Exploiting Visual Word Co-occurrence for Image Retrieval *

Miaojing Shi ‡, Xinghai Sun ‡, Dacheng Tao #, Chao Xu ‡

‡ Key Laboratory of Machine Perception (Ministry of Education), Peking University, Beijing, P.R.China †
# Centre for Quantum Computation and Intelligent Systems, University of Technology, Sydney, Australia
{shimj,sunxh,xuchao}@cis.pku.edu.cn, Dacheng.Tao@uts.edu.au

ABSTRACT

Bag-of-visual-words (BOVW) based image representation has received intense attention in recent years and has improved content based image retrieval (CBIR) significantly. BOVW does not consider the spatial correlation between visual words in natural images and thus biases the generated visual words towards noise when the corresponding visual features are not stable. In this paper, we construct a visual word co-occurrence table by exploring visual word co-occurrence extracted from small affine-invariant regions in a large collection of natural images. Based on this visual word co-occurrence table, we first present a novel high-order predictor to accelerate the generation of neighboring visual words. A co-occurrence matrix is introduced to refine the similarity measure for image ranking. Like the inverse document frequency (idf), it down-weights the contribution of the words that are less discriminative because of frequent co-occurrence. We conduct experiments on *Oxford* and *Paris Building* datasets, in which the *ImageNet* dataset is used to implement a large scale evaluation. Thorough experimental results suggest that our method outperforms the state-of-the-art, especially when the vocabulary size is comparatively small. In addition, our method is not much more costly than the BOVW model.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Retrieval models

General Terms

Algorithms, Experimentation, Performance.

*Area Chair: Gang Hua

†This paper is supported by NBRPC 2011CB302400, NSFC 60975014, 61121002 and NSFB 4102024.

MM'12, October 29–November 2, 2012, Nara, Japan.

Keywords

BOVW, high-order predictor, co-occurrence matrix

1. INTRODUCTION

Bag-of-visual-words (BOVW) based image representation has received intense attention in recent years and has improved content based image retrieval (CBIR) significantly [23]. BOVW represents an image as a visual document composed of distinctive visual words, which is very important for both the effectiveness and efficiency of image retrieval, especially in a large scale database. The visual document is in the same format as a text document, and image retrieval can therefore be improved by many mature text retrieval techniques, and run as fast as text retrieval. It has been demonstrated that BOVW is one of the most promising approaches for large scale image retrieval [17, 18, 19].

The visual words are derived from clustering and quantization of local features. The scale-invariant feature transform (SIFT) [14] is adopted as the local feature. SIFT features are clustered and quantized into visual words with the K-means algorithm [17]. The tf-idf weight [2] is widely used, which up-weights the contribution of a word that occurs frequently in an image with the tf (term frequency), while it down-weights the contribution of a word that commonly occurs in many images with the idf (inverse document frequency), and is regarded as being less discriminative to the relevance score. Image similarity is measured with the cosine similarity between the query image and an image in the collection. The ranked list is determined according to the values of similarity scores.

Two important issues need to be addressed for the BOVW representation: 1) how to quickly map the local feature to the visual word via the correlation between visual words; 2) how to refine the cosine similarity by reducing the correlation between visual words. Regarding the first issue, as is known from the state-of-the-art methods, each feature is independently mapped to a word, which causes the word generation to be the most time-consuming step in BOVW. Regarding the second issue, the tf-idf weight does not take into account the correlation between visual words, which is important for similarity measure. Like the idf, a word co-occurring with many words is also regarded as being less discriminative and should be down-weighted.

In this paper, we present two approaches for the two issues: the fast visual word generation method and the refined cosine similarity measure, both based on the spatial co-occurrence of visual words. Spatial co-occurrence means

(a) radcliffe_camera (b) vw pattern (17998, 10363) (c) vw pattern (43393, 65762)

Figure 1: An illustration of different co-occurring patterns from a $100K$ vocabulary constructed on *Oxford Building* dataset: (a) one of the selected landmarks: radcliffe_camera, two pairs of visual words (the red dot and the green dot inside the red frames) are selected with their ID (17998, 10363), and (43393, 65762); (b) nine images share a visual word (vw) pair (17998, 10363), but although they are related, the locations of the pair are not fully matched; (c) 9 other images share another pair (43393, 65762), although they are irrelevant.

that visual words co-occur in a small spatial region of an image, instead of in the entire image.

We find that the features of natural images correlate substantially, as illustrated in Fig. 1. One local feature's existence can semantically indicate the presence or absence of certain other features in its neighborhood. This is what we unconsciously do in human vision: a local feature or structure spotted by the human eyes can intuitively hint to our brain that certain other features or structures in the vicinity are possible or impossible. In mathematical language, we build a visual word co-occurrence table to record the co-occurring number of any two visual words in the vocabulary, to indicate the semantic correlation of any two visual words.

Our first perspective is inspired by *predictive coding*, in which random variables can be predicted from previously observed random variables. As shown in Fig .1, some visual word pairs, e.g., (43393, 65762), (17998, 10363), selected from the $100K$ vocabulary on *Oxford Building* dataset [18] frequently co-occur in many images, both relevant and irrelevant. In this context, visual word can be predicted by its co-occurring visual words. The co-occurring visual words are collected from the image database. We develop a high-order predictor to accelerate the generation of visual words. Here the high-order predictor refers to the prediction based on multiple co-occurring words. The information from multiple co-occurring words is used to estimate posterior probability for the purpose of choosing a few candidate words. The prediction can significantly decrease the number of words to search, with a consequent saving in computation.

Our second contribution is to embed the co-occurring information into the cosine similarity. We believe that if a visual word co-occurs with many words, its uniqueness and distinctiveness decline, such as a word occurring in many irrelevant images, as shown in Fig. 1(c). We design a co-occurrence matrix to refine the cosine similarity measure for image ranking, and this co-occurrence matrix helps to increase the discriminative capability of visual words by subtracting the co-occurring redundancies from other visual words.

The rest of the paper is organized as follows. In Section 2, related works are introduced. We present the fast visual word generation algorithm in Section 3. In Section 4 we propose the refined cosine similarity measure for image ranking. Section 5 presents the experimental results. The conclusion is discussed in Section 6.

2. RELATED WORK

This section reviews the state-of-the-art methods in two related aspects: 1) visual word generation for local features; 2) image ranking by exploiting spatial correlation.

Visual word generation. The common method of visual word generation is to index the visual words through a multi-branch tree. Representative tree based algorithms include KD-tree [3] [1] [22] and K-means tree [28] [13] [17]. Arya et al. [1] utilized a priority queue to the tree to speed up the search; Anan et al. [22] utilized multiple random KD-trees (RKD) simultaneously to search words; Uhlmann [28] proposed a "RkNN" tree, which evaluated an efficient approximative search in arbitrary metric spaces; Nister et al. [17] presented a new K-means tree by accessing a single leaf hierarchically; that is, the hierarchical K-means (HKM) tree; Muja et al. [16] selected two best tree structures(RKD and HKM), and utilized a fast library for approximate nearest neighbors (FLANN) to automatically determine the best algorithm and parameters for a given dataset.

Typical algorithms search words for each feature in an image independently of other features, and some researchers have already tried to exploit the nearest neighbor information for certain word generation [8] [31]. In this paper, we endeavor to exploit the correlation information to predict the likely words for each feature, thus the number of words to be searched is reduced significantly. To enhance the precision of the prediction, we design the predictor with high-order posterior probability containing the co-occurring information of multiple words.

Image ranking. Spatial correlation is extensively explored in image retrieval [18, 30, 36, 32, 35, 25, 34, 12, 10]. General approaches come from bundling features in concrete structures and segments; for example, bounding box was manually initialized in [18] and different weighing terms were

added to the visual words inside and outside the bounding box [32]. In [30, 36, 35, 10] features are bundled in maximally stable extremal region (MSER) [15] or a feature's affine-invariant region and taken as contextual visual phrases; these phrases are leveraged to provide more information for the indexing and retrieval processes. Furthermore, spatial correlation has been embedded in building a dictionary of contextual synonyms in [19] and [25].

The spatial co-occurrence used in this paper seems to resemble the visual phrase [33, 35]. But in fact they are utilized in totally different ways. A visual phrase can be considered as a kind of feature expansion to provide better image representation [37]. By contrast, our spatial co-occurrence is used to down-weight a single word: in other words, it removes the correlated noise to refine each local feature. Researches that share a similar motivation in down-weighting the tf-idf of visual words in consideration of their correlations can be found in [7, 29, 26]. Other important papers striving to increase the discriminative capability of visual words include the contextual dissimilarity measure [9] and reciprocal neighborhoods [8].

3. FAST VISUAL WORD GENERATION

Our image retrieval scheme is based on the BOVW model. An image is represented by a vector descriptor, and an entry of the vector is a tf-idf weight for a visual word. The visual words are derived from the clustering and quantization of local SIFT features. As many features co-occur in images, we construct a spatial co-occurrence table to speed up the word generation of the query image and refine the similarity measure in image ranking.

3.1 Co-occurrence Table

Given two visual words w_m and w_n co-occurring in an affine-invariant region, where w_n is located at the center of the region, $N^i(w_m, w_n)$ is their co-occurring number in the i-th image, $N(w_m, w_n)$ is their total co-occurring number over the entire database,

$$N(w_m, w_n) = \sum_i N^i(w_m, w_n) \quad (1)$$

visual word w_m can be regarded as co-occurring with w_n, if its feature is located inside the feature's affine region of w_n. Considering the relatively small size of a feature's affine-invariant region, we magnify the region five times to incorporate more features. We record the co-occurring number of any two visual words and thereby construct a visual word co-occurrence table. Apart from the co-occurring number, the occurrence $N(w_m)$ of every single visual word over the entire database is also recorded.

3.2 Prediction Based on Approximate Posterior Probability

Inspired by *predictive coding*, if we have one letter I, we can claim that the next letter is in a decreased letter set less than 26 letters, furthermore, if we already have four letters $I-m-a-g$, then the next letter is highly likely to be e, for the known word $Image$. We propose a high-order predictor to predict the corresponding word of a feature depending on the multi-words of its neighboring features.

Given a number of visual words $S = \{w_0, w_1 ... w_{s-1}\}$, where s is the set size, we plan to predict the word of their neighboring feature on the basis of S. The co-occurring words and numbers for each visual word collected from the entire database, have been recorded in the co-occurrence table. Since the table is sparse and the size of each list is small, we bundle all the co-occurring visual words of the sample set S as the candidate visual word set $W = \{w_{S0}, w_{S1} ... w_{Sv}\}$ of their neighboring features, v is the size of W.

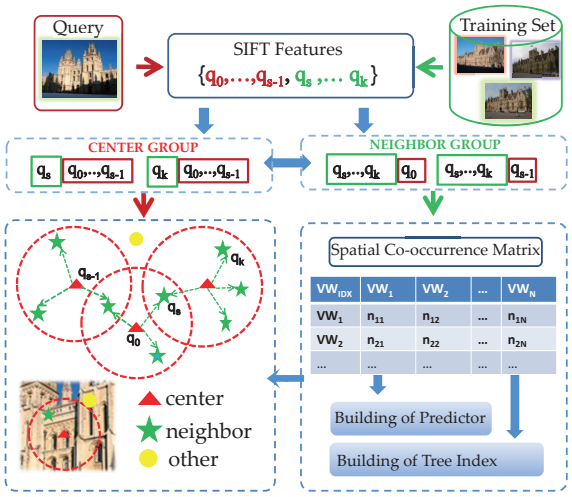


Figure 2: visual word generation with high-order prediction. For the training set, neighboring features are grouped in their affine-invariant regions; in the right block, a co-occurrence table is constructed off-line after mapping *Neighbor Group* to that of the visual words; the high-order predictor is then built for visual word prediction; the tree index is necessary for precise search. For the test images, grouped features are in the inverse form, called *Center Group*, and the left block indicates the prediction modes of the high-order predictor.

We adopt a Bayesian criterion to predict the optimal visual word $\overset{*}{w_s}$ of the neighboring feature from W that is most likely to co-occur with these w_0, w_1,...,w_{s-1}. This likelihood is calculated in terms of the posterior possibility $p(\overset{\wedge}{w_s} | w_0, ..., w_{s-1})$, and the feature in W with the largest conditional probability of occurring is predicted,

$$\overset{*}{w_s} = \arg\max_{\overset{\wedge}{w_s} \in W} p(\overset{\wedge}{w_s} | w_0, ..., w_{s-1}) \quad (2)$$

this conditional probability can be computed from the joint probability,

$$p(\overset{\wedge}{w_s} | w_0, ..., w_{s-1}) = \frac{p(\overset{\wedge}{w_s}, w_0, ..., w_{s-1})}{p(w_0, ..., w_{s-1})} \quad (3)$$

where $p(\overset{\wedge}{w_s}, w_0, ..., w_{s-1})$ is the joint probability of $\overset{\wedge}{w_s}, w_0, ..., w_{s-1}$ in the neighborhood over the entire database, they can be obtained after normalizing their total co-occurring number $N(\overset{\wedge}{w_s}, w_0, ..., w_{s-1})$ by the number of all the visual word occurrences N_T . We decompose $p(\overset{\wedge}{w_s}, w_0, ..., w_{s-1})$:

$$\begin{aligned} &p(\overset{\wedge}{w_s}, w_0, ..., w_{s-1}) \\ &= p(w_{s-1} | \overset{\wedge}{w_s}, w_{s-2}, ..., w_0) ... p(w_1 | \overset{\wedge}{w_s}, w_0) p(w_0 | \overset{\wedge}{w_s}) p(\overset{\wedge}{w_s}) \end{aligned} \quad (4)$$

where the prior $p(w_{s-1} | \overset{\wedge}{w_s}, w_{s-2}, ..., w_0) ... p(w_0 | \overset{\wedge}{w_s})$ measure

the possibility that certain word may co-occur with its nearby words $\hat{w_s}$,..,w_0. The prior $p(\hat{w_s})$ can be estimated from $N(\hat{w_s})$. To formulate this equation, we assume the sampled words $w_0, w_1 ... w_{s-1}$ are conditionally independent, because they have already been generated. In this manner $p(w_0, ..., w_{s-1})$ can be approximated as $\prod_{w_i} p(w_i)$, while $p(w_{s-1}|\hat{w_s}, w_{s-2}, ..., w_0) ... \ p(w_0|\hat{w_s})$ are only dependent on the unknown word $\hat{w_s}$, so they can be approximated by the first-order probabilities on $\hat{w_s}$,

$$\begin{aligned} p(\hat{w_s}, w_0, ..., w_{s-1}) &\approx p(\hat{w_s}) \prod_{w_i \in S} p(w_i|\hat{w_s}) \\ p(w_0, ..., w_{s-1}) &\approx \prod_{w_i \in S} p(w_i) \end{aligned} \tag{5}$$

where $p(w_i|\hat{w_s})$ can be estimated from the prior co-occurring number of $N(w_i, \hat{w_s})$ collected by the co-occurrence table. The posterior probability of Eqn. 2 corresponds to the following decomposition:

$$\begin{aligned} \overset{*}{w_s} &= \arg\max_{\hat{w_s} \in W} p(\hat{w_s}|w_0, ..., w_{s-1}) \\ &\approx \arg\max_{\hat{w_s} \in W} \frac{p(\hat{w_s}) \prod_{w_i \in S} p(w_i|\hat{w_s})}{\prod_{w_i \in S} p(w_i)} \\ &= \arg\max_{\hat{w_s} \in W} \frac{N(\hat{w_s}) \prod_{w_i \in S} \frac{N(w_i, \hat{w_s})}{N(\hat{w_s})}}{\prod_{w_i \in S} N(w_i)} \end{aligned} \tag{6}$$

the approximation indicates that the cascade co-occurring possibilities of $w_0, w_1, ..., w_{s-1}$ on the condition of current optimal visual word $\overset{*}{w_s}$ should be the largest, which means the small region composing the words $w_0, w_1, ..., w_{s-1}, w_s$ is the most probable co-occurring pattern at the current location.

Since the co-occurrence table is sparse, zero terms in the table will affect the calculation in Eqn. 6, which makes the probability zero. To solve the problem, we group $\hat{w_s}$ based on its number of zero items in $N(w_i, \hat{w_s})$. In the same group, we can compare two $p(\hat{w_s}|w_0, ..., w_{s-1})$ by simply removing the zero items. For those $\hat{w_s}$ with different numbers of zeros terms in $N(w_i, \hat{w_s})$, we regard the one $p(\hat{w_s}|w_0, ..., w_{s-1})$ with less zero items $N(w_i, \hat{w_s})$ in its decomposition is larger than that with more zero items.

3.3 Grouping of Neighbors and Centers

Because the affine invariant region of one feature is too small to contain more neighbors, we attempt to access more co-occurring neighbors for the generation of visual words by enlarging the region five times.

Our proposed scheme is shown in Fig. 2. Instead of looking for neighboring features $q_s, .., q_k$ around the center q_0, we record all the centers whose regions include a certain feature, i.e., q_s. In this manner, more neighboring centers of q_s are incorporated in the inverse form. The observation of the inverse form is because of the asymmetry between the center q_0 and its neighbor q_s, that is, when q_s is taken as a center, its region might not include q_0.

In summary, the co-occurrence table is constructed and utilized in the usual way; every center feature is grouped with its neighbors in the affine-invariant region, called *Neighbor Group* in Fig. 2. After mapping *Neighbor Groups* of features into those of visual words, the co-occurrence table is constructed. In contrast, the prediction of a word is carried out in an inverse way. We record the centers that one feature belongs to, as shown in Fig. 2, the *Center Group*. After generating all the center words $(w_0, ..., w_{s-1})$ from their corresponding features $(q_0, ..., q_{s-1})$, the next step is to determine the candidate visual word set of features, i.e., q_s. Since the sizes of *Center Group* and *Neighbor Group* are small and the computation of the approximate posterior probability is easy, all the neighboring words collected from the co-occurrence table of q_s's center lists are regarded as its candidate visual word set. Thus a number of approximate posterior probabilities are calculated from Eqn. 6, and the optimal $\overset{*}{w_s}$ is discovered through comparison. The center independence assumption in Eqn. 6 is an approximation to simplify the calculation of the high-order probability. Empirically, the number of co-occurring center words is not large and usually no more than 4 for predicting each single word, so the center independence assumption is reasonable and performs well. This manner of prediction is easy but not precise, because it only indicates higher likelihood. To generate the precise word, Euclidean distance is necessary. We select K-candidates with the top K-maximal probabilities to compute their Euclidean distances to the feature q_s. The word with the minimal Euclidean distance is the selected word.

.

3.4 Procedure of Visual Word Generation

We have presented key techniques for fast word generation based on spatial co-occurrence information, and we now detail the procedure.

1. Given a query image, extract the local features.

2. Randomly select 20% features as centers (as illustrated in Fig. 2, the left block, the red triangles) determine their neighboring features (the green stars) in their magnified affine-invariant regions (the red circles).

3. Map the 20% center features to their words through FLANN.

4. Predict those neighboring features of centers whose *Center Groups* contain the words of the quantized centers. Provide the K-candidate words by Eqn. 6 and generate the words for the features by their Euclidean distances.

5. If any features (the yellow dot) do not belong to any centers, or the Euclidean distance from the current optimal visual word to the corresponding feature is still large, apply FLANN to action further search.

The proportion of the initial center features is set to 20%. This setting maintains a good balance between a wide coverage of the whole image and the small time cost of an exact nearest neighbor search for the center features. Random selection offers a simple yet effective treatment for every feature in an image. The word generation algorithm runs fast because it utilizes correlation information between features instead of generating each feature independently. The prediction based on approximate posterior probabilities provides fewer candidates that are more likely to occur than the widely used tree structure, so computation complexity is reduced.

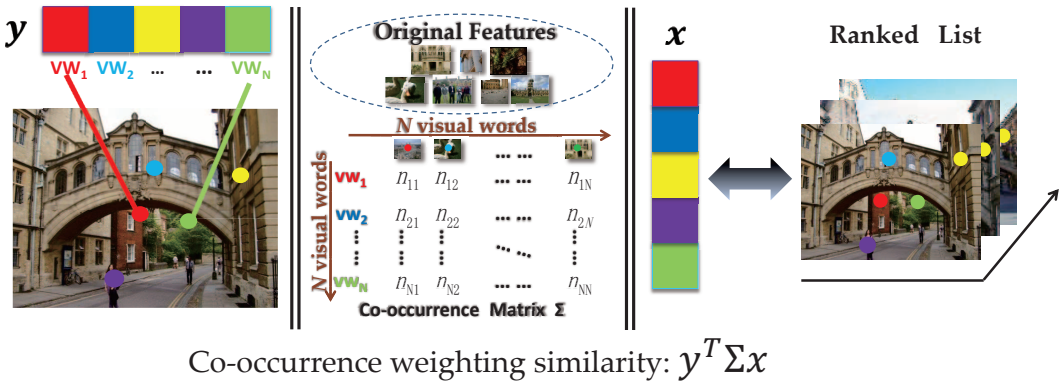


Figure 3: Overview of similarity measure by co-occurrence weighting scheme.

4. IMAGE RANKING

In the BOVW model, once a query image is given, the images in the database are ranked in the order of their similarity scores to query. The general similarity measure is the cosine distance. As previously noted, the BOVW model does not take the correlation between visual words into account, so the spatial information is ignored in the cosine similarity. We embed the spatial co-occurrence information in a refined cosine similarity to improve the ranking performance, instead of adding a re-ranking stage to conduct spatial verification generally.

4.1 Cosine Similarity

In the BOVW representation, an image is represented by a N-dimensional vector (N is the vocabulary size), and an element of the vector is the tf-idf value of a word. The tf-idf weight [2] is a commonly used scheme in image ranking. It down-weights the contribution that commonly occurs and and less discriminative, words are therefore included in the relevance score. The simplest ranking function is by the normalized cosine measure:

$$Sim(x,y) = \frac{x^T y}{||x||||y||} \tag{7}$$

where y is query vector, x is the vector of an image in database. The larger $Sim(x,y)$ is, the more relevant x is to the query image, thus, the higher it is ranked in the returned list.

We embed the co-occurrence information into the cosine similarity measure. Mathematically, the cosine similarity is equivalent to using a unit diagonal matrix I to measure the similarity between two vectors, $Sim(x,y) = \frac{x^T I y}{||x||||y||}$. In the following, we utilize a co-occurrence matrix $\sum$ to refine the similarity measure for image ranking $Sim(x,y) = \frac{x^T \sum y}{||x||||y||}$, as shown in Fig. 3.

4.2 Co-occurrence Weighting Similarity

As is known, idf is a measure of whether a word is common or rare across entire database. If a word is common, it is less discriminative to the relevance score and its contribution will be down-weighted through idf. In this section we manage to further exploit the importance of a visual word from its co-occurring attribute with other words. As mentioned in the Introduction and Fig. 1: 1) the images containing the same co-occurring pairs might be totally different; 2) the locations of co-occurring pairs for the related images could be different; 3) if a visual word co-occurs with a large number of words, its uniqueness and distinctiveness decline. Thus, although a visual word associated with high occurrence or co-occurrence can be regarded as having a high probability of occurring in the word generation process, it performs a less discriminative role in retrieving a visual object. Like idf, we claim that if a visual word commonly co-occurs with other words, and is therefore less discriminative to the relevance score, we should down-weight its contribution in the similarity measure.

Since our approach focuses on near duplicated image retrieval, the number of relevant images for a given query should be small in large scale dataset. All irrelevant images in the database can be taken as noises that produces negative information in image ranking [27]. We build a co-occurrence matrix $\mathbf{\Sigma} = \{n_{ij}\}$ on the noisy set, referred to Section 3.1, Fig.3; its element $n_{ij} = N(w_i, w_j)$ denotes the co-occurring number of visual word w_j with visual word w_i. We propose a new similarity measure, called the co-occurrence weighting similarity measure (Co-Sim):

$$\begin{aligned} Sim(x,y) &= \frac{x^T(I - \frac{1}{\beta}\Sigma)y}{\|x\|\|y\|} \\ &= \frac{x^T y}{\|x\|\|y\|} - \frac{1}{\beta}\frac{x^T \Sigma y}{\|x\|\|y\|} \end{aligned} \tag{8}$$

where $\frac{x^T y}{\|x\|\|y\|}$ is the basic cosine similarity and $\frac{x^T \Sigma y}{\|x\|\|y\|}$ is the new term introduced to encode the correlation between two any visual words. The linear combination coefficient $\frac{1}{\beta}$ allows a continuum of the model between the cosine form and the Σ form. Σ describes the co-occurring distribution of the noisy images, if $\frac{x^T \Sigma y}{\|x\|\|y\|}$ is comparative large, it means that the visual words of the current image are more likely to be drawn from the noisy distribution. To indicate this negative effect, a subtraction is operated. This idea is similar to the contextual dissimilarity measure [9] and cross-category knowledge [6, 20]. In fact, because the number of images relevant to a query is so small compared to the large scale database, the negative effect of the entire corpus obviously overwhelms the trivial positive effect of the relevant images. In practice, we do not need the labels of the images, like-

wise [5], because the co-occurrence matrix is constructed on the entire database and naturally reflects the statistical distribution of the database.

In fact, β indicates a normalization of the matrix. Empirically, if we normalize the co-occurrence matrix Σ by rows ($\sum_j N(w_i, w_j) = 1$), only slight adjustment by β is then needed to overcome the statistical uncertainty of the matrix when applied to a different database. In general $\beta \in (1.2, 1.5)$, the performance stays stable in this case.

Our proposed similarity (Eqn.8) can be considered as an inner product of a query vector y and vector $x^T(I - \frac{1}{\beta}\Sigma)$, where $x^T(I - \frac{1}{\beta}\Sigma)$ is a linear transformation performed on the vector x (or the same transformation from the query viewpoint, $(I - \frac{1}{\beta}\Sigma)y$ and x). Such a linear transformation acts a down-weighting role on the tf-idf of each visual word. The transformed vector is denoted as $x' = x^T(I - \frac{1}{\beta}\Sigma)$ (or $y' = (I - \frac{1}{\beta}\Sigma)y$), where each element of x' can be rewritten as:

$$x_i^{'} = x_i - \frac{1}{\beta}\sum\nolimits_j N(w_i, w_j)x_j, \tag{9}$$

here, x_i and x_j denote the corresponding tf-idf values of the i-th and j-th visual words for the image and $N(w_i, w_j)$ is the co-occurring number as a weighting term. As previously mentioned, if a visual word co-occurs with a large number of words, its uniqueness and distinctiveness decline.

The proposed method appears to be similar to query expansion [19] [10]. In fact, we do not add any new visual words for retrieval; on the contrary, we down-weight the contribution of each visual word in the similarity measure. Moreover, we do not require any knowledge of the matched visual words, our observation is focused on the considerable unmatched words. They are penalized in scoring by subtracting the co-occurring redundancies from other visual words. We admit that mis-scores may exist as a result of building the co-occurrence matrix on the entire database, but the general elimination of redundant information always outweighs the mistakes we have made, which means that in spite of decline of score on each visual word, the relative scores of importance for distinctive words is comparatively enhanced.

4.3 Algorithm Flow

We describe the image retrieval method based on the co-occurrence of visual words, as shown in Algorithm 1. In the offline process, the co-occurrence table is built on the entire image database. The predictor and the co-occurrence matrix are learned for visual word generation and image ranking, respectively. In the online phase, given a query image, we first generate image representation through the visual word high-order predictor. The co-occurrence weighting similarity measure is then adopted to evaluate the candidate images in the database. The ranked list is returned according to the similarity scores.

Algorithm 1 Exploiting visual word co-occurrence for image retrieval

build co-occurrence table (matrix Σ);
input: query image y, max-check
output: ranked list S
visual word generation
randomly select 20% features as centers
group features with their centers in the neighborhood
for center feature q_c **do**
  accurate generate w_c by FLANN
end for
for neighboring features q_s **do**
  while current check $K <$ length $(W)<$ max-check **do**
    provide K candidate words with
    $K - \arg\max_{\hat{w_s} \in W} p(\hat{w_s} | w_0, ..., w_{s-1})$
    compute $\overset{*}{w}_s$ with minimal Euclidean distance to q_s
  end while
  while length $(W) <$ current check $K <$ max-check **do**
    search $\overset{*}{w}_s$ by FLANN
  end while
end for
image ranking
for each image x **in** database **do**
  similarity measure $Sim(x, y)$ to query
  $Sim(x, y) = \{x | \forall y : x^T(I - \frac{1}{\beta}\Sigma)y\}$
end for
return the ranked list $S : S - \arg\max Sim(x, y)$.

5. EXPERIMENTAL RESULTS

5.1 Dataset and Evaluation

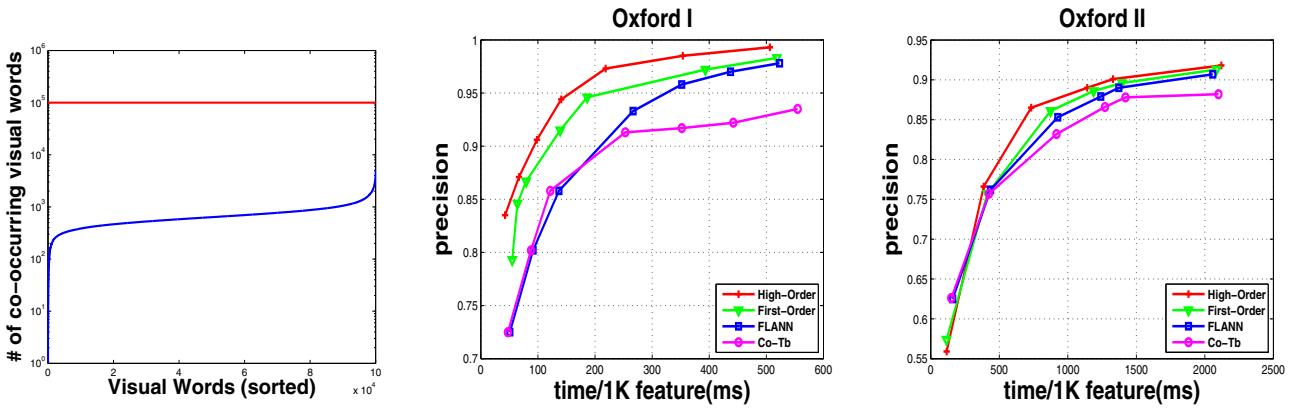
The *Oxford* dataset. This dataset [18] of 5062 images is a standard image retrieval test set, which we name *Ox* for short. 55 images comprising 11 *Oxford* landmarks are selected as query images, and their ground truth retrieval results are provided.

The *ImageNet* dataset. Approximately $100K$ and $500K$ images are sampled from $10M$ images in *ImageNet* [4], which we respectively call *I1* and *I2* for short.

The *Paris* dataset. This dataset contains 6300 images by querying the associated text tags for famous *Paris* landmarks such as "*Paris Eiffel Tower*" or "*Paris Arc de Triomphe*" [21].

Four databases are constructed for conducting the experiments: the first experiment is on the *Oxford* (*Ox*, 5062 images) and the second is on *Paris* (6390 images) dataset. To implement the proposed method on a large scale dataset, we construct two new databases by combining the *Oxford* dataset with *ImageNet* $100K$ and $500K$ datasets, *Ox+I1*, *Ox+I2*. The new collections are composed of approximately $105K$ and $505K$ images. Since the images from $100K$ and $500K$ *ImageNet* are not relevant to the 55 queries in *Oxford* dataset, they are noisy images that disturb the image retrieval process. Evaluations are still carried out on the 55 *Oxford* queries.

According to our two contributions, the experiment results are presented in two parts: the visual word generation results and the similarity measure in image ranking results. Average precision (AP) is evaluated to measure retrieval performance, which is defined as the area under the precision-recall curve. The AP score is computed for each query and averaged to obtain a mean average precision (mAP). Comparisons on hard- and soft-assignment, with or without bounding box, demonstrate a superior performance to the state-of-the-art presented in [19] [32] [35] [10].



(a) statistics of the co-occurrence table; (b) vocabulary size is $100K$; (c) vocabulary size is $1M$;

Figure 4: (a)The corresponding co-occurring visual word numbers for every visual word, they are sorted in ascending order and shown in the blue line. It can be intuitively compared with the total visual word number (the red line, $100K$); (b) (c) the visual word generation results of *Oxford* dataset.

5.2 Visual Word Generation

The experiments are tested on the *Oxford* dataset. The entire database is split into two parts: one part containing 90% images is used to build the co-occurrence table, while the other 10% is retained for query evaluation. We randomly split the dateset 10 times and report the average visual word generation time per $1K$ feature. The precision of the approximate nearest neighbor search is defined as the proportion of the number of exact nearest neighbors to the total. We set different max-checks and compare the corresponding precision and time. Since the generation results are used to represent the image, we need to guarantee the level of precision. Once the max-check is larger than the number of co-occurring visual words to be searched, the tree index FLANN [16] is adopted to continue the search; those nodes that have been checked in the predictor will not be checked again. In this manner, the high-order predictor will not affect retrieval effectiveness but will significantly improve the overall retrieval efficiency.

Note that the quantization time we provide is tested on our computer using the public FLANN code on the UBC website. For different datasets and computer configurations, the time costs might be different.

***Oxford* dataset.** Vocabularies are clustered by the hierarchical K-means tree [17]. An average of $3,228$ local SIFT descriptors are extracted from each image in the *Oxford* dataset. Corresponding average generation time per image on the *Oxford* dataset are shown in Fig. 4. Comparative tests are mainly carried out on the representative fast library approximate nearest neighbor (FLANN) algorithm [16], indicated by the blue line, which is a combination library of two best tree structures of random KD-trees (RKD)[22] and hierarchal K-means tree (HKM)[17]. The magenta line shows the performance of the co-occurrence table (Co-Tb) method [31], which is superior to FLANN when the approximate nearest neighbor precision is comparatively low, although limited to a certain value with an increase in the number of predefined checks. To demonstrate the effectiveness of the high-order predictor, we also give the generation performance of first-order predictor (green line). We select its nearest center for each feature as its co-occurring center and build the predictor on first-order conditional probabilities, which are easily obtained from the visual word co-occurrence table.

From the results in Fig. 4 we conclude that, in general, the high-order predictor (red line) is clearly superior to FLANN (blue line), especially when the max-check is small. This is because the high-order co-occurring information contributes to the generation during this stage, which holds a nearly constant time complexity. By comparing the first-order predictor with high-order predictor, we see that the high-order predictor exhibits a more appealing performance as a result of incorporating more co-occurring information.

The experimental performance on the $100K$ vocabulary is better than that of the $1M$ vocabulary. With increasing in vocabulary size, data bias is introduced to the conditional probability. Moreover, the co-occurrence is more stochastic when vocabulary size is large, which leads to a situation, for any given center, where the number for many of its co-occurring visual words is 1. Their conditional probabilities can therefore hardly be distinguished, nor can they be indexed in the predictor. Notwithstanding, this does not mean that the smaller the vocabulary is, the better the performance will be. A certain degree of sparsity is necessary to predict the visual word in a nearly constant time. On the other hand, with a decrease in vocabulary size, the performance of the tree index is closer to its theoretically optimal time complexity [1].

5.3 Image Ranking

We report the image ranking results with our proposed co-occurrence weighting similarity in *Oxford*, *Paris* and *ImageNet* datasets. Parameter sensitivity is analyzed and we then carefully test our proposed similarity measure in multi-scenarios: with hard-assignment or soft-assignment, with or without query bounding box, The performance in each case proves that our method achieves a considerable improvement over other methods.

Parameter sensitivity. Table 1 shows the retrieval performances with different parameter β settings on the *Oxford* and *Paris* datasets. Here we normalize the co-occurrence matrix Σ by rows ($\sum_j N(w_i, w_j) = 1$). It can be seen from

(a) Oxford ($100K$); (b) Paris ($500K$);

Figure 5: An illustration of retrieval results for three query images on *Oxford* and *Paris* datasets. The two rows of search results correspond respectively hard-assignment and our proposed method. False alarms are marked with red boxes.

Table 1: Comparison of different parameter settings on the *Oxford* and *Paris* datasets. Corresponding numbers are their mAP values.

β	Oxford($100K$)	Oxford($1M$)	Paris
1.5	0.5827	0.6590	0.7034
1.4	**0.5844**	0.6596	**0.7037**
1.3	0.5830	**0.6598**	0.7035
1.2	0.5803	0.6597	0.7025
baseline	0.514[18]	0.613[18]	0.666[21]

the table that the retrieval performances vary little with changing β. In particular, $\beta \in (1.2, 1.5)$, the ranking results stay stable. We also find that the parameter β stays stable across different vocabulary sizes (as shown in the table, the best parameter does not change greatly in the $100K$ vocabulary vs. the $1M$ vocabulary), and across different datasets (*Oxford* vs. *Paris*). Therefore, in real implementation, regardless of vocabulary size and dataset to retrieval, β in our proposed similarity measure can simply set to 1.35 for satisfactory performance. A small adjustment (± 0.15) might be added to refine the final performance, as shown in Fig. 5.

Effect of vocabulary size. We evaluate the effectiveness of our co-occurrence weighting similarity measure (Co-Sim) on the *Oxford* dataset for different hard-assigned (HA) vocabularies, as shown in Fig. 6. The mAP of baselines and the corresponding improvements by Co-Sim are shown by the red line and blue line, respectively.

One visual word is too coarse to distinguish descriptors extracted from semantically different objects. This polysemic phenomenon is particularly prominent when the visual vocabulary is small. Our method, however, is of greater benefit on smaller vocabularies. We attribute this to its ability to overcome the uncertainty of visual word co-occurrence when

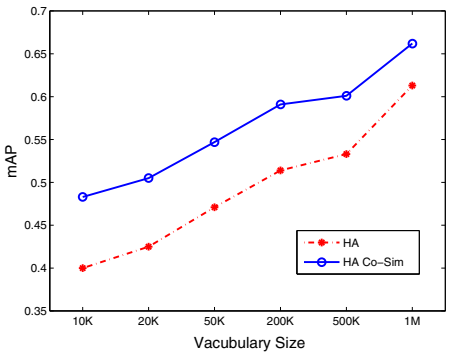


Figure 6: Comparisons of two approaches with different sizes of hard-assigned vocabulary

the smaller vocabularies are adopted. With an increase in vocabulary size, the improvement produced by our scheme declines due to the bias of the matrix.

Hard assignment vs. soft assignment. The proposed Co-Sim is simply a novel similarity measure, which can be embedded into any ranking or re-ranking method. In Table 2 we test its performance with both hard- and soft-assigned vocabularies [19] on the *Oxford* dataset. It can be seen that the increases in mAP on different vocabularies are considerable (14.8% and 13.8% improvements for the hard- and soft-assigned $100K$ vocabulary, 8.0% and 11.4% improvements for the $1M$ vocabulary).

Though soft-assignment achieved an apparent improvement by assigning a weighted combination of visual words to each descriptor, our Co-Sim measure on the hard-assigned vocabulary (Hard + Co-Sim) has already clearly outperformed it.

Table 2: Performance of Co-Sim embedded on hard- and soft-assigned vocabularies with or without bounding box (BB) on the *Oxford* dataset. These results are the corresponding mAP values for $100K$ and $1M$ vocabularies.

	BB	Co-Sim	Ox ($100K$)	Ox ($1M$)
Hard			0.514	0.613
Hard		+	**0.584**	**0.660**
Hard	+		0.514	0.613
Hard	+	+	**0.577**	**0.648**
Soft			0.529	0.640
Soft		+	**0.602**	**0.719**
Soft	+		0.554	0.673
Soft	+	+	**0.611**∗	**0.730**∗

With vs. without query bounding box. In real implementation, visual bounding boxes are often manually labeled for query images to get rid of the nonsensical parts of images. We compare the results of our Co-Sim with and without query bounding box. Table 2 shows that the new scheme without bounding box apparently outperforms the results of hard- and soft-assignment even with bounding box. We suggest that this is because, the proposed similarity, while it does not need a manually labeled bounding box, functions intrinsically as a virtual bounding box, with the contribution of nonsensical words being smoothed down to trivia. Such a virtual bounding effect can work better than a manually labeled bounding box, because the elimination of redundant information is carried out in the feature space rather than the image geometric space.

We also note that if we add a real hard bounding box to our scheme, the performance improves little, and even in some cases it declines. This is due to the penalty effect implemented on all the visual words as mentioned at the end of Section 4. If we add a real bounding box, the general elimination of the redundancy effect might not always outweighs the mistakes we have made on the matched points, although in general mis-matched points are in the majority.

By embedding the Co-Sim into soft-assignment with query bounding box, the final mAP for $1M$ vocabulary reaches 0.73, and for the $100K$ vocabulary it reaches 0.611. Note that even without spatial verification techniques, our method has already outperformed the retrieval results in [30] [11].

Comparison with the state-of-the-art. Detailed comparisons are evaluated on the *Oxford* dataset with $100K$ and $1M$ hard-assigned vocabularies. Representative approaches include the baseline (Philbin07) [18], Soft Assignment (SA) [19], Contextual Model (CM) [32] constructed from the language model [24], Contextual Visual Vocabulary (CVV) [35] and Spatial Co-occurrence Query Expansion (SCQE) [10].

Results are summarized in Table 3. In general, our Co-Sim method is superior than the state-of-the-art. The mAP for the $100K$ vocabulary is 0.584, and achieves the best performance of 0.611 on soft-assignment (Co-Sim (Soft)); for the $1M$ vocabulary, Co-Sim is a little lower than soft-assignment [19] due to the increased bias of the co-occurrence matrix. Nevertheless, because of the scalability of our Co-Sim measure, it can be evaluated on the soft-assigned vocabulary, and the mAP reaches the highest value of 0.73.

Large scale evaluation. To validate the effectiveness of our proposed similarity measure on a large scale dataset, *ImageNet* dataset is introduced to carry out large scale evaluation[1]. Comparisons with Soft Assignment (SA) [19], Contextual Visual Vocabulary (CVV) [35], and SCQE [10] on the *Ox+I1* and *Ox+I2* datasets are given in Table 4. For [35] and [10], they exhibit inferior performances on the large scale dataset as a result of the noise introduced by the *ImageNet*. In contrast, our similarity measure is robust enough to denoise the irrelevant images on a large scale. Overall, we show that it is always beneficial for us to embed Co-Sim in image retrieval.

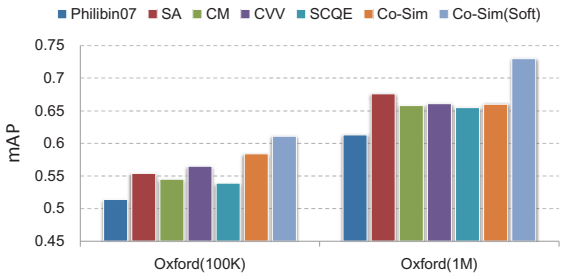


Figure 7: Bar graph of the results in comparison with representative approaches.

Table 3: Detailed experimental mAPs in comparisons with representative approaches.

Vocabulary	*Ox(100K)*	*Ox(1M)*
Philbin(07)[18]	0.514	0.613
SA[19]	0.554	0.676
CM[32]	0.545	0.658
CVV[35]	0.565	0.661
SCQE[10]	0.539	0.655
Co-Sim	0.584	0.660
Co-Sim(Soft)	**0.611**	**0.730**

Table 4: Large scale retrieval results in comparison with state-of-the-art models. Corresponding numbers are their mAP values.

Vocabulary	*Ox+I1(1M)*	*Ox+I2(1M)*
Philbin(07)[18]	0.566	0.499
SA[19]	0.603	0.534
CVV[35]	0.610	0.549
SCQE[10]	0.616	0.574
Co-Sim	**0.630**	**0.615**

5.4 Computational Complexity

The co-occurrence matrix is sparse since the local regions are small and the number of co-occurrence patterns is also small. Thus, 1) the storage overheads for $100K$ and $1M$ vocabularies (it is mainly dependent on the method for extracting the local regions and the sizes of the extracted regions) are roughly identical, around $500MB$ and is ac-

[1]We have also evaluated the cross-dataset performance by obtaining the co-occurrence table from the *ImageNet* dataset and testing it on the *Oxford* dataset. The mAP is mildly improved from 0.613 (baseline) to 0.625, instead of 0.66 on its own co-occurrence table.

ceptable for practical systems, and 2) the average computational time for the *Oxford* dataset ($100K$ vocabulary) is around $245ms$ per query for the improved cosine distance and around $28ms$ for the cosine distance. However, this time increment is compensated for by the time decrement in word prediction (i.e., at the 0.95 precision, the cost of time per image (around $3K$ features) drops from $1100ms$ by FLANN [16] to $500ms$ by the proposed high-order predictor shown in Fig. 4). Notwithstanding, for large scale image retrieval, the calculation of the similarity between image pairs will dominate the retrieval time, i.e., for the *Ox+I2* test with $1M$ vocabulary, the calculation time of the proposed Co-Sim can reach $3.2s$ per query compared to that of $0.98s$ for single *Ox* test.

6. CONCLUSIONS

This paper has proposed a novel image retrieval approach that exploits the spatial co-occurrence of visual words. It improves the retrieval performance on standard datasets by presenting two novel methods: fast word generation via candidate prediction and refined cosine similarity measure via down-weighting. By exploiting the visual word co-occurrence information, a high-order predictor and co-occurrence weighting cosine distance are developed and embedded into our scheme for visual word generation and similarity measure in image ranking, respectively. The word generation is faster than approaches using tree index, and the refined similarity measure is more precise than the original cosine similarity measure, thus the entire image ranking performance is improved. The theoretical analysis presented in this paper also proves the superiority of our method. The two novel techniques can be used independently and can be embedded in most image retrieval algorithms, as shown in our experiments. They can also be applied to other applications, such image classification, object recognition and video surveillance.

7. REFERENCES

[1] S. Arya, D. Mount, N. Netanyahu, R. Silverman, and A. Wu. An optimal algorithm for approximate nearest neighbor searching fixed dimensions. *JACM*, 1998.

[2] R. Baeza-Yates and B. Ribeiro-Neto. *Modern information retrieval*. 1999.

[3] J. Beis and D. Lowe. Shape indexing using approximate nearest-neighbour search in high-dimensional spaces. In *CVPR*, 1997.

[4] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[5] B. Geng, Y. Li, D. Tao, M. Wang, Z. Zha, and C. Xu. Parallel lasso for large scale video concept detection. *IEEE TMM*, 2012.

[6] B. Geng, D. Tao, and C. Xu. Daml: Domain adaptation metric learning. *IEEE TIP*, 2010.

[7] H. Jegou, M. Douze, and C. Schmid. On the burstiness of visual elements. In *CVPR*, 2009.

[8] H. Jegou, M. Douze, and C. Schmid. Exploiting descriptor distances for precise image search. *Research Report*, 2011.

[9] H. Jegou, C. Schmid, H. Harzallah, and J. Verbeek. Accurate image search using the contextual dissimilarity measure. *TPAMI*, 2010.

[10] Y. Li, B. Geng, Z. Zha, Y. Li, D. Tao, and C. Xu. Query expansion by spatial co-occurrence for image retrieval. In *ACM MM*, 2011.

[11] Y. Li, B. Geng, Z. Zha, D. Tao, L. Yang, and C. Xu. Difficulty guided image retrieval using linear multiview embedding. In *ACM MM*, 2011.

[12] L. Liu and L. Wang. Exploring latent class information for image retrieval using the bag-of-feature model. In *ACM MM*, 2011.

[13] T. Liu, A. Moore, A. Gray, and K. Yang. An investigation of practical approximate nearest neighbor algorithms. *NIPS*, 2004.

[14] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004.

[15] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. In *BMVC*, 2004.

[16] M. Muja and D. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. In *VISAPP*, 2009.

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

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

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

[20] G. Qi, C. Aggarwal, Y. Rui, Q. Tian, S. Chang, and T. Huang. Towards cross-category knowledge propagation for learning visual concepts. In *CVPR*, 2011.

[21] D. Qin, S. Gammeter, L. Bossard, T. Quack, and L. van Gool. Hello neighbor: accurate object retrieval with k-reciprocal nearest neighbors. In *CVPR*, 2011.

[22] C. Silpa-Anan and R. Hartley. Optimised kd-trees for fast image descriptor matching. In *CVPR*, 2008.

[23] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *CVPR*, 2003.

[24] F. Song and W. Croft. A general language model for information retrieval. In *ACM CIKM*.

[25] W. Tang, R. Cai, Z. Li, and L. Zhang. Contextual synonym dictionary for visual object retrieval. In *ACM MM*, 2011.

[26] Q. Tian, S. Zhang, W. Zhou, R. Ji, B. Ni, and N. Sebe. Building descriptive and discriminative visual codebook for large-scale image applications. *Multimedia Tools and Applications*, 2011.

[27] X. Tian, D. Tao, and Y. Rui. Sparse transfer learning for interactive video search reranking. *ACM TMCCA*, 2011.

[28] J. Uhlmann. Satisfying general proximity/similarity queries with metric trees. *IPL*, 1991.

[29] X. Wang, M. Yang, T. Cour, S. Zhu, K. Yu, and T. Han. Contextual weighting for vocabulary tree based image retrieval. In *ICCV*, 2011.

[30] Z. Wu, Q. Ke, M. Isard, and J. Sun. Bundling features for large scale partial-duplicate web image search. In *CVPR*, 2009.

[31] R. Xu, M. Shi, B. Geng, and C. Xu. Fast visual word assignment via spatial neighborhood boosting. In *ICME*, 2011.

[32] L. Yang, B. Geng, A. Hanjalic, and X. Hua. Contextual image retrieval model. In *ACM CIVR*, 2010.

[33] J. Yuan, Y. Wu, and M. Yang. Discovery of collocation patterns: from visual words to visual phrases. In *CVPR*, 2007.

[34] J. Yuan, Z. Zha, Y. Zheng, M. Wang, X. Zhou, and T. Chua. Learning concept bundles for video search with complex queries. In *ACM MM*, 2011.

[35] S. Zhang, Q. Huang, G. Hua, S. Jiang, W. Gao, and Q. Tian. Building contextual visual vocabulary for large-scale image applications. In *ACM MM*, 2010.

[36] S. Zhang, Q. Tian, G. Hua, Q. Huang, and S. Li. Descriptive visual words and visual phrases for image applications. In *ACM MM*, 2009.

[37] Y. Zhang, Z. Jia, and T. Chen. Image retrieval with geometry-preserving visual phrases. In *CVPR*, 2011.