1.1.1

Gaussian Filter – It smoothens (blurs) an image, removing high frequencies and noise from an image

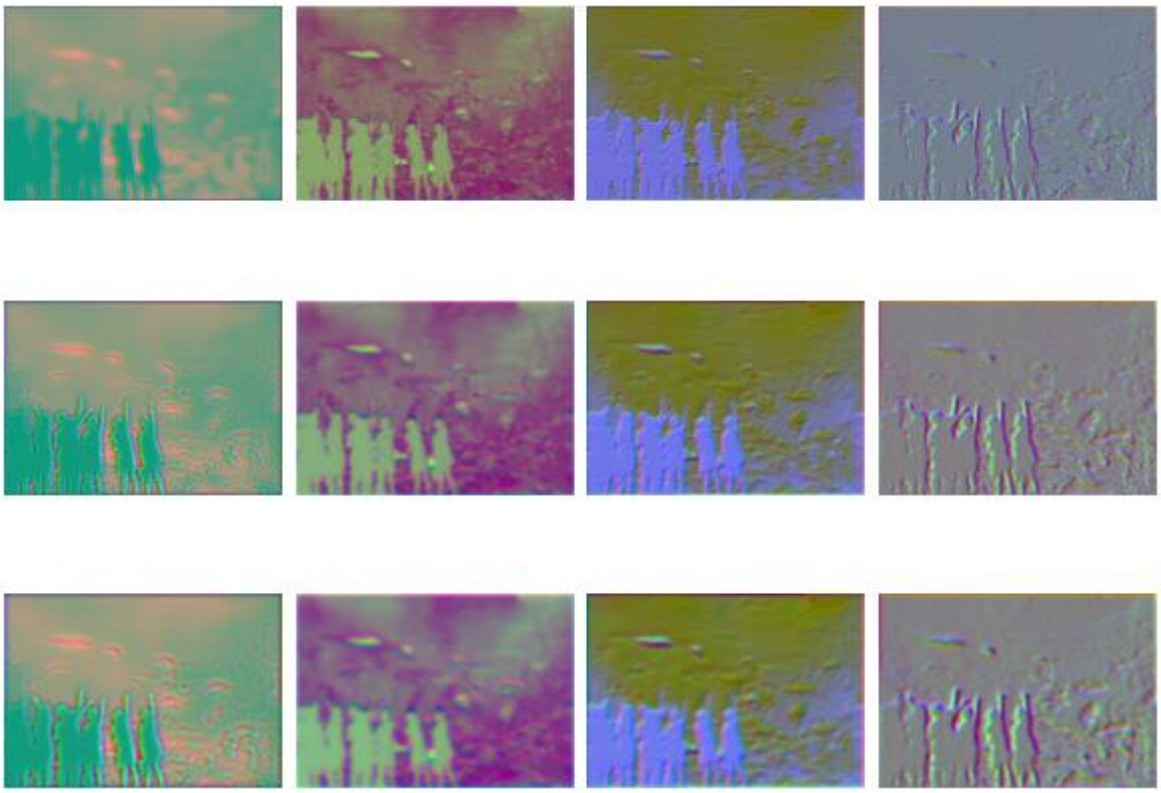
Laplacian of Gaussian Filter - It combines the Gaussian smoothing and Laplacian operator to highlight regions of rapid intensity changes in an image.

Gaussian x-derivative: Detects vertical edges or changes in intensity in the x-direction of the image.

Gaussian y-derivative: Detects horizontal edges or changes in intensity in the y-direction of the image.

Use of Multiple scales: Different scales of the same image contain different types of information. Some features are picked up by the filter bank in smaller scales and some in larger scales. For instance in edge detection, an edge far away in the scene can be detected by smaller kernel (small sigma). Opposite is true for the larger edges.

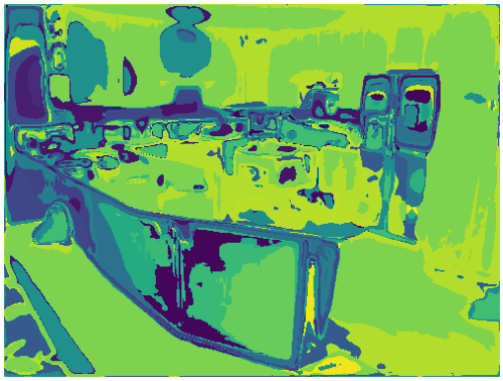
1.1.2

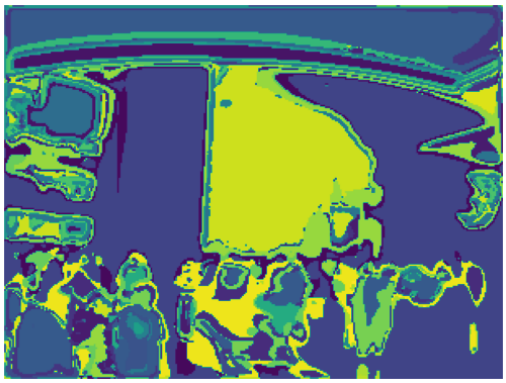


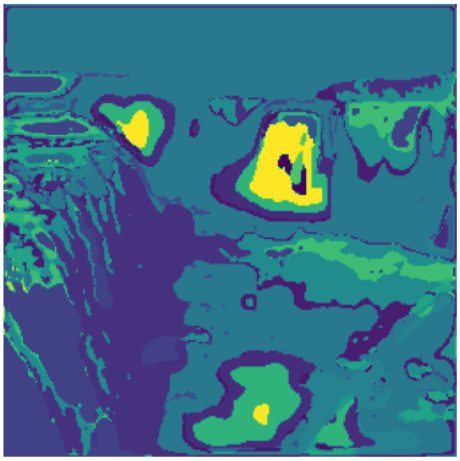
1.2 See Code

1.3

From the images shown below juxtaposed with their wordmaps (or texton maps), it can be noted that more distinct boundaries form in the presence of distinct color contrasts. Strong edges in the image are captured very well in the wordmaps, which is a very good indicator of our model (given the dictionary formation was done through random alpha pixel sampling). But in the case of waterfall example image, the details in the wordmap are not very distinct (visually atleast).





2.1 See Code

2.2 See code

2.3 See code

2.4 See code

2.5 Used default hyperparameters

Confusion matrix:

```
[[29  0  4  6  2  5  5  1]
 [ 5 26  4  2  5  2  1  5]
 [ 5  9 25  2  2  5  2  7]
 [ 0  4  2 16 16  1  4  0]
 [ 3  2  1 18 17  6 10  3]
 [ 0  4  1  1  5 27  6  4]
 [ 5  1  2  4  1  2 21  0]
 [ 3  4 11  1  2  2  1 30]]
```

Accuracy = 47.75%

2.6 My system is getting confused a lot while classifying class 4 and class 3. That is, it's classifying laundromat as kitchen and vice-versa in a lot of cases. I think it might be due to similar-looking indoor setting – flooring, boxy structures for cabinets and washing machines etc.

It's also getting confused between the classes highway and windmill. This confusion might be due to not taking the perspective depth into account. For an unbiased system, a windmill outer concrete structure is basically a cemented region wider at the bottom of the image and narrower at the top, which is the same for a typical highway picture.

Also, random sampling of alpha pixels might have ended up in image regions that have very less information for the classisfier.

3.1

S.No	Filter_Scales	K	Alpha	L	Accuracy (%)
1	[1,2,3]	10	20	2	47.75
2	[1,2,3,4]	60	80	2	56.75
3	[1,2,3,4,5]	80	160	2	61.25

Row 1 contains the default parameters

Effect of hyperparameters tuning:

Increase in K: As K increases, the number of clusters also increases in the k means model. That means, more features are captured in the model for it to refer back to for comparison.

Increase in alpha: A greater number of sampled pixels directly corresponds to more information being collected about an image.

Increase in filter_scales: Various scales of filter responses aid in capturing all the multi-sized features

3.2 Improved Accuracy: 63%

I tweaked the weights in spatial matching for each layer a little bit to yield an accuracy of 63%. I kept the same hyperparameters as before. For L=2 (Layers -0,1,2), the weights before were ¼, ¼, ½ respectively. I halved the weights for each layer. So the new weights used become 1/8, 1/8, ¼ respectively.

The reason for applying the same factor of reduction in weights for each layer is to keep the proportion of the inference of information about features from the images to the model similar as was in the original model.

As mentioned above, the overall accuracy of the model improved to 63%.