Computer Vision, 16720A - Homework 1

Neeraj Basu
neerajb@andrew.cmu.edu

September 23, 2020

1 Q 1.1.1 - Extracting Filter Responses

What properties do each of the filter functions pick up? Why do we need multiple scales of filter responses?

Gaussian: Acts like a low-pass filter to remove any high frequencies inside of an image. The end result is a blurred image.

Laplacian of Gaussian: Detects sudden intensity transitions in a pixel and determines where vertical and horizontal edges live.

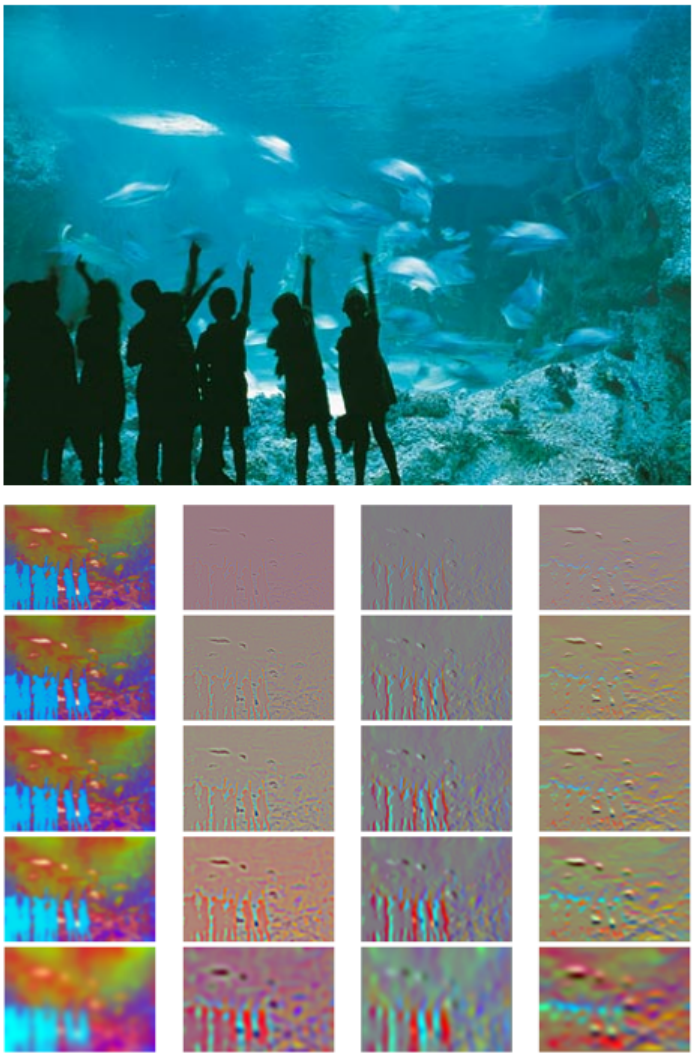
Derivative of Gaussian - x direction: Detects sudden intensity transitions in a pixel and determines where vertical edges live.

Derivative of Gaussian - y direction: Detects sudden intensity transitions in a pixel and determines where horizontal edges live

The reason for using multiple scales when creating our filter responses is so we can analyze the image with different pass-bands. Each sigma will carry information about the image at a particular frequency band. When we sub-sample pixels across our filter responses, we get more variety of the image data therefore improving our ability to classify.

2 Q 1.1.2 - Extracting Filter Responses

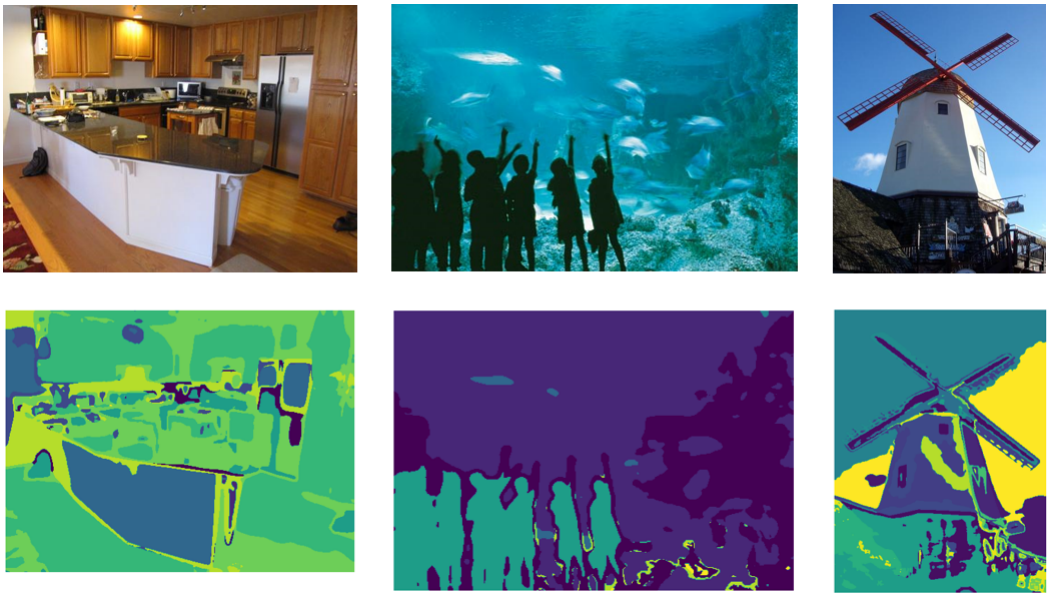
Apply all 4 filters at least 3 scales on aquarium/sun aztvjgubyrgvirup.jpg, and visualize the responses as an image collage. Submit the collage of images in your write-up.



Four filters applied over five filter scales [1,2,3,5,10] for aquarium/sun aztvjgubyrgvirup.jpg.

3 Q 1.3 - Visualization of Wordmaps

Visualize wordmaps for three images. Include these in your write-up, along with the original RGB images. Include some comments on these visualizations: do the "word" boundaries make sense?



kitchen/sun_aasmevtpkslccptd.jpg
aquarium/sun_aztvjgubyrgvirup.jpg
windmill/sun_bdmdzedcanjjntkv.jpg

Overall, the word maps above make sense. These word maps represent the fact that similar features/texture are classified as the same visual word. A great example is the representation of the children in aquarium/sunaztvjgubyrgvirup.jpg. The image data in the pixels representing the children should be very similiar and therefore it makes sense they were group together.

4 Q 2.5 - Quantitative Evaluation

Include the confusion matrix and your overall accuracy in your write-up.

Confusion Matrix:

```
27 1 4 0 2 1 9 6
1 25 4 6 7 0 5 2
0 8 21 2 2 1 7 9
2 4 2 27 11 2 2 0
1 2 4 13 18 5 6 1
3 0 6 0 3 31 5 2
10 1 2 2 7 6 18 4
2 2 6 0 3 5 10 22
```

Accuracy:

47.2%

Parameters:

Filter Scales: [1,2]
Alpha: 25
K = 10
L = 1

5 Q 2.6 - Find the Failures

There are some classes/samples that are more difficult to classify than the rest using the bags-of-words approach.In your writeup, list some of these hard classes/samples, and discuss why they are more difficult than the rest.

class	aquarium	desert	highway	kitchen	laundromat	park	waterfall	windmill
# correct	357	263	264	347	230	331	239	239

Class vs. # of correct classifications over 10 runs

The chart above shows the sum of the diagonals over my 10 runs to complete my ablation table, summed together. In other words, aquarium was classified properly 357 times, desert 263 times, etc. The two pairs of classes that were most commonly confused according to the confusion matrix was:

1. kitchens mistaken for laundromats
2. highways mistaken for windmills

It looks like there were visual words which could have been captured in both collections of scene data. From a classification point of view, a dishwasher could look similar to a washer/dryer. Or the images containing windmills, could primarily be composed of a green grass horizon and a blue sky, similar to the images of highways. Therefore, it's explainable why the classifier might have mislabeled these.

6 Q 3.1 - Hyperparameter Tuning

Tune the system you build to reach around 65% accuracy on the provided test set (data/test files.txt). Include a table of ablation study containing at least 3 major steps (changing parameter X to Y achieves accuracy Z%). Also, describe why you think changing a particular parameter should increase or decrease the overall performance in the table you show.

Ablation Table:

Run	Scales	K	Alpha	L	Conf
0	[1,2]	10	25	1	47.25%
1	**[1,2,3,4,5]**	10	25	1	47.25%
2	**[1,2,3,5,10]**	10	25	1	46.00%
3	[1,2,3,5,10]	**25**	25	1	53.00%
4	[1,2,3,5,10]	**30**	25	1	59.50%
5	[1,2,3,5,10]	30	**30**	1	58.25%
6	[1,2,3,5,10]	30	30	**2**	63.00%
7	[1,2,3,5,10]	30	30	**3**	64.00%
8	[1,2,3,5,10]	30	30	3	63.00%
9	[1,2,3,5,10]	30	**35**	3	65.75%

filter_scales: I was expecting an increase in the number of sigmas to have a positive impact but as it turns out it played a small role in increasing accuracy. In fact in some trials it decreased accuracy. My rationale is that by increasing the depth of our filter response without also increasing the number of words we compare against, we are expanding our data in too many dimensions without enough comparison points. For example with a filter bank of length 5, we would have a M X N X 60 dimensional filter response space but only 10 points to represent the visual words.

K: K had a significant role in increasing accuracy. In my ablation table, I saw a maximum of 6.5% increase in accuracy by increasing my K from 25 to 30. This makes sense as we are giving the system more features to compare against.

alpha: Alpha also had some unexpected results on the system. In an exercise of increasing alpha from 25 to 30, the system experienced a 1.25% decrease in accuracy. However, when increased from 30 to 35, the accuracy increased by 2.75%. This lack of consistency is likely due to the stochastic process of pixel selection during the sub-sampling of the filter_response matrix.

L: The maximum accuracy increase I got was 4.75% when changing L from 2 to 3. An increase in accuracy is expected since the number of feature histograms will increase with L. By computing features locally in individual subsets of an image, we also can incorporate spatial information into our histograms increasing classification accuracy.

7 Extra Credit

Can you improve your classifier, in terms of accuracy or speed?

1. While I did not have time for implementation, there was room for improvement in the SPM function. For each layer of the pyramid (l) I am breaking the image up into 2^l x 2^l tiles. I am preforming the construction of each layer inside of it's own for loop. The ideal thing to do was to create the bottom most layer, and then construct the remaining layers based on these smaller subsets of the image. This would have saved computational time since I would not need to iterate over a for loop for each layer.

2. My current approach to the tweaking of hyper parameters was very manual. As my ablation tables shows, an increase in any hyper parameter does not always lead to an increase in accuracy. In fact, there were some instances were accuracy decreased. Instead of manually inputting new hyper parameters based on observing performance, it would make more sense to iterate through the many combinations of hyper parameters and log which ones led to an increase. I would need to bound the sample size for each parameter so there was not infinite possibilities - however, this process would be both computationally and time intensive.