y Pyramid of Histograms of Visual Words (PHOW) for image classification

Johana M. Ramirez Borda
Los Andes University
jm.ramirez11@uniandes.edu.co

Francisco A. Rozo Forero
Los Andes University
fa.rozo1843@uniandes.edu.co

1. Introduction

In 2003, in the Institute of Technology in California, Fei-Fei et al. created and compiled the data ser of digital images called Caltech 101. Caltech 101 contains about 9146 images that are distributed between 101 categories such as faces, watches, ants, pianos, airplanes, motornbikes, etc. It also contain an additional background category. Within it, the images of oriented objects were mirrored to be left to right aligned. Each category contains about 40 to 800 images (the majority contains about 50). The size of each image is roughly $300x200$ pixels. The annotations were carefully clicked for each image and they are contained in the file "Annotations.tar". This file contains two pices of information: the general bounding box and a detailed outline specified by humans, enclosing the object. In their website, their recommend a popular number of train images of 1 to 30 and for test, from 20 to 30. These recommendations were taken into account when varying the parameters of the classifier. One of the advantages of this dataset is that almost all the images are uniform in image size and in the relative position of objects.Therefore, users usually don't have to resize, scale or crop images [2],[3]. Examples of this dataset are shown in Figure 1.

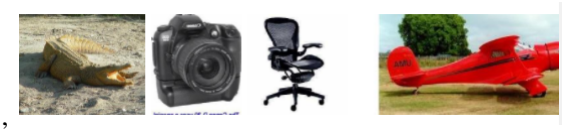

Figure 1. Example of Caltech101 images.

On the other hand, in 2009, in the Conference on Computer Vision and Pattern Recognition (CVPR)the ImageNet database was presented by researches of the Science department at Princeton University. This database was built based on the hierarchical structure provided by WordNet. In this sense, the categories are shown in relation to other categories, for example, within the WorldNet, the word "dog" would be nested under "canine" and this one under "mammal". See Figure 2. Therefore, ImageNet provides 12 subtress consisting of a total of 3.2 million image with annnotations and spreaded over 5247 categories. There are about 600 images in each category. This database is commonly preffered since it was constructed with the goal that all the obbjects in the images should have variable positions, view points, poses, background etc [1].

Figure 2. Example of ImageNet images and their structure based on WorldNet.

2. Materials and methods

For this lab, the software implemented was MATLAB R2016b. Moreover, the data set Caltech101 provided was used in order to classify and evaluate trough ACA index when varying the followin parameters: size of train set, size of test set, number of categories of images to classify, number of dictioonary words and the parameter C in the SVM classifier. Then, when the combination of parameters that gave the best value ACA $-$calculated from de confusion matrix$-$ was found, this combination was used to run the code in the other data set $(Imagenet)$. The code that was run in both cases is in the function $phow\_caltech101$ that was provided and can be found in [3]. This function downloads the database, it allows the change of the parameters mentioned above, then it train the vocabulary, compute spatial histograms, compute the feature map ontained, then train the classifier (SVM) and finally it test this classifier and evaluate, giving as result the confusion matrix. We changed the parameters mentioned and calculated the ACA in each case.

2.1. PHOW

The sift method was first proposed by David Lowe in 2004 and consists on extracting image features so they can be used for further matching and recognition of objects regardless of their position or orientation. At first, the algorithm uses difference of gaussian pyramid of images to determine the key points where features don't change at mul-

tiple scales. Then, gradients of oriented directions are measured and furthermore transformed into a representation of shape and illumination of images. These are called features. Moreover, is the dense sift algorithm, which instead of finding keypoints in the images, filter them with a gaussian kernel and find the features for the whole image (pixel windows every [x,y] step). Phow is a method that implements the dense sift algorithm at an image for multiple σ values for the gaussian kernel.

The phow method uses as hyperparameters the different σ that will smooth the image before every dense sift algorithm and the step size, which is every X and Y pixels will the features be extracted. Therefore, $phow_method$ is scale variant and thats precissely what the code does. It repeatedly iterates the dsift code by varying the scales (sigma) that are used when identifying the features. Also, it takes into account the "step" which establishes every few pixels the features will be calculated.

On the other hand, the combination of parameters in this case was obtained by observing the value of ACA obtained for each combination. It is determined experimentally. All the results of performance were illustrated with the ACA index that is shown in the tables in the results section.

Classes that have distinctive features such as specific objects are easier to classify. However, classes that share many characteristics such as different sports' balls and some mammals are harder to classify because they look alike. Modest and strong boosting methods, that can improve the performance of the classifier by combining multiple classifying algorithms.

3. Results and discussion

3.1. Variation of parameters in Caltech101

In the first place, we tried to find the best combination of parameters that improved the algorithms performance. To do so, the first parameter to evaluate was the number of train images. Values of 5, 15 and 40 were tested.

Table 1. Variation in the number of train images. As this was the first variation in parameters, the rest of them are the default parameters: test=15,categories=5,words=300 and C from SVM=10.

Caltech		
Variation of train set	**ACA**	**Time**
5	0,8551	3:11
15	0,8776	2:47
40	0,9339	3:09

As can be seen in Table 1., the best value of ACA was obtained with a train set of 40 images. This may be because, the more images the algorithm learns to identify and classify, the more objects or patterns it will recognize when evaluating the test images for the first time. However, it is importat to mention that a value extremely high wont be good either because the algorithm will detect much noise intead of objects of interest.

Second, using $train = 40$, the number of images in the test set was varied.

Table 2. Variation in the number of test images.As the best ACA was obtained with train=40, this value was kept and the rest of them are the default parameters: categories = 5,words = 300 and C from SVM = 10.

Caltech		
Variation of test set	**ACA**	**Time**
5	0,96	2:51
15	0,88	2:47
40	0,89	3:01

As can be seen in Table 2. the best value of ACA was obtained when the test set was very small (5). However, we decided that it was best to use a larger set $(test = 15)$ since it contains more variety of images which can give a better reason to be the results. Additionally, it is understood that when having more images to evaluate, it is more likely that the classification is incorrect in some cases. This is a price that is paid (tradeoff) in exchange for having more robust results (which can be obtained by evaluating more images).

Then, using $train = 40$ and $test = 15$, the parameter of number of categories to calssify was varied.

Table 3. Variation in the number of categories. Train=40 was kept because of Table1. and as the ACA was very similar varying test, test=15 was selected. The rest of them are the default parameters: words = 300 and C from SVM = 10.

Caltech		
Variation in number of categories	**ACA**	**Time**
5	0,97	5:00
30	0,72	6:33
65	0,66	6:58
101	0,61	8:10

As can be seen in Table 3. the highest value of ACA was obtained with the default parameter $categories = 300$. This may be because when the number of categories increase, objects are more likely to have characteristics or features to each one of those. Hence, the probability of belonging to the correct category decreases.

Then, using the parameters mentioned above, the number of dictionary words was varied.

Table 4. Variation in the number of number of words. Train=40 and test=15 were kept because of Table 1 and Table 2. The rest of them are the default parameters: C from SVM = 10.

Caltech		
Variation in number of words	**ACA**	**Time**
150	0,66	5:00
300	0,97	5:48
450	0,74	6:04

Here, the number of words influence it as well; if there is a large number of categories, few words won't be enough to have a good representation that can differentiate the images from each other. Likewise, having too many words overrepresentate the images, reducing the specificity of their features, making them more likely to share characteristics with other classes and hence be misclassified (see *table 3* and *table 4*).

Finally, the parameter C in the SVN classifier was varied.

Table 5. Variation in the parameter C of SVM. Train=40, test=15 and words=300 were kept because of Table 1 and Table 2 and Table 3.

Caltech		
Variation in parameter C-SVM	**ACA**	**Time**
5	0,72	8:24
10	0,97	5:00
20	0,71	4:50

In the Table 5. the highest value of ACA was also obtained with th default value. The parameter *C* in SVM's tells how much misclassification will be allowed while training. If the parameter is to low, the weights will have more significant changes during the training session because it tries to find the longest distance that separates the two classes at a cost of misclassification. In contrast, a large *C* can end in a bad classifier because it allows more misclassification in the trainin. In consequence, more images will be misclassified.

Lastly, observing the "Time" columns in all the tables, it was noted that the processing time is proportional to the number of categories, size of train set, size of test set and the number of dictionary words, because the more data is asked to process, the longer it takes too. Moreover, when the algorithm is run for the first time, the time can be longer than the second time if the parameters stay the same or differ a little. This happens because the second time, most of the data is already held in the RAM memory. So, the computer won't have to search it into the ROM memory, which happens the first time the code runs (see *table 1* and *table 2*). It was observed when running the codes because on the first run, the memory used was longer (ie. 516.8 MB when varying the train set with train=40) than later processes (ie. 385MB when varying test set).

4. Conclusions

Similarly to textons, phow algorithm uses features to represent characteristic of objects by using the dense sift algorithm. Even though the process is not fast, once the features are extracted, it can be trained fast and the classifier can provide a robust differentiation of the classes based on their features.

References

[1] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE, 2009.

[2] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer vision and Image understanding*, 106(1):59–70, 2007.

[3] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. `http://www.vlfeat.org/`, 2008.