

ECE 1395 – Homework 9 – report

PS: The pictures are saved in folder “output” as follow: ImagNumber_Kvalue_IterationsNbre_Rvalue

For example, im1_3_7_5 is the output for image 1 when K=3, iters=7 and R=5

Image 1

Experimenting with the first image (panda image), I calculated the SSD for different combinations of K, iters and R. As you can see below, for k=3, increasing the number of iterations and/or the number of restarts R only decreases SSD by a little bit. The drastic change in SSD happens when the number of clusters K changes. In fact, when I changed K from 3 to 5, SSD dropped to 16642 which for iters =7 and R = 5, and to 11852 when iters=7 and R=5.

```
img = cv2.imread("im1.jpg")
```

```
Segment_kmeans(img,3,7,5)
```

```
Segment_kmeans(img,3,7,15)
```

```
Segment_kmeans(img,3,13,5)
```

```
Segment_kmeans(img,3,20,5)
```

```
Segment_kmeans(img,5,7,5)
```

```
Segment_kmeans(img,7,7,5)
```

```
SSD: 41035.87967332427
```

```
SSD: 41030.04074951011
```

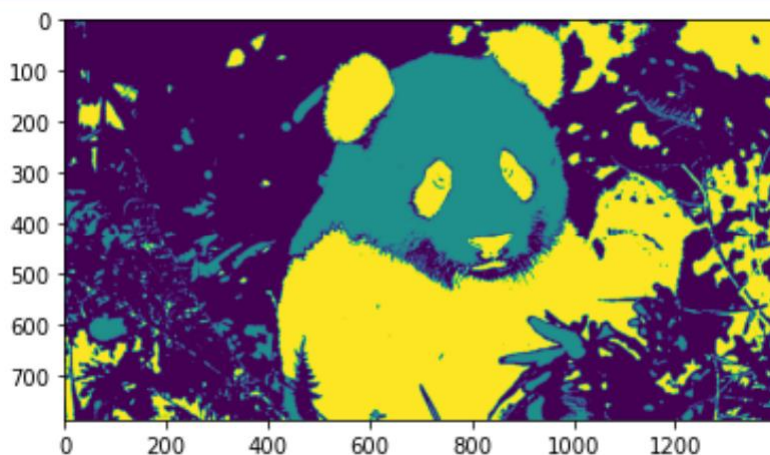
```
SSD: 41029.77559476969
```

```
SSD: 41029.774509674244
```

```
SSD: 16642.498579121588
```

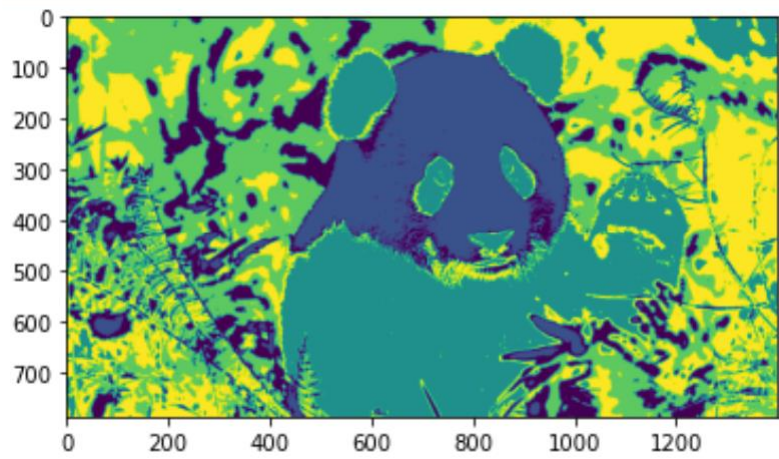
```
SSD: 11852.565691861595
```

Here are the pictures obtained when K=3,iters=7,R=5 and K=5,iters=13,R=15 and K=7,iters=20,R=30 respectively

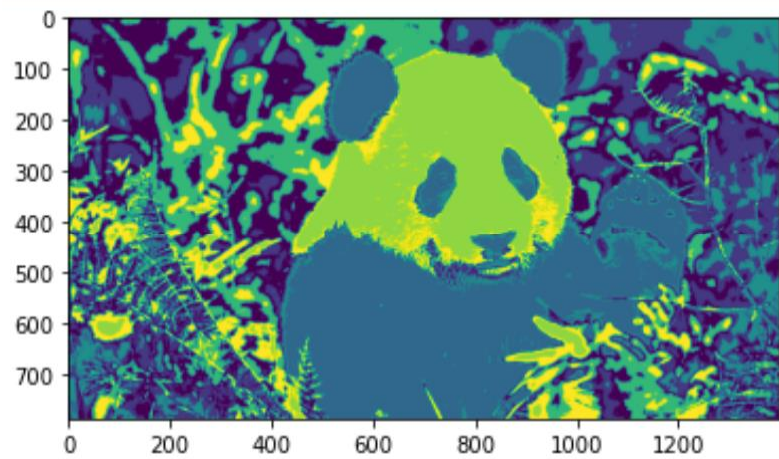


Im1_3_7_5.png

Rayan Hassan
4511021



Im1_5_13_15.png

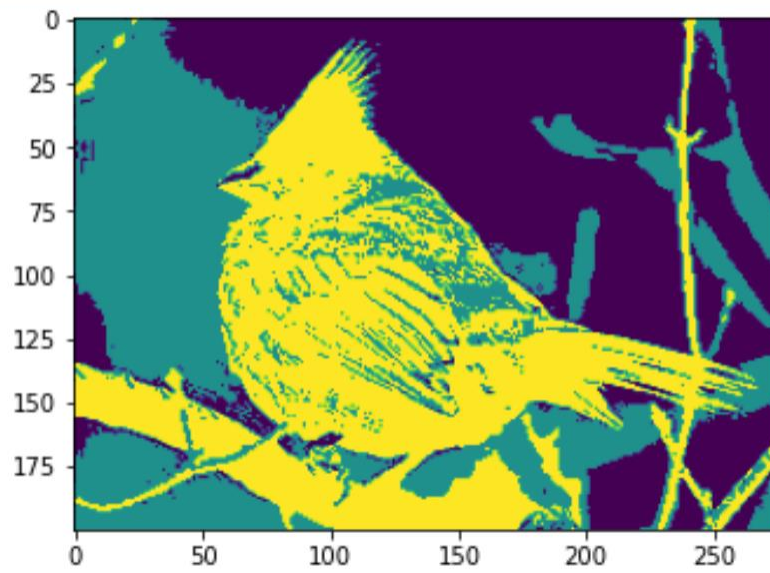


Im1_7_20_30.png

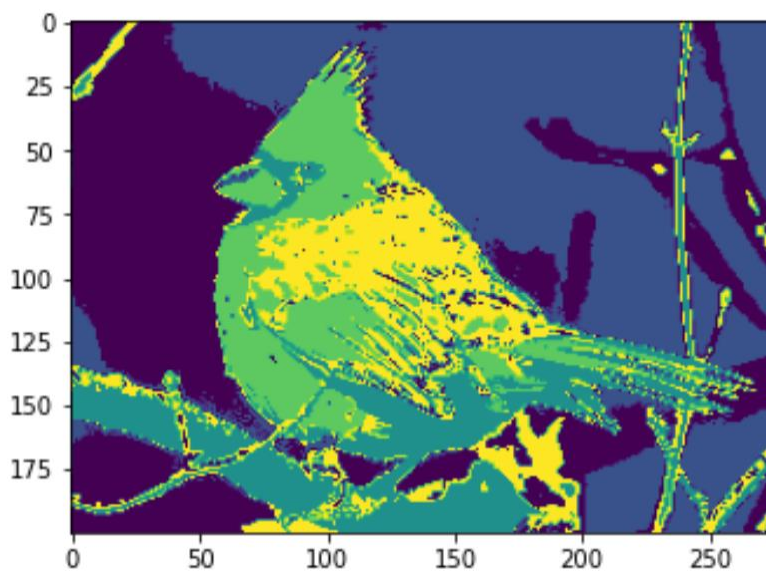
Their corresponding (and respective) SSD values are as follow:

```
Reloaded modules: kmeans_single, kmeans_multiple  
SSD: 41048.30088083433  
SSD: 16637.78844678351  
SSD: 11712.412594417929
```

Image 2

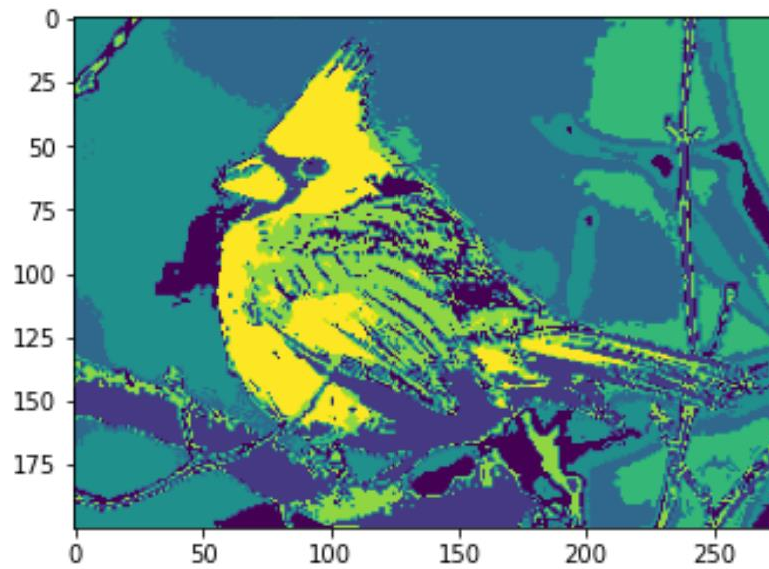


Im2_3_7_5.png



Im2_5_13_15.png

Rayan Hassan
4511021



lm2_7_20_30.png

Their respective SSD values are shown below

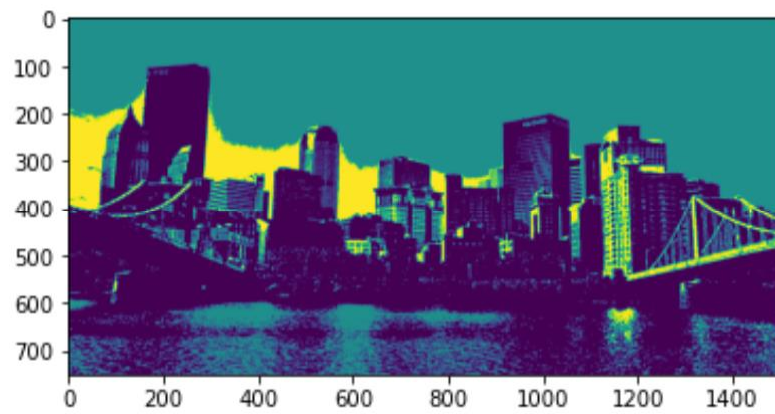
```
Reloaded modules: kmeans_single, kmeans_multiple  
SSD: 2069.0331000593856
```

```
Figures now render in the Plots pane by default.  
also appear inline in the Console, uncheck "Mute  
under the Plots pane options menu.
```

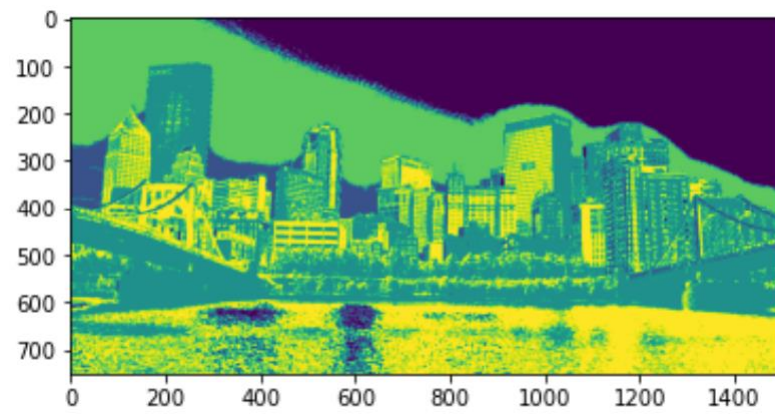
```
SSD: 888.5240007258361  
SSD: 635.6336425611368
```

Rayan Hassan
4511021

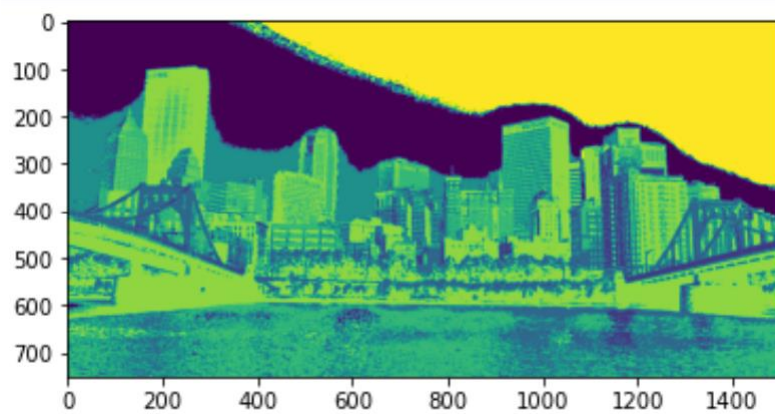
Image 3



Im3_3_5_7.png



Im3_5_13_15.png



Im3_7_20_30.png

Rayan Hassan
4511021

Their respective SSD values are the following

```
Reloaded modules: kmeans_single, kmeans_multiple
SSD: 52353.26761800543
SSD: 31624.335113159203
C:\Users\RAYAN\Anac3\lib\site-packages\numpy\core\fr
3419: RuntimeWarning: Mean of empty slice.
    return _methods._mean(a, axis=axis, dtype=dtype,
SSD: 22134.973031842615
```

Conclusion

As expected, when number of clusters k is the lowest (3), the clusters are too large and lack specificity. As the number of clusters increases, the clusters become smaller and represent more details in the picture. However, if k is too large, we can have overfitting and poor generalization performance.

On the other hand, increasing the number of iterations allows multiple updates for the cluster assignments and centroids, which slightly decreases the SSD value. If number of iterations is too high however, we waste computational time without any significant improvement in the clustering performance (SSD barely changes).

Finally, increasing the number of restarts R allow for different random assignment for initial centroids, which helps the chances to converge to the global minima instead of a local one. Just like the number of iterations, increasing R to a very high value wastes time and doesn't necessarily improve performance significantly.