Answers to questions in
Lab 3: Image segmentation

Name: __Mert Alp Taytak__________________________________ Program: __EECS_

Instructions: Complete the lab according to the instructions in the notes and respond to the questions stated below. Keep the answers short and focus on what is essential. Illustrate with figures only when explicitly requested.

Good luck!

Question 1: How did you initialize the clustering process and why do you believe this was a good method of doing it?

Answers:

I created a random permutation of indices without replacement. Then, I used those indices to select pixels from the image. Those pixels then became initial cluster centers.

I would think that there are other ways to achieve a more diverse segmentation. However, from my experiments this method yields very good results. Moreover, it is simple to implement.

Question 2: How many iterations L do you typically need to reach convergence, that is the point where no additional iterations will affect the end results?

Answers:

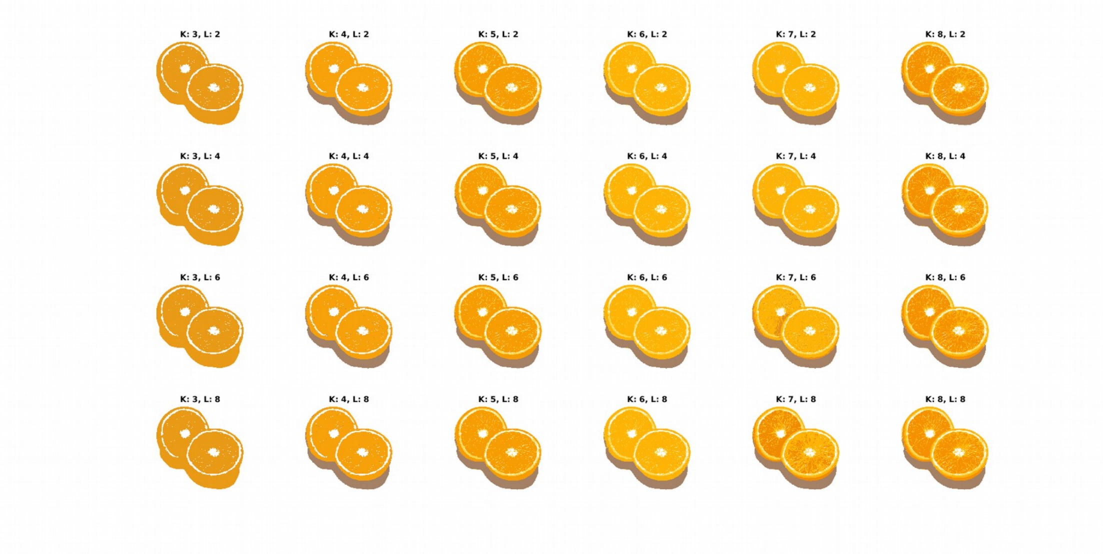
In my current implementation a single iteration does most of the work. Such that after the first iteration changes are almost mostly imperceptible. I have observed in my experiments that playing around with the seed changes the situation. However, a single iteration is enough with the default seed.

Please refer to the figure in question 3 for a visual. Also, I have noticed that for higher K values, smaller L values may lead to a perceptible change. However, I cannot do extensive tests due to the time cost of calculating over a large range of values.

Question 3: What is the minimum value for K that you can use and still get no superpixel that covers parts from both halves of the orange? Illustrate with a figure.

Answers:

Following figure shows results using varying K and L values:

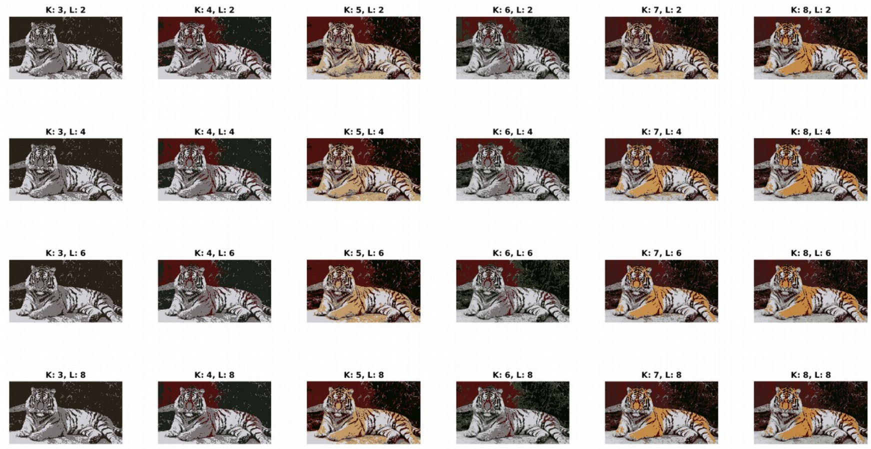


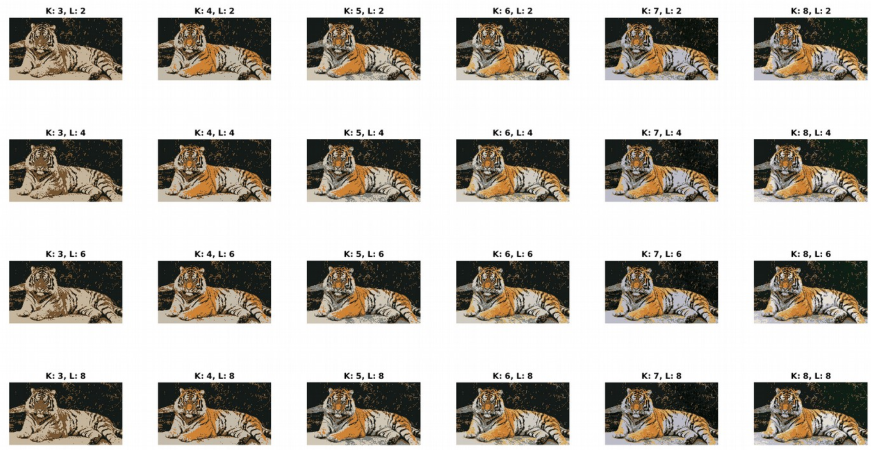
Depending on the L value K = 7 or K = 8 is the minimum value where two oranges do not share superpixels.

Question 4: What needs to be changed in the parameters to get suitable superpixels for the tiger images as well?

Answers:

Initial cluster centers need to be chosen well in order to get a more varied segmentation. In my implementation this can be done by changing the seed. Compare the following figures using the default seed and the number 161 as the seed respectively:





In the bottom picture features of the tiger are better segmented.

Question 5: How do the results change depending on the bandwidths? What settings did you prefer for the different images? Illustrate with an example image with the parameter that you think are suitable for that image.

Answers:

Question 6: What kind of similarities and differences do you see between K-means and mean-shift segmentation?

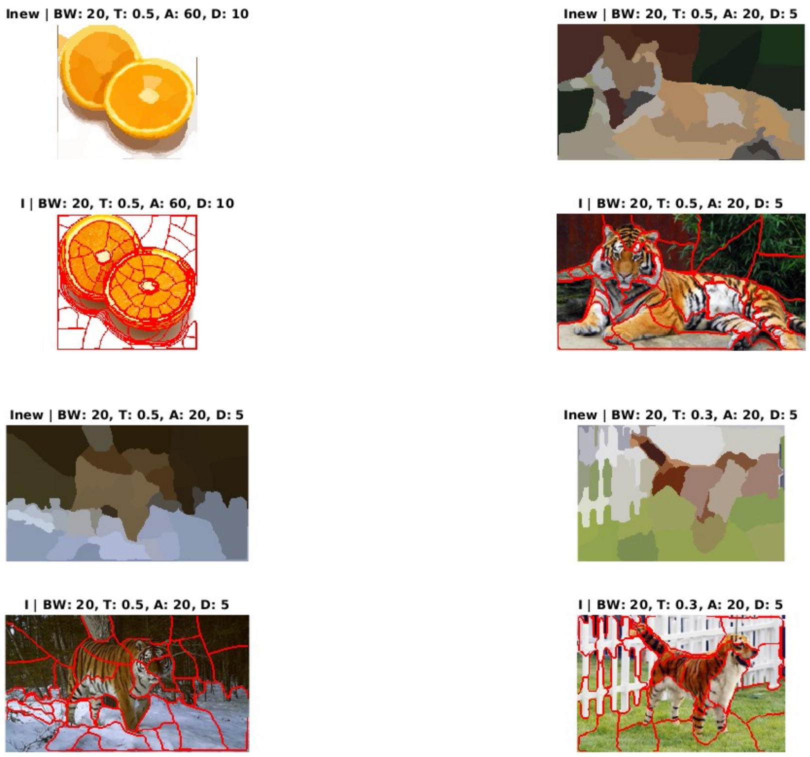
Answers:

Question 7: Does the ideal parameter setting vary depending on the images? If you look at the images, can you see a reason why the ideal settings might differ? Illustrate with an example image with the parameters you prefer for that image.

Answers:

Yes, it does. Normalized cut tries to separate an image into homogeneous regions of similar size, however some pictures are made up of features that do not allow for that well. For example, picture of the orange consists of large blocks of similar pixels. On the other hand, picture of the tiger has large blocks that make up the fur broken up by the stripes.

In the following pictures, BW stands for color_bandwidth, T stands for ncuts_threshold, A stands for min_area and D stands for max_depth.



Question 8: Which parameter(s) was most effective for reducing the subdivision and still result in a satisfactory segmentation?

Answers:

- max_depth is an obvious candidate. It directly controls the depth of recursion.
- min_area allows smaller features to be represented, hence it is effective for controlling divisions.

Question 9: Why does Normalized Cut prefer cuts of approximately equal size? Does this happen in practice?

Answers:

The method tries to minimize the following value:

$$Ncut(A,b) = \frac{cut(A,B)}{assoc(A,V)} + \frac{cut(A,B)}{assoc(B,V)}$$

To minimize this we need to minimize the values of cuts and maximize the values of assocs. Normally, cuts of equal size is the best way to achieve this.

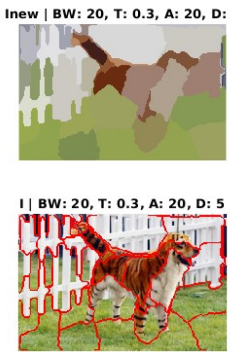
In practice, regions cannot be always evenly divided into two. Features of the image would result in a cut that reaches optimality through similarity of the regions over sizes of the regions.

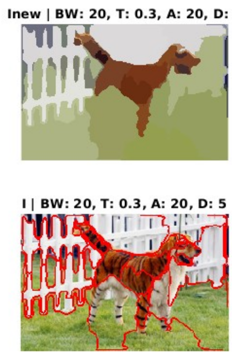
Question 10: Did you manage to increase *radius* and how did it affect the results?

Answers:

I did. First of all, it increases computation time. But, it allows a wider area to be searched at each recursion level. Hence it allows areas broken up with smaller features such as a tiger's stripes to be merged into a single region by looking through a larger area.

Compare the following, left-side picture has radius = 5, right-side has radius = 10:





Most obvious change is that fences are better segmented with the increased radius. Also, tiger's fur is segmented in a way it was not in the left-side picture. That is the effect of better being able to ignore the stripes.

Question 11: Does the ideal choice of *alpha* and *sigma* vary a lot between different images? Illustrate with an example image with the parameters you prefer.

Answers:

Question 12: How much can you lower K until the results get considerably worse?

Answers:

Question 13: Unlike the earlier method Graph Cut segmentation relies on some input from a user for defining a rectangle. Is the benefit you get of this worth the effort? Motivate!

Answers:

Question 14: What are the key differences and similarities between the segmentation methods (K-means, Mean-shift, Normalized Cut and energy-based segmentation with Graph Cuts) in this lab? Think carefully!!

Answers:
