Answers to questions in Lab 3: Image segmentation

Name: Serkan Arda Yılal
Program: Msc. Information and Network Engineering (TINNM),

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:

For each cluster, we have initialized centers by randomizing RGB values of related centers in interval of [0,255]. By doing this, we became able to utilize the whole range of values and have a good spread of cluster centers.

Another good approach would be also to select most commonly used color values in the beginning and adding randomized RGB centers for remaining cluster centers.

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:

Convergence in the segmentation depends on more than one factor when we consider the different applications we are using on our image. As we increase the number of clusters, it will take much more time for convergence due to the increase in the variety of segments. Also, if we have a complex image which means there is a variety in pixel RGB colors, it will also take more iterations to reach to the convergence level. The minimum L values for convergence in different number of clusters and images are as in the following:

K	Min L for Convergence in Orange	Min L for Convergence in Tiger1
2	6	10
6	16	26

After these iteration values for corresponding number of clusters, no big difference occurred in the segmentation.

Also, one should consider that since we are using gaussian filter before segmentation, the blurring effect of it on the image is going to decrease the variety in the pixel values of the image which will result with a lower value requirement for number of iterations for a convergence.

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:
In order to analyze and find the minimum value for K, we will need to have enough iteration so that the results of segmentation will converge regardless of defined K value. When we loop over different K values we can see that after around K = 7, there is no clear superpixel that covers both orange halves and the oranges are segmented properly enough to be observed as different halves.

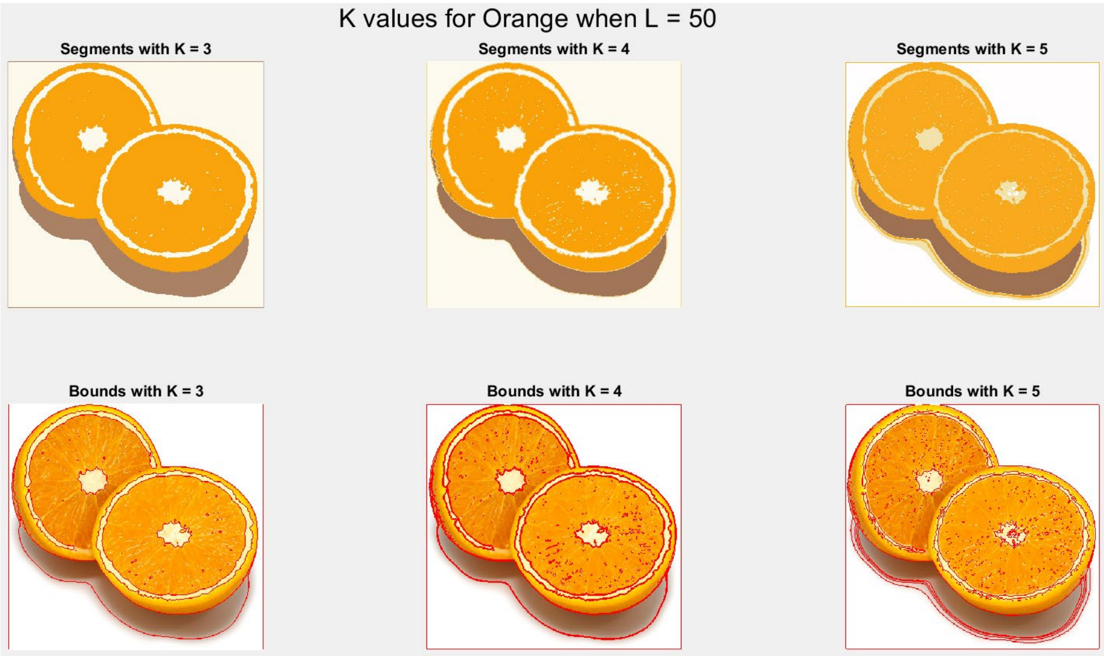

Figure 3.1-Segmentations with different number of Clusters (K = 3, 4, 5)

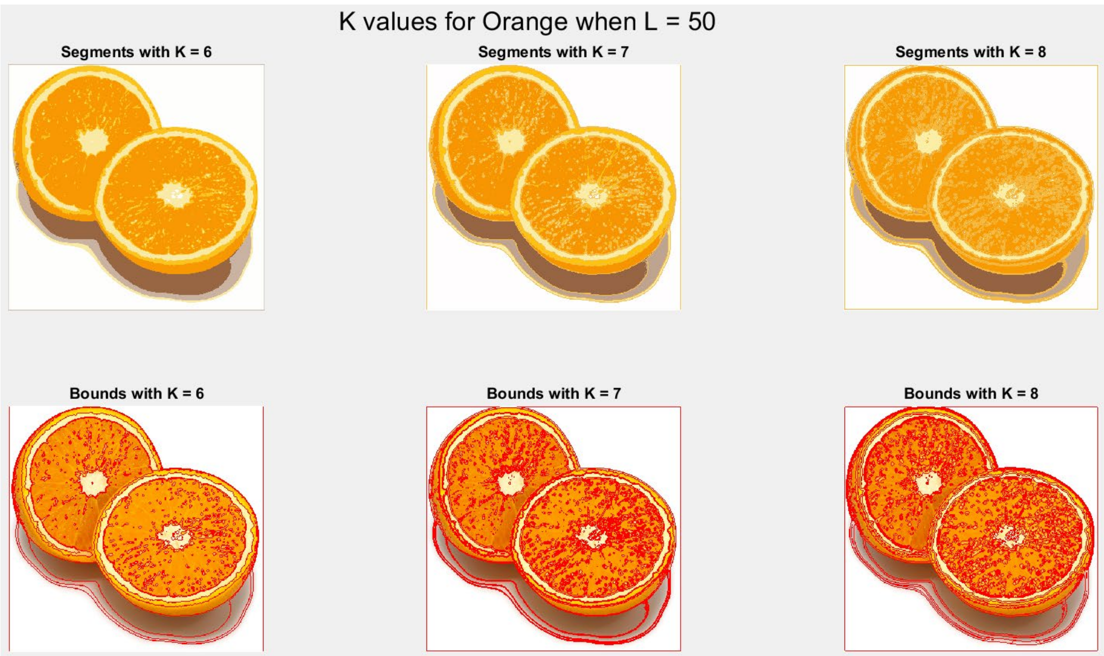

Figure 3.2-Segmentations with different number of Clusters (K = 6, 7, 8)

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

Answers:

As we observe, tiger images are more complex due to the higher variety of pixels and their distribution and it requires us to use higher number of clusters and iteration. As we increase those parameters, we see the improvement in the segmentation of the tiger image. On the other hand, when we increase smoothing constant sigma for gaussian filter we start to lose details of the image and as a result segmentation also loses its quality. For that reason, a proper value of sigma should be chosen before filtering the image prior to the segmentation process.

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:

For Spatial-Bandwidth: As we increase the Spatial Bandwidth, we see that there is a decrease in the number of modes. The reason is that as the bandwidth increases, the gaussian distribution becomes more spread and spatially it leads more pixels to correspond to a particular mode. As a result, for low spatial-bandwidths we have more peaky points and for high spatial-bandwidths we have more spread-out modes.

For Color-Bandwidth: For low values of color-bandwidth we see that there are distinct segments that do not unify easily in terms of colors. On the other hand, for high values of color-bandwidth similar colored pixels are more tend to unify with each other. So, in a way, increasing the color-bandwidth will lead increasing the radius and cover more area in color space.

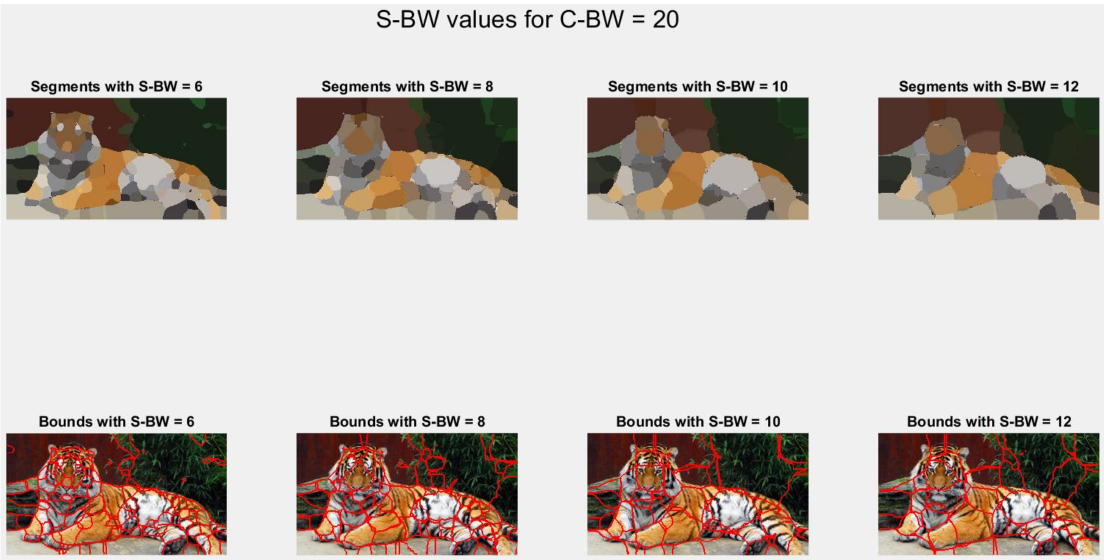

Figure 5.1-Spatial-bandwidth value comparisons for constant color-bandwidth in tiger1

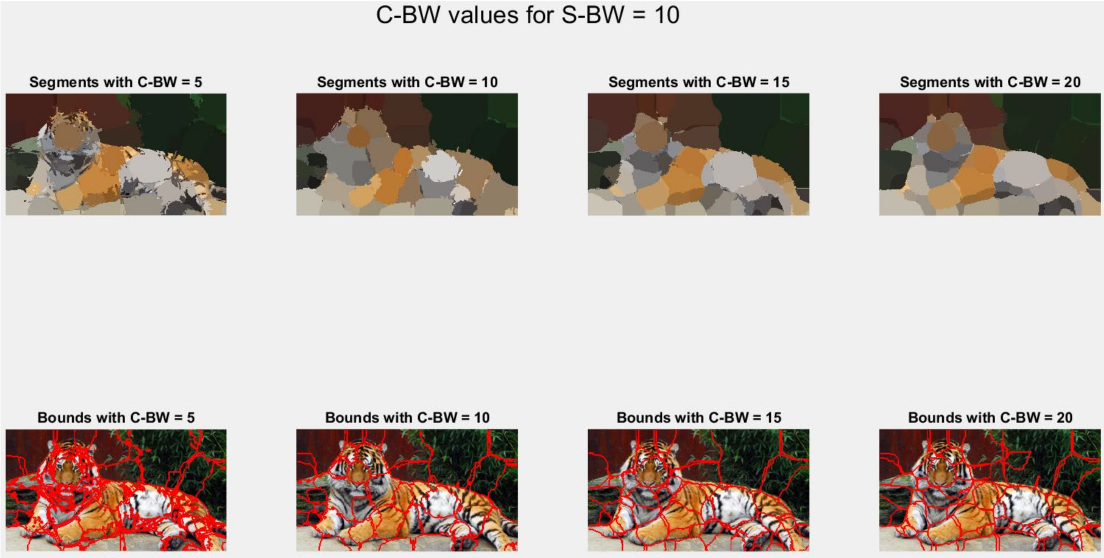

Figure 5.2-Color-bandwidth value comparisons for constant spatial-bandwidth in tiger1

The images with suitable bandwidth values are as in the following:

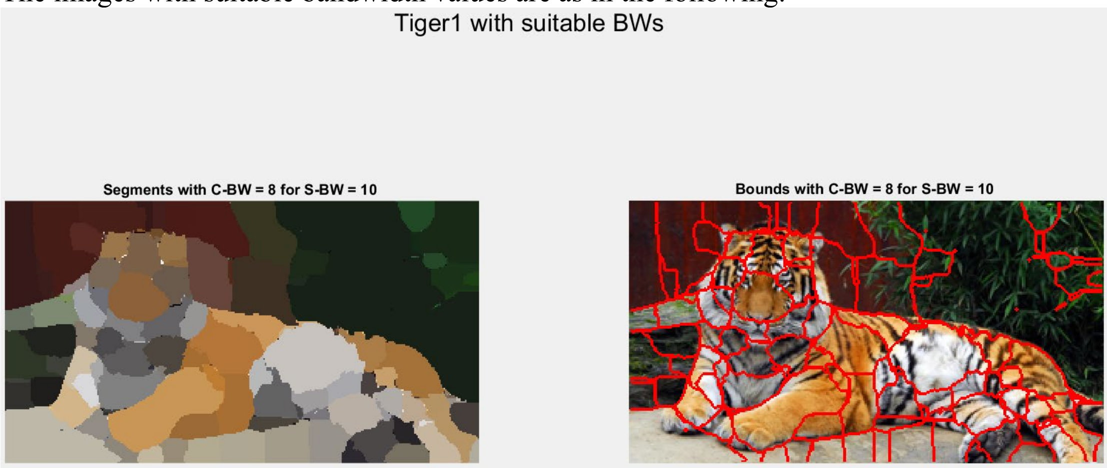

Figure 5.3-Ideal Mean-Shift Segmentation bandwidths for tiger1

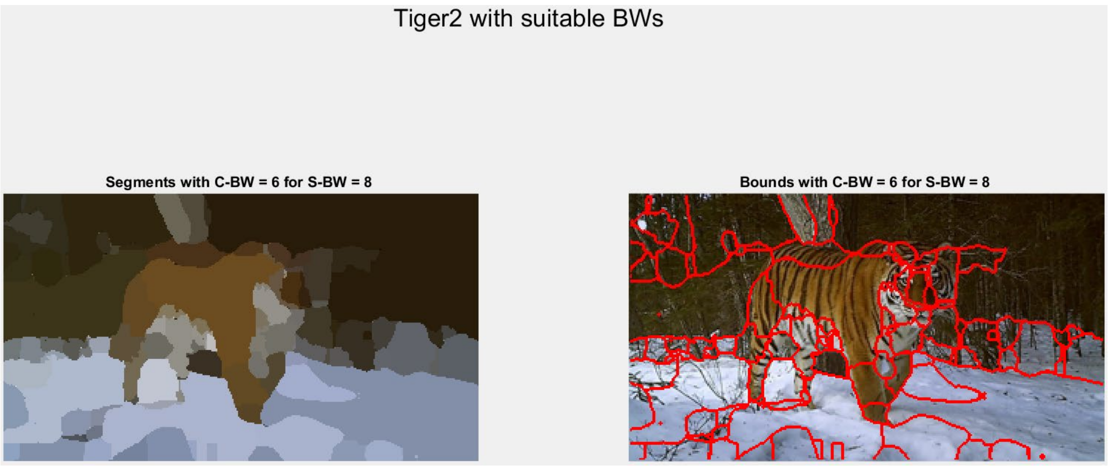

Figure 5.4-Ideal Mean-Shift Segmentation bandwidths for tiger2

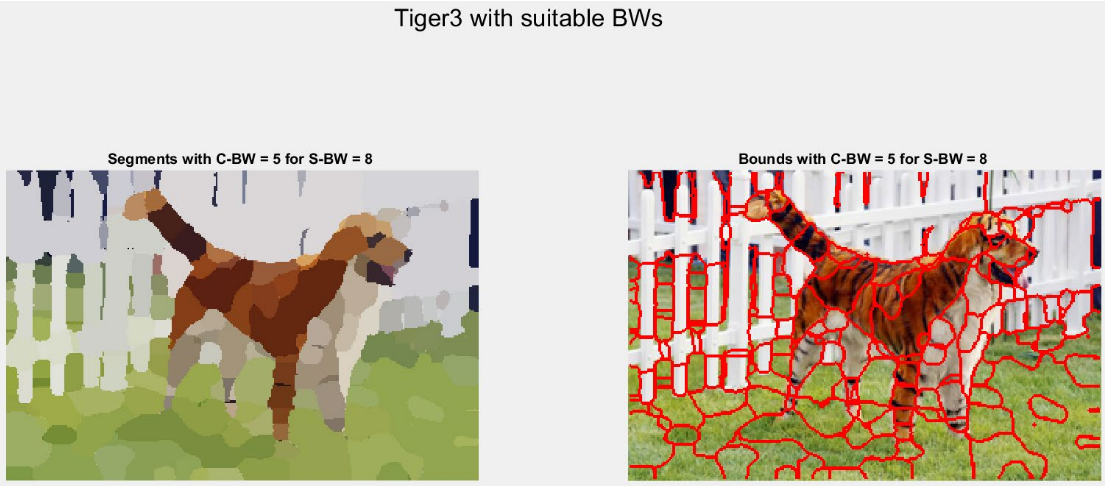

Figure 5.5-Ideal Mean-Shift Segmentation bandwidths for tiger3

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

Answers:

Similarities:
Both methods are using cluster centers as segmentation and they are run iteratively.

- For K-means, we are updating cluster centers in terms of mean of colors that are assigned to each of those clusters.
- For Mean-shift segmentation, we are updating cluster centers in terms of spatial position and searching for maximum local density.

Differences:

- K-means considers only the color values while Mean-shift considers additionally the spatial information.
- The number of clusters are defined at the beginning for K-means, while for the mean-shift it is defined during the segmentation process according to pre-defined spatial and color bandwidths.

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:

The ideal parameter settings definitely vary depending on the chosen image. We see that max_depth parameter is really important especially in complex images. If the define it big enough, then it will be possible to segment complex structure of the image. Otherwise, a low valued max_depth will not give satisfactory results for that complex image.

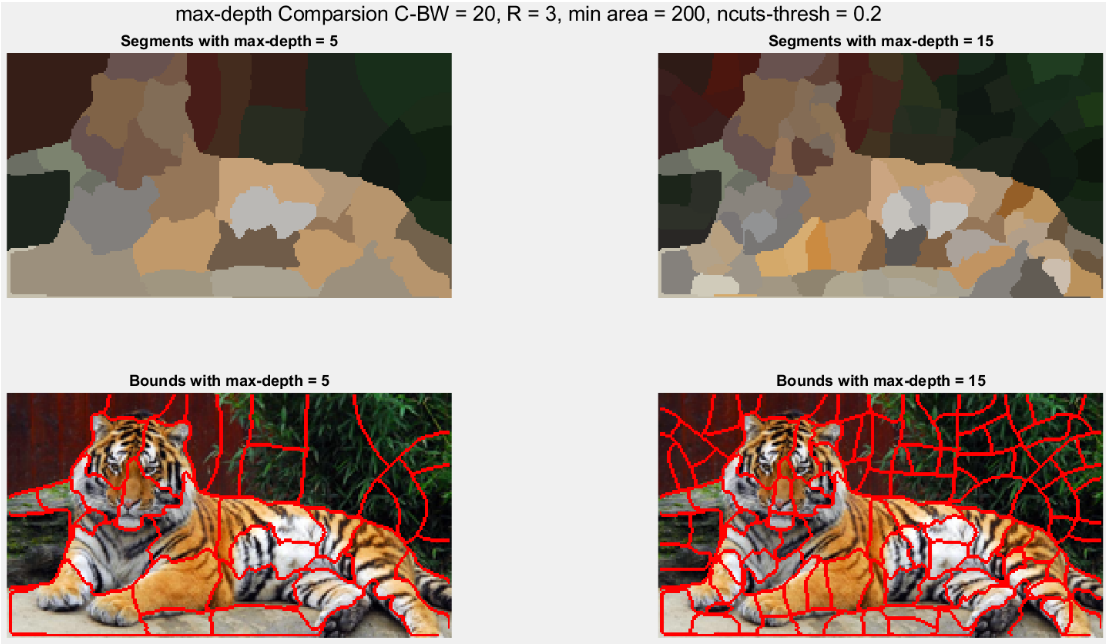

Figure 7.1-Max-depth comparison for tiger1

Also considering ncut_thresh allows more similar areas to be cut, increasing this parameter will also help in complex images in order to segment it enough.

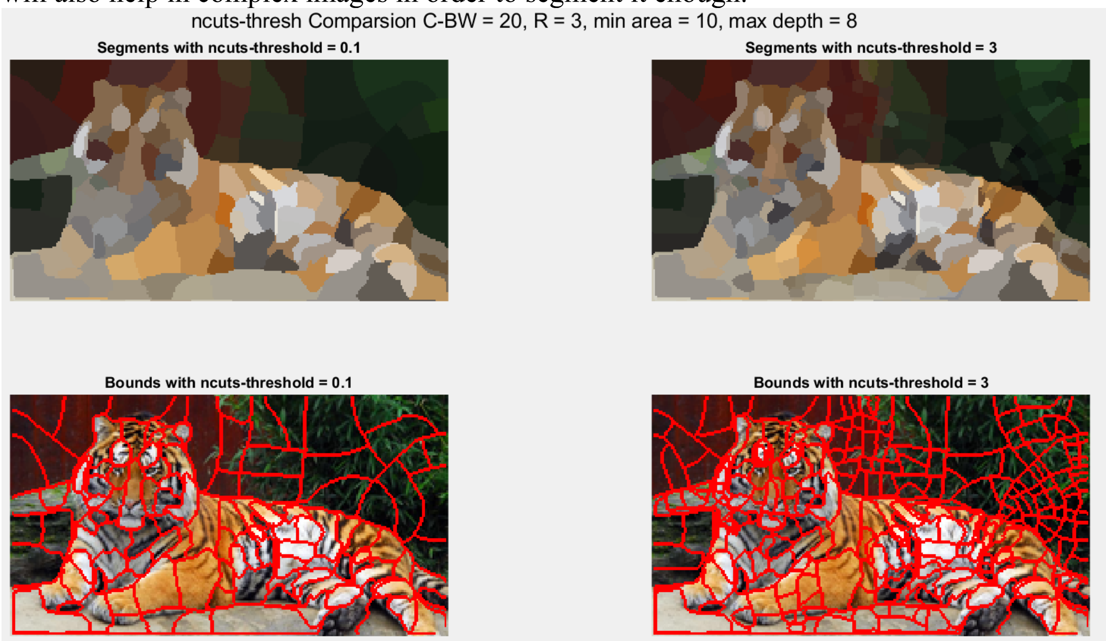

Figure 7.2-ncuts_tresh comparison for tiger1

For color bandwidth, we can say that it affects the weightage of similar and unsimilar pixels. When we increase the bandwidth, we see that unsimilar pixels are weighted more and similar pixels are weighted less and the opposite is valid for the case when we decrease the bandwidth. This results with bigger segments for unsimilar pixels and smaller segments for similar pixels as we increase the bandwidth.

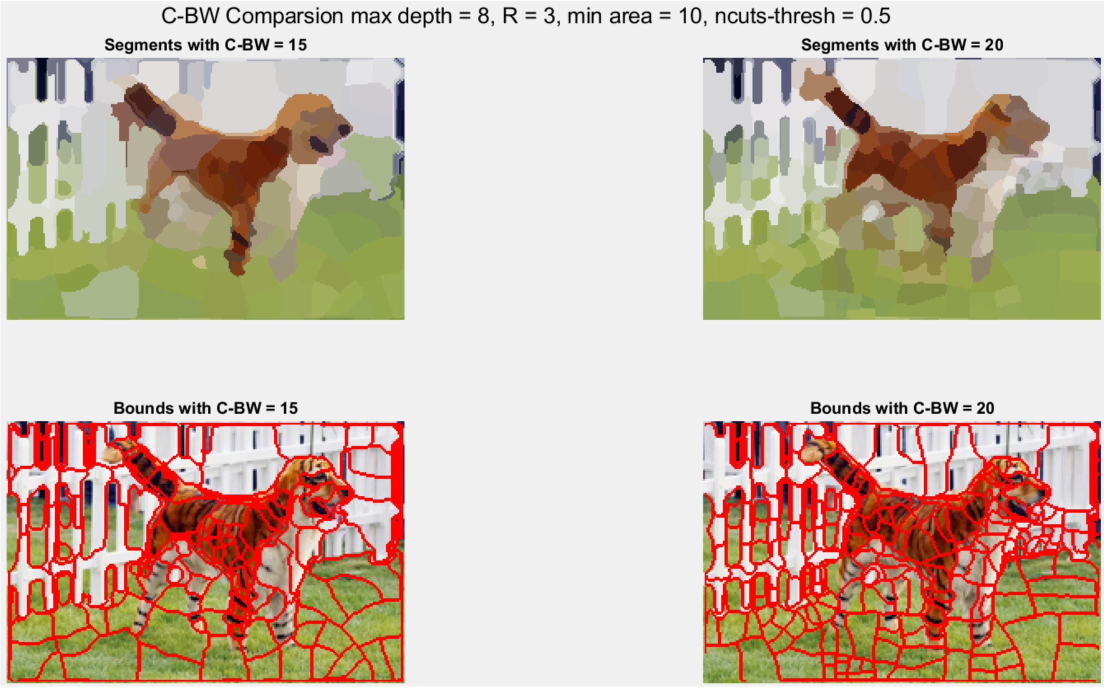

Figure 7.3-Color Bandwidth comparison for tiger3

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

Answers:

The main parameters that affected the reduction of the subdivision the most were max_depth, ncut_thresh, and min_area. Decreasing max_depth and ncut_thresh decreased the subdivision while increasing min_area had a limiting affect on it and prevented it from more subdivision.

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

Answers:

We can show it by the general edge equation as in the following:

$$assoc(V) = assoc(A,V) + assoc(B,V) - cut(A,B)$$

In addition to that equation, we are also provided with the following equation in the lab and the lecture notes as well:

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

If we take the assoc(B,V) from the first equation and replace it with the remaining part of that equation, we will have:

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

In order to minimize this expression, we can take its derivative and make it equal to zero.

$$\frac{Ncut(A,B)}{d\,assoc(A,V)} = 0$$

With the help of Wolfram Alpha, we see that our root corresponds to:

$$assoc(A,V) = \frac{assoc(V) + cut(A,B)}{2}$$

If we insert it to our very first equation, we come up with following result:

$$\frac{1}{2}assoc(A,V) = \frac{1}{2}assoc(B,V) => assoc(A,V) = assoc(B,V)$$

From that result, we can see the reasoning behind preference of equal size cuts. In practical considering we are not just using maximum depth as a parameter but also using other parameters too, the balance between cuts may change.

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

Answers:

As we increase the radius, it definitely increases the computation time as well. For example, while radius = 2 takes around 3.5 seconds, radius = 10 takes around 27 seconds. It is because it considers much more neighbor pixels during calculation. Also, we can see that segments are getting bigger as we increase the radius as in the following:

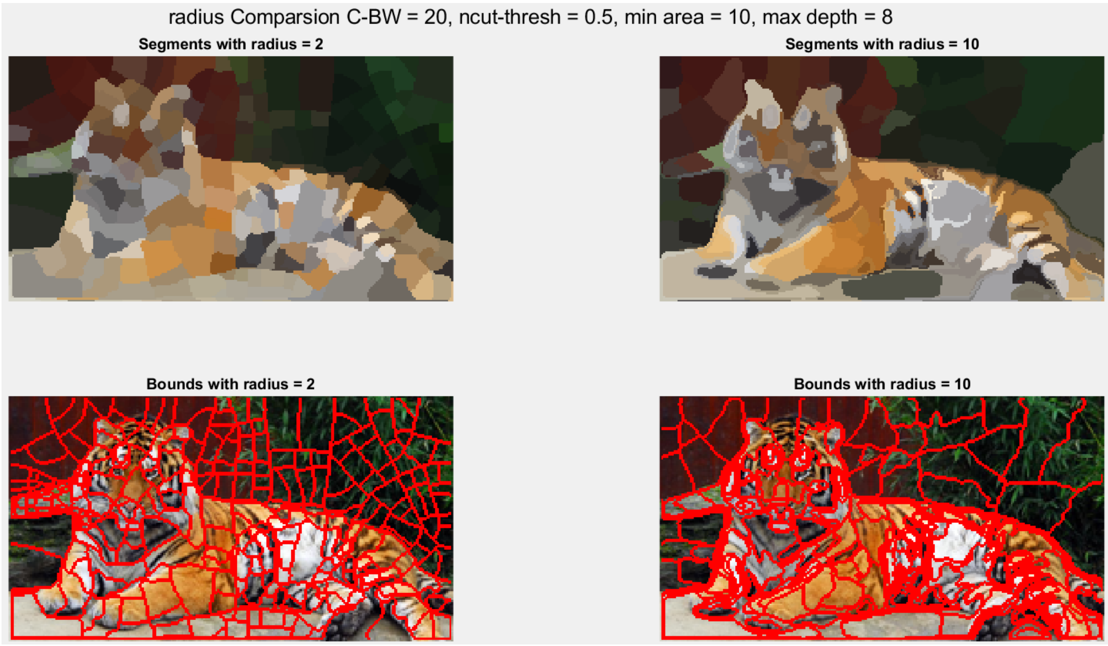

Figure 10.1-Radius comparison for tiger1

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:

As we increase the alpha in our application, it becomes even much harder to cut through similar edges considering it increases maximum cost of an edge.

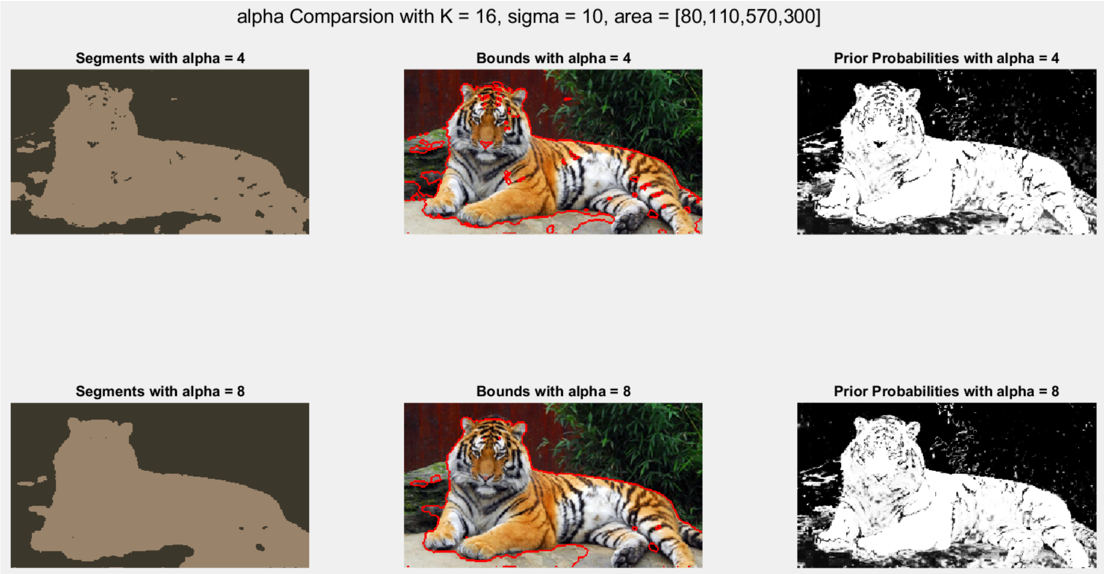

Figure 11.1-alpha value comparison for tiger1

On the other hand, we know that sigma defines the decaying speed of edge cost as the similarity decreases. In that case since the cost of the edges will decrease as well, it will be possible to cut through some stronger edges as well if we have a lower value of sigma.

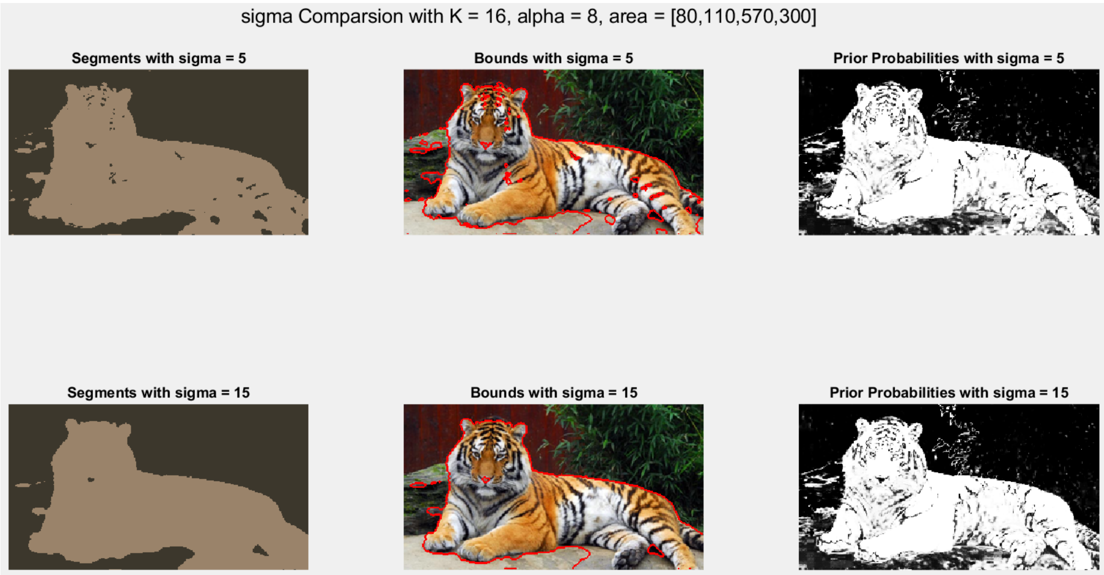

Figure 11.2-sigma value comparison for tiger1

So. the ideal values should be defined properly for images with different structures as in the following:

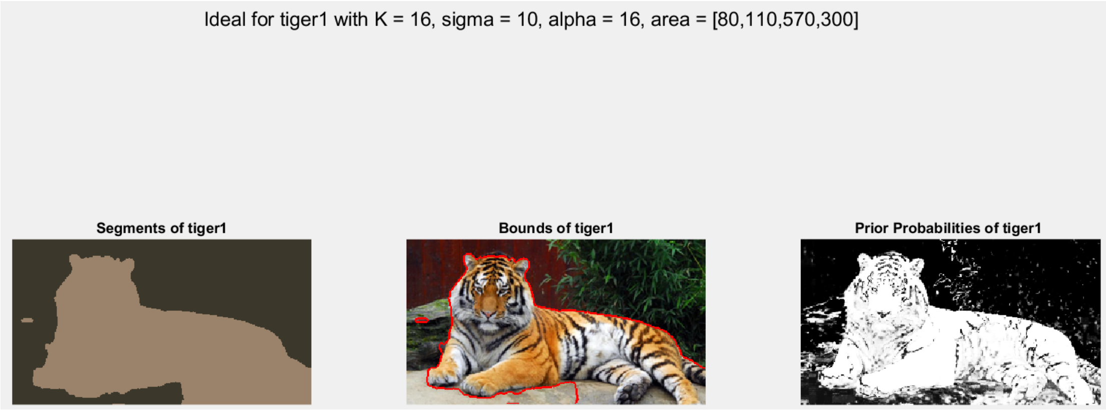

Figure 11.3-Ideal graph cut parameters for tiger1

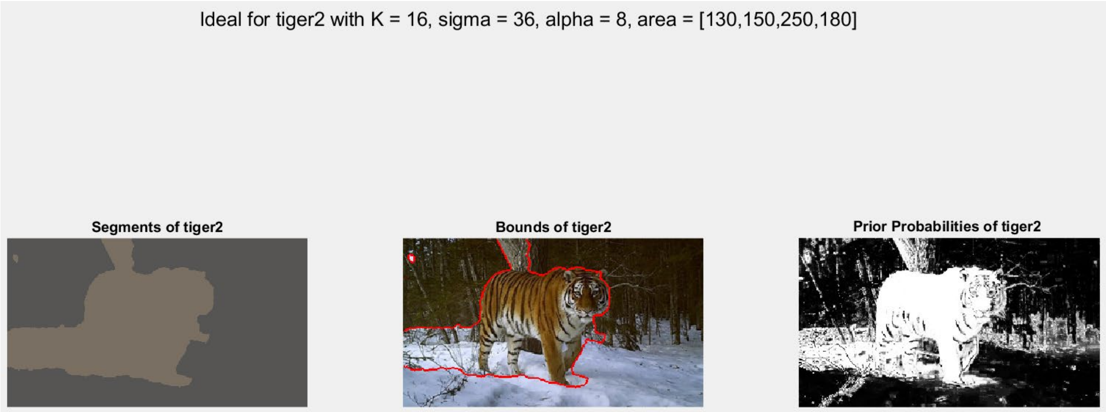

Figure 11.4-Ideal graph cut parameters for tiger2

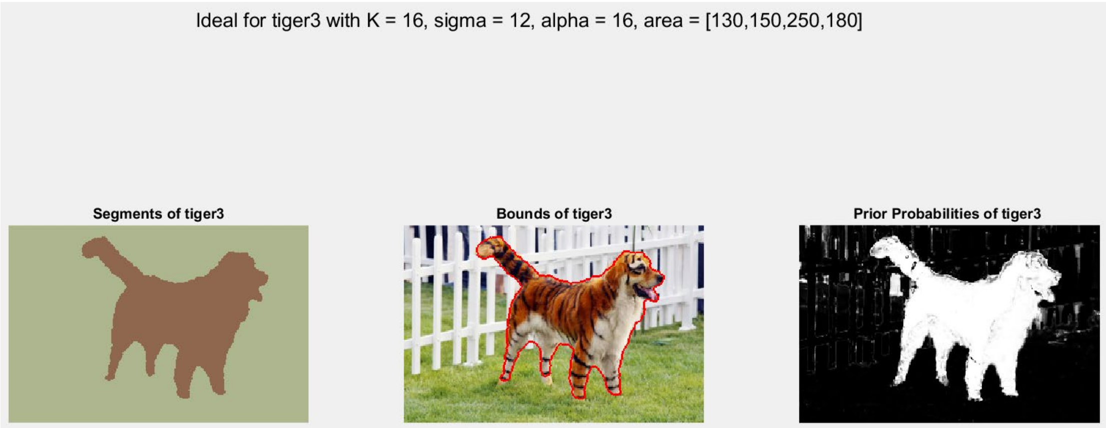

Figure 11.5-Ideal graph cut parameters for tiger3

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

Answers:

Around below K =4, it will start to lose its accuracy and will not be able to differ foreground and background. Because as we decrease K, we have less gaussian components which will make it harder to separate foreground and background. For that reason, K usually should be chosen more than 4.

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:

We are usually able to get the benefits of rectangle when there is a specific clear object in the image. It really helps us to separate foreground and background in those images. On the other hand, there can be some images where we have multiple objects that are included in the rectangle so that foreground and background will be in a mixed combination in that rectangle. In those cases, it may not be that much useful.

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:

Similarities:

- All of these methods are deploying clusters in order to have a segmentation based on similarity criteria (colors or distance)
- The purpose is the same which is having similar points in the same group and having unsimilar points in different groups.
- Each may require different parameters according to the complexity of the image
- We are benefiting gaussian distributions in methods like mean-shift and graph cuts
- Normalized cuts and Graph cuts methods are checking vertices and edges information between pixels for similarity check and achieve separation according to this information.

Differences:

- While graph cut and k- means clustering requires a number of clusters to be defined, mean-shift is not dependent on number of clusters but dependent on spatial and color bandwidths.
- Graph cut requires prior probabilistic information about the distributions of foreground and background while normal cut cares about only vertices and edges and their similarity values between pixels.