CS270 Homework 3

Due Date: 11:59 pm, May 21

Notes:

1. Discussions are encouraged while plagiarism is strictly prohibited, source code should not be shared in any form.
2. The total score of the homework is 100.
3. The example results are just for reference. Please try to achieve the best performance as you can.
4. Please send your homework zip file (report in .pdf format, code and supplement materials) to `dip_2021@163.com`. The **filename** is supposed to be `hw3_[Chinese_name]_[student_ID].zip`, for example, `hw3_马保国_hw3.zip`. And the **topic of your email** should be the same as the filename of your zip, with **suffix removed**.
5. Please provide the SHA1 code in the body of your email. Submissions without SHA1 code provided may be considered as INVALID submissions!
6. Please place the files properly. The illustration of file structure can be referred in Piazza: https://piazza.com/class/klakhm7kxf2i9?cid=92.
7. All source images are placed in `images` folder.

Question 1: Graph Cut for Image Segmentation (50 points)

Graph cut is a popular energy optimization technique for interactive image segmentation, which is widely used in foreground and background segmentation, stereo vision, image matting and medical image lesion segmentation, etc.

In this question, you are required to implement the graph cut algorithm using any programming language you like. First, you need to segment the foreground and background of the provided image using your algorithm. And then you are required to implement multi-class segmentation by your graph-cut algorithm.

Task:

1. Please describe your algorithms in words or flowcharts. (15')
2. Implement your foreground and background segmentation algorithm via graph cut methods.
   a) As shown in figure 1 you need to perform graph cut method to segment the foreground and background of the given image `q1_1.jpeg`. (15')

Figure 1

   b) Modify your graph cut program for multi-class segmentation. You need to segment the given image `q1_2.jpeg` to four parts, and show the segmentation results via transparent painting overlayed with the origin image as shown in Figure 2. (20')

Figure 2

Question 2: Canal classification (50 points)

File `q2.jpeg` shows a cross-section of human cortical bone imaged by synchrotron radiation microcomputed tomography. The black areas are the canals hosting bone vasculature. In this question, it is required to classify the canals based on their sizes.

Requirements:

- The result of classification is supposed to contain 3 images, which are pure background (the gray color component occupying most areas in the image), background with large canals and background with small canals. We mark these as I_b, I_l and I_s.
- The criteria discriminating big and small canals are not restricted. You can define it by yourself. Please ensure that the metric does not result in mixed classification results, i.e., the smallest canal in the large canal category is expected to be larger than the largest canal in the small canal category.
- All information in the image should be preserved during processing. The information distortion map can be calculated by $I_\epsilon = I_l + I_s - I_b - I_0$ where I_0 is the original image, and the extent of distortion is measured by the element-wised summation of I_ϵ, which is $\sum_{i \in I_\epsilon} I_\epsilon(i)$. The lower the value, the better the information is preserved during the processing.
- Please clearly illustrate the process you designed and describe the programming details with flowcharts or your own language in your report. Meanwhile, the **filenames and variable names** of the results ($I_b, I_l, I_s, \sum_{i \in I_\epsilon} I_\epsilon(i)$) must be specified as well.
- **Important**: Usage of built-in functions is restricted in your implementation. In your implementation, **only basic bulit-in functions** including I/O functions, plot functions, OS functions and basic mathematical / morphological operations are allowed to use for necessary processing, key steps in the process must be implemented by your own code instead of using bulit-in functions. Your score will be deducted if your implementation **contains built-in functions** which are not allowed to use (other than the basic ones), depending on how many are used. In the worst case, if the process is totally implemented with built-in functions, you may receive only 40% of your original score (for example, if your total score is 40, then your final score is $40 \times 40\% = 16$.

Checkpoints

Your report should contain the following information:

- Overall illustration of the process you designed. (5')
- Your metric of dividing canals into large and small canals. You may describe it in mathematical and natural languages. (10')

- Classification results, I_b, I_l and I_s. (24', 7' for each)
- Result of element-wised summation of I_ϵ, i.e., $\sum_{i \in I_\epsilon} I_\epsilon(i)$. (6')
- Code and result illustrations, including filenames and variable names of the results above. (5')