Consistent Visual Words Mining with Adaptive Sampling

Pierre Letessier, Olivier Buisson, Alexis Joly

To cite this version:

Pierre Letessier, Olivier Buisson, Alexis Joly. Consistent Visual Words Mining with Adaptive Sampling. ICMR'11 - International Conference on Multimedia Retrieval, Apr 2011, Trento, Italy. pp.0-0, 10.1145/1991996.1992045 . hal-00642202

HAL Id: hal-00642202

https://hal.inria.fr/hal-00642202

Submitted on 17 Nov 2011

Consistent Visual Words Mining with Adaptive Sampling

Pierre Letessier
INA and INRIA Rocquencourt
94366 Bry-sur-Marne, France
pierre.letessier@ina.fr

Olivier Buisson
INA
94366 Bry-sur-Marne, France
obuisson@ina.fr

Alexis Joly
INRIA Rocquencourt
78153 Le Chesnay, France
alexis.joly@inria.fr

ABSTRACT

State-of-the-art large-scale object retrieval systems usually combine efficient Bag-of-Words indexing models with a spatial verification re-ranking stage to improve query performance. In this paper we propose to directly discover spatially verified visual words as a batch process. Contrary to previous related methods based on feature sets hashing or clustering, we suggest not trading recall for efficiency by sticking on an accurate two-stage matching strategy. The problem then rather becomes a sampling issue: how to effectively and efficiently select relevant query regions while minimizing the number of tentative probes? We therefore introduce an adaptive weighted sampling scheme, starting with some prior distribution and iteratively converging to unvisited regions. Interestingly, the proposed paradigm is generalizable to any input prior distribution, including specific visual concept detectors or efficient hashing-based methods. We show in the experiments that the proposed method allows to discover highly interpretable visual words while providing excellent recall and image representativity.

1. INTRODUCTION

State-of-the-art object retrieval systems have demonstrated impressive performances in very large image datasets. These methods, based on fine local descriptions and efficient matching techniques, can detect accurately very small rigid objects with unambiguous semantic such as logos, buildings, manufactured objects, posters, etc. Mining such small objects in large collection is a challenging task gaining more and more interest. Applying naively usual local queries methods might indeed be a tricky task. Constructing a full local matching graph with these methods would indeed require to probe all candidate query regions around each local feature leading to an intractable algorithm complexity.
To avoid querying all possible regions of interest while keeping a good coverage of the contents, we propose in this paper a *weighted* and *adaptive* sampling strategy aiming to select the most relevant query regions. *Sampling* is indeed a simple yet efficient statistical paradigm allowing to yield some knowledge about a population without surveying it entirely. Adaptive weighted sampling is a more advanced paradigm allowing to iteratively update the sampling distribution according to the results obtained during previous iterations. This allows our mining method to progressively focus on unvisited image regions and consequently reduce the number of required probes for achieving a good completeness of the description. The resulting set of discovered objects can therefore be used as a new type of visual word vocabulary allowing to describe images with very compact global descriptions. We show in the experiments that better retrieval performances than those obtained with usual visual words might be achieved with extremely smaller vocabulary size (e.g. 3,000 visual words for the OxfordBuildings dataset). We therefore refer to the visual words generated by our method as *Consistent Visual Words* (CVW). A CVW is a set of image patches. These patches are described by local features sets. A CVW models a small rigid object, or a piece of a bigger object. A CVW is defined by the geometric consistency between the feature points of the considered patches.

Interestingly, the initial distribution used by our adaptive sampling method can be initialized with any prior knowledge, typically visual saliency measures or visual concepts detectors. The produced visual vocabularies might therefore be adapted to specific targeted objects or concepts.
Besides Bag-of-Words description concerns, our method can simply be used for discovering instances of very small rigid objects in large datasets. We show as well in this context that our adaptive sampling matching method allows to reach high recall while being very flexible in terms of prior knowledge about the targeted objects.

2. RELATED WORKS

State-of-the-art object discovery and mining methods [14, 19] can be summarized by two main successive steps: *matching graph* construction, and analysis of this graph. The nodes of a *matching graph* typically represent images whereas edges correspond to common matching regions between the images. The first step to generate the matching graph is usually based on large scale object retrieval methods [14, 5, 3]. The objectives addressed by the graph analysis methods are various and include: construction of the object models, objects linking or summarization. To discover or extract such information, most of these methods are based on Latent Topic Models such as probabilistic Latent Semantic Analysis (pLSA) [8] and Latent Dirichlet Allocation (LDA)

ICMR '11, April 17-20, Trento, Italy

[1, 14]. Most matching graph methods are based on Bags-of-visual-Words models (BoW), describing images from a set of quantized local features lying in a so called visual vocabulary. The visual vocabulary is usually generated by a K-means clustering algorithm [15] applied on the local features set of the considered corpus. Such representations are equivalent to standard vector-space models in text information retrieval allowing to efficiently measure similarity between items with classical operators (dot product, histogram intersection, etc.). It is important to notice that such similarities only consider global statistics of the image even if the sparsity of the vectors allows to somehow embed some local properties. This type of global strategies is therefore not adapted to discover very small objects representing small subparts of images. In [16, 4, 14], for example, the objects having a size lower than 25% of the image area are not considered. So that usual BoW methods are mostly interesting for their excellent efficiency/effectiveness tradeoff not for their effectiveness in retrieving or mining objects.

When dealing with small objects, the most adapted methods to generate a matching graph are the one based on probing local query regions. This second type of strategies [15, 9, 11] independently consider each local feature of an image. They have been shown to be effective in retrieving small objects, such as trademark logos [11]. To guarantee this quality level of precise objects retrieval, these methods stick on an accurate local description of the images. For a 10K image corpus, more than tens of millions local features are independently indexed. In these methods, the number of candidate local regions to be searched might thus be huge. Even with the most efficient indexing structures that drastically reduce the computational costs, the overall complexity remains too expensive to generate a full matching graph.

Instead of searching every candidate query regions, Chum et al. [5, 3] propose to automatically select Regions Of Interest (ROI) with a very low computational cost. Their method combines BoW models with a Min-Hash hashing scheme [2] and can be considered as a trade-off between global and local strategies. Min-Hash is an algorithm commonly used in text retrieval for finding near-duplicates [2] and works by approximating the intersection between two sets of words. Applied on visual words, it allows to discover efficiently very discriminant candidate visual sketches that are likely to be parts of more reliable objects. But the recall of this method for small objects is far from sufficient as pointed out in further works of the authors [3]. The first Min-Hash/BoW method [5] was therefore only used to detect Image Near Duplicates.
To reduce this drawback, [3] proposed a new Min-Hash based strategy called Geometric Min-Hash. In that method, the first Min-Hash value of the sketches is still generated from the whole image but the second and following hash values are randomly sampled in the spatial neighborhood of the first selected visual word. This version is able to discover more relevant local sketches and is therefore a very efficient way to discover candidate query regions that are likely to contain object instances. But the first global hashing step still makes it not robust to strong occlusions and the performances might therefore degrade in highly cluttered contexts. We also point out that this method might be used as an effective prior knowledge in our adaptive sampling matching method. To the best of our knowledge, our method is the first matching graph construction method directly based on probing local queries.

3. PROPOSED METHOD

Our proposed mining method is an iterative process composed of three main stages processed at each iteration: *Adaptive Sampling* of a query image region, *Search* of the selected local query region and *Decision* of whether this query region might be considered as a consistent visual word in the final output vocabulary. The full algorithm repeats these 3 steps T times until a fixed number of visual words has been found. More formally, let Ω be an input dataset of N images $\mathbf{I}_i$, $i \in 1, ..., N$. Each image $\mathbf{I}_i$ is represented by a set of N_i local visual features $F_{i,j}$ (typically SIFT [12]) localized by their position $\mathbf{P}_{i,j}$. $N^F = \sum_{i=0}^{N-1} N_i$ is the total number of features $F_{i,j}$. Each local feature $F_{i,j}$ is associated with a fixed candidate query region $R_{i,j}$ defined as the bounding box centered around $\mathbf{P}_{i,j}$, with height $H_{i,j}$ and width $W_{i,j}$. In this paper, $H_{i,j}$ and $W_{i,j}$ are set up as fixed ratio of image width and height according to:

$$H_{i,j} = \sqrt{\gamma} * H_i$$

$$W_{i,j} = \sqrt{\gamma} * W_i$$

where γ is a parameter of the method corresponding to the percentage of the image area covered by a candidate query region (H_i and W_i are respectively the height and width of image $\mathbf{I}_i$). For example, a γ equals to 0.10 means that the query width will be equal to one third of the image width. This parameter is designed to adapt the query size and shape to those of the image. Now, the following three steps are processed at the t-th iteration :

1. *Adaptive Sampling:* This step proposes a candidate query region R_q^t, centered around a randomly selected feature F_q^t. The feature F_q^t is selected according to a probability mass function $p^t(i,j)$ over the set of all $F_{i,j}$. Such random sample can be generated easily from any distribution p^t by using an *inverse transformation method* [6] (also called *Smirnov* method). Such method consists in transforming a uniform random number by integrating the probability mass function up to an area greater than its value. The algorithm computing p^t according to the results of the previous steps is detailed in section 3.1.

2. *Local Region Search*: The candidate query region R_q^t (centered around F_q^t) is processed by a three-step matching procedure described in section 3.2. It returns a set of geometrically verified matching regions in the dataset. We refer to any of these matched regions as R_m^t, $m \in 1, ..., M_t$.

3. *Decision*: Matching scores are then normalized and thresholded according to the procedure described in section 3.3. If the final results set contains more than two images, it means that we have found a recurrent object in the database. This threshold can be adapted, depending on desired minimal frequency of retrieved objects. Then, the tentative query region R_q^t and the matching images R_m^t are kept to form a consistent visual word.

Finally, after T tentative probes, the algorithm outputs a vocabulary V of $|V| \leqslant T$ consistent visual words v_t. Each visual word is associated with an image region R_q^t and represented by the set of local features belonging to R_q^t.

3.1 Adaptive Sampling

To avoid querying all possible regions of interest while keeping a good coverage of the contents, we propose a *weighted and adaptive* sampling strategy aiming at selecting the most relevant query regions. *Sampling* is a statistical paradigm concerned with the selection of a subset of individual observations within a population of objects intended to yield some knowledge about the population without surveying it entirely. If all items have the same probability to be selected, the problem is known as *uniform random sampling*. In *weighted* sampling methods [13], the items might be weighted individually and the probability of each item to be selected is determined by its relative weight. In conventional sampling designs, either *uniform* or *weighted*, the selection for a sampling unit does not depend on the observations made during previous surveys. On the other side, *adaptive sampling* [18] is an alternative strategy aiming at selecting more relevant sampling regions based on the results observed during the previous surveys.

Our method starts with an initial probability mass function $p^0(i,j)$ over the whole set of candidate query regions $R_{i,j}$. This initial distribution might be either uniform or determined by some prior knowledge as discussed in Section 3.1.1. Steps 1, 2 and 3 are then computed (see above) providing a selected query region R_q^0 and a set of matching regions R_m^0, $m \in 1, ..., M_0$.

Further probability mass functions $p^t(i,j)$ are then updated in a recursive manner:

$$p^t = f\left(p^{t-1}, R_q^{t-1}, \{R_m^{t-1}\}\right)$$

As done in conventional weighted random sampling methods [13], the probability mass functions p^t are in practice computed by normalizing a weighting function w^t such as:

$$p^t(i,j) = \frac{w^t(i,j)}{\sum_{i,j} w^t(i,j)} \tag{1}$$

The recursive updates are thus rather computed on the weights:

$$w^t = g\left(w^{t-1}, R_q^{t-1}, \{R_m^{t-1}\}\right)$$

Our proposed updating function g is defined as follows:

$$w^t(i,j) = \begin{cases} 0 & \text{if } F_{i,j} = F_q^{t-1} \\ \alpha_1 \; w^{t-1}(i,j) & \text{if } F_{i,j} \in R_q^{t-1} \\ \alpha_2 \; w^{t-1}(i,j) & \text{if } F_{i,j} \in \left\{R_m^{t-1}\right\}_m \\ w^{t-1}(i,j) & \text{otherwise} \end{cases} \tag{2}$$

The **first condition** means that the weights of already visited query region centers are set up to zero so that the probability to re-issue them as a new query is null (i.e. to guaranty a sampling without replacement).

The **second condition** aims at decreasing the weights of the features belonging to the previous query region R_q^{t-1}, so that their probability to be re-issued as new query region centers is decreased. This avoids selecting new query regions that have too much overlap with previous query regions.

The **third condition** aims at decreasing the weights of the features belonging to already matched regions, so that their probability to be re-issuing as new query region centers is also decreased.

On the other side, the **fourth condition** keeps the weights of unmatched features unchanged. These three first conditions allow to iteratively focus the selected query regions on objects that were never found in previous steps.

In practice, α_2 is chosen to be greater than α_1. Decreasing too much the weights of matched regions might indeed degrade the overall recall. This comment can be related to the success of query expansion methods [11] aiming at boosting object retrieval recall by re-issuing different instances of the same object as new queries.

In our experiments we used:

$$\alpha_1 = 0.1 \qquad \alpha_2 = 0.5$$

3.1.1 Prior saliency measures

As discussed above, the initial probability mass function p^0 can be either uniform or determined by some prior knowledge. In this section we propose and discuss several prior distributions. A first group consists of generalist visual saliency measures that might be used in any case. The second group consists of specific priors aiming to focus on the discovered visual words on specific visual concepts. The core idea is that the visual vocabulary produced by our adaptive sampling method might be adapted to the user's query. We notice that the output score of any visual concept detector could be used as prior knowledge making our method widely generic (e.g. animals, plants, indoor/outdoor, etc.).

Generalist Saliency Measures The purpose of this group of priors is to guide the sampling process on informative regions according to some visual saliency measures. We describe here two saliency measures but any other of the literature might be used as well.

Spatial Density of Local Features: The principle of this saliency measure is to emphasize centers of spatially dense regions since they are more likely to be the centers of interesting objects. Concretely, we measure for each local feature $F_{i,j}$ the spatial density of its neighboring features with a Parzen window:

$$\pi^D(i,j) = \sum_{c=0}^{N_i - 1} \mathcal{K}_\sigma(\mathbf{P}_{i,j}, \mathbf{P}_{i,c})$$

where $\mathcal{K}_\sigma$ is typically an RBF kernel. In our experiments, we chose two distinct values of σ for the vertical and horizontal coordinates ($\sigma_x = \frac{W_i}{3}$ and $\sigma_y = \frac{H_i}{3}$).

Feature Space Dispersion: One drawback of matching techniques is due to the degradation of performances on repeated structures (such as textured regions) because of an increasing probability of multiple matches [17]. To avoid such regions, we propose to measure the feature-space dispersion of the features belonging to a candidate region $R_{i,j}$. To estimate the dispersion, we simply average the variance along each dimension of the feature space:

$$\pi^\delta(i,j) = \frac{1}{d} \sum_{l=0}^{d-1} \mathrm{Var}_{R_{i,j}}(F\,[l])$$

where d is the dimension of the feature vector.

A result of this saliency measure is illustrated in Figure 1.

Specific Saliency Measures

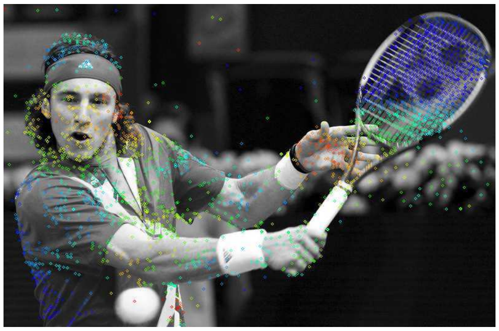

Figure 1: Feature space dispersion saliency measure - A hot color means that the center feature is different from its neighbors, a cold one means that it is likely to be in a texture region

Face saliency: In order to build face specific vocabularies of visual words, we implemented a prior distribution based on the OpenCV face detector [1]:

$$\pi^F(i,j) = \sum_{l=0}^{N_i^f-1} \mathcal{K}_{\sigma_{i,l}^f}(\mathbf{P}_{i,j}, \mathbf{P}_{i,l}^f)$$

where $\mathcal{K}$ is the RBF kernel, N_i^f is the number of faces detected in image $\mathbf{I_i}$, $\mathbf{P}_{i,l}^f$ and $\sigma_{i,l}^f$ are respectively the center and the scale of the l-th face in the current image i.

Image center saliency: In many applications, the assumption that the image center contains more objects of interest might be relevant. We therefore implemented the following saliency to boost the features closer to the image center:

$$\pi^P(i,j) = \mathcal{K}_\sigma(\mathbf{C}_i, \mathbf{P}_{i,j})$$

where $\mathbf{C}_i$ is the center of image I_i and $\mathcal{K}$ the RBF kernel parameterized with $\sigma_i^x = \frac{W_i}{8}$, $\sigma_i^y = \frac{H_i}{8}$.

3.1.2 Transforming prior saliency measures into probability mass functions

We introduce here two generic schemes to transform any saliency measure into a probability mass function p^0.

Linear function: For a given saliency measure π, we define the initial weighting function w^0 as

$$w^0(i,j) = \frac{\pi(i,j) - min_\pi}{max_\pi - min_\pi}$$

where min_π and max_π are respectively the minimum and maximum values reached by the saliency measure on the whole dataset. p^0 is then obtained from w^0 with Equation 1.

Rank-based function: For a given saliency measure π, we rank every value $\pi(i,j)$ and use the resulting ranks as initial weights:

$$w^0(i,j) = \frac{N^F - \rho(i,j)}{N^F}$$

where N^F is the total number of features and $\rho(i,j)$ the rank of the feature $F_{i,j}$. p^0 is then obtained from w^0 with Equation 1.

Our different experiments demonstrate that Linear function is adapted for $\pi^D(i,j)$, $\pi^F(i,j)$, $\pi^P(i,j)$ and Rank-based function is better for $\pi^\delta(i,j)$.

3.2 Local Region Search

For local region search, we used the following retrieval framework proposed in [11]. In this step, a region R_q^t is a local query Q represented by a set of n_q features $\mathbf{F}_{qi}$, with corresponding positions $\mathbf{P}_{qi}$ for $i \in 1, ..., n_q$. The Local Region Search method works as follows:

STEP 1 - SIFT's matching Each query feature $\mathbf{F}_{qi}$ is matched to the dataset thanks to an efficient approximate similarity search technique. We used the recent A Posteriori Multi-Probe Locality Sensitive Hashing (APMP-LSH) method proposed by Joly et al. [10] that allows sublinear search time with reduced memory space costs. This Approximate Nearest Neighbor search method has a parameter α to control the search quality. This parameter allows to control the trade-off between search quality and efficiency. This control is very important to limit the computational costs in function of the desired performances. We study the effects of α in Section 4.3. To drastically improve the performances, we combine the APMP-LSH with Hamming Embedding [7, 9]. This Embedding reduces the memory costs of the local descriptors and replaces the $L2$ distance with Hamming Distance.

STEP 2 - Filtering of potential matching images Then, we keep only the retrieved images which have more than a given number of matching features for the next step.

STEP 3 - Geometric consistency For each remaining image result, we compute a geometric consistency score by estimating an affine transformation model between the query and the retrieved images. An affine transformation model ($\mathbf{A}_j$,$\mathbf{B}_j$) with 5 degrees of freedom is first estimated for each remaining image by a RANSAC algorithm. Finally, the similarity score of an image $\mathbf{I}_j$ for query Q is given by the number of inliers according to the affine transformation model:

$$S_Q(\mathbf{I}_j) = \sum_{m=1}^{M_j} \delta(\|\mathbf{P}_{qm} - \mathbf{A}_j\mathbf{P}_{jm} + \mathbf{B}_j\| \geq t) \qquad (3)$$

where $\delta(d \geq t)$ equals 1 if $d \geq t$ and 0 otherwise. t is a fixed threshold setting the position error tolerance ($t = 8$). M_j is the number of matches kept in image $\mathbf{I}_j$ after step 2, $\mathbf{P}_{qm}$ and $\mathbf{P}_{jm}$ are the query and matched spatial coordinates of the p-th match in image $\mathbf{I}_j$. The output is a list of images ranked in the decreasing order of the number of inliers $S_Q(I)$.

3.3 Decision

To decide if the tentative query region R_q^t is kept as a part of a consistent visual word, we filter the previous results by normalized scores which are obtained by an *a contrario* normalization technique [11]. The *a contrario* normalization technique allows to accurately control the percentage of false alarms. This method involves a consistent visual word selection with a low level of false alarms. This technique is based on an estimation of the false alarms distribution $\hat{N}_{fa}(S)$ with respect to the discrete random variable $S = S_Q(I), I \in \Omega$. According to Equation 3, S_Q depends only on the set of M_j spatial coordinates pairs corresponding to the n_p matches

[1] http://sourceforge.net/projects/opencvlibrary

found in Step 1. High S_Q scores are thus directly related to the statistical dependence between the spatial positions of the query and the matched features. We thus define our *a contrario* background model by the probability mass function $\hat{p}_{fa}(S)$ of the variable S under the hypothesis $\mathcal{H}_0^Q$ that $\mathbf{P}_{qm}$ and $\mathbf{P}_{jm}$ are mutually independent random variables for all j:

$$\hat{p}_{fa}(S) = \Pr[S_Q(I) = S \mid \mathcal{H}_0^Q]$$

The cumulative distribution function $\hat{N}_{fa}(S)$ can be obtained by:

$$\hat{N}_{fa}(S) = \sum_{s=0}^{S} \hat{p}_{fa}(s)$$

We finally keep as the normalized score $\hat{S}_Q(I)$ an estimation of the results precision according to $\hat{N}_{fa}(S)$:

$$\hat{S}_Q(I) = \frac{\#\{\mathbf{I}_j \in \Omega, S_Q(\mathbf{I}_j) > S_Q(I)\} - N.\hat{N}_{fa}(S_Q(I))}{\#\{\mathbf{I}_j \in \Omega, S_Q(\mathbf{I}_j) > S_Q(I)\}}$$

In practice, we estimate the probability mass function $\hat{p}_{fa}(S)$ for each query Q by a Monte Carlo simulation. We generate independent spatial positions of the query features $\mathbf{P}_{qm}$ and we keep the matched positions $\mathbf{P}_{jm}$ unchanged. More precisely, we affect to a given query feature $\mathbf{F}_{qi}$ a new spatial position $\mathbf{P}_{qj}$ randomly selected among the other points positions of the query. Compared to a purely uniform random generation of point positions, this method has the advantage to preserve some prior knowledge about the points distribution, such as bounds and principal orientations. We then simply recompute Step 3 and we estimate $\hat{p}_{fa}(S)$ by counting the number of results having a score S_Q equal to S. To limit the estimation bias due to the presence of correct images in the random results list, we only keep in the count the results having a score higher than the one obtained with the normal query.

4. EXPERIMENTS

4.1 Software Implementation

To reduce the main computational costs which is the Approximate Nearest Neighbor search of each query local feature, we developped a Thread Safe index structure based on A Posteriori Multi-Probe Locality Sensitive Hashing (APMP-LSH) [10] and Hamming Embedding [7, 9]. The Thread Safe version of APMP-LSH provides a shared index structure between different threads, that avoids the duplication of data. Our tests of this version show that on two 4-cores processors (with HyperThreading), the optimum computational costs are obtained with 24 threads. These 8 cores and 24 threads provide a gain factor of 6 compared to 1 core and 1 thread. To drastically improve the performances, we combine the Thread Safe version of APMP-LSH with Hamming Embedding. This embedding reduces the memory costs of the local descriptors and replaces the $L2$ distance with Hamming Distance. Our different experiments in terms of object retrieval performances (mean Average Precision) has shown that the SIFT descriptors represented with 128 bits in the Hamming space provides equivalent performances than the 32 bits float precision version in the Euclidean space. In this case, the compression factor is 32. Moreover, the use of Hamming distance allows to reduce the computational costs compared to $L2$ distance. The fast Hamming distance implementation is composed of two steps: XOR operation between two descriptors and counting of number of bits set to 1 of the previous result. The standard version fast Hamming distance is based on a lookup table to count number of bits set to 1. Instead of using this version, we count number of bits set to 1 with a SSE4.2 instruction [2]: popcnt. This instruction allows to divide by 4 the computational costs compared to the lookup table version.

4.2 Datasets

During our evaluations, we use the following image datasets:

- BelgaLogos (**BEL**): The BelgaLogos dataset [11] [3] is composed of 10,000 news images. A manually annotated ground-truth for 26 logos and 55 queries where each query is a logo is provided with this dataset. This dataset is adapted to evaluate small objet retrieval performances. For the current evaluations, we describe this corpus with 38 millions of SIFT [12].

- OxfordBuildings (**OXF**): The 5,000 images of Oxford Buildings dataset [4] are composed of buildings from Oxford and miscellaneous images. A ground-truth is provided for 55 queries where each query is a piece of a building. For the current evaluations, we describe this corpus with 30 millions of SIFT.

To evaluate the objet retrieval performances, we mainly use the usual criterion: mAP (mean Average Precision) computed between our retrieval results and the provided ground-truth.

4.3 Parameters Tuning for Local Region Search

To evaluate the involvements of our Adaptive Sampling method, we need to choose the parameters of Local Region Search. Our different experiments of APMP-LSH coupled with Hamming Embedding allows us to use the following parameters: 128 bits for the Hamming Embedding representation of SIFT, 300 for the KNN size. We evaluated these parameters with the two considered datasets (with $\alpha = 90\%$) and we obtain: $mAP = 0.30$ for **BEL** and $mAP = 0.72$ for **OXF**. In [11], the system without Query Expansion has a $mAP = 0.20$ with 11 millions of SIFT for **BEL** and a $mAP = 0.608$ with 12 millions of SIFT for **OXF**. The main difference of performances is due to the use of more SIFT descriptors than in the experiments of [11].

The computational costs of each Tentative Probe mainly depend of the Local Region Search complexity. This complexity can be changed by the parameter α of the APMP-LSH which allows to control the trade-off between the KNN retrieval quality and the computational cost. Specifically, the value of α is the rate of KNN that we are statistically ensured to retrieve with the approximative search of APMP-LSH. To choose an adapted value for α, we tested a serie of α values for the Local Region Search applied on **OXF** dataset. We then observed the variation of the mAP depending of α (and by consequence the execution time) in Figure 2. This figure perfectly shows that it is not profitable in terms of mAP and computational cost to use an α value greater than

[2] http://en.wikipedia.org/wiki/SSE4

[3] http://www-rocq.inria.fr/imedia/belga-logo.html

[4] http://www.robots.ox.ac.uk/vgg/data/oxbuildings/

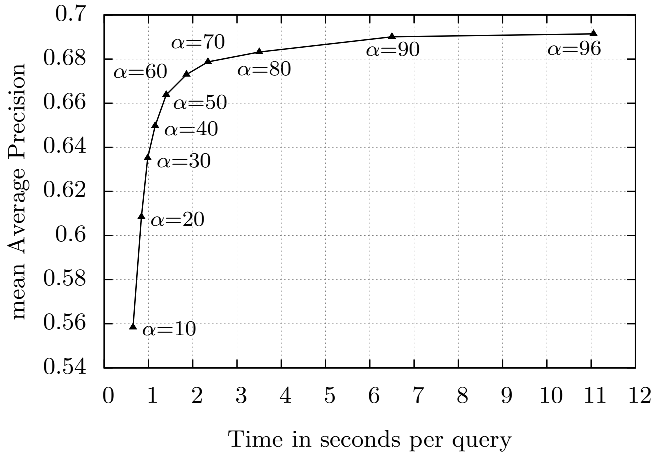

Figure 2: Trade-off between precision and time (features retrieval quality)

80%. Therefore we will use this value for the following experiments.

4.4 Adaptive Sampling and Priors Evaluation

The curves on the Figure 3 have been created with the BelgaLogos dataset, $\alpha = 80\%$, and un-weighting factors $\alpha_1 = 0.1$ and $\alpha_2 = 0.5$.

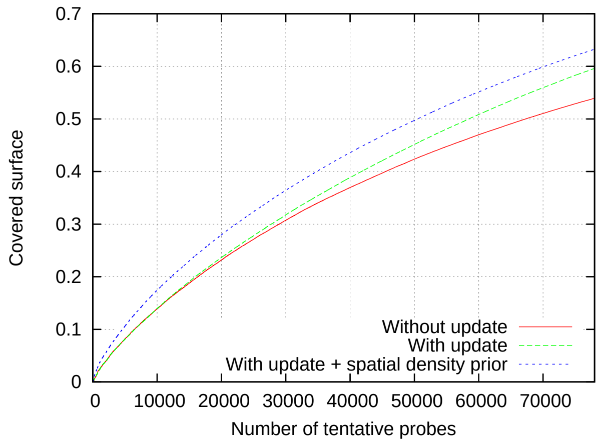

Figure 3: Comparison of the covered surface versus the number of tentative probes with or without using the update of the sampling distribution, and with the prior density

They show that after 10,000 tentative probes, updating the sampling distribution allows to cover a greater surface. We can also see that using the prior spatial density speeds-up the discovering process very early. For 70,000 tentative probes with the prior spatial density, we cover 10% more than without update.

Figure 4 illustrates the effect of updating the weighted distribution. We stopped our two tests (with and without update) when the same number of words has been created. The left image shows a lot of overlapping words, while the right image presents almost the same covered area, with fewer words.

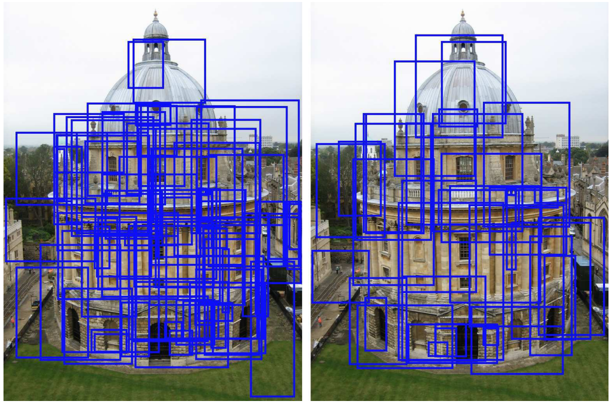

Figure 4: Bounding boxes of the consistent visual words generated on the same image without update (left image) and with update (right image)

Figure 5 shows that priors significantly improve the mAP. We demonstrate that adding *a priori* information based on our knowledge of the database ("objects of interest are often centered in the photography", or "the best described areas are more discriminant") can speed up the object discovery. We also see that focusing on non-texture areas gives better results.

The tests were made on **OXF** dataset with a low quality of feature retrieval ($\alpha = 40\%$), in order to gain time and to allow us to make 10 runs for each prior. We then computed the median curves.

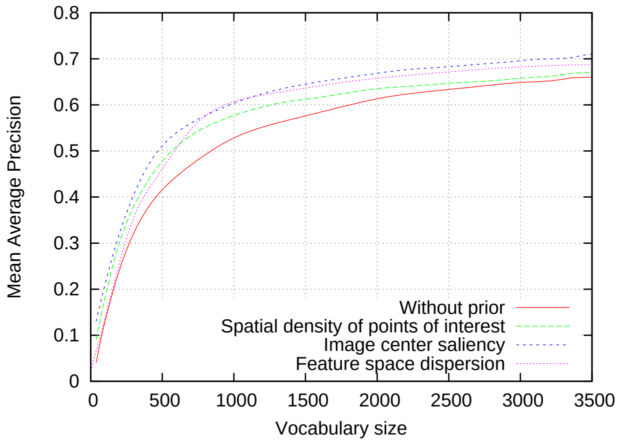

Figure 5: Comparison of priors

4.5 Small object discovery

To observe the effect of the priors on the type of objects detected, we launched 3 runs per prior. Each run stopped after having generated 600 words. Then we manually counted the mean number of detections for each prior/type pair among the words having over 5 matching images.

As we can see in Table 1, without using a prior, we mostly detect textured objects, which are not often very interesting. We show that we can decrease this number of textures and therefore increase the number of objects of interest, like logos. The use of an appropriate prior allows to discover a minimum of 47 faces or 70 logos, among 600 visual words and 10,000 images, in less than 15 minutes.

	Without prior	Dispersion	Face
Objects	31	53	20
Logos	46	**70**	27
Textures	**72**	26	23
Faces	12	13	**47**

Table 1: Number of detection per categories and priors

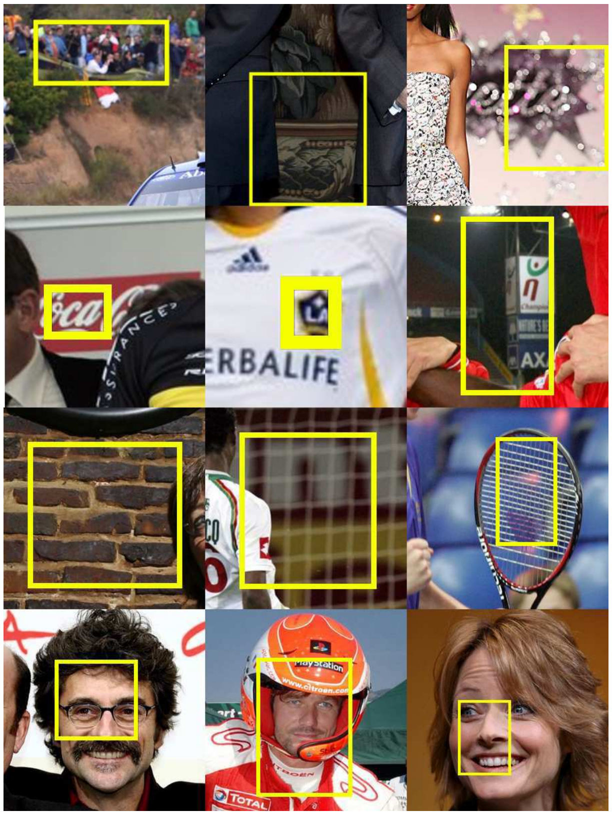

Figure 6: Examples of twelve generated CVW represented by one of their patches. The original images are cropped around the matching regions. 1st row: Miscellaneous objects, 2nd row: Logos, 3rd row: Textures, 4th row: Faces

4.6 Bags Of Consistent Visual Words Evaluation

We would like to illustrate that our consistent visual words are highly discriminant and robust. To demonstrate it, we propose to construct Bags-of-Consistent-Visual-Words to compare to the state-of the art methods based on BoW. Our proposal is a strategy equivalent to the BoW in [16, 14], but in our case, we replace the descriptor quantization by our consistent visual words. For **OXF** dataset, to our knowledge [14] has the best performances in terms of mAP score: 0.825. In [14], to get these performances, their proposal is based on BoW, Scalar Product as the similarity measure and Recursive Average Query Expansion method. With our Bags-of-Consistent-Visual-Words, we also used the Scalar Product and the Recursive Average Query Expansion method.

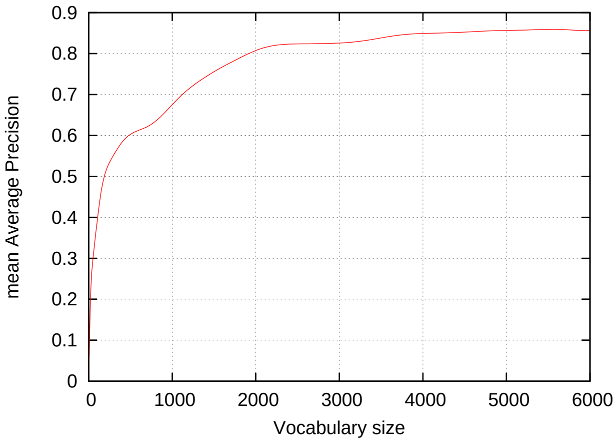

Figure 7: Test on OXF using $\alpha = 85\%$ and image center prior

	Philbin [14]	CVW
Vocabulary size	1,000,000	5,576
Words per image	3,000	9.14
mAP	0.825	0.861

Table 2: Comparison between the best BoW method (reproduced from [14]) and our consistent visual words

In [14], the size of their visual vocabulary is 1 million and each image is described with an average of 3,000 SIFT. As we can see in Table 2 and in Figure 7, we achieve our best mAP score (0.861) in less than 4 hours with only 5,576 words, and a mean of 9.14 words for each image. We can also notice in Figure 7 that we can reach a high mAP with a very few words. For example, a minimum of 2,500 words allows us to obtain a state-of-the-art score, while with only 250 words we got over 0.50 of mAP, in approximately 10 minutes. In the case of Bags-of-Words indexed by inverted lists, this low number of words involves a high degree of compression and a very fast similarity search.

5. CONCLUSION AND FUTURE WORKS

We have proposed the concept of consistent visual words and the experiments show that this type of words is very effective to describe and retrieve local visual contents. The experimentations with **OXF** demonstrate that these consistent visual words largely overcome the usual visual words. To generate consistent visual words, we developed a framework based on Adaptive Sampling and Priors.

The core idea is that the visual vocabulary produced by our adaptive sampling method might be adapted to what the user is searching for. We observe that any visual concept detector (e.g. animals, plants, indoor/outdoor, etc.) could be used as prior knowledge, which demonstrates that our method is widely generic.

This framework is very flexible and it is easy to integrate information by formulating a specific prior.

In future works, we plan to study the effects of the parameter γ which defines the maximum scale of consistent visual words. We could also propose to adapt this object scale in function of image contents and then the local region queries could be adaptive. To reduce the number of Tentative Probes, we would like to develop new types of priors, and to study the fusion of different priors. For example, we could fuse the priors by using the standard AND and OR Probabilistic Operators.

6. ACKNOWLEDGEMENTS

This work was partially funded by the EU through the Integrated Project GLOCAL (http://www.glocal-project.eu), and by ANR (France) through the Project OTmedia (www.otmedia.fr).

7. REFERENCES

[1] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, March 2003.

[2] A. Broder. On the resemblance and containment of documents. In *SEQUENCES '97: Proceedings of the Compression and Complexity of Sequences 1997*, page 21, Washington, DC, USA, 1997. IEEE Computer Society.

[3] O. Chum, M. Perdoch, and J. Matas. Geometric min-hashing: Finding a (thick) needle in a haystack. In *CVPR*, pages 17–24. IEEE, 2009.

[4] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman. Total recall: Automatic query expansion with a generative feature model for object retrieval. In *Proceedings of the 11th International Conference on Computer Vision, Rio de Janeiro, Brazil*, 2007.

[5] O. Chum, J. Philbin, and A. Zisserman. Near duplicate image detection: min-hash and tf-idf weighting. In *Proceedings of the British Machine Vision Conference*, 2008.

[6] L. Devroye and L. Devroye. Non-uniform random variate generation. 1986.

[7] W. Dong, M. Charikar, and K. Li. Asymmetric distance estimation with sketches for similarity search in high-dimensional spaces. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pages 123–130, New York, NY, USA, 2008. ACM.

[8] T. Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Mach. Learn.*, 42:177–196, January 2001.

[9] H. Jégou, M. Douze, and C. Schmid. Hamming Embedding and Weak Geometry Consistency for Large Scale Image Search. Research Report RR-6709, INRIA, 2008.

[10] A. Joly and O. Buisson. A Posteriori Multi-Probe Locality Sensitive Hashing. In *ACM International Conference on Multimedia (MM'08)*, pages 209–218, Vancouver, British Columbia, Canada, oct 2008.

[11] A. Joly and O. Buisson. Logo retrieval with a contrario visual query expansion. In *Proceedings of the seventeen ACM international conference on Multimedia*, MM '09, pages 581–584, New York, NY, USA, 2009. ACM.

[12] D. Lowe. Object recognition from local scale-invariant features. In *iccv*, page 1150. Published by the IEEE Computer Society, 1999.

[13] F. Olken. *Random Sampling from Databases*. PhD thesis, 1993.

[14] J. Philbin. *Scalable Object Retrieval in Very Large Image Collections*. PhD thesis, University of Oxford, 2010.

[15] 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.

[16] 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.

[17] J. Rabin, J. Delon, Y. Gousseau, and L. Moisan. MAC-RANSAC: a robust algorithm for the recognition of multiple objects. 2009.

[18] T. S.K. Adaptive sampling. In *The Survey Statistician*, 1995.

[19] T. Tuytelaars, C. H. Lampert, M. B. Blaschko, and W. Buntine. Unsupervised object discovery: A comparison. *Int. J. Comput. Vision*, 88:284–302, June 2010.