CS216 HW3 Name: Yifan Tian ID:78921267

1. Color quantization

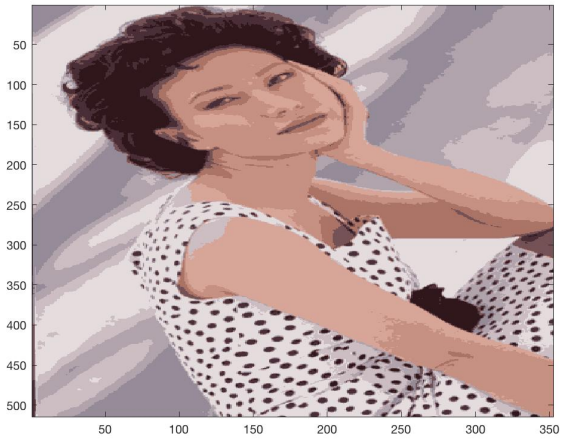
The way to call Colorquantization() function:

```
image = imread('Lady.png');
```
imagesc(Colorquantization(image,10))

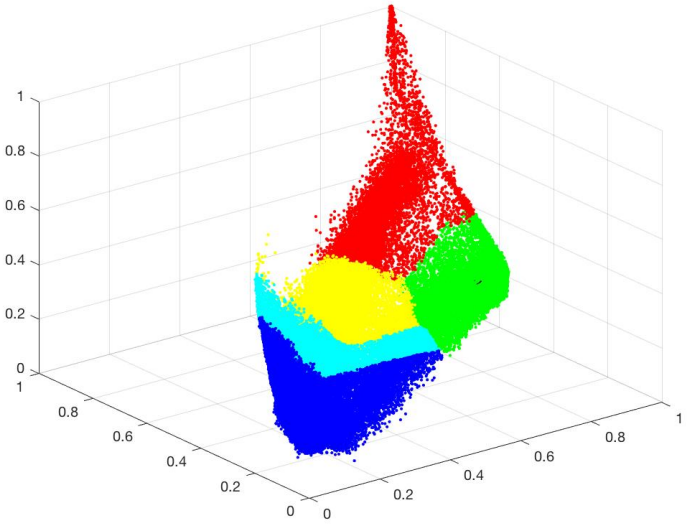
The original picture:

K = 2,

k = 5



k =10

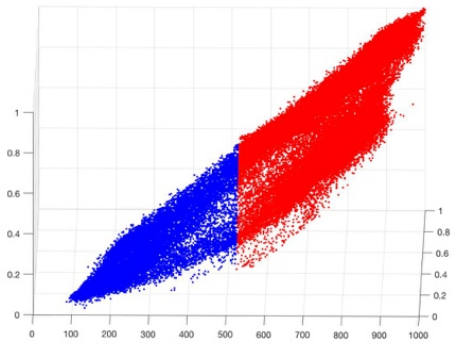


The clusters in the 3d color dimension.

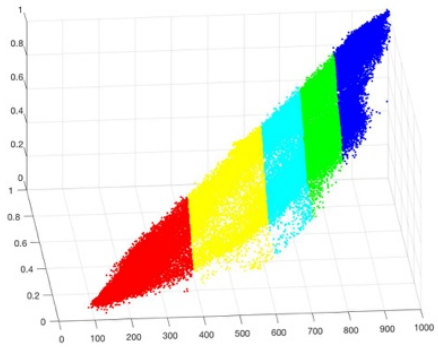
Scaling one of the feature coordinates

When we rescale one color, then it will change the distribution of dots thus change the distribution of clusters. Which will yield a different result.

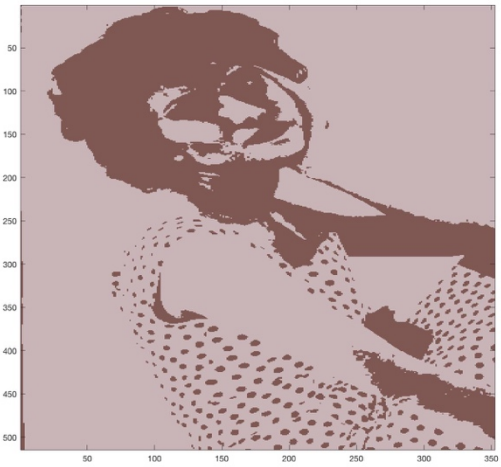
Rescale on red color:
k = 2

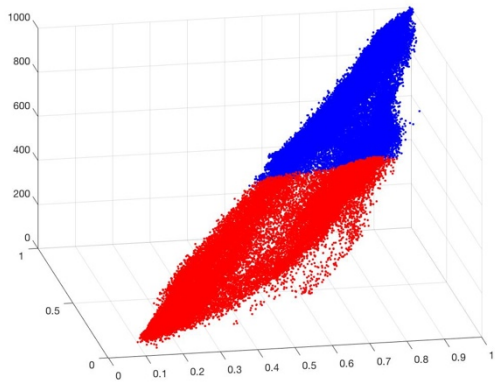


k = 5

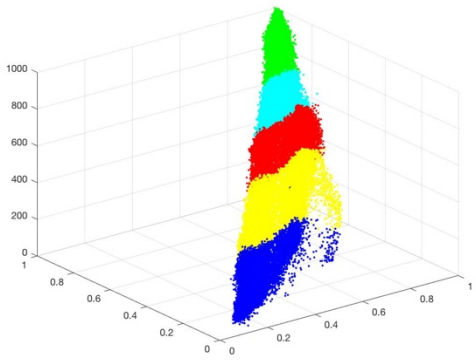


Rescale on blue color:





k = 2



k = 5

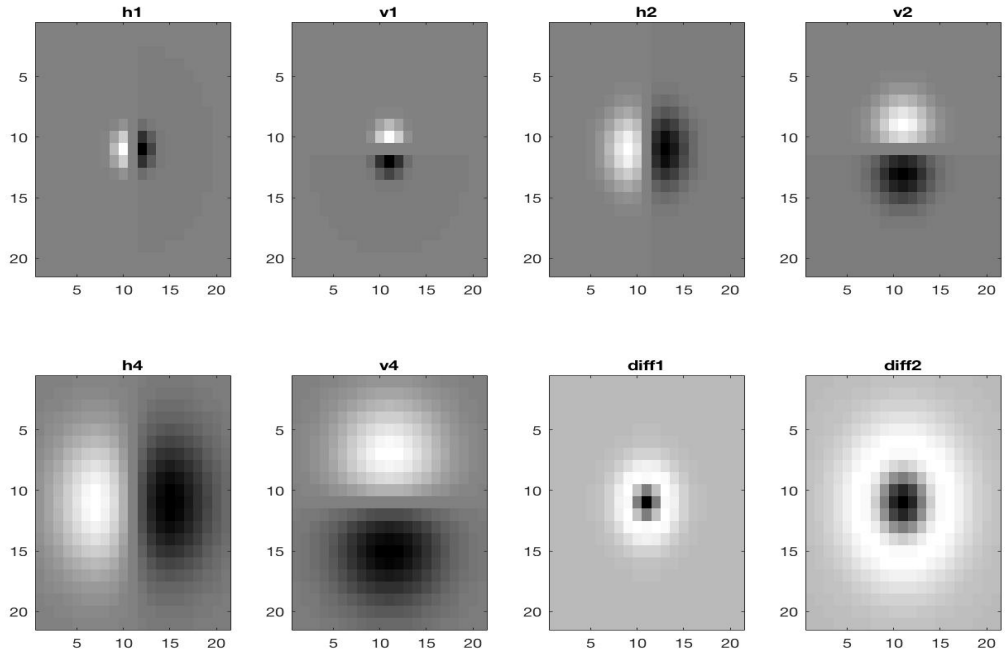
So if we rescale one color (times 1000 then divide 1000) it will not change the original color drastically but will change the distribution. Basically all clusters now will distribute according to the color you choose to rescale.

2. FilterBank

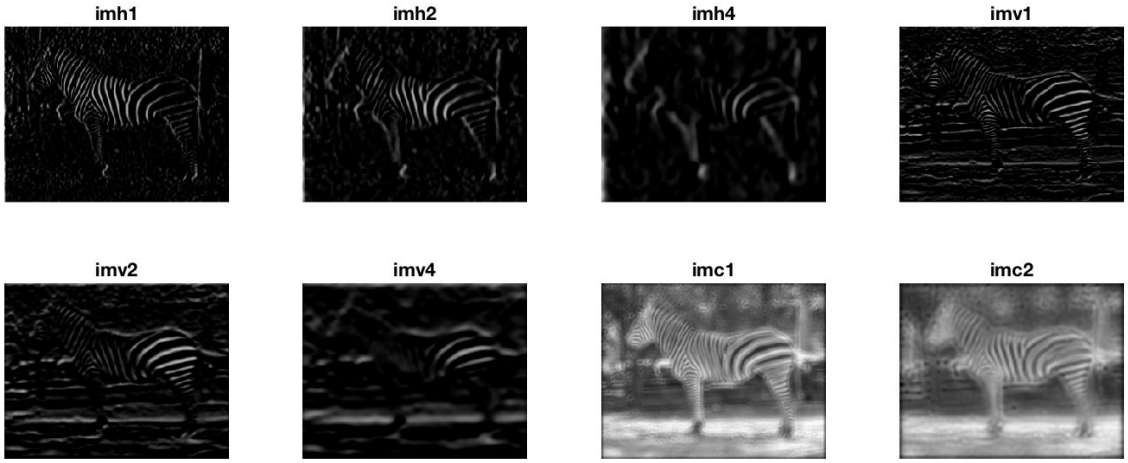
The way to call FilterBank() function:

image = imread('Lady.png');
filterbank(image)

The different Gaussian filters including derivative filter and difference filter.



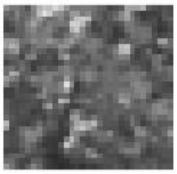
The image of ‘samll zebra’ after filters processing:



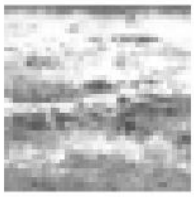
3. Filter Distribution

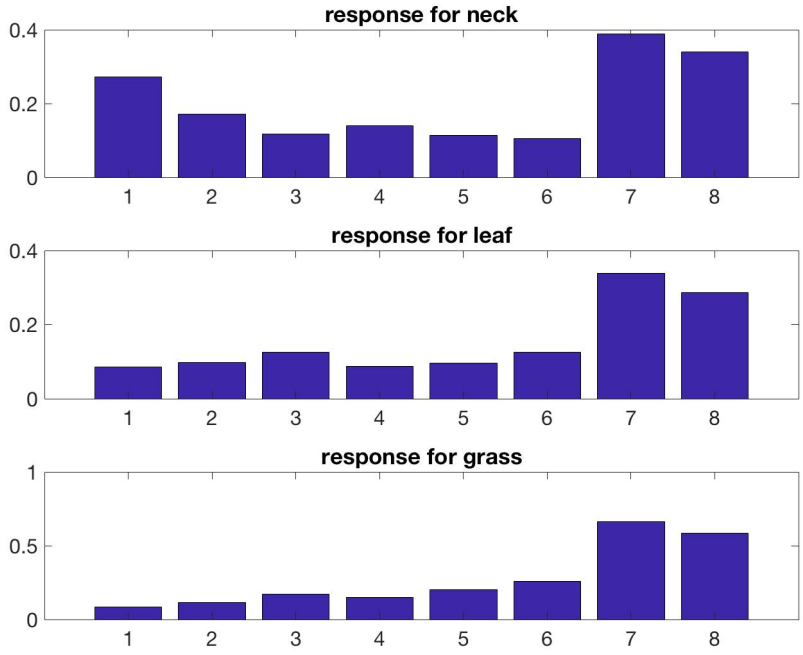
The pack on neck:

The pack on leaves:



The pack on grass:

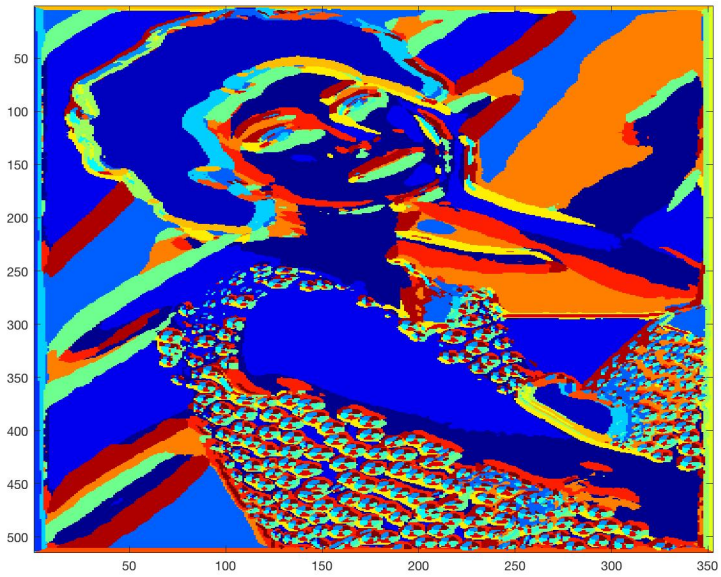




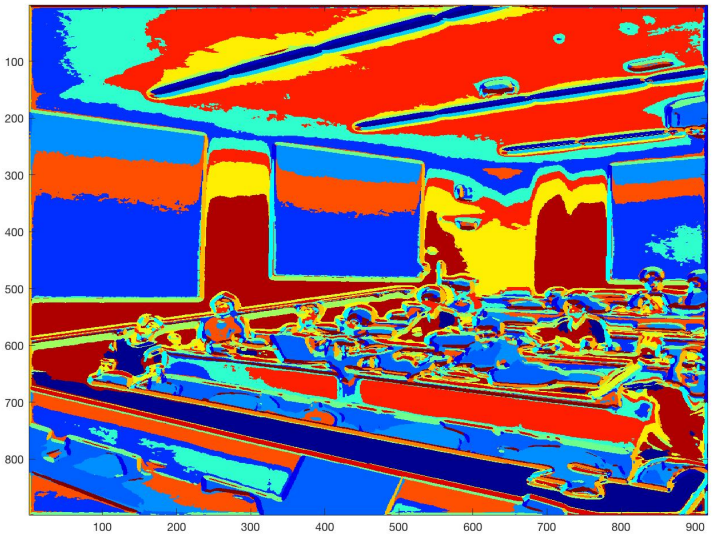
The 1-3 correspond horizontal derivative filters in different scale, the 4-6 correspond vertical derivative filters in different scale. We can clearly see that for neck, there is a strong response for horizontal edges, while for leaf, edges are isotropic, so response for horizontal and vertical filters are almost same. For grass, because the light pattern is vertical, so we see a stronger response for vertical filters.

4. Textons

k = 20 textons for image: 



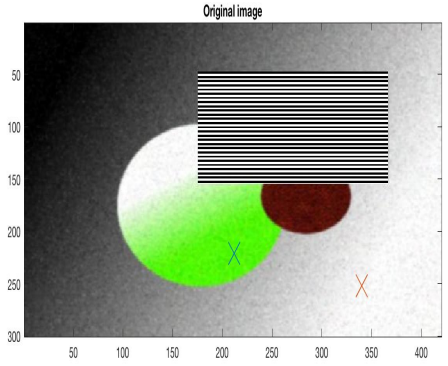
k = 20 textons for image: 

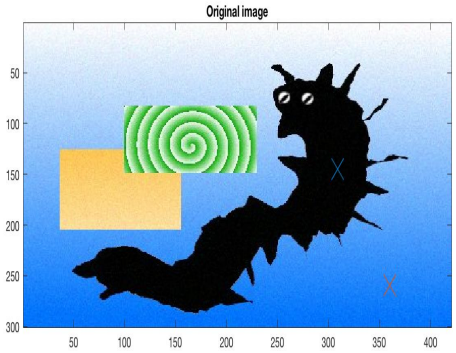


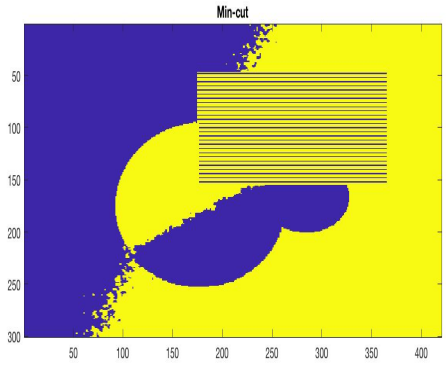
5. MRF Segmentation.

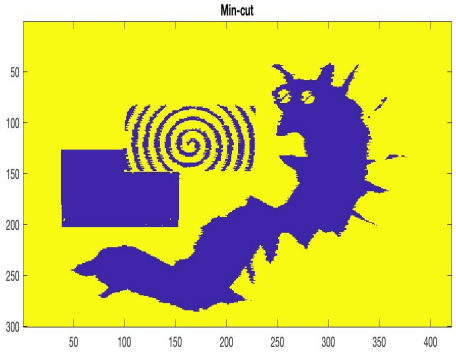
1. Use front and back colors.

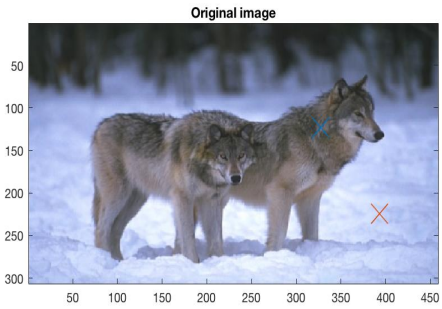
The code for this part is "hw3_demo.m". In each plot, I use '*' to indicate where I seed the points 's' and 't'.

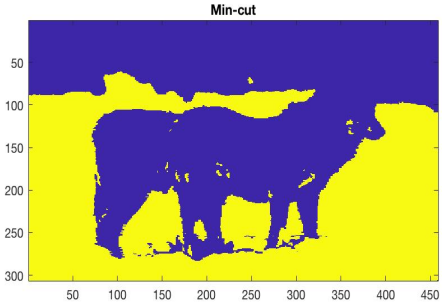

Original image
50
100
150
200
250
300
50
100
150
200
250
300
350
400


Original image
50
100
150
200
250
300
50
100
150
200
250
300
350
400


Min-cut
50
100
150
200
250
300
50
100
150
200
250
300
350
400

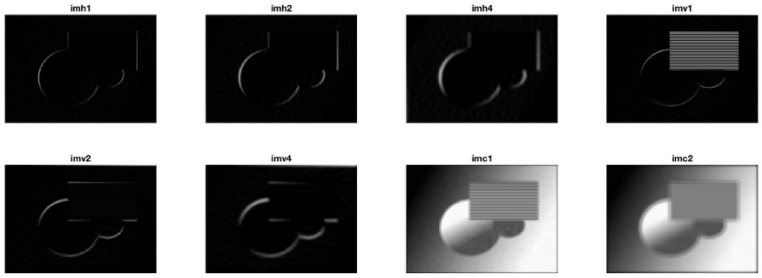

Min-cut
50
100
150
200
250
300
50
100
150
200
250
300
350
400

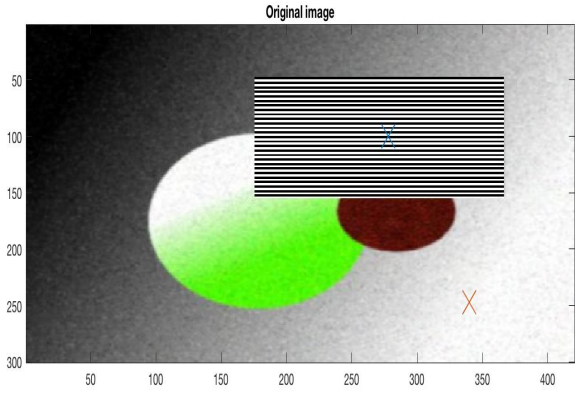

Original image
50
100
150
200
250
300
50
100
150
200
250
300
350
400
450

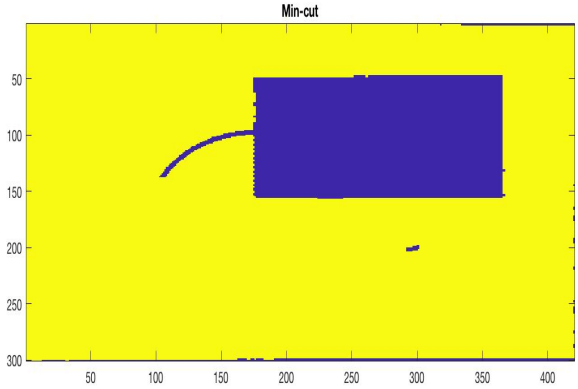

Min-cut
50
100
150
200
250
300
50
100
150
200
250
300
350
400
450

2. Use histogram of filters.
The code for this part is “hw3_demo_histogram.m”

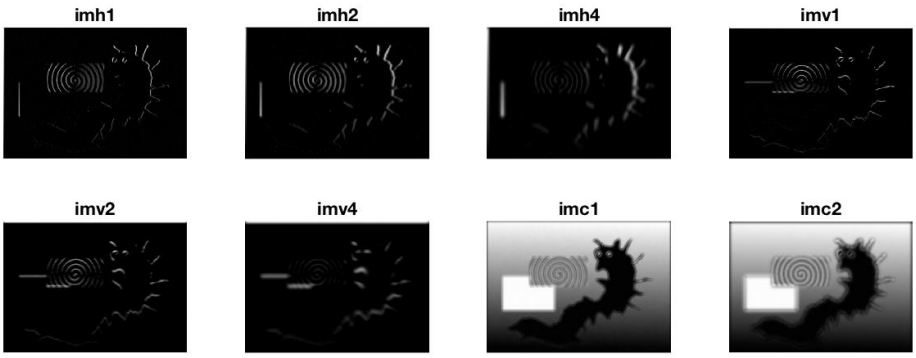
Here I want to use histogram of filters to search for feature pattern to implement cutting.

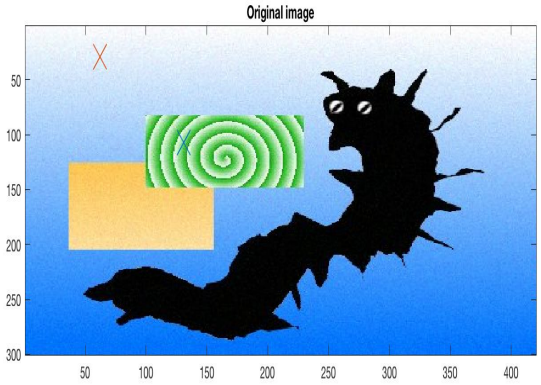


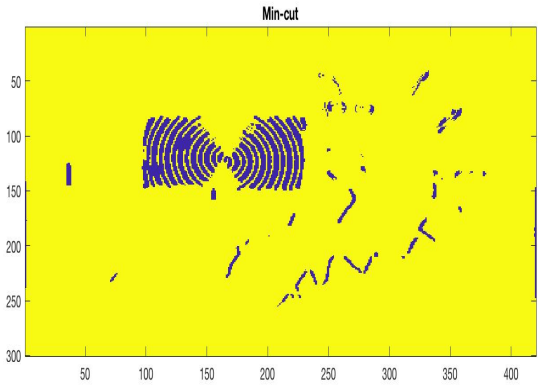




For above image, I want to segmentation of the rectangular with horizontal lines. We can see that it introduce high response for vertical derivative filters.







For above image, I want to segmentation of the rectangular with spiral lines. We can see that it can introduce high response for derivative filters.

6. Project idea:

Basically I want to explore and repeat part of the work of paper:" Low-rank Bilinear Pooling for Fine-Grained Classification**".**

The points that I want to learn and implement:

1. Use vgg16 to construct a bilinear multiclass-classifier using SVM.
2. Use SVD to show that the features have a low rank decomposition.
3. Use gradient descent to optimize the decomposition.