International Journal of Computer Vision manuscript No.
(will be inserted by the editor)

Learning Vocabularies over a Fine Quantization

Andrej Mikulik · Michal Perdoch · Ondřej Chum · Jiří Matas

Received: date / Accepted: date

Abstract A novel similarity measure for bag-of-words type large scale image retrieval is presented. The similarity function is learned in an unsupervised manner, requires no extra space over the standard bag-of-words method and is more discriminative than both L2-based soft assignment and Hamming embedding.

The novel similarity function achieves mean average precision that is superior to any result published in the literature on the standard Oxford 5k, Oxford 105k and Paris datasets/protocols.

We study the effect of a fine quantization and very large vocabularies (up to 64 million words) and show that the performance of specific object retrieval increases with the size of the vocabulary. This observation is in contradiction with previously published methods. We further demonstrate that the large vocabularies increase the speed of the tf-idf scoring step.

1 Introduction

Recently, large collections of images have become readily available [Google Street View, , Panoramio, , Flickr,] and image-based search in such collections has attracted significant attention of the computer vision community [Sivic and Zisserman, 2003, Nister and Stewenius, 2006, Chum et al., 2007, Jegou et al., 2008, Perdoch et al., 2009]. Most, if not all, recent state-of-the-art methods extend the bag-of-words representation introduced by Sivic and Zisserman [Sivic and Zisserman, 2003] who represented the image by a histogram of "visual words", *i.e.*, discretized SIFT descriptors [Lowe, 2004]. The bag-of-words representation possesses many desirable properties required in large scale retrieval. If implemented as an inverted file, it is compact and supports fast search. It is sufficiently discriminative and yet robust to acquisition "nuisance parameters" like illumination and viewpoint change as well as occlusion[1].

Discretization of SIFT features is necessary in large scale problems as it is neither possible to compute distances on descriptors efficiently nor feasible to store all the descriptors. Instead, only the identifier of the vector quantized prototype for visual word is kept. After quantization, Euclidean distance in a high (128) dimensional space is approximated by a 0–∞ pseudo-metric – features represented by the same visual word are deemed identical, while the others are treated as "totally different". The computational convenience of such a crude approximation of the SIFT distance has a detrimental impact on discriminative power of the representation. To relieve this problem, recent methods like soft assignment [Philbin et al., 2008] and in particular the Hamming embedding [Jégou et al., 2010] aim at obtaining a better space-speed-accuracy trade off.

In this paper, unsupervised learning on a large set of images is exploited to improve the 0–∞ metric. First, an efficient clustering process with spatial verification establishes correspondences within a large (>5M) image collection. Next, a fine-grained vocabulary is obtained by 2-level hierarchical approximate nearest neighbour clustering. The automatically established correspondences are then used to define a similarity measure on the basis of a probabilistic relationships of visual words; we call it the *PR visual word similarity*.

A. Mikulik · M. Perdoch · O. Chum · J. Matas
CMP, Dept. of Cybernetics, Faculty of Electrical Engineering,
Czech Technical University in Prague
Fax: +420-24355731
E-mail: {mikulik,predom1,chum,matas}@cmp.felk.cvut.cz

[1] We only consider and compare with methods that support queries that cover only a (small) part of the test image. Global methods like GIST [Oliva and Torralba, 2006] achieve a much smaller memory footprint at the cost of allowing whole image queries only.

$ The authors were supported by the GACR P103/12/2310 project.

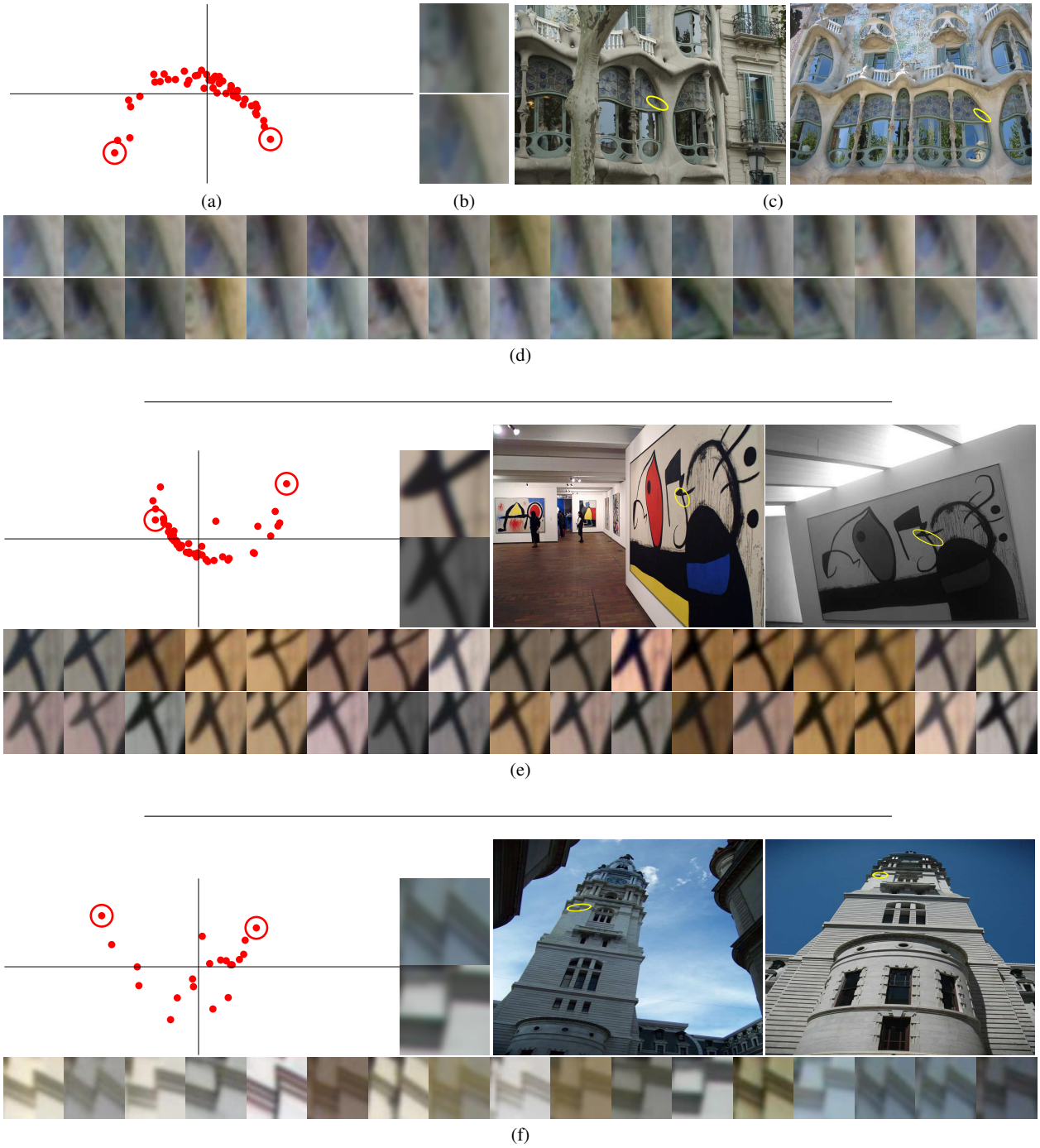


Fig. 1 Examples of "tracks", *i.e.*, of corresponding patches. (a) A 2D PCA projection of their SIFT descriptors; (b) two most distant patches in the SIFT space and (c) the images where they were detected; (d) a set of sample patches. The average SIFT distance within the cluster is 278, the maximal distance is 591. For a comparison, an average distance of two randomly selected SIFT descriptors is 540. Two other examples of tracks where (e) enormous change of viewpoint caused that the maximal SIFT distance inside track is 542 and (f) 593 respectively, where change in scale is also present.

When combined with a large vocabulary, several millions of words (one or two orders of magnitude larger than commonly used), the PR similarity has the following desirable properties:

(i) it is more accurate (discriminative), than both standard 0–∞ metric and Hamming embedding.
(ii) the memory footprint of the image representation for PR similarity calculation is roughly identical to the standard method and smaller than that of Hamming embedding.
(iii) search with the PR similarity is faster than the standard bag-of-words.

As a main contribution of the paper, we present a novel similarity measure that is learned in an unsupervised manner, requires only negligible extra space (only O(1)) in comparison with the bag-of-words and is more discriminative than both 0–∞ and L2-based soft assignment.

Further, we experimentally disprove the common assumption which is present in community that is not worth to build vocabularies larger than 1M. To construct a well performing large vocabulary, we propose to build shallow hierarchical – tree based – vocabularies with adaptive branching to speed up the process but not to bring the disadvantage of big imbalance factor of deeper ones.

A preliminary version of this paper [Mikulik et al., 2010] appeared in European Conference on Computer Vision 2010.

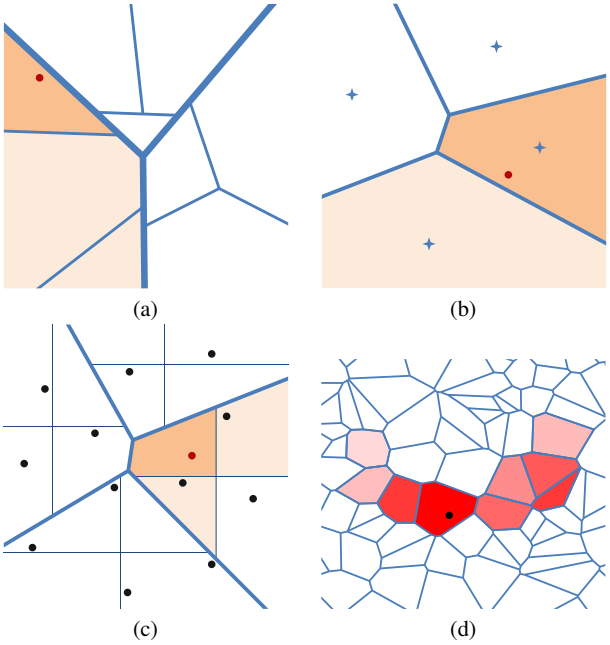


Fig. 2 Different approaches to soft assignment (saturation encodes the relevance): (a) hierarchical scoring [Nister and Stewenius, 2006] – the soft assignment is given by the hierarchical structure of the assignment tree; (b) soft clustering [Philbin et al., 2008] assigns features to r nearest cluster centers; (c) hamming embedding [Jégou et al., 2010] – each cell is divided into orthants by a number of hyperplanes, the distance of the orthants is measured by the number of separating hyperplanes; (d) the set of alternative words in the proposed PR similarity measure.

2 Related Work

In this section, approaches to vocabulary construction and soft assignment suitable for large-scale image search are reviewed and compared.

In [Sivic and Zisserman, 2003], the first 'bag of words' approach to image retrieval was introduced. The vocabulary (with the number of visual words $\approx 10^4$) is constructed using a standard k-means algorithm. Adopting methodology from text retrieval applications, the image score is efficiently computed by traversing inverted files related to visual words present in the query. The inverted file related to a visual word W is a list of image ids that contain the visual word W. It follows that the time required for scoring the documents is proportional to the number different visual words in a query and the average length of an inverted file.

Hierarchical clustering. The hierarchical k-means and scoring of Nistér and Stewenius [Nister and Stewenius, 2006] is the first image retrieval approach that scales up. The vocabulary has a hierarchical structure which allows efficient construction of large and discriminative vocabularies. The quantization effect are alleviated by the so called hierarchical scoring. In such a type of scoring, the scoring visual words are not only stored in the leafs of the vocabulary tree. The non-leaf nodes can be thought of as virtual or generic visual words. These virtual words naturally score with lower *idf* weights as more features are assigned to them (all features in their sub-tree).

The advantage of the hierarchical scoring approach is that the soft assignment is given by the structure of the tree and no additional information needs to be stored for each feature. On the downside, experiments in [Philbin et al., 2008] show that the quantization artefacts of the hierarchical k-means are not fully removed by hierarchical scoring, the problems are only shifted up a few levels in the hierarchy. An illustrative example of the soft assignment performed by the hierarchical clustering is shown in Fig. 2(a).

Soft assignment. In [Philbin et al., 2008], an approximate soft assignment is exploited. Each feature is assigned to $n = 3$ (approximately) nearest visual words. Each assignment is weighted by $e^{-\frac{d^2}{2\sigma^2}}$ where d is the distance of the feature descriptor to the cluster center.

The soft assignment is performed on features in the database as well as the query features. This results in n times higher memory requirements and n^2 times longer running time – the average length of the inverted file is n times longer and there are up to n times more visual words associated with the query features. For an illustration of the soft assignment, see Fig. 2(b).

Hamming embedding. Jégou et al. [Jégou et al., 2010] proposed to combine k-means quantization and binary vector

signatures. First, the feature space is divided into relatively small number of Voronoi cells (20K) using k-means. Each cell is then divided by n axis-parallel hyper-planes into 2^n subcells. Each subcell is described by a binary vector of length n. Results reported in [Jégou et al., 2010] suggest that the Hamming embedding provides good quantization. However, the good results are traded off for higher running time requirements and high memory requirements.

The higher running time requirements are caused by the use of coarse quantization in the first step. The average length of an inverted file for vocabulary of 20K words is 50 times longer than the one of 1M words. Recall that the time required to traverse the inverted files is given by the length of the inverted file. Hence 50 times smaller vocabulary results in approximately 50 times longer scoring time on average. Even if two query features are assigned to the same visual word, the relevant inverted file has to be processed for each of the features separately as they will have different binary signatures.

While the reported bits per feature required in the search index ranges from 11 bits [Perdoch et al., 2009] to 18 bits [Philbin et al., 2008], hamming embedding adds another 64 bits. The additional information reduces the number of features that can be stored in the memory by a factor of 6.8.

Gaussian Mixture Model (GMM). Learned with expectation-maximization (EM) algorithm [Duda et al., 1995], GMM is seen as generalization of k-means clustering [Perronnin, 2008]. While standard approach does not allow a large number of clusters due to slow convergence Approximate Gaussian Mixture (AGM) [Avrithis and Kalantidis, 2012] uses approximate nearest neighbor search to constrain the number of clusters, which interacts with a point. This enables clustering into 10^6 clusters and provides a natural way for soft assignment on both query and vocabulary side. Another advantages are is that AGM dynamically estimates the number of clusters.

While [Avrithis and Kalantidis, 2012] reports better results on vocabularies with up to 1M visual words, it does not exceed recall and precision of our system on compared standard dataset.

Summary All approaches to soft clustering mentioned above are based on the distance (or its approximation) in the descriptor (SIFT) space. It has been observed that the Euclidian distance is not the best performing measure. Learning a global Mahalanobis distance [Hua et al., 2007, Mikolajczyk and Matas, 2007] showed that the matching is improved and / or the dimensionality of the descriptor is reduced. However, even in the original work on SIFT descriptor matching [Lowe, 2004] it is shown that the similarity of the descriptors is not only dependent on the distance of the descriptors, but also on the location of the features in the feature space. Therefore, learning a global Mahalanobis metric is suboptimal and a local similarity measure is required. Examples of corresponding patches where SIFT distance does not predict well the similarity are depicted in Figure 1.

Similar approach [Makadia, 2010] to ours was published at the same conference as the preliminary version of this paper. In this work Makadia used simpler, symmetrical similarity measure, which together with much smaller training set yielded inferior results to ours. A complementary approach to identify alternative words was proposed in [Tang et al., 2011], where authors observed that visual words representing the same semantic meaning, tend to have similar visual contextual distributions.

3 The Probabilistic Relation Similarity Measure

Consider a feature in the query image with descriptor $D \in \mathcal{D} \subset R^d$. For most accurate matching, the query feature should be compared to all features in the database. The contribution of the query feature to the matching score should be proportional to the probability of matching the database feature. It is far too slow, *i.e.*, practically not feasible, to directly match a query feature to all features in a (large) database. Also, the contribution of features with low probability of matching is negligible.

The success of fast retrieval approaches is based on efficient separation of (potentially) matching features from those that are highly unlikely to match. The elimination is based on a simple idea – the descriptors of matching patches will be close in some appropriate metric (L2 is often used). With appropriate data structures, enumeration of descriptors in proximity is possible in time sub-linear in the size of the database. All bag-of-words based methods use partition $\{w_i\}$ of the descriptor space $\mathcal{D}$: $\bigcup w_i = \mathcal{D}, \quad w_i \cap w_{j \neq i} = \emptyset$. The cells are then used to separate features that are close (potentially matching) from those that are far (non-matching).

In the case of hard assignment, features are associated with the visual words defined by the closest cluster center. In the scoring that evaluates query and database image match, only features with the same visual word as the query feature are considered.

We argue that the descriptor distance is a good indicator of patch similarity only up to a limited distance, where the variation in the descriptors is caused mostly the imaging and detector noise. We abandon the assumption that the descriptor distance provides a good similarity measure of patches observed under different viewing angles or under different illumination conditions. Instead, we propose to estimate the probability between a feature observed in the query image and a database feature. Since our aim is to address retrieval in web-scale databases where store requirements are critical, we constrained our attention to solution that have a minimal overhead in comparison with the standard inverted file representation.

The proposed approach. We propose to use a fine partition of the descriptor space, to minimize a probability of false match inside a single cell. Even though the fine partition is learned in a data dependent fashion (as in the other approaches), the fine partition unavoidable separates matching features into a number of cells.

For each cell (visual word) we learn which other cells (called *alternative visual words*) are likely to contain descriptors of matching patches with the same pre-images. This step consist of estimating the probability of observing visual word w_j in a matching database image when visual word w_q was observed in the query image

$$P(w_j|w_q). \tag{1}$$

The probability (1) is estimated from a large number of matching patches.

A simple generative model, independent for each feature, is adopted. In the model, image features are assumed to be (locally affine) projections of a (locally close to planar) 3D surface patches z_i. Hence, matching features among different images are those that have the same pre-image z_i. To estimate the probability $P(w_j|w_q)$ we start with (a large number of) sets of matching features, each set being different projections of a patch z_i. Using the fine vocabulary (partition) the sets of matching features are converted to sets of matching visual words. We estimate the probability $P(w_j|w_q)$ as

$$P(w_j|w_q) \approx \sum_{\mathcal{Z}} P(w_j|z_i)P(z_i|w_q). \tag{2}$$

For each visual word w_q, a fixed number of alternative visual words that have the highest conditional probability (eqn. 2) is recorded.

4 Learning a PR similarity

The first step of our approach is to obtain a large number of matching image patches. The links between matching patches are consequently used to infer relationship, between quantized descriptors of those patches, *i.e.*, between visual words. As a first step towards unsupervised collection of matching image patches, called "feature tracks", clusters of matching images are discovered. Within each cluster, feature tracks are found by a wide-baseline matching method. This approach is similar to [Agarwal et al., 2009], where the feature tracks are used to produce 3D reconstruction. In our case, it is important to find larger variety of patch appearances than precise point locations. Therefore, we adopt a slightly different approach to the choice of image pairs investigated.

4.1 Image clusters

We start with analyzing connected components of the image matching graph (graph with images as vertices, edges connect images that can be matched) produced by a large-scale clustering method [Chum and Matas, 2010, Li et al., 2008]. Any matching technique is suitable provided it can find clusters of matching images in a very large database. In our case, an image retrieval system was used to produce the clusters of spatially related images. The following structure of image clusters is created. Each cluster of spatially related images is represented as an oriented tree structure (the skeleton of the cluster). The children of each parental node were obtained as results of an image retrieval using the parent image as a query image. Retrieved images, which are already in the cluster, are ignored. Together with the tree structure, an affine transformation (approximately) mapping child image to its parent are recorded. These mappings are later used to guide (speed-up) the matching.

4.2 Feature tracks

To avoid any kind of bias (by quantization errors, for example), instead of using vector quantized form of the descriptors, the conventional image matching (based on the full SIFT [Lowe, 2004]) has to be used. In principle, one can go back even to the pixel level [Ferrari et al., 2004, Cech et al., 2008], however such an approach seems to be impractical for large volumes of data.

It is not feasible to match all pairs of images in the image clusters, especially not of clusters with large number of images (say more than 1000). It is also not possible to simply follow the tree structure of image clusters because not all features are detected in all images (in fact, only a relatively small portion of features is actually repeated). The following procedure, that is linear in the number of images in the cluster, is adopted for detection of feature tracks that would exhibit as large variety of patch appearances as possible. For each parental node, a sub-tree of height two is selected. On images in the sub-tree, a $2k$-connected graph called circulant graph [Godsil and Royle, 2001] is constructed. Vertices of a graph are ordered and connected with K steps of the length random chosen between 1 and $\lfloor (N-1)/2 \rfloor$ but always including step 1, to force connectivity. (*i.e.*. for chosen step 4, the edges are created between vertices $v_i, v_j \in V$, where $i - j \bmod N = 4$). The algorithm for construction of minimal $2k$-connected graph is summarized in Algorithm 1.

Images connected by an edge in such a graph are then matched using standard wide-baseline matching. Since each image in the image cluster participates in at most 3 sub-trees (as father, son and grand-son), the number of edges is limited to $6kN$, where N is the size of the cluster. Instead of using epipolar geometry as a global model, a number of close-

to-planar (geometrically consistent) structures is estimated (using affine homography). Unlike the epipolar constraint, such a one-to-one mapping enables to verify the shape of the feature patch. Connected components of matching and geometrically consistent features are called *feature tracks*.

Tracks that contain two different features from a single image are called inconsistent [Agarwal et al., 2009]. These features clearly cannot have a single pre-image under perspective projection and hence cannot be used in the process of 3D reconstruction. Such inconsistent tracks are often caused by repeated patterns. Inconsistent feature tracks are (unlike in [Agarwal et al., 2009]) kept as they provide further examples of patch appearance.

Input: K - requested connectivity, N - number of vertices
Output: V a set of vertices, $E \subset V \times V$ a set of edges of $2K$ connected graph (V,E).

1. **if** $2K \geq N - 1$ **then**
   return fully connected graph with N vertices.
   end
2. $S := \{1\} \bigcup$ a random subset of $\{2, \ldots, \lfloor \frac{N-1}{2} \rfloor\}$,$|S| = K$
3. $V := \{v_0, \ldots, v_{N-1}\}$
4. $E := \{(v_i, v_j) \,|\, v_i, v_j \in V, i - j \bmod N \in S\}$

Algorithm 1: Construction of the 2K connected graph with a minimal number of edges as a union of circulants.

4.3 Computing the conditional probability.

To compute the conditional probability (eqn. 2) from the feature tracks, an inverted file structure is used. The tracks are represented as forward files (named z_i), *i.e.*, lists of matching SIFT descriptors. The descriptors are assigned to their visual word from the large vocabulary. Then, for each visual word w_k, a list of patches z_i so that $P(z_i|w_k) > 0$ (the inverted file) is constructed. The sum (eqn. 2) is evaluated by traversing the relevant inverted file.

4.4 Statistics.

Over 5 million images were processed using geometric minhash technique [Chum et al., 2009]. Almost 20,000 clusters containing 750,000 images were found. Out of those 733,000 were successfully matched in the wide-baseline matching stage. Over 111 million of feature tracks were established, out of which 12.3 millions are composed of more than 5 features. In total, 564 million features participated in the tracks, 319.5 million features belong to tracks of more than 5 features. Some examples of feature tracks are shown in Figures 6 and 7. Only negligible portion of visual words

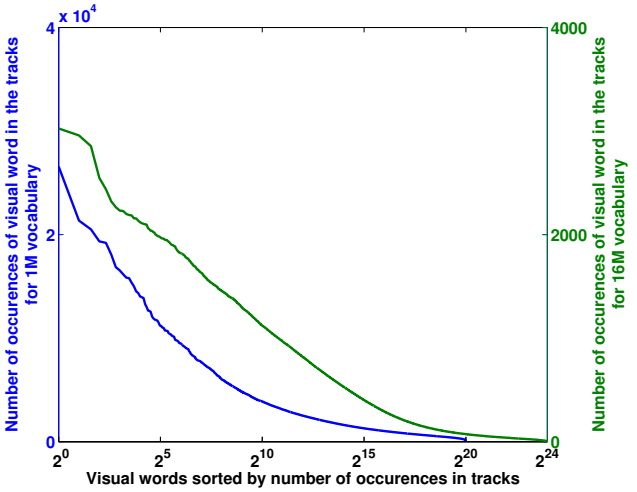


Fig. 3 The distribution of visual words over tracks in 1M and 16M vocabulary. Only negligible part of the visual words were not present in any feature track.

were not present in any feature track. There was 2 such words in the 1M vocabulary and 74005 (0.4%) in 16M vocabulary. The distribution of visual words over tracks in these vocabularies are shown in Figure 3.

4.5 Memory and time efficiency.

For the alternative words storage, only constant space is required, equal to the size of the vocabulary times the number of alternative words. The pre-processing consists of image clustering ([Chum and Matas, 2010] reports near linear time in the size of the database), intra-cluster matching (linearity enforced by the $2k$-connected circulant matching graph), and of the evaluation of expression eqn. (2) for all visual words. The worst case complexity of the last step is equal to the number of tracks (correspondences) times size of the vocabulary squared. In practice, due to the sparsity of the representation, the process took less than an hour in our settings for over 5 million images.

5 Large Vocabulary Generation

To efficiently generate a large visual vocabulary we employ a hybrid approach – approximate hierarchical k-means. A hierarchy tree of two levels is constructed. For instance, for vocabulary of 16M words, each level has 4K nodes on average. In the assignment stage of k-means, an approximate nearest neighbour, FLANN [Muja and Lowe, 2009], is used for efficiency reasons.

First, a level one approximate k-means is applied to a random sub-sample of 5 million SIFT descriptors. Then, a two pass procedure on $\approx$ 11 billion SIFTs (from almost 6

million images) is performed. In the first pass, each SIFT descriptor is assigned to a word in the level one of a vocabulary. For each visual word in the first level a list of descriptors assigned to it is recorded. In the second pass, approximate k-means on each list of the descriptors is applied. The whole procedure takes about one day on a cluster of 20 computers.

5.1 Balancing the tree structure.

For the average speed of the retrieval, it is important that the vocabulary is balanced, *i.e.*, there are approximately the same number of instances of each visual word in the database.

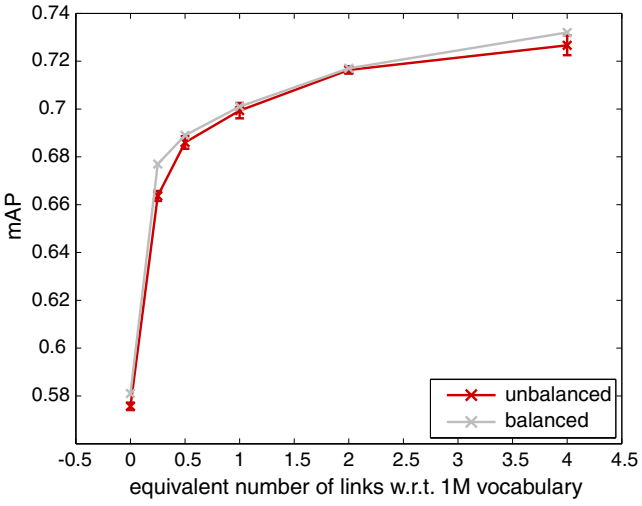
We compared unbalanced and balanced vocabulary constructions (Figure 4). In the balanced construction, the second level of the vocabulary uses an adaptive branching factor, which is proportional to the weight of the branch (*i.e.* cluster A with 2 times more features than cluster B will be split into two times more clusters in the second level of hierarchy than cluster B). We also explored the balancing on the first level by constraining the length of the mean vectors (this stems from the fact that SIFT features live approximately on a hyper-sphere), which is similar to the method [Tavenard and Amsaleg, 2010]. As the latter method has not brought better results while implied higher computation costs, it was not explored further.

In our experiments, a balanced vocabulary with adaptive branching factor at the second level is used. With such a construction we reached an imbalance factor [Jégou et al., 2010] of 1.09 for the training image set (>5M images) (compared to 1.21 in [Jégou et al., 2010]) and 1.26 for the testing set – Oxford 105k. Fraundorfer et al. [Fraundorfer et al., 2007] report estimate of imbalance factor 5 for hierarchical trees introduced in [Nister and Stewenius, 2006]. The experiment shows that the balancing does not significantly affect mAP. The advantage is the gain in query speed.

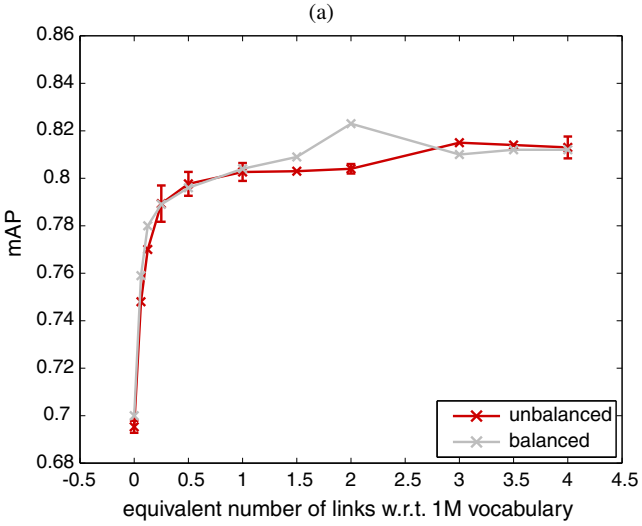
Comparison of the imbalance factors of our balanced and unbalanced vocabulary is show in Table 1.

5.2 Size of the vocabulary.

There are different opinions about the number of visual words in the vocabulary for image retrieval. Philbin et. al. in [Philbin et al., 2007] achieved the best mAP for object recognition with a vocabulary of 1M visual words and predict a performance drop for larger vocabularies. We attribute the result in [Philbin et al., 2007] to a too small training dataset (16.7M descriptors). In our case the vocabularies with up to 64M words is built using 11G training descriptors. Experiments show that the larger the vocabulary is, the better performance is achieved, even for plain bag-of-words retrieval.



(a)



(b)

Fig. 4 A comparison of the mean average precision (mAP) for an unbalanced and balanced 16M vocabulary (a) with and (b) without the query expansion. The experiment shows that the balancing does not significantly affect mAP (the advantage is the gain in query speed). The error bars are shown where three vocabularies with different random initialization were evaluated.

Introducing the alternative words, the situation is changed even more rapidly and, as expected, they are more useful for larger vocabularies (Figure 5). We have not built vocabularies larger than 64M because the memory footprint of the assignment tree started to be impractical and the performance has almost converged.

6 Experiments

The implementation of the retrieval stage is fairly standard, using inverted files [Sivic and Zisserman, 2003] for candidate image selection which is followed by fast spatial verification and query expansion [Chum et al., 2007]. The modi-

method of the vocab. construction	imbalance factor level 1		imbalance factor level 2	
	training set	testing set	training set	testing set
unbalanced	1.028	1.097	1.122	1.311
balanced	1.028	1.097	1.093	1.259

Table 1 Comparison of the imbalance factor [Jégou et al., 2010] of the unbalanced and balanced version of the two level hierarchical vocabulary. Adaptive branching factor was used at the second level of the tree hierarchy to balance the vocabulary.

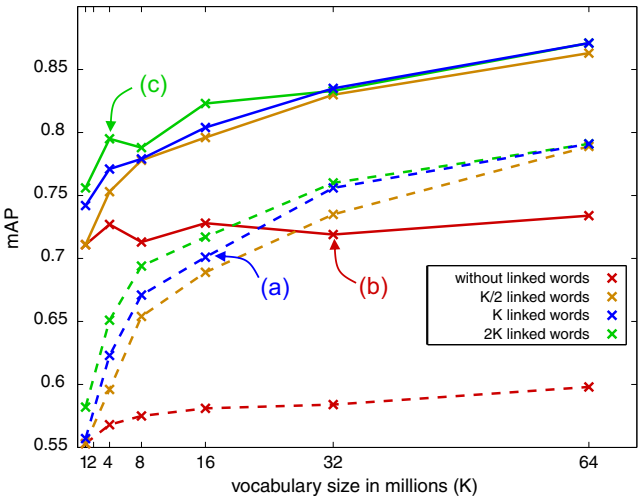


Fig. 5 Comparison of mAP for the balanced vocabularies of from 1 to 64 millions visual words. Solid lines show results after the query expansion (QE), dashed lines without QE. Red lines show results using plain bag-of-words (without alternative words). The number of alternative words is proportional to vocabulary size to compare results of equal time complexity. This way, approximately the same amount of entries of the inverted file is traversed (*e.g.* since the average length of a list of an inverted file for 16M vocabulary is 16 times smaller than for 1M vocabulary, 16 lists with alternative words can be crawled within the same time). To clarify the plot: (a) the result of 16M vocabulary with (L16) 16 linked words (1 original and 15 alternatives) and without QE. (b) 32M vocabulary (L1) without alternative words with QE, and finally (c) 4M vocabulary (L8) 8 linked words with QE.

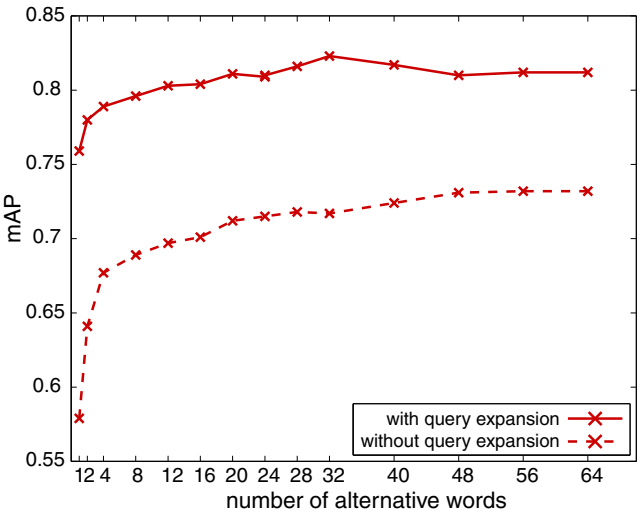


Fig. 8 The quality of the retrieval, expressed as the mean average precision (mAP), increases with the number of alternative words. The mAP after (upper curve) and before (lower curve) query expansion is shown.

fications listed below are the major differences implemented in our retrieval stage.

Unique matching. Despite being assigned to more than one visual word, each query feature is a projection of a single physical patch. Thus it can match only at most one feature in each image in the database. We find that applying this uniqueness constraint adds negligible computational cost and improves the results by approximately 1%. The order in which are the alternative words traversed and matched in an inverted file is given by their probability of being an alternative word (2).

Weights of alternative words. Contribution of each visual word is weighted by the *idf* weight [Baeza-Yates and Ribeiro-Neto, 1999]. A number of re-weighting schemes for alternative words have been tried, none of them affecting significantly the results of the retrieval.

Datasets. We have extensively evaluated the performance of the PR similarity on a standard retrieval datasets Oxford 105K and Oxford 5K[2], INRIA Holidays[3], PARIS[4] and PARIS combined with 100.000 distractor images from Oxford 105K dataset (Paris + Oxford 100K). The experiments focus on retrieval accuracy and the retrieval speed. Since our training set of 6 million images were downloaded from FLICKR in a similar way as the testing datasets Oxford and PARIS, we have explicitly removed all testing images (or their scaled duplicates) from the training set.

6.1 Retrieval quality

We follow the protocols of testing datasets defined in [Philbin et al., 2007] and use the mean average precision as a measure of retrieval performance. We start by studying the properties of the PR similarity for a visual vocabularies of 1, 4, 8, 16, 32 and 64 million words.

In the first experiment, the quality of the retrieval as a function of the number of alternative words is measured, see Figure 8. The plots show that performance improves for visual vocabularies of all tested sizes monotonically for plain retrieval without query expansion and almost monotonically when query expansion is used.

[2] http://www.robots.ox.ac.uk/∼vgg/data/oxbuildings/
[3] http://lear.inrialpes.fr/∼jegou/data.php
[4] http://www.robots.ox.ac.uk/∼vgg/data/parisbuildings/

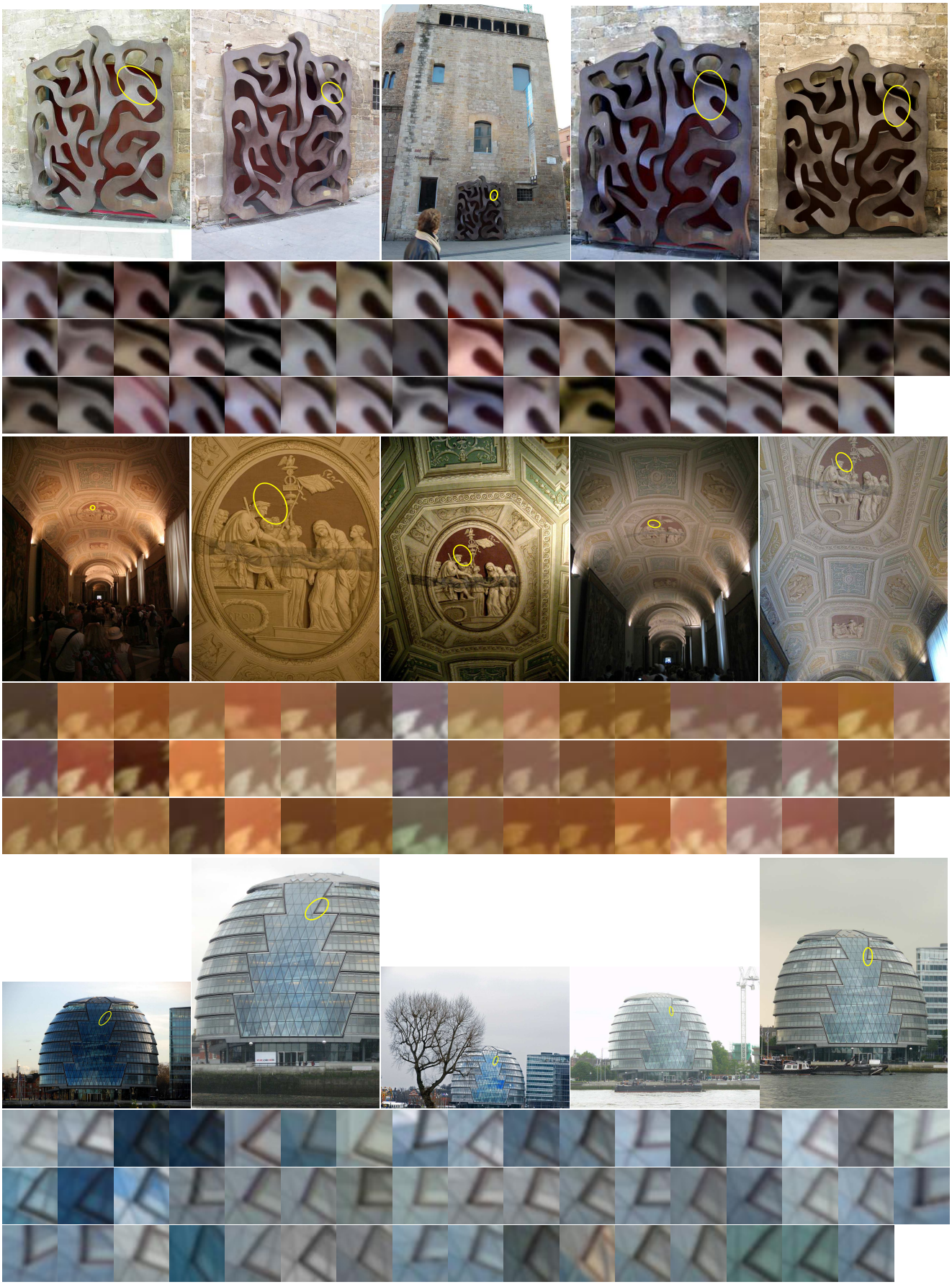


Fig. 6 Three examples of feature tracks of size 50. Five selected images (top row) and all 50 patches of the track. Even though the patches are similar, the SIFT distance of some pairs is over 500.

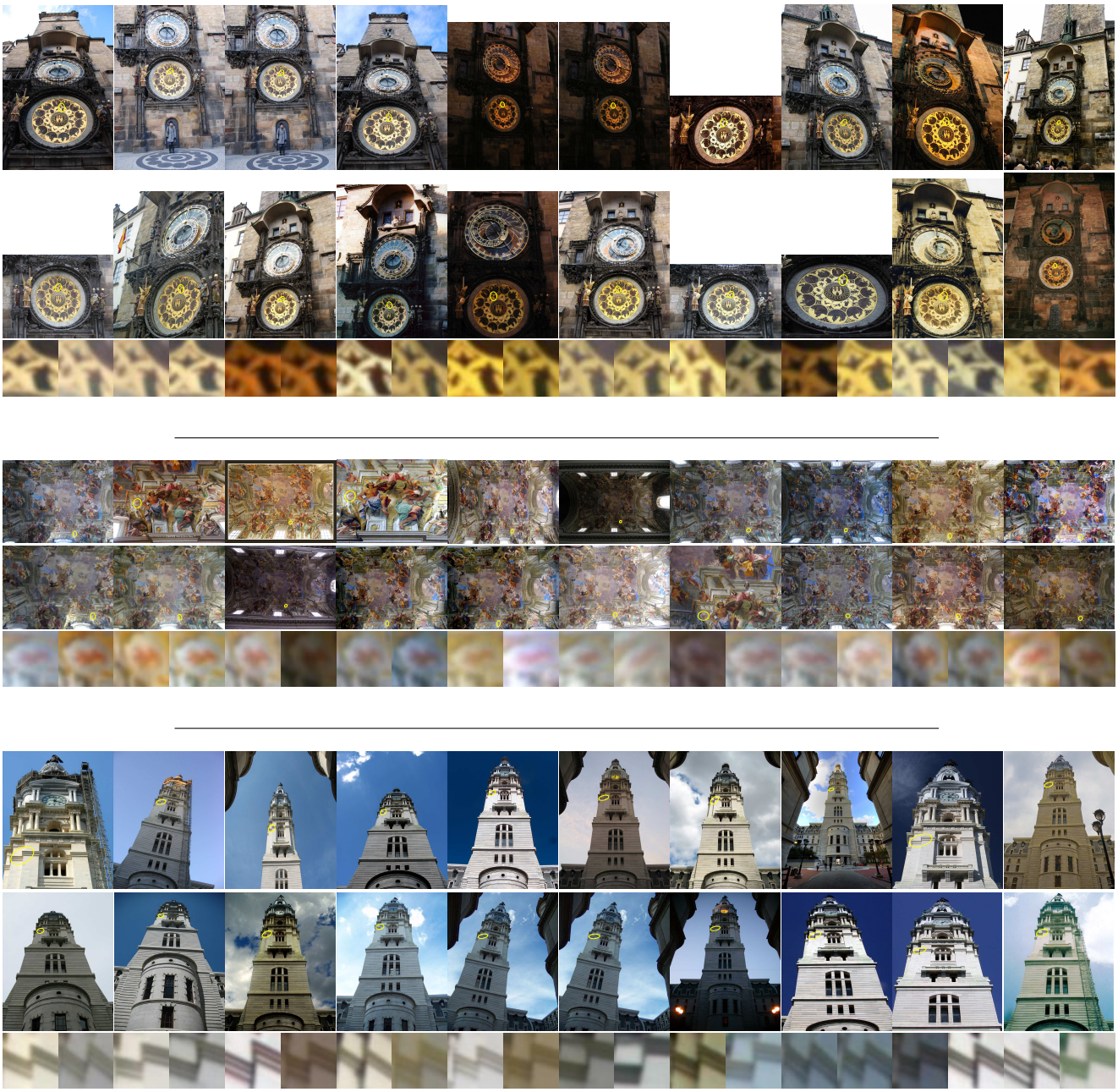


Fig. 7 Three examples of feature tracks of size 20. Images (first two rows) and corresponding patches (third row). Note the variation in appearance of the patches.

The second experiment studies the effects of the vocabulary size (Figure 5), and compares the alternative words in the PR similarity with the euclidean nearest neighbours in soft assignment. The left-hand part of Table 2 shows results obtained with the 16M vocabulary with three different settings 'L1' – standard tf-idf retrieval with hard assignment of visual words; 'L5' and 'L16' – retrieval using alternative words (4 and 15 respectively). The righthand part presents results of reference state-of-the-art results [Perdoch et al., 2009] obtain with a vocabulary of 1M visual words learned on the PARIS dataset. Two version of the reference algorithm are tested, without ('L1') and with the query soft assignment to 3 nearest neighbours ('SA 3NN').

The experiments support the following observations:

(i) PR similarity calculation with using the learned alternative words increases significantly the accuracy of the retrieval, both with and without query expansion.
(ii) Alternative words are more useful for larger vocabularies
(iii) The PR similarity outperforms soft SA in term of precisions, yet does not share the drawbacks of SA.

	16M L1	16M L5	16M L16	PARIS 1M L1
Oxford 105K	0.071	0.114	0.195	0.247

Table 3 Average execution time per query in sec for selected vocabularies on Oxford 105k dataset. The proposed 16M vocabulary is compared with the state-of-the-art method [Perdoch et al., 2009].

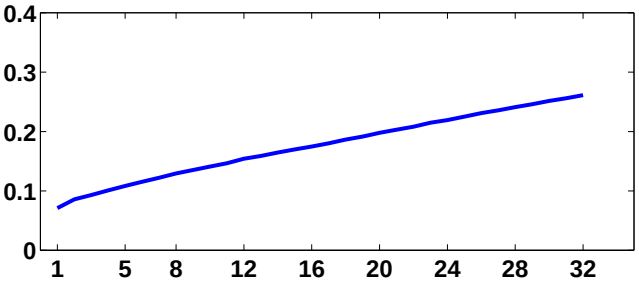


Fig. 9 The dependence of the query time on the number of linked words for Oxford 105k dataset and 16M vocabulary.

(iv) The PR similarity outperforms the Hamming embedding approach combined with query expansion, Jegou et al. [Jégou et al., 2009, Jégou et al., 2010] report the mAP of 0.692 on this dataset.

(v) The mAP result for 16M L16 is superior to any result published in the literature on the Oxford 105k dataset.

(vi) Balancing by uneven splitting of the second layer discard drawbacks of growing imbalance factor for hierarchical vocabularies. We predict that this approach will be even more significant for deeper vocabularies.

6.2 Query times

To compare the speed of the retrieval, an average query time over the 55 queries defined on the Oxford 105K data set was measured. Running times recorded for the same methods and parameter settings as above are shown in Table 3.

The plot showing dependency of the query time on the number of alternative words is depicted in Figure 9. The time for the reference PARIS 1M std method and the 16M L16 are of the same order. This is expected since the average length of inverted files is of the same order for both methods. The proposed method is about 20% faster, but this might be just an implementation artefact.

We looked at the dependence of the speed of the proposed method as a function of the number alternative words. The relationship shown in Fig. 9 is very close to linear plus a fixed overhead. The plot demonstrates that speed-accuracy trade-off is controllable via the number of alternative words.

Finally, the average query time for plain bag-of-words (no alternative words) as a function of the dictionary size was evaluated. To measure directly the speed of traversing the inverted file, the query time without the spatial verification is measured. Results are shown in Figure 10.

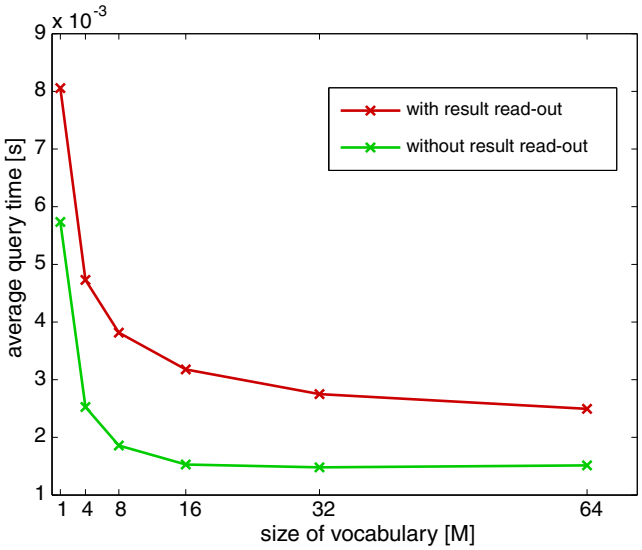


Fig. 10 The dependence of the query time on the vocabulary size. The times were measured on the Oxford 105k dataset. To measure the speed of inverted file, we are not using spatial verification, alternative words, or query expansion. The green line shows times measured without sorting the documents according the score and copying them to the output.

6.3 Results on other datasets

The proposed approach has been tested on a number of standard datasets. These include Oxford, INRIA holidays [5], and Paris datasets. In all cases (Table 4), the use of the alternative visual words improves the results. On all datasets except the INRIA holidays the method achieves the state-of-the-art results.

The proposed method is designed and trained to improve retrieval of specific object by better matching of features that are projections of *identical physical scene patch*. In the INRIA dataset, it is known that many queries rely on retrieving similar content rather than on exact feature matching. We consider this property of the dataset to be the reason for relatively small increase in the performance by our method.

7 Conclusions

We presented a novel similarity measure for bag-of-words type large scale image retrieval. The similarity function is learned in an unsupervised manner using geometrically verified correspondences obtained with an efficient clustering method on a large image collection.

[5] The Holidays dataset presented in [Jegou et al., 2008] contains about 5%-10% of the images rotated unnaturally for a human observer. Because the rotational variant feature descriptor was used in our experiment, we report the performance on a version of the dataset, with corrected orientation of the images according to EXIF, or manually (by 90°, 180° or 270°), where the EXIF information is missing and the correct (sky-is-up) orientation is obvious.

	16M L1	16M L5	16M L16	PARIS 1M L1	PARIS 1M SA 3NN
plain	0.554	0.650	**0.674**	0.574	0.652
QE	0.695	0.786	**0.795**	0.728	0.772

Table 2 The mean average precision for the 16M vocabulary on the Oxford 105k dataset is compared with the previous stat-of-the-art 1M vocabulary learned on Paris dataset [Perdoch et al., 2009]. Setups with hard assignment (L1), 4 alternative words (L5), 15 alternative words (L16) and soft-assignment with 3 nearest neighbours (SA 3NN) were considered. Results without (plain) and with query expansion (QE) are shown.

Dataset	16M L1	16M L16	16M QE	16M L16 QE
Oxford 5k	0.618	0.742	0.740	0.849
Oxford 105K	0.554	0.674	0.695	0.795
Paris	0.625	0.749	0.736	0.824
Paris + Oxford 100k	0.533	0.675	0.659	0.773
INRIA holidays rot	0.742	0.749	0.755	0.758

Table 4 Results of the proposed method on a number of publicly available datasets for a vocabulary with 16 millions visual words. Four setups are compared: (L16) with 15 alternative words, (L1) without alternative words, with and without (QE) query expansion. (The result for the Oxford 105K is duplicated for completeness.)

The similarity measure requires only negligible extra space in comparison with the standard bag-of-words method. Experimentally we show that the novel similarity function achieves mean average precision that is superior to any result published in the literature on the standard Oxford 5k, Oxford 105k and Paris datasets/protocols. At the same time, retrieval with the proposed similarity function is faster than the reference method.

We showed that using 2 layer hierarchical approach enables to build a large vocabulary, which performs better and faster and proposes the simple balancing method, which helps to keep imbalance factor low.

As a secondary contribution we make available the database of matching SIFT features, together with the source code of the feature detector (Hessian affine) and descriptor used to extract and describe the features [Project page, 2012].

References

[Agarwal et al., 2009] Agarwal, S., Snavely, N., Simon, I., Seitz, S., and Szeliski, R. (2009). Building rome in a day. In *Proc. ICCV*.

[Avrithis and Kalantidis, 2012] Avrithis, Y. and Kalantidis, Y. (2012). Approximate gaussian mixtures for large scale vocabularies. In *Proceedings of European Conference on Computer Vision (ECCV 2012)*.

[Baeza-Yates and Ribeiro-Neto, 1999] Baeza-Yates, R. and Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. ACM Press, ISBN: 020139829.

[Cech et al., 2008] Cech, J., Matas, J., and Perdoch, M. (2008). Efficient sequential correspondence selection by cosegmentation. In *Proc. CVPR*.

[Chum and Matas, 2010] Chum, O. and Matas, J. (2010). Large-scale discovery of spatially related images. *IEEE PAMI*, 32:371–377.

[Chum et al., 2009] Chum, O., Perdoch, M., and Matas, J. (2009). Geometric min-hashing: Finding a (thick) needle in a haystack. In *Proc. CVPR*.

[Chum et al., 2007] Chum, O., Philbin, J., Sivic, J., Isard, M., and Zisserman, A. (2007). Total recall: Automatic query expansion with a generative feature model for object retrieval. In *Proc. ICCV*.

[Duda et al., 1995] Duda, R., Hart, P., and Stork, D. (1995). *Pattern Classification and Scene Analysis 2nd ed.*

[Ferrari et al., 2004] Ferrari, V., Tuytelaars, T., and Van Gool, L. (2004). Simultaneous object recognition and segmentation by image exploration. In *Proc. ECCV*.

[Flickr,] Flickr. http://www.flickr.com/.

[Fraundorfer et al., 2007] Fraundorfer, F., Stewénius, H., and Nistér, D. (2007). A binning scheme for fast hard drive based image search. In *Proc. CVPR*.

[Godsil and Royle, 2001] Godsil, C. and Royle, G. (2001). *Algebraic Graph Theory*. Springer.

[Google Street View,] Google Street View. http://books.google.com/help/maps/streetview/.

[Hua et al., 2007] Hua, G., Brown, M., and Winder, S. (2007). Discriminant embedding for local image descriptors. In *Proc. ICCV*.

[Jegou et al., 2008] Jegou, H., Douze, M., and Schmid, C. (2008). Hamming embedding and weak geometric consistency for large scale image search. In *Proc. ECCV*.

[Jégou et al., 2009] Jégou, H., Douze, M., and Schmid, C. (2009). On the burstiness of visual elements. In *Proc. CVPR*.

[Jégou et al., 2010] Jégou, H., Douze, M., and Schmid, C. (2010). Improving bag-of-features for large scale image search. *IJCV*, 87(3):316–336.

[Li et al., 2008] Li, X., Wu, C., Zach, C., Lazebnik, S., and Frahm, J.-M. (2008). Modeling and recognition of landmark image collections using iconic scene graphs. In *Proc. ECCV*.

[Lowe, 2004] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110.

[Makadia, 2010] Makadia, A. (2010). Feature tracking for wide-baseline image retrieval. pages 310–323. Springer.

[Mikolajczyk and Matas, 2007] Mikolajczyk, K. and Matas, J. (2007). Improving sift for fast tree matching by optimal linear projection. In *Proc. ICCV*.

[Mikulik et al., 2010] Mikulik, A., Perdoch, M., Chum, O., and Matas, J. (2010). Learning a fine vocabulary. In Daniilidis, K., Maragos, P., and Paragios, N., editors, *Proc. ECCV*, volume 6313 of *Lecture Notes in Computer Science*, pages 1–14, Heidelberg, Germany. Foundation for Research and Technology-Hellas (FORTH), Springer. CD-ROM.

[Muja and Lowe, 2009] Muja, M. and Lowe, D. G. (2009). Fast approximate nearest neighbors with automatic algorithm configuration. In *VISSAPP*.

[Nister and Stewenius, 2006] Nister, D. and Stewenius, H. (2006). Scalable recognition with a vocabulary tree. In *Proc. CVPR*.

[Oliva and Torralba, 2006] Oliva, A. and Torralba, A. (2006). Building the gist of a scene: The role of global image features in recognition. *Visual Perception, Progress in Brain Research*, 155.

[Panoramio,] Panoramio. http://www.panoramio.com/.

[Perdoch et al., 2009] Perdoch, M., Chum, O., and Matas, J. (2009). Efficient representation of local geometry for large scale object retrieval. In *Proc. CVPR*.

[Perronnin, 2008] Perronnin, F. (2008). Universal and adapted vocabularies for generic visual categorization. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(7):1243–1256.

[Philbin et al., 2007] Philbin, J., Chum, O., Isard, M., Sivic, J., and Zisserman, A. (2007). Object retrieval with large vocabularies and fast spatial matching. In *Proc. CVPR*.

[Philbin et al., 2008] Philbin, J., Chum, O., Isard, M., Sivic, J., and Zisserman, A. (2008). Lost in quantization: Improving particular object retrieval in large scale image databases. In *Proc. CVPR*.

[Project page, 2012] Project page (2012). Data, binaries, and source codes released with the paper. http://cmp.felk.cvut.cz/ qqmikula/publications/ijcv2012/index.html.

[Sivic and Zisserman, 2003] Sivic, J. and Zisserman, A. (2003). Video google: A text retrieval approach to object matching in videos. In *Proc. ICCV*, pages 1470 – 1477.

[Tang et al., 2011] Tang, W., Cai, R., Li, Z., and Zhang, L. (2011). Contextual synonym dictionary for visual object retrieval. In *Proceedings of the 19th ACM international conference on Multimedia*, pages 503–512. ACM.

[Tavenard and Amsaleg, 2010] Tavenard, R. and Amsaleg, L.and Jégou, H. (2010). Balancing clusters to reduce response time variability in large scale image search. Research Report RR-7387, INRIA.