Graph Fusion Using Global Descriptors for Image Retrieval

Tomás Mardones[1(✉)], Héctor Allende[1], and Claudio Moraga[2,3]

[1] Universidad Técnica Federico Santa María, CP 110-V, Valparaíso, Chile
tomas.mardones@alumnos.usm.cl
[2] European Centre for Soft Computing, 33600 Mieres, Spain
hallende@inf.utfsm.cl
[3] Faculty of Computer Science, TU Dortmund University, Dortmund, Germany
claudio.moraga@udo.edu

Abstract. This paper addresses the problem of content-based image retrieval in a large-scale setting. Recently several graph-based image retrieval systems to fuse different representations have been proposed with excellent results, however most of them use at least one representation based on local descriptors that does not scale very well with the number the images, hurting time and memory requirements as the database grows. This motivated us to investigate the possibility to retain the performance of local descriptor methods while using only global descriptions of the image. Thus, we propose a graph-based query fusion approach -where we combine several representations based on aggregating local descriptors such as Fisher Vectors- using distance and neighborhood information to evaluate the individual importance of each element in every query. Performance is analyzed in different time and memory constrained scenarios. Experiments are performed on 3 public datasets: the UKBench, Holidays and MIRFLICKR-1M, obtaining state of the art performance.

Keywords: Fisher vector · Graph fusion · Large scale image retrieval · Global descriptors

1 Introduction

Content-based image retrieval (CBIR) is an important area of research in Multimedia, since it is linked to numerous image applications, especially web and mobile image search. Given a query image, the problem consists in finding the most similar images in a database. The image presentation that has received the most attention corresponds to the Bag of Words (BoW) representation [12]. Still it has an important limitation regarding the amount of images it can handle at a time due to its time response and memory usage, becoming impractical when working with a 100M image database (a database with 100 million images).

To overcome the database size limitation, the Fisher Vector (FV) and Vector of Locally Aggregated Descriptors (VLAD) were proposed [7] as a global image

A. Pardo and J. Kittler (Eds.): CIARP 2015, LNCS 9423, pp. 290–297, 2015.
DOI: 10.1007/978-3-319-25751-8_35

descriptor being able to leverage the advantages of powerful local descriptors like RootSIFT. A global descriptor usually implies that the memory usage per image is fixed and is represented by a dense vector. Dense vectors can have their dimension reduced by powerful methods, such as Principal Component Analysis (PCA) and Optimized Product Quantization (OPQ) [2].

To improve the performance of image retrieval systems, global descriptors have been combined with local descriptors representations [17]. Some of these systems achieve State of the Art performance, but the presence of local descriptors representations makes them unsuitable to very large scale image retrieval. Only a few works have used exclusively combination of global descriptors [3,8] and none of them, in our knowledge, have explored the query results fusion area.

Therefore in this work we propose a new unsupervised method to combine query results using only Fisher Vectors as image representations with the objective of meeting a low memory requirement and to enhance the performance achieved by the use of global descriptors.

The rest of the paper is organized as follows. Section 2 reviews relevant work regarding fusion methods using global descriptors. In Section 3 our proposal is detailed and its experimental results are discussed in Section 4. Concluding remarks are given in Section 5.

2 Related Work

2.1 Feature Combination

To build a large scale image retrieval system it is important that a compact image representation is used in addition to the descriptors fusion, such as BoW, FVs or GIST [3,8,17]. In particular, in the work of Zhang et al. [17] a graph method is employed to combine the individual ranking list from a BoW, a GIST and a global color representation. Only a few works have explored the combination of exclusively global descriptors for large-scale image retrieval [3,8]. Gordo et al. [3] proposed to use category-level labels of image classification datasets to learn sub-spaces to reduce the dimension of two concatenated Fisher Vectors based on SIFT and statistical color features respectively. On the other hand, Mardones et al. [8] used three concatenated Fisher Vectors based only on SIFT descriptors, but varying the sampling method used to obtain them, demostrating that the use of different sampling methods is an important way to introduce diversity in the representations. Both works achieved an important boost in performance compared to results obtained with the individual Fisher Vectors employed.

2.2 Graph Based Methods in Image Retrieval

Utilizing graphs is a natural way to introduce the neighborhood relationships when using several representations as noted in several works [11,17]. The closest inspiring works to ours, regarding the fusion method, include [11] and [17]. Qin et al. work [11] introduced a simple, albeit effective, method based on the analysis of

the K-reciprocal nearest neighbor structure in the image space with the objective of re-ranking the retrieval images. Zhang et al. [17] proposed an unsupervised graph fusion method, capable of combining any type of image representations, relying only on a set of ranking list obtained, by exploiting the neighborhood structure. In contrast, since we use only Fisher Vectors we concentrate on how to leverage the use of similar representations - as they have comparable distances - and their neighborhood structures to obtain a new ranking list.

3 Proposed Approach

3.1 Individual Image Representations

As previously stated, Fisher Vectors (FVs) are the selected global descriptors to represent the images in this work. This is a method that aggregates local descriptors of an image to build a vectorial representation of it by comparing the image descriptors with a general probabilistic distribution model of them. To model the distribution of the local descriptors a Gaussian Mixture Model (GMM) is used [7] and the Fisher Vector of an image is the derivative of the likelihood of this image descriptors distribution with respect to the learnt GMM parameters. The FV's part corresponding to the GMM mean and variance parameters will be referred to as the mean and variance component of the Fisher Vector respectively.

3.2 Basic Definitions

In the next subsections a few definitions will be shared. To avoid redundancy or confusion, they will be written here. Most of the notation is similar to the one in [17] for consistency.

Denote q and d as the query image and a database image respectively, and i as either of them. Given a distance function $\text{dist}(\cdot, \cdot)$ between images representation vectors, the rank list corresponds to the sorted list of candidate relevant images by the distances between the query image and every database image. Since this list can be very large depending on the number of database images, normally a short rank list - called shortlist - of the first L images of the rank list will be used. Now it is possible to define the K nearest neighbours (NN) of an image i as $N_K(i)$, where this corresponds to the top-K candidates obtained using i as the query. Finally we define the reciprocal neighbour relation between to images i and i' as:

$$R_K(i, i') = i \in N_K(i') \wedge i' \in N_K(i). \tag{1}$$

3.3 Preprocessing Steps for Ranking Reordering and Similarity Computing

Every representation share a common space -being all of them Fisher Vectors- though their distances are not directly comparable since they are based on different descriptors. To take advantage of this, it is important to normalize the

distances list in some way. To do this, the influence of arbitrary long distances is limited by using the query adaptive criterion C_a to transform the distances into similarities [6]. C_a works by replacing the distance of a query and an image by the difference between the distance of the k-th NN match of the query and this distance (or 0 if the difference is negative). This has the effect of aligning the rank-distance curves by a translation using as a reference the distance to the k-th NN match. However, the highest similarity is not bounded, so every non-zero value is divided by the highest similarity. This will be called bounded query adaptive criterion C_{ba}. $C_{ba}^{L}(q,d)$ will be used to transform the distances between the query and its L-nearest-neighbors (rank shortlist elements) into similarities.

An additional step taken before the graph construction is the use of the *maximum reciprocal rank* (MRR) algorithm [1]. The objective is to increase the number of reciprocal neighbors in the ranking's first positions, thus determining a higher quality shortlist. Using this method, a new ranking list is build inserting first the reciprocal neighbors according to the worst ranking position between the rank of the query in the specific neighbor rank shortlist and the rank of the neighbor the query rank shortlist. When no further reciprocal neighbors are found, the non-recriprocal neighbors are added in the same order of the original ranking list. For further details refer to [1].

3.4 Graph Construction and Fusion

For each representation, a weight undirected graph $G = (V, E, w)$ centered at the query q is initialized, where the nodes are the images. The graph will be constructed using nodes from levels 0, 1 and 2. Each level indicates a set of conditions to select the nodes to add to the graph and the weight of its edge and K is a user defined parameter (in Section 4.4 it will be discussed further):

- *Level 0.* The only level 0 node is the query.
- *Level 1.* Every K-nearest neighbor d of the query that satisfies $R_K(q,d)$ is linked by an edge $(q,d) \in E$ to the query. The weight associated with this edge is $C_{ba}^{K}(q,d)$.
- *Level 2.* Each K-nearest neighbor i of the level 1 nodes is added to the graph, linked to its respective level 1 node. The attached edge weight is computed using the extended Jaccard similarity coefficient $\bar{J}(d,i)$ [1] between the neighborhoods of the level 1 image d and the level 2 image i,

Using $\bar{J}$, instead of the regular Jaccard similarity coefficient J, has the advantage of taking into account the rank of the neighbors, therefore if the K parameter is high and there is only a strong connection between a few nodes in the first rank positions (important nodes probably), the similarity coefficient will be still high making the method more stable respect to the K parameter.

To design the conditions of every level, a vast amount of possible configurations were tested, combining the use of shared neighborhood measurement (J or $\bar{J}$), reciprocal neighbors criterion ($R_K(i,i')$) and similarity measures (C_{ba}). The first two are strong indicators of the quality of the result, as it has been extensively studied in several works [1,11,17]. The similarity plays a fundamental role

when there exists a substancial similarity difference between the first images in the rank and the rest, giving a strong indicator of being a relevant image. Several strategies for the graph construction had similar results. The common factor observed was the combination of adding reciprocal neighbors as the level 1 nodes and using the similarities as their weights. Then adding a second layer of neighbors of the level 1 nodes with or without any condition and using their "neighborhood score" ($\bar{J}$) as their weights.

After obtaining a graph for every m-th representation $G^m = (V^m, E^m, w^m)$, they are fused as described in [17]: the final graph $G = (V, E, w)$ with $V = \cup_m V^m$, $E = \cup_m E^m$ and $w(i, i') = \sum_m w^m(i, i')$ for every existing edge. To obtain the final rank list from the graph, the *Ranking by Maximizing Weighted Density* is used [17].

4 Experiments

4.1 Databases and Evaluation Protocol

The following two public benchmarks are employed. INRIA Holidays [5] consists of 1,491 images of 500 scenes and objects. Each scene / object has a query image and the accuracy is measured as the Mean Average Precision (MAP). The University of Kentucky Benchmark (UKB) [9] consists of 10,200 images of 2,550 objects. Each image is used alternatively as a query to search within the 10,200 images (including itself) and the performance is measured as 4×recall@4 (called Kentucky Score, KS or N-S sometimes) averaged over the 10,200 queries. Therefore, the score goes from zero to four on this dataset.

Ten thousand images of the MIRFLICKR-1M dataset [4] are used to learn the GMM parameters and the PCA matrices of the different Fisher Vector representations, the rest are used as distractor images for the large-scale experiments.

4.2 Implementation Details

Several ROI detectors and descriptors will be employed to introduce diversity on the different Fisher Vectors. The base algorithm -independently of the detector and descriptor used- follows the same guidelines and parameters used in [7].

Features. Two types of descriptors are used in this work: 128-dimensional RootSIFT descriptors and Color descriptors [3]. Four sampling methods are employed to extract descriptors in most experiments: 3 scales dense sampling (D3), Hessian affine (HA), Hessian Laplace and Perdoch's Hessian affine variation (HAP) interest point detectors [15] [10]. RootSIFT is used with every sampling method, while the Color descriptor is used only with the dense sampling method. In the rest of the section we will loosely refer to the Fisher Vectors based on the RootSIFT descriptors sampled with the previously mentioned methods as HA, HL, HAP, D3M and D3V (M and V stand for mean and variance components respectively) and ColorM and ColorV as the Fisher Vectors based on the Color descriptors.

4.3 Scenarios to Consider

There are two principal scenarios of interest to evaluate the performance of the proposed method. The first one focuses on performance, sacrificing response time and memory usage, while the second focuses on balancing the three aspects. All the times reported were measured when processing with an Intel Core i5 2500S (using a single thread).

Performance Scenario (PS). As time is not a primary concern, 640x480 images are used for every representation. The FVs are just reduced to 1024 dimensions using PCA. Table 1 shows that the query time and the memory usage per image are not excellent, but sufficient depending on the application. The performance is better than the obtained in the next scenario.

Balanced Scenario (BS). A combination of 320x240 and 640x480 images is used. Every FV has its dimension reduced to 32 bytes using PCA and OPQ. In Table 1 it can be seen that this configuration reduces the response time to less than a second and, more important, the memory requirements to 792 bytes and 660 bytes on Holidays and UKBench respectively (each uses different representation mixtures). Given the low memory usage of this scenario, it will also be used for the large-scale experiments in Section 4.5.

4.4 Test Datasets Results

On Holidays and UKB the results are encouraging in both scenarios as seen in Table 1. In the performance scenario, the MAP and KS (on Holidays and UKB respectively) increase respect to their best individual representations (D3V and HA respectively) is 18.6% and 0.46 respectively. These results improve over the State of the Art in both datasets when considering methods based only on global descriptors, to the best of our knowledge, and are among the State of the Art in general, as seen in Table 2.

In the balanced scenario the improvement of MAP and KS is equally important over the best individual representations (D3V and ColorV), being 14.9% and 0.54 respectively. These results are close in both datasets to the best in the State of the Art. However the memory usage is much lower compared to the best performing methods, as it is possible to see in Table 2.

It is important to mention that the fine-tuning of the K parameter does not make a sustancial difference, but it should be adjusted depending on the expected number of relevant nearest neighbors. For example, the difference from using K=4 to K=15 results in a loss of MAP and KS of 2.4% and 0.08 respectively. In comparison, if Jaccard is used instead of $\bar{J}$, the loss increases to 3.3% and 0.15.

The use of MRR and C_{ba} also enhanced significantly the performance of the system. On Holidays and UKB, the improvement was of 2.8% and 0.05 KS.

4.5 Large-Scale Experiments

An important point of the proposal is its ability to scale due to the relatively small memory usage per image. Equally important is the performance stability as more

Table 1. Holidays and UKB Results. Mean average precision (Kentucky Score for UKB), memory usage per image and total query time (including feature extraction).

Scenario	Holidays			UKB		
	MAP	Memory	Query Time	KS	Memory	Query Time
Performance	86.0%	29 KB	1853 ms	3.85	29 KB	1875 ms
Balanced	80.2%	792 B	768 ms	3.75	660 B	777 ms
Balanced + 1M	76.9%	792 B	950 ms	3.70	660 B	918 ms

Table 2. Comparison with the State of the Art. Results from Holidays, UKB and average memory per image, if available.

	[16]	[14]	[13]	[17]	[11]	[3]	Ours PS	Ours BS
Holidays, MAP	80.2%	84.1%	88.0%	84.6%	-	78.3%	86.0%	80.2%
UKB, KS	-	-	-	3.83	3.67	3.36	3.85	3.75
Memory	8192 B	≈143 KB	-	≈20.2 KB	-	1024 B	≈29 KB	660-792 B

images are being integrated in the database. In Table 1 it can be observed that the total decrease of MAP and KS - with MIRFLICKR-1M's distractor images - is 3.5% and 0.05 respectively. This shows that the proposed system is very robust to the number of images in the database. We believe that this robustness is due to the relevant neighbor consistency across the different representations, e.g. if a rank list is modified due to the insertion of a new image, there are several other rank lists that are not modified, being robust to insertions. Furthermore, a new insertion in the rank shortlist only implies that two image representations are within a short distance, but not a neighborhood similarity. In additional experiments it was seen that the decrease in precision was more significant with the insertion of the first 50K images, but after that is was much slower.

5 Conclusions

In this paper, a query fusion method for Fisher Vectors based on different features is presented. Multiple RootSIFT and Color features are extracted from the image using multiple sampling methods and a Fisher Vector is computed for each of them. Using the ranking and distance lists from every representation a graph is constructed weighting its edges considering a distance similarity measure and the neighborhood structure. Finally every graph is fused and a ranking list is obtained. Using this proposal, it is shown that by the use of solely global descriptors, State of the Art performance is achievable, while maintaining a fixed low memory usage per image.

Acknowledgements. This work was supported by the following research and fellowship grants: DGIP-UTFSM and MECESUP. The work of C. Moraga was partially supported by the Foundation for the Advance of Soft Computing, Mieres, Spain.

References

1. Delvinioti, A., Jégou, H., Amsaleg, L., Houle, M.E.: Image retrieval with reciprocal and shared nearest neighbors. In: Proc. International Conference on Computer Vision Theory and Applications, pp. 321–328, January 2014
2. Ge, T., He, K., Ke, Q., Sun, J.: Optimized product quantization. IEEE Trans. Pattern Analysis and Machine Intelligence **36**(4), 744–755 (2014)
3. Gordo, A., Rodriguez-Serrano, J.A., Perronnin, F., Valveny, E.: Leveraging category-level labels for instance-level image retrieval. In: Proc. CVPR, pp. 3045–3052 (2012)
4. Huiskes, M.J., Thomee, B., Lew, M.S.: New trends and ideas in visual concept detection: the MIR Flickr retrieval evaluation initiative. In: Proc. ACM International Conference on Multimedia Information Retrieval, pp. 527–536. ACM (2010)
5. Jegou, H., Douze, M., Schmid, C.: Hamming embedding and weak geometric consistency for large scale image search. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 304–317. Springer, Heidelberg (2008)
6. Jégou, H., Douze, M., Schmid, C.: Exploiting descriptor distances for precise image search. Research report, INRIA, Jun 2011
7. Jégou, H., Perronnin, F., Douze, M., Sánchez, J., Pérez, P., Schmid, C.: Aggregating local image descriptors into compact codes. IEEE Trans. Pattern Analysis and Machine Intelligence **34**(9), 1704–1716 (2012)
8. Mardones, T., Allende, H., Moraga, C.: Combining fisher vectors in image retrieval using different sampling techniques. In: Proc. International Conference on Pattern Recognition Applications and Methods, pp. 128–135, January 2015
9. Nister, D., Stewenius, H.: Scalable recognition with a vocabulary tree. In: Proc. CVPR, pp. 2161–2168 (2006)
10. Perdoch, M., Chum, O., Matas, J.: Efficient representation of local geometry for large scale object retrieval. In: Proc. CVPR, pp. 9–16, Jun 2009
11. Qin, D., Gammeter, S., Bossard, L., Quack, T., Van Gool, L.: Hello neighbor: Accurate object retrieval with k-reciprocal nearest neighbors. In: Proc. CVPR, pp. 777–784, Jun 2011
12. Sivic, J., Zisserman, A.: Efficient visual search of videos cast as text retrieval. IEEE Trans. Pattern Analysis and Machine Intelligence **31**(4), 591–606 (2009)
13. Tolias, G., Avrithis, Y., Jégou, H.: To aggregate or not to aggregate: selective match kernels for image search. In: Proc. IEEE ICCV, pp. 1401–1408 (2013)
14. Tolias, G., Furon, T., Jégou, H.: Orientation covariant aggregation of local descriptors with embeddings. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part VI. LNCS, vol. 8694, pp. 382–397. Springer, Heidelberg (2014)
15. Tuytelaars, T., Mikolajczyk, K.: Local invariant feature detectors: A survey. Foundations and Trends in Computer Graphics and Vision **3**(3), 177–280 (2008)
16. Gong, Y., Wang, L., Guo, R., Lazebnik, S.: Multi-scale orderless pooling of deep convolutional activation features. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part VII. LNCS, vol. 8695, pp. 392–407. Springer, Heidelberg (2014)
17. Zhang, S., Yang, M., Cour, T., Yu, K., Metaxas, D.: Query specific rank fusion for image retrieval. IEEE Trans. Pattern Analysis and Machine Intelligence **37**(4), 803–815 (2015)