Laboratory 8: PHOW

Ramón Emiliani
Universidad de los Andes
201125694
rd.emiliani689@uniandes.edu.co

Alejandro Posada
Universidad de los Andes
201227104
a.posada10@uniandes.edu.co

Abstract

Image representation is a crucial step in image classification tasks. In this laboratory, we used Pyramids of Histograms of Visual Words to represent images and a Support Vector Machine (SVM) to classify images. We adapted the code from the VLFeat Library. We trained and tested the model in the Caltech101 database and obtained an ACA of 70.00%*. We modified the code to allow it to run in Imagenet. In this database, we obtained an ACA of* 52.59%.

1. Introduction

Image classification is one of computer vision's problems in which more work has been done. It consists in producing a class label for an image according to the object category it contains. Classifiers are built by using labeled examples to build a rule that assigns a label to new images. In this type of task, the classification methos is important; however, image representation is a fundamental factor to be taken into account when performing image classification.

A very useful image representation that has produced great results is Scale-Invariant Feature Transform (SIFT) [4]. This method extracts features from interest points or by densely sampling the image. In order to calculate SIFT features, $M \times M$ pixel patches are sampled uniformly on a regular grid. The normalized histogram of local gradient directions at every $M \times M$ patch around the interest point is computed. Then, these descriptors are concatenated and this vector becomes the SIFT representation of the image.

An effective way to describe images to perform classification is Pyramid of Histograms of Visual Words (PHOW) [3]. This technique is an extension of the bag-of-words model (BoW). A bag of words is a sparse vector that represents the frequency of "words" (image features) that appear in a "document" (an image). The principal problem of the BoW model is that it doesn't encode spatial information: the model informs the presence and frequency of particular feature in the image but doesn't inform where that feature is present. A solution to this problem is PHOW, that works by dividing the image into increasingly smaller subregions and by computing the histogram of visual words at each local sub-region. PHOW features can be calculated by computing dense SIFT descriptors for the training images image. Then, the descriptors are grouped via k-means (or another clustering algorithm) to form a visual vocabulary. A spatial pyramid of visual word histograms is computed to represent each image.

In this laboratory, we used PHOW to train and test a classifier in a small subset of the ImageNet database. We also performed experiments in the Caltech101 dataset [2]. We adapted the script from the VLFeat Library, originally developed by Andrea Vedaldi and Brian Fulkerson.

2. Materials and methods

Caltech101 comprises 101 categories and each one of them is composed by between 40 and 800 images. Each image is roughly 300×200 and in most cases the object is centered in the image and the background is mostly uniform (figure 1).

Figure 1: Examples of Caltech101 images.

The second database we used is ImageNet [1], which was built according to the WordNet hierarchy. The database has 100000 categories, each one with 1000 images. This is a much more complex dataset than Caltech 101: object are not always centered and in general backgrounds are not uniform (figure 2).

Figure 2: Examples of ImageNet images.

We performed our experiments on Caltech101. We used the VLFeat Library script, which uses PHOW features (dense SIFT), spatial histograms of visual words, and a χ^2 SVM as a classifier. We evaluated the effect the number of classes, the spatial partition of the images, the number of train and test images, the number of words used to represent the images. Additionally, we varied the SVM C parameter, that controls the tradeoff between a smooth decision boundary and classifying points correctly. Tables 1 resumes the experiments we performed on Caltech101.

Table 1: Experiments performed on Caltech101

Experiment	**No. of classes**	**Train images**	**Test images**	**No. of words**	**SVM C param.**	**X**	**Y**
No. of classes	1:101	15	15	300	10	2	2
Size of train set	32	1:30	15	300	10	2	2
Size of test set	32	15	1:30	300	10	2	2
No. of words	32	15	15	100:100:1500	10	2	2
SVM C param.	32	15	15	600	1:30	2	2
Spatial part.	32	15	15	600	10	1:8	1:8

We trained and tested our model (600 words, SVM C parameter $= 10$, $X = 4$, $Y = 4$) on a subset of ImageNet. The model was trained on the the 90 images per each one of the 200 categories of the train set. We evaluated the model in the test subset, which is composed by 200 categories, each one with 90 images.

3. Results

3.1. Experiments

Figures 3, 4, 5, 6, 7 and 8 show the effects that each of the selected variables have on the classification accuracy when using Caltech101.

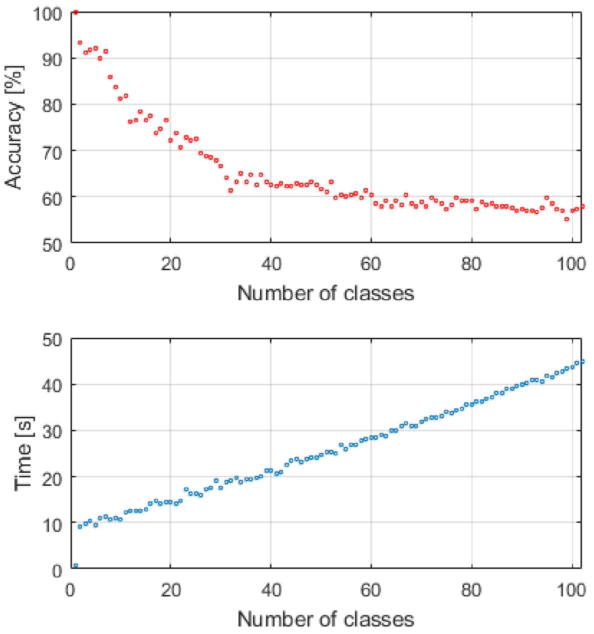


Figure 3: Effect of the number of classes.

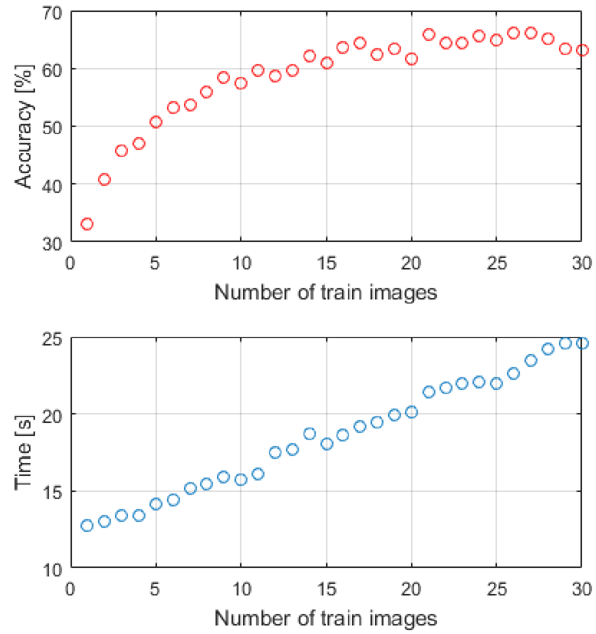


Figure 4: Effect of the train set size.

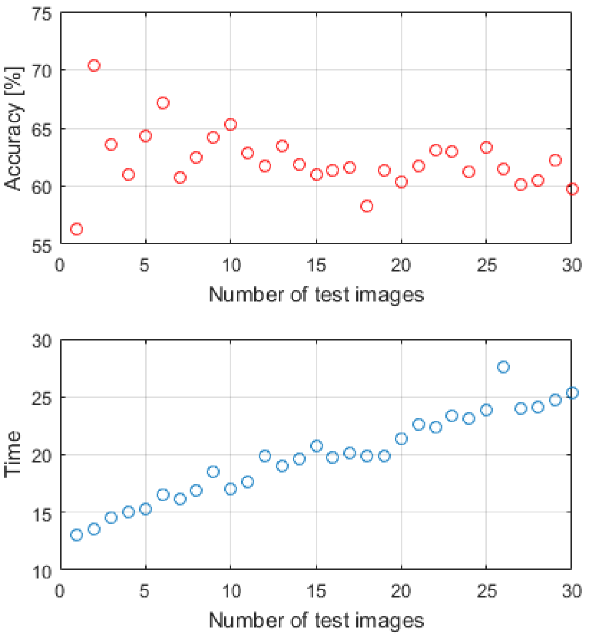


Figure 5: Effect of the test set size.

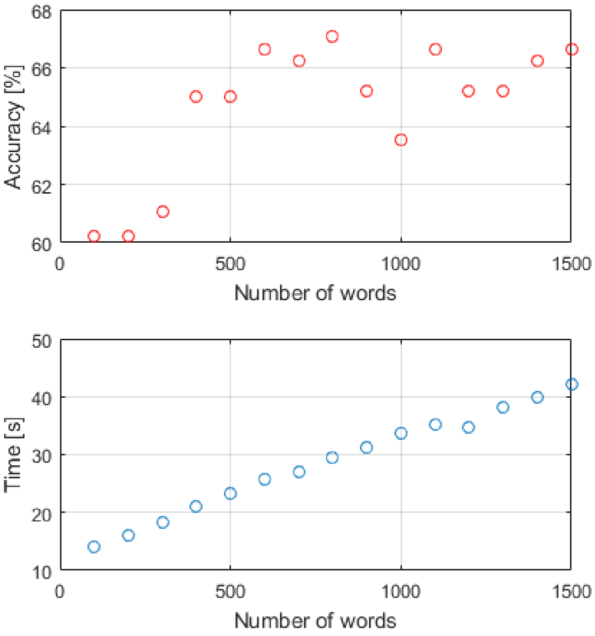


Figure 6: Effect of the number of words.

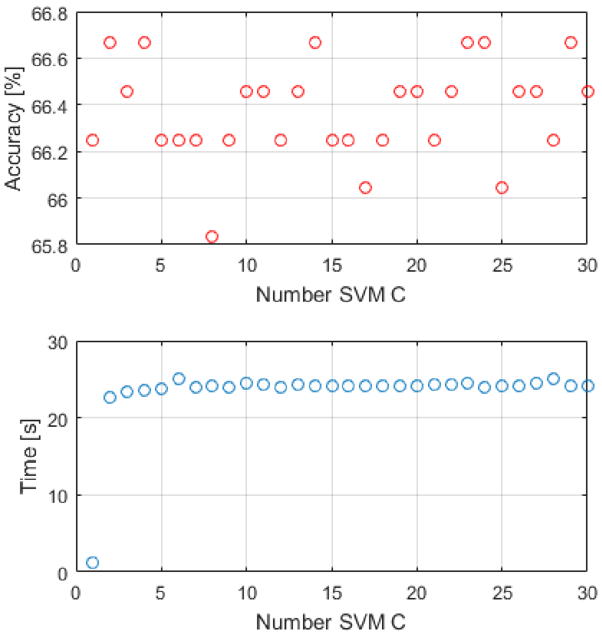


Figure 7: Effect of the SVM C parameter.

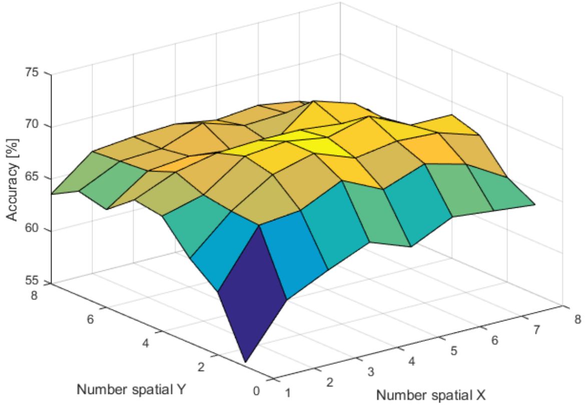


Figure 8: Effect of the spatial partitioning.

The experiment's results can be grouped in three categories according to the curves' shape:

1. Accuracy decreases as the variable increases (number of classes and number of test images).

2. Accuracy increases as the variable increases (number of words and number of train images).

3. No clear correlation between the accuracy and the variable (SVM C parameter).

The first category can be explained as follows. As the variable (i.e. the number of classes or the number of test images) increases, the classification task becomes more complex since the probability of misclassifying an image augments. This means that these parameters are not *per se* a part of the model; they merely simplify the problem. As the classification task becomes more complex, the computation time increases because there are more images that must be taken into account and therefore the number of calculations increases.

The second category states that the accuracy is proportional to the number of train images and –less clearly– to the number of words. Indeed, a greater number of train images allow to build a model that is better fitted to the data. Additionally, the classifier's accuracy generally increases as the number of visual words augments. This is because as the number of words increases, the images are better described and thus the classifier has more information to discrimine between classes. As in the previous case, the computation time is proportional to the variable, which means that more number of words and more train images make the classification more complex without negatively affecting the results.

The third category shows a bizzarre behaviour of the effect of the SVM C parameter. One would expect that for large values of C, the optimization would choose a smaller-margin hyperplane if that hyperplane does a better job of getting all the training points classified correctly. Conversely, a very small value of C would cause the optimizer to look for a larger-margin separating hyperplane, even if that hyperplane misclassifies more points. However, figure 7 doesn't show this expected behaviour. This might be due to the fact that the SVM C parameter values were not searched in the correct range. It is also noteworthy that this parameter has no effect on the computation time (except for the case of very small SVM C parameters).

The spatial partitioning doesn't belong to any of these three categories. Figure 8 shows that the classification accuracy is low when X or Y is close to zero. However, the maximum accuracy is not reached when both X and Y are maximum. The maximum accuracy was obtained with $X = Y = 4$.

3.2. Results on Caltech101 and ImageNet

Figure 9 shows the confusion matrix obtained after evaluating our method in Caltech101.

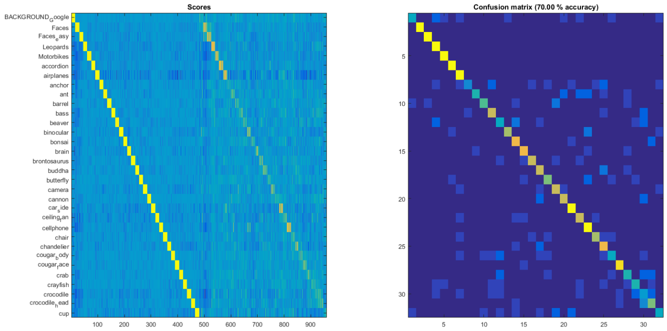


Figure 9: Results from Caltech101. ACA $= 70.00\%$.

Figure 10 shows the confusion matrix obtained after evaluating our best method in the ImageNet subset.

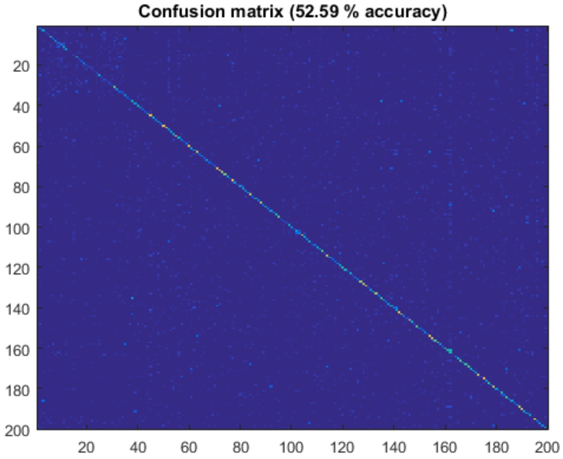
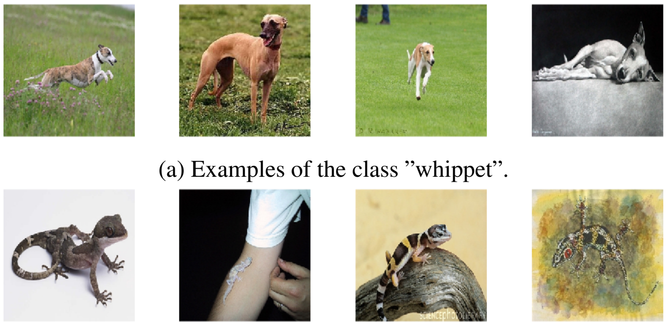


Figure 10: Results from Imagenet. ACA $= 52.59\%$.

One of the main differences between Caltech101 and ImageNet is the number of classes and the number of images. The fact that ImageNet has more classes and images than Caltech101 explains in part why the obtained ACA is smaller in ImageNet (as figures 3 and 5 show, accuracy is inversely proportional to these two variables). However, the difference between both ACA is explained because Caltech101 images are very simple and thus the classification task is easier. The "easiest" class is "whippet" (which was classified correctly 80.77% of the times). This can be explained because, as figure 11a shows, most whippets have characteristics that distinguish them from the other types of dogs present in the database (for example, chihuaha and great dane). Additionally, most whippet's have a similar green background. On the other hand, the category that was correctly classified the less times (0.95% of the times) is "banded gecko". As figure 11b shows, not all banded geckos are similar (some of them are colorful while others are not, etc.) and most of the backgrounds of the images that belong to this class are very different.



(a) Examples of the class ”whippet”.

(b) Examples of the class ”banded gecko”.

4. Conclusions

PHOW proved to be a great method to classify the simple images of Caltech101. However, ImageNet is a more complex dataset and thus PHOW didn’t perform as well in this dataset. In order to improve the results, more parameters could be taken into account. One of the PHOW parameters that could improve the results is the number of pyramid levels. Indeed, splitting the image with pyramids encodes image information in multiple resolutions, which results in a higher-dimensional representation that preserve more information. Moreover, varying the grid size (”Steps”) could give valuable information to improve the quality of the SIFT dictionary. Additionally, choosing different values for the classifier’s parameters can help improve the classification accuracy. For example, varying the SVM’s γ, ϵ and the kernel parameters affects the way the classifier discriminates the data and thus might help to obtain better classification results. The image representation could be improved by taking more attributes into account. Using texture and giving importance to the color characteristics of the image could help identify features that can otherwise be ignored by PHOW. Finally, external information could also be valuable. For example, location context (*i.e.* knowing where an image was taken via GPS coordinates) could provide helpful information to classify images: it is more likely that an image of a llama was taken in South America than in Canada, as it is more likely for a sea cucumber to appear in a picture that was taken in the Caribbean Sea and not in the Sahara.

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] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Computer vision and pattern recognition, 2006 IEEE computer society conference on*, volume 2, pages 2169–2178. IEEE, 2006.

[4] D. G. Lowe. Object recognition from local scale-invariant features. In *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, volume 2, pages 1150–1157. Ieee, 1999.