CENTER FOR
MACHINE PERCEPTION

CZECH TECHNICAL
UNIVERSITY

RESEARCH REPORT

ISSN 1213-2365

Large Scale Content Based Image Search

PhD Thesis Proposal

Andrej Mikulik

mikulik@cmp.felk.cvut.cz

CTU–CMP–2012–05

February, 2012

Available at
ftp://cmp.felk.cvut.cz/cmp/articles/mikulik-proposal.pdf

Supervisor: Prof. Ing. Jiří Matas Dr.
Co-supervisor: Mgr. Ondřej Chum Ph.D.

Author was supported by Microsoft Research Cambridge Scholarship.

Research Reports of CMP, Czech Technical University in Prague, No. 5, 2012

Published by

Center for Machine Perception, Department of Cybernetics
Faculty of Electrical Engineering, Czech Technical University
Technická 2, 166 27 Prague 6, Czech Republic
fax +420 2 2435 7385, phone +420 2 2435 7637, www: http://cmp.felk.cvut.cz

Abstract

In this report we summarize recent progress in large image retrieval, investigate ways to improve its robustness to viewpoint and environmental conditions change, increase mean average precision and increase recall for difficult or small objects.

Most effective particular object and image retrieval approaches are based on the bag-of-words (BoW) model. We present a novel similarity measure for this model. 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.

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 dataset/protocol.

We study 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 contrast with previously published methods. We further demonstrate that the large vocabularies increase the speed of the tf-idf scoring step.

All state-of-the-art retrieval results have been achieved by methods that include a query expansion that brings a significant boost in performance.

We introduce three modifications to automatic query expansion: (i) a method capable of preventing *query expansion failure* caused by the presence of confusers, (ii) an improved spatial verification and re-ranking step that incrementally builds a statistical model of the query object and (iii) we learn relevant spatial context to boost retrieval performance.

The three improvements of query expansion were evaluated on established Paris and Oxford datasets according to a standard protocol, and state-of-the-art results were achieved.

Contents

Chapter 1

Introduction

Large scale image search – recognition of objects in the images – is a challenging problem in computer vision. Recently, large collections of images become readily available due to commercial effords [Strww, Sanww] and photo sharing of individual people [Panww, Fliww].

Object recognition methods and image retrieval suitable for large datasets must not only have sublinear running time that grows slowly with the size of the collection, but must be very efficient in the memory usage. As soon as the representation of the complete collection fails to fit into operation memory, running time jumps by orders of magnitude (15-35s per query) as reported in [CPS+07].

1.1 Large Scale Image Search – Problem Definition

In this work, we are interested in the problem of specific object retrieval from a set of images. In other words, given a query image in which a particular object has been selected, the system should find and return ranked list of similar and representative images from its database in which a particular object appears 1.1. This is a more difficulte problem than whole-image retrieval, since the image of the query object may be taken under different conditions, changes in viewpoint, light or even partial occlusion.

Large scale, in this case, means databases with about 10^7 images. Meantime it is two or three order of magnitude less than in web-scale databases or collections stored on social networks, our retrieval system is running in real time on a single conventional machine.

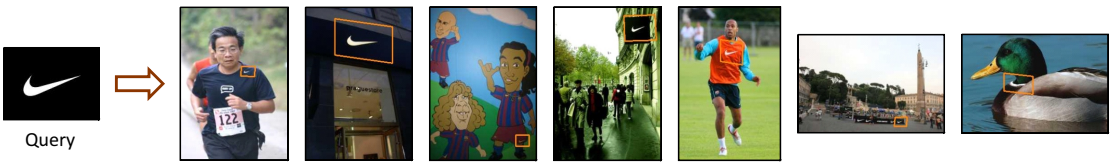


Figure 1.1: An example of a query and ranked results from the image retrieval system.

1.2 Large Scale Image Search Systems – the Baseline

The process of image search using bag-of-words techniques consist of two parts which can be roughly divided into several stages common to most of the search systems.

Preparation of the system – building the database index of given images for the fast image search – runs off-line (Fig. 1.2):

- **Feature detection and description**, the process of reduction of image data that allows to speed-up the search by keeping only relevant and well discriminable information from image
- **Vocabulary learning** is dividing the descriptor space into clusters – visual words, which are later used instead of descriptors for building index file. Using only the ID (word) in the index file instead of the descriptor allows to index large collections of images and search among them in real time.
- **Word assignment** is the process of assigning one or more words to descriptors of all images in the database.
- **Building the index**, the inverted file is basically storing of a list of documents for every visual word according to its appearence.

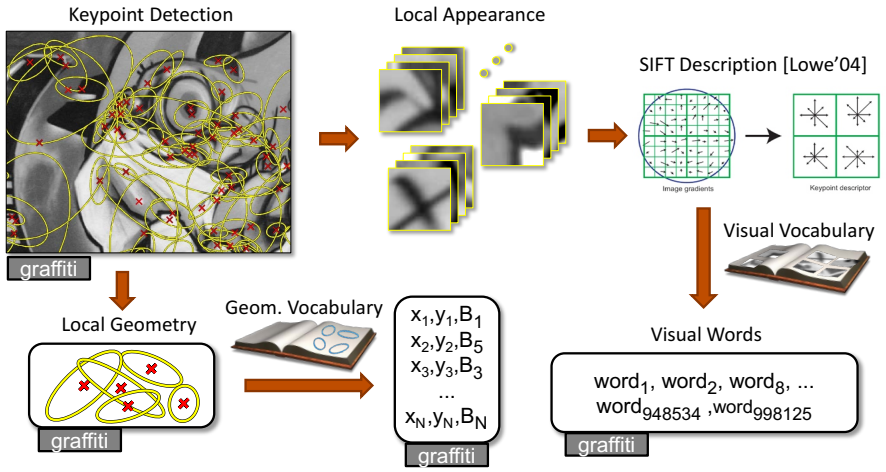


Figure 1.2: Image indexing during the offline stage. Courtesy of Michal Perdoch.

Querying the system with given image – the on-line phase (Fig. 1.3):

- **Feature detection and description** of the query image.
- **Word assignment** to the descriptor of given image using the vocabulary learned during the off-line phase.
- **Index look-up** is creating a list of images from database containing same words with the query image. Ranking the images according to number of common words using a word weighting.

- **Spatial verification** of the given number images with the highest rank (short list) by looking for geometrical transformation between retrieved image and the query.
- **Query expansion** is re-using the words from retrieved images with the highest score to create a new query intended to recall more results from the database.

1.3 Outline of the Work

The outline of this document is following. In the next chapter, we discuss recent approaches in field of large scale image search. Different state-if-the-art methods used for vocabulary learning and word assignment, as well as details and different ways of query expansion – the last stage of image search 2.

A novel method for vocabulary learning together with achieved results is proposed in chapter 3 and two extensions for query expansion are proposed in 4. In the last chapter we conclude our work and discuss further plans.

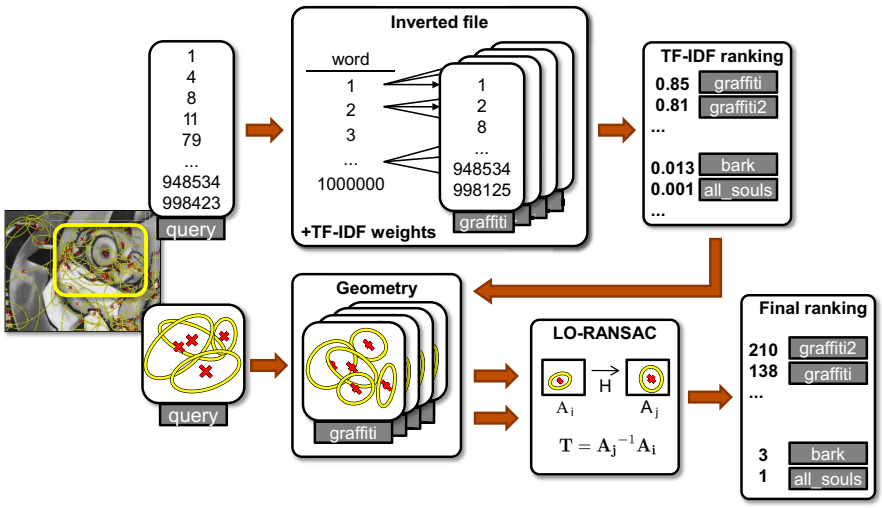


Figure 1.3: The process of querying the search system with a new image. Courtesy of Michal Perdoch.

Chapter 2

State of the Art

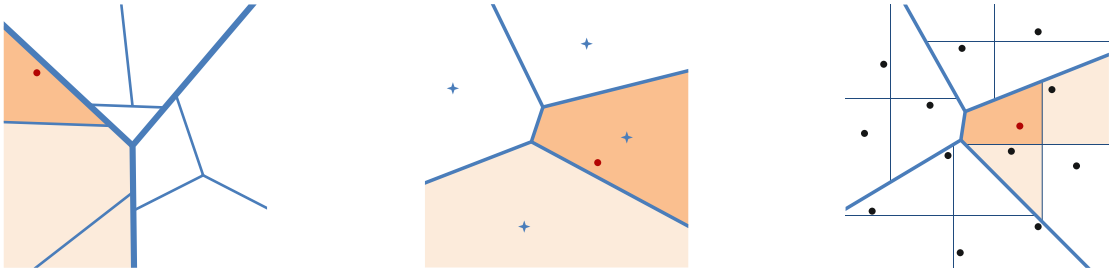
Virtually all aspects of particular object BoW-type retrieval have been intensively studied: feature detectors and descriptors [Low04, BTVG06, WHB09, MS04, MTS+05], vocabulary construction [SZ03, NS06, PCI+07, JDS08], spatial verification and re-ranking [PCI+07, JDS08], document metric learning [JHS07, JDS09, CM10b] and dimensionality reduction [JDSP10, PLSP10].

This chapter presents recent approaches to three subproblems of content based image search. First, vocabulary learning, which is one of the main and also one of the most time consuming part of the preparation of the system running off-line, is reviewed in detail. Next, the spatial verification and query expansion is described. These are the parts of the search system, which we are improving in next chapters.

2.1 Vocabulary Learning

Most if not all recent state-of-the-art methods build on [SZ03] who represented the image by a histogram of 'visual words', i.e. discretized SIFT descriptors [Low04]. The bag-of-words representation possesses many desirable properties required in large scale retrieval. If represented 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. In this work we will only consider and compare methods that support queries that cover only a (small) part of the test image. Global methods like GIST [OT06] achieve a much smaller memory footprint at the cost of allowing whole image queries only.

The discretization of the 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–∞ metric - features represented by the same visual word are deemed identical, else they 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. Recent methods like soft assignment and in particular the Hamming embedding aim at obtaining a better space-speed-accuracy trade off.



(a) (b) (c)

Figure 2.1: Different approaches to the soft assignment (saturation encodes the relevance): (a) hierarchical scoring [NS06] – the soft assignment is given by the hierarchical structure; (b) soft clustering [PCI+08] assigns features to r nearest cluster centers; (c) hamming embedding [JDS10] – each cell is divided into orthants by a number of hyperplanes, the distance of the orthants is measured by the number of separating hyperplanes.

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

In [SZ03], the first 'bag of words' approach to image retrieval was introduced. The vocabulary (the number of visual words $\approx 10^4$) was 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 [NS06] 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 [PCI+08] 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. 2.1(a).

Lost in quantization

In [PCI+08], 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. 2.1(b).

Hamming embedding

Jégou et al. [JDS10] have 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 independent hyper-planes into 2^n subcells. Each subcell is described by a binary vector of length n. Results reported in [JDS10] suggest that the hamming embedding provides good quantization. The good results are traded off with 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 approximately 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 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 signature.

While the reported bits per feature required in the search index ranges from 11 bits [PCM09] to 18 bits [PCI+08], 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.

2.2 Spatial Verification

As shown in [PCI+07, PCM09], the results can be significantly improved using the feature layout to verify the consistency of the retrieved images with the query region. The initially returned result list is re-ranked by estimating an affine transformation between the query image and result image. However, the spatial verification is significantly more time consuming than the BoW scoring, and is performed only on a shortlist of top scoring images. The shortlist is subsequently re-ranked based on the number of spatially verified inliers.

2.3 Query Expansion

In the text retrieval literature, one of the standard methods for improving performance is query expansion. A number of the highly ranked documents from the original query are re-issued as a new query. In this way, additional relevant terms can be added to the query.

In [CPS+07], the authors brought query expansion into the visual domain. A strong spatial constraint between the query image and each result enables an accurate verification for each return, resulting in a suppression of false positives that typically ruin text-based query expansion. These verified images can be used to learn a latent feature model to enable controlled construction of expanded queries.

In [CPS+07], the authors proposed a number of query expansion strategies. All of them follow a similar pattern: images in a shortlist are spatially verified against the query features, images with sufficient numbers of matches (inliers) are back-projected by the estimated affine transformation into the query region, and, finally, a new query is issued. The differences in the proposed strategies are either in the number of repeated applications of the process, or in the method of feature selection.

The simplest well performing query expansion method is called average query expansion. A new query is constructed by averaging a number of document descriptors. This approach is the quickest from all the suggested strategies, and has been adopted in a number of publications [PCI+08, PCM09, JDS09]. We use the average query expansion as the baseline method.

Chapter 3

Learning a Fine Vocabulary

3.1 Motivation

All approaches to soft clustering mentioned in chapter 2 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 [HBW07, MM07] 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 [Low04] 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. For examples of corresponding patches where SIFT distance does not predict well the similarity see Figures 3.1, 3.5, and 3.6.

In this chapter, unsupervised learning on a large set of images is exploited to improve on the 0–∞ metric. First, an efficient clustering process with spatial verification establishes correspondences within a huge (>5M) image collection. Next, a fine-grained vocabulary is obtained by 2-level hierarchical approximate nearest neighbour. The automatically established correspondences are then used to define a similarity mea-

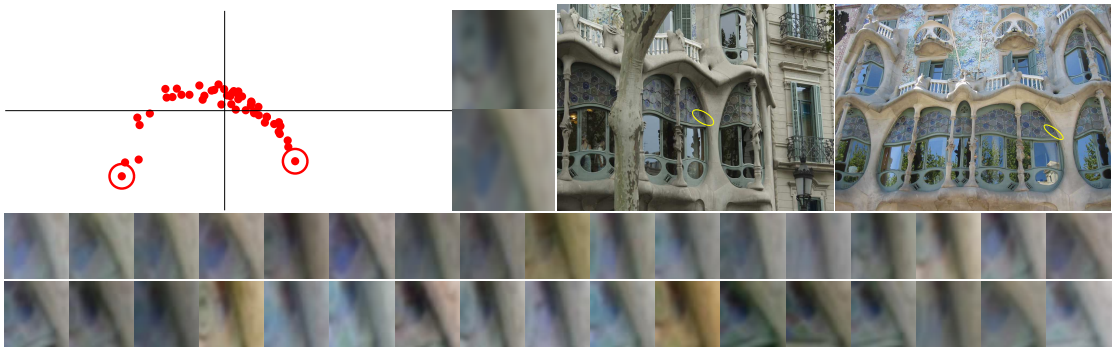


Figure 3.1: An example of corresponding patches. A 2D PCA projection of the SIFT descriptors (left); two most distant patches in the SIFT space and the images where they were detected (right); a set of sample patches (bottom). The average SIFT distance within the cluster is 278, the maximal distance is 591.

sure on the basis of a probabilistic relationships of visual words; we call it the *PR visual word similarity*.

When combined with a larger 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, i.e. it is more 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 PR similarity is faster than standard bag-of-words.

A novel similarity measure presented in this chapter is learned in an unsupervised manner, requires no extra space (only O(1)) in comparison with the bag-of-words and is more discriminative than both 0–∞ and L2-based soft assignment.

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 factor to speed up the process but not to bring the disadvantage of big imbalance factor of deeper ones.

3.2 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 structure, enumeration of descriptors in proximity is possible in time sub-linear in the size of the database. All bag-of-words based methods use partitioning $\{w_i\}$ of the descriptor space $\mathcal{D}$: $\bigcup w_i = \mathcal{D}, \quad w_i \cap w_{j \neq i} = \emptyset$. The partitions 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 quantized visual word 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 by the

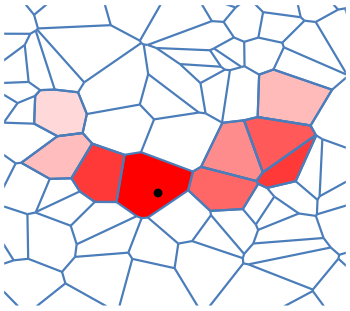


Figure 3.2: The set of alternative words in the proposed PR similarity measure.

imaging and detector noise. In our approach, 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 of this, we propose to exploit the matching probability between a feature observed in the query image and a database feature. Since our aim is to address retrieval in large scale databases where store requirements are critical, we focus attention on solution that requires no extra information per feature, or more exactly, that have a minimum overhead in comparison with the standard inverted file representation.

3.2.1 The Proposed Approach

We propose to use a fine partitioning of the descriptor space, so that the partitions only compensate for the noise (or even less). Even though the fine partitioning is learned in a data dependent fashion (as in the other approaches), the fine partitioning unavoidable separates matching features into a number of clusters.

For each partition (visual word) we learn which other partitions (called *alternative visual words*) that are likely to contain descriptors of matching features (Fig 3.2). This step is based on the probability of observing visual word w_j in a matching database image when visual word w_q was observed in the query image

$$P(w_j|w_q). \tag{3.1}$$

The probability (eqn. 3.1) is estimated by 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 (partitioning) the sets of matching features are converted to sets of matching visual words. We estimate the probability $P(w_j|w_q)$ from the feature tracks as

$$P(w_j|w_q) \approx \sum_{\mathcal{Z}} P(w_j|z_i)P(z_i|w_q). \tag{3.2}$$

For each visual word w_q, a fixed number of alternative visual words that have the highest conditional probability (eqn. 3.2) is recorded.

3.2.2 Learning Stage

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 links 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 [ASS$^+$09], where the feature tracks are used to produce 3D reconstruction. In our case, it is important to find a larger variety of patch appearances than precise point locations. Therefore, we adopt a slightly different approach to the choice of image pairs investigated.

Image Clusters

The algorithm starts 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 [CM10a, LWZ$^+$08]. 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. 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.

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 [Low04]) has to be used. In principle, one can go back even to the pixel level [FTVG04, CMP08], 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 [GR01] is constructed. 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 [ASS+09]. 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 [ASS+09]) kept as they provide further examples of patch appearance.

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

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. $S :=$ a random subset of $\{2, \ldots, \lfloor \frac{N-1}{2} \rfloor\}$, $|S| = K - 1$
3. $V := \{v_0, \ldots, v_{N-1}\}$
4. $E := \{(v_i, v_j) \,|\, v_i, v_j \in V, j = (i+1) \bmod N\}$
5. **for** $s \in S$
6. $\quad E := E \cup \{(v_i, v_j) \,|\, v_i, v_j \in V, j = (i+s) \bmod N\}$
7. **end**

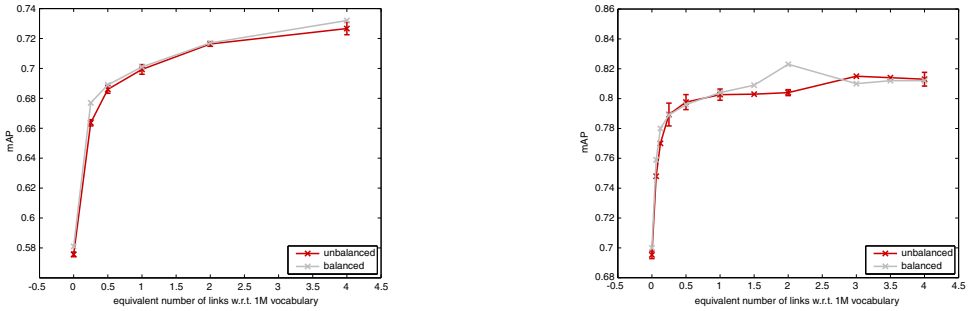
Large Vocabulary Generation

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, approximate nearest neighbour, FLANN [ML09], 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 10,713 million SIFTs (from almost 6 million images) is performed. In the first pass, each SIFT descriptor is assigned to the level one vocabulary. For each level one visual word 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.

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.



(a) (b)

Figure 3.3: Comparison of mAP for unbalanced and balanced 16M vocabulary with (a) and without (b) query expansion. Experiment shows that the balancing does not significantly affect mean average precision (the advantage is the gain in query speed). Error bars are shown where three vocabularies with different random initialization were evaluated.

We compared unbalanced and balanced vocabulary constructions. In the balanced construction, the second level of the vocabulary uses adaptive branching factor, which is proportional to the weight of the branch (Figure 3.3). 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 [TAJ10]. 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 was used. With such a construction an imbalance measure [JDS10] of 1.09 for the training image set (>5M images) (compared to 1.21 in [JDS10]) and 1.26 for the testing set – Oxford 105k. Fraundorfer et al. [FSN07] report estimate of imbalance factor 5 for hierarchical trees introduced in [NS06].

Size of the Vocabulary

There are different opinions how many visual words should have the vocabulary for image retrieval. Philbin et. al. in [PCI$^+$07] achieved the best mAP for object recognition with a vocabulary of 1M visual words and predict a performance drop for larger vocabularies. We contribute the result in [PCI$^+$07] to too small training dataset (16.7M descriptors). In our case the vocabularies with up to 64M words were build using 11G training descriptors. Experiment shows that the larger vocabulary, the better performance, even for plain bag-of-words retrieval. Introducing the alternative words, the situation is changed even more rapidly and according to expectations links are more useful for larger vocabularies (Figure 3.4). We have not built vocabularies larger than 64M because the memory footprint of the assignment tree started to be impractical.

Computing the Conditional Probability

To compute the conditional probability (eqn. 3.2) 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

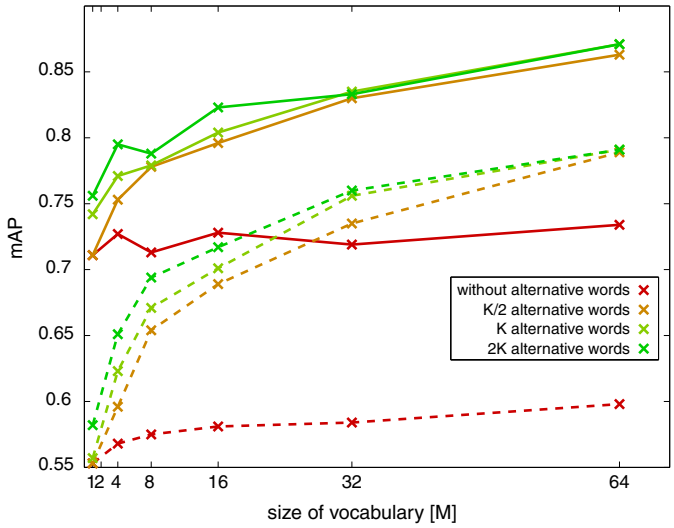


Figure 3.4: Comparison of mAP for the balanced vocabularies of from 1 to 64 millions visual words. Solid lines show results after the query expansion (QE), dashed lines without QE. Red lines show results using plain bag-of words (alternative words). The number of alternative words is proportional to vocabulary size to compare results of equal time complexity. In 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). K is the size of the vocabulary $/2^20$.

method	**imbalance measure level 1**		**imbalance measure 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 3.1: Comparison of imbalance factor [JDS10] 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.

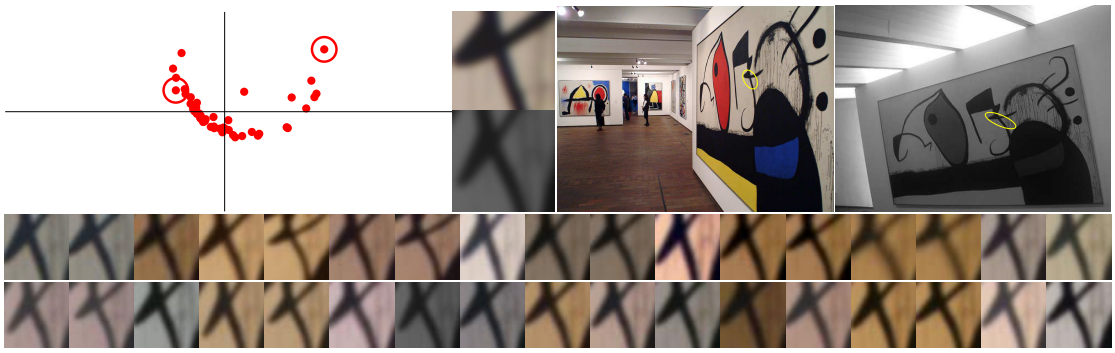


Figure 3.5: A 2D PCA projection of a feature track of SIFT descriptors (left); the most distant patches and their images (right); sample of feature patches from the track. The distance of the most distant SIFT descriptors is 542 and is caused by an enormous change in the viewpoint.

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. 3.2) is evaluated by traversing the relevant inverted file.

Statistics

Over 5 million images were clustered into almost 20 thousand clusters of 750 thousand images. Out of those 733 thousand 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 3.7 and 3.8.

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 ([CM10a] 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. (3.2) 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.

3.2.3 Retrieval Stage

The implementation of the retrieval stage is fairly standard, using inverted files [SZ03] for candidate image selection which is followed by fast spatial verification and query expansion [CPS$^+$07]. The modifications listed below are the major differences implemented in our retrieval stage.

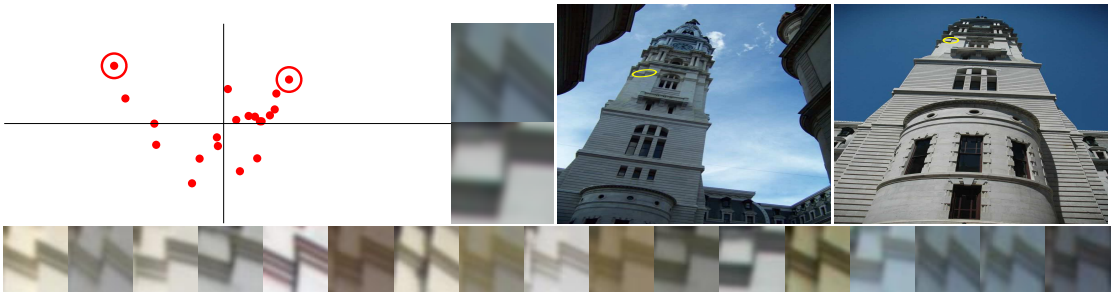


Figure 3.6: A 2D PCA projection of a feature track of SIFT descriptors (left); the most distant patches and their images (right); sample of feature patches from the track. The distance of the most distant SIFT descriptors is 593 and is caused by the viewpoint and scale change.

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%.

Weights of Alternative Words

Contribution of each visual word is weighted by the *idf* weight [BYRN99]. A number of re-weighting schemes for alternative words have been tried, none of them affecting significantly the results of the retrieval.

3.3 Experiments

We have extensively evaluated the performance of the PR similarity on a standard retrieval datasets Oxford 105K and Oxford 5K[1], INRIA Holidays[2] and PARIS[3]. 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.

3.3.1 Retrieval Quality

We follow the protocols of testing datasets defined in [PCI+07] 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.

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

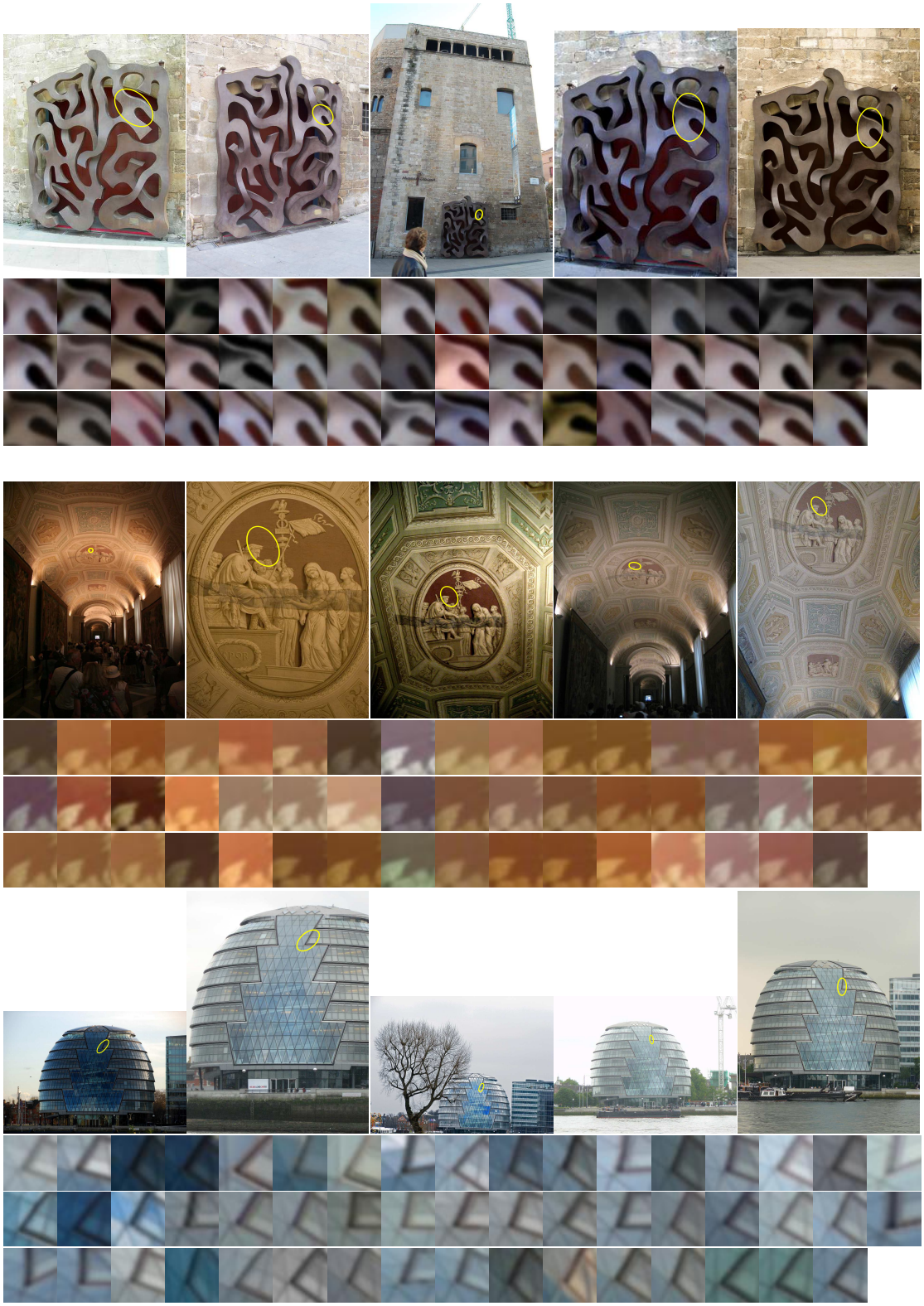


Figure 3.7: Three examples of feature tracks of size 50. Five selected images and all 50 patches of the track. Even though the patches are similar, the SIFT distance of some pairs is over 500.

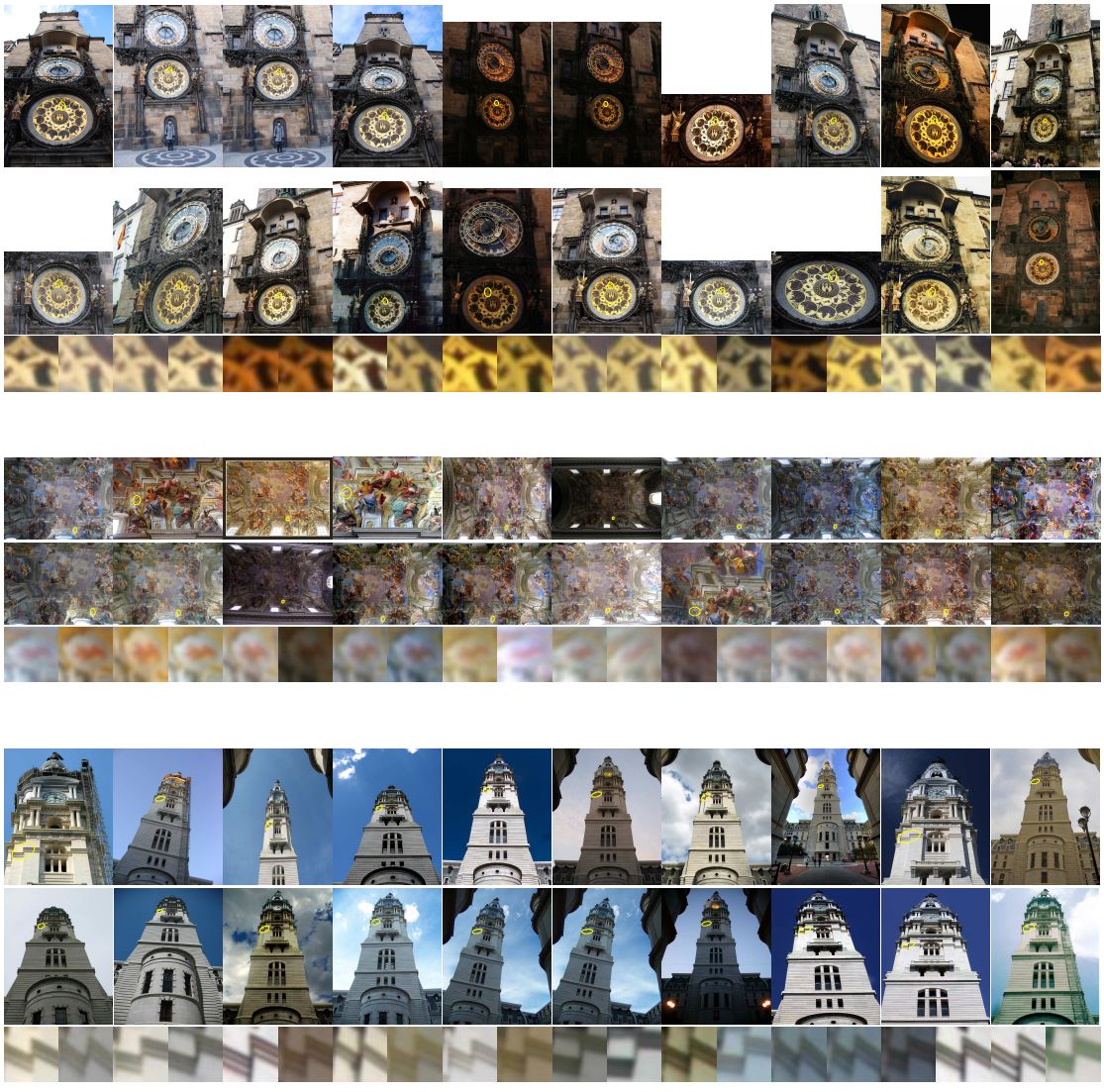


Figure 3.8: Three examples of feature tracks of size 20. Images and corresponding patches, note the variation in appearance.

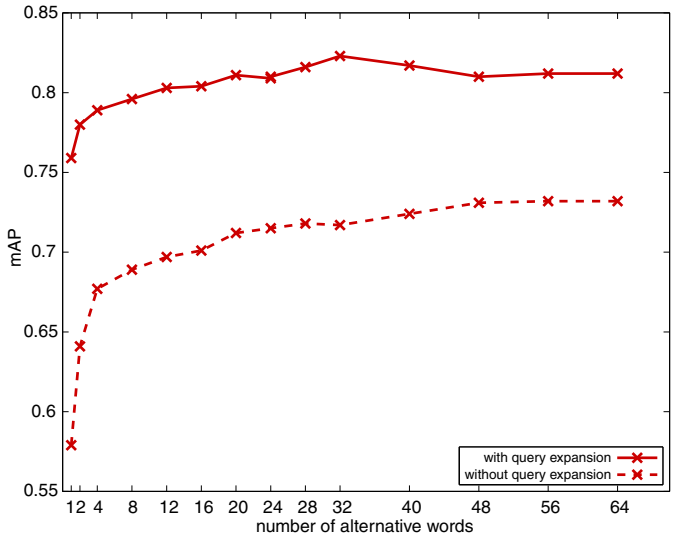


Figure 3.9: The quality of the retrieval, expressed as mean average precision (mAP), increases with the number of alternative words. The mAP after (upper curve) and before (lower curve) query expansion is shown.

In the first experiment, the quality of the retrieval as a function of the number of alternative words and vocabulary size was measured, see Figure 3.9. 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.

The second experiment studies the effects of the vocabulary size, and compares the alternative words in the PR similarity with the euclidean nearest neighbours in soft assignment. The left-hand part of Table 3.2 shows results obtained with the 16M vocabulary with three different settings 'std' – standard tf-idf retrieval with hard assignment of visual words; '5L' and '16L' – retrieval using alternative words (4 and 15 respectively). The righthand part presents results of reference state-of-the-art results [PCM09] obtain with a vocabulary of 1M visual words learned on the PARIS dataset. Two version of the reference algorithm are tested, without ("std") and with the query soft assignment to 3 nearest neighbours ("SA 3NN").

The experiments supports the following observations:

(i) 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.

(iv) The PR similarity outperforms the Hamming embedding approach combined with query expansion, Jegou et al. [JDS09, JDS10] 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.

	16M std	16M L5	16M L16	PARIS 1M std	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 3.2: Mean average precision for selected vocabularies on the Oxford 105k dataset.

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

Table 3.3: Average execution time per query in sec.

(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.

3.3.2 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 3.3.

The plot showing dependency of the query time on the number of alternative words is depicted in Figure 3.10. The times 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. 3.10 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) was evaluated depend on the dictionary size. To measure directly the speed of traversing the inverted file, the query time without the spatial verification was measured. Results are shown in Figure 3.11.

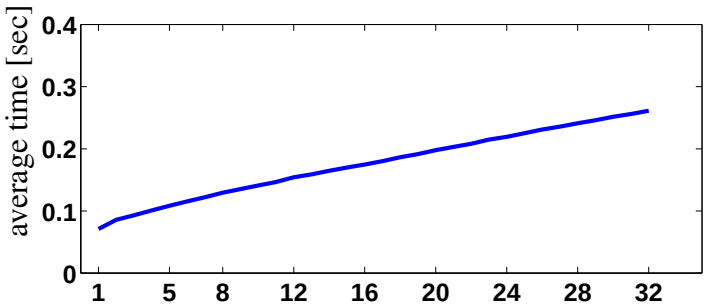


Figure 3.10: Dependence of the query time on the number of alternative words.

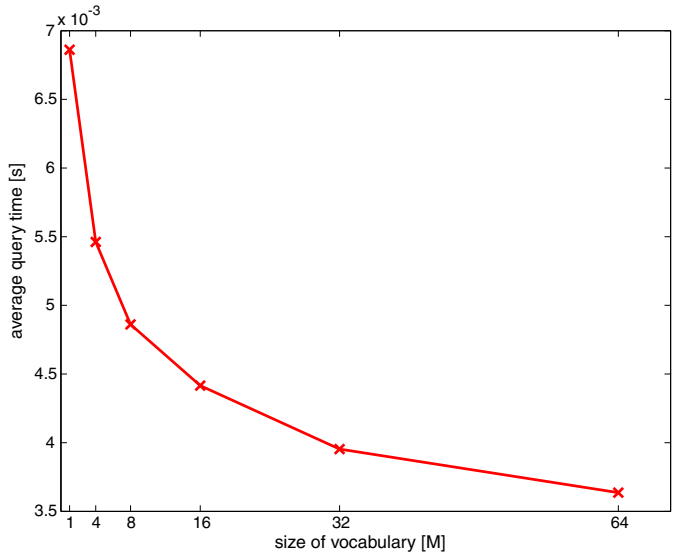


Figure 3.11: Dependence of the query time on the vocabulary size.

Dataset	16M std	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 3.4: Results of the proposed method on a number of publicly available datasets. (The result for the Oxford 105K is duplicated for completeness.)

3.3.3 Results on Other Datasets

The proposed approach has been tested on a number of standard datasets. These include Oxford, INRIA holidays (with manually corrected orientation of images, where the correct (sky-is-up) orientation is obvious), and Paris datasets. In all cases (Table 3.4), 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.

Chapter 4

Query Expansion with Context Growing

In this chapter, we focus on the query expansion (QE) step. Automatic query expansion [CPS+07] has been shown to bring a significant boost in performance [CPS+07, PCI+08, JDS09, PCM09], and all state-of-the-art retrieval results have been achieved by methods that include a QE step. Published QE methods focus on enriching the query model by adding spatially verified features. Retrieval with the "expanded" query follows. It has been observed that if the shortlist has enough true positives, the spatial verification re-ranking almost always correctly identifies relevant images, and, consequently, results for the expanded query are significantly better than the original single image query.

As a first contribution, we improve spatial verification and re-ranking by taking account of already evaluated results. The *incremental spatial re-ranking* (**iSP**) allows verification and subsequent use of images for query expansion that do not have a significant match against the original query, but do match a statistical model gradually built from the query and previously verified images.

As a second contribution, we propose a method that exploits spatial context by incorporating matching features outside the initial query boundary into the query expansion. Since the content outside the query region is not known at query time, the method requires efficient spatial verification of the retrieved images (Fig. 4.1).

In the next section, the novel incremental spatial verification is proposed. The query expansion with context growing is described in Section 4.3. Finally, the performance is evaluated in the last section of this chapter.

4.1 Improving Blind Relevance Feedback in QE

In QE, spatial verification and re-ranking plays the role of blind relevance feedback. Spatially consistent images retrieved with the original query are deemed "relevant", similarly to the images chosen by the user in manual relevance feedback. The selected parts of "relevant" images then contribute to the new, expanded query. The quality of the decision on relevance significantly influences the success of query expansion.

In this section, two improvements of spatial re-ranking are presented. First, we introduce *incremental spatial re-ranking* (**iSP**), where the verification accounts for not

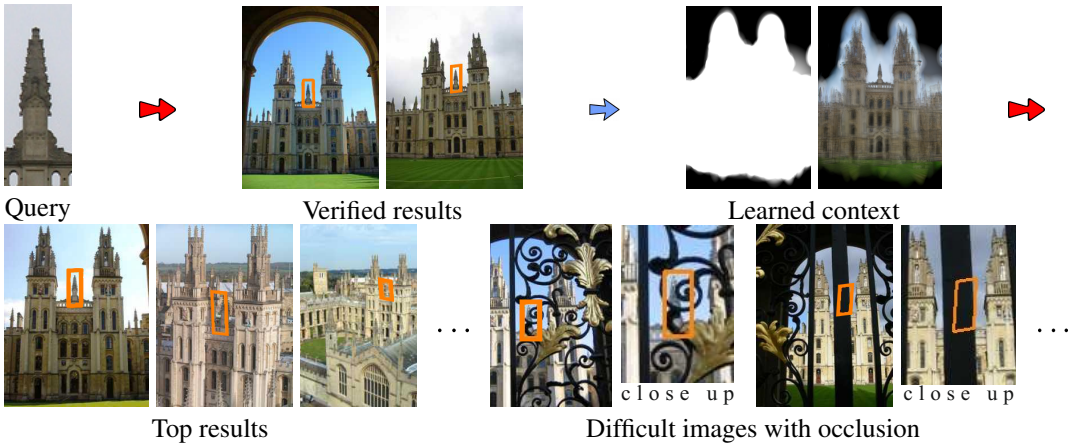


Figure 4.1: **Context expansion**: Consistent context is learned from retrieved images. The context enables successful retrieval (before to first false positive image) and localization of heavily occluded objects.

just spatial agreement with the initial query, but also agreement with all previously verified images. Second, we show that it is beneficial to "grow" the model of the object beyond the boundaries of the initial query, and to examine the spatially consistent neighbourhood of the query.

4.2 Incremental Spatial Re-ranking

In this section, an improvement of the spatial re-ranking (**SP**) phase of the baseline method (see Section 2.2) is proposed. As in the baseline method, the novel incremental spatial re-ranking (**iSP**) starts with the shortlist $\mathcal{S}$ of images ordered by the BoW score. The objective of **iSP** is to form a statistical model of the query object.

Initially, the statistical model M^0 includes only features from the query. Next, images in the shortlist are considered in the order given by BoW scoring. Each image $X \in \mathcal{S}$ is geometrically matched against the current model M^i. If the image matching quality $I_{M^i}(X)$ is greater than θ, the query object model is updated, and M^{i+1} is formed.

The quality function $I_{M^i}(.)$ is defined as the number of geometrically consistent features with the same visual word in image X and model M^i. The threshold θ was set to 15 after extensive preliminary experiments. The updated model M^{i+1} is the union of features in model M^i and features in image X, back-projected using function $f(.)$ onto the query image, clipped by the query bounding box. The final ranking of a shortlisted image is defined by the quality function. The method is described in Algorithm 2.

Since the simplest quality measure described above performed well, no alternatives, e.g. accounting for inlier ratio, geometric overlap, or weights of matching features, were evaluated.

Algorithm 2 Incremental spatial re-ranking

Input: query image X_q, shortlist $\mathcal{S}$ of images
Output: ranking $R : \mathcal{S} \leftrightarrow \{1..|\mathcal{S}|\}$, expanded model M^n of the object

$M^0 := X_q$
$Q := [], i := 0$
for $k := 1$ to $|\mathcal{S}|$ **do**
  $X := \mathcal{S}[k]$
  $Q[k] := I_{M^i}(X)$
  if $Q[k] > \theta$ **then**
    $M^{i+1} := M^i \cup f(X)$
    $i := i + 1$
  end if
end for
$R :=$ ranking of the images according to $Q[k]$.

4.3 Outside the Query Boundaries: Incorporating Context

The content outside the query region is not known at the query time. It is clear that learning the query context must be done by the "matching results to results" approach. The process of the *context learning* takes place either after spatial re-ranking, or, in the case of **iSP**, after each update of the query object model. The latter has the advantage that an image may be verified with the help of the context. In this case, implementation of context growing is trivial. As in **iSP**, features are back-projected to query image and are added to the model regardless of whether they are inside or outside the query bounding-box. The extension of the object model beyond the boundary of the original query only requires relaxing this constraint.

At the beginning of the learning phase, the context is identified with the area inside the query boundary. A feature added into the model that is not inside the context is inactive until confirmed by feature(s) from another image with the same visual word and similar geometry. Once a feature is confirmed, it adds the neighbourhood around its center to the context. All the confirmed features in the context are treated as active. The active features are considered the same as those inside the bounding box, and are used in spatial verifications, and, finally, in the query expansion. This is efficiently implemented by spatial binning. The process is summarized in Fig. 4.1.

The progress of context growth for two queries is visualized in Fig. 4.2. The learned model of the query is shown as the mean of elliptic patches associated with its features back-projected to the query. The query bounding box is drawn as an orange rectangle. To save space, the area not covered by the model, or equivalently, the area not covered by a single feature, is cropped. Experiment 2, summarized in Tab. 4.3, shows that including the context improves performance.

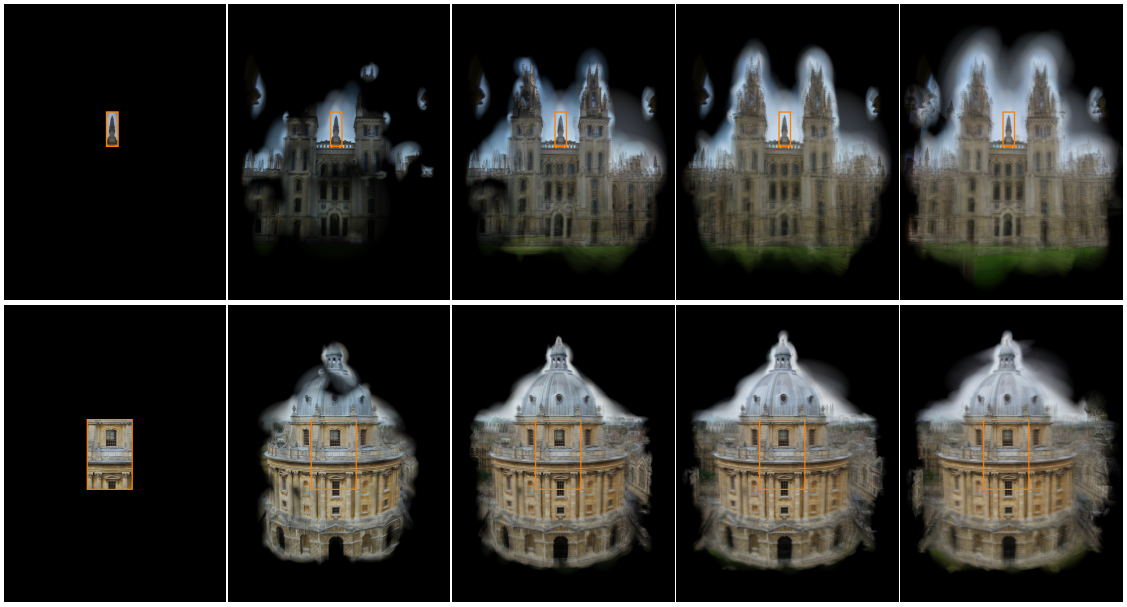


Figure 4.2: The process of context learning. Left column: the original query. Other columns: feature patches back-projected into the context from 2, 5, 10 and 20 spatially verified images.

Oxford 5k	5062 images of 11 Oxford landmarks
Oxford 105k	Oxford 5k + 100k Flickr distractors
Paris 6k	6414 images of 11 Paris landmarks
Paris 106k	Paris 6k + 100k Flickr distractors

Table 4.1: The datasets used in experiments.

4.4 Experiments

Datasets and Evaluated Methods

The image retrieval methods proposed in Sections 4.2 and 4.3 were evaluated according to the standard protocol [PCI$^+$07] on the Oxford Buildings [PCI$^+$07] and Paris datasets [PCI$^+$08]. Additionally, $\approx$100k confuser images of the 75 most popular Flickr tags are provided with the Oxford Buildings dataset [PCI$^+$07]. The datasets used in experiments are presented in Tab. 4.1. For each of the Oxford and Paris datasets, the evaluation protocol defines 55 queries, five for each landmark, with precise ground truth. The performance of all retrieval experiments is measured using the mean average precision (mAP), i.e. the area under the precision-recall curve for the further details see [PCI$^+$07].

The proposed incremental spatial re-ranking and context growing methods are compared with the state-of-the-art image retrieval approaches, see the list in Tab. 4.2. All methods use, in experiments on the Oxford dataset, a 1M visual word vocabulary trained on the Paris dataset and vice versa.

SP	BoW scoring, spatial re-ranking, no query expansion, see Sections 2.2 and 2.3
iSP	BoW scoring, incremental spatial re-ranking, no query expansion
SP + avg QE	BoW scoring, spatial re-ranking, average query expansion, see Sections 2.2 and 2.3
iSP + avg QE	BoW scoring, incremental spatial re-ranking, average query expansion
SP + ctx QE	BoW scoring, spatial re-ranking, context query expansion, see Section 4.3
iSP + ctx QE	BoW scoring, spatial re-ranking, incremental spatial re-ranking with context and context query expansion.

Table 4.2: Description of the state-of-the-art (rows 1 and 3) and the proposed methods (rows 2,4,5 and 6).

	I. w/o QE		**II. avg QE**		**III. ctx QE**	
	SP	**iSP**	**SP**	**iSP**	**SP**	**iSP**
Oxford 5k	0.616	0.741	0.785	0.825	0.781	0.827
Oxford 105k	0.553	0.649	0.725	0.761	0.731	0.767
Paris 6k	0.617	0.679	0.720	0.772	0.753	0.805
Paris 106k	0.508	0.556	0.627	0.687	0.653	0.710

Table 4.3: Comparison of image retrieval methods with standard (**SP**) and incremental spatial re-ranking (**iSP**).

Experiment 1. Evaluation of Incremental Spatial Re-ranking

The experiment compares all image retrieval methods listed in Tab. 4.2 on the Oxford and Paris datasets. We observe that **iSP** outperforms **SP** in all cases; compare the left and right columns of sections I, II and III of Tab. 4.3. The **iSP** improves performance by approximately one half of the query expansion effect; compare columns I right, and II left. Since only the shortlist is accessed, the performance improvement is obtained at a negligible cost compared to issuing a second query. This encourages the use of **iSP** instead of the standard **SP** re-ranking. Additionally, the benefits of **iSP** and query expansion are additive; compare columns I right and II right. Finally, adding context has negligible effect on the Oxford dataset and improves performance on the Paris dataset. This is due to the fact that on the Oxford protocol, queries include entire objects, and there is little gained by growing the context.

Experiment 2. Evaluation of Context Expansion

Next we study the influence of incorporating the context of the query i.e. extending the model of the query outside its bounding box. The behaviour is demonstrated on the same datasets by using a novel protocol.

As shown in experiment 1, the effect of context learning is not significant in the case

of the Oxford dataset. To model a situation where only a detailed or partial view of the object is available, the following protocol was devised: The query bounding boxes were symmetrically reduced to 10% of their area in nine steps, see Fig. 4.3. The maximum spatial extent of the context was limited to an area 25× larger than the reduced query bounding box.

The results (see Fig. 4.4) show that the performance of the retrieval method using both context and incremental spatial re-ranking (**iSP + ctx QE**) drops below the state-of-the-art (black dashed line in Fig. 4.4) method only after reducing the bounding box area to 40%, (Fig. 4.4b,d), or even to 20% (Fig. 4.4a,c) of the full query bounding box. One of the reasons for the drop in performance is that to keep the number of features in the model, and thus the speed of spatial re-ranking reasonable, we limit the number of images added to the model to ten, which is insufficient to reconstruct the model to the quality of the original query. Also, the results of initial queries on the standard datasets already contain many of true positives, and even the standard query expansion manages to retain a sufficient model of the object.

Some examples of contexts learned for some of the Oxford protocol queries are shown in Fig. 4.3.

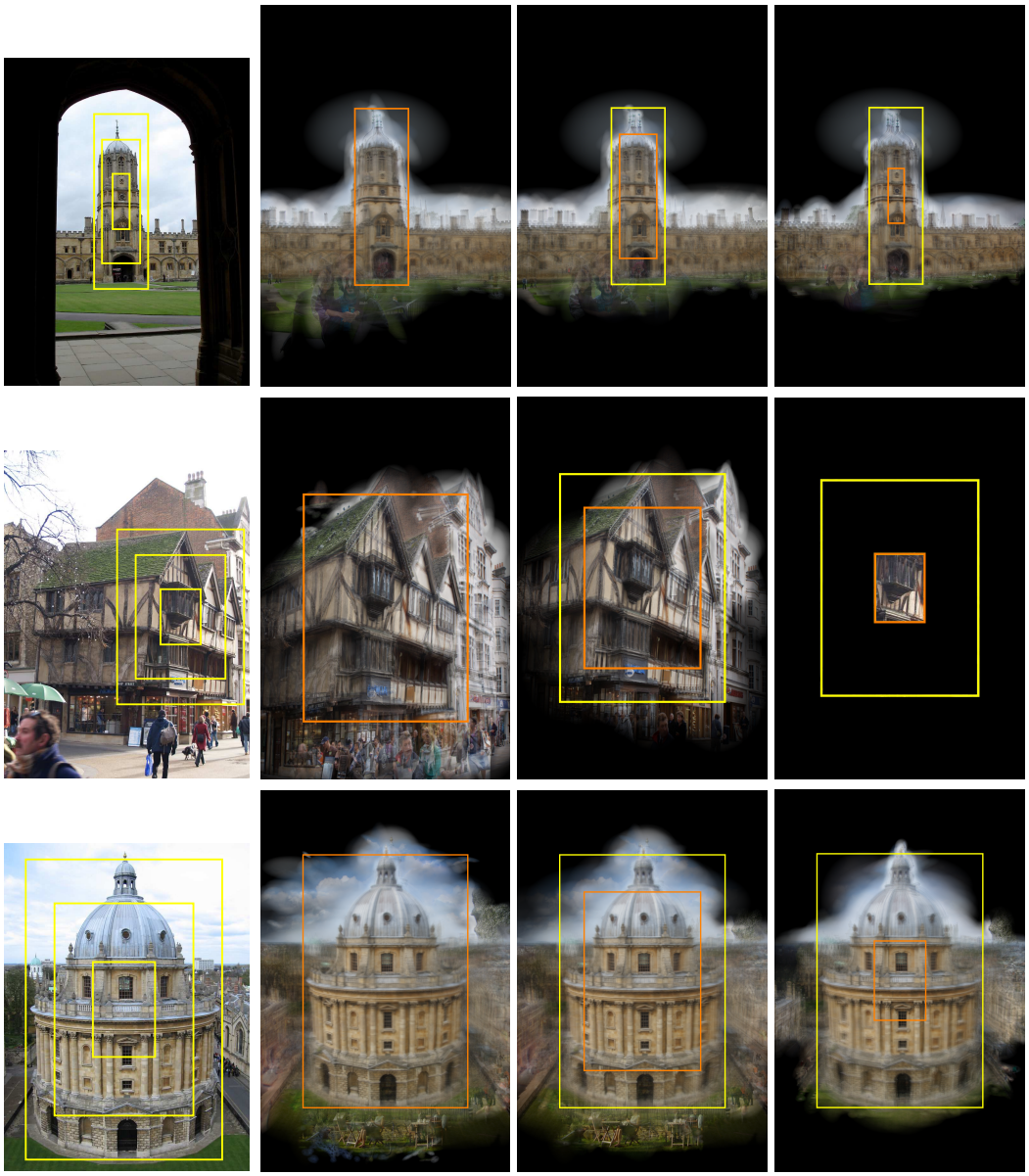
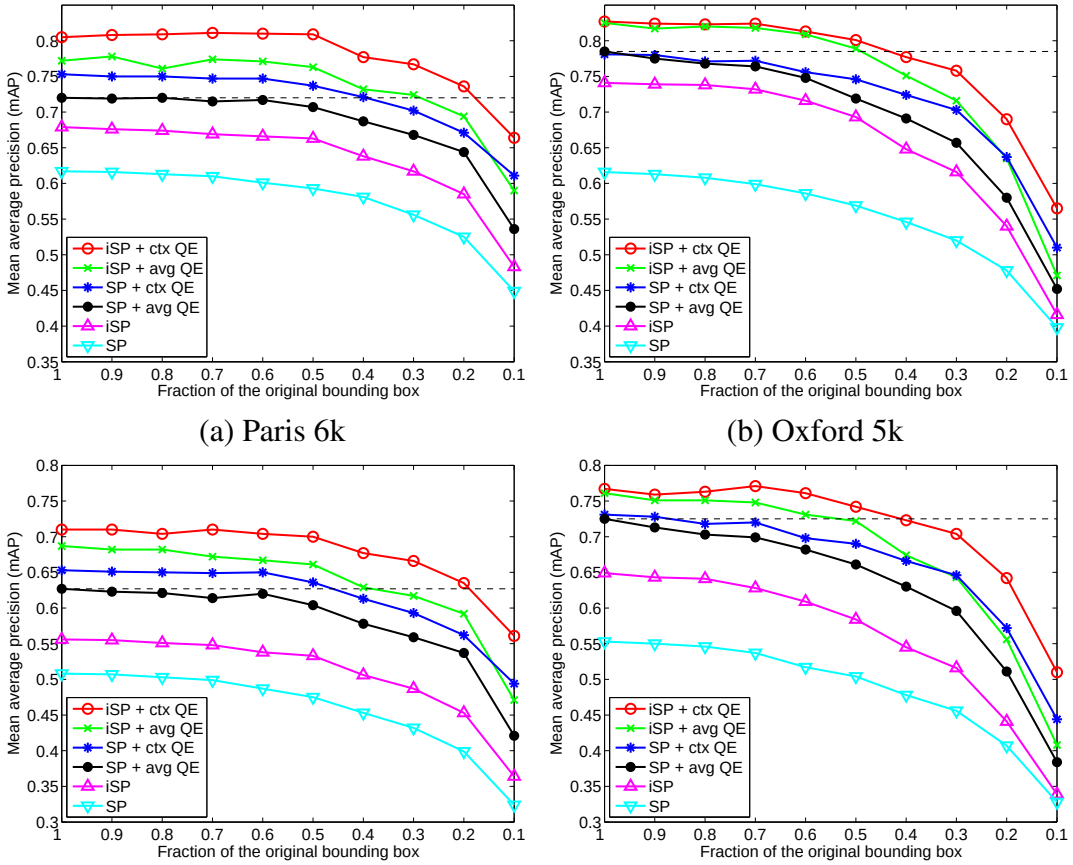


Figure 4.3: Left column: Examples of the full (100%) bounding box of some Oxford protocol queries (outer rectangle) and the query bounding boxes reduced to 50% and 10%. Columns 2, 3 and 4 depict the context learned from the full, 50% and 10% bounding boxes respectively (the orange rectangles). The yellow rectangle shows the original bounding box. Note the ability of the **iSP + ctx QE** to learn the context even from the smallest query. The method failed on the CORNMARKET 10% (right column, middle) due to the insufficient number of spatially verified images.



(a) Paris 6k (b) Oxford 5k

(c) Paris 106k (d) Oxford 105k

Figure 4.4: The influence of decreasing the query bounding box size on image retrieval methods. The black dashed line is the performance of the state-of-the-art [CPS+07] method with the original bounding box. The performance of the proposed **iSP + ctx QE** is superior to the state-of-the-art method, if the query covers more than 20% of the bounding box on the Paris datasets, and more than 40% of the bounding box on the Oxford datasets. The compared methods are listed in Tab. 4.2.

Chapter 5

Automatic Failure Recovery in Query Expansion

One of the key issue in an image retrieval system based on bag-of-words is the definition of image similarity. The most common approach is to define visual similarity as a normalized sum, over all visual words, using the *tf-idf* weightening scheme [SZ03]. The weight of the word increases proportionally to the number of times it appears in a document, but is offset by the frequency of the word in the corpus. This helps to handle the fact that not every visual word is equaly important – has the same discriminability. The *tf-idf* is commonly used no matter which type of vobabulary is used (Section 2.1).

In the image retrieval systems, the use of a similarity measure is typically justified by its probabilistic interpretation. Any similarity measure emplying summing over all visual words implicitly assumes that visual words occure independetnly on each other. This assumption is made because of computational convenience and it is intuitively obvious that it does not hold. Groups of correlated features typically occur on the water surface, on vegetation, images of text, faces, net-like structures, repetitive patterns, and statistical textures [CN08]. If such group is visible on the image query, and is not related to the object of interest, the BoW retrieval, without any special treatment, fails to select

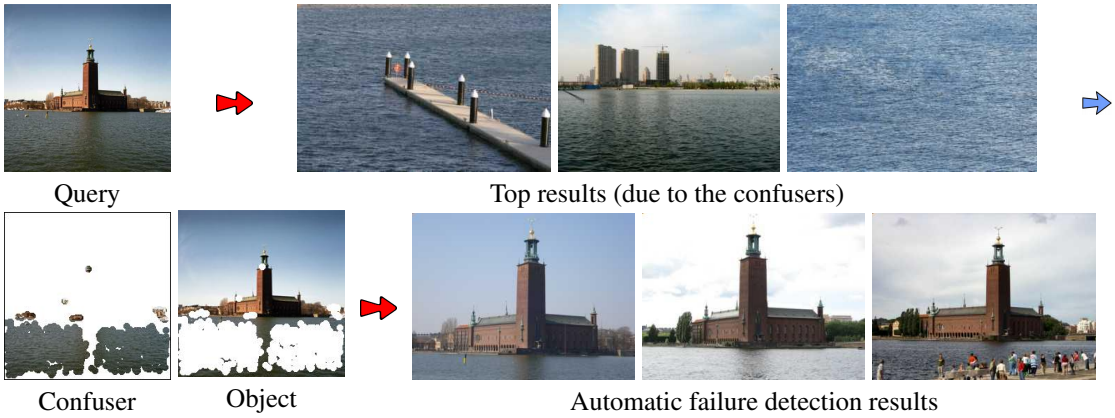


Figure 5.1: **Automatic Failure Recovery**: Initial retrieval results corrupted by confusing water features. The confuser model is learned dynamically. Successful subsequent query using the confuser model.

Figure 5.2: Examples of queries (leftmost images) where standard image retrieval fails to return images of the query object (upper rows of results). The results of the method [CM10b] with removed cooc-sets, which were discovered during the off-line stage. Courtesy of Ondrej Chum [CM10b].

relevant images into the shortlist. This is a consequence of correlated voting for images that contain the same type of 'confusers', which suppresses the relative contribution of the specific object. (see Fig. 5.2)

Moreover if BoW fails to the extent that there are no, or very few, correctly retrieved images in the shortlist, standard QE is no help. Such situations, which arise in the presence of structures with multiple correlated features, have been referenced in the literature as cooc-sets [CM10b] or confusers [KSP10].

In this chapter we show how to detect and recover from the *failing query expansion* situation. Unlike other approaches, the proposed method handles the presence of confusers in the query region on-the-fly, with no prior learning step required. We achieve performance that is comparable to the state-of-the-art without the need for off-line and potentially time-consuming processing that is difficult to execute in a continuously updated database.

The proposed method is orthogonal to previously published approaches, and can be used in conjunction with them. Previously, the methods of query expansion analysed results of the initial query to prepare a new query that is a *richer* representation of the object of interest. In contrast, the proposed approach tries to first obtain a *cleaner* model of the object by eliminating irrelevant confuser features.

5.1 Query Model

We model the query (visual) words as a mixture of words generated by three processes (topics): the object words $\mathcal{O}$, the confuser words $\mathcal{C}$, and the random words $\mathcal{R}$. The three types of words, and their properties, are described in the following paragraph.

We address the retrieval of *particular* objects, defined as a collection of features that preserves appearance and spatial layout over a range of imagining conditions such as viewpoint change, and scale change.

The object words $w \in W_{\mathcal{O}}$ are likely to be observed in images containing the object of interest, i.e. $P(w|\mathcal{O})$ is high, $P(w|\mathcal{O}) \gg P(w)$. Moreover, the features associated with words, $w \in W_{\mathcal{O}}$, appear at fixed coordinates with respect to the canonical frame of the object, and thus allow for the geometric consistency check. The confuser words $w \in W_{\mathcal{C}}$ are defined as sets of correlated words, satisfying $P(w|\mathcal{C}) \gg P(w)$. However, confuser words are not significantly spatially consistent[1]. Randomly occurring words, $w \in W_{\mathcal{R}}$, generated from spurious features, and corrupted descriptors form the most frequently occurring class. As reported in [TL09], object features cover as few as 4% of the total features.

Algorithm 3 Automatic failure recovery

Input: query features Q_0
Output: $\langle$ query features, query results, feature mask $\rangle$

Execute query Q_0 including spatial verification
if $\rho(Q_0) > \rho_0$ **then**
  return $\langle$ Q_0, results (Q_0), empty $\rangle$
end if
Learn a set of confuser words $W_{\mathcal{C}}$ (eqn. 5.1)
$Q_N = Q_0 \setminus W_{\mathcal{C}}$
Execute query Q_N including spatial verification
if $\rho(Q_0) > \rho(Q_N)$ **then**
  return $\langle$ return Q_0, results (Q_0), empty $\rangle$
else
  return $\langle$ return Q_N, results (Q_N), mask($W_{\mathcal{C}}$) $\rangle$
end if

5.2 Recovery

We propose a modification, called automatic failure recovery, to the retrieval scheme. First, standard BoW retrieval with spatial verification is performed. The BoW scoring is used to produce a shortlist of documents. The images in the shortlist are checked for spatial consistency with the query features. The shortlist is significantly shorter than the

[1] We do not aim to solve a philosophical question regarding whether recurring objects, such as phone booths, are objects or confusers. According to our model, appearance and spatially consistent features form objects.

size of the database[2]. If relevant images are included in the shortlist, these are identified by spatial re-ranking. Once relevant documents are retrieved, automatic query expansion techniques are used to improve the object model $\mathcal{O}$. When a significant number of confuser words $\mathcal{C}$ is present in the query, the whole shortlist can be populated by images containing features generated from $\mathcal{C}$, and hence the spatial re-ranking cannot improve the search results. We call this situation a *retrieval failure*. Even though the shortlist does not contain relevant images, it still conveys valuable information. A statistical model of the confusers $\mathcal{C}$ present in the query can be learned from the images in the shortlist, since a vast majority in the shortlist score higher than the relevant images. Once the confuser model $\mathcal{C}$ is known, its influence on the query is suppressed. There are three issues that need to be addressed: (i) efficiently estimate the confuser model $\mathcal{C}$, (ii) down-weight the effect of the confusers to the query result, and (iii) decide if the retrieval failure has arisen. The algorithm is summarized in algorithm 3

Ad (i) The distribution $P(w|\mathcal{S})$ of visual words in the shortlist $\mathcal{S}$ is learned at virtually no cost during the tentative correspondence construction in the spatial re-ranking phase. Features whose visual words appear significantly more frequently than in the database are deemed to be part to the confuser model $\mathcal{C}$:

$$W_{\mathcal{C}} = \{w|P(w|\mathcal{S})/P(w) > r_0\}\,. \tag{5.1}$$

The likelihood ratio threshold was $r_0 = 10$ in our experiments.

Ad (ii) There are many options to reduce the influence of the estimated confusers $\mathcal{C}$. We choose to simply remove the confuser features from the query. This approach, while seeming naive, has been shown to be effective and efficient [CM10b]. If a query expansion, average or any other type, is used after the failure recovery, features that back-project to regions occupied by the confusers are also removed. This prevents back-projected confuser features from entering the expanded query from the result images.

Ad (iii) To check whether a retrieval failure has arisen, we compare the estimated quality $\rho(Q)$ of results of two queries: the original query, Q_0, and the query after the recovery, Q_R. We estimate the quality of the results by the inlier ratios in the top matching results. First, the acceptable result images that each have an absolute and relative non-random number of inliers are selected. The score of the retrieval is then defined as the sum of inlier ratios over the acceptable results. Formally, let $\mathcal{S}_Q$ be a BoW shortlist of query Q, $T_Q(X)$ be a number of tentative correspondences between query Q and image X, and let $I_Q(X)$ be the number of geometrically consistent features between Q and X. The acceptable result of Q is a set of images

$$\mathcal{A}_Q = \left\{ X | X \in \mathcal{S}_Q \;\&\; I_Q(X) > I_0 \;\&\; \frac{I_Q(X)}{T_Q(X)} > \epsilon_0 \right\}.$$

The quality of the shortlist result of query Q is defined as

$$\rho(Q) = \sum_{X \in \mathcal{A}_Q} \frac{I_Q(X)}{T_Q(X)}. \tag{5.2}$$

[2]In our experiments, a shortlist of 1000 documents is used.

	AFR		Cooc [CM10b]		Baseline	
	AP	FP	AP	FP	AP	FP
Stockholm	**0.659**	**16**	0.569	15	0.032	1
Dragon Wall	**0.797**	**56**	0.726	52	0.065	5
St Ignazio	**0.945**	**17**	0.737	14	0.105	2
Colloseum	**0.762**	**514**	0.136	85	0.018	13
Barcelona	**0.895**	**17**	0.789	15	0.053	1
St Mary	**0.943**	**57**	0.895	51	0.020	1
Vaticane	**0.957**	**22**	0.870	20	0.130	3
Bridge	0.583	4	**0.716**	**5**	0.143	1

Table 5.1: Quantitative comparison of the proposed method with [CM10b] and the baseline method on the Q8 dataset: Estimated average precision AP and the rank of the first false positive FP.

To avoid wasted computation when improvement is unlikely, the estimated quality of the original query Q_0 is thresholded. If $\rho(Q_0) > \rho_0$, then the hypothesis of the query failure is directly rejected. In the experiments, the following parameters were used: minimal acceptable number of inliers $I_0 = 5$, minimal acceptable inlier ratio $\epsilon_0 = 0.2$, and the failure rejection threshold $\rho_0 = 5$.

5.3 Efficiency

The proposed method introduces no extra cost for queries that return reasonable number of matching results (this is the case for almost all images in the standard Oxford and Paris datasets, where the query bounding box is tightly around the query building). For such queries the result is also unaffected because the original query is accepted. For other queries, one extra BoW scoring and spatial re-ranking step is executed. Since the new query is a subset of the original query, this additional step is faster than the original query.

5.4 Experimental Results

In this section, we compare the results of the confuser model learned in the proposed automatic failure recovery step with results obtained by cooc-sets [CM10b]. The quantitative results on the Q8 dataset [CM10b] embedded in a database of over 5 million images are shown in Tab. 5.1. It is not feasible to obtain all true positives, so the average precision is only an upper bound estimate. New positive results have been discovered by the proposed method, and the AP values are not directly comparable to values in [CM10b].

Table 5.1 shows, that for most cases, the two methods give comparable results. For the 'Colloseum' query, the proposed approach gives significantly better results than results obtained by the cooc-sets approach. This is because the cooc-sets approach excludes object-relevant cooc-set features as well as the confuser features, as opposed

to the proposed approach which correctly learns the confuser model.

Pros and Cons

Pros: No pre-learning step is required, so the method is applicable to any dataset and any vocabulary, and additionally, it does not require good training sets that generalize well, or retraining for different vocabularies. The method is specific to the current database and to the current query, so features for some queries that are confusers can be useful for other queries. **Cons:** The proposed method requires the execution of the original query, while the cooc-set approach can filter confuser features beforehand. In some queries, the confuser features may represent significant proportion of the features and thus the full query takes longer to execute.

Chapter 6

Summary

Firstly, we presented a novel similarity measure. The similarity function is learned in an unsupervised manner using geometrically verified correspondences obtained with an efficient clustering method on a large image collection.

The similarity measure requires no extra space in comparison with the standard bag-of-words method. Experimentally we showed that the novel similarity function achieves mean average precision that is superior to any result published in the literature on the standard Oxford 105k dataset/protocol. 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 built a larger vocabulary, which performs better and faster, and proposes the simple balancing method, which helps to keep imbalance factor low.

Next, we proposed two modification of the query expansion step in the BoW-based particular object and image retrieval. The spatial verification and re-ranking step was improved by incrementally building a statistical model of the query object. Subsequently, we showed that relevant spatial context significantly improves retrieval performance.

The proposed improvements of query expansion were evaluated on established Paris and Oxford datasets according to a standard protocol and state-of-the-art results were achieved.

Finaly a method capable of preventing *query expansion failure* caused by the presence of confusers was introduced. Unlike other approaches, the proposed method handles the presence of confusers in the query region on-the-fly, with no prior learning step required. We achieve performance that is comparable to the state-of-the-art without the need for off-line and potentially time-consuming processing that is difficult to execute in a continuously updated database.

Bibliography

[ASS+09] S. Agarwal, N. Snavely, I Simon, S. Seitz, and R. Szeliski. Building Rome in a day. In *Proc. ICCV*, 2009. 11, 12

[BTVG06] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. In *Proc. ECCV*, 2006. 4

[BYRN99] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. ACM Press, ISBN: 020139829, 1999. 16

[CM10a] O. Chum and J. Matas. Large-scale discovery of spatially related images. *IEEE PAMI*, 32:371–377, 2010. 11, 15

[CM10b] O. Chum and J. Matas. Unsupervised discovery of co-occurrence in sparse high dimensional data. In *Proc. CVPR*, 2010. 4, 31, 33, 34

[CMP08] J. Cech, J. Matas, and M. Perdoch. Efficient sequential correspondence selection by cosegmentation. In *Proc. CVPR*, 2008. 11

[CN08] Mark Cummins and Paul Newman. FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance. *The International Journal of Robotics Research*, 27(6):647–665, 2008. 30

[CPS+07] 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 *Proc. ICCV*, 2007. 1, 7, 15, 22, 29

[Fliww] Flickr. http://www.flickr.com/, www. 1

[FSN07] F. Fraundorfer, H. Stewénius, and D Nistér. A binning scheme for fast hard drive based image search. In *Proc. CVPR*, 2007. 13

[FTVG04] V. Ferrari, T. Tuytelaars, and L. Van Gool. Simultaneous object recognition and segmentation by image exploration. In *Proc. ECCV*, 2004. 11

[GR01] C. Godsil and G. Royle. *Algebraic Graph Theory*. Springer, 2001. 11

[HBW07] G. Hua, M. Brown, and S. Winder. Discriminant embedding for local image descriptors. In *Proc. ICCV*, 2007. 8

[JDS08] H. Jegou, M. Douze, and C. Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *Proc. ECCV*, 2008. 4

[JDS09] H. Jégou, M. Douze, and C. Schmid. On the burstiness of visual elements. In *Proc. CVPR*, 2009. 4, 7, 19, 22

[JDS10] Hervé Jégou, Matthijs Douze, and Cordelia Schmid. Improving bag-of-features for large scale image search. *IJCV*, 87(3):316–336, 2010. 5, 6, 13, 14, 19

[JDSP10] H. Jégou, M. Douze, C. Schmid, and P. Parez. Aggregating local descriptors into a compact image representation. In *Proc. CVPR*, 2010. 4

[JHS07] Herve Jegou, Hedi Harzallah, and Cordelia Schmid. A contextual dissimilarity measure for accurate and efficient image search. In *Proc. CVPR*, 2007. 4

[KSP10] J. Knopp, J. Sivic, and T. Pajdla. Avoding confusing features in place recognition. In *Proc. ECCV*, 2010. 31

[Low04] D.G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. 4, 8, 11

[LWZ+08] X. Li, C. Wu, C. Zach, S Lazebnik, and J.-M. Frahm. Modeling and recognition of landmark image collections using iconic scene graphs. In *Proc. ECCV*, 2008. 11

[ML09] Marius Muja and David G. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. In *VISSAPP*, 2009. 12

[MM07] Krystian Mikolajczyk and Jiří Matas. Improving sift for fast tree matching by optimal linear projection. In *Proc. ICCV*, 2007. 8

[MS04] K. Mikolajczyk and C. Schmid. Scale & affine invariant interest point detectors. *IJCV*, 1(60):63–86, 2004. 4

[MTS+05] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool. A comparison of affine region detectors. *IJCV*, 65(1/2):43–72, 2005. 4

[NS06] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *Proc. CVPR*, 2006. 4, 5, 13

[OT06] A. Oliva and A. Torralba. Building the gist of a scene: The role of global image features in recognition. *Visual Perception, Progress in Brain Research*, 155:23–36, 2006. 4

[Panww] Panoramio. http://www.panoramio.com/, www. 1

[PCI+07] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Proc. CVPR*, 2007. 4, 6, 13, 16, 25

[PCI+08] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Lost in quantization: Improving particular object retrieval in large scale image databases. In *Proc. CVPR*, 2008. 5, 6, 7, 22, 25

[PCM09] M. Perdoch, O. Chum, and J. Matas. Efficient representation of local geometry for large scale object retrieval. In *CVPR*, 2009. 6, 7, 19, 22

[PLSP10] F. Perronnin, Y. Liu, J. Sanchez, and H. Poirier. Large-scale image retrieval with compressed fisher vectors. In *Proc. CVPR*, 2010. 4

[Sanww] Nokia Sanfrancisco. http://www.nn4d.com/site/global/developer_resources/nokia_data_share/overview/p_overview.jsp, www. 1

[Strww] Google Streetview. http://books.google.com/help/maps/streetview/, www. 1

[SZ03] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *Proc. of ICCV*, pages 1470 – 1477, Oct 2003. 4, 5, 15, 30

[TAJ10] Romain Tavenard, Laurent Amsaleg, and Hervé Jégou. Balancing clusters to reduce response time variability in large scale image search. Research Report RR-7387, INRIA, 09 2010. 13

[TL09] P. Turcot and D. G. Lowe. Better matching with fewer features: The selection of useful features in large database recognition problems. In *ICCV Workshop LAVD*, 2009. 32

[WHB09] S. Winder, G. Hua, and M. Brown. Picking the best daisy. In *Proc. CVPR*, 2009. 4