Lab 08: PHOW

Adrian S. Volcinschi M.
Universidad de los Andes
https://uniandes.edu.co/en
as.volcinschi@uniandes.edu.co

Abstract

The task of object recognition is a task far from being solved. It is still one of the prevalent challengers in the computer vision area. One of the first and best attempts to solve this was Pyramid Histogram Of visual Words where a Bag of Visual Words was implemented. The model used SIFT to extract some keypoints and Oriented Gradients to create a representation space which would be clustered later on. This attempt gave very good results for the time it was proposed (2004) such as an ACA of 0.7083 in the Caltech-101 dataset. From this that dataset was deemed solved and moved into harder datasets such as ImageNet. In this new dataset the proposed model is not even close to solving the dataset with an ACA of 0.2743.

1. Introduction

We as humans classify almost everything we see to understand what's going on around us. For example, when you are crossing a road you check both sides first to see if a car or truck is coming so that you wait until it passes to cross the road safely. This task of recognition is what we would want to teach a computer: how to recognise objects in a picture. This computer vision task consists of giving an input image to the computer and it has to be able to determine what kind of object is the one on the image and assign it its corresponding label. Different levels of this problem can be perceived: fine-grained or coarse-grained. The main difference between them is that in fine-grained object recognition the task is to label a Ferrari car as a Ferrari. On the other side, the task of coarse-grained object recognition is to label the image as a Car. One example of a coarse-grained dataset can be Caltech-101 with categories such as cellphone, camera, butterfly, etc [2]. On the other hand, a fine-grained dataset can be ImageNet where categories are Banded gecko, Bedlington terrier, milk can, etc [1].

Depending on the dataset there are many ways to approach this task. One intuitive way is to create a representation space using color and shape and that is exactly Moghimi [5] did with the CUB200 dataset for example. As datasets became more complex the algorithms had to get more complex themselves. A more robust algorithm called SIFT was proposed in 2004 to expand the representation space of images. The Scale-Invariant Feature Transform (SIFT) proposed by Lowe in 2004 [3] wanted to create a representation using oriented gradients in specific keypoints calculated with Difference of Gaussians (DoG). This algorithm inspired other researchers to use it because of its efficiency and quality and that's why it was used in this paper.

2. Materials and Methods

The Bag of visual Words (BoVW) algorithm consists in creating a vocabulary of "visual" words that are able to describe a set of images. The first step is to extract some keypoints that remain invariant to scale, rotations and translations. This is were the SIFT algorithm helps us find these keypoints. The difference is that here we used dense SIFT which instead of calculating the keypoints using DoG it runs SIFT at a dense grid of locations at a fixed scale and orientation [6]. When the keypoints are obtained then a window around it is considered and divided into 16 subcells. In each of this cells is calculated a Histogram of Oriented Gradients (HOG) [4].
After having this representation vectors a clustering method such as Kmeans is used to find the centroids of each "visual word" and create the corresponding dictionary. This way when a new image comes in its visual words histogram can be calculated and then classified using a classifier such as K-Nearest Neighbors. When the dense SIFT is done at different scales with the rest of the process being the same this algorithm is called a Pyramid Histogram Of visual Words (PHOW) and it is the algorithm used in this paper.

Two datasets were used for testing this proposed strategy: Caltech-101 (`http://www.vision.caltech.edu/Image_Datasets/Caltech101/101_ObjectCategories.tar.gz`) and a subsampled

ImageNet dataset with 200 classes (`http://bcv001.uniandes.edu.co/imageNet200.tar`). Some pictures of each dataset and be seen in Figure 1 and Figure 2 respectively.

Figure 1: Pictures of the Caltech-101 dataset

Figure 2: Pictures of the subsampled ImageNet dataset

As it can be seen from Figures 1 and 2 the images of the Caltech-101 dataset are much more simple because of the fact that most of them come from catalogs. On the other hand, the images of the subsampled ImageNet are more natural images and therefore pose a tougher task to be solved in comparison to Caltech-101. Additionally, the Caltech-101 has 101 categories whereas the subsampled ImageNet dataset has 200. This means that, both in type of images and amount of categories, the subsampled ImageNet dataset is harder to solve.

3. Results

Various experiments involving number of train images, number of test images, number of visual words, spatial partitioning, and cost of the SVM classifier were run in each of the datasets. The original script provided by VLFeat uses 15 train images, 15 test images and 600 visual words as well as [2 4] for spatial partitioning and 10 as the cost of the SVM.

3.1. Number of train images

To begin with, the number of train images was variated in the range (15, 30, 60, 100) and the results can be seen in Figure 3.

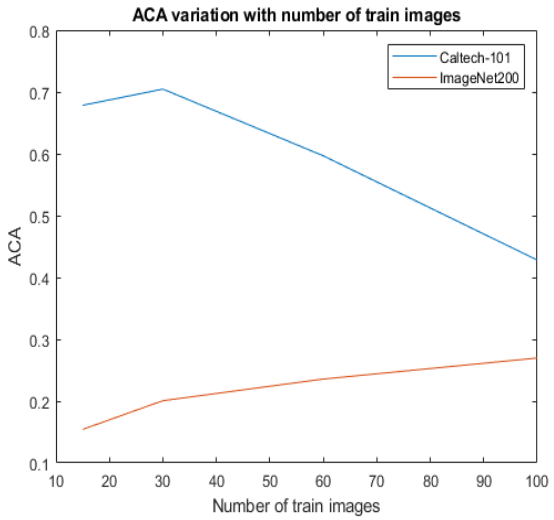

Figure 3: ACA for Caltech-101 and subsampled ImageNet datasets with different number of train images

From this it can concluded that for the Caltech-101 dataset the best performance comes with 30 train images and for the subsampled ImageNet dataset 100 train images. This means that the more train images, the more this model overfits in the Caltech dataset and therefore ends up with a lower ACA for the test images. For the subsampled ImageNet dataset the ACA got better with more images which means this dataset is less prone to overfitting.

For the next experiments a fixed number of training images (30) is going to be used.

3.2. Number of test images

Another aspect to be taken into consideration is the number of test images used to calculate the ACA on each run. This number was variated in the same way as the number of train images and the results are shown in Figure

4.

Figure 4: ACA for Caltech-101 and subsampled ImageNet datasets with different number of test images

As expected, if the number of test images increases the ACA decreases for every case. On the other hand, the ACA didn't decrease that much in the subsampled ImageNet dataset. This means that the model is not overfitting to this dataset because more test images didn't decrease the ACA significantly compared to the Caltech-101. To maintain a sense of equality the combination 30 train images and 30 test images was used for the rest of the datasets experiments.

3.3. Number of visual words

The number of visual words was preset in 600 in the original code of VLFeat. To observe how the amount of words changed the ACA the values of 400, 800 and 1000 visual words were used. This changes in the ACA can be observed in Table 1.

Table 1: ACA for the Caltech-101 and subsampled ImageNet datasets when the number of words is variated

Dataset\Number of Visual Words	400	600	800	1000
Caltech-101	0.6199	0.6239	**0.6278**	0.6265
Subsampled ImageNet	0.1943	0.2012	**0.2033**	N.A

As expected, more words give a better result up to a point where it starts overfitting. For the Caltech-101 it was 800 visual words whereas the ImageNet was Y. Since the ImageNet dataset is more complex and variated it is normal for it to overfit at a higher number of visual words, or in other words this dataset needs more visual words to represent its images. The 1000 visual words for the subsampled ImageNet dataset was omitted because it consumed too many resources and the improvement from 600 to 800 was not that significant and therefore left aside.

3.4. Number of spatial partitions

The spatial partitioning is important because it expands or contracts the representation space. To see if the partitioning was important, an additional 6 and 8 size partitionings were implemented. This results can be observed in Table 2.

Table 2: ACA for the Caltech-101 and subsampled ImageNet datasets when the number of spatial partitions varies

Dataset\Spatial Partitionings	[2 4]	[2 4 6]	[2 4 6 8]
Caltech-101	0.6278	**0.6301**	0.6206
Subsampled ImageNet	0.2012	**0.2360**	0.2337

For both datasets the additional 6 spatial partition gave better results and for the 8 it decreased again. This means that the method gives a better result when having more representation with the additional partition of 6 but overfits when the partition of 8 is added and hence the ACA goes down again due to the low bias of the model.

3.5. Cost in the SVM

The original cost for the SVM was 10. To see how this affected the model it was variated to 20 and 30. The results can be observed in Table 4.

Table 4: ACA for the Caltech-101 and subsampled ImageNet datasets when the SVM Cost is variated

Dataset\SVM Cost	10	20	30
Caltech-101	0.6301	**0.6304**	0.6255
Subsampled ImageNet	**0.2360**	0.2320	0.2330

It can be seen that for the Caltech-101 dataset the cost that gave a better result was 20 when for the subsampled ImageNet dataset it was 10. This means that the subsampled ImageNet dataset has a lower error margin compared to the Caltech-101 dataset. In other words, the Caltech-101 dataset needs more error freedom to obtain a better result compared to the subsampled ImageNet dataset.

3.6. Best parameters for each dataset and final results

As it was shown, different datasets end up needing different parameters for their models. The best combination of parameters for each dataset is shown in Table 3. Additionally, because of resource limitations the tests for the

Table 3: Best parameters for the Caltech-101 and subsampled ImageNet datasets and their corresponding ACA

Dataset	# Train images	# Test images	# Visual Words	Spatial Partitionings	SVM Cost	ACA
Caltech-101	30	15	800	[2 4 6]	20	**0.7083**
Subsampled ImageNet	100	30	800	[2 4 6]	10	**0.2743**

subsampled ImageNet dataset were done on the most part with only 101 categories. The final results with all 101 categories for Caltech-101 are shown in Figure 5 and for the 200 categories of the subsampled ImageNet in Figure 6.

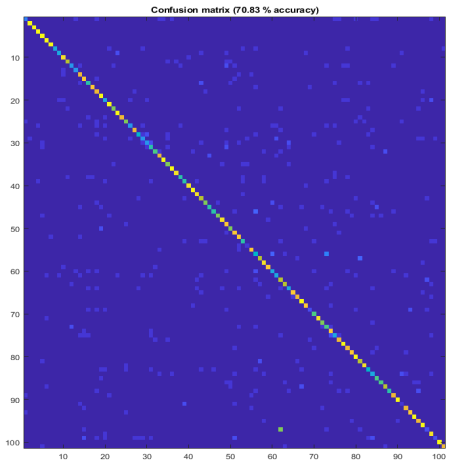

Figure 5: ACA for Caltech-101 with the best parameters

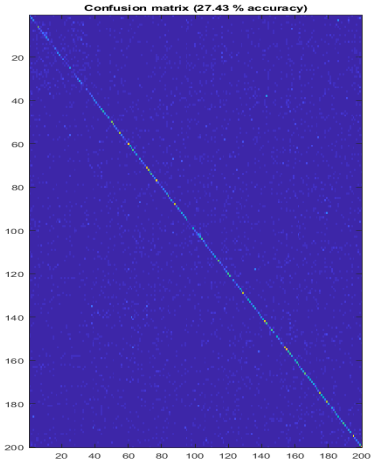

Figure 6: ACA for subsampled ImageNet with the best parameters

By analyzing this confusion matrix it can be seen which objects were best labeled and which weren't. Some good and bad results for both datasets are shown in Figures 7,9 and 8,10 respectively.

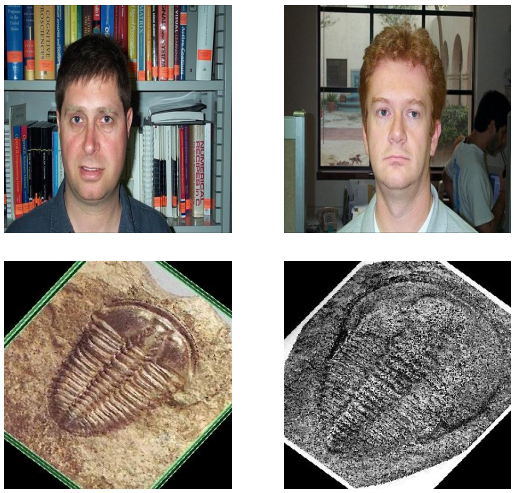

Figure 7: Good results for Caltech-101 dataset

Figure 8: Bad results for Caltech-101 dataset

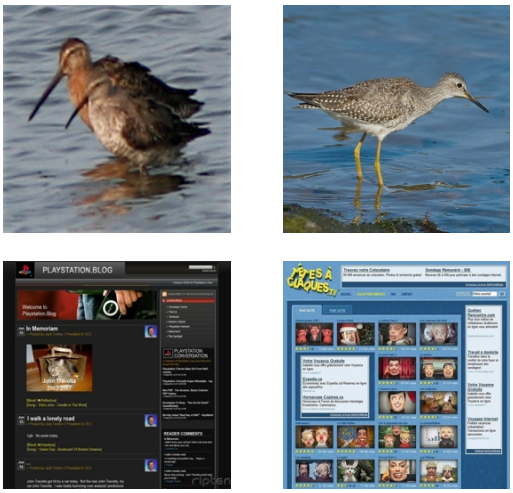

Figure 9: Good results for subsampled ImageNet dataset

Figure 10: Bad results for subsampled ImageNet dataset

As it can be seen in Figure 7 the images that are classified correctly are very similar and share some very specific patterns. The variance between them is not high enough so the algorithm just learns it and that's why it ends up overfitting. If we look at the poorly classified objects in Figure 8 it is clear that when the variance between images increases the model starts to underperform.

For the subsampled ImageNet dataset really good results are scarce. Some of the best results can be seen in Figure 9. Although the variance between photos is high enough, the model is good enough to label them together even with the extra noise of each image (Persons, brightness, etc). On the other hand, as most of the categories in this dataset, the results for other categories are really poor. With an ACA of only 0.2697 it is to be expected to get bad results overall. If we look closely at the images in Figure 10 we can see that the images are very different one from another. This can explain why the model is not so good for this particular categories. For example, the ambulance on the top left side of Figure 10 doesn't even compare to the helicopter ambulance in the top right side.

Figure 11: Most frequent confusions in bad results of the subsampled ImageNet dataset

It is pretty obvious from Figure 11 that wrong classifications are going to occur between these categories (Ambulance vs Recreational-vehicle and African-elephant vs Tusker). This shows that the PHOW approach can be good for classifying objects that are very different but when it comes to a fine-grained classification it performs really bad.

4. Conclusions

The PHOW method was a really good model for the time it was proposed, 2004, and was able to obtain really good results in the Caltech-101 dataset. The good enough results led to try the model in a harder dataset such as ImageNet and it performed very poorly. What this means is that the model was not good enough for a new tough dataset such as ImageNet and that the Caltech-101 dataset was not a very good one since it could be solved so easily. In terms of the algorithm, it falls short in labeling images that are the same object but have a high variance between them. Similarly, the use of only Bag of Visual Words with SIFT can lead us to think that the shape representation needs to be expanded for a better result.

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 *CVPR09*, 2009.

[2] F.-F. Li, M. Andreetto, and M. A. Ranzato. Caltech101 image dataset. 2003.

[3] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, Nov. 2004.

[4] S. Mallick. Histogram of oriented gradients. https://www.learnopencv.com/histogram-of-oriented-gradients/, dec 2016.

[5] M. Moghimi. Using color for object recognition. 2011.

[6] A. Vedaldi and B. Fulkerson. Dense sift as a faster sift. `http://www.vlfeat.org/overview/dsift.html#ref1`, 2008.