Using BoW and VLAD in search engines

Lauren Silva Rolan Sampaio
Majeure Image, Sound and Artificial Intelligence
Ecole Naationale Supérieure d'Ingénieurs de Caen
Caen, France
lauren.rolan@ecole.ensicaen.fr

***Abstract*—The objective of this paper is to present the theoretical basis of search engines based on Visual Bag-of-Words and VLAD descriptors. We will introduce these aspects, as well as the results obtained in the practical exercises. We analyze multiple factors, such as the impact of normalization of the data and the number of clusters used.**

***Index Terms*—bag-of-words, VLAD, content-based image search, search engines**

I. Introduction

Most search engines nowadays are description-based, which means they rely in some form of description of the image, usually made manually by a human. Even if effective, this method is too expensive in resources, since there are multiple images that are not yet annotated and demand a human task-force to do so. From this necessity of finding images without a well detailed annotation is born the content-based image research.

This type of research uses the descriptors of the image as parameters.

II. Explaining BoW and VLAD

A. SIFT descriptors

The Scale Invariant Feature Transform is a method developed in [1], where the descriptors of a image are calculated after four stages:

- **Scale-space extrema detection**, where it searches over all scales and image locations.
- **Keypoint localization** selects the most stable keypoints.
- **Orientation assignment**, where orientations are assigned to the keypoints based on the image gradient.
- **Keypoint descriptor** are the result of the measuring of the gradient of the image in a given scale.

These descriptors are then stored in the place of the image, since they represent them in a unique manner. The content-based image search is done using these descriptors instead of the pixels of the image. Different images have a different number of descriptors.

A SIFT descriptor has a size of 128 numbers, since it has a 4×4 grid, where each cell contains 8 directions. Therefore, the constant 128 will be considered as implicit in this paper.

B. Bag-of-Words applied to image search

The BoW method is usually applied to natural language processing. It considers the existence of a vocabulary containing N words. Let's say we have D documents. To represent the whole database we will have a matrix $D \times N$, where the presence of a number in a column indicates the frequency of a word in the document.

Applying this method to images implies in identifying some characteristics as the "words", and group them in a "vocabulary".

Let's consider the database as having $D = 1491$ images. If we apply a clustering algorithm in each descriptor of each image, we obtain $N = 8{*}128$ columns in our matrix (assuming we have 8 clusters).

C. VLAD

The VLAD method has a similar approach, better explained in the article [2]. We consider, as before, the descriptors SIFT and their resulting clusters. However, instead of just counting the frequency of each cluster in the image, we calculate the sum of the difference between each descriptor in a cluster and this cluster's centroid.

The Equation 1 gives the mathematical form of the calculus we have done, with $i = 1, \ldots, n_{clusters}$ and $j = 1, \ldots, n_{SIFTdimension}$.

$$v_{ij} = \sum_{x|x \in NN(c_i)} (x_i - c_{ij}) \tag{1}$$

III. Experiments

A. SKLearn implementation

For these experiments, we used the *INRIA Holidays Database*. This database contains images with some famous locations.

The first part of our implementation, contained in the file *Part1.ipynb*, uses the BoW and VLAD descriptors that were furnished. The BoW descriptors were composed by $D = 1491$ images, each containing $N = 1024$ values (8 clusters represented by SIFT descriptors). The VLAD descriptors contained also $D = 1491$ images, however they were clustered in 64 groups, giving a total of $N = 8182$ values.

Using the function *cdist* of *SciPy*, we were able to perform the calculation of the euclidean distance between each element in the database and one given request. We apply the same method to both data sets (VLAD and BoW) and we obtain the results given in Table I.

We verify that the normalization has an important role for the VLAD descriptors, increasing its performance. On the

model	norm	MAP	time
BoW		0.6439	2.1638s
	L1	0.5982	1.0604s
VLAD		0.5666	4.9484s
	L2	0.7339	4.9788s

TABLE I
COMPARISON BETWEEN THE DATA SETS GIVEN.

other hand, its time complexity continues to be basically the same.

The BoW method had a better performance w.r.t. the MAP score without normalization. When the normalization is applied, the score is lower, but the time complexity is reduced by half.

It can be explained by how BoW and VLAD calculate their descriptors. VLAD uses multiple factors, such as the variance and the direction of each patch, therefore the normalization tends to increase it's performance. The BoW method, in the other hand, deals with the frequency of appearance of a pattern in the document. When normalized, this frequency is modified.

B. Our implementation

In the second part of our experiment, contained in the file *Part2.ipynb*, we implement our own BoW and VLAD descriptors, taking as basis the SIFT descriptors given by the professors. However, the computer where the experiments were executed did not have enough memory to store the whole data set, so we only processed a fraction of it (files 1 to 257).

First of all, we defined a clustering method. It clusters the SIFT descriptors around k clusters, which are passed as parameter. The basis for this method is the class *KMeans* of *SKLearn*.

The second part consisted in creating the methods of BoW and VLAD. Starting by the BoW function, we create a matrix $D \times k$, where D is the number of images and k the number of clusters, and we count the frequency, for each image, of all clusters individually.

The VLAD function is a computational representation of the Equation 1. For each image in the database, we analyze a subgroup formed by the descriptors contained in the same cluster. We then get the difference between each element of this cluster and the centroid, and we attribute the sum of these values to the columns of the resulting matrix. The resulting matrix has a size $D \times 128k$, where k is the number of clusters. Further, we normalize the resulting matrix using a L2-norm.

In Figure 1 we analyze the performance of the methods accordingly with the number of clusters used. We realize that the VLAD descriptors have an increasingly performance with the augmentation of the number of clusters, while the BoW method has an actual decrease in performance when we consider higher dimensions of clusters.

This phenomena can be explained by the fact that the Bag-of-Words method does not perform well when it encounters what we call "stop-words", which are basically the noise in the vocabulary. When we cut these stop-words (usually by reducing the number of clusters), we arrive to better results.

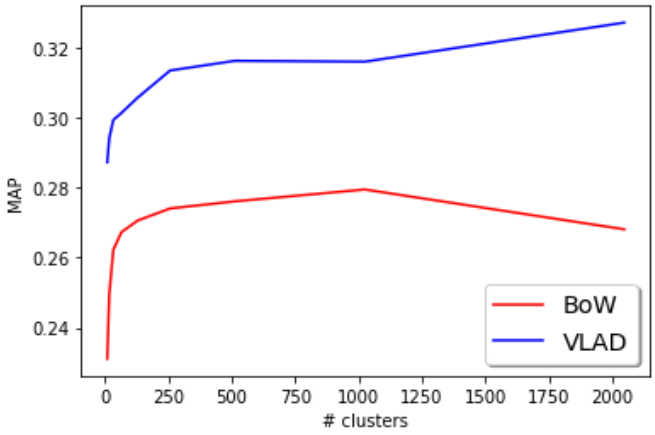

Fig. 1. Comparison between the two implemented methods.

Another relevant point is the MAP score of both functions. It must be considered the fact that we were processing only a fraction of the data set (256 images from a total of 1491, resulting in approximately $\frac{1}{6}$ of the database). Our results, however, are approximately $\frac{1}{3}$ of the original ones, obtained after processing the entire data set.

IV. CONCLUSION

The experimentation with the libraries *SciPy* and *SKLearn* allowed us to analyze the impact of these methods without dive into their implementation. We were able to measure the impact of the post-processing, where we verified how the normalization influences the results.

Based on the acquired knowledge of the previous part, we were able to implement a close solution. The results, however, where not similar to the ones obtained in Part 1, but we attribute this variance to the fact we could not import the entire data set to the engine.

REFERENCES

[1] David G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints". International Journal of Computer Vision, vol. 60, pp. 91–110, November 2004.

[2] R. Arandjelovi, A. Zisserman, "All about VLAD". 2013 IEEE Conference on Computer Vision and Pattern Recognition. October 2013.