Answers to questions in
Lab 3: Image segmentation

Name: ____Casper__Augustsson _____ Program: _____Celte-19____________

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 took K random numbers, this worked decent. Another method i used was to use the HSV space and strongly favour different colours. This to make sure the image had multiple colours and can adjust value and saturation by iterating L

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?

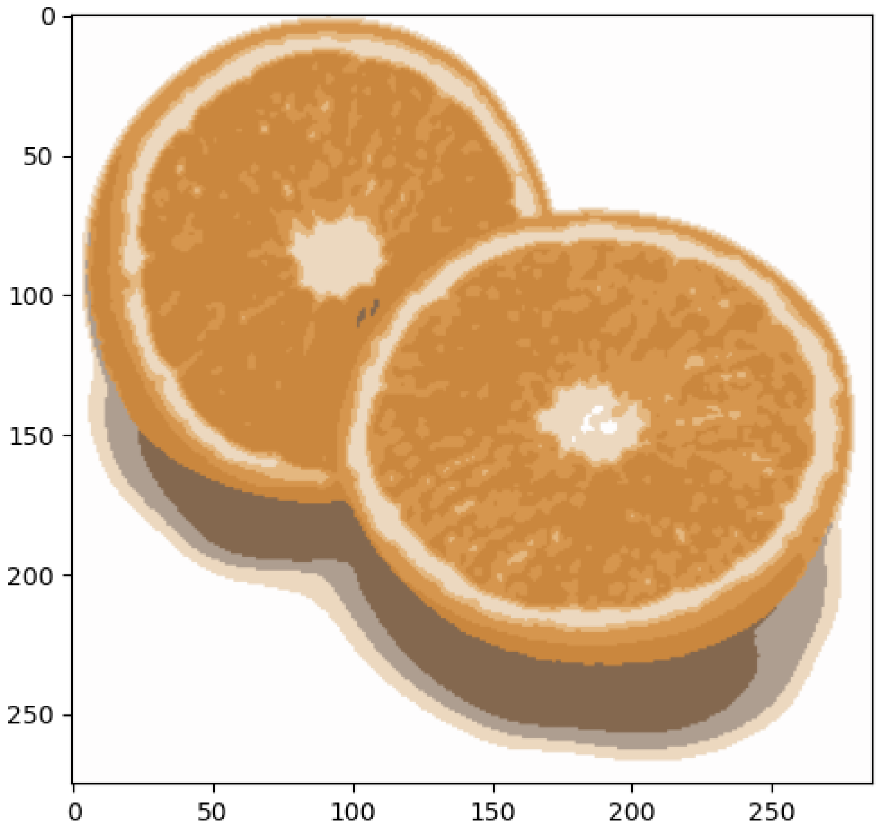
Answers:

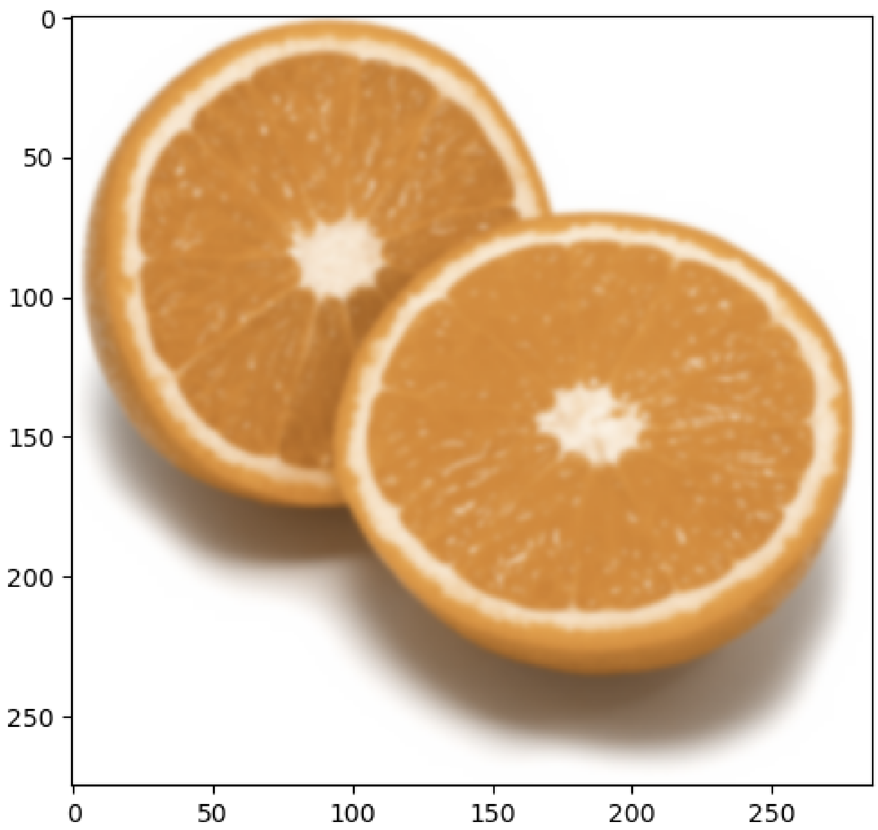
Around 5, but it depends on picture, colours and etc. But around 5 visually, and around 40-50 by def.

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:

As I understand, we use segments of pixels associated with one color, but they don't need to be connected in the picture. But with that said, K = 7 was the lowest

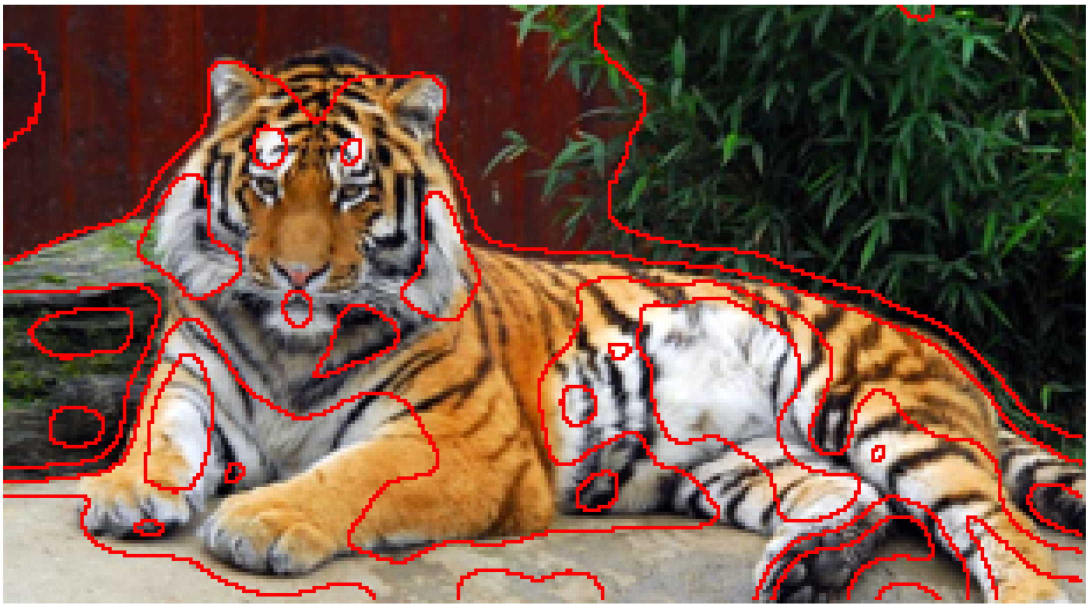

0
50
100
150
200
250
0
50
100
150
200
250


0
50
100
150
200
250
0
50
100
150
200
250

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

Answers:
We lower k to make it see the tiger and not the tiger eyebrow, tiger stripes, etc. We can also increase blurring a little bit

We need to make it less blurry, by reducing sigma.

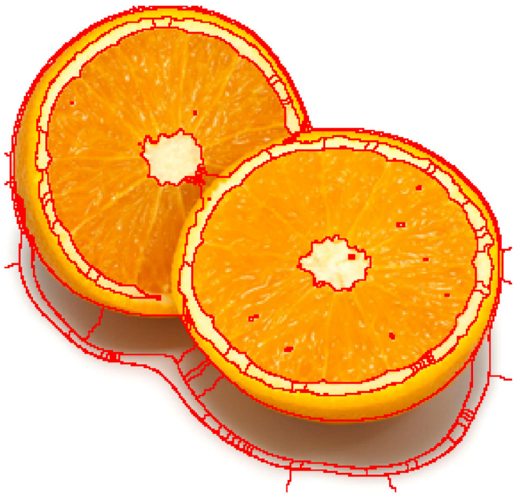


Question 5: How do the results change depending on the bandwidths? What settings did you prefer for the different images? Illustrate with an example image with the parameter that you think are suitable for that image.

Answers:

k_spartial, color = 5,50

k_spatial,color = 5,50



spatial,color = 10,10

Large uncertainty in spatial bandwidth groups by colour. Large uncertainty in colour bandwidth results in grouping by spatial features.

#Lower values tend to be better. But a little bit higher color bandwidth than spatial bandwidth #is also good.

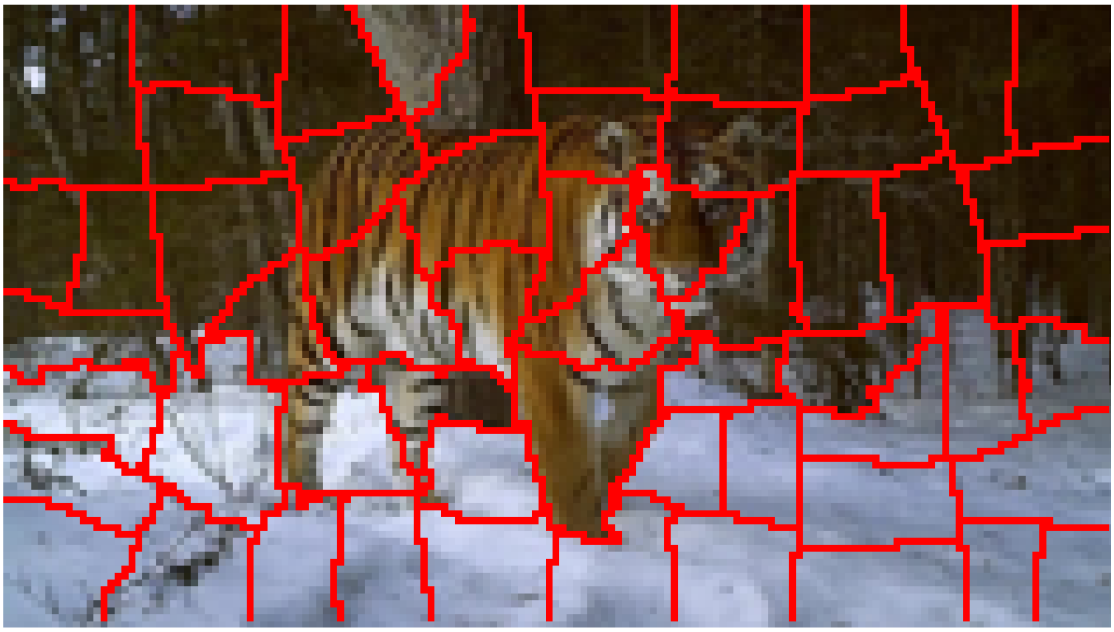
Question 6: What kind of similarities and differences do you see between K-means and mean-shift segmentation?

Answers:
Both algorithms are unsupervised and used to cluster data points, both minimize an objective function, both are robust to outliers. K-means require prior knowledge about amount of

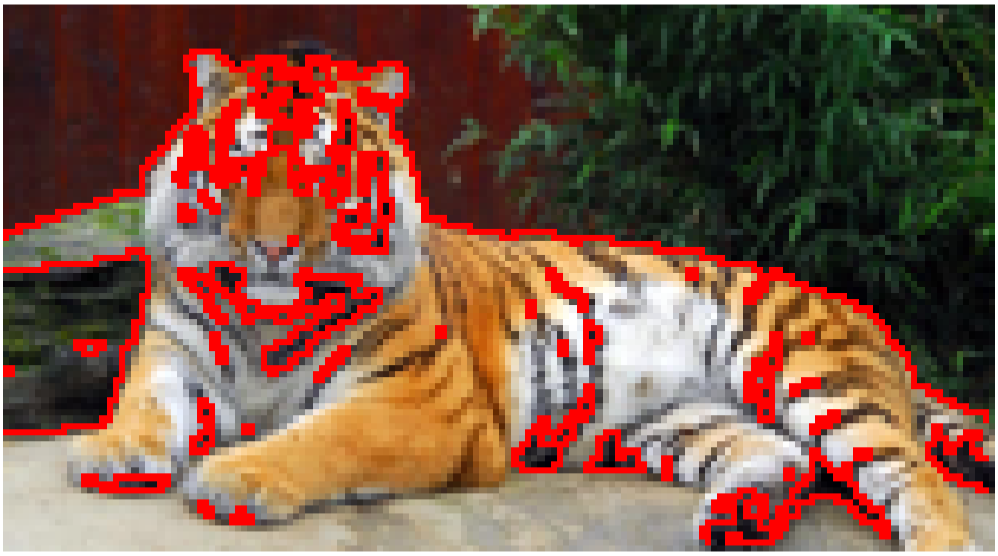
colors, while being faster. Mean shift can find none spherical data points dependent on position not only color.

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:



```
colour_bandwidth = 45.0  # color bandwidth
radius = 2               # maximum neighbourhood distance
ncuts_thresh = 0.10      # cutting threshold
min_area = 150           # minimum1 area of segment
max_depth = 12           # maximum splitting depth
scale_factor = 0.25      # image downscale factor
image_sigma = 0.5        # image preblurring scale
```



```
colour_bandwidth = 20.0   # color bandwidth
radius = 20 #10          # maximum neighbourhood distance
ncuts_thresh = 0.3 #0.3   # cutting threshold
min_area = 150            # minimum area of segment
max_depth = 1            # maximum splitting depth
scale_factor = 0.25       # image downscale factor
image_sigma = 0.5         # image preblurring scale
```

Yes, if the image have strong colors less colour bandwidth is needed.
If it is stripped, like a tiger,radius might be good to be able to skip, if it is really small objects a lower min area is needed, and if it is very edge more smoothing is needed.

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

Answers:
max_depth: lower value, stops it shallower.

For me it iwas colour bandwidth together with radius, depth. Increased radius a little bit and increased colour_bandwidth.

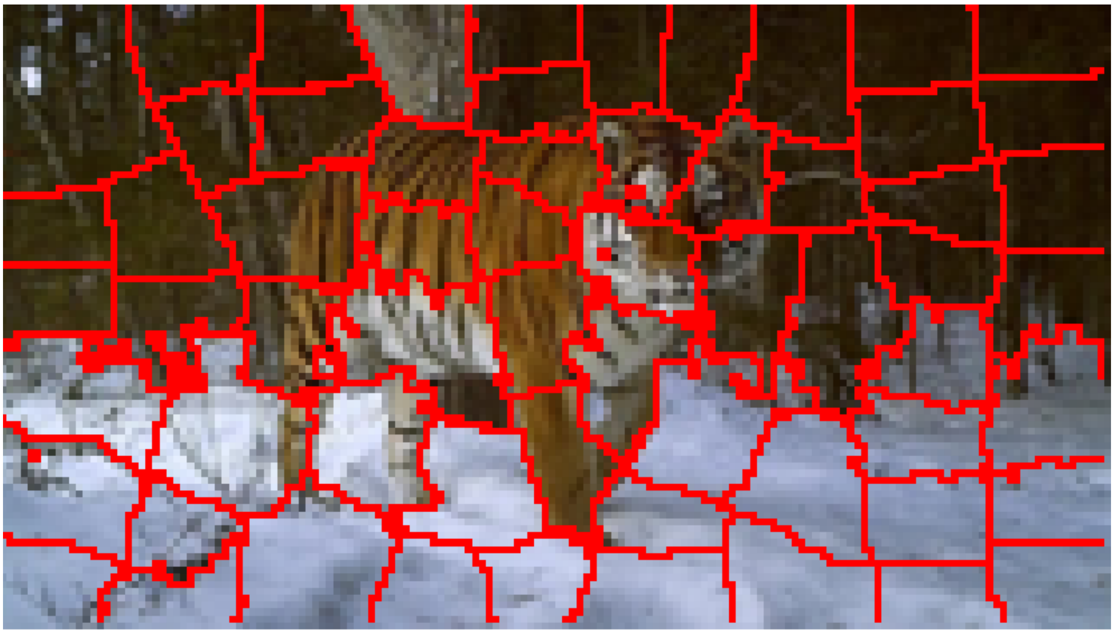
Question 9: Why does Normalized Cut prefer cuts of approximately equal size? Does this happen in practice?

Answers:
It tries to minimize the cost of the linked between pixels. If the image is uniform the costs will also be uniform if added up. Each segment will therefore contain about the same amount of links and therefore be of equal size. This is the case in reality. Very similar pixels tend to stay together and we see them as one object.

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

Answers:
It made everything better, mush better.

r=10



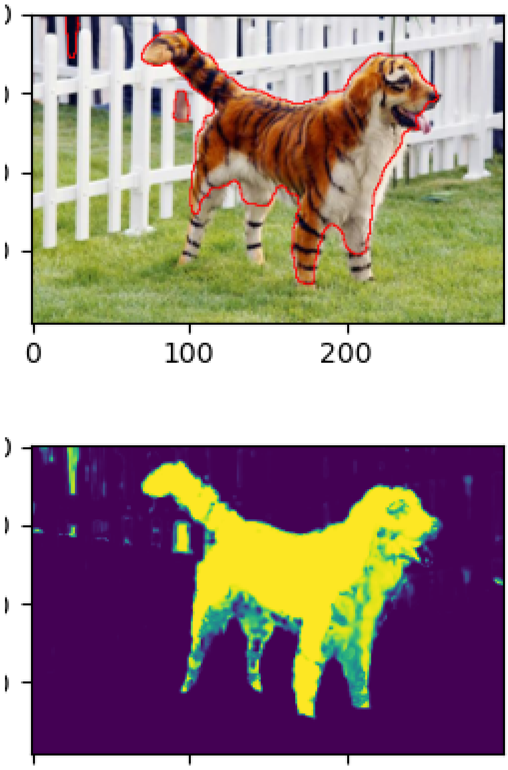
r=5

Best so far:

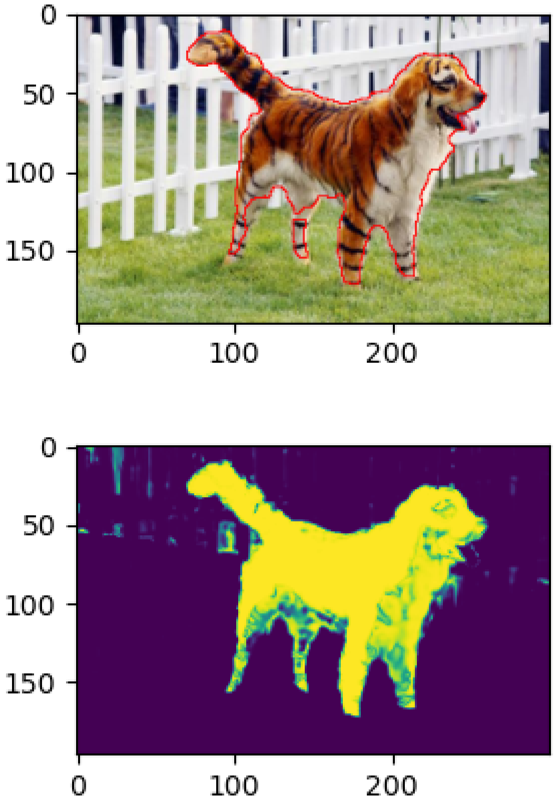
```
colour_bandwidth = 45.0  # color bandwidth
radius = 10  #10              # maximum neighbourhood distance
ncuts_thresh = 0.3 #0.3      # cutting threshold
min_area = 150           # minimum area of segment
max_depth = 12           # maximum splitting depth
scale_factor = 0.25      # image downscale factor
image_sigma = 0.5        # image preblurring scale
```

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.

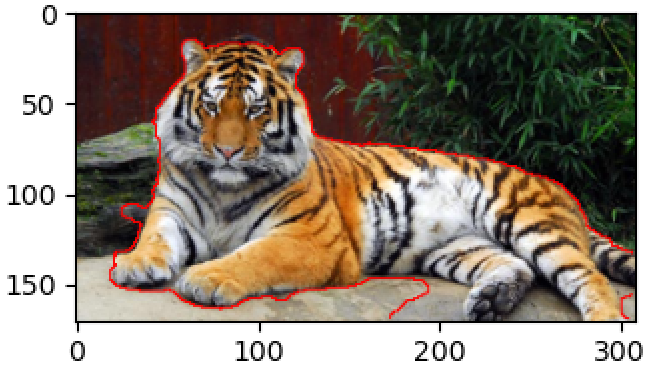
Answers:

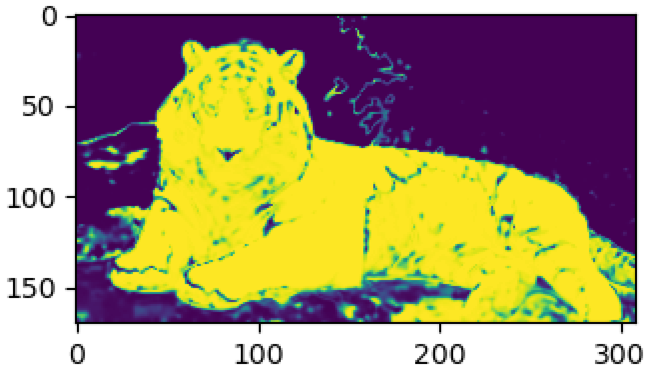


a=8,sig = 20



a=6,sig = 25





a=8, sig =20

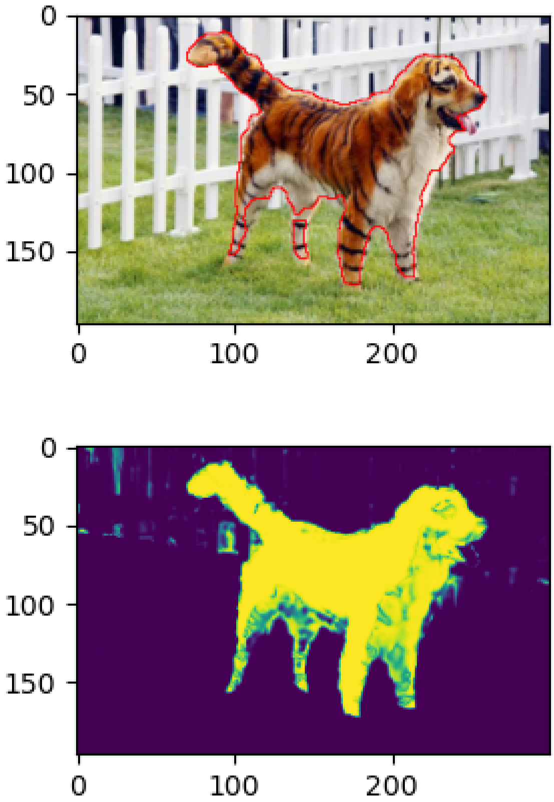
No, you can tune it a little bit with sigma and alpha, but I wouldn’t say it varies a lot

Theroy: If a lot of similar colour, you should decay fast when colours are similar because you want to differentiate between similar colours. Therefore a high sigma is wanted.

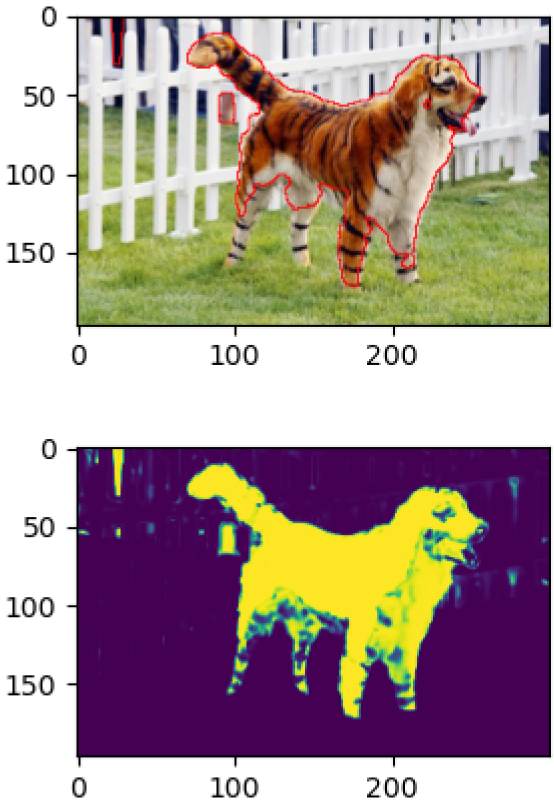
alpha, cost magnitude
sigma, cost decay speed

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

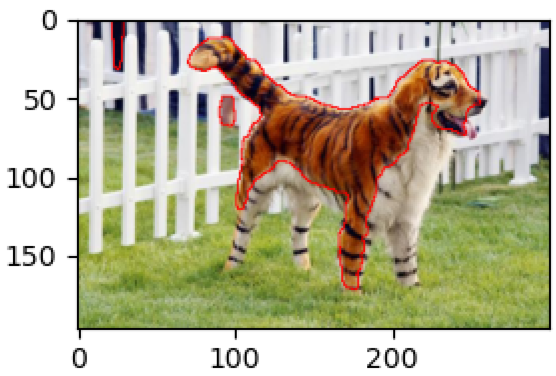
Answers:

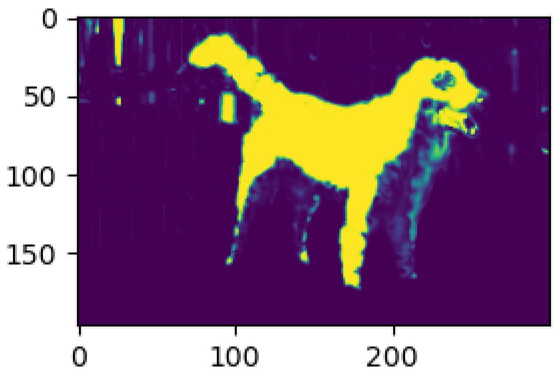

0
50
100
150
0
100
200
0
50
100
150
0
100
200

k=16


0
50
100
150
0
100
200
0
50
100
150
0
100
200

k=10





k=5

Around 5 things start to mess up, but it also depends on the luck you have with random which determines witch colour you caught. With good bounding box you can go even lower

---

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:
Yes it is, we find k colour gaussians in that windows that best represent the windows. Then we check how probable it is that the rest of the picture is made out of these colour gaussians and gave rise to them. The results becomes much better.

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:

The key differences between the four segmentation methods (K-means, Mean-shift, Normalized Cut and energy-based segmentation with Graph Cuts) are the approaches they use to segment an image. K-means, Mean-shift, and Normalized Cut are all partitioning-based segmentation algorithms, meaning they divide an image into regions using the same criteria, such as color or texture. K-means clustering divides an image into clusters based on the average color of each pixel, while Mean-shift uses a local search to identify clusters. Normalized Cut is a graph-based segmentation algorithm which divides an image into regions based on the similarity of their pixels. Energy-based segmentation with Graph Cuts is also a graph-based segmentation algorithm, but it uses an energy minimization approach to split an image into regions.

The similarities between the four methods are that they all use the same criteria—such as color or texture—to divide an image into regions, and they all aim to identify distinct regions within an image and label them accordingly. Furthermore, each of the algorithms takes into account the properties of the image when segmenting it.

Similarities:

- Ncut and graphcut both works on an optimization of cost on garph.
- K-means and means shift both iterate to find local maximum densities in their variables, k means optimaze colour space and mean shift optimize colourspace and space. They both try to find an statistical mean of “base fetures/colours”.
- Graphcut and meanshift implement features as gaussians.
- Ncut and graph cut both work on graphs.

Key differences:

- K-means have pre determined clusters K. The other methods do not. But Graph cut have in sort of a way, it have 2 clusters. Background and foreground.
- Graphcut tries to maximize flow between
