

Web scale image retrieval using compact tensor aggregation of visual descriptors

Romain Negrel, David Picard, Philippe-Henri Gosselin

▶ **To cite this version:**

Romain Negrel, David Picard, Philippe-Henri Gosselin. Web scale image retrieval using compact tensor aggregation of visual descriptors. IEEE MultiMedia, 2013, 20 (3), pp.24-33. 10.1109/MMUL.2013.14 . hal-00832760

HAL Id: hal-00832760
https://hal.science/hal-00832760

Submitted on 11 Jun 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Web scale image retrieval using compact tensor aggregation of visual descriptors

Romain Negrel, David Picard and Philippe-Henri Gosselin
ETIS/ENSEA - University of Cergy-Pontoise - CNRS, UMR 8051
6, avenue du Ponceau, BP44, F95014 Cergy-Pontoise, France
{*romain.negrel,picard,gosselin*}*@ensea.fr*

***Abstract*—The main issues of web scale image retrieval are to achieve a good accuracy while retaining low computational time and memory footprint. In this paper, we propose a compact image signature by aggregating tensors of visual descriptors. Efficient aggregation is achieved by preprocessing the descriptors. Compactness is achieved by projection and quantization of the signatures. We compare our method to other efficient signatures on a 1 million images dataset, and show the soundness of the approach.**

***Index Terms*—Image/video retrieval, Image Processing and Computer Vision.**

I. Introduction

With the globalization of internet, collections with tremendous amounts of images are available. For instance, more than 6 billions images were hosted on Flickr[1] in 2011. Images similarity search in these web scale databases is thus becoming a hot topic in the multimedia indexing community. Given a query image, image similarity search is to find similar images in a huge collection of images. Similar images are defined as the images with similar visual content (same object, same action, same scene, ...), without any meta data such as textual tags, time or location.

The two main problems of this task are the search time and the storage size of indexes. To index an image, common systems use a set of local visual descriptors extracted from images called "bag of descriptors". The main problem of bags of descriptors is their prohibitive storage cost. Many methods consist in computing a lightweight signature using the bag of descriptors.

In this paper, we propose a very compact signature which gives good performance in similarity search with a linear metric. Our signature is based on compressed aggregation of tensor products of local descriptors. In the first step, we perform a preprocessing on the descriptors. Then, we aggregate tensors of preprocessed descriptors. Finally, we compress the signature by projection in a well chosen subspace. Extra compression is achieved by binary quantization of the projected signatures.

The paper is organized as follows: First, we give an overview of the state-of-the-art to compute similarity between images. Then we detail our propositions in the third section. In section IV, we present results for similarity search tasks on well known web scale datasets and compare with recent methods. In the last section, we conclude and discuss the possible improvements of our method, as well as the ongoing challenges in web scale image indexing.

[1]As stated on Flickr's blog on August 4th 2011.
http://blog.flickr.net/en/2011/08/04/6000000000/

II. State-of-the-art

Most similarity search methods use a two steps scheme. In the first step, a set of local visual descriptors is extracted from the images. Regions of interest in the image can be selected by automatic point of interest detection, or by uniform sampling. The most commonly used visual descriptors are highly discriminant local descriptors [1] (HOG, SIFT, SURF, ...). The set of descriptors extracted from an image is called a *bag*. We denote by $\mathbf{B}_i = \{\mathbf{b}_{ri}\}_r$ the set of descriptors $\mathbf{b}_{ri} \in \mathbb{R}^D$ in image i. $\mathcal{B}$ is the union of $\mathbf{B}_i$ for all image i in the dataset.

In the second step, a similarity between two bags of descriptors is defined. There are two main approaches to compute such similarities. The first approach performs a straight matching between descriptors in bags, for instance using a voting approach. The second approach is to compute a signature (generally a single vector) from the bag of descriptors, and then to use similarity measures between vectors.

In both cases, the similarity measure is used to sort all images of the database according to a query image. To work with web-scale image databases, it is essential to have extremely fast similarity computation.

A. Voting based approaches

In the approaches based on voting, the descriptors of the query image are matched to the descriptors of the dataset $\mathcal{B}$. Each descriptor of the query votes for its k-Nearest Neighbors (k-NN) in $\mathcal{B}$. Then each image counts the number of votes obtained by its descriptors. The image with the most votes is the most similar image. The similarity score of bags $\mathbf{B}_j$ relative to a query $\mathbf{B}_i$ is thus obtained with the following equation:

$$\mathrm{k}(\mathbf{B}_i, \mathbf{B}_j) = \sum_{\mathbf{b}_{ri} \in \mathbf{B}_i} \mathrm{card}\left(\underset{\mathcal{B} \setminus \mathbf{B}_i}{k\text{-NN}}(\mathbf{b}_{ri}) \cap \mathbf{B}_j \right). \quad (1)$$

Naive k-NN search has a complexity linear with the number of descriptors in $\mathcal{B}$, which is prohibitive at web scale. Computation time can be saved using approximated k-NN search,

where a subset $\mathcal{B}'(\mathbf{b})$ of candidate is selected thanks to a sub-linear algorithm. A subset $\mathcal{B}'(\mathbf{b})$ is defined for each query descriptor $\mathbf{b}$ as:

$$\mathcal{B}'(\mathbf{b}) = \{\mathbf{b}_i \in \mathcal{B} | \mathbb{P}\left(\mathrm{d}(\mathbf{b}_i, \mathbf{b}) < R\right) > P\} \tag{2}$$

with R the distance threshold, P a probability of being similar and d a distance function.

Locality Sensitive Hashing (LSH) [2] uses hash functions to produce the descriptor subset. The hash function h is defined such that:

- if $\mathrm{d}(\mathbf{b}_i, \mathbf{b}) \leq R_1$ then $\mathbb{P}(\mathrm{h}(\mathbf{b}_i) = \mathrm{h}(\mathbf{b})) \geq P_1$,
- if $\mathrm{d}(\mathbf{b}_i, \mathbf{b}) \geq R_2$ then $\mathbb{P}(\mathrm{h}(\mathbf{b}_i) = \mathrm{h}(\mathbf{b})) \leq P_2$,
- $R_2 > R_1$,
- $P_1 > P_2$.

By properly choosing the (R_1, R_2, P_1, P_2) parameters, it is guaranteed that the descriptors that are colliding (same hash) have a high probability of being similar.

Another approach is to split the descriptor space with a hierarchical tree structure such that all elements of a leaf are very similar. Lejsek et al. [3] propose a method called Nearest Vector Tree (NV-Tree). In this method each node of the tree contains a subset of the descriptors, and each child node a splitting of this subset. The nearest neighbor candidates of query descriptor are all elements of the leaf to which it belongs.

Voting based approaches give good results in similarity search with very short response time. However, these approaches require the storage of all descriptors in $\mathcal{B}$ and also the index structure for approximate nearest neighbor search. In [4], the authors estimate around 100-500 bytes per descriptor for the LSH indexing. For web scale databases with more than 1 billion descriptors (around 1 million images), the storage cost of these approaches is prohibitive and not tractable.

B. Kernels on Bags approaches

Kernels on Bags approaches are an extension of kernel functions commonly used in machine learning. These approaches are similar to voting based approaches as they estimate the number of similar descriptors. Unlike voting based approaches, they use similarity functions to weight the vote. The similarity function between two descriptors is called minor kernel and is defined as:

$$\mathrm{k} : (\mathbb{R}^D, \mathbb{R}^D) \rightarrow [0, 1]. \tag{3}$$

The minor kernel is chosen such that, for similar descriptors $\mathrm{k}(\cdot, \cdot) \approx 1$ and for dissimilar descriptors $\mathrm{k}(\cdot, \cdot) \approx 0$.

In [5], the authors proposed to compute the sum of similarity of all possible pairings between elements of $\mathbf{B}_i$ and $\mathbf{B}_j$:

$$\mathrm{K}(\mathbf{B}_i, \mathbf{B}_j) = \sum_r \sum_s \mathrm{k}(\mathbf{b}_{ri}, \mathbf{b}_{sj}). \tag{4}$$

Thus the higher the number of similar descriptors the more the two bags are similar.

However, such kernel on bags produces a similarity of low variance. To overcome this problem, Lyu proposed in [6] to raise the minor kernel to power p. Therefore only highly similar descriptors are considered.

Kernels on bags approaches have good results in the fields of image retrieval and classification, but are rarely used in web scaled problems [7]. Indeed the computational cost of these approaches is prohibitive when the size of the bags becomes too large, especially with dense sampling extraction strategies. To compute the similarity between two bags of 10,000 descriptors, 100 million evaluations of the minor kernel have to be performed.

To address these computational problems, only the most similars descriptors of the bags can be considered, like in voting based approaches. In [8] the problem is seen as the following kernel on bag:

$$\mathrm{K}_{fast}(\mathbf{B}_i, \mathbf{B}_j) = \sum_r \sum_s \mathrm{k}(\mathbf{b}_{ri}, \mathbf{b}_{sj}) f(\mathbf{b}_{ri}, \mathbf{b}_{sj}), \tag{5}$$

with $f(\cdot, \cdot)$ a indicator function based on k-NN:

$$f(\mathbf{b}_r, \mathbf{b}_s) = \begin{cases} 1 & \text{if } d(\mathbf{b}_r, \mathbf{b}_s) < R, \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

$f(\mathbf{b}_r, \mathbf{b}_s)$ is obtained by previously described methods such as LSH. These methods result in fast kernels on bags, but they have same problem of storage cost as voting based approaches.

C. Statistical approaches

Statistical approaches have been inspired by text retrieval methods. In these approaches, we assume a visual codebook composed of descriptor prototypes (called visual words) can be computed. A bag can then be described by a statistical analysis of occurrences of visual words. The visual codebook is generally computed by a clustering algorithm (*e.g.*, k-means) on a large set of descriptors. We denote by C the number of visual words in the codebook.

The first method of this kind, named Bag of Words (BoW)[9] counts the number of descriptors belonging to each cluster. The size of the signature is C.

Avila et al. [10] suggested an extension of BoW called Bag Of Statistical Sampling Analysis (BOSSA). This method aims to keep more information on the distribution of descriptors in the clusters. In this method, histograms of distances from centers of clusters are computed. The signature size is $C \times H$ with H the number of bins in distance histograms.

However, BoW approach is subject to codeword ambiguity. This problem arises when a descriptor lies at the boundary between two clusters or away from all the cluster centers. To solve this problem Gemert et al. [11] proposed a robust alternative to histograms using kernel density estimation (typically Gaussian functions) to smooth the local neighborhood of descriptors. This method allows for a soft assignment of a descriptor to several codewords. The size of the signature is C. These approaches obtain better results than BoW approaches.

D. Coding approaches

The coding approaches are borrowed from the telecommunications and signal processing communities. The main idea of these approaches is to use coding methods based on reconstruction problems [12] (notably used in data compression). In

most cases the encoding methods minimize a reconstruction error.

The signature is obtained with a two-step scheme. The first step consists in encoding each descriptor of the bag (coding step). The second step consists in aggregating all codes in a single vector (pooling step). Many coding functions have been proposed with different structural constraints on the code.

A sparsity regularization term is usually added in order to have good compression and aggregation properties on the code. Wang et al. [13] proposed a coding constraint such that similar descriptors are always coded with the same visual words by adding a Locality-constrained term:

$$q_{llc}(\mathbf{B}_i) = \arg\min_{C_i} \sum_r \|\mathbf{b}_{ri} - \mathbf{D}\mathbf{c}_{ri}\|^2 + \lambda \|\mathbf{d}_{ri} \odot \mathbf{c}_{ri}\|^2 \quad (7)$$

with $\mathbf{d}_{ri}$ a locality constraint and $\odot$ the Hadamard product.

The most common polling methods are:

- sum pooling : $\mathbf{c}_i = \sum_r \mathbf{c}_{ri}$
- max pooling : $\mathbf{c}_i = \max_r(\mathbf{c}_{ri})$

where "max" functions in a row-wise manner, returning a vector of size C.

E. Model Deviation approaches

Model Deviation approaches are based on a model of the descriptors space. The signature of a bag of descriptors is the deviation between the descriptors of the bag and the model.

Recently, Perronnin et al. [14] proposed a successful method called Fisher Vectors. The authors proposed to model the descriptors space by a probability density function denoted by u_λ of parameters λ. To describe the image, they compute the derivative of the log-likelihood of image descriptors to the model:

$$\mathcal{G}_\lambda^{\mathbf{B}_i} = \frac{1}{T} \nabla_\lambda \log u_\lambda(\mathbf{B}_i). \quad (8)$$

The model used is a Gaussian Mixture Model (GMM) of parameters $\boldsymbol{\mu}_c$ and σ_c. Elements of the Fisher Vector for each Gaussian c can be written as:

$$\mathcal{G}_{\boldsymbol{\mu},c}^{\mathbf{B}_i} = \frac{1}{T\sqrt{\omega_c}} \sum_r \gamma_c(\mathbf{b}_{ri}) \left(\frac{\mathbf{b}_{ri} - \boldsymbol{\mu}_c}{\sigma_c} \right), \quad (9)$$

$$\mathcal{G}_{\boldsymbol{\sigma},c}^{\mathbf{B}_i} = \frac{1}{T\sqrt{\omega_c}} \sum_r \gamma_c(\mathbf{b}_{ri}) \left[\frac{(\mathbf{b}_{ri} - \boldsymbol{\mu}_c)^2}{\sigma_c^2} - 1 \right]. \quad (10)$$

Where $\mathbf{b}_{ri}$ are the descriptors of image i, $(\omega_c, \boldsymbol{\mu}_c, \sigma_c)$ are the weight, mean and standard deviation of Gaussian c, and $\gamma_c(\mathbf{b}_{ri})$ the normalized likelihood of $\mathbf{b}_{ri}$ to Gaussian c. The final descriptor is obtained by concatenation of $\mathcal{G}_{\boldsymbol{\mu},c}^{\mathbf{B}_i}$ and $\mathcal{G}_{\boldsymbol{\sigma},c}^{\mathbf{B}_i}$ for all Gaussians. Fisher Vectors achieve very good results [14]. However, Fisher Vectors are limited to the simple model of mixtures of Gaussians with diagonal covariance matrices. Moreover, the GMM algorithm is computationally very intensive.

Jegou et al. [15] proposed a simplified version of Fisher Vector by aggregating local descriptors, called Vectors of Locally Aggregated Descriptors (VLAD). They proposed to model the descriptors space by a small codebook obtained by clustering a large set of descriptors. The model is simply the sum of all centered descriptors $\mathbf{B}_{ci} = \{\mathbf{b}_{rci}\}_r \subseteq \mathbf{B}_i$ from image i and cluster c:

$$\boldsymbol{\nu}_{ci} = \sum_r \mathbf{b}_{rci} - \boldsymbol{\mu}_c \quad (11)$$

with $\boldsymbol{\mu}_c$ the center of cluster c. The final signature is obtained by a concatenation of $\boldsymbol{\nu}_c$ for all c. The signature size is $D \times C$.

Picard et al. [16] proposed an extension of VLAD by aggregating tensor products of local descriptors, called Vector of Locally Aggregated Tensors (VLAT). They proposed to use the covariance matrix of the descriptors of each cluster. Let us denote by "$\boldsymbol{\mu}_c$" the mean of cluster c and "$\mathcal{T}_c$" the covariance matrix of cluster c with $\mathbf{b}_{rci}$ descriptors belonging to cluster c:

$$\boldsymbol{\mu}_c = \frac{1}{|c|} \sum_i \sum_r \mathbf{b}_{rci} \quad (12)$$

$$\mathcal{T}_c = \frac{1}{|c|} \sum_i \sum_r (\mathbf{b}_{rci} - \boldsymbol{\mu}_c)(\mathbf{b}_{rci} - \boldsymbol{\mu}_c)^\top, \quad (13)$$

with $|c|$ being the total number of descriptors in cluster c.

For each cluster c, the signature of image i is the sum of centered tensors of centered descriptors belonging to cluster c:

$$\mathcal{T}_{ic} = \sum_r (\mathbf{b}_{rci} - \boldsymbol{\mu}_c)(\mathbf{b}_{rci} - \boldsymbol{\mu}_c)^\top - \mathcal{T}_c. \quad (14)$$

Each $\mathcal{T}_{ic}$ is flattened into a vector $\mathbf{v}_{ic}$. The VLAT signature $\mathbf{v}_i$ for image i consists of the concatenation of $\mathbf{v}_{ic}$ for all clusters:

$$\mathbf{v}_i = (\mathbf{v}_{i1} \dots \mathbf{v}_{iC}). \quad (15)$$

For better results, normalization steps are added:

$$\mathbf{x}_i = \frac{\mathbf{v}'_i}{\|\mathbf{v}'_i\|}, \quad \forall j, \mathbf{v}'_i[j] = \text{sign}\left(\mathbf{v}_i[j]\right) \left|\mathbf{v}_i[j]\right|^\alpha, \quad (16)$$

with α typically set to 0,5. $\mathbf{x}_i$ is the normalized VLAT signature.

As the $\mathcal{T}_{ic}$ matrices are symmetric, only the diagonal and the upper part are kept while flattening $\mathcal{T}_{ic}$ into a vector $\mathbf{v}_{ic}$. The size of the signature is then $C \times \frac{D \times (D+1)}{2}$.

III. Compact VLAT

In this paper, we propose to improve VLAT by increasing their discriminative power while reducing their size. The first improvement consists in preprocessing the descriptors to optimize the model $(\boldsymbol{\mu}_c, \mathcal{T}_c)_c$. Then we present a method to reduce the size of the VLAT signatures while preserving the dot product. Our dimensionality reduction is based on linear projections that have been made more efficient thanks to the model optimization.

A. PCA cluster-wise of VLAT

The signature is composed of deviations between covariance matrices of the clusters and covariance matrices of the image descriptors. To optimize this deviation, we propose to perform a Principal Component Analysis (PCA) within each cluster.

First, we compute the Takagi decomposition of the covariance matrix of each cluster c:

$$\mathcal{T}_c = \mathbf{V}_c \mathbf{D}_c \mathbf{V}_c^\top, \quad (17)$$

where $\mathbf{D}_c$ is a real non-negative diagonal matrix (eigenvalues), and $\mathbf{V}_c$ is unitary (eigenvectors). Then we project the centered descriptors belonging to c on the eigenvectors:

$$\mathbf{b}'_{rci} = \mathbf{V}_c^\top(\mathbf{b}_{rci} - \boldsymbol{\mu}_c). \tag{18}$$

Combining eq.(18) and eq.(14), we get:

$$\begin{aligned}\mathcal{T}_{ic} &= \mathbf{V}_c^\top \left(\sum_r (\mathbf{b}_{rci} - \boldsymbol{\mu}_c)(\mathbf{b}_{rci} - \boldsymbol{\mu}_c)^\top - \mathcal{T}_c \right) \mathbf{V}_c \\ &= \sum_r \mathbf{V}_c^\top \left((\mathbf{b}_{rci} - \boldsymbol{\mu}_c)(\mathbf{b}_{rci} - \boldsymbol{\mu}_c)^\top \right) \mathbf{V}_c - \mathbf{D}_c \\ &= \sum_r \left(\mathbf{V}_c^\top (\mathbf{b}_{rci} - \boldsymbol{\mu}_c) \right) \left(\mathbf{V}_c^\top (\mathbf{b}_{rci} - \boldsymbol{\mu}_c) \right)^\top - \mathbf{D}_c.\end{aligned}$$

The new VLAT signature of image i in cluster c is the sum of tensors of projected descriptors $\mathbf{b}'_{rci}$ belonging to cluster c, centered by $\mathbf{D}_c$:

$$\mathcal{T}_{ic} = \sum_r \mathbf{b}'_{rci} \mathbf{b}'^\top_{rci} - \mathbf{D}_c. \tag{19}$$

The optimized VLAT signature is obtained by the same steps of flattening, concatenation and normalization as the standard signature. This optimization has the very interesting property that most of the variance is concentrated among the first dimensions of each cluster.

B. Compact VLAT

We propose to reduce drastically the size of the VLAT signature while retaining its discriminative power. We seek a linear projection into a subspace in which the original similarity between two signatures is retained. Hence, we want to solve the following problem:

$$\begin{aligned}\mathbf{P}_N = &\arg\min_{\mathbf{A}} \sum_{\mathbf{x}_i \in \mathcal{S}} \sum_{\mathbf{x}_j \in \mathcal{S}} \left(\langle \mathbf{x}_i | \mathbf{x}_j \rangle - \langle \mathbf{A}^\top \mathbf{x}_i | \mathbf{A}^\top \mathbf{x}_j \rangle \right)^2 \\ &\text{s.t. } \mathbf{A} \in \mathcal{M}_{S,N} \text{ with } N < L \ll W\end{aligned}$$

with $\mathcal{S}$ a training set of L images, N the size of subspace and W the size of VLAT signature. We solve this problem by performing a low rank approximation of the Gram matrix and computing the linear projectors of the associated subspace.

We compute the Gram matrix of a training set $\mathcal{S}$ ($L \times L$):

$$\mathbf{G}_{ij} = (\mathbf{x}_j^\top \mathbf{x}_i)_{ij} \tag{20}$$

Then, we perform the Takagi factorization of $\mathbf{G}$:

$$\mathbf{G} = \mathbf{U}\mathbf{L}\mathbf{U}^\top \tag{21}$$

$$\mathbf{L} = \operatorname{diag}(\lambda_1, \ldots, \lambda_L) \tag{22}$$

$$\mathbf{U} = (\mathbf{u}_1, \ldots, \mathbf{u}_L) \tag{23}$$

with $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_L \geq 0$ and $\mathbf{u}_i$ the eigenvector associated with the eigenvalue λ_i . We denote by $\mathbf{L}_N$ the matrix with the N largest eigenvalues of $\mathbf{L}$ on the diagonal:

$$\mathbf{L}_N = \operatorname{diag}(\lambda_1, \ldots, \lambda_N) \tag{24}$$

and we denote by $\mathbf{U}_N$ the matrix of the N first eigenvectors of $\mathbf{U}$:

$$\mathbf{U}_N = (\mathbf{u}_1, \ldots, \mathbf{u}_N). \tag{25}$$

The approximated Gram matrix is then:

$$\mathbf{G}_N = \mathbf{U}_N \mathbf{L}_N \mathbf{U}_N^\top \tag{26}$$

We compute the projection matrix signatures in the subspace:

$$\mathbf{P}_N = \mathbf{X}\mathbf{U}_N \mathbf{L}_N^{-1/2}. \tag{27}$$

For each image, we compute the projection of VLAT in the sub-space as:

$$\mathbf{y}_i = \mathbf{P}_N^\top \mathbf{x}_i. \tag{28}$$

$\mathbf{y}_i$ contains an approximate and compressed version of $\mathbf{x}_i$. The subspace defined by the projectors preserves most of the similarity even for very a small dimension and for small training sets because the optimization of section III-A concentrated the information in a small number of dimensions. One can note this procedure is analog to that of a kernel PCA with a linear kernel.

For a more robust similarity, we use the dot product associated with Mahalanobis distance:

$$\mathrm{k}(\mathbf{y}_i, \mathbf{y}_j) = \mathbf{y}_j^\top \mathbf{L}_N^{-1} \mathbf{y}_i. \tag{29}$$

This normalization can be integrated in our projection step:

$$\mathrm{k}(\mathbf{y}_i, \mathbf{y}_j) = (\mathbf{L}_N^{-\frac{1}{2}} \mathbf{y}_j)^\top (\mathbf{L}_N^{-\frac{1}{2}} \mathbf{y}_i), \tag{30}$$

$$\mathbf{y}'_i = \mathbf{L}_N^{-\frac{1}{2}} \mathbf{P}_N^\top \mathbf{x}_i. \tag{31}$$

The compact signature has a size N, therefore $4 \times N$ bytes of storage space (in single precision) are used.

C. Binarized Compact VLAT

The storage size of signatures is a key point in the field of web scale similarity search. To produce ultra compact signatures, we propose to perform a binary quantization of compact VLAT signatures. We assume that signatures are sampled from a normal distribution which is consistent with the projections used in eq. (31). To maximize the retained information, we propose to set the threshold such that each class contains 50% of density. The binarized compact signature is then computed as:

$$\hat{\mathbf{y}}_i[j] = \begin{cases} 1 & \text{if } \mathbf{y}_i[j] \geq 0, \\ -1 & \text{otherwise.} \end{cases} \tag{32}$$

This quantization reduces the signatures size to $N/8$ bytes of storage space.

To sum up, our signature is computed in three steps: First we perform an optimization of the model with a PCA for each cluster of codebook. Secondly, we compute the VLAT signatures with preprocessed descriptors. Then, we compress the signatures by projection onto a subspace with a low rank approximation of some training Gram matrix. Finally, we reduce the storage size with a binary quantization of Compact VLAT signatures.



Fig. 1. Images from Holidays dataset.

IV. Experiments

In this section, we evaluate and compare our Compact VLAT signatures and our Binarized Compact VLAT signatures with the state-of-the-art. We use two evaluation datasets (INRIA Holidays and Oxford datasets) and three additional independent datasets to evaluate the performance of all methods:

INRIA Holidays dataset (Fig. 1) is a set of images drawn from personal holidays photos, created to test similarity search methods. It contains 1,491 images gathered in 500 subgroups, each of them being a distinct scene or object.

Oxford dataset is a set of images collected from Flickr by searching for particular Oxford landmarks. It contains 5,062 images gathered in 11 different landmarks, each represented by 5 possible queries.

Holidays Flickr1M dataset is a set of high quality pictures from Flickr. It contains 1 million images, commonly used as distractors for testing the Holidays dataset in large scale context.

Oxford Flickr100k dataset is a set of high quality pictures from Flickr. It contains 100,000 images, commonly used as distractors for testing the Oxford dataset in large scale context.

Holidays Flickr60K dataset is a set of high quality pictures from Flickr. It contains 60,000 images, commonly used as training set.

The three Holidays datasets are completely independent and include SIFT descriptors [17]. For the two Oxford datasets, we use a dense extraction of HOG descriptors.

For the INRIA Holidays dataset, we use the same evaluation setup as Jegou et al. [15] and for the Oxford dataset, we use the same evaluation setup as Philbin et al. [18].

For both, the accuracy of search is measured by the mean Average Precision (mAP).

To evaluate our methods at web scale, we merge a large images set (distactors set) with the standard evaluation dataset. For the INRIA Holidays dataset, we use the Flickr1M dataset as distactors set and for the Oxford datasets, we use the Flickr100k dataset as distactors set.

To study the influence of the parameters of our method, we use the INRIA Holidays dataset. For all experiments on INRIA Holidays, we compute a set of codebooks (32, 64, 128 visual words) with SIFT descriptors from the Flickr60K dataset. For each cluster of each codebook, we compute their mean and covariance matrix $(\boldsymbol{\mu}_c, \mathcal{T}_c)_c$ with SIFT descriptors of the Flickr60K dataset. We use these covariance matrices to compute the cluster-wise PCA. To compute the projectors of Compact VLAT signatures, we use a sample of 10k images extracted from Flickr60K dataset.

	32	64	128	256	9000	FULL
VLAT	-	-	-	-	-	64.0
PVLAT	-	-	-	-	-	**66.4**
CVLAT	46.3	49.6	53.3	54.9	**58.2**	-
CPVLAT	**47.1**	**51.9**	**53.9**	**55.6**	55.0	-
CVLAT-M	47.1	50.0	55.1	57.5	**70.0**	-
CPVLAT-M	**48.5**	**54.3**	**57.3**	**60.6**	**70.0**	-

TABLE I
Parameters study on Holidays dataset with $D = 64$ (mAP).

In this section we denote by D the number of clusters in the codebooks and by N the size of the signatures. We denote by "CVLAT" the Compact VLAT signatures, "CPVLAT" for the Compact VLAT signatures with Cluster-wise PCA and "-M" suffix denotes the use of the dot product associated with Mahalanobis distance.

A. Parameters study

In this section, we study the behavior of Compact VLAT signatures according to their parameters. All experiments are done with Holidays dataset, unless another setup is specified.

Table I shows the influence of the different stages of our method on the mAP. Rows are the differents configuration of our methods stages and columns represent the size N of the signature ("FULL" means uncompressed signature). We observe a gain of 2.4% between VLAT and PVLAT which highlights the improvements brought by the model optimization. Rows 3 and 4 show that the model optimization allows to retain more information at higher compression ratio (typically $N \leq 256$). We can see that using of the dot product associated with Mahalanobis distance greatly increases the performance with compressed signature. For $D = 64$ and $N = 256$, we divided by 2,000 the signature size for a loss of only 3.4% of mAP.

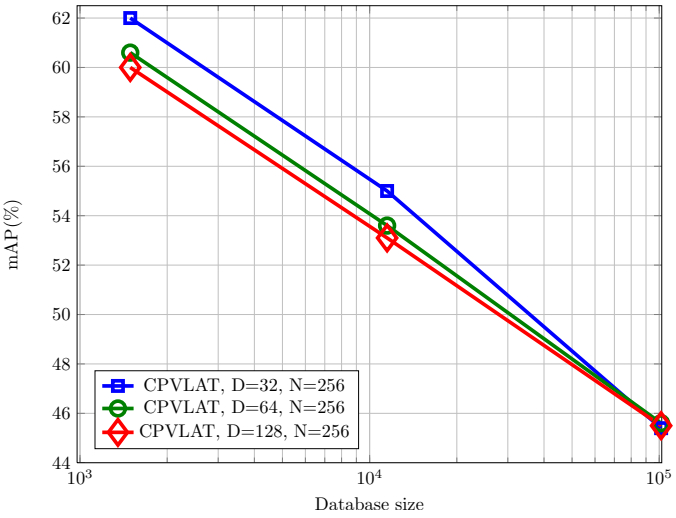


Fig. 2. Comparison of Compact VLAT signatures with cluster-wise PCA as a function of the database size and the D numbers of clusters in codebook.

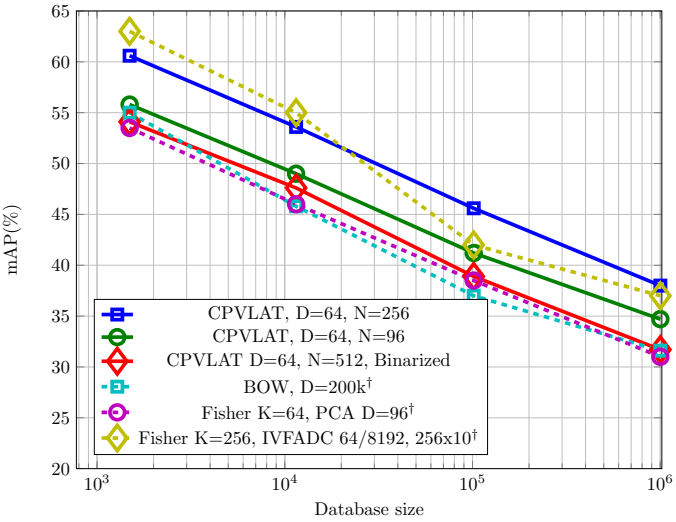


Fig. 3. Comparison of state-of-the-art signatures as a function of the database size († extract from [15]).

	CPVLAT, D=64					
N	16	32	64	128	256	512
Standard	22.1	33.5	38.9	42.7	45.6	47.6
Binarized	2.1	9.6	18.3	28.3	34.7	38.9

TABLE II
COMPARISON OF BINARIZED AND STANDARD COMPACT VLAT SIGNATURES WITH CLUSTER-WISE PCA ON 100K EXTENDED HOLIDAYS DATASET (MAP).

To study the influence of the number D of clusters on CPVLAT signature, we fixed the size to $N = 256$. Figure 2 shows the variation of the mAP according to the size of the database on Extended Holidays dataset. We show that for databases with fewer images, a small codebook gives better results. However, the results become similar when numbers of images in the database increases. This shows that a medium codebook ($D = 64$) leading to less computational time of projection gives sufficiently good results at larger scale.

To study the influence of binarization, we consider CPVLAT signature, and a codebook of 64 visual words. Table II shows the mAP (%) with the columns representing the size N of the signatures. We show that binarization reduces drastically the accuracy. However, since it leads to a strong compression of the storage size, a larger number of projectors can then be retained. Furthermore, we note that the loss of accuracy is lower for larger projections.

B. Comparison with the state-of-the-art

In this section, we compare our signatures with the results of [15] on the Extended Holidays dataset and with the results of [18] on the Oxford dataset.

For the Holidays dataset, we compute the CPVLAT signatures with a codebook of 64 visual words. CPVLAT signatures are computed with a subspace projection of size $N = 96$ and $N = 256$. We compute the Binarized CPVLAT signatures with a subspace projection of size $N = 512$. Results are shown in Figure 3.

Compare to BoW computed with a codebook of 200k visual words, CPVLAT signatures computed with a smaller codebook give better results. With CPVLAT signatures of size $N = 96$, we have a gain of $\sim 5\%$ of mAP, while our signatures are about 2,000 times smaller.

Compared to the Fisher signature computed with a codebook of size 64 (different from our codebook) and keeping the first 96 dimensions with PCA, we obtain better results with same size of codebook and signature. We also have similar results with Binarized CPVLAT signatures. However, our storage size is much smaller with 64 bytes compared to 384 bytes for the Fisher signature with PCA.

Compared to the Fisher Vectors signature indexed by IVFADC with a codebook of size 256, we obtain lower results on small size databases. However, this signature is more sensitive to the increased number of images. For more than 10k images, we have better results with a smaller codebook.

To test the universality of our method, we use default parameters on Oxford datasets. We use Oxford images as training set for all parameters. We compute the VLAD, VLAT, and CPVLAT signature with a dense extraction of HOG descriptors. We use the same codebook of 64 visual words for all signatures. For compressed VLAD signatures, we use the same protocol as in [18]. Results are shown in Table III.

We can see that using dense extraction of HOG descriptors increases the performance of VLAD signature of 6%. The compression of VLAD@HOG signature has about the same loss that the compression of VLAD in [18]. We observe that the VLAT signature has much better performance than the VLAD signature. With this setup, we observe that our method has much better performances at large scale for the same size (around 20% mAP improvement).

C. Scalability

In this section, we study the influence of the storage size of our signatures. We compute the CPVLAT signatures with

	Oxford	Oxford + 100k
Fisher [18]	31.7	-
VLAD [18]	30.4	-
Fisher-PCA (N=128) [18]	24.3	-
VLAD-PCA (N=128) [18]	25.7	-
VLAD@HOG	36.6	-
VLAT@HOG	50.7	-
PVLAT@HOG	54.2	-
VLAD@HOG-PCA (N=128)	32.7	25.6
CPVLAT@HOG (N=128)	**54.3**	**46.6**

TABLE III
PERFORMENCE OF THE ROW DESCRIPTORS AS WELL AS DESCRIPTORS COMPRESSED ON OXFORD DATASET AND OXFORD DATASET + 100K DISTRACTORS (MAP).

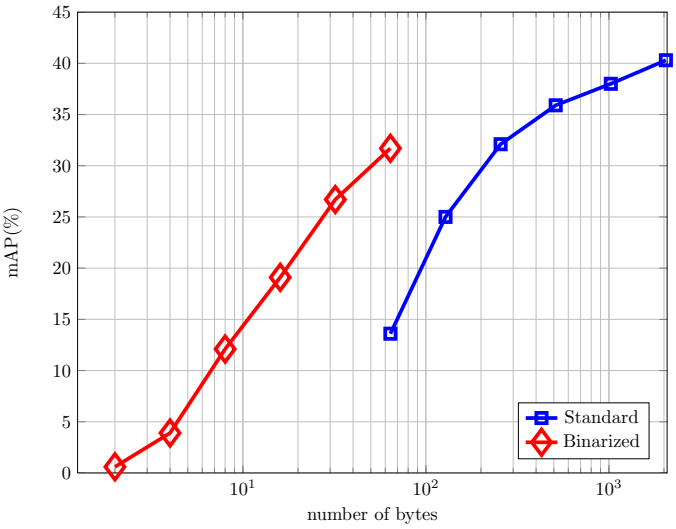


Fig. 4. Comparison of Binarized and Standard Compact VLAT signatures with cluster-wise PCA on 1M Extended Holidays dataset ($D = 64$).

varying number of selected projectors and the same binarized versions. We set the size D of the codebook to 64. In Figure 4, we plot the standard CPVLAT signature and Binarized CPVLAT signature against the storage size for 1 million images. We observe that binarized version of the signature leads to much better results at similar storage size. For a storage size of 64 bytes, we obtain a mAP of 14.6% with a standard CPVLAT signature and a mAP of 31.6% with a Binarized CPVLAT signature (gain of 17% of mAP). With the Binarized CPVLAT signature of dimension $N = 512$, all signatures of the Extended Holidays dataset are stored in only 61 MB of memory.

V. CONCLUSION

In this paper, we proposed a new compact signature for similarity search in web scale databases called Compact VLAT. Our method belongs to the model deviation approaches. First, we preprocess descriptors using PCA in each cluster to ensure good properties for the compression step. We use an aggregation of tensors of preprocessed descriptors. Then we compress the signatures using projections onto a subspace analog to kernel-PCA. We carried out similarity search experiments on the Extended INRIA Holidays dataset (1M images) and Oxford dataset (100k images). We presented the impact of the signatures size on its performance. We compared our results with popular methods, and showed the competitiveness of our approach for large scale datasets.

Future works include the following issues: First, combining VLAT and VLAD signatures before performing the projection step; Secondly, using a soft assignment of descriptors inspired by coding techniques; Third, using a non-binary quantization for the extra compression step. Finally, we want to stress that the next challenge to be addressed in web scale image retrieval will be the loss of performances occurring when the number of distractors increases.

REFERENCES

[1] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 2, no. 60, pp. 91–110, 2004.
[2] A. Gionis, P. Indyk, and R. Motwani, "Similarity search in high dimensions via hashing," in *Proceedings of the 25th International Conference on Very Large Data Bases*, ser. VLDB '99. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1999, pp. 518–529.
[3] H. Lejsek, B. T. Jónsson, and L. Amsaleg, "Nv-tree: nearest neighbors at the billion scale," in *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, ser. ICMR '11. New York, NY, USA: ACM, 2011, pp. 54:1–54:8.
[4] H. Jegou, M. Douze, and C. Schmid, "Improving bag-of-features for large scale image search," *International Journal of Computer Vision*, vol. 87, no. 3, pp. 316–336, 2010.
[5] J. Shawe-Taylor and N. Cristianini, *Kernel methods for Pattern Analysis*. Cambridge University Press, ISBN 0-521-81397-2, 2004.
[6] S. Lyu, "Mercer kernels for object recognition with local features," in *IEEE International Conference on Computer Vision and Pattern Recognition*, San Diego, CA, 2005, pp. 223–229.
[7] P.-H. Gosselin, M. Cord, and S. Philipp-Foliguet, "Kernel on bags for multi-object database retrieval," in *ACM International Conference on Image and Video Retrieval*, Amsterdam, The Netherlands, July 2007, pp. 226–231.
[8] F. Precioso, M. Cord, D. Gorisse, and N. Thome, "Efficient bag-of-feature kernel representation for image similarity search," in *International Conference on Image Processing*, 2011, pp. 109–112.
[9] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *IEEE International Conference on Computer Vision*, vol. 2, 2003, pp. 1470–1477.
[10] S. Avila, N. Thome, M. Cord, E. Valle, and A. De A. Araújo, "BOSSA: extended BoW formalism for image classification," in *International Conference on Image Processing*, Brussels, Belgique, Sep. 2011, cAPES, CNPq, FAPESP (Brazil), COFECUB (France).
[11] J. C. van Gemert, J.-M. Geusebroek, C. J. Veenman, and A. W. M. Smeulders, "Kernel codebooks for scene categorization," in *ECCV 2008, PART III. LNCS*. Springer, 2008, pp. 696–709.
[12] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *in IEEE Conference on Computer Vision and Pattern Recognition(CVPR*, 2009.
[13] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *IEEE International Conference on Computer Vision and Pattern Recognition*, 2010, pp. 3360–3367.
[14] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *European Conference on Computer Vision*, 2010, pp. 143–156.
[15] 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, qUAERO.
[16] D. Picard and P.-H. Gosselin, "Improving image similarity with vectors of locally aggregated tensors," in *International Conference on Image Processing*, Brussels, Belgique, September 2011.

[17] H. Jegou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *European Conference on Computer Vision*. Springer, 2008, pp. 304–317.
[18] H. Jégou, F. Perronnin, M. Douze, C. Schmid *et al.*, "Aggregating local image descriptors into compact codes," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, no. 9, pp. 1704–1716, 2012.

VI. Biography



Romain Negrel is currently a second year PhD student in multimedia retrieval at the ETIS Lab at the University of Cergy-Pontoise (France), working under the joint supervision of Philippe Henri Gosselin and David Picard. His thesis title is "Représentations Optimales pour la Recherche dans les Bases d'Images Patrimoniales."



David Picard received the M.Sc. in Electrical Engineering in 2005 and the Ph.D. in image and signal processing in 2008. He joined the ETIS laboratory at the ENSEA (France) in 2010 as an associate professor within the MIDI team. His research interests include computer vision and machine learning for visual information retrieval, with focus on kernel methods for multimedia indexing.



Philippe-Henri Gosselin received the PhD degree in image and signal processing in 2005, and joined the MIDI Team in the ETIS Lab as an assistant professor in 2007. His research focuses on machine learning for online multimedia retrieval. This includes studies on kernel functions on histograms, bags and graphs of features, but also weakly supervised semantic learning methods.