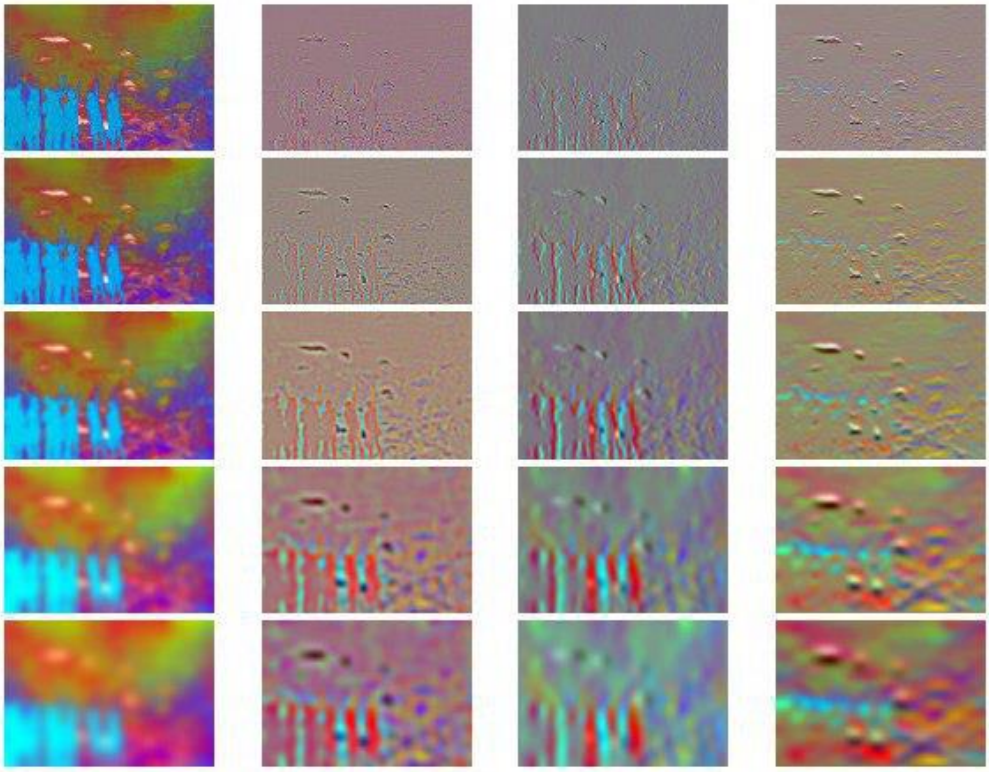
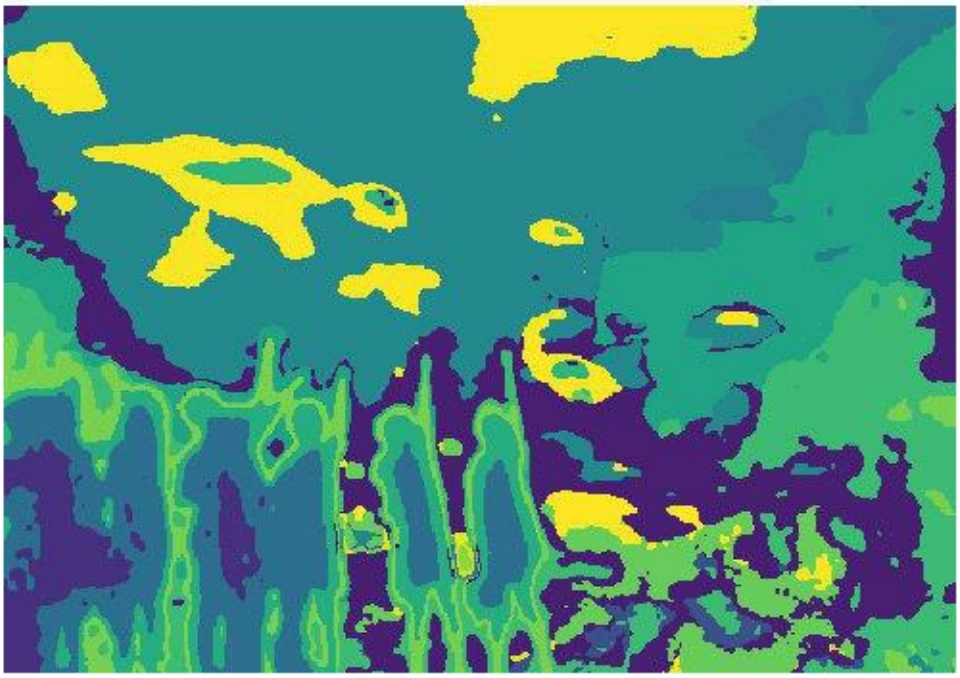
Q1.1.1

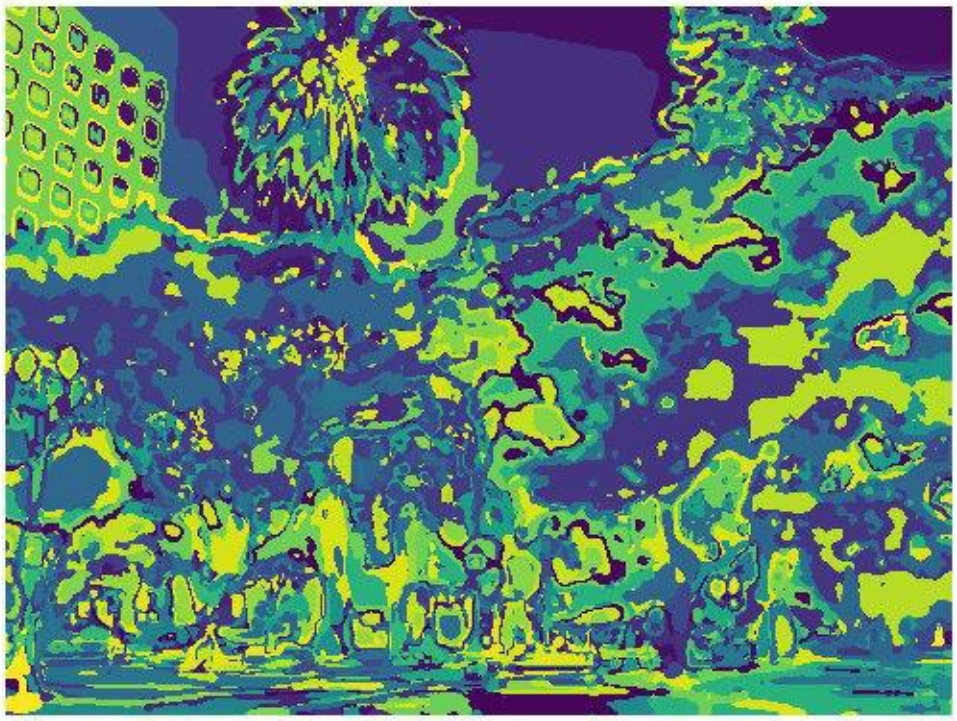
The Gaussian filter is used for smoothing images and removing the noise. Laplacian filter is used for edge detection. Since it is based on derivation, it is sensitive to noise so before applying Laplacian the image needs to be smoothed first by Gaussian filter which results in Laplacian of Gaussian. Derivation of Gaussian in the x direction detects vertical lines and in the y direction detects horizontal lines. The bigger the scale of pixel units become, it blends a larger area of pixels together which causes for the image to lose more information and become blurrier. We are using different scales of filters since dependent on the image, features are presented in different scales(some are more zoomed in) and also to get different ranges of detailed information from the images based on filter size.

Q1.1.2 Visualization of filter responses



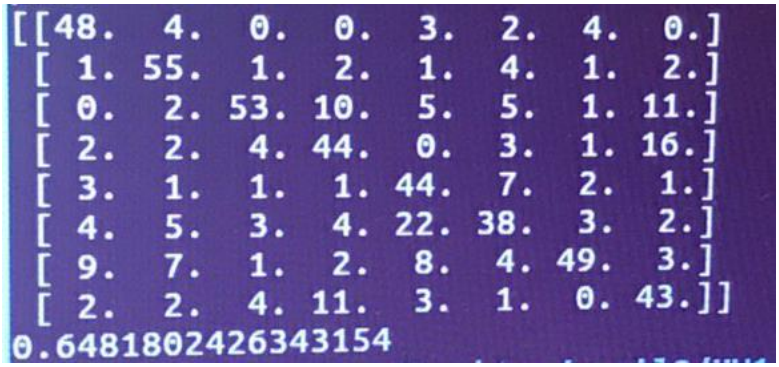
Q1.3 Visualization of wordmaps





Based on the pictures above, we can see that based on the computed dictionary and assigning words to each of the pixels, the image is converted to wordmaps containing representation of the shapes in the image and also is based on color information in the original RGB image which means the areas with similar range of colors get close intensity of color range.

Q 2.5 (Confusion Matrix and Accuracy)



Confusion Matrix:

```
[[ 48   4   0   0   3   2   4   0]
 [  1  55   1   2   1   4   1   2]
 [  0   2  53  10   5   5   1  11]
 [  2   2   4  44   0   3   1  16]
 [  3   1   1   1  44   7   2   1]
 [  4   5   3   4  22  38   3   2]
 [  9   7   1   2   8   4  49   3]
 [  2   2   4  11   3   1   0  43]]
```

Accuracy: 64.81 %

Q2.6

The classification procedure is affected by different things.
One is the input image in test set that we actually want to classify. If the quality of the test image is not good and is blurred too much, the extracted features would not contain valuable information and it is more likely for that image to be confused by other images. This is true for the image below which is classified mistakenly as aquarium.

The other thing that also might affect the classification procedure is the color range. In the below picture it can be seen that the kitchen is classified as desert because of the wide brown color range in the image.

Another thing that also causes miss classification is that some of the images have basically the same feature spaces and contain very mutual information. As an example, the image below is presenting a park but is miss classified as highway because of the road shaped region in the middle

Also it can be said that since the feature extraction is done randomly, some valuable features for distinguishing between the images might be lost. So if there are similar features in different types of images, they can be classified wrong.
Based on the derived confusion matrix, it can be seen that the algorithm is not very good at distinguishing between kitchen and laundromat and about 38 images of kitchen images have been classified wrongly as laundromat. This shows the high similarity between derived features of these two types of images.

Extra Credit
When I ran my code for the first time since my functions had for loops, it used to run so slow. I tried to vectorize all of my functions by going through the documentations of the used command in pyhton which resulted in very faster operation.