AN EMPIRICAL STUDY ON THE COMBINATION OF SURF FEATURES WITH VLAD VECTORS FOR IMAGE SEARCH

E. Spyromitros-Xioufis[1,2], *S. Papadopoulos*[1], *I. Kompatsiaris*[1], *G. Tsoumakas*[2], *I. Vlahavas*[2]

[1]Informatics and Telematics Institute, Center for Research and Technology Hellas, Thessaloniki, Greece
[2]Department of Informatics, Aristotle University of Thessaloniki, Greece

ABSTRACT

The study of efficient image representations has attracted significant interest due to the computational needs of large-scale applications. In this paper we study the performance of the recently proposed VLAD method for aggregating local image descriptors when combined with SURF features, in the domain of image search. The experiments show that when SURF features are used as local image descriptors, VLAD attains better performance compared to using SIFT features. We also study how the average number of local image descriptors extracted per image affects the performance and show that by controlling this number we are able to adjust the trade off between feature extraction time and search accuracy. Finally, we examine the retrieval performance of the proposed scheme with varying levels of distractor images.

1. INTRODUCTION

VLAD (vector of locally aggregated descriptors) [1] has recently emerged as an alternative to the bag-of-words (BOW) image representation method. VLAD transforms a variable-size set of local image descriptors into a compact, fixed-size vector representation. The method has been shown to significantly outperform BOW and to be comparable with the more sophisticated Fisher Vector (FV) [2] method in terms of search accuracy, when a same dimensional representation is used. Furthermore, VLAD is cheaper to compute than BOW and it has been shown that its dimensionality can be significantly reduced by principal component analysis (PCA) without a significant loss in accuracy. The above characteristics make it ideal for large-scale image search, given that the produced vectors will be subsequently indexed.

Originally, VLAD was proposed and evaluated using SIFT features [3]. So far, SIFT features have shown excellent performance and are considered state-of-the-art in the domain of image search. However, a major drawback is their slow computation time which makes the VLAD+SIFT combination unsuitable for large-scale image search applications, especially those that require real-time responses using limited computational power such as landmark or product recognition on mobile devices. On the other hand, Speeded-Up Robust Features (SURF) [4] are primarily designed for speed and at the same time have shown comparable performance to SIFT in image search applications. In this paper, we study the performance of VLAD when combined with SURF features. The results show that the VLAD+SURF combination is better than VLAD+SIFT in terms of search accuracy and at the same time the vectors can be computed much faster. We also study the effect that the number of local image descriptors extracted from each image has on both vector generation time and accuracy. Finally, we examine how the retrieval performance of the proposed scheme is affected under the presence of varying levels of distractor images.

The next section discusses VLAD and SURF in more detail, while section 3 shows the results of the empirical evaluation. Finally section 4 concludes our study.

2. BACKGROUND

BOW is the most popular approach of aggregating a set L of d-dimensional local descriptors $x = [x_1, ..., x_d]$ into a single fixed-length vector representation v. In BOW, a codebook of k visual words is first created, usually by performing k-means clustering into a large set of local descriptors. Given an input image, each local descriptor extracted from this image is assigned to the closest visual word (centroid). Essentially, BOW represents the histogram of the number of local descriptors assigned to each centroid. Therefore, the produced vector is k-dimensional. In this paper, we utilize an alternative aggregation method named VLAD which has been recently proposed as an extension of BOW. As in BOW, a codebook $C = \{c_1, ..., c_k\}$ of k visual words is first created using k-means and each descriptor is associated to its closest centroid $NN(x)$. Then, instead of just recording the number of local descriptors assigned to each centroid, VLAD accumulates the differences $x - c_i$ of the vectors x assigned to c_i into a vector:

$$v_i = \sum_{x:NN(x)=c_i} x - c_i$$

The final VLAD vector v is the concatenation of all d-dimensional vectors v_i and is therefore kd-dimensional. Intuitively, the difference between BOW and VLAD is that BOW records the number of image descriptors associated with each center while VLAD records their position relatively to the center. As a final step, power and L_2 normalization are performed. Lately, [5] showed that VLAD is a simplified extreme case of the FV method. Empirical results in the domain of image search indicate that both VLAD and FV provide excellent results compared to BOW and achieve these results using as few as $k = 64$ visual words which results in $64 * d$-dimensional VLAD vectors. However, this dimensionality is still prohibitive for large-scale search applications. For this reason [1] proposes a dimensionality reduction step which results in even smaller vectors. To this end, PCA is used to define a $d' \times d$ matrix M which maps a vector $v \in R^d$ into a transformed vector $v' = Mv \in R^{d'}$. Importantly, it has been shown that this dimensionality reduction is able to reduce the dimensionality of the vectors by an order of magnitude (e.g. 4096 to 512) with negligible impact in accuracy.

So far, VLAD was used to aggregate 128-dimensional SIFT descriptors in [1] and SIFT descriptors reduced to 64 dimensions with PCA in [5]. However, while the aforementioned papers are dealing with the problem of efficient large-scale image search, they overlook the fact that the significant amount of time required for the extraction of SIFT descriptors can outweigh the important gains in response time achieved through the dimensionality reduction and subsequent indexing of the vector representations of the images. Motivated by this limitation, we investigate whether the remarkable performance of the method is retained when efficiently computed features such as SURF are used. SURF features are based on a high-performance scale- and rotation-invariant interest point detector and descriptor. SURF are claimed to approximate or even outperform SIFT with respect to repeatability, distinctiveness and robustness and at the same time can be computed several times faster. The reduced computation time is mainly due to the use of integral images for image convolutions and the use of a simplified, fast Hessian matrix-based measure for the detector. Furthermore, the descriptor is only 64-dimensional which is a further advantage compared to the standard SIFT.

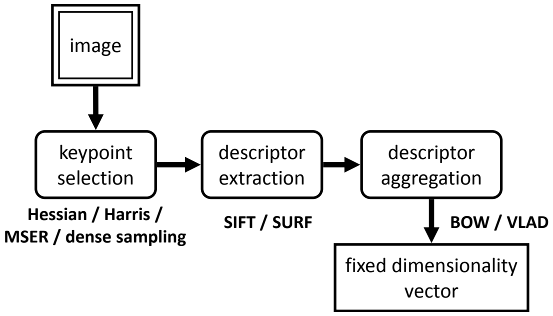

Fig. 1. The steps of the vector generation process.

Fig. 1 illustrates the various steps involved in the process that transforms an input image into a fixed dimensionality vector and some of the alternative methods to tackle each step.

3. EXPERIMENTAL STUDY

In this section we provide a comparison of VLAD with SURF features against VLAD with SIFT features. We also evaluate the performance when a different number of features are extracted from each image and we finally study the robustness of the representation when distractor images are added to the image collections.

3.1. Evaluation Setup and Datasets

All the images used in this evaluation are first scaled to a maximum size of 1024×768 pixels prior to feature extraction. To extract SURF features we used the BoofCV[1] library. The library provides two SURF implementations namely *surf* and *surfm*. We used the second implementation as in preliminary experiments we noticed significantly better accuracy results with a small increase in computation time. In all experiments except for those reported in subsection 3.3, SURF descriptions were computed around all the detected interest points as described in [4]. For the learning stages (codebook, PCA computation) we used an independent dataset[2] containing 70K Flickr images. Specifically, we created a codebook of $k = 64$ words by applying k-means on a randomly chosen sample of 1M features from around 70M features extracted from the independent dataset. PCA was learned on a sample of 40K images from the independent dataset for which we generated $64 * 64 = 4096$ dimensional VLAD vectors using the previously learned codebook. Finally, the VLAD vectors were power and L_2 normalized as suggested in [5].

To evaluate the retrieval accuracy we use mean Average Precision (mAP). The datasets that we use in our experiments are the following:

- The **Holidays** dataset [6] contains 1491 holiday images, 500 of which are used as queries.
- The **Oxford** dataset [7] consists of 5063 images collected from Flickr by searching for particular Oxford landmarks. 55 of the images are treated as queries.
- The **Paris** dataset [8] consists of 6412 images collected from Flickr by searching for particular Paris landmarks. 55 of the images are treated as queries.
- The **Flickr200K** dataset is a subset of 200K images taken from the 1 million MIR Flickr dataset. We merge this dataset with the other datasets to evaluate the accuracy when distractor images are added.

3.2. VLAD+SIFT vs VLAD+SURF

Table 1 shows the results of using VLAD to aggregate SURF vs SIFT features in Holidays and Oxford and also results of

[1] http://boofcv.org

[2] Kindly provided by Hervé Jégou (herve.jegou@inria.fr)

Table 1. Comparison of VLAD+SIFT with VLAD+SURF on Holidays and Oxford. d denotes the dimensionality of the vectors used. $d = 4096$ corresponds to the full vectors while $d \leq 4096$ corresponds to PCA-reduced vectors.

DATASET	DESCRIPTOR	$d = 4096$	$d = 2048$	$d = 1024$	$d = 512$	$d = 128$	$d = 64$	$d = 32$
HOLIDAYS	SIFT [5]	55.6	57.6		59.8	55.7	52.3	48.4
	SURF	**66.2**	**67.5**	68.9	**69.1**	**68.5**	**65.2**	**59.9**
OXFORD	SIFT [5]	30.4				25.7		
	SURF	**32.6**	33.3	34.1	33.1	**28.2**	25.1	20.8
PARIS	SURF	42.2	43.0	43.5	43.0	39.8	36.6	32.6

the VLAD+SURF combination obtained on Paris. Note that the results of the VLAD+SIFT combination are taken from [5] (hence the missing values from some entries) where SIFT features reduced to 64 dimensions by PCA were used and were found to perform better than standard 128 dimensional SIFT. We can see that VLAD+SURF largely improves the accuracy of VLAD+SIFT across all dimensions in Holidays. Remarkably, using 32 dimensional vectors we obtain a mAP score of 59.9 which is larger than the mAP score obtained using 4096 dimensional VLAD+SIFT vectors. In Oxford, the VLAD+SURF combination is again better using both 4096 and 128 dimensional vectors.

3.3. Tuning the Average Number of Features per Image

In this set of experiments we study how the search accuracy is affected when we alter the number of features extracted per image on average. To control this number we impose a threshold to the maximum number of features extracted per scale and compute descriptions around only the N most salient of the interest points detected in each scale. Since the number of interest points detected per scale decays very quickly [4], this threshold has the effect of pruning interest points mainly from the smaller scales thus balancing the distribution of features among scales. Figures 2 and 3 show the results on Holidays and Oxford respectively using both full and PCA-reduced vectors (we do not report results on Paris due to space limitations). The N parameter is tuned to take around 250, 500, 1000 and 2000 total features per image on average. We also report results when all the detected features are used as in the original version of SURF.

By looking at the figures we observe that extracting descriptions from all the detected interest points is beneficial only for low dimensional representations. In Holidays, extracting as few as 1000 features per image yields near optimal results for higher dimensional vectors (4096-1024) while the best results are obtained when 2000 features are extracted. When the dimensions are reduced below 512, we notice that vectors coming from the maximum number of features lead to better results but still vectors coming from 2000 features are near optimal. A similar trend is observed in the other datasets, indicating that with the proposed way of pruning SURF features we are able not only to accelerate the extraction process but also to improve the quality of the representation. A reasonable trade-off between accuracy, speed and descriptor size is provided using 512 dimensional VLAD vectors extracted by aggregating around 2000 SURF features per image.

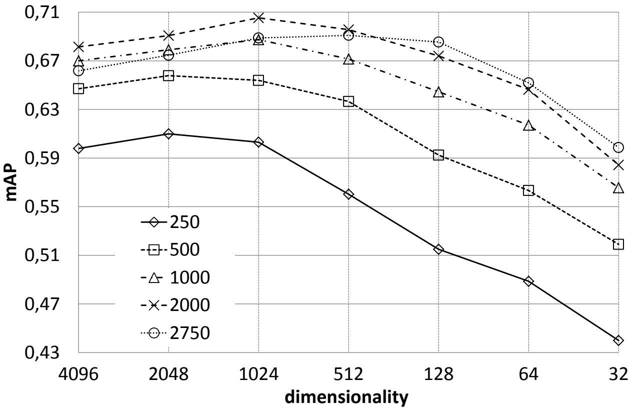

Fig. 2. Comparison of different average numbers of SURF features extracted per image (mAP on Holidays).

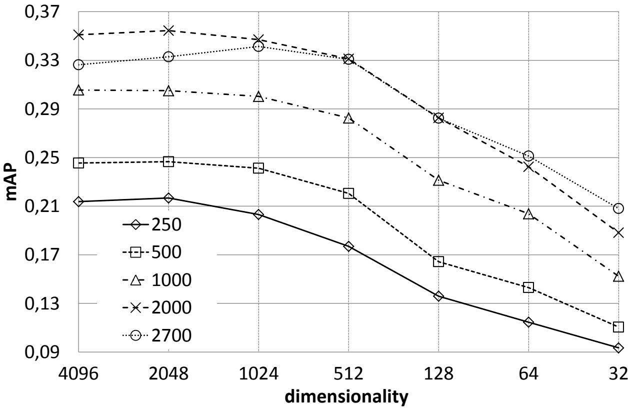

Fig. 3. Comparison of different average numbers of SURF features extracted per image (mAP on Oxford).

Table 2 reports the average amount time required to extract SURF features from 1024×768 sized images under each setting. From the results, it is clear that by tuning the num-

Table 2. Average amount of time required for SURF extraction when different numbers of features are extracted. Timing experiments have been performed on a single i5 2.4 GHz processor core.

# FEATURES:	250	500	1000	2000	2700
AVG. TIME (MS):	205	338	495	825	993

ber of extracted features we can adjust the trade-off between search accuracy and extraction time.

3.4. Adding Distractor Images

In this section we want to study how the accuracy is affected when distractor images are added to the collections. Table 3 shows the results on Holidays, Oxford and Paris when 5K, 50K and 200K distractor images are added. We present results using 64, 128 and 512 dimensional vectors as our 2GB of RAM would not be sufficient for larger vectors. We observe that in all datasets, 512 dimensional representations are able to retain a high percentage of the initial accuracy and are more robust to the addition of distractor images compared to lower dimensional (d=64, d=128) representations. However, there is large variation in performance among different datasets. For instance, we can see that when 64 dimensional vectors are used in Holidays, the addition of 200K distractor images causes a 21% drop in accuracy while in Paris the addition of the same number of distractors causes a much higher drop even for 512 dimensional vectors.

Table 3. Testing the robustness of the representation by adding distractor images.

DATASET	$d = 64$	$d = 128$	$d = 512$
HOLIDAYS	65.2	68.5	69.1
+5K	60.2	64.5	67.7
+50K	55.0	60.3	63.6
+200K	51.6 -21%	57.2 -16%	61.9 -10%
OXFORD	25.1	28.2	33.1
+5K	23.4	26.7	32.5
+50K	19.6	23.4	30.3
+200K	16.5 -34%	20.5 -27%	28.2 -15%
PARIS	36.6	39.8	43.0
+5K	29.9	33.6	40.4
+50K	21.1	25.1	33.9
+200K	16.9 -54%	20.7 -48%	28.6 -33%

4. CONCLUSIONS

The paper presented and evaluated the novel combination of the VLAD method for image representation with SURF local features in the domain of image search. The results show that when VLAD vectors are generated from SURF features, better performance is attained compared to using SIFT features. We also evaluated the impact of the average number of extracted SURF features per image on the quality of the vectorized representation. The reported results indicate that extracting all the detected features, not only slows down the extraction but also negatively affects the accuracy, especially for higher dimensional VLAD vectors. Overall, we have presented and evaluated an efficient, accurate and robust instantiation of an image representation methodology which is expected to be of practical use in real-time large-scale search applications. In the future, we plan to compare the performance of VLAD+SURF to the performance of VLAD+SIFT after a product quantization step is employed and to investigate the potential of improving retrieval quality by combining BOW and VLAD representations. We also plan to conduct a more in-depth analysis on why the proposed way of pruning SURF features leads to improved performance.

5. ACKNOWLEDGEMENTS

This work was supported by the SocialSensor FP7 project, partially funded by the EC under contract number 287975.

6. REFERENCES

[1] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[2] F. Perronnin and C. Dance, "Fisher Kernels on Visual Vocabularies for Image Categorization," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[3] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, 2004.

[4] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer Vision and Image Understanding*, 2008.

[5] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, and C. Schmid, "Aggregating local image descriptors into compact codes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011.

[6] H. Jégou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *European Conference on Computer Vision*, 2008.

[7] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[8] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Lost in quantization: Improving particular object retrieval in large scale image databases," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2008.