DOI:
10.1109/TMM.2015.2389616

Document Version
Peer reviewed version

Database Saliency for Fast Image Retrieval

Yuan Gao, Miaojing Shi, Dacheng Tao, *Fellow, IEEE*, and Chao Xu, *Member, IEEE*

***Abstract*—The bag-of-visual-words (BoW) model is effective for representing images and videos in many computer vision problems, and achieves promising performance in image retrieval. Nevertheless, the level of retrieval efficiency in a large-scale database is not acceptable for practical usage. Considering that the relevant images in the database of a given query are more likely to be distinctive than ambiguous, this paper defines "database saliency" as the distinctiveness score calculated for every image to measure its overall "saliency" in the database. By taking advantage of database saliency, we propose a saliency-inspired fast image retrieval scheme, S-sim, which significantly improves efficiency while retains state-of-the-art accuracy in image retrieval. There are two stages in S-sim: the bottom-up saliency mechanism computes the database saliency value of each image by hierarchically decomposing a posterior probability into local patches and visual words, the concurrent information of visual words is then bottom-up propagated to estimate the distinctiveness, and the top-down saliency mechanism discriminatively expands the query via a very low-dimensional linear SVM trained on the top-ranked images after initial search, ranking images are then sorted on their distances to the decision boundary as well as the database saliency values. We comprehensively evaluate S-sim on common retrieval benchmarks, e.g., Oxford and Paris datasets. Thorough experiments suggest that, because of the offline database saliency computation and online low-dimensional SVM, our approach significantly speeds up online retrieval and outperforms the state-of-the-art BoW-based image retrieval schemes.**

***Index Terms*—Bag-of-visual-words (BoW), bottom-up saliency, database saliency, image retrieval, top-down saliency.**

I. Introduction

BAG-OF-VISUAL-WORDS (BoW) representation has been effectively adopted in a number of computer vision problems, e.g., image retrieval [1]. Visual images are ranked using term frequency inverse document frequency (TFIDF) of visual words computed efficiently via an inverted index [2]. The advantage of the algorithm is the high efficiency, while the disadvantage is the low effectiveness for lack of spatial information among the visual words. For example, similar to a word identification, when we receive three letters, "a, e, r," how to decide it is the word "are" or "ear", even just a part of "hear"? Therefore, many re-ranking methods are proposed to improve retrieval performance. Representative schemes include spatial re-ranking [1], [3], query expansion (QE) [4], [5], and relevance feedback [6], [7]. Geometric structure [8], [9] and text/meta features [10]–[12] are also taken into account to improve precision, and significant improvements are achieved in state-of-the-art BoW-based retrieval schemes.

Despite the promising performance after re-ranking, however, the level of retrieval efficiency in a large-scale database is not acceptable for practical usage. The existing re-ranking schemes either come at the cost of manual intervention [6], [7] or are time-consuming [10]–[12] for online search. Candidate visual images from a short ranked list have to be spatially verified before queries can be expanded or the list re-ranked [1], [4], [5]. To improve re-ranking efficiency, we propose a novel concept in this paper: database saliency.

In large-scale image retrieval, the number of relevant images in the database for a given query, regardless of its identity, is extremely small compared to the entire image collection. That is to say, queries are always discriminative to the entire database. One of the main challenges, in the retrieval for a query image, is thus to distinguish relevant images from images that are similar (in TFIDF scoring) to query as well but are irrelevant actually. Building upon this observation, we claim that query's relevant images in the database are more likely to be distinctive images than ambiguous images. In this paper, we define database saliency as the distinctiveness score calculated for every image to measure its overall saliency in the database. Less distinctive images that are associated with smaller weights are regarded as less relevant.

The distinctiveness of visual images is computed offline, independently of the query. As a new strategy of exploiting the saliency in a database, database saliency can be integrated with any standard image retrieval architecture, and is always beneficial to retrieval performance. In this paper, we plug it into the discriminative query expansion [5] and propose a saliency-inspired fast image retrieval scheme, S-sim. We demonstrate that S-sim significantly improves efficiency and at least retains state-of-the-art accuracy in image re-ranking by simply evaluating a very low dimensional linear SVM.

There are two stages in the proposed method: the bottom-up saliency mechanism computes the database saliency value of each image by hierarchically propagating a posterior probability in it, while the top-down saliency mechanism discriminatively expands the query from top-ranked images after the initial search. It is similar in spirit to [13], in which saliency is modeled using both top-down visual cue (color) and bottom-up

Manuscript received May 06, 2014; revised November 03, 2014; accepted December 29, 2014. Date of publication January 08, 2015; date of current version February 12, 2015. This work was supported by NBRPC 2011CB302400, NSFC 61375026, 2015BAF15B00, JCYJ 20120614152136201, and the Australian Research Council under Project DP-140102164, Project FT- 130101457, and Project LP-140100569. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. K. Selcuk Candan. *(Yuan Gao and Miaojing Shi contributed equally to this work.)*

Y. Gao, M. Shi, and C. Xu are with the Key Laboratory of Machine Perception (Ministry of Education), Peking University, Beijing 100871, China (e-mail: yuangaopkucis@gmail.con; shimj@cis.pku.edu.cn; xuchao@cis.pku.edu.cn).

D. Tao is with the Centre for Quantum Computation and Intelligent Systems and the Faculty of Engineering and Information Technology, University of Technology, Sydney NSW 2007, Australia (e-mail: dacheng.tao@uts.edu.au).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2015.2389616

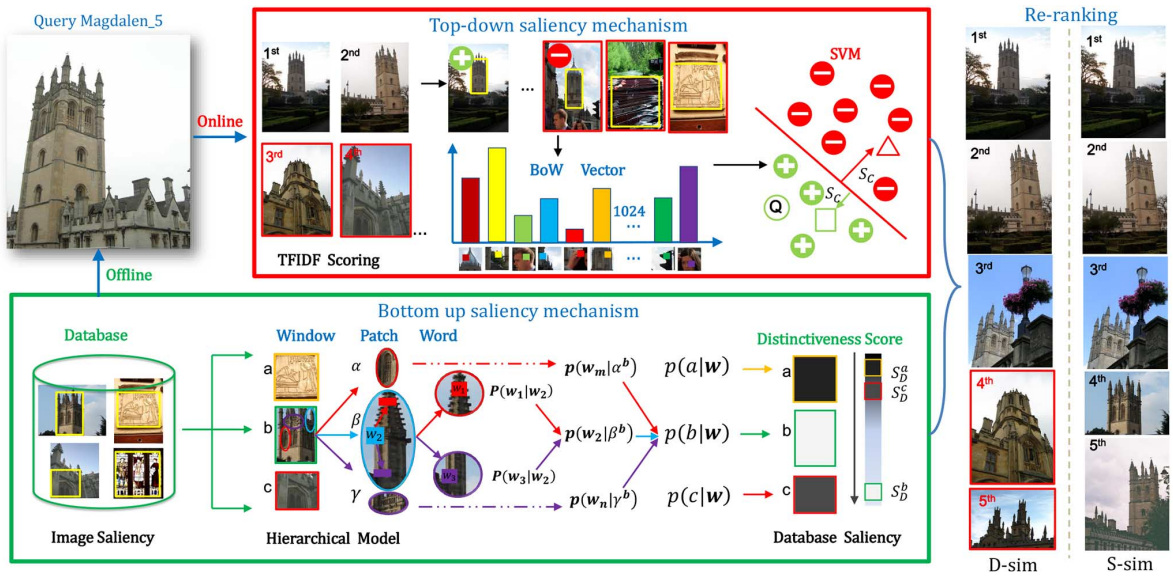

Fig. 1. Overview of the proposed saliency-inspired fast image retrieval scheme. In the offline stage, image saliency detection is carried out first in database images (yellow frames); database saliency values for a, b and c are computed by formulating posterior probabilities $p(a|\mathbf{w})$, $p(b|\mathbf{w})$ and $p(c|\mathbf{w})$ hierarchically in each salient window; every image is thus associated with a distinctiveness score, i.e., S_D^a, S_D^b and S_D^c, which is illustrated by the brightness of the square in the grey bar. In the online stage, given a query image, we first obtain its initial search results using TFIDF scoring; positive (green "+") and negative (red "-") samples are selected from the top and bottom of the ranked list, respectively; every sample is represented by a 1024-dimensional BoW vector describing the salient window (yellow frame) in it; an SVM is learnt on the positive and negative samples to compute the distance S_C from decision boundary. Initial search results are re-ranked by either S_D (D-sim) or both S_D and S_C (S-sim). False alarms are marked with red boxes.

visual cue (shape), but is different in detail given that the bottom-up saliency is not true saliency; we only employ the concept of saliency in the database rather than the visual cues conventionally used in image saliency.

Fig. 1 explains the proposed approach: the bottom-up saliency mechanism is carried out offline, while the top-down saliency mechanism is evaluated at the query time. In the offline stage (green block), we make use of image saliency, which is different from the proposed database saliency, to detect the salient window in each database image as its representative [14]. We compute the database saliency value of each salient window (e.g., window b) using the visual word co-occurrence matrix [15], [29]: a posterior probability (e.g., $p(b|\mathbf{w})$, $\mathbf{w}$ denotes the visual word set inside b.) is calculated in a hierarchical model, namely salient window, local patch, and visual word (feature). Visual word concurrent information is (e.g., $p(w_1|w_2)$) is propagated bottom-up to local patches by multiplying them together (e.g., $p(w_1|\beta^b)$), while the conditional probabilities on the patch level are further aggregated to produce an estimation of window distinctiveness, $p(b|\mathbf{w})$. Every salient window is thus associated with a distinctiveness weight S_D in the database, which is illustrated by the brightness of the square, the larger S_D is, the brighter it is located in the grey bar. At query time (red block), we evaluate a top-down discriminative query expansion by first conducting TFIDF scoring. False alarms are marked with red boxes. Salient windows from top-ranked images of initial returned list are used as positive samples, while negative samples are selected from the bottom of the ranked list. A weight vector is discriminatively learnt via a very low-dimensional linear SVM. Each candidate in the searched list is thus associated with an online weight S_C that depends on the distance from the decision boundary. The initial search results are accordingly re-ranked (denoted by S-sim) by the learnt weight S_C in SVM together with the distinctiveness weights S_D in the database saliency.

II. Related Work

This section first reviews the literature in image retrieval in two aspects: 1) visual ranking by exploiting distinctiveness of images, and 2) query expansion from initial returned list; afterwards, it details the comparison with one of the closest work to this paper, discriminative query expansion [5].

Visual Ranking. Exploiting the distinctiveness of visual images in the database is an actively researched topic [9], [16]–[20]. In [16], [17], only the query's discrimination was measured, and the performance was limited; in [18]–[20], [9], specific similarity measures were learnt and significant improvements were obtained by taking into account the neighborhood of the image space, e.g., k-nearest neighbors (k-NN). The only unsatisfactory aspect of these works is the ranking time: the whole database has to be ranked several times for each query as a result of its k-NN [9], [19], [20].

To tackle this computational concern, we exploit the visual image's distinctiveness from a different perspective: each image is represented by its most salient window inside [14]. A posterior probability of the salient window is calculated to stand for the database saliency value of the area. We derive a formula to approximate the posterior probability. It is based on a hierarchical model, which separates a salient window into several small patches [21], and then into local features [22]. The hierarchical model includes the spatial information implicitly, and is widely exploited in [23]–[26]. The basic element in the model is the co-occurring conditional probability of visual words [15], [27]–[29], it is propagated partially similar to the

Dirichlet process in [24]. In Section III the discussion, we show that the proposed database saliency indeed shares the same intuition with [18], [9] by querying every image with the rest of the database. Except that in this work, we do not really query each image, but propose an offline unsupervised manner, it saves unnecessary computation in the online stage.

Query Expansion. Cosine similarity based visual ranking is not accurate, retrieval performance can be further boosted by expanding the query from the initial returned list. Typical query expansion schemes include average query expansion (AQE) [30], [4], discriminative query expansion (DQE) [5], and visual query suggestion [31], [32]. This paper follows the architecture of [5] by learning an SVM from the top-ranked images of the initial returned list. It is not a novel idea, and many works have re-ranked images by either their classified labels [11], [7], [10] or their distances to the decision boundary [5].

It has been observed that [4], if the top-ranked images contain enough true positives, the re-ranking results of the learnt SVM are significantly better than the initial search results; conversely, if no correctly retrieved images are in the top-ranked list, or very few, the learnt SVM does not help. To enhance the learning performance, active learning [33], relevance feedback [6], [7], spatial verification [1], [4], or text/meta features [10], [11] are added to refine the positive samples. Despite the encouraging improvements they achieve, these techniques are time-consuming for online search.

Comparison With DQE. DQE [5] learns the weight vector discriminatively from the spatially verified BoW vectors. Spatial verification is performed to refine the positive samples from the top-ranked images, and n_f-dimensional BoW vectors (n_f is the number of feature dimensions of BoW vectors, it can be reduced less than 10k in the end) are utilized to represent images and input to a linear SVM. Accordingly, the overhead of gathering negative training data and training the linear SVM is $30ms$ on average on a 3 GHz single core machine. Notwithstanding, considering the time of gathering positive training data, which is indeed the spatial verification time, the entire overhead is still not efficient for online search.

By embedding database saliency into this architecture, it is computed offline to re-weight the discriminative learning, spatial verification is consequently no longer indispensable in SVM. Instead, the detection results of image saliency are used to estimate the location (ROI) of the queried objects in the retrieved images. Compared to spatial verification in DQE, image saliency detection is not accurate, however, is carried out offline and further advantageously utilized in a very low-dimensional (1k) linear SVM, which allows us to achieve significant increase in speed with at least comparable accuracy to state-of-the-art BoW-based image retrieval schemes.

III. Database Saliency

Image Saliency. Generally, each image is specifically represented by the window area with the highest saliency in it, which is the most distinctive part of an image and can be taken to calculate its distinctiveness in the database. For image of multiple landmarks, we could use the entire image to calculate the online weight S_C and distinctiveness weight S_D for re-ranking. In spite of this, in most standard retrieval benchmarks, e.g, Oxford and Paris datasets, each image is mainly composed of one landmark, and single salient window detection works best in image retrieval and is adopted by default in the following section. We use an effective saliency detection method [14] to detect the salient area of an image. A sliding window-based paradigm is used in [14] and window saliency is optimized according to a specifically defined window composition cost function which encodes different visual cues such as appearance, position, and size in the window.

Image saliency is arranged in re-ranking instead of ranking because its detection is not precise enough to locate the queried object in each image, several objects or parts of objects might be included in the salient area that can lower the recall due to ignoring some target objects in the initial ranking. However, in re-ranking, the ignored objects are just moved down on the retrieval list rather than removed from the list. In general, it is observed that, retrieving a specific salient region rather than the whole image effectively helps to prevent negative visual words being brought into SVM in query expansion and therefore avoids a mismatch with the background object in a natural image; SVM generalizes better on the top-ranked salient windows than on the top-ranked images.

Bottom-Up Database Saliency. This section computes the database saliency value of each image. Rather than using visual cues, such as appearance, position, and size in image saliency, we are more concerned with the statistical distinctiveness of an image, as we claim that query-relevant images in the database are more likely to be distinctive than ambiguous. To measure the distinctiveness, we make use of visual word co-occurrence matrix [15], [29], and propagate the visual word concurrent information by simply formulating a posterior probability in each salient window.

What property of the posterior probability can be used for image retrieval? Given all features in the salient window area, we wish that these features denote the area completely and uniquely, that is, the posterior probability of the window W, given all features $\mathbf{f}$, should be, $p(W|\mathbf{f}) = 1$. Similar to the word "and", when we get three letters "a, d, n", we can definitely determine that they denote unique "and". So the larger $p(W|\mathbf{f})$ is, the better the completeness and uniqueness of $\mathbf{f}$ is, and we gain more confidence from the group of features $\mathbf{f}$, and can rank the salient window W higher.

The computation of $p(W|\mathbf{f})$ is not easy. First, features $\mathbf{f}$ have to be quantized into visual words $\mathbf{w}$. Thus, $p(W|\mathbf{f})$ needs to be changed to $p(W|\mathbf{w})$. Then, the key point is the representation of W, $\mathbf{w}$ is just a group of visual words in a region. We propose a hierarchical model, W is separated into a group of affine invariant patches R_r^W, and R_r^W is partitioned into a group of visual words $w_k^{R_r^W}, k = 1, 2, \ldots s$. $p(W|\mathbf{w})$ is decomposed along the hierarchical model, while image's distinctiveness is thus calculated in a bottom-up fashion. Supposing each visual word is approximately independent of other visual words in a window, we have [34]

$$p(W|\mathbf{w}) \propto \prod\nolimits_i p(w_i|W). \tag{1}$$

Since each visual object is perceptually decomposed into a set of generic components, likewise, $p(w_i|W)$ is mathematically

decomposed into amplified scale-invariant local patches R_r^W. Each visual word w_i can occur in a small number of patches, and all the other $p(w_i|R_r^W)$ are zeros

$$p(w_i|W) = p(w_i|R_1^W \cup \ldots \cup R_l^W) \simeq \sum\nolimits_r p(w_i|R_r^W), \quad (2)$$

we drop the joint term in (2) for better performance in practical implementation. Each R_r^W is composed of the visual words $w^{R_r^W}$ lying in patch R_r and belonging to window W

$$p(w_i|R_r^W) = P(w_i|w_1^{R_r^W}, \ldots, w_s^{R_r^W}). \quad (3)$$

Without considering the mutual dependencies between $w_1^{R_r^W}, \ldots, w_s^{R_r^W}$ in the patch, similar to (1), we have

$$p(w_i|R_r^W) \propto \prod\nolimits_k p(w_k^{R_r^W}|w_i). \quad (4)$$

Substituting (2)(3)(4) into (1), we approximate $p(W|\mathbf{w})$ to

$$p(W|\mathbf{w}) \propto \prod\nolimits_i \sum\nolimits_r \prod\nolimits_k p(w_k^{R_r^W}|w_i), \quad (5)$$

where $p(w_k|w_i)$ (superscript omitted in w_k) is a special conditional probability, w_i and w_k must appear in the same affine invariant patch, named the co-occurring conditional probability of visual words [15], [29]. $p(w_k|w_i)$ can be obtained from the statistics of the database. Usually, $p(w_k^{R_r^W}|w_i) \ll 1$, it is difficult to calculate (5), we take a logarithm to the right hand side of it and define a visual object's distinctiveness weight S_D as

$$S_D = \sum\nolimits_i \log \sum\nolimits_r \prod\nolimits_k p(w_k^{R_r^W}|w_i). \quad (6)$$

S_D is independent of the query and related to $p(W|\mathbf{w})$. We take S_D obtained for the salient window as the saliency of that image in the database. For images with multiple landmarks or smaller objects, S_D is directly calculated in the entire image.

Variant. Several assumptions and simplifications are made in (1) $\sim$ (6). In the Appendix, we present a full formulation of database saliency, denoted by database saliency variant (S_D-V). We show that the difference between S_D and S_D-V is simply a matter of window or patch prior simplification. Both the database saliency value and its variant keep the essence of the posterior probability $p(W|\mathbf{w})$: indicating the uniqueness and completeness of the salient window and its features, to help re-rank corresponding images.

Discussion. Through (6), we analyze the proposed distinctiveness measurement. Imagine every visual image (supposing it is represented by one salient window) in the database is taken as a query: the candidate relevant images of each query are thus retrieved via the inverted file index in the BoW model. Every visual word in the query has an entry in the inverted file followed by a list of all the visual images in the database in which the word occurs. To simplify this, we assume the same list length Len of candidate images for each visual word. Suppose S_D is normalized to $[-1, 1]$ over the entire database. To calculate S_D, we assign a very small value to $p(w_k^{R_r^W}|w_i)$ when it is zero.

1. If any two visual words occur independently in different images, all the $p(w_k^{R_r^W}|w_i)$ in (6) are zeros, and $S_D = -1$. Each retrieved image shares only one common word with the query, and the number of whole candidates is $M \times Len$, where M is the visual word number in the query. The query is ambiguous and retrieves many candidates in the database which are not really relevant.

2. If there two visual words w_1 and w_2 that always occur together and all the other words independently occur, the conditional probability $p(w_1^{R_r^W}|w_2)$ (or $p(w_2^{R_r^W}|w_1)$) is 1, while all the other $p(w_k^{R_r^W}|w_i)$ are zeros. S_D is thereby larger than -1, and the candidate lists of w_1 and w_2 are the same in the inverted file because of their co-occurrence. The number of all candidates retrieved by the query is $(M-1) \times Len$, Len of them share two common words w_1, w_2 with query.

3. If any of the two visual words occur together, their co-occurring probability is 1: $p(w_k^{R_r^W}|w_i) = 1$, S_D is therefore the largest, $S_D = 1$. The number of candidates is Len, and they all share M common words with the query. The query is very distinctive, and retrieves a small number of candidates from the database which are very likely to be relevant.

The observations apply to more generalized case as well: when $0 < p(w_k^{R_r^W}|w_i) < 1$ and the list length of candidates for each visual word is not the same in the inverted file, S_D therefore belongs to $(-1, 1)$ and database saliency endeavors to relate images's distinctiveness with the number of its retrieved images and the common words of those images shared with the query. Larger distinctiveness weights are supposed to be associated with those images, when we query them in the database, which have fewer candidates yet share more common words with the retrieved candidates.

We represent each image's distinctiveness weight S_D with a grey square, as illustrated in Fig. 2, each grey square beneath the image represents the distinctiveness weight of its salient window, the larger S_D is, the brighter the square is. We choose two query landmarks, *All_souls* and *Bodleian*, in Oxford dataset and obtain their top-ranked results using TFIDF scoring. False alarms are marked with red boxes, and their distinctiveness

Fig. 2. Illustration of the top-ranked retrieval results for two queries on Oxford dataset. The two rows of search results correspond to the initial TFIDF scoring and D-sim re-ranking results. Each grey square beneath the image represents its distinctiveness weight in the context of brightness. False alarms are marked with red boxes.

weights are of smaller values and darker squares, while relevant images have brighter squares and larger distinctiveness weights. If we re-rank the initial search results using S_D to weight the cosine similarity, the performance (D-sim) is improved as shown in Fig. 1 and Fig. 2. The result is consistent with our assumption that a query's relevant images are more likely to be distinctive than ambiguous in the database.

IV. Online Query Expansion

Query Expansion. We follow the architecture of DQE [5]: an SVM weight vector is discriminatively learnt from a shortlist (the initial search results). Top-ranked images are usually utilized as positive samples n_+, while negative samples n_- are collected from the bottom of a ranked list, because the probability of finding relevant images in the bottom of ranked list is very low.

Each salient window is described by the frequency histogram of visual words obtained by assigning each SIFT descriptor to its closest visual word. The visual word vocabulary size is 1024 for fast online query expansion, because it is able to describe the salient window appropriately without incurring much computational complexity [24]. Every salient window is thus represented by a BoW descriptor vector with the entries $(x_1, x_2, \ldots, x_{1024})$ being the visual word frequency histograms. Note that, except for the small vocabulary used for learning SVM, the vocabulary size used for visual word co-occurrence matrix construction and TFIDF scoring is usually large and reaches 1M.

Given an image described by its salient window (x, y), x is the window descriptor vector and y is its label. $y = +1$ for the positive sample, otherwise $y = -1$. The same as that in DQE, we train a linear SVM using these positive and negative BoW vectors to obtain a weight vector ω. The learnt weight vector is used to calculate image distance from the decision boundary $\omega^T x$. We denote by S_C the result of $\omega^T x$.

We name the sum of S_C and S_D as the saliency-inspired weighting score, S, of which S_C is a measurement of the detected salient window's relevance to the query while S_D is a measurement of the salient window's distinctiveness in the database. The larger S_C is, the more relevant the salient window is to the query; the smaller S_C is, the more irrelevant the salient window is to the query window; despite that S_D is calculated independently of the query, the larger it is, the more relevant to the query the salient window is regarded as being. Therefore, $S = S_C + S_D$ relies on both online and offline contributions, can be either positive or negative. For database images contain a number of scene types rather than single landmark, S_C and S_D are computed over the entire image instead of a single salient window, we have tried to utilize multiple salient windows and sum their weights together in one image, unfortunately, we couldn't find sufficient benefits.

Ranking images are sorted on the value S as well as the cosine similarity value sim in TFIDF scoring. We therefore summarize our saliency-inspired ranking function S-sim as

$$S - sim = sim + \beta min(0, S) \tag{7}$$

where the parameter β is introduced to adjust the contribution of S in S-sim. We slightly modify (7): sim is weighted by S when it is negative, otherwise we set S to zero. We find an interesting phenomenon in the real implementation: we are more inclined to remove irrelevant images by subtracting a large value in the similarity measure [9]; by contrast, adding a positive term to up-weight the relevant images usually does not work, or the improvement is negligible. It suggests that removing irrelevant images is much easier, since the majority images in the database are irrelevant to certain query.

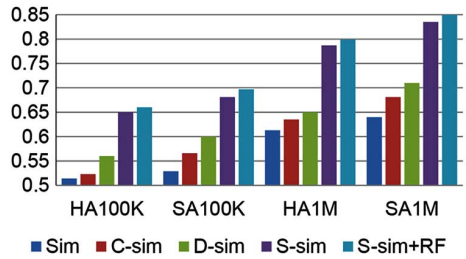

Fig. 3. mAP results for visual object re-ranking in multiple schemes on Oxford dataset. Sim denotes the original cosine similarity measure, C-sim, D-sim, and S-sim denote that Sim is respectively weighted by S_C, S_D, and S. RF means adding user feedback. HA and SA denote hard- and soft-assigned vocabularies.

Discussion. Though neither S_C nor S_D is accurate enough, the corresponding distinctiveness weight and the online query expansion weight have complementary effect as illustrated in Fig. 3, the combination of S_C and S_D guarantees a reliable estimation of an image's relevance to a query.

We interpret this from a probabilistic view: in [35], Platt proposed to approximate the posterior $p(y = 1|x, \theta)$ by a sigmoid function via the SVM output S_C

$$p(y = 1|x, \theta) \approx \frac{1}{1 + \exp(A\theta(x) + B)},$$
$$\theta(x) = S_C = \omega^T x. \tag{8}$$

The parameters (A, B) are found by minimizing the negative log likelihood of training samples referring to [35]. The SVM prior θ is proved to be simply a Gaussian process (GP) over ω [36], ω is determined by the distribution of trainings. Thus, S_C indeed discloses the likelihood value that an image is drawn from the distribution of either positive or negative samples. On the other hand, S_D is related to $p(W|\mathrm{w})$, which describes the distinctiveness of an image drawn from the entire database. sim can also be interpreted as a probability in language model [37], if we multiply the three probabilities together (it is not as good as the sum in (7) in practice), it is indeed a joint probability of the three different manners of relevance estimation.

Regarding S_C and S_D, they are both some manner of measurements underlying the visual distribution inside certain image, and in two aspects they complement, 1) S_C computes the probability $p(y = 1|x)$ from the object view, while S_D decomposes $p(W|w)$ into local patches; 2) S_C measures a similarity score between certain image and images from top-ranked list, while S_D measures a dissimilarity score between an image and images over the entire database. Recall the claim we make, the number of relevant objects in the database for a given query, regardless of its identity, is extremely small compared to the entire image collection. In complement of S_D and S_C, we select a small group of images distinctive to the entire

database and refine the selection in consistent with distribution generalized from the top-ranked images in TFIDF scoring.

Computation Cost. The proposed S-sim is similar to DQE, but much faster. The time complexity of DQE is

$$T_{DQE} = O(SP) + O(SVM) + O(QE) \tag{9}$$

where the first term corresponds to the time complexity of spatial verification before expanding the query. Candidate images in the initial returned list are verified via RANSAC

$$O(SP) = O(L_{RANSAC} k(t_m + n_c t_c)) \tag{10}$$

L_{RANSAC} denotes the length of the verified list. For each candidate image in the list, $O(k(t_m + n_c t_c))$ is the time complexity of RANSAC, where t_m is the time cost of computing a single model, t_c is the time cost of fitting one point, n_c is the number of feature points, and k is the iteration. The second term $O(SVM)$ is the computational complexity for training an SVM [38]

$$O(SVM) = O(n_s^3 + n_s^2 L_{SVM} + n_s n_f L_{SVM}) \tag{11}$$

where n_s is the number of support vectors, n_f is the number of feature dimensions of BoW vectors, and L_{SVM} is the size of the training set. The third term $O(QE)$, the computational complexity for query expansion, is the computational cost of testing SVM

$$O(QE) = n_s n_f \cdot (\otimes + \oplus) \tag{12}$$

$\otimes$ and $\oplus$ denote a multiplication and addition of two real values, respectively [39].

Compared to DQE, the time complexity of S-sim is

$$T_{S-sim} = \widetilde{O}(SVM) + \widetilde{O}(QE) \tag{13}$$

where $\widetilde{O}(SVM) = O(\tilde{n}_s^3 + \tilde{n}_s^2 L_{SVM} + \tilde{n}_s \tilde{n}_f L_{SVM})$ and $\widetilde{O}(QE) = \tilde{n}_s \tilde{n}_f (\otimes + \oplus)$. Instead of using spatial verification, we adopt offline image saliency detection to roughly locate the salient window in an image, and a saving in computation is achieved. In addition, a very low dimensional BoW vector is used in this paper, $\tilde{n}_f < n_f$. The dimensionality is 1k compared to around 10k used in DQE after truncating the BoW vectors [5]; in other words, $T_{S-sim} \ll T_{DQE}$, our method runs much faster than DQE.

V. Experimental Results

A. Dataset and Evaluation Protocol

Oxford5k [1]. This dataset of 5062 images is a standard image retrieval test set, which we call Ox for short. 55 images of 11 Oxford landmarks are selected as the query images, and their ground truth retrieval results are provided.

Paris6k [9] contains 6390 images by querying the associated text tags for famous Paris landmarks, such as "Paris Eiffel Tower" or "Paris Arc de Triomphe". Similar to *Oxford5k*, 55 query images are selected from Paris landmarks, and their ground truth are provided as well.

INRIA Holidays [40]. This dataset is a set of images which mainly contains holidays photos. We name it Ho for short. It includes a large variety of scene types (natural, man-made, water and fire effects, etc.).

UKB. The University of Kentucky Benchmark dataset [41] consists of 10200 images grouped into 2550 subsets of corresponding images. Each subset contains four images. For a given query, the system is expected to return the four relevant images in the first four positions.

ImageNet [42]. Approximately 100k and 500k images are randomly sampled from 10M images in ImageNet, which we respectively call I1 and I2 for short. We use I1 and I2 as distractors to implement the test on a large-scale.

Evaluation Protocol. Evaluation of impact of parameters is first conducted on Oxford dataset. SIFT files and visual word vocabularies were downloaded from the Oxford VGG website.[1] Single salient object is extracted in each image, the training ratio and BoW vector's dimension in SVM are respective $10/200(n_+/n_-)$ and 1024 by default, whilst vocabulary size for TFIDF scoring is 1M. To evaluate the performance on large-scale, we add $100k$ and $500k$ images from ImageNet dataset as distractors. In this case, the evaluation is performed on the 55 Oxford queries, since the images from ImageNet are not relevant to queries. Performances on Paris, Holidays and UKB are presented in the end, of which Holidays and UKB are used to discuss two critical issues in this paper, which are the failure of image saliency detection and database saliency computation.

Overall comparisons are carried out with other representative approaches such as [3], [15], [16], [19], [9], [8], [21], [18], [43], [5], [20], [2]. The baseline follows the architecture of [1]. The performance for Oxford, Paris and Holidays datasets is measured in terms of the average precision (AP), which is defined as the area under the precision-recall curve for each query. The AP score is computed for each query ad averaged to obtain a mean average precision (mAP). For the UKB dataset, the score is standardly computed as the average number of correct images in the top-4 positions (4-recall@4), the best score is 4. Notice that, to make a quick test on UKB, we train a vocabulary from Holidays.

B. Impact of Parameters

Parameter β. Fig. 4(a) shows the mAPs with variation of parameter β. The optimal performances occur when β is smaller than $1(log\beta < 0)$. When $\beta \to 0$, the ranking result is determined by the original cosine similarity sim in (7); when $\beta \to \infty$, the ranking result is determined by the weighting term S. This shows that the initial TFIDF scoring output is necessary as an anchor, so that we re-rank the visual images in the returned list. Note that the selection of β is generally related to the dataset size: for datasets like Oxford and Paris that have the same scale, we can roughly choose the optimal $log\beta$ around $-3 \sim -4$.

Training Ratio. We evaluate different combinations of training samples. We select n_+ positive samples from the top-ranked list, and n_- negative samples from the bottom of the ranked list. Results are given in Table I for various choices of n_+ and n_-. For each choice, we use 10-fold cross validation during the training stage. The improvement achieved

[1][Online] Available: http://www.robots.ox.ac.uk/vgg/data/

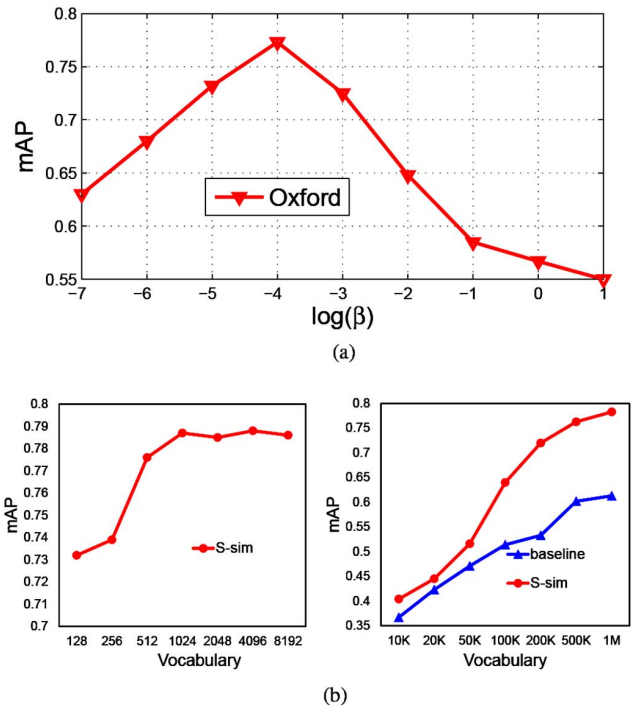

Fig. 4. (a) mAPs for Oxford dataset with different values of parameter $\beta(log\beta)$. (b) Left: mAP values correspond to different hard-assigned vocabulary sizes in SVM. Right: Comparisons of baseline and S-sim with different hard-assigned vocabulary sizes in TFIDF scoring. (a) Parameter variation. (b) Effect of vocabulary size.

TABLE I
MAP VALUES FOR DIFFERENT TRAINING RATIOS n_+/n_- ON THE OXFORD DATASET; HA [1] AND SA [3] DENOTE HARD- AND SOFT-ASSIGNED VOCABULARIES, RESPECTIVELY

n_+/n_-	Oxford(HA)	Oxford(SA)
5/200	0.784	0.803
10/200	**0.787**	**0.835**
20/200	0.787	0.820
40/200	0.766	0.820
80/200	0.761	0.810
baseline	0.614	0.640

by the proposed S-sim over the baseline [1] is clear in Table I. Increasing n_+ tends to include more irrelevant samples into positive samples in SVM, which biases the online learning. Generally, we consider that top $10n_+$ is a reliable choice. The choice of n_- hardly affects mAP in a range of $100 \sim 1000$. By default, we choose the training ratio as $10/200(0.05)$ for all the queries.

Effect of Vocabulary Size. We evaluate the effectiveness of S-sim in the Oxford dataset for different hard-assigned vocabularies, as shown in Fig. 4(b). The mAPs of baselines and the corresponding improvements obtained by S-sim are shown by the blue line and red line, respectively.

The left figure shows that when we fix the vocabulary size for co-occurrence matrix construction and TFIDF scoring to 1M, and change the vocabulary size for learning SVM from 128 to 8k, the mAP value does not change significantly. Due to the polysemy in all the small visual vocabularies [21], the $8k$ vocabulary does not effectively outperform the 512 vocabulary. In real implementation, we choose the vocabulary size as 1024 for SVM. The same mAP can also be obtained by using high-dimensional SVM and spatial verification as in DQE [5], despite the good performance, it is very time-consuming for online search.

TABLE II
LARGE-SCALE EVALUATION. MAP AND AVERAGE TIME OVERHEAD PER QUERY. VOCABULARY SIZE IS 1M USING HARD ASSIGNMENT

Dataset	TFIDF scoring	Avg. time (ms)
Ox+I1	0.566	219.9
Ox+I2	0.499	1036.4
Dataset	S-sim	Avg. time (ms)
Ox+I1	0.661	258.8
Ox+I2	0.630	1101.4

The right figure shows corresponding mAP values when we fix the vocabulary size for SVM to 1024 and change the TFIDF scoring vocabulary size from $10k$ to 1M. For the large vocabulary, significant improvement has been achieved at every point compared to the baseline. However, for the small vocabularies, mAP increments are very trivial, most initial search results are irrelevant (mAPs are lower than 0.5), and the online weight learnt from the top-ranked salient images is not reliable, neither is the distinctiveness weight computed from the co-occurrence matrix. To overcome this dilemma and guarantee efficiency, we suggest adding user feedback and show our results in the following section.

C. Results for Visual Object Re-Ranking

Experiments are carried out on both hard- and soft-assigned (HA [1] and SA [3]) $100k$ and 1M Oxford vocabularies. mAP values are reported accordingly using different weighting techniques.

Fig. 3 provides the re-ranking results (C-sim and D-sim) depending on whether we utilize the query expansion score S_C or the distinctiveness score S_D to weight sim. We can see that, even though S_D is computed offline independently of the query, we can still achieve an improvement by embedding it into the TFIDF scoring scheme. It supports our motivation that a query object's relevant objects are more likely to be distinctive rather than ambiguous. On the other hand, compared with C-sim and D-sim, their combination S-sim apparently yields superior results. It increases mAPs to 0.65 and 0.681 on $100k$ HA and SA vocabularies with 26.5% and 28.7% increments, and to 0.787 and 0.835 on the 1M HA and SA vocabularies with 28.3% and 30.5% increments, by comparing with baselines.

By adding user relevance feedback (RF) to refine the top-ranked salient samples in the discriminative query expansion, the mAP can be further enhanced. The highest mAPs can reach 0.697 and 0.850 on the 100k and 1M soft-assigned vocabularies. Since adding RF is laborious work, we only ask the user to label the relevant salient objects from the top 10 ranked images.

D. Large-Scale Evaluation

To implement the proposed method on a large-scale database, we carry out the experiment by adding the ImageNet (100k and 500k images denoted by I1 and I2) dataset to the Oxford (Ox) datasets. The results are provided in Table II, the mAP improvements are impressive. It also shows the average query time overhead for each query (Note that we don't really rank the entire

TABLE III
MAPS FOR DIFFERENT DATASETS COMPARED TO THE STATE-OF-THE-ARTS. WE PRESENT S-SIM RESULTS BY USING SOFT-ASSIGNMENT VOCABULARIES IN OXFORD AND PARIS DATASETS

Dataset	S-sim	[1]	[15]	[16]	[19]	[9]	[8]	[21]	[18]	[44]	[5]	[2]	[20]
Oxford	0.835	0.614	0.730	0.675	**0.882**	0.814	0.713	0.811	0.764	0.685	0.847	0.696	0.850
Paris	0.814	0.66	0.703		**0.911**	0.803		0.791	0.728		0.818	0.562	0.855
Holidays	0.840				0.762	0.423			0.844	**0.848**			0.844

Note: The mAP 0.847 is reported from Table I in [5] without using RootSIFT and SPAUG, which is with the same setting of S-sim. The mAP 0.818 is obtained from our own implementation.

TABLE IV
TIME ELAPSED IN THE SVM IN DQE AND S-SIM ON OXFORD AND PARIS DATASETS. SP DENOTES SPATIAL VERIFICATION

Dataset	DQE	S-sim
Oxford	$2.375s$(SP) + $22ms$	$19.9ms$
Paris	$2.195s$(SP) + $20ms$	$19.3ms$

dataset, instead, for those images with very small ranking values we ignore them directly.). Since the database saliency is computed offline, the time overhead of S-sim is in the same order of magnitude as that for TFIDF scoring. Additional time is mainly required for SVM training and test, which does not necessarily grow up with the database size, because we choose the same number of training samples for SVM.

The additional memory overhead is simply to record the database saliency value for each image. They are 821KB and 3946KB for Ox + I1 and Ox + I2, respectively. Regarding the offline process of database saliency computation, we need to load the visual word co-occurrence matrix into the memory, as suggested in [15], by setting a limit of the length of the co-occurrence list, we could keep the storage overhead around 500 MB regardless of the database size.

E. Comparisons

Table III shows the comparison of our method with other state-of-the-art approaches [15], [16], [19], [9], [8], [21], [18], [43], [5], [2], [20] in Oxford, Paris and Holidays datasets. Most of these methods employ additional techniques such as soft (multiple) assignment [15], [43], [21], geometric verification [21], [43], [5], [8] and k-NN[2] re-ranking [20], [19], our results (S-sim) are competitive among them.

Specifically, we compare our work with one of the closest works, DQE [5], in Table IV. Time of gathering negative training data and training the linear SVM reported in [5] is $30ms$ on Oxford dataset on a 3 GHz single core machine. To make a fair comparison, we implement it on a 2.4 GHz dual-core machine as the same with S-sim. Time elapsed includes SVM training and test. We also report the spatial verification time in DQE which is not counted in [5]. Although the mAP of S-sim is slightly lower than DQE, its speedup of time efficiency over DQE is impressive.

More importantly, we propose a new concept database saliency, it could be either adopted independently (Fig. 1, D-sim) or plugged into standard retrieval architecture (Fig. 1, S-sim). Unlike other representative approaches, i.e., [19], [20], we re-rank the whole dataset at most once, which is much more efficient.

[2]k-NN re-ranking is very time-consuming. Accordingly, the mAPs of [19], [20]without k-NN re-ranking are respective 0.752 and 0.780 on Oxford dataset, 0.741 and 0.736 on Paris dataset, which are inferior to S-sim.

TABLE V
S-SIM RESULTS ARE REPORTED IN TERMS OF MAP AND 4-RECALL@4 FOR OXFORD, PARIS, HOLIDAYS, AND UKB, ACCORDINGLY. "+" REFERS TO THE USAGE OF SALIENT WINDOW (SW FOR SHORT), WHILE "$\sim$" INDICATES USING THE ENTIRE IMAGE FOR SVM TRAINING AND DATABASE SALIENCY COMPUTATION, "$*$" MEANS NO VALUE. 1M HARD-ASSIGNED VOCABULARIES ARE USED

SW	Oxford	Paris	Holidays	UKB
+	0.787	0.763	$*$	3.24
$\sim$	0.720	0.755	0.840	3.24
Baseline	0.614	0.660	0.816	3.00

TABLE VI
LARGE-SCALE EVALUATION ON HO + I1 AND UKB + I1 USING 1M HA VOCABULARY. HO + I1 AND UKB + I1 SIGNIFY HOLIDAYS AND UKB DATASETS PLUS 100K IMAGES FROM IMAGENET. MAP AND 4-RECALL@4 ARE REPORTED, RESPECTIVELY

Method	Ho + I1	UKB + I1
TFIDF scoring	0.673	2.86
S-sim	0.743	3.17

F. Discussion

In this section, we discuss two critical issues in this paper: 1) image saliency detection fails to detect the queried object in database image, and 2) database saliency fails to distinguish relevant images as there exist no noisy images in the database. We add two datasets, Holidays and UKB, in the test.

In Holidays dataset, images include a number of scene types rather than simply the building landmarks, image saliency detection could possibly detect very smooth or flat area with the highest salience, i.e., sky, water and fire effects. Few features can be extracted in this area to represent the image and for sure the queried object is not in this salient window. We can not train an SVM or compute database saliency from such a salient window. To tackle this problem, we propose to use the entire image to compute the database saliency and re-rank. Table V reports the corresponding results. It is worth of remarking that, 1) we conduct the same test by using multiple salient windows in one image, we find that increasing the number of salient windows does not clearly affect or improve the mAP, and 2) in Oxford, Paris and UKB, single salient window is capable of detecting the queried object and works best.

Observing that the mAP (4-recall@4) improvement over baseline on Holidays and UKB is not as impressive as that on Oxford and Paris. This is because, given any image in Holidays and UKB, it is relevant to certain query in the benchmark test and should be treated as distinctive, the proposed database saliency is thereby no longer rational. To address this issue,

we augment the database by adding distractors from 100k ImageNet dataset. Table VI shows that S-sim is much more effective in this context.

VI. Conclusion

In this paper, we have demonstrated that a significant improvement in visual image re-ranking can be achieved by exploiting saliency in the database. We measure the distinctiveness of each visual image in the database. The distinctiveness score is calculated in a hierarchical manner in the salient window of each image. By taking advantage of this database saliency, we propose a saliency-inspired fast image retrieval scheme: salient windows from top-ranked images are taken as positive samples after the initial search, a very low-dimensional linear SVM is discriminatively learnt for online query expansion. The initial returned list is re-ranked according to the distinctiveness weight and online weight, and experimental results on several standard benchmarks prove the efficiency and effectiveness of the proposed scheme.

The image salience utilized in this paper is only a rough detection of the queried object in each image. In future work, we could run some structure-from-motion tools, i.e.,[44], [45], to discover semantic-level objects and precisely identify important visual words and their concurrent combinations associated with each semantic objects.

Appendix

Database Saliency Variant. This appendix details the variant mentioned in Section III. It provides a full formulation of (1)–(6) without simplifications

$$p(W|\mathbf{w}) = p(\mathbf{w}|W)\frac{p(W)}{p(\mathbf{w})} = p(W)\prod\nolimits_i \frac{p(w_i|W)}{p(w_i)}. \quad (14)$$

Similar to (2) and (3), each visual word w_i can occur in a small number of patches R_r^W, while each R_r^W is composed of a group of visual words $w^{R_r^W}$,

$$p(w_i|W) = \sum\nolimits_r p(w_i|R_r^W), \quad (15)$$

$$p(w_i|R_r^W) = p(w_i|w_1^{R_r^W}, \ldots, w_s^{R_r^W}). \quad (16)$$

Similar to (14), we can write (16) as

$$\begin{aligned} p(w_i|R_r^W) &= p(R_r^W|w_i)\frac{p(w_i)}{p(R_r^W)} \\ &= \frac{p(w_i)}{p(R_r^W)}\prod\nolimits_k p(w_k^{R_r^W}|w_i). \end{aligned} \quad (17)$$

Substituting (17) and (16) into (15) and (14), we have

$$\begin{aligned} p(w_i|W) &= \sum\nolimits_r \frac{p(w_i)}{p(R_r^W)}\prod\nolimits_k p(w_k^{R_r^W}|w_i) \\ &= p(w_i)\sum\nolimits_r \frac{\prod_k p(w_k^{R_r^W}|w_i)}{p(R_r^W)} \end{aligned} \quad (18)$$

$$p(W|w) = p(W)\prod\nolimits_i \sum\nolimits_r \frac{\prod_k p(w_k^{R_r^W}|w_i)}{p(R_r^W)} \quad (19)$$

Comparing (19) with (5), the difference simply lies on the estimation of the window and patch priors, $p(W)$ and $p(R_r^W)$. Basically, we have no prior idea of the distinctiveness of any window or patch in the database, we assume them to be the same over the database, $p(W) = p(W)$, $p(R) = p(R_r^W)$, and similar to (6), we take a logarithm to the right hand side of (19) and define the database saliency variant S_D-V as

$$\begin{aligned} S_D\text{-}V &= \log p(W) + \sum\nolimits_i \log \frac{\sum_r \prod_k p(w_k^{R_r^W}|w_i)}{p(R)} \\ &= \log p(W) \\ &\quad + \sum\nolimits_i (\log \sum\nolimits_r \prod\nolimits_k p(w_k^{R_r^W}|w_i) - \log p(R)) \\ &= \sum\nolimits_i \log \sum\nolimits_r \prod\nolimits_k p(w_k^{R_r^W}|w_i) \\ &\quad + \log p(W) - n_{\mathrm{W}} \log p(R) \\ &= \sum\nolimits_i \log \sum\nolimits_r \prod\nolimits_k p(w_k^{R_r^W}|w_i) + \log \frac{p(W)}{p(R)^{n_{\mathrm{W}}}} \end{aligned} \quad (20)$$

where $\sum_i = n_{\mathrm{W}}$. Referring to (6), S_D can be regarded as a simplification of (20) without considering the second term, which only varies with the number of visual words n_{W} in window W.

We make a toy test of (20): given that we know the number of noise images in a database is N_{n}, the number of distinctive images N_{d} is then given by $N_{\mathrm{d}} = N - N_{\mathrm{n}}$, where N is the image collection size, we propose to approximate $p(W)$ by $p(W) = \frac{N_{\mathrm{d}}}{N}$, and $p(R)$ by $p(R) = \sum_{i=1}^{N_d} n_{\mathrm{r}}^i / \sum_{i=1}^{N} n_{\mathrm{r}}^i$, by supposing that each image is only represented by one salient window, each window is composed of n_{r}^i patches, and each patch is associated with exactly one visual feature. We conduct the experiment on the Oxford and Paris datasets using 1M HA vocabulary, the mAPs are 0.656 and 0.704, respectively. Although it is less effective than S-sim, we still obtain benefits in this toy example.

References

[1] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2007, pp. 1–8.

[2] L. Zheng, S. Wang, Z. Liu, and Q. Tian, "Lp-norm IDF for large scale image search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 1626–1633.

[3] 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. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2008, pp. 1–8.

[4] O. Chum, A. Mikulik, M. Perdoch, and J. Matas, "Total recall II: Query expansion revisited," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2011, pp. 889–896 .

[5] R. Arandjelovic and A. Zisserman, "Three things everyone should know to improve object retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2012, pp. 2911–2918.

[6] A. Kovashka, D. Parikh, and K. Grauman, "Whittlesearch: Image search with relative attribute feedback," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2012, pp. 2973–2980.

[7] V. Jain and M. Varma, "Learning to re-rank: Query-dependent image re-ranking using click data," in *Proc. Int. Conf. Word Wide Web*, 2011, pp. 277–286.

[8] Y. Zhang, Z. Jia, and T. Chen, "Image retrieval with geometry-preserving visual phrases," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2011, pp. 809–816.
[9] D. Qin, S. Gammeter, L. Bossard, T. Quack, and L. Van Gool, "Hello neighbor: Accurate object retrieval with k-reciprocal nearest neighbors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2011, pp. 777–784.
[10] J. Krapac, M. Allan, J. Verbeek, and F. Juried, "Improving web image search results using query-relative classifiers," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2010, pp. 1094–1101.
[11] F. Schroff, A. Criminisi, and A. Zisserman, "Harvesting image databases from the web," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 4, pp. 754–766, Apr. 2011.
[12] L. Chen, D. Xu, I. W. Tsang, and J. Luo, "Tag-based image retrieval improved by augmented features and group-based refinement," *IEEE Trans. Multimedia*, vol. 14, no. 4, pt. 1, pp. 1057–1067, Aug. 2012.
[13] F. Shahbaz Khan, J. van de Weijer, and M. Vanrell, "Top-down color attention for object recognition," in *Proc. Int. Conf. Comput. Vis.*, 2009, pp. 979–986.
[14] J. Feng, Y. Wei, L. Tao, C. Zhang, and J. Sun, "Salient object detection by composition," in *Proc. Int. Conf. Comput. Vis.*, 2011, pp. 1028–1035.
[15] M. Shi, X. Sun, D. Tao, and C. Xu, "Exploiting visual word co-occurrence for image retrieval," in *Proc. ACM Int. Conf. Multimedia*, 2012, pp. 69–78.
[16] Y. Li, B. Geng, D. Tao, Z.-J. Zha, L. Yang, and C. Xu, "Difficulty guided image retrieval using linear multiple feature embedding," *IEEE Trans. Multimedia*, vol. 14, no. 6, pp. 1618–1630, Dec. 2012.
[17] X. Tian, Y. Lu, and L. Yang, "Query difficulty prediction for web image search," *IEEE Trans. Multimedia*, vol. 14, no. 4, pp. 951–962, Aug. 2012.
[18] H. Jégou *et al.*, "Exploiting descriptor distances for precise image search," INRIA, Rocquencourt, France, Tech. Rep. RR-7656, 2011.
[19] X. Shen, Z. Lin, J. Brandt, S. Avidan, and Y. Wu, "Object retrieval and localization with spatially-constrained similarity measure and k-NN re-ranking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2012, pp. 3013 –3020.
[20] D. Qin, C. Wengert, and L. Van Gool, "Query adaptive similarity for large scale object retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 1610–1617.
[21] W. Tang, R. Cai, Z. Li, and L. Zhang, "Contextual synonym dictionary for visual object retrieval," in *Proc. ACM Int. Conf. Multimedia*, 2011, pp. 503–512.
[22] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.
[23] J. Deng, A. C. Berg, and L. Fei-Fei, "Hierarchical semantic indexing for large scale image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2011, pp. 785–792.
[24] D. Liu, G. Hua, and T. Chen, "A hierarchical visual model for video object summarization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 12, pp. 2178–2190, Dec. 2010.
[25] N. Verma, D. Mahajan, S. Sellamanickam, and V. Nair, "Learning hierarchical similarity metrics," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2012, pp. 2280 –2287.
[26] H. Zhang, Z.-J. Zha, Y. Yang, S. Yan, Y. Gao, and T.-S. Chua, "Attribute-augmented semantic hierarchy: Towards bridging semantic gap and intention gap in image retrieval," in *Proc. ACM Int. Conf. Multimedia*, 2013, pp. 33–42.
[27] R. Xu, M. Shi, B. Geng, and C. Xu, "Fast visual word assignment via spatial neighborhood boosting," in *Proc. 19th Int. Conf. Multimedia Expo*, Jul. 2011, pp. 262–270.
[28] M. Shi, R. Xu, D. Tao, and C. Xu, "W-tree indexing for fast visual word generation," *IEEE Trans. Image Process.*, vol. 22, no. 3, pp. 1209–1222, Mar. 2013.
[29] M. Shi, X. Sun, D. Tao, C. Xu, B. George, and H. Liu, "Exploring spatial correlation for visual object retrieval," in *ACM Trans. Intell. Syst. Technol.*, 2015, pp. 00:1–00:21.
[30] 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. Int. Conf. Comput. Vis.*, Oct. 2007, pp. 1–8.
[31] Z.-J. Zha, L. Yang, T. Mei, M. Wang, and Z. Wang, "Visual query suggestion," in *Proc. ACM Int. Conf. Multimedia.*, 2009, pp. 5–24.
[32] Z.-J. Zha, L. Yang, T. Mei, M. Wang, Z. Wang, T.-S. Chua, and X.-S. Hua, "Visual query suggestion: Towards capturing user intent in internet image search," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 6, no. 3, p. 13, 2010.
[33] Z.-J. Zha, M. Wang, Y.-T. Zheng, Y. Yang, R. Hong, and T.-S. Chua, "Interactive video indexing with statistical active learning," *IEEE Trans. Multimedia*, vol. 14, no. 1, pp. 17–27, Feb. 2012.
[34] J. Revaud, M. Douze, and C. Schmid, "Correlation-based burstiness for logo retrieval," in *Proc. ACM Int. Conf. Multimedia*, 2012, pp. 965–968.
[35] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *Advances in Large Margin Classifiers*. Cambridge, MA, USA: MIT Press, 1999, pp. 61–74.
[36] P. Sollich, "Bayesian methods for support vector machines: Evidence and predictive class probabilities," *Mach. Learn.*, vol. 46, no. 1–3, pp. 21–52, 2002.
[37] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge, U.K.: Cambridge Univ. Press, 2008, vol. 1.
[38] C. J. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining Knowl. Discovery*, vol. 2, no. 2, pp. 121–167, 1998.
[39] D. Tao, X. Tang, X. Li, and X. Wu, "Asymmetric bagging and random subspace for support vector machines-based relevance feedback in image retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 7, pp. 1088–1099, Jul. 2006.
[40] H. Jegou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 304–317.
[41] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2006, vol. 2, pp. 2161–2168.
[42] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. and Pattern Recog.*, Jun. 2009, pp. 248–255.
[43] H. Jégou, M. Douze, and C. Schmid, "On the burstiness of visual elements," in *Proc. IEEE Conf. Comput. Vis. and Pattern Recog.*, Jun. 2009, pp. 1169–1176.
[44] A. Irschara, C. Zach, J.-M. Frahm, and H. Bischof, "From structure-from-motion point clouds to fast location recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2009, pp. 2599–2606.
[45] T. Sattler, B. Leibe, and L. Kobbelt, "Fast image-based localization using direct 2D-to-3D matching," in *Proc. Int. Conf. Comput. Vis.*, 2011, pp. 667–674.

Yuan Gao received the B.E. degree from Chongqing University of Posts and Telecommunications, Chongqing, China, in 2011, and the M.S. degree from the Key Laboratory of Machine Perception (Ministry of Education), Peking University, Beijing, China, in 2014.

She is currently a Software Developer with LeTV.com, Beijing, China. Her research interests include image retrieval, multimedia computing, and computer vision.

Miaojing Shi received the B.E. degree from Tongji University, Shanghai, China, in 2010, and he is currently working toward the Ph.D. degree with the Key Laboratory of Machine Perception (Ministry of Education), Peking University, Beijing, China.

Between 2012 and 2013, he was with the Department of Physiology, Anatomy and Genetics, University of Oxford, Oxford, U.K., as a Ph.D. student, and between 2014 and 2015, he was a visiting Ph.D. student with the Institut National de Recherche en Informatique et en Automatique, INRIA, Rennes, France. His research interests include multimedia search, image processing, computer vision, and human vision.

Dacheng Tao (A'07–M'07–SM'12–F'15) is Professor of Computer Science with the Centre for Quantum Computation and Intelligent Systems and the Faculty of Engineering and Information Technology, University of Technology, Sydney, Australia. His research results have expounded in one monograph and over 100 publications in prestigious journals and prominent conferences. His research interests include computer vision, data science, image processing, machine learning, neural networks, and video surveillance.

Dr. Tao has won several Best Paper awards, the Best Theory/Algorithm Paper Runner-Up Award at IEEE ICDM'07, the Best Student Paper Award at IEEE ICDM'13, and the 2014 ICDM 10 Year Highest Paper Award.

Chao Xu (M'00) received the B.E. degree from Tsinghua University, Beijing, China, in 1988, the M.S. degree from the University of Science and Technology of China, Hefei, China, in 1991, and the Ph.D. degree from the Institute of Electronics, Chinese Academy of Sciences, Beijing, China, in 1997.

Between 1991 and 1994, he was an Assistant Researcher with the University of Science and Technology of China, Hefei, China. Since 1997, he has been with the National Laboratory on Machine Perception, Peking University, Beijing, China, where he has been Professor since 2005. He has authored or co-authored more than 120 publications and eight patents. His research interests include image and video processing and multimedia technology.