Hamza Benjelloun

Lab 3 : Image segmentation

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

The easy way to initialize the center clusters is to assign to each center a random color (R, V, G). However we notice that is not a good approach, because it is possible to initialize some clusters with a color that doesn't appear in the image or it is very far from the real colors that actually appear in the image. Then the appropriate way to initialize the K center clusters is to give to each center a color of a random pixel in the image.

Let's try our algorithm on tiger1 image by modifying K, L, the scale factor and the amount of pre blurring:


K = 8 L = 10 scale_factor = 1.0 image_sigma = 1.0


K = 4 L = 10 scale_factor = 1.0 image_sigma = 1.0

K = 8   L = 10   scale_factor = 1.0   image_sigma = 1.0

K = 4   L = 10   scale_factor = 1.0   image_sigma = 1.0

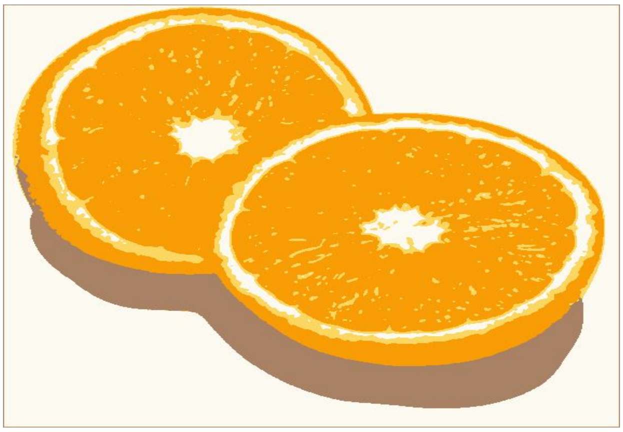
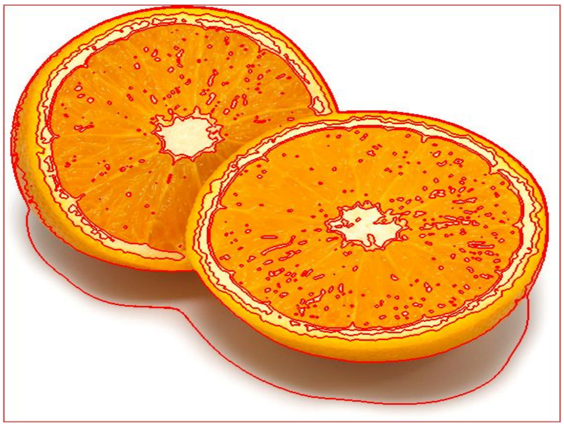
Question 2:How many iterations do you typically need to reach convergence, that is the point where no additional iterations will affect the end results?

There is no general rule that specifies the number of iterations needed to reach convergence. It depends on the number of clusters, the initialization of the k center clusters, the image ...

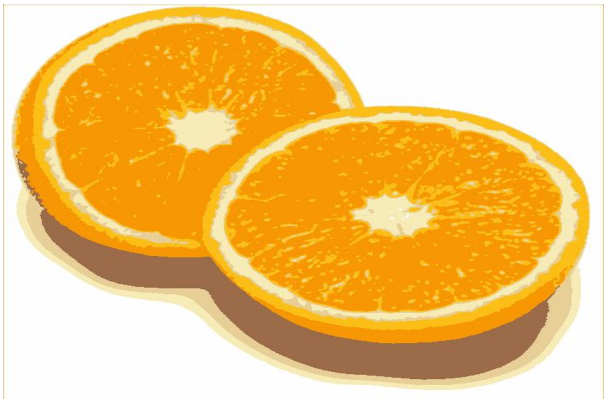
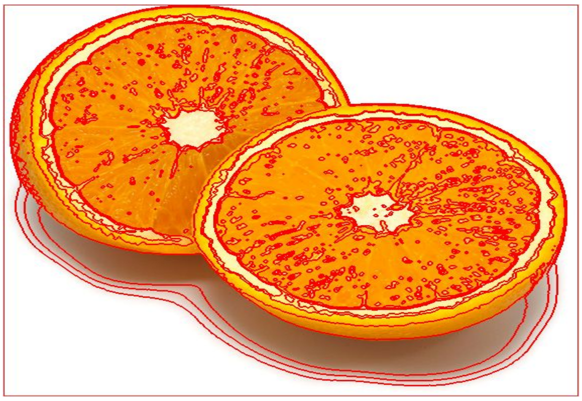
For example when we use the image ‘orange.jpg’ with k = 8, scale_factor = 1.0 and image_sigma = 1.0 , we reach convergence in 86 iterations.With the same parameters the image ‘tiger_1.jpg’ reaches convergence in 103 iterations.

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

K means using 4 clusters

K means using 4 clusters

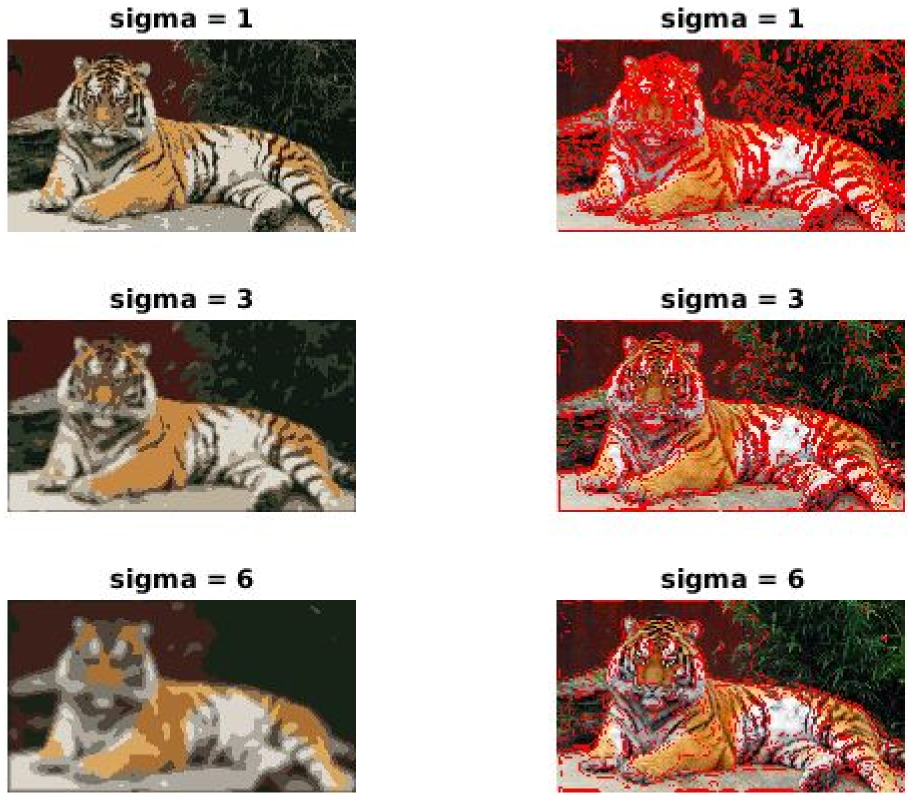



K means using 5 clusters

We notice that with k = 4 there is a superpixel that covers the inside of the left half orange and the peel of the right half. However with k = 5 the two halves of the orange are separated.

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

There are a lot of details in tiger images. We should blur it a little bit to remove noises and tiny details. Then we should change image_sigma.



<u>3 Mean-shift segmentation</u>

<u>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.</u>

Let’s first change spatial bandwidth:

We notice that the spatial bandwidth controls the size of segmented areas. The greater spatial bandwidth is, the wider segmented areas are.
The best result obtained with spacial bandwidth = 10

Now let’s changing the color bandwidths and fixing spatial bandwidth = 10:

color bandwidth = 3

color bandwidth = 3

color bandwidth = 5

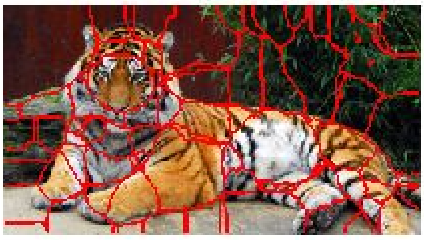
color bandwidth = 5

color bandwidth = 10

color bandwidth = 10

color bandwidth = 15

color bandwidth = 15



We notice that the greater the color bandwidth is, the smoothed the image is.
The best result we obtained is : color bandwidth = 15 and spatial bandwidth = 10

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

similarities:

Both methods are iterative.
Both methods take into account pixel colors to segmentate images.

differences:

K-means takes into account only pixel colors, however mean-shift takes into account color and position.

K-means needs a predefined number of clusters, however mean-shift finds the appropriate number of modes.

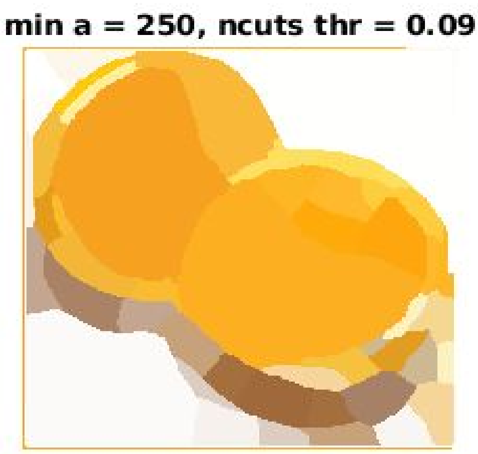
k-mean is very sensitive to outliers, contrary to mean-shift.

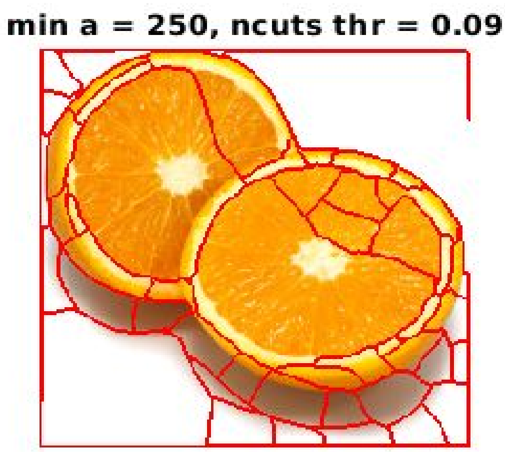
<u>3 Normalized cut:</u>

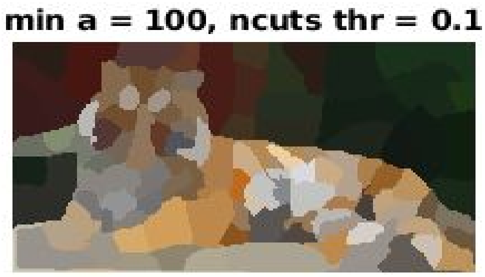
<u>Question 7:Does the ideal parameter setting vary depending on the images? If you look at theimages, can you see a reason why the ideal settings might differ? Illustrate with an example image using the parameters you prefer for that image.</u>

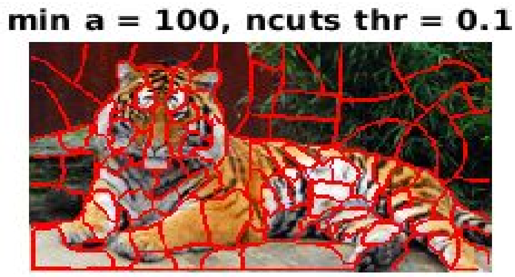
The ideal parameters depend a lot on the image.

The parameter Ncuts_treshold depends on the variation of color in the image, if we have a brutal variation like in the orange image from the white to the orange we need a low Ncuts_treshold. However if the variation of colors is very small, then we need a large value of Ncuts_treshold to have a good segmentation, for example in tigers images there are a lot of details and the variation of colors is very small then we need a high value of Ncuts_treshold. Moreover the parameter min_area depends on the size of features, if we have more details and we want to keep them then we decrease min_area.





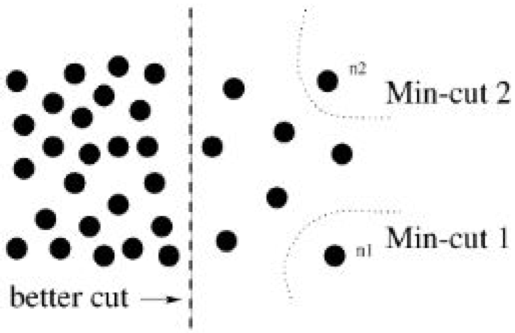




Question 8:Which parameter(s) was most effective for reducing the subdivision and still result in a satisfactory segmentation?

The main parameters that reduce the subdivisions and still result in a satisfactory segmentation are: min_area, ncuts_threshold and maxdepth.

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



We try to find a cut that minimize Ncut(A, B) where:

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

The cut that partitions out small isolated points will no longer have small cuts value, since the cut value will almost certainly be a large percentage of the total connection from that small set to all other nodes. In the case illustrated in the figure above, we see that the cut value across node n1 will be 100% of the total connection from that node. (The solution is inspired from the article '*Normalized Cuts and Image Segmentation*').

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

The radius controls the size of the neighboring area of a pixel. When we increase the radius, we take into account even further pixels, we decrease the number of segments and we increase time complexity.

radius = 3

radius = 12

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.

alpha = 8 sigma = 10

alpha = 15 sigma = 10

As we see the ideal choice of alpha and sigma vary between different images.And we should find a balance between them to have a good result. Let’s present the role of alpha and sigma:

<u>Sigma:</u>

If sigma is too large then $e_{ij} \approx \alpha$, therefore there are no big differences between pairwise edge costs. Thus segments will be large, and some background pixels will be included in the foreground.

if sigma is too small then a considerable color variation will decrease e_{ij} . And then we may miss some foreground pixel. For example in tiger image stripes will be included in the background .

<u>Alpha:</u>

If alpha is too large then only high color variation may decrease e_{ij} , Then some background pixels will be in the cut foreground.

if alpha is too small then we will have the same result as a too small sigma.

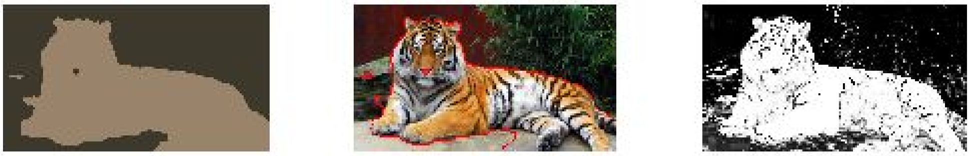
<u>Question 12:How much can you lower until the results get considerably worse?</u>

It depends on the image. For example by using tiger_1.jpg image we find:

k = 16

k = 8

k = 4



k = 3

We notice that when k = 3 the result becomes very bad.

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!

The graph cut method based on the rectangle gives good results as we see in question 11. But it depends a lot on the area chosen by the user. If the mask misses the foreground just slightly, we will have bad results. Moreover if we cannot surround the foreground in a rectangle (a case where foreground is mixed with background) this method will not work.

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!!

Similarities :

1/ All methods used to image segmentation.
2/ Mean-shift and graph cuts use the gaussian distribution model to model data.
3/ Normalized cut and graph cut transform image into graph based on some similarities measure.

Différences :

1/ Graph-cut needs the prior probability of foreground and background to construct the graph, however normalized cut doesn’t need any such information.

2/ Mean-shift uses both color and localisation to segmentate images however k-means uses just colors.

3/ Graph-cut is only about finding foreground and background. However normalized cut is about recursively dividing into two sets of vertices.

4/ the number of clusters given by meanèshift is undetermined . However K-means gives exactly k clusters.