CS 558: Homework Assignment 3 - Image Segmentation
Due: April 2, 11:59pm

Philippos Mordohai
Department of Computer Science
Stevens Institute of Technology
Philippos.Mordohai@stevens.edu

Collaboration Policy. Homeworks will be done individually: each student must hand in their own answers. It is acceptable for students to collaborate in understanding the material but not in solving the problems. Use of the Internet is allowed, but should not include searching for previous solutions or answers to the specific questions of the assignment. I will assume that, as participants in a graduate course, you will be taking the responsibility of making sure that you personally understand the solution to any work arising from collaboration.

Late Policy. No late submissions will be allowed without consent from the instructor. If urgent or unusual circumstances prohibit you from submitting a homework assignment in time, please e-mail me explaining the situation.

Submission Format. Electronic submission on Canvas is mandatory. Submit a zip file containing:

- a pdf file with the source code (excluding libraries), the resulting images and a brief explanation of the implementation.
- the code,
- the output images.

Problem 1: k-means Segmentation. (35 points) Apply k-means segmentation on white-tower.png with k=10, according to slide 11 of Week 7. The distance function should only consider the RGB color channels and ignore pixel coordinates. Randomly pick 10 RGB triplets from the existing pixels as initial seeds and run to convergence.

After k-means has converged, represent each cluster with the average RGB value of its members, creating an image as in slide 19 of Week 7.

Problem 2: SLIC. (65 points) Apply the SLIC algorithm to wt_slic.png. See slide 47 in Week 7, but implement the following steps:

1. Divide the image in blocks of 50×50 pixels and initialize a centroid at the center of each block.

2. Compute the magnitude of the gradient in each of the RGB channels and use the square root of the sum of squares of the three magnitudes as the combined gradient magnitude. Move the centroids to the position with the smallest gradient magnitude in 3×3 windows centered on the initial centroids.

3. Apply k-means in the 5D space of x, y, R, G, B. Use the Euclidean distance in this space, but divide x and y by 2.

4. Optionally: only compare pixels to centroids within a distance of 100 pixels (twice the block size) during the updates.

5. After convergence, display the output image as in slide 47: color pixels that touch two different clusters black and the remaining pixels by the average RGB value of their cluster.

Requirements and notes.

- You can use any programming language. You may have to explain the homework to me in person, if I am not familiar with your choice.
- You are allowed to use image reading and writing functions, as well as plotting functions, but you are not allowed to use filtering, edge detection or other image processing functions. You can convert the images to a different format for reading them.
- The complexity of k-means is linear with respect to the number of pixels and k. Start developing on small images with small values of k.