DD2423
Lab III
Alexandros Filotheou

1 K-means clustering

1.1 Question 1

The result of the initialization of the clustering process depends on whether there is a priori knowledge about the image to be segmented. In this example, where image "orange" is known, we can specify the colour of each initial cluster depending on what the image's dominant colours are. In this case where the colour diversity is low, it is possible to set two initial kernel values to the colours "orange" (roughly $(255, 150, 0)$ in the image in question) and "white" (exactly $(255, 255, 255)$ in the image in question) and let the rest of the kernels be chosen in random, to the extent of what level of detail is desirable.

In general, where the image to be segmented and its context may be unknown, the best way to initialize the kernels, so as to achieve a result that is not wrongly biased, is to let them be chosen at random, optinally with provision having been taken regarding the diversity of initial colours.

Figures 1, 2 and 3 illustrate the k-means segmentation method for image `orange` with

$$\texttt{(K,L,scale\_factor, }\sigma\texttt{)} \equiv \texttt{(8,10, 1, 1)}$$

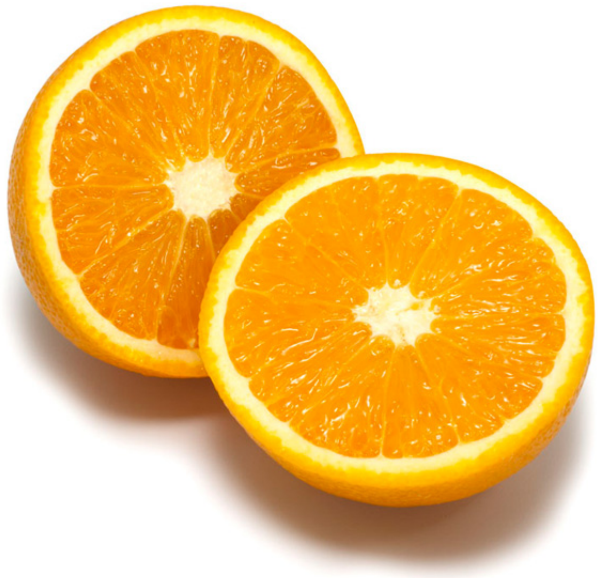

Figure 1: Image `orange`.

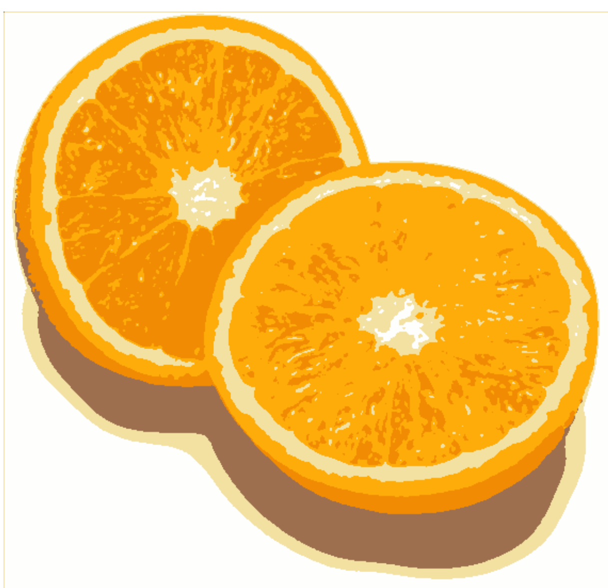

Figure 2: Image `orange` segmented.

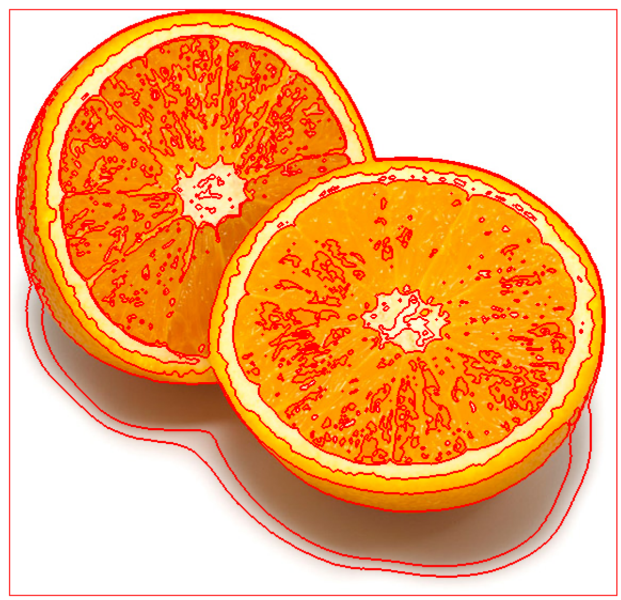

Figure 3: Image `orange` and the bounds of its segments.

1.2 Question 2

Convergence time depends linearly on the number of clusters K chosen and the size of the image. Furthermore it depends on the colour diversity of the image. Tables 1 and 2 illustrate the number of iterations it takes for the k-means clustering method to reach convergence depending on the number of clusters.

K	convergence iteration
1	2
2	9
3	14
4	11
5	18
6	21
7	17
8	35
9	46
10	23
11	38
12	31

Table 1: Number of centers K and number of iterations taken to reach convergence with regard to image `orange`. `(scale_factor,` σ`)` $\equiv$ `(1, 1)`

K	convergence iteration
1	2
2	12
3	31
4	83
5	67
6	52
7	57
8	208
9	90
10	103
11	124
12	132

Table 2: Number of centers K and number of iterations taken to reach convergence with regard to image `tiger1`. `(scale_factor,` σ`)` $\equiv$ `(1, 1)`

1.3 Question 3

If I understand the question correctly, this value is $K = 13$. Figures 4 and 5 illustrate that a superpixel covers parts from both halves of the orange for $K = 12$, whereas in figures 6 and 7, where $K = 13$, there is a clear boundary between them.

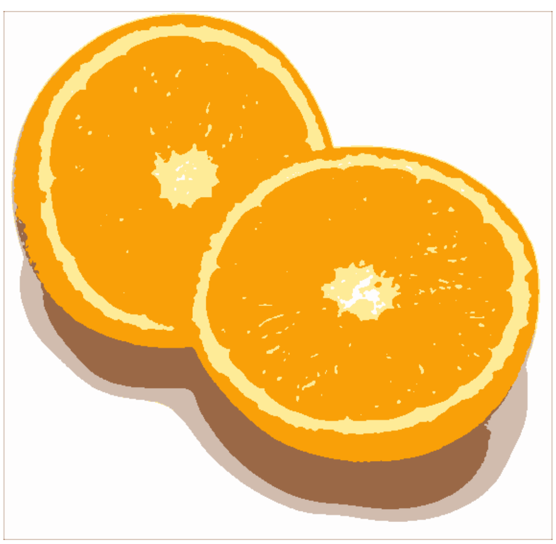

Figure 4: Image `orange` segmented using $K = 12$ clusters.

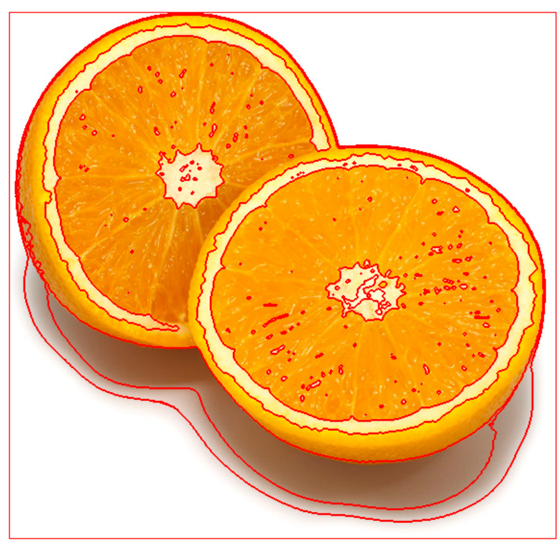

Figure 5: Image `orange` and the bounds of its segments for $K = 12$.

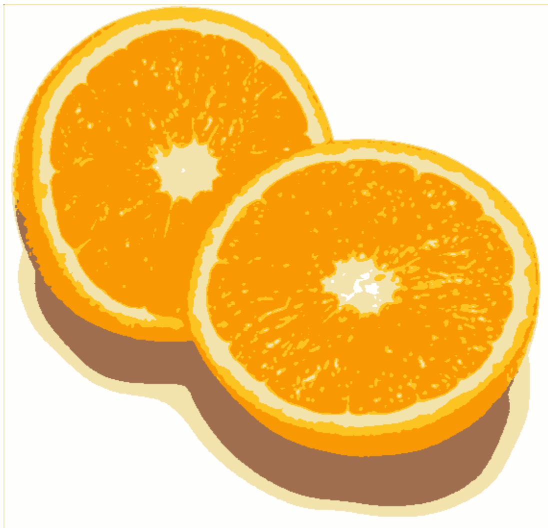

Figure 6: Image `orange` segmented using $K = 13$ clusters.

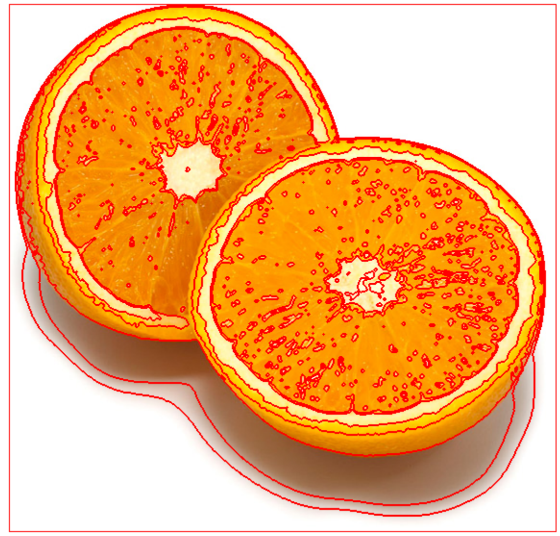

Figure 7: Image `orange` and the bounds of its segments for $K = 13$.

1.4 Question 4

Initially, what should be done is to increase the number of clusters K since image `tiger1` is more diverse in colours than image `orange`. As convergence takes more iterations to be met for increasing number of clusters, so should the iteration upper threshold L.

In the case where the clusters are not initialized in random but with a certain colour set that is desired to be achieved, then the centers of these clusters should be set to the values of those clusters.

2 Mean-shift segmentation

2.1 Preliminary results

2.1.1 Image **orange**. Varying σ_s^2.

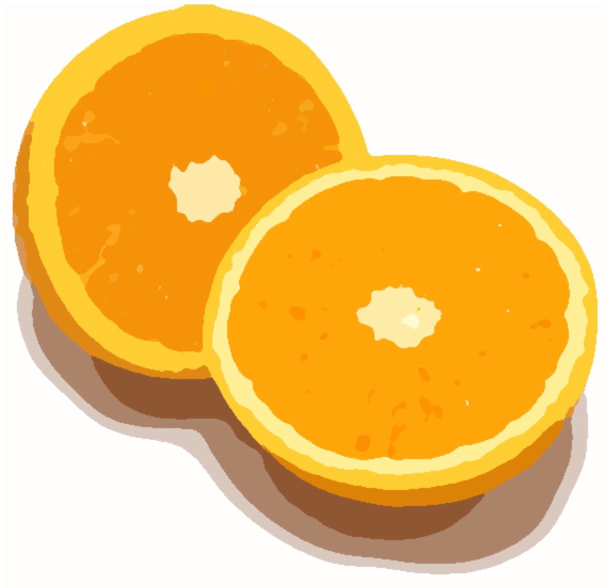

Figure 8: Image **orange** segmented with **mean-shift**. $(\sigma_s^2, \sigma_c^2) \equiv (3.0, 4.0)$

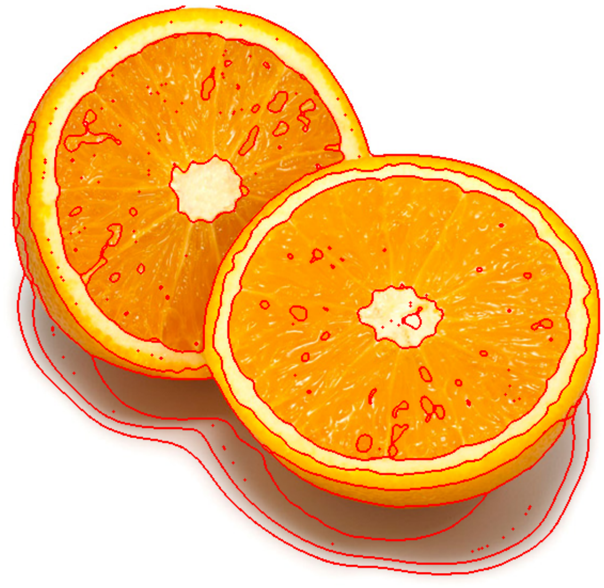

Figure 9: Segmentation bounds of image **orange**. $(\sigma_s^2, \sigma_c^2) \equiv (3.0, 4.0)$

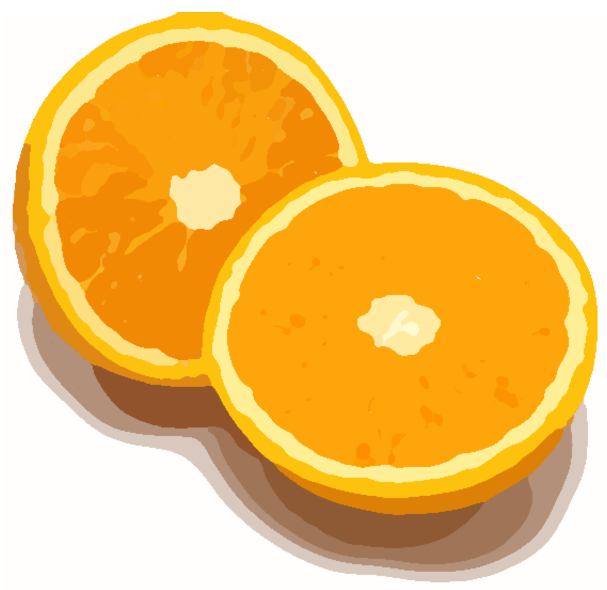

Figure 10: Image **orange** segmented with **mean-shift**. $(\sigma_s^2, \sigma_c^2) \equiv (4.0, 4.0)$

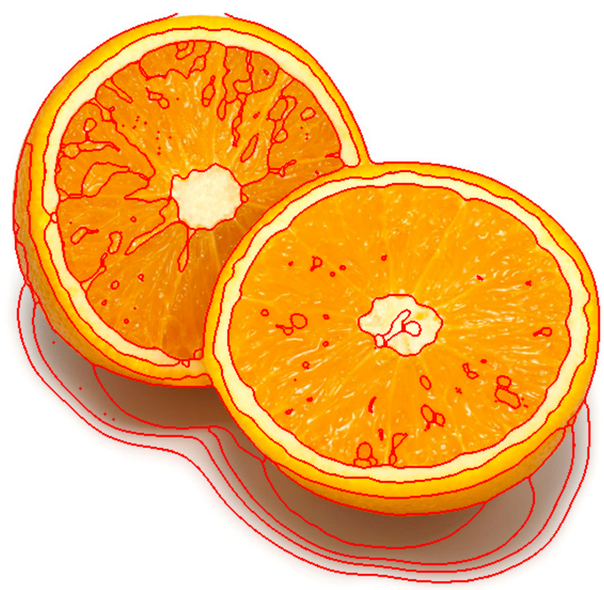

Figure 11: Segmentation bounds of image **orange**. $(\sigma_s^2, \sigma_c^2) \equiv (4.0, 4.0)$

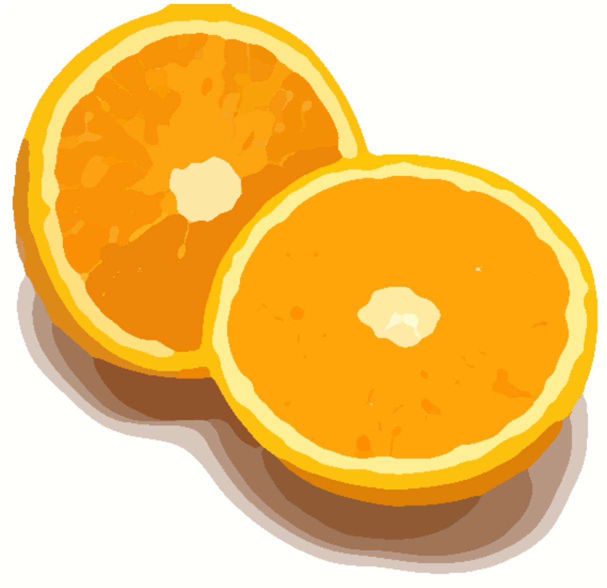

Figure 12: Image `orange` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (5.0, 4.0)$

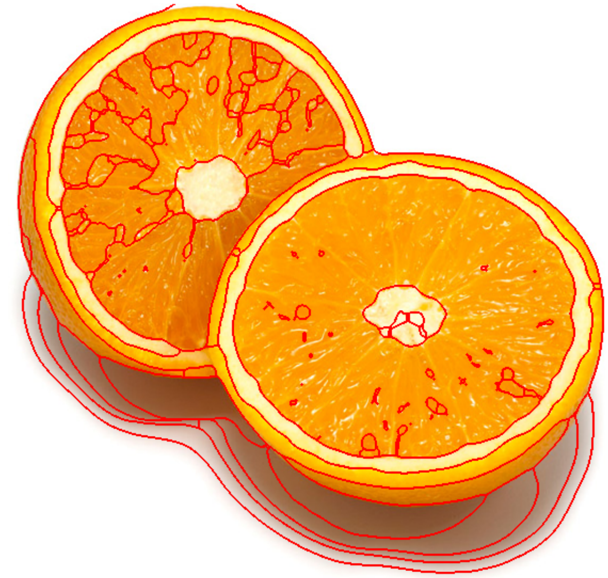

Figure 13: Segmentation bounds of image `orange`. $(\sigma_s^2, \sigma_c^2) \equiv (5.0, 4.0)$

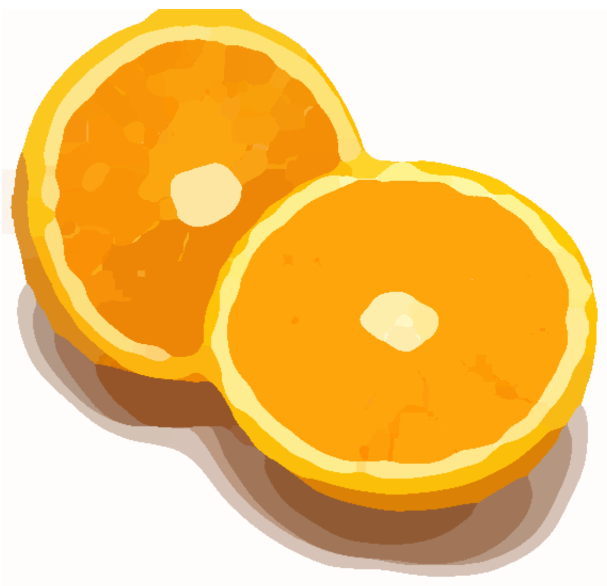

Figure 14: Image `orange` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 4.0)$

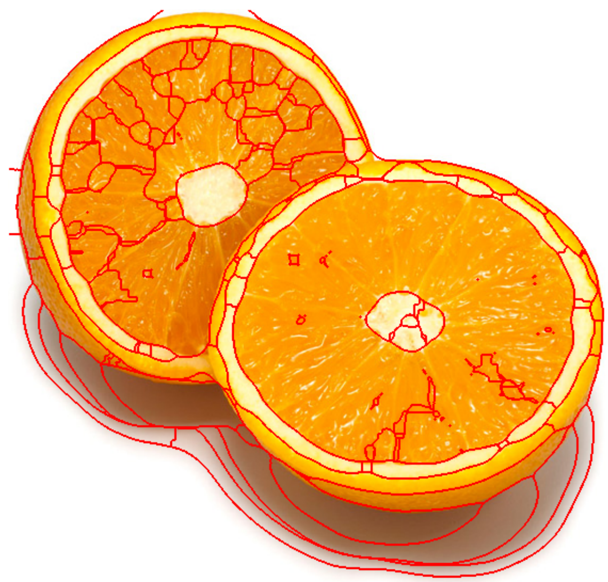

Figure 15: Segmentation bounds of image `orange`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 4.0)$

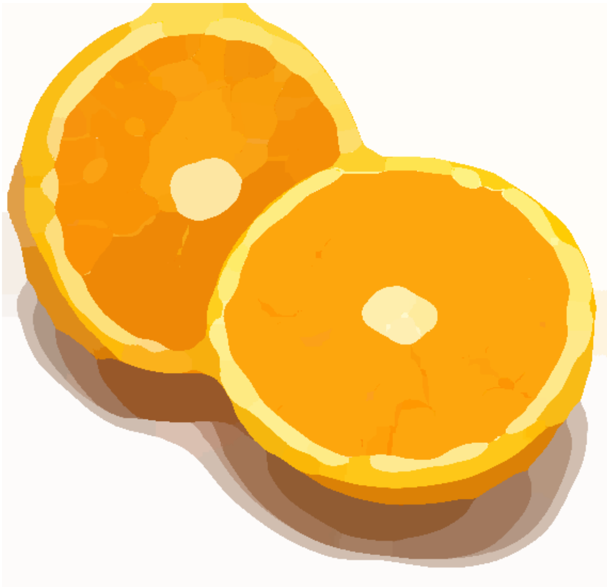

Figure 16: Image `orange` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (10.0, 4.0)$

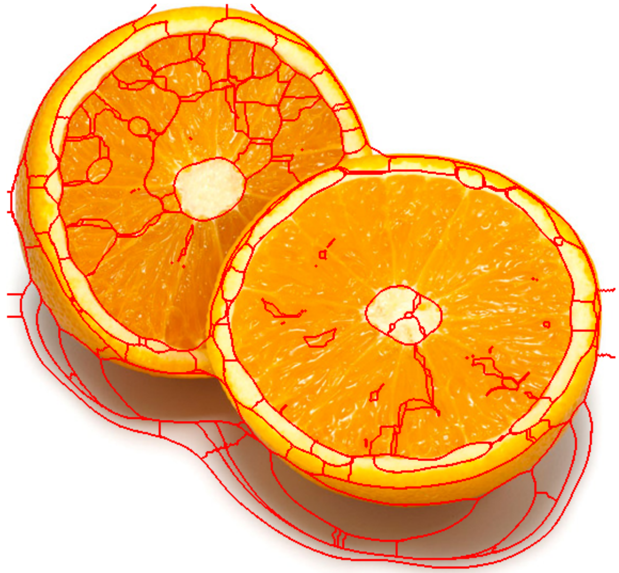

Figure 17: Segmentation bounds of image `orange`. $(\sigma_s^2, \sigma_c^2) \equiv (10.0, 4.0)$

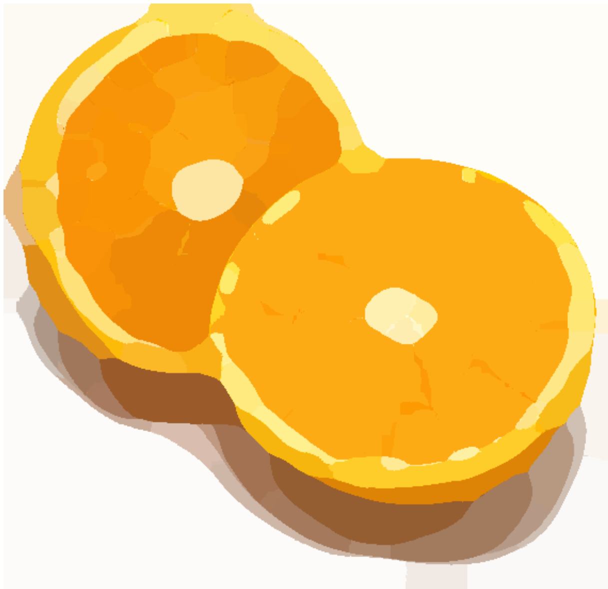

Figure 18: Image `orange` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (12.0, 4.0)$

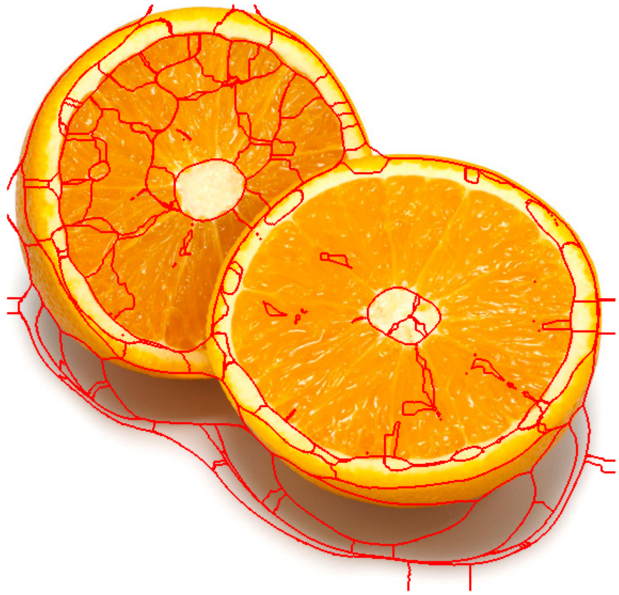

Figure 19: Segmentation bounds of image `orange`. $(\sigma_s^2, \sigma_c^2) \equiv (12.0, 4.0)$

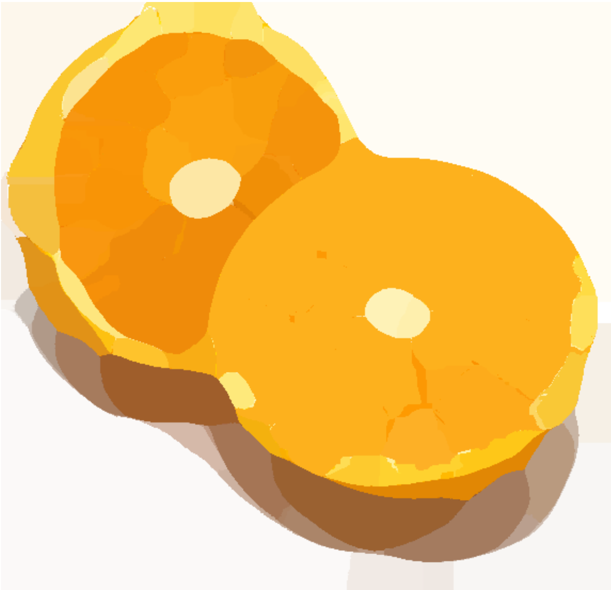

Figure 20: Image `orange` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (15.0, 4.0)$

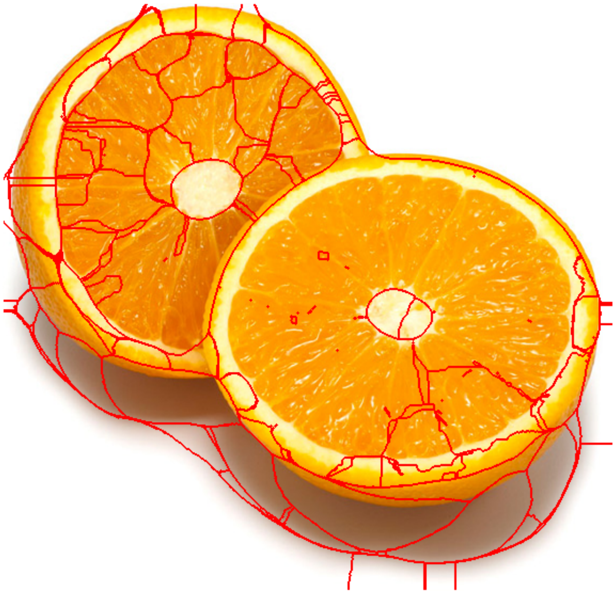

Figure 21: Segmentation bounds of image `orange`. $(\sigma_s^2, \sigma_c^2) \equiv (15.0, 4.0)$

Figure 22: Image `orange` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (20.0, 4.0)$

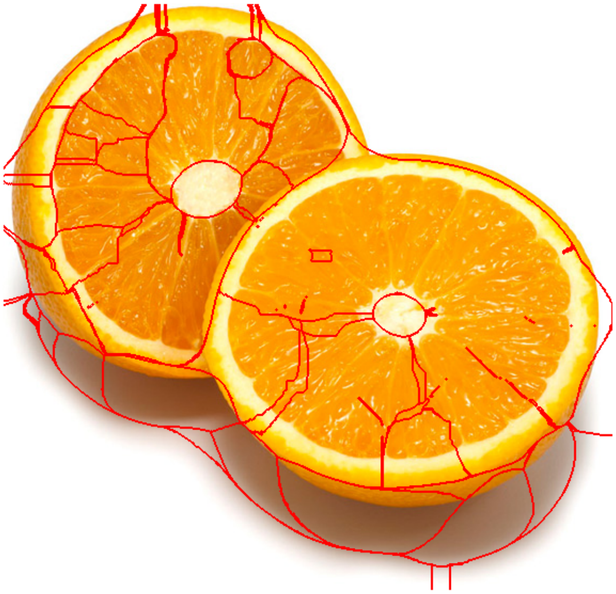

Figure 23: Segmentation bounds of image `orange`. $(\sigma_s^2, \sigma_c^2) \equiv (20.0, 4.0)$

2.1.2 Image **orange**. Varying σ_c^2.

Figure 24: Image `orange` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.0)$

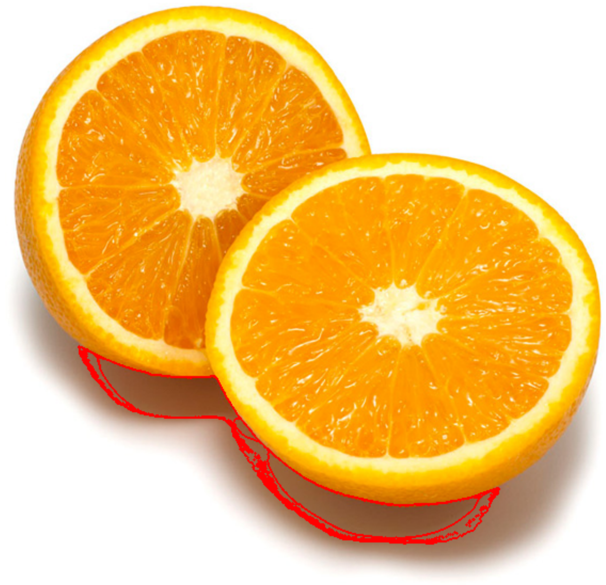

Figure 25: Segmentation bounds of image `orange`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.0)$

Figure 26: Image `orange` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.2)$

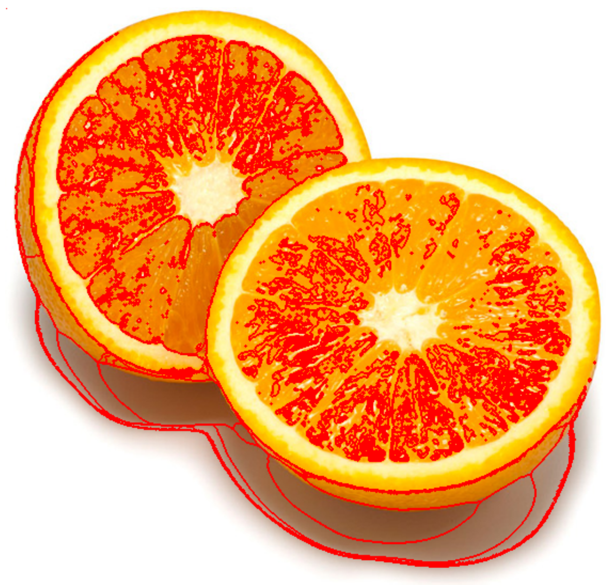

Figure 27: Segmentation bounds of image `orange`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.2)$

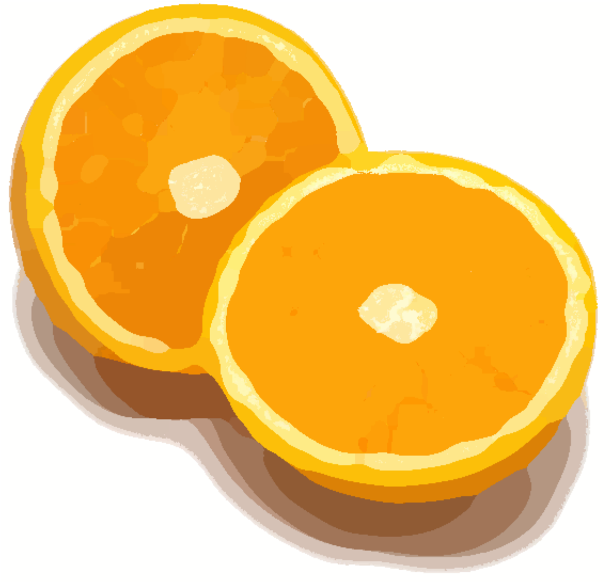

Figure 28: Image `orange` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.4)$

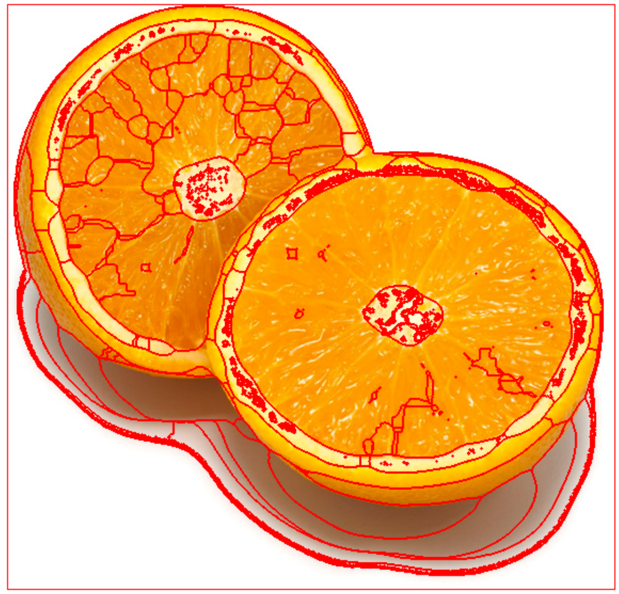

Figure 29: Segmentation bounds of image `orange`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.4)$

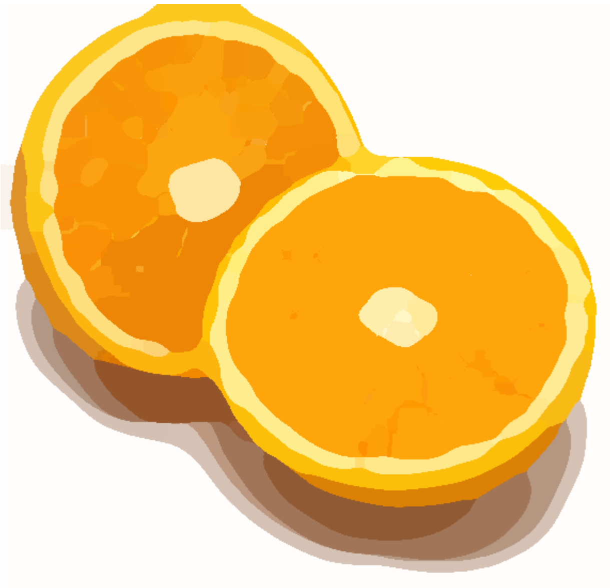

Figure 30: Image `orange` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.6)$

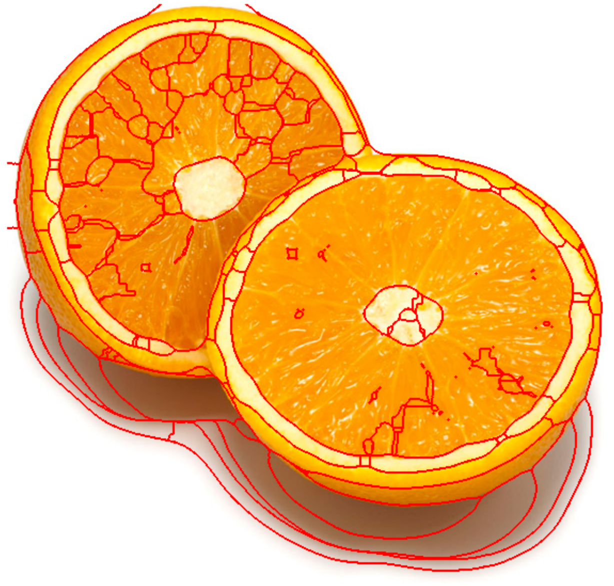

Figure 31: Segmentation bounds of image `orange`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.6)$

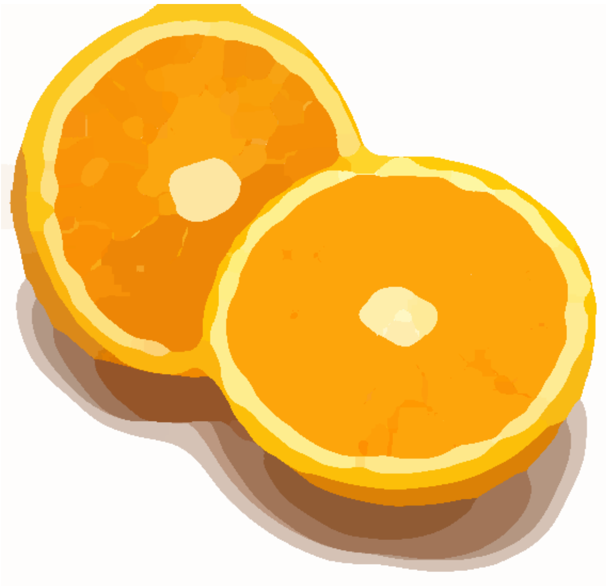

Figure 32: Image `orange` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.8)$

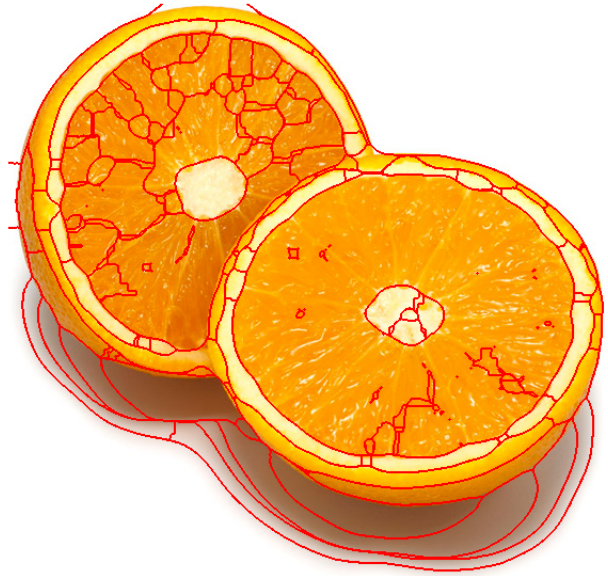

Figure 33: Segmentation bounds of image `orange`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.8)$

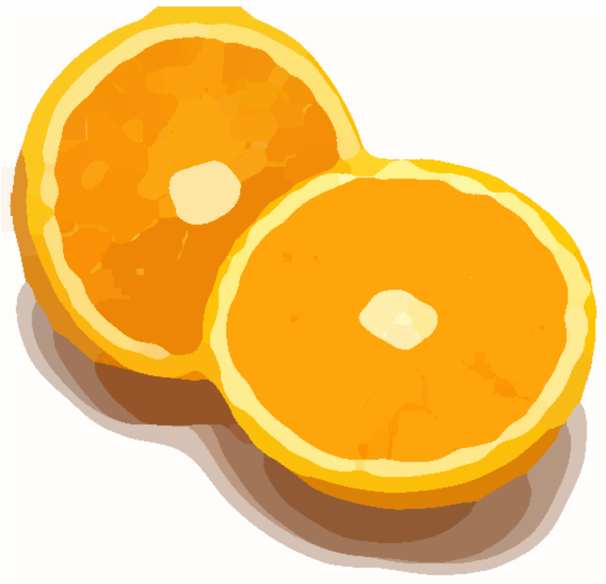

Figure 34: Image `orange` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 2.0)$

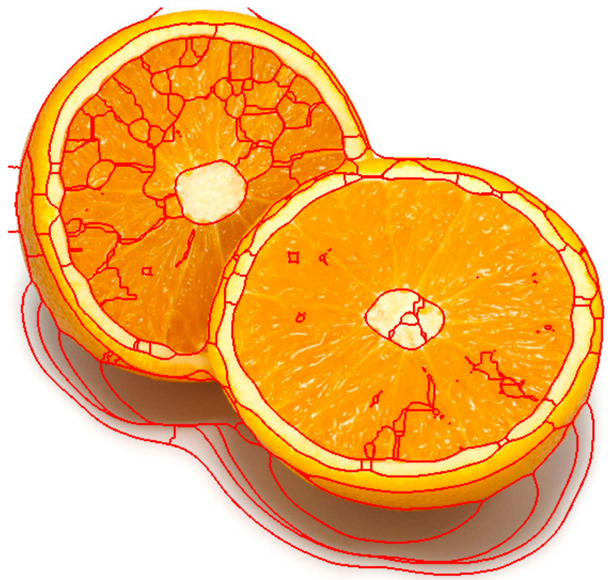

Figure 35: Segmentation bounds of image `orange`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 2.0)$

2.1.3 Image **tiger1**. Varying σ_s^2.

Figure 36: Image `tiger1` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (3.0, 4.0)$

Figure 37: Segmentation bounds of image `tiger1`. $(\sigma_s^2, \sigma_c^2) \equiv (3.0, 4.0)$

Figure 38: Image `tiger1` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (4.0, 4.0)$

Figure 39: Segmentation bounds of image `tiger1`. $(\sigma_s^2, \sigma_c^2) \equiv (4.0, 4.0)$

Figure 40: Image `tiger1` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (5.0, 4.0)$

Figure 41: Segmentation bounds of image `tiger1`. $(\sigma_s^2, \sigma_c^2) \equiv (5.0, 4.0)$

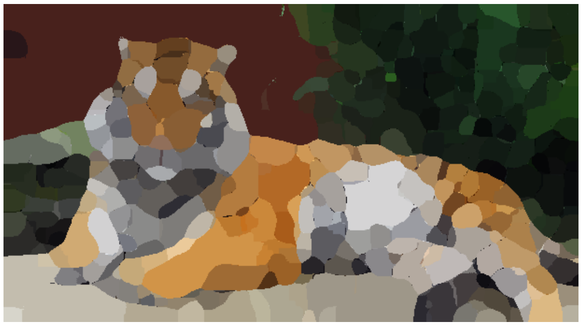

Figure 42: Image `tiger1` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 4.0)$

Figure 43: Segmentation bounds of image `tiger1`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 4.0)$

Figure 44: Image `tiger1` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (10.0, 4.0)$

Figure 45: Segmentation bounds of image `tiger1`. $(\sigma_s^2, \sigma_c^2) \equiv (10.0, 4.0)$

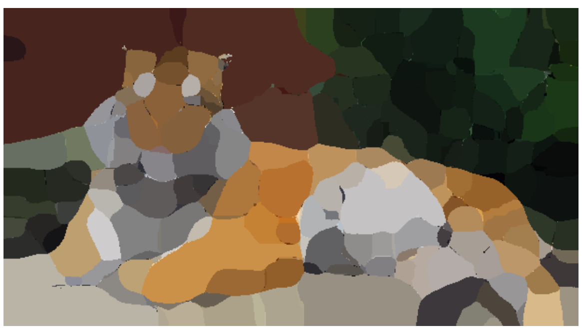

Figure 46: Image `tiger1` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (12.0, 4.0)$

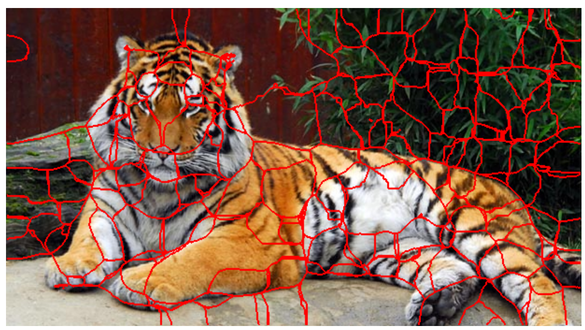

Figure 47: Segmentation bounds of image `tiger1`. $(\sigma_s^2, \sigma_c^2) \equiv (12.0, 4.0)$

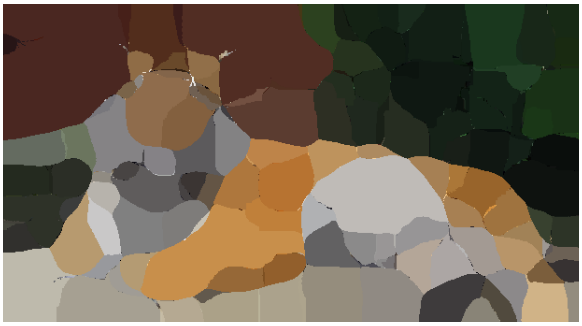

Figure 48: Image **tiger1** segmented with **mean-shift**. $(\sigma_s^2, \sigma_c^2) \equiv (15.0, 4.0)$

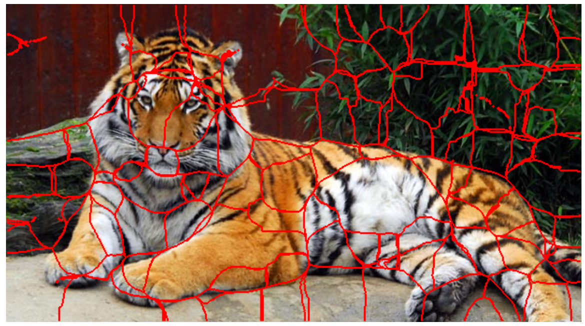

Figure 49: Segmentation bounds of image **tiger1**. $(\sigma_s^2, \sigma_c^2) \equiv (15.0, 4.0)$

Figure 50: Image **tiger1** segmented with **mean-shift**. $(\sigma_s^2, \sigma_c^2) \equiv (20.0, 4.0)$

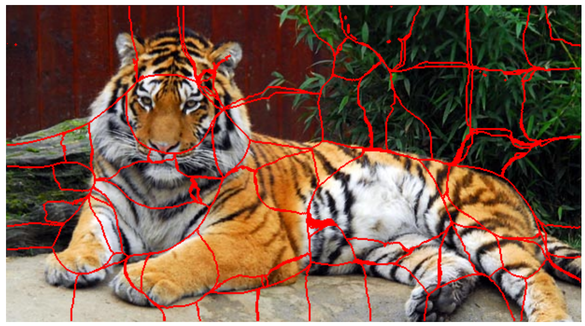

Figure 51: Segmentation bounds of image **tiger1**. $(\sigma_s^2, \sigma_c^2) \equiv (20.0, 4.0)$

2.1.4 Image **tiger1**. Varying σ_c^2.

Figure 52: Image **tiger1** segmented with **mean-shift**. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.0)$

Figure 53: Segmentation bounds of image **tiger1**. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.0)$

Figure 54: Image `tiger1` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.2)$

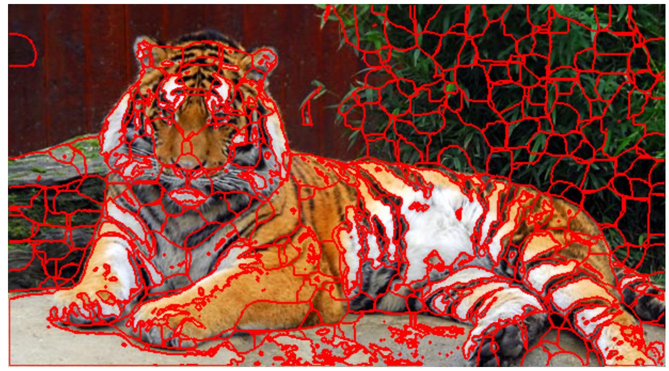

Figure 55: Segmentation bounds of image `tiger1`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.2)$

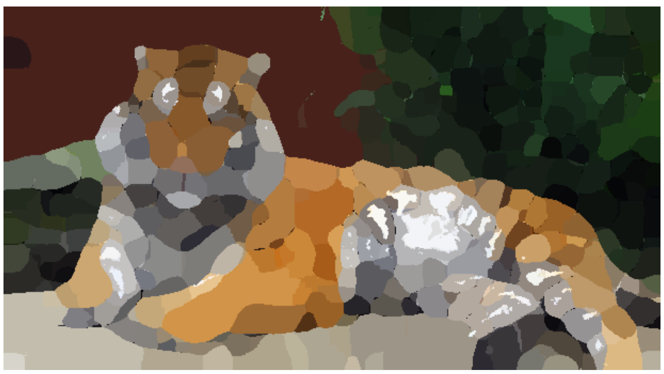

Figure 56: Image `tiger1` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.4)$

Figure 57: Segmentation bounds of image `tiger1`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.4)$

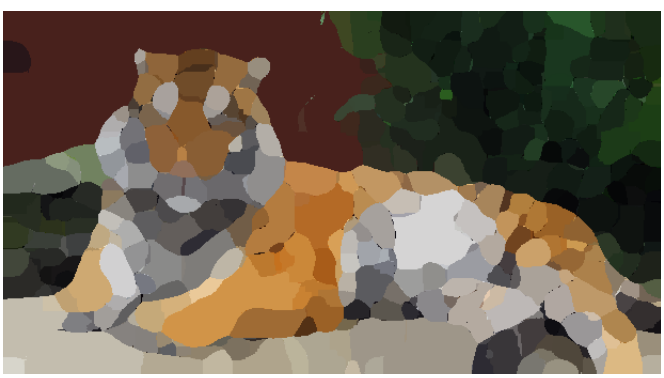

Figure 58: Image `tiger1` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.6)$

Figure 59: Segmentation bounds of image `tiger1`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.6)$

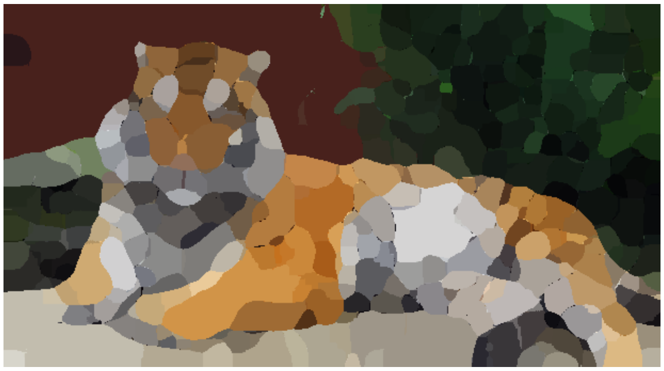

Figure 60: Image `tiger1` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.8)$

Figure 61: Segmentation bounds of image `tiger1`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.8)$

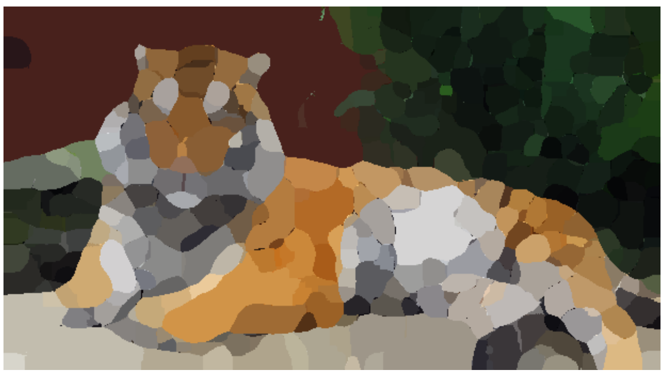

Figure 62: Image `tiger1` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 2.0)$

Figure 63: Segmentation bounds of image `tiger1`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 2.0)$

2.1.5 Image **tiger2**. Varying σ_s^2.

Figure 64: Image `tiger2` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (3.0, 4.0)$

Figure 65: Segmentation bounds of image `tiger2`. $(\sigma_s^2, \sigma_c^2) \equiv (3.0, 4.0)$

Figure 66: Image `tiger2` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (4.0, 4.0)$

Figure 67: Segmentation bounds of image `tiger2`. $(\sigma_s^2, \sigma_c^2) \equiv (4.0, 4.0)$

Figure 68: Image `tiger2` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (5.0, 4.0)$

Figure 69: Segmentation bounds of image `tiger2`. $(\sigma_s^2, \sigma_c^2) \equiv (5.0, 4.0)$

Figure 70: Image `tiger2` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 4.0)$

Figure 71: Segmentation bounds of image `tiger2`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 4.0)$

Figure 72: Image `tiger2` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (10.0, 4.0)$

Figure 73: Segmentation bounds of image `tiger2`. $(\sigma_s^2, \sigma_c^2) \equiv (10.0, 4.0)$

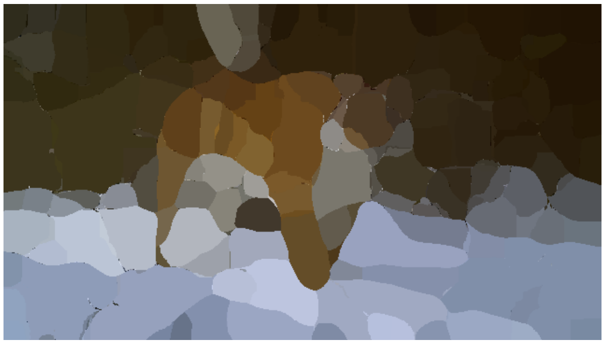

Figure 74: Image `tiger2` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (12.0, 4.0)$

Figure 75: Segmentation bounds of image `tiger2`. $(\sigma_s^2, \sigma_c^2) \equiv (12.0, 4.0)$

Figure 76: Image `tiger2` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (15.0, 4.0)$

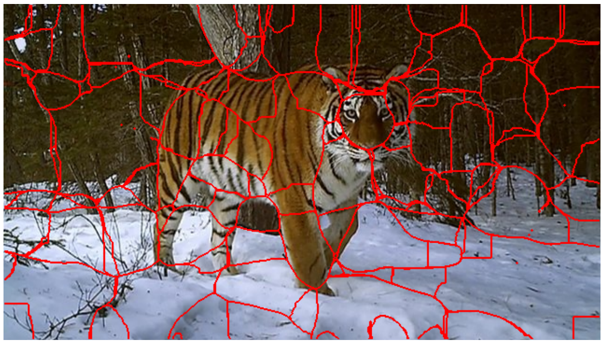

Figure 77: Segmentation bounds of image `tiger2`. $(\sigma_s^2, \sigma_c^2) \equiv (15.0, 4.0)$

Figure 78: Image **tiger2** segmented with **mean-shift**. $(\sigma_s^2, \sigma_c^2) \equiv (20.0, 4.0)$

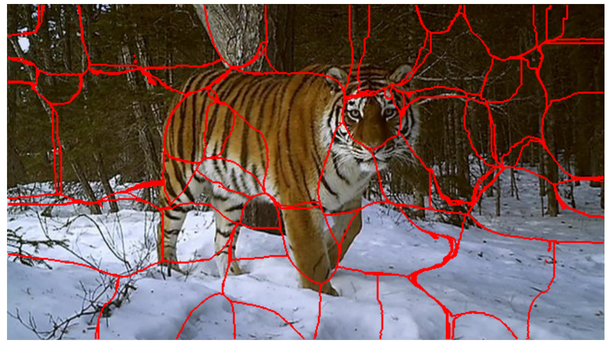

Figure 79: Segmentation bounds of image **tiger2**. $(\sigma_s^2, \sigma_c^2) \equiv (20.0, 4.0)$

2.1.6 Image **tiger2**. Varying σ_c^2.

Figure 80: Image **tiger2** segmented with **mean-shift**. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.0)$

Figure 81: Segmentation bounds of image **tiger2**. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.0)$

Figure 82: Image **tiger2** segmented with **mean-shift**. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.2)$

Figure 83: Segmentation bounds of image **tiger2**. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.2)$

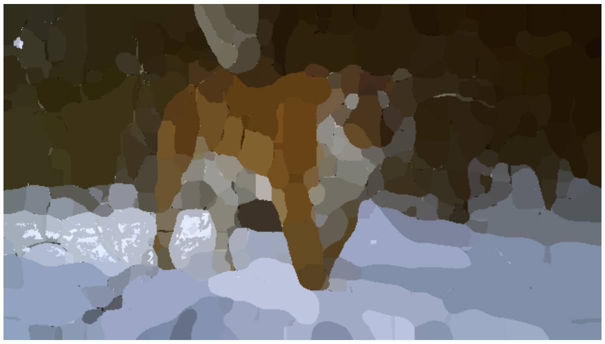

Figure 84: Image `tiger2` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.4)$

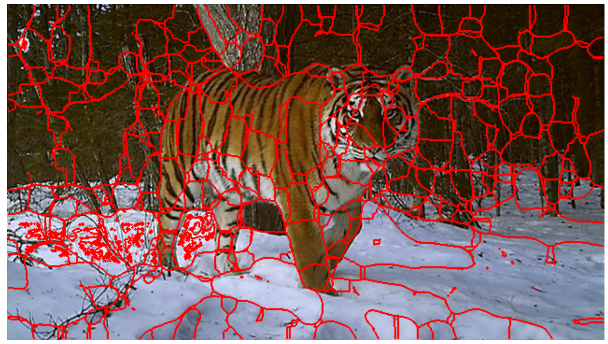

Figure 85: Segmentation bounds of image `tiger2`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.4)$

Figure 86: Image `tiger2` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.6)$

Figure 87: Segmentation bounds of image `tiger2`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.6)$

Figure 88: Image `tiger2` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.8)$

Figure 89: Segmentation bounds of image `tiger2`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.8)$

Figure 90: Image **tiger2** segmented with **mean-shift**. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 2.0)$

Figure 91: Segmentation bounds of image **tiger2**. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 2.0)$

2.1.7 Image **tiger3**. Varying σ_s^2.

Figure 92: Image **tiger3** segmented with **mean-shift**. $(\sigma_s^2, \sigma_c^2) \equiv (3.0, 4.0)$

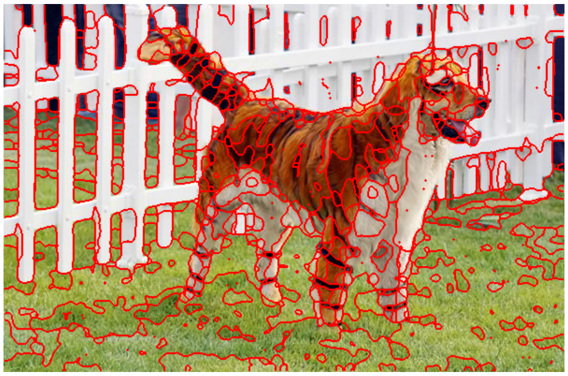

Figure 93: Segmentation bounds of image **tiger3**. $(\sigma_s^2, \sigma_c^2) \equiv (3.0, 4.0)$

Figure 94: Image **tiger3** segmented with **mean-shift**. $(\sigma_s^2, \sigma_c^2) \equiv (4.0, 4.0)$

Figure 95: Segmentation bounds of image **tiger3**. $(\sigma_s^2, \sigma_c^2) \equiv (4.0, 4.0)$

Figure 96: Image `tiger3` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (5.0, 4.0)$

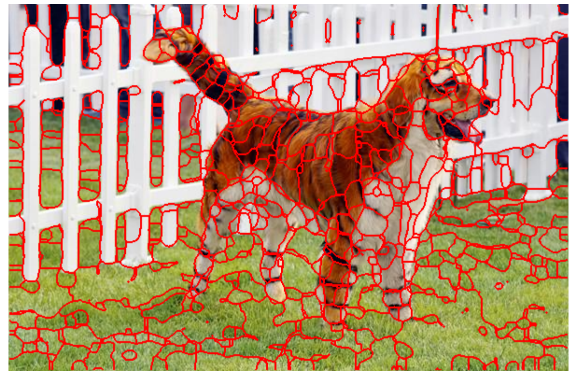

Figure 97: Segmentation bounds of image `tiger3`. $(\sigma_s^2, \sigma_c^2) \equiv (5.0, 4.0)$

Figure 98: Image `tiger3` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 4.0)$

Figure 99: Segmentation bounds of image `tiger3`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 4.0)$

Figure 100: Image `tiger3` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (10.0, 4.0)$

Figure 101: Segmentation bounds of image `tiger3`. $(\sigma_s^2, \sigma_c^2) \equiv (10.0, 4.0)$

Figure 102: Image `tiger3` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (12.0, 4.0)$

Figure 103: Segmentation bounds of image `tiger3`. $(\sigma_s^2, \sigma_c^2) \equiv (12.0, 4.0)$

Figure 104: Image `tiger3` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (15.0, 4.0)$

Figure 105: Segmentation bounds of image `tiger3`. $(\sigma_s^2, \sigma_c^2) \equiv (15.0, 4.0)$

Figure 106: Image `tiger3` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (20.0, 4.0)$

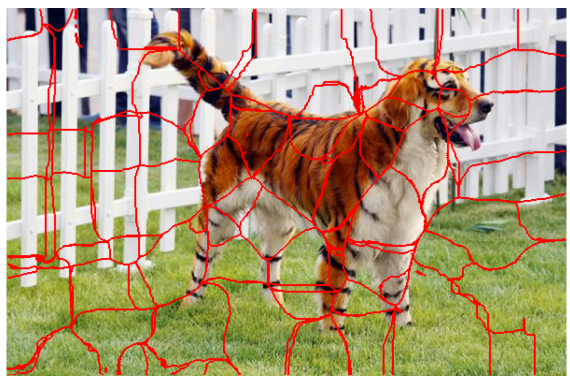

Figure 107: Segmentation bounds of image `tiger3`. $(\sigma_s^2, \sigma_c^2) \equiv (20.0, 4.0)$

2.1.8 Image **tiger3**. Varying σ_c^2.

Figure 108: Image `tiger3` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.0)$

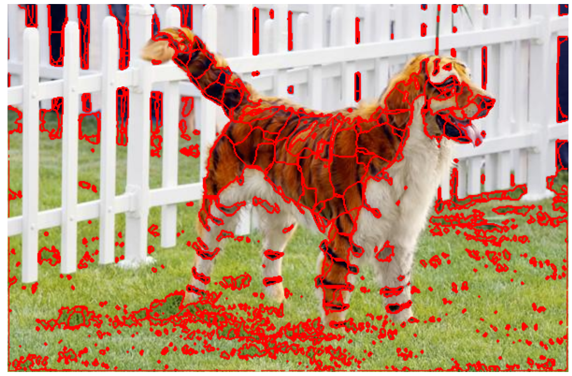

Figure 109: Segmentation bounds of image `tiger3`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.0)$

Figure 110: Image `tiger3` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.2)$

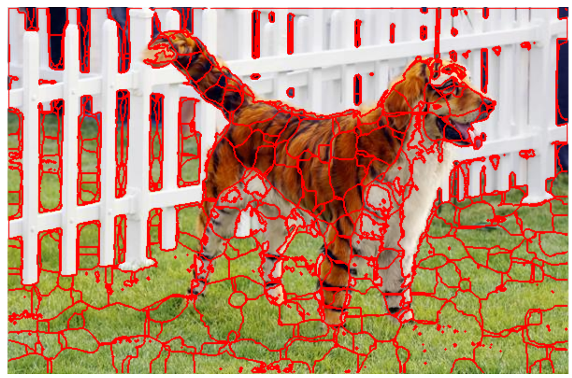

Figure 111: Segmentation bounds of image `tiger3`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.2)$

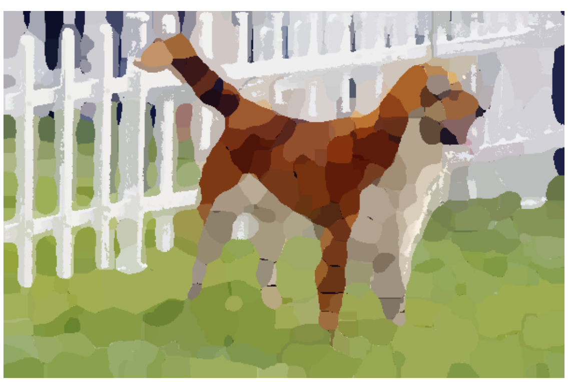

Figure 112: Image `tiger3` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.4)$

Figure 113: Segmentation bounds of image `tiger3`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.4)$

Figure 114: Image `tiger3` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.6)$

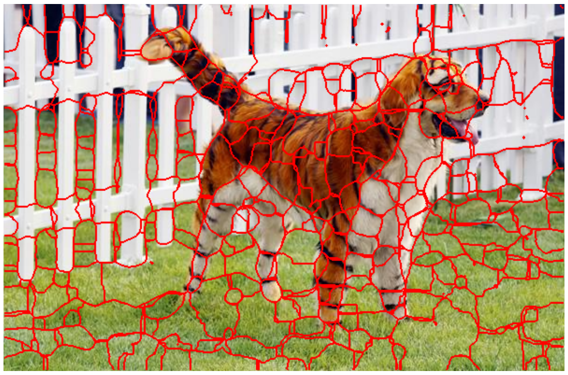

Figure 115: Segmentation bounds of image `tiger3`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.6)$

Figure 116: Image `tiger3` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.8)$

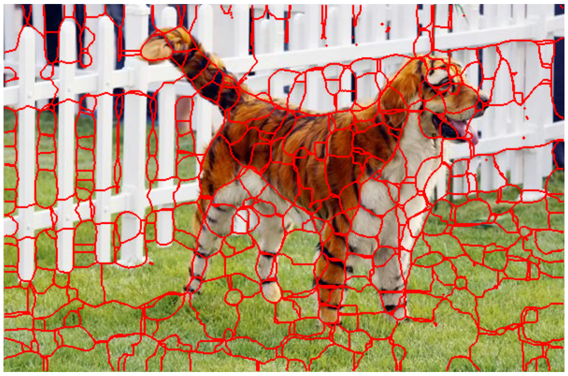

Figure 117: Segmentation bounds of image `tiger3`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 1.8)$

Figure 118: Image `tiger3` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 2.0)$

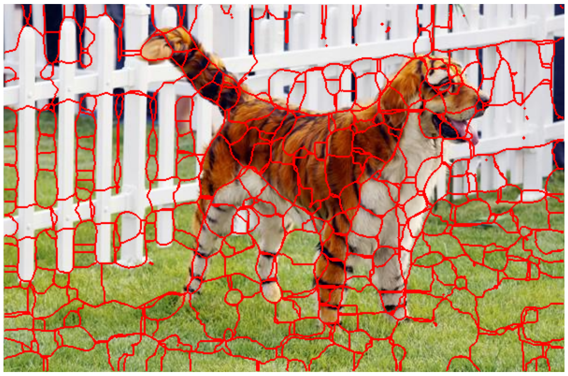

Figure 119: Segmentation bounds of image `tiger3`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 2.0)$

2.2 Best results

Having as purpose to segment the 4 images in as large segments as possible, without them covering more than one object in each one, the configuration that was used is illustrated in table 3.

image	σ_s^2	σ_c^2
`orange`	3.0	4.0
`tiger1`	8.0	4.0
`tiger2`	10.0	4.0
`tiger3`	8.0	4.0

Table 3: Spatial and colour bandwidths that result in good segmentation by the above purpose per image.

Figures 120 - 127 illustrate these results.

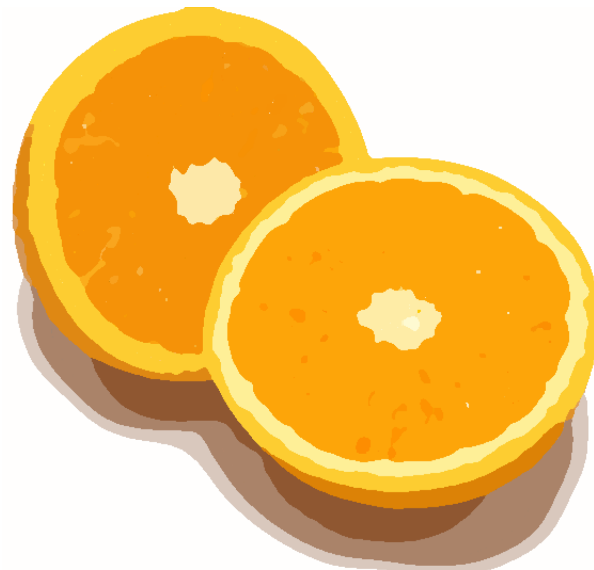

Figure 120: Image `orange` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (3.0, 4.0)$

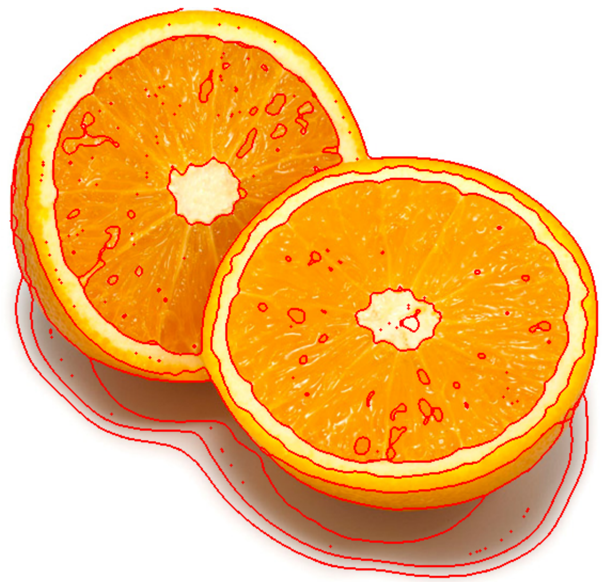

Figure 121: Segmentation bounds of image `orange`. $(\sigma_s^2, \sigma_c^2) \equiv (3.0, 4.0)$

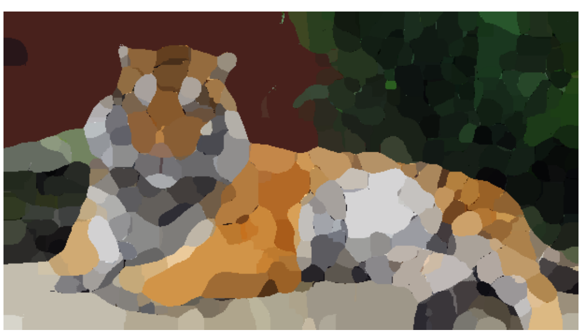

Figure 122: Image `tiger1` segmented with `mean-shift`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 4.0)$

Figure 123: Segmentation bounds of image `tiger1`. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 4.0)$

Figure 124: Image **tiger2** segmented with **mean-shift**. $(\sigma_s^2, \sigma_c^2) \equiv (10.0, 4.0)$

Figure 125: Segmentation bounds of image **tiger2**. $(\sigma_s^2, \sigma_c^2) \equiv (10.0, 4.0)$

Figure 126: Image **tiger3** segmented with **mean-shift**. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 4.0)$

Figure 127: Segmentation bounds of image **tiger3**. $(\sigma_s^2, \sigma_c^2) \equiv (8.0, 4.0)$

2.3 Question 5

Figures 8 - 23, 36 - 51, 64 - 79 and 92 - 107 illustrate the **mean-shift** segmentation method for varying values of spatial bandwidths, and colour bandwidth set to $\sigma_c^2 = 4.0$.

As seen in the above figures, the value of the spatial bandwidth is proportional to the area of the regions of interest, that is, the resulting segments. As the area of the segments becomes larger, their number becomes smaller, and thus so does the number of modes. Hence, there is a coarser approximation both spatial-wise and colour-wise.

Figures 24 - 35, 52 - 63, 80 - 91 and 108 - 119 illustrate the **mean-shift** segmentation method for varying values of colour bandwidths, and spatial bandwidth set to $\sigma_s^2 = 8.0$.

As seen in the above figures, a higher value for the colour bandwidth results in a better colour approximation.

2.4 Question 6

K-means and mean-shift both treat the colour and/or position of pixels as samples from a probability distribution and try to determine its clusters or modes.

Unlike k-means, mean-shift cannot segment an image into a predefined number of segments. Furthermore k-means does not take the spatial dimension into account when segmenting; only the colour component. This means that a segment, which is only specified by its colour, can span over multiple regions, while this cannot be the case with mean-shift, whose segments are spatially concentrated.

3 Normalized Cut

3.1 Question 7

The ideal parameter setting for each of these images depend on various factors, along with the degree of segmentation that is desired.

For example, parameter `min_area` depends on the size of the image, the size of the objects that are to be segmented and the complexity of each object's structure. Figures 128 - 131 illustrate the effect that the variation of the value of `min_area` has on image `orange`. Notice that when `min_area` ≥ 100 the center of the left half of the orange cannot be found and is taken over by its surroundings.

Figure 128: Image `orange` segmented with the default settings and `min_area = 500`

Figure 129: Image `orange` segmented with the default settings and `min_area = 200`

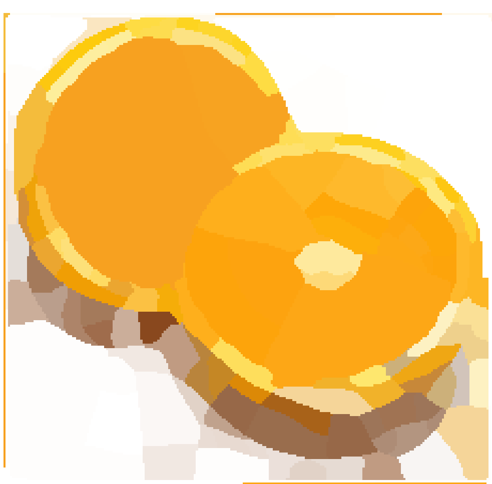

Figure 130: Image `orange` segmented with the default settings and `min_area = 100`

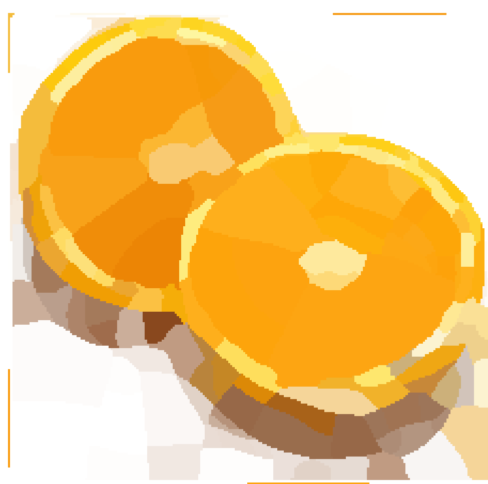

Figure 131: Image `orange` segmented with the default settings and `min_area = 10`

Parameter `ncut_thresh` depends on the degree of derisable segmentation, the spatial complexity and the diversity in colour of each image. Figures 132 - 137 illustrate the effect that the increase in the maximum allowed value for a cut to made has on image `tiger1`. This image has significantly more colour diversity than image `orange`.

Figure 132: Image `tiger1` segmented with the default settings and `ncuts_thresh = 0.01`

Figure 133: Image `tiger1` segmented with the default settings and `ncuts_thresh = 0.02`

Figure 134: Image `tiger1` segmented with the default settings and `ncuts_thresh = 0.05`

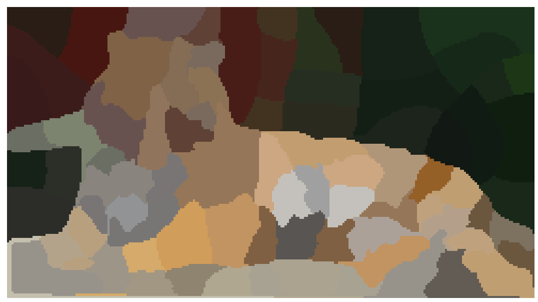

Figure 135: Image `tiger1` segmented with the default settings and `ncuts_thresh = 0.1`

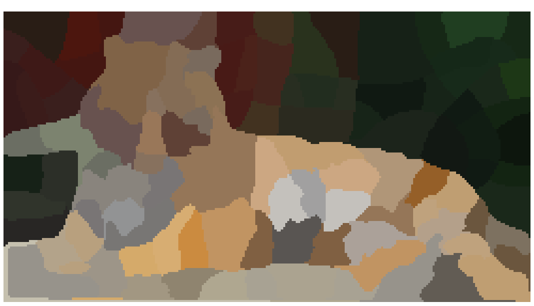

Figure 136: Image `tiger1` segmented with the default settings and `ncuts_thresh = 0.2`

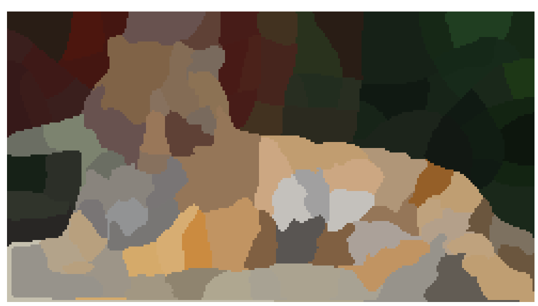

Figure 137: Image `tiger1` segmented with the default settings and `ncuts_thresh = 0.5`

Figures 138 - 143 illustrate the effect that the increase in the recursion depth has on image `tiger3`. Parameter `max_depth`.

Figure 138: Image `tiger3` segmented with the default settings and `max_depth = 1`

Figure 139: Image `tiger3` segmented with the default settings and `max_depth = 2`

Figure 140: Image `tiger3` segmented with the default settings and `max_depth = 4`

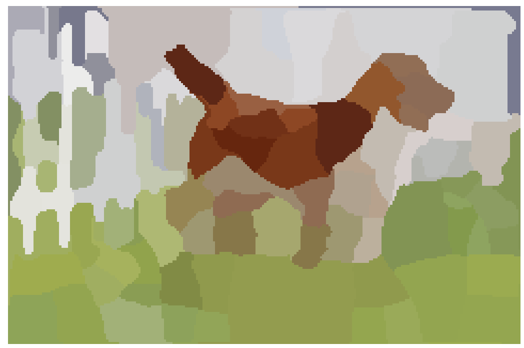

Figure 141: Image `tiger3` segmented with the default settings and `max_depth = 8`

Figure 142: Image `tiger3` segmented with the default settings and `max_depth = 10`

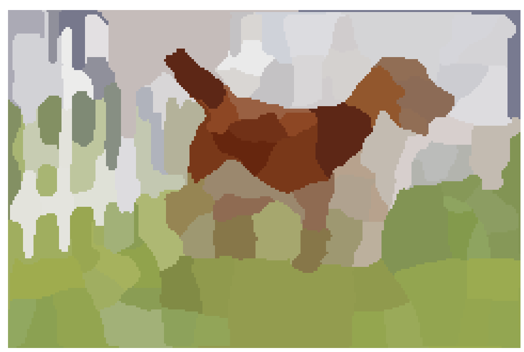

Figure 143: Image `tiger3` segmented with the default settings and `max_depth = 16`

The combination of the three parameters `(min_area, ncuts_thresh, max_depth)` will be different for the 4 images to the extent of the different attributes of each image. For instance figures 144 - 147 show that not only `min_area` has to be small (~ 10) in order for the center of the left-half of the orange to be depicted, but also higher values are needed for `ncuts_thresh` and `max_depth` in order for a more accurate segmentation to take place. The latter can be also said for images `tiger{1,2,3}`, since their colour and spatial diversity is higher than those of image `orange`, and more cuts need to be made in order to depict the little details in the depicted animal's skin.

Figures 144 - 159 illustrate the most reasonably well segmented results of applying the Normal Cut segmentation method to images `orange`, `tiger{1,2,3}`.

3.1.1 Best results - image `orange`

Figure 144: Image `orange` segmented with `(min_area, ncut_thresh, max_depth)` ≡ `(10, 0.2, 10)`.

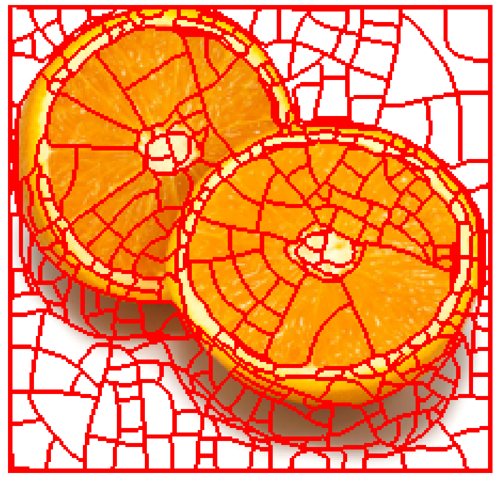

Figure 145: Image `orange` and the bounds of its segments. `(min_area, ncut_thresh, max_depth)` ≡ `(10, 0.2, 10)`.

Figure 146: Image `orange` segmented with `(min_area, ncut_thresh, max_depth)` ≡ `(10, 0.5, 16)`.

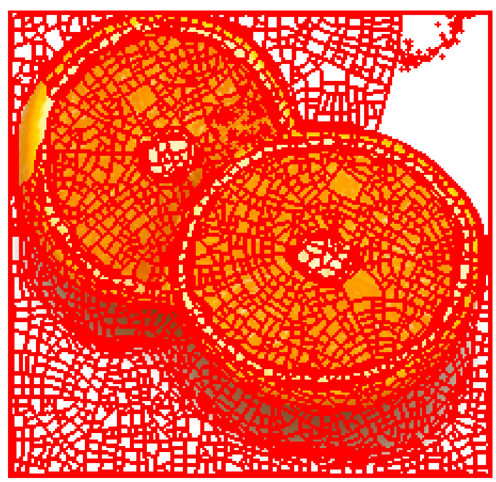

Figure 147: Image `orange` and the bounds of its segments. `(min_area, ncut_thresh, max_depth)` ≡ `(10, 0.5, 16)`.

3.1.2 Image **tiger1**

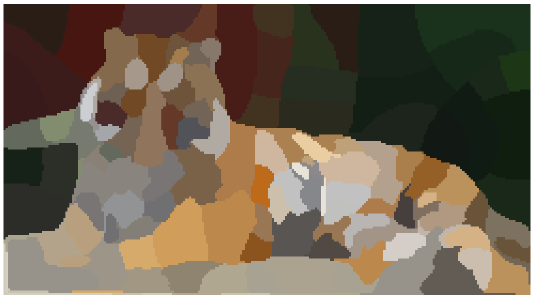

Figure 148: Image `tiger1` segmented with `(min_area, ncut_thresh, max_depth)` ≡ `(10, 0.1, 8)`.

Figure 149: Image `tiger1` and the bounds of its segments. `(min_area, ncut_thresh, max_depth)` ≡ `(10, 0.1, 8)`.

Figure 150: Image `tiger1` segmented with `(min_area, ncut_thresh, max_depth)` ≡ `(10, 0.2, 16)`.

Figure 151: Image `tiger1` and the bounds of its segments. `(min_area, ncut_thresh, max_depth)` ≡ `(10, 0.2, 16)`.

3.1.3 Image **tiger2**

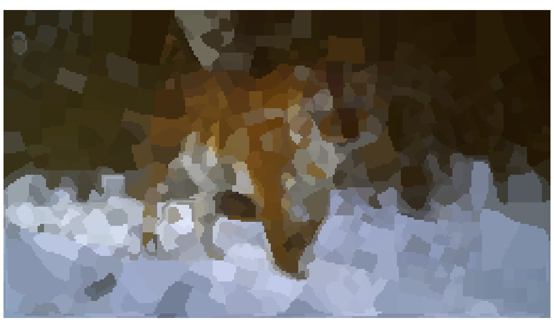

Figure 152: Image `tiger2` segmented with `(min_area, ncut_thresh, max_depth)` ≡ `(10, 0.5, 10)`.

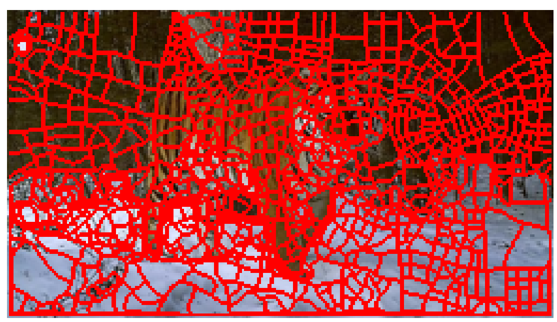

Figure 153: Image `tiger2` and the bounds of its segments. `(min_area, ncut_thresh, max_depth)` ≡ `(10, 0.5, 10)`.

Figure 154: Image **tiger2** segmented with (min_area, ncut_thresh, max_depth) ≡ (10, 0.5, 16).

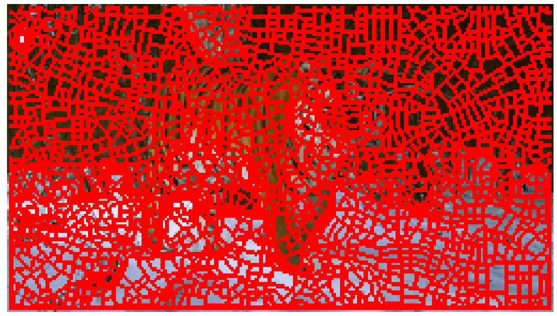

Figure 155: Image **tiger2** and the bounds of its segments. (min_area, ncut_thresh, max_depth) ≡ (10, 0.5, 16).

3.1.4 Image **tiger3**

Figure 156: Image **tiger3** segmented with (min_area, ncut_thresh, max_depth) ≡ (100, 0.1, 8).

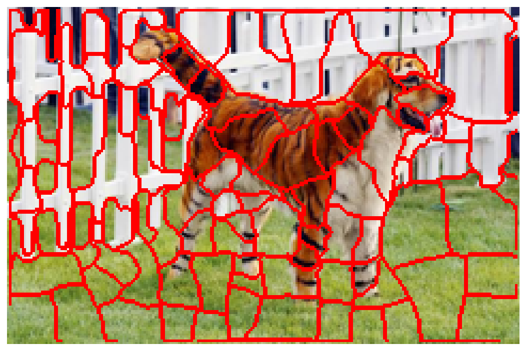

Figure 157: Image **tiger3** and the bounds of its segments. (min_area, ncut_thresh, max_depth) ≡ (100, 0.1, 8).

Figure 158: Image **tiger3** segmented with (min_area, ncut_thresh, max_depth) ≡ (10, 0.5, 10).

Figure 159: Image **tiger3** and the bounds of its segments. (min_area, ncut_thresh, max_depth) ≡ (10, 0.5, 10).

3.2 Question 8

The two parameters that were able to reduce the number of segments while at the same time keeping a reasonable segmentation accuracy were both the `ncuts_thresh` and `max_depth` parameters as it can be seen in figures 132 - 143. Furthermore, parameter `min_area` can also affect the subdivision when increased, although not in the same capacity as the former two parameters.

3.3 Question 9

Since

$$Ncut(A,B) = cut(A,B)\Big(\frac{1}{assoc(A,V)} + \frac{1}{assoc(B,V)}\Big) \tag{1}$$

if $assoc(V)$ represents the total number of edges in the graph, then

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

since $cut(A,B)$ is included in each $assoc(*,V)$, and equation 1 is then formed as

$$Ncut(A,B) = cut(A,B)\Big(\frac{1}{assoc(A,V)} + \frac{1}{assoc(V) - cut(A,B) - assoc(A,V)}\Big) \tag{3}$$

In equation 3, we consider all variables except $assoc(A,V)$ to be constants. Then, minimizing equation 3 means finding the value of $assoc(A,V)$ that satisfies equation 4:

$$\frac{d}{dassoc(A,V)}Ncut(A,B) = 0 \tag{4}$$

The derivative of $Ncut(A,B)$ with respect to $assoc(A,V)$, $\frac{d}{dassoc(A,V)}Ncut(A,B)$ is

$$\begin{gathered}\frac{d}{dassoc(A,V)}Ncut(A,B) = -\frac{cut(A,B)}{assoc(A,V)^2} + \frac{cut(A,B)}{(assoc(V)+cut(A,B)-assoc(A,V))^2} = \\ \frac{cut(A,B)(assoc(A,V)^2 - (assoc(A,V) - assoc(V) - cut(A,B))^2)}{(assoc(A,V)(assoc(A,V) - assoc(V) - cut(A,B))^2)} = \\ \frac{cut(A,B)(-2assoc(A,V)(assoc(V)+cut(A,B)) + (assoc(V)+cut(A,B))^2)}{(assoc(A,V)(assoc(A,V) - assoc(V) - cut(A,B))^2)} = \\ \frac{cut(A,B)(assoc(V)+cut(A,B))(-2assoc(A,V) + (assoc(V)+cut(A,B)))}{(assoc(A,V)(assoc(A,V) - assoc(V) - cut(A,B))^2)}\end{gathered} \tag{5}$$

Hence, if equation 4 is to be satisfied

$$assoc(A,V) = \frac{assoc(V) + cut(A,B)}{2} \tag{6}$$

and from equation 2

$$assoc(B,V) = \frac{assoc(V) + cut(A,B)}{2} = assoc(A,V) \tag{7}$$

3.4 Question 10

4 Segmentation using graph cuts