running without error. Add a README file if required.

7. If external libraries were used in your code please specify its name and version in the README file.

8. Answers to reasoning questions should be comprehensive but concise.

9. Submissions that do not follow the format will be penalized 10%.