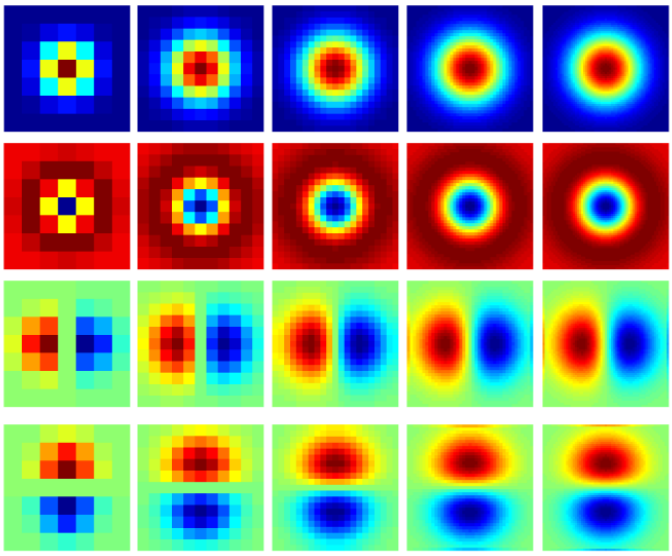
Computer Vision 16-720A: Assignment 1

David Wong Bingxiong (Andrew ID: DBWONG)

23 September 2020

Q.1.1.1



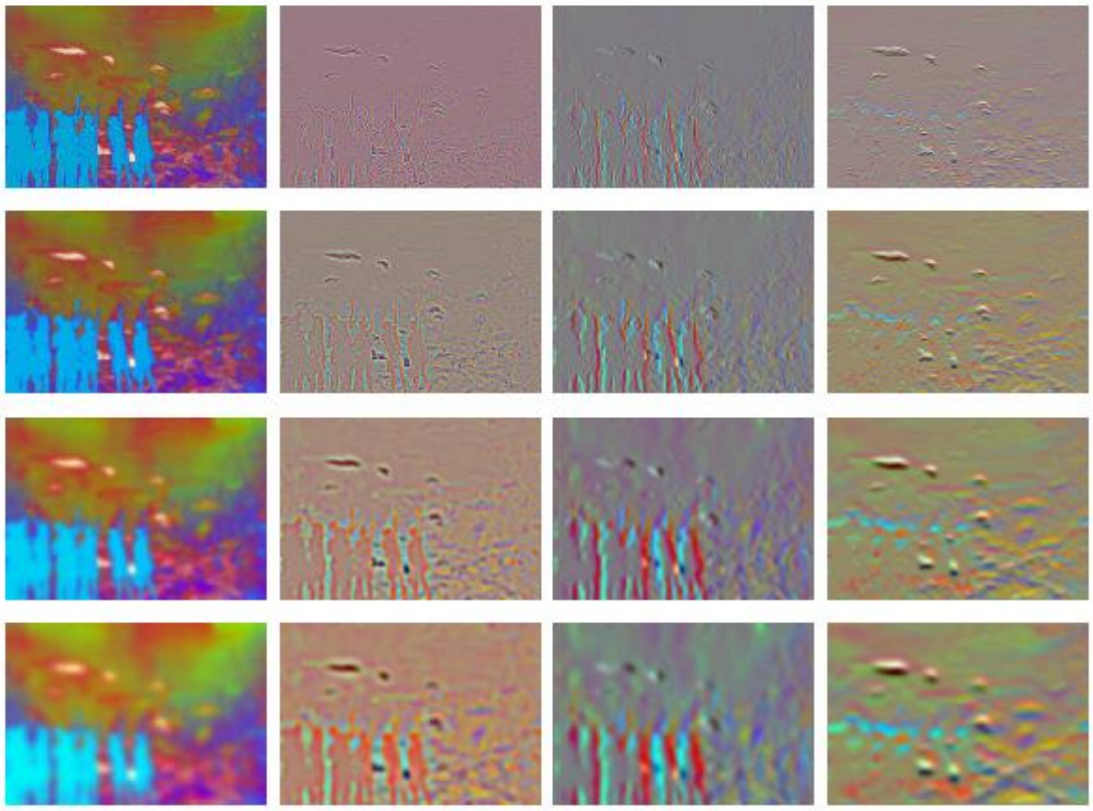
The first row of filters are Gaussian filters which blur the image and result in isotropic noise reduction.
The second row of filters are Laplacian of Gaussian that isotopically detect edges (areas of rapid change).
The third row of filters are derivatives of Gaussian in the x-direction to detect vertical edges.
The fourth row of filters are derivatives of Gaussian in the y-direction to detect horizontal edges.

We need multiple scales of filter responses to handle different level of blurring. The higher the sigma value, the greater the blurring effect.

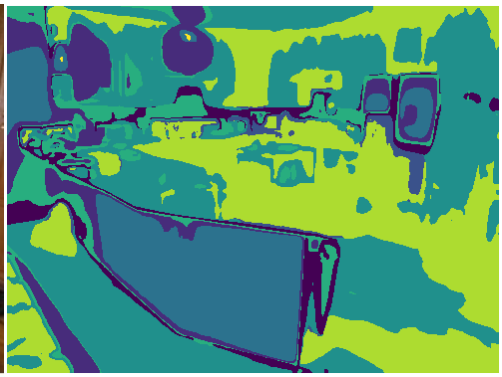
Q1.1.2 (visualization of filter responses)

Original:

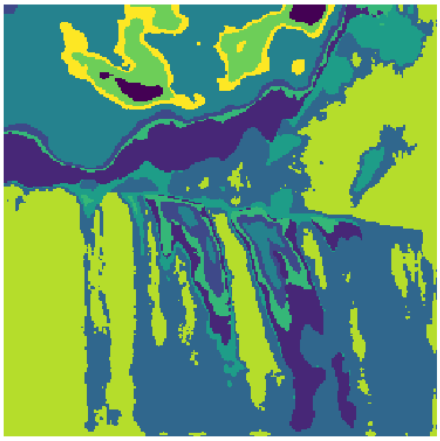
Filter response (scales: 1, 2, 4, 6):



Q1.3 (visualization of wordmaps)

Kitchen

Waterfall


Windmill

Some visual elements like the base of windmill and table-looking furniture in the laundromat and kitchen stand out and are grouped together. Sparse or weak features like some edges of the windmill vanes, equipment in the kitchen/laundromat and thin grass in natural scenes may not be picked up as easily, hence not all word boundaries appear like they make sense.

Q2.5 (confusion matrix and accuracy value)

1	0	0	0	0	0	0
0	1	0	0	0	0	0
0	0	1	0	0	0	0
0	0	0	1	0	0	0
0	0	0	0	1	0	0
0	0	0	0	0	1	0
0	0	0	0	0	0	1

(placeholder while debugging in progress)

Accuracy value: ## % (still debugging)

Q2.6 (hard examples, and an explanation)

My hypothesis of top classes of misclassified images were kitchen and laundromat (work in progress due to debugging. I have a hypothesis in lieu of the confusion matrix that would be tested once I can get the code to fully work.

Upon inspection into training data set, there were some images of kitchens that were misclassified as laundromats. This concludes that the human factor and minimising potential for error during compilation of accurate training and test data is important in building the recognition system.

My hypothesis is the scenery images with (1) common visual elements or (2) less significant or visually obscured features were easily mis-classified. Examples of pairs in the former include aquarium-waterfall (due to common elements of blue pixels, perhaps); park-windmill (due to common elements of trees/fields, perhaps) ad kitchen-laundromat (tiles and squarish appliances as common visual elements, perhaps). For the second category of visually obscured features, we would see some mis-classified due to inability of vision classifier to cognitively discern features well. This was corroborated by Feng, J., Liu, Y, and Wu, L. (2017)[1] which concluded that subtle classification of interior layouts poses a challenge to the sensitivity of geospatial pattern recognition.

Key feature (windmill) is not a dorminant feature (park-looking environment)

Example of obscured features due to flora and fauna

Visual similarity - Example of kitchen that could be misclassified as laundromat or vice versa due to visually similar elements (e.g washing machine and dryer look alike, common feature of tables)

Mislabelled training data - this image was placed in the laundromat folder; however, it is a kitchen

[1] Jiangfan Feng, Yuanyuan Liu, Lin Wu, "Bag of Visual Words Model with Deep Spatial Features for Geographical Scene Classification", *Computational Intelligence and Neuroscience*, vol. 2017, Article ID 5169675, 14 pages, 2017. https://doi.org/10.1155/2017/5169675

Q3.1 (table of ablation study and final accuracy)

Ablation study plan and hypothesis:

Method	Hypothesized Accuracy	Obtained Accuracy
Increasing number of filter scales	>85%	TBD
Increasing K (dictionary size)	85-90%	TBD
Increasing alpha (subset of pixels)	>85%	TBD
Increasing L (spatial pyramid layers)	>85%	TBD

Hypothesis in lieu of fully working code:

K increases accuracy up to a plateau of 85-90%[2]; in the referenced study, K was found to be ideal around 400 to 500. In another paper[3], a range from 80 to 500 clusters showed a increase in accuracy from 45% to 65%, thus indicating that incorrect classifications were due to spatial information accounting.

Increasing alpha and L would also increase accuracy at the expense of computing time as a greater number of pixel samples would enable more spatial features (and histograms) to be obtained within each image and provides more data points for comparison. Apart from images with key features obscured and visually similar interiors, my hypothesis is increasing alpha would increase accuracy above 85%.

[2] Jiangfan Feng, Yuanyuan Liu, Lin Wu, "Bag of Visual Words Model with Deep Spatial Features for Geographical Scene Classification", *Computational Intelligence and Neuroscience*, vol. 2017, Article ID 5169675, 14 pages, 2017. https://doi.org/10.1155/2017/5169675

[3] Vyas, K., Vora, Y., Vastani, R. (2016) "Using Bag of Visual Words and Spatial Pyramid Matching for Object Classification along with Applications for RIS", *Procedia Computer Science* 89, pp457-464. DOI: 10.1016/j.procs.2016.06.102

Extra credit (idea, expectation, result)

Summary of ideas to improve code and expectation:

1) Subtracting mean color using sklearn.preprocessing's StandardScaler function. This function is used to remove the training image mean and scales the image to unit variance. Centering the image data balances signals about a point to overcome rapid changes in gradients to increase speed and accuracy. I was unable to validate the code run due to ongoing debugging of recognition system. In lieu of this I have included modular code blocks in *custom.py*. We would expect to see faster training and recognition as we are forcing the data to have a common standard deviation of 1. The trade-off of this is approach is it may result in spatial information loss for sparse histograms, which is less of a concern for our scene recognition and more a concern for text analysis applications.

2) Next, from Ref [2] a proposed alternative method for classification would be using a new GMM-based codebook learning approach successfully replaced the single codeword-based codebook representation (used in the design of HW1). GMM was found to better characterize clusters' distribution than a single mean value, and thus provided better results. This was attributed to the observation that each cluster typically has different mean values and covariance, and single codeword-based representation results in significant information loss since covariance is lost. A support vector machine (SVM) with histogram intersection kernel as classifier to train histograms for classifications. Only a literature review was done for this in the interest of meeting the assignment deadline, no code blocks were added.

3) Using alternative distance functions [3] could also yield better results. Code blocks found within custom.py in lieu of full implementation due to ongoing debugging of code.

    a. Hellinger Distance
    b. Manhattan Distance – computes rectilinear distance.
    c. Chebyshev Distance

References:

[1] Bityukov, S. I., Maksimushkina, A.V., Smirnova, V. V. (2016) Comparison of histograms in physical research. *Nuclear Energy and Technology*. *2*(2), 108-113. Doi: https://doi.org/10.1016/j.nucet.2016.05.007.

[2] Liu, G., Wang X. (2012) Improved Bags-of-Words Algorithm for Scene Recognition. International Conference on Applied Physics and Industrial Engineering 2012. 24B, 1255-1261. DOI: https://doi.org/10.1016/j.phpro.2012.02.188

[3] K. Meshgi, and S. Ishii, "Expanding Histogram of Colors with Gridding to Improve Tracking Accuracy," in Proc. of MVA'15, Tokyo, Japan, May 2015. DOI: 10.1109/MVA.2015.7153234