Cascade Category-Aware Visual Search

Shiliang Zhang, Qi Tian, *Senior Member, IEEE*, Qingming Huang, *Senior Member, IEEE*,
Wen Gao, *Fellow, IEEE*, and Yong Rui, *Fellow, IEEE*

***Abstract*— Incorporating image classification into image retrieval system brings many attractive advantages. For instance, the search space can be narrowed down by rejecting images in irrelevant categories of the query. The retrieved images can be more consistent in semantics by indexing and returning images in the relevant categories together. However, due to their different goals on recognition accuracy and retrieval scalability, it is hard to efficiently incorporate most image classification works into large-scale image search. To study this problem, we propose cascade category-aware visual search, which utilizes weak category clue to achieve better retrieval accuracy, efficiency, and memory consumption. To capture the category and visual clues of an image, we first learn category-visual words, which are discriminative and repeatable local features labeled with categories. By identifying category-visual words in database images, we are able to discard noisy local features and extract image visual and category clues, which are hence recorded in a hierarchical index structure. Our retrieval system narrows down the search space by: 1) filtering the noisy local features in query; 2) rejecting irrelevant categories in database; and 3) preforming discriminative visual search in relevant categories. The proposed algorithm is tested on object search, landmark search, and large-scale similar image search on the large-scale LSVRC10 data set. Although the category clue introduced is weak, our algorithm still shows substantial advantages in retrieval accuracy, efficiency, and memory consumption than the state-of-the-art.**

***Index Terms*— Large-scale visual search, image annotation.**

I. Introduction

ATTRACTING lots of research efforts, large-scale image retrieval and image classification have achieved significant progress in recent years. By modeling the spatial configurations among local features and building efficient index structure, recent retrieval systems [10], [12], [17], [19], [35] have shown decent scalability and accuracy. Image classification also has exhibited significant progress by extracting informative image representations [20], modeling contextual relationships among attributes or objects [6], [25], and learning powerful classification models [14], [15], *etc.* Despite of their great success, large-scale image search and image classification largely remain independent research efforts due to their different goals on retrieval scalability and recognition accuracy, respectively. For instance, large-scale image retrieval has to guarantee the scalability in terms of accuracy, computation, and memory consumption. Differently, image classification can take advantage of high-dimensional features [20] or deep learning models [14].

However, it is desirable to incorporate image classification into retrieval system, especially if the images in database belong to different categories. For example, the crawled product images can be divided into categories such as “handbags”, “shoes”, *etc.* The landmark images can be categorized with keywords like “Eiffel Tower”, “Big Ben”, *etc.* Image classification can reliably reveal semantic contents of images and serve as a strong cue to identify irrelevant images and narrow down the search space.

Despite of these appealing properties, most of current retrieval systems [19], [29]–[31], [33], [35], [36] do not consider category clue, or simply use it as an additional feature for early fusion with other features [5] or late fusion with other retrieval results [28], [34]. Not withstanding their success, these algorithms either need to extract multiple features or have to fuse multiple retrieval results. They thus need to consume more retrieval time and extra memory. In this paper, we study how to take advantage of category clue in a more comprehensive way to achieve better retrieval accuracy, efficiency, and memory consumption. We call this new research strategy as category-aware visual search, which is compared with traditional image search in Fig. 1.

As shown in Fig. 1, category-aware visual search first classifies and indexes database images into different categories. During online retrieval, the category and visual clues of query are first extracted, then used to reject irrelevant categories and perform visual search, respectively. It is easy to infer that, one of the key tasks in category-aware visual search is to extract the category and visual clues illustrated in Fig. 1 from images. We could refer to existing image classification works for this task. However as discussed above, most of image classification works are expensive for image retrieval that requires fast response and high scalability. For instance, the state-of-the-art recognition approaches [14], [15], [20] generally require substantial computation with considerable memory requirements for thousands of classifiers.

To extract the category and visual clues efficiently, we propose to generate *category-visual vocabulary* illustrated in

Manuscript received June 28, 2013; revised December 20, 2013; accepted April 7, 2014. Date of publication April 17, 2014; date of current version May 7, 2014. This work was supported in part by the Army Research Office under Grant W911NF-12-1-0057, in part by the Faculty Research Awards through the NEC Laboratories of America, in part by the 2012 UTSA START-R Research Award, in part by the National Science Foundation of China under Grant 61128007, 61025011, and 61332016, and in part by the National Basic Research Program of China (973 Program) under Grant 2012CB316400. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Carlo S. Regazzoni.

S. Zhang and Q. Tian are with the Department of Computer Science, University of Texas at San Antonio, San Antonio, TX 78249 USA (e-mail: slzhang.jdl@gmail.com; qitian@cs.utsa.edu).

Q. Huang is with the Graduate University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: qmhuang@jdl.ac.cn).

W. Gao is with the School of Electronic Engineering and Computer Science, Peking University, Beijing 100871, China (e-mail: wgao@pku.edu.cn).

Y. Rui is with Microsoft Research Asia, Beijing 100080, China (e-mail: yongrui@microsoft.com).

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/TIP.2014.2317986

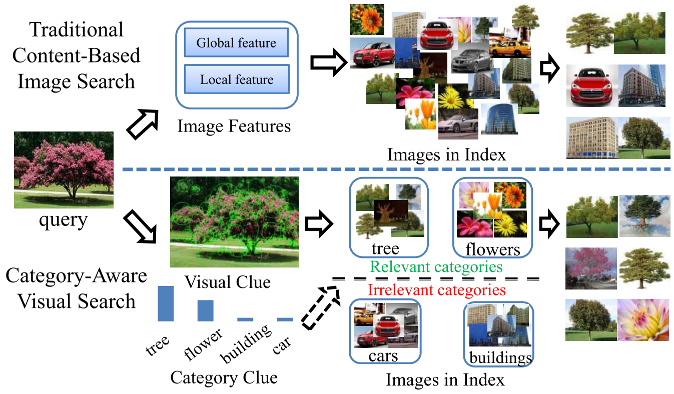


Fig. 1. Illustration of the traditional content based image search and the idea of category-aware visual search.

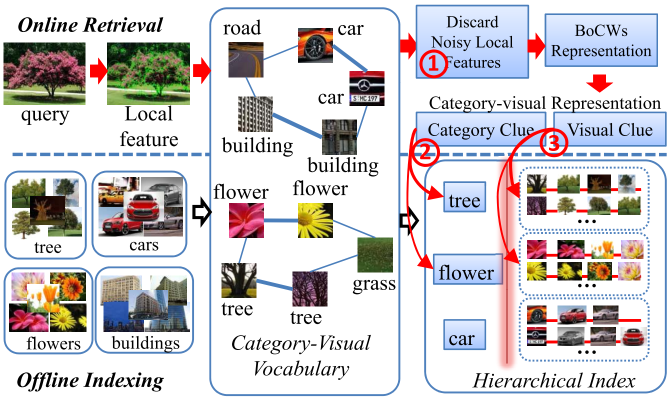


Fig. 2. Illustration of the category-visual vocabulary and the proposed cascade category-aware visual search.

Fig. 2. Category-visual vocabulary is composed of *category-visual words*, which are representative and discriminative local features annotated with category labels and contextual clues, *i.e.*, the co-occurrence relationships denoted by the connections. We generate category-visual vocabulary in a scalable and weakly supervised manner, hence claim to embed *weak category clue* in category-visual words. With category-visual vocabulary, we compute the Bag-of-Category visual Words (BoCWs) model for images, *i.e.*, extracting and representing local features as their nearest category-visual words like the process of Bag-of-visual Words (BoWs) model generation in [19]. During this process, local features not similar to any category-visual word are discarded as noisy features. Meanwhile, visual and category clues of images can be inferred by the category-visual words contained in their BoCWs models.

In the indexing stage, we first extract local features from database images, out of which we identify category-visual words to compute the BoCWs models and discard the noisy local features. Then as illustrated in Fig. 2, we index database images into their respective categories, where visual clue is considered to build inverted file index. Because many noisy local features are discarded during BoCWs model generation, this step generates compact index file.

In the retrieval stage, we follow a cascade retrieval procedure which: 1) discards the noisy local features and computes the BoCWs model of query, 2) filters irrelevant categories using the category clue of query to narrow-down the search space, 3) searches visually similar images in relevant categories. Because we gradually narrow down the search space by discarding noisy features, irrelevant categories, and visually dissimilar images, our retrieval system is expected to be more efficient than the conventional ones. Additionally, our algorithm only extracts local feature for query, but encourages semantic consensus in retrieved images because only categories semantically relevant to the query are searched. This is superior to most of conventional BoWs based retrieval systems [12], [19], [29], [31], [35], [36], which mostly focus on retrieving partial-duplicate images and hardly consider the semantics. We hence call the proposed method in Fig. 2 as *Cascade Category-Aware Visual Search*.

Image classification and large-scale image retrieval largely remain independent efforts due to different focuses on recognition accuracy and retrieval scalability. The state-of-the-art classification approaches generally require substantial computations, which are hardly affordable in online image retrieval. Most large-scale visual search works rely on local feature matching or global feature comparison that hardly consider the category clues. To the best of our knowledge, our work is one of the first efforts that embed category clue in large-scale image indexing and retrieval to achieve better retrieval accuracy, efficiency, and memory consumption. The contributions of this work can be summarized into the following aspects:

- We propose cascade category-aware visual search, which considers category clue in a more comprehensive way and achieves better retrieval performance than conventional systems.
- We embed weak category clue in category-visual vocabulary, which extracts both the category and visual clues from images efficiently.
- Our experimental results on three retrieval tasks validate the advantages of the proposed retrieval framework. These warrant the further investigation of incorporating image categorization in large-scale visual search.

The remainder of this paper is organized as follows. Section II reviews the related work on image retrieval and image classification. Section III presents our category-visual vocabulary generation. Section IV discusses the indexing and retrieval algorithms. Section V analyzes experimental results, followed by the conclusions in Section VI.

II. Related Work

This work focuses on improving large-scale image retrieval by considering categorical clues, which is closely related to vocabulary tree based image retrieval, image classification, and retrieval fusion. In the followings, we will briefly discuss related works as well as emphasize our differences with them.

Indexing bag of local invariant features [17] by visual vocabulary tree [19] has demonstrated an exceptional scalability for large-scale image retrieval. Visual vocabulary tree is generated by clustering a lot of local descriptors, *e.g.*, SIFT [17], hierarchically [19]. The resulting cluster centers in the leaf nodes of vocabulary tree are considered as visual words. To improve the discriminative power of visual word, lots of efforts have been made by spatial verification [32], [37] among matched visual words, embedding extra Hamming code to decrease the quantization error [11], building high-order features [30], [31], [36], co-indexing discriminative semantic attributes [35], *etc*. A review of local feature based image search can be found in [24]. Although significant progress has been made, most of these works focus on searching partial-duplicate images and hardly consider semantic clues, thus present limited functionality in the broader context of semantic-aware image retrieval.

BoWs based image retrieval commonly uses inverted file to index images [19]. In inverted file, each visual word is attached with a list of images where this visual word appears. Suppose the database contains N images, and each image contains V distinct visual words on average, the size of the index file would be proportional to $N \cdot V$. Hence, the size of inverted file index and the memory consumption are largely decided by the number of visual words extracted from images. Our work and some other works [18], [29] select discriminative local features from images, thus could be utilized to decrease the index size. However, these works [18], [29] either need extra image transformations to check the robustness of local features or require image pairwise relationships that are difficult to acquire in large-scale image database.

A large portion of image classification works feed high dimensional feature vectors into strong classifiers to identify the image categories. For instance, in the ImageNet Challenge [1], the recognition accuracy of top-5 candidates among 1000 categories has improved gradually to about 84% by extracting Fisher vectors [20], coding BoWs features [23], and learning deep network models [14]. Another portion of works focus on extracting mid-level features, *e.g.*, semantic attribute vector [4], [25] to bridge the semantic-gap.

Since semantic attribute vector reflects the image semantics, it presents a strong cue to find similar images [4]. In [4], Deng *et al.* propose to compute the image similarity by matching the hierarchy relationship between attributes conveyed by the WordNet [7]. Besides that, attribute vector is also commonly used in early fusion with visual features [5], or late fusion with other retrieval results [28], [34]. In [5], Douze *et al.*, reveal that combining a 2659 dimensional attribute vector and fisher vector substantially improves the retrieval performance. In [34], Zhang *et al.*, generate a graph by fusing multiple retrieval results. Then, a link analysis is conducted on the fused graph to merge and re-rank the initial retrieval results.

Early fusion of multiple cues requires the computation of multiple features, which substantially increases the computational complexity of online search. For instance, semantic attribute extraction [5], [20], [25] is generally expensive due to the extraction of high dimensional features and classification of thousands of object categories. In our experiments, extracting a 2048 dimensional attribute vector takes about 8 seconds on average on a single core using the package provided by [25]. Besides the computation, late fusion of multiple retrieval results [34] needs to store multiple indexes corresponding to different feature representations, which leads to considerable memory and computation costs.

Our work also uses category clue to improve image retrieval. Rather than extracting high dimensional feature vectors and training expensive classifiers, we estimate the category clue in a more efficient way, *i.e.*, quantizing image local features into category-visual words. Our work also differs from most BoWs based retrieval fusion works in that, it embeds category clue in index file and use it to narrow down the search space, rather than using it for late or early fusion. Therefore, besides boosting the accuracy, categorical clue also improves retrieval efficiency and memory cost.

III. Category-Visual Vocabulary

Suppose the database images could be divided into C categories. To utilize such category clue to improve the retrieval performance, we first generate category-visual vocabulary annotated with these category labels. Then, both the visual and category clues of an image can be efficiently estimated from the category-visual words contained in its Bag-of-Category visual Words (BoCWs) model.

We define category-visual vocabulary as a collection of category-visual words annotated with three clues: visual descriptor, category label, and contextual clue. We hence denote the category-visual vocabulary χ as, *i.e.*,

$$\{f_{cw}, l_{cw}, r_{cw}\}_{cw \in \chi}\ , \tag{1}$$

where f_{cw}, l_{cw}, r_{cw} are the visual descriptor, category label, and contextual clue of a category-visual word cw in χ, respectively.

Suppose χ is trained on a dataset containing C categories, χ^c denotes the set of category-visual words annotated with a category c. Denote q the query image and d an image in the database, and q is represented by a bag $\{x\}_q$ of local descriptors, where $x \in \mathbb{R}^D$ indicates dense or sparse image local descriptors of dimension D. So does $\{x\}_d$ for the database image d.

Based on χ, we compute the normalized visual word histogram $\mathbf{H}_q$ of q by extracting its local descriptors, quantizing each local descriptor with its nearest category-visual word, and counting the Term Frequency (TF) of each category-visual word in q. Each entry in $\mathbf{H}_q$ reflects the TF of the corresponding category-visual word. Hence $\mathbf{H}_q$ represents the visual clue of q. Based on the histogram, the possibility that q is labeled with c could be computed by voting the TFs of category-visual words cw annotated with category c in q, *i.e.*,

$$p\left(c|q\right) = \sum_{cw \in \chi^c} \mathbf{H}_q(cw). \tag{2}$$

Intuitively, the validity of the above image category clue computation largely depends on the discriminative power of the category-visual words annotated with category c, *i.e.*, χ^c. To make the computed $p\left(c|q\right)$ valid, χ^c should satisfy,

$$\chi^c = \arg\min_{\chi^c} \left(f_1\left(\chi^c\right) + \beta \times f_2\left(\chi^c\right)\right), \tag{3}$$

$$f_1\left(\chi^c\right) = \sum_{m,n} \left|\mathbf{H}_m\left(\chi^c\right) - \mathbf{H}_n\left(\chi^c\right)\right|_1 \times \mathrm{I}^c(m,n), \tag{4}$$

$$f_2\left(\chi^c\right) = \sum_{m} \left|\mathbf{H}_m\left(\chi^c\right)\right|_1 \times \mathrm{P}^c(m), \tag{5}$$

$$\mathrm{I}^c\left(m,n\right) = \begin{cases} 1, \text{ if } m,\ n \in c \\ -1, \text{ otherwise} \end{cases} \mathrm{P}^c\left(m\right) = \begin{cases} -1, \text{ if } m \in c \\ 1, \text{ if } m \notin c \end{cases}, \tag{6}$$

where $|\cdot|_1$ returns L-1 norm of a vector, m and n are two images in the database, $\mathrm{I}^c(\cdot)$ and $\mathrm{P}^c(\cdot)$ are the indicator functions denoting the categorical relationships between images and category c. In Eq. (3) we consider two factors, *i.e.*, repeatability and discriminative power. The conceptual parameter β balances these two factors to evaluate the validity of category-visual words.

It is easy to infer that, χ^c is expected to satisfy two constraints in Eq. (3): 1) *high repeatability in* c: minimize the distance between two images if both of them are from category c, *i.e.*, Eq. (4); 2) *high discriminative power to* c: category-visual words in χ^c are expected to appear frequently in images from category c, but appear rarely in the ones from other categories, *i.e.*, Eq. (5). Intuitively, the desired category-visual words annotated with c are expected to be the repeatable and discriminative exemplar local features in the images of category c. For instance, the image patches on the body of tiger could be the category-visual words annotated with "tiger", because they satisfy both of the above two constraints.

To generate category-visual vocabulary, we could introduce distance metric learning strategies or supervised visual vocabulary selection with boosting [16], *etc.*, to acquire an optimal solution to Eq. (3). However, the large number of categories and images in large-scale database make solving Eq. (3) with most of existing learning strategies too expensive to be applicable. For instance, it is hard to select millions of visual words with boosting strategy [16]. Lot of distance metric learning algorithms, *e.g.*, the Maximally Collapsing Metric Learning proposed in [9], are $\mathrm{O}(n^2)$ in complexity. Thus, they are also not scalable for this problem. Moreover, the category clues in large-scale image datasets could be weak and noisy, *e.g.*, different categories like "great white shark" and "tiger shark" may share similar visual appearance and semantics. Therefore, weakly supervised algorithms with high efficiency are more optimal for our problem.

Classic visual vocabulary is efficiently generated by hierarchically clustering the local features and the generated cluster centers are taken as visual words [19]. Inspired by this, we propose a weakly supervised strategy to guide the generation of category-visual vocabulary. Specifically, 1) within each category: we select a collection of visual words that constantly appear in most images to satisfy the repeatability constraint in Eq. (4); 2) across different categories: we identify category-visual word candidates by selecting visual words that appear frequently in one category but rarely in the others to satisfy the discriminative power constraint in Eq. (5); 3) the candidates with high repeatability and discriminative power are annotated with category labels and contextual clues to be the category-visual words. Fig. 3 illustrates the category-visual word generation, which will be discussed in detail in the following parts.

A. Within Category Clustering

As illustrated in Fig. 3, within a category c we first hierarchically cluster the set of extracted local descriptors, *e.g.*, G^c, to generate a visual vocabulary γ^c. In order to control the number of generated clusters and make sure the local features in the same cluster share consistent visual similarity with each other, we introduce a similarity constraint ω to guide the clustering. We denote the generation of γ^c as

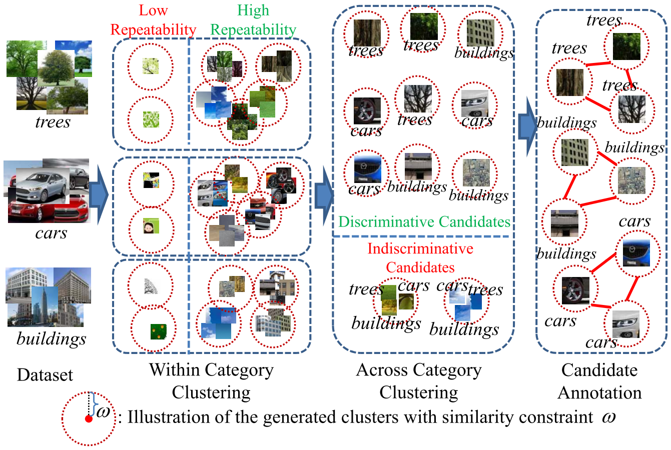


Fig. 3. Illustration of the category-visual vocabulary generation.

$$\gamma^c = \mathbb{C}\left(G^c, \omega, B\right), \tag{7}$$

where γ^c is the cluster centers on leaf nodes of the generated vocabulary tree, $\mathbb{C}(\cdot)$ is the clustering function, and B represents the maximum branch number in the vocabulary tree. During clustering, in one generated cluster, if the similarity between each local descriptor and the cluster center is larger than ω, the cluster center will be taken as a visual word. Otherwise, this cluster will be partitioned for further clustering. Hence, larger ω means finer partition of the feature space, thus generates more visual words and improves the discriminative power of each visual word. However, over-partitioning the feature space may result in low recall rate. We will discuss this parameter in detail in Sec. V.

To satisfy the repeatability constraint in Eq. (4), from γ^c we discover visual words that appear constantly in most images in c. We compute the repeatability factor $\phi(vw, c)$ to evaluate the repeatability of visual word $vw \in \gamma^c$ in c, *i.e.*,

$$\phi(vw,c) = \frac{\sum\limits_{d\in c} \mathrm{t}_d(vw)}{\mathrm{std}\left(\{\mathrm{t}_d(vw)\}_{d\in c}\right)}, vw \in \gamma^c, \tag{8}$$

where $\mathrm{t}_d(vw)$ is the number of appearance of visual word vw in image d, $\mathrm{std}(\cdot)$ denotes the standard deviation, which reflects the consistency that visual word vw repeats in each image from category c. As illustrated in Fig. 3, visual words with high repeatability and low repeatability in c can be distinguished by the computed $\phi(vw, c)$.

Because many noisy visual words appear on frequently repeated backgrounds like grass, sky, *etc.*, they may also show high repeatability. For instance, the patches on sky show high repeatability in "buildings" category in Fig. 3, but they are not discriminative to the building object. Therefore, the repeatability factor does not reflect the discriminative power. In order to satisfy the discriminative power constraint in Eq. (5), *i.e.*, visual words annotated with certain category should appear rarely in other categories, we proceed to evaluate the discriminative powers of the generated visual words by across category clustering.

B. Across Category Clustering

To test the discriminative power of a visual word $vw \in \gamma^c$ to category c, an effective way is to test if vw is unique for c, *i.e.*, vw would be discriminative to c, if no visual word similar to vw could be generated from other categories. To check if there exist similar visual words to vw in other categories, we cluster the visual words generated from different categories with a similarity constraint, *i.e.*,

$$(\hat{\chi}, O) = \mathbb{C}\left(\{\gamma_c\}_{c\in C}, \omega, B\right), \tag{9}$$

where ω is the same similarity constraint in Eq. (7) and $\{\gamma_c\}_{c\in C}$ is the collection of generated visual words in C categories. O denotes the set of generated clusters and $\widehat{\chi}$ is the set of generated cluster centers.

In Eq. (9), visual words from different categories with similarity larger than ω are clustered together. As illustrated in Fig. 3, the unique visual words with strong discriminative power will distribute far from the others. While the visual words on similar backgrounds would stay close to each other. Hence, the discriminative power of a visual word to its source category could be reasonably estimated by the size of the cluster it belongs to.

Denote $o \in O$ as a cluster containing the visual word vw generated in c. We compute the discriminative power of vw to its source category c as

$$\eta(vw, c) = \left|\gamma^c \cap o\right| / |o|, vw \in \gamma^c \cap o, \tag{10}$$

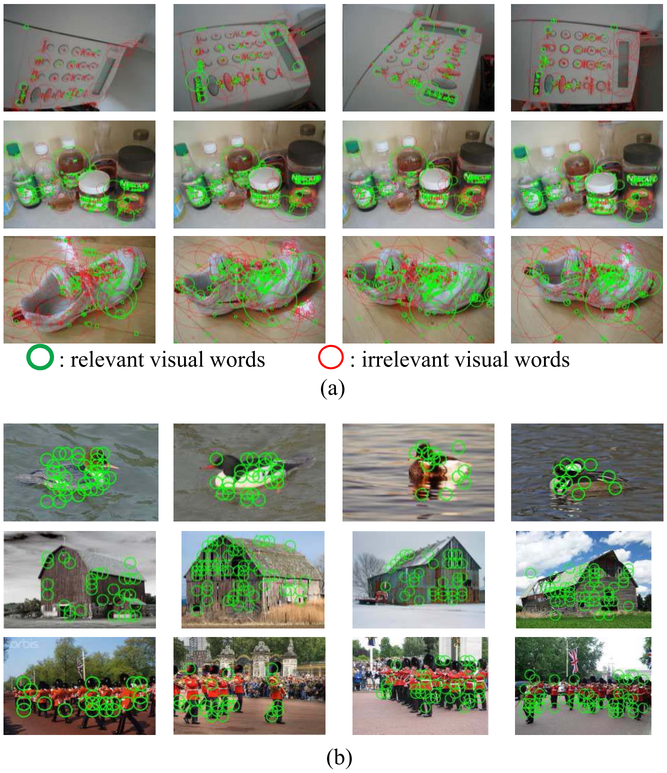
where $|o|$ denotes the total number of visual words in cluster o. According to Eq. (10), the unique visual words in each category would have large discriminative power.

Category-visual words annotated with a category should satisfy two constraints, *i.e.*, high repeatability and high discriminative power to this category. To evaluate if a visual word vw satisfies the two constraints, we combine the two clues ϕ and η to compute the relevance degree of this visual word to its source category c, *i.e.*, $\mu(vw, c)$ as

$$\mu(vw, c) = \phi(vw, c) \times \eta(vw, c), vw \in \gamma^c. \tag{11}$$

Examples of visual words with large and small relevance degrees to different categories are illustrated in Fig. 4. It can be seen that most of relevant visual words are located on the objects and are repeated across different images in the same category. It is also obvious that visual words with small relevance degrees are mostly located on the cluttered background. The examples in Fig. 4 intuitively illustrate the validity of the computed relevance degree.

After across category clustering, the generated clusters in O contain both relevant visual words for each category and irrelevant visual words on clutter backgrounds. We consequently take the corresponding cluster centers $\widehat{\chi}$ as the set of *category-visual word candidates*. Hence, the category-visual word candidates are computed with many clustering tasks in two stages, *i.e.*, within category clustering and across category clustering. Because the scale of data processed in each clustering task is limited, *i.e.*, local descriptors in one category or cluster centers from different categories, our algorithm is parallelizable and scalable. Our algorithm also shows substantial flexibility because different clustering or vector quantization strategies could be applied as the function $\mathbb{C}(\cdot)$ in Eq. (7) and Eq. (9). In this paper, we use hierarchical k-means clustering for its high scalability and reasonably good performance in large-scale image retrieval [19]. In the following part, we introduce how to annotate these candidates and identify category-visual words.



(a)

(b)

Fig. 4. Illustration of relevant visual words and irrelevant visual words on two datasets. (a) The relevant visual words and irrelevant visual words for three categories in UKbench [19] Local descriptors are detected by Difference-of-Gaussian [17]. (b) The relevant visual words (green circles) for three categories in LSVRC10 [1]. The other irrelevant visual words are not shown. The local features are densely sampled from the image.

C. Candidate Annotation

Denote o as a cluster in O and $\widehat{\chi}_o$ as the cluster center, we define the category label l_o of cluster center $\widehat{\chi}_o$, *i.e.*, a candidate category-visual word, as

$$l_o(c) \in \mathbb{R}^C, \sum_{c=1}^{C} l_o(c) = 1, \tag{12}$$

where C is the number of categories in database. The dimensionality of the label vector l_o hence equals to C. We use $l_o(c)$ to denote the probability that $\widehat{\chi}_o$ is labeled with category c.

We annotate the category-visual word candidates based on the visual words contained in their corresponding clusters. For instance, suppose a cluster o contains two visual words from categories "great white shark" and "tiger shark", we could reasonably annotate $\widehat{\chi}_o$ with these two category labels. The probability of these two category labels can be estimated according to the relevance degrees of the two visual words to "great white shark" and "tiger shark", respectively. We hence compute the category label l_o of candidate $\widehat{\chi}_o$ with Eq. (13)

and Eq. (14), *i.e.*,

$$\bar{l}_o(c) = \sum_{vw \in o \cap \gamma^c} \mu(vw, c), \tag{13}$$

$$l_o = \mathbb{N}(\bar{l}_o) \tag{14}$$

where $\bar{l}_o$ denotes the category label before normalization. After summarizing the relevance degree of each visual word in o, we normalize their sums to 1 with function $\mathbb{N}(\cdot)$ to get the final category label l_o.

After annotating all category-visual word candidates in $\widehat{\chi}$, we identify the candidates with discriminative labels as category-visual words. It is easy to infer that, a label would be discriminative if it focuses on limited number of categories. Hence, discriminative labels would show larger STandard Deviations (STDs) than fuzzy labels. To identify and keep the discriminative candidates, we sort the candidates with the STDs of their category labels and remove the bottom 20% with the smallest STDs. Like the stop word removal strategy in [19], our strategy removes many noisy candidates, hence is helpful to generate a discriminative category-visual vocabulary. However, removing too many candidates may discard some useful candidates, and is harmful for the performance. We will discuss this threshold setting in our experiments.

The finally kept candidates in $\widehat{\chi}$ are hence taken as category-visual words. Intuitively, lots of noisy candidates on cluttered backgrounds could be discarded in this step. We denote the generated category-visual vocabulary as χ and category-visual word in it as cw. Because $\widehat{\chi}$ is generated by hierarchical k-means clustering, the category-visual word candidates in it would be the leaf nodes of a vocabulary tree. After candidate selection, we simply discard the noisy leaf nodes and do not change the hierarchical structure of the tree. Therefore, the kept leaf nodes, *i.e.*, category-visual words are still organized in the hierarchical tree structure, which allows for fast feature quantization during BoCWs model generation.

D. Contextual Clues Computation

After acquiring category-visual words, we proceed to discover their contextual clues, *i.e.*, the co-occurrence relationship among category-visual words in the same category. We define the contextual clue r_{cw} of a category-visual word cw as

$$r_{cw}(c) \in \mathbb{R}^C, \sum_{c=1}^{C} r_{cw}(c) = 1, \tag{15}$$

where C is the number of categories and $r_{cw}(c)$ represents the co-occurrence probability between the category-visual word cw and others category-visual words labeled with category c.

To capture such contextual clue, we first represent images in each category as BoCWs models with Eq. (16), *i.e.*,

$$\{cw\}_d = \mathbb{Q}\left(\{x\}_d, \chi, \omega\right), \tag{16}$$

where $\{cw\}_d$ is the BoCWs model of image d, $\{x\}_d$ is its set of image local descriptors, χ is the category-visual vocabulary, and $\mathbb{Q}(\cdot)$ denotes the vector quantization function which quantizes a local descriptor as its nearest category-visual word if their similarity is larger than ω, or discard this local descriptor if the largest similarity is smaller than ω. Hence, the noisy local features not similar to any category-visual word could be efficiently identified and discarded by the similarity constraint ω.

With the BoCWs models, the contextual clue of a category-visual word cw in a category c is computed as

$$\bar{r}^c_{cw} = \sum_{v \in \{cw\}^c, v \neq cw} l_v, \tag{17}$$

where $\{cw\}^c$ is the collection of category-visual words appear in category c. Hence, $\bar{r}^c_{cw}$ reflects the labels of co-occurred category-visual words with cw in c.

Since cw may appear in different categories, we hence summarize and normalize its contextual clues computed in different categories to get the final contextual clue, *i.e.*,

$$r_{cw} = \mathbb{N}\left(\sum_{c \in C} \bar{r}^c_{cw}\right). \tag{18}$$

Suppose two category-visual words $cw1$ and $cw2$ are labeled with category a and b, respectively. If $cw1$ and $cw2$ co-occur frequently in the same category, their corresponding co-occurrence probabilities, *i.e.*, $r_{cw1}(b)$ and $r_{cw2}(a)$ would show relatively large values. In Sec. IV-A, we introduce how we use contextual clue to improve the image annotation.

After computing the contextual clue, each category-visual word contains three aspects of information, *i.e.*, visual feature f, category label l, and contextual clue r, we hence represent the category-visual vocabulary as: $\{f_{cw}, l_{cw}, r_{cw}\}_{cw \in \chi}$. In the followings, we proceed to introduce our cascade indexing and retrieval strategies.

IV. Indexing and Retrieval

As illustrated in Fig. 2, before indexing and retrieval, our method needs to compute the category and visual clues of images. In this section, we first present how to compute the category-visual representation with category-visual vocabulary, then proceed to the introduction of our indexing and retrieval strategies.

A. Category-Visual Representation

Based on the category-visual vocabulary, images could be represented as BoCWs models with Eq. (16). The visual clues and potential categories of each image could then be estimated with the category-visual words in its BoCWs model. We define the category-visual representation of an image d as $(\mathbf{H}_d, \mathbf{L}_d)$, where $\mathbf{H}_d$ is the normalized category-visual word histogram and $\mathbf{L}_d$ is the category label vector of the image.

Suppose the BoCWs model of d is $\{cw\}_d$, based on which we compute the histogram $\mathbf{H}_d$ by counting the TF, *i.e.* term frequency of each category-visual word appears in the image. Referring to the category clue computation in Eq. (2), we compute the initial category label vector $\widehat{\mathbf{L}}_d$ of d as

$$\widehat{\mathbf{L}}_d = \mathbb{N}\left(\sum_{v \in \{cw\}_d} l_v\right). \tag{19}$$

After computing $\widehat{\mathbf{L}}_d$, we proceed to use contextual clues of category-visual words to refine it. Inspired by PageRank [3], we propose the LabelReRank. In PageRank, a matrix is built to record the contextual relationships between webpages. Iterations are then carried out to update the weight of each webpage based on this matrix. After several iterations, the weight of each webpage is obtained combining both its inherent importance and contextual relationships with the other webpages. In LabelReRank, we construct a $C \times C$ sized matrix M to show the contextual clues recorded in category-visual words. The element $M(i, j)$ is computed with Eq. (20) and Eq. (21) to reflect the probability of co-occurrence between category-visual words labeled with category i and j, *i.e.*,

$$\overline{M}(i, j) = \sum_{v \in \{cw\}_d} l_v(i) \times r_v(j), \tag{20}$$

$$M(i, j)|_{i=1}^{C} = \overline{\mathbb{N}}\left(\overline{M}(i, j)|_{i=1}^{C}\right), \tag{21}$$

where $\overline{\mathbb{N}}(\cdot)$ normalizes the sum of each column in $\overline{M}$ to 1.

With M, we utilize LabelReRank to update the initial category label iteratively in Eq. (22), *i.e.*,

$$\mathbf{L}_d = (1 - \sigma) \times M \times \widehat{\mathbf{L}}_d + \sigma \times \frac{1}{C}, \tag{22}$$

where $\mathbf{L}_d$ is the updated category label vector, the damping factor σ denotes the random transitions among different categories. $\sigma < 0.2$ is often chosen in practice in many image search works using Random Walk model [3], [13]. As discussed in [13], the choice of damping factor does not significantly revise the contextual relativity recorded in the matrix M, thus it has minor affect to the final category label. We hence set σ as 0.15, which is the recommended value by various studies [3]. The effectiveness of LabelReRank can be explained by an example. Suppose an image contains flowers and leaves. The initial labels of this image may include "flower", "leaf" and some noisy category tags. Because the category-visual words annotated with "flower" and "leaf" show high contextual relationship to each other, the values of $M(flower, leaf)$ and $M(leaf, flower)$ would be relatively larger than the others. As a result, the weights of these two categories will be boosted during the iterations in Eq. (22). Meanwhile, the influences of other noisy categories are suppressed. Hence, LabelReRank is expected to eliminate the noise and get more accurate category clues.

After computing the category-visual word histogram and the category label vector, we could represent each image as the category-visual representation, *i.e.*, $(\mathbf{H}, \mathbf{L})$.

Examples of BoCWs models of queries on two datasets are illustrated in Fig. 5(a) and (b). Note that the queries are independent of the training set of category-visual vocabulary. It can be observed that most of local features on the objects are identified as category-visual words and lots of noisy ones on the backgrounds are discarded. In Fig. 5(b) we also show the top four estimated category labels of queries in LSVRC10. It can be seen that most of the estimated labels are reasonable. These examples intuitively manifest the validity of the category-visual vocabulary and the computed category-visual representation. In the following parts, we proceed to introduce our indexing and retrieval strategies.

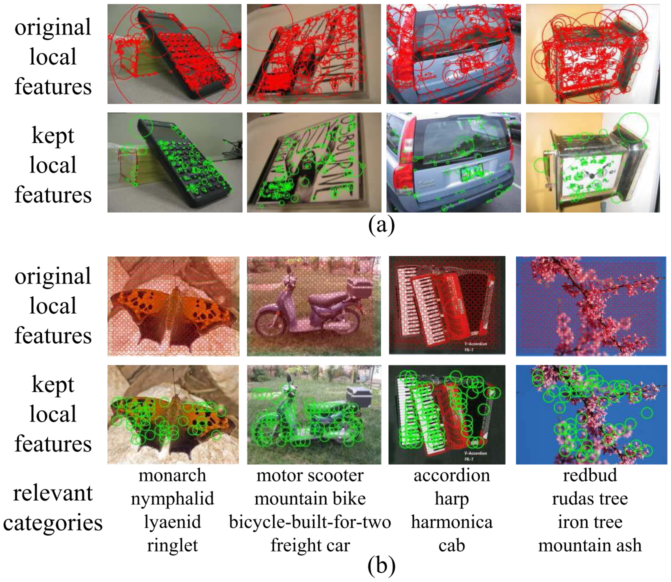


Fig. 5. Examples of original local features and kept local features after BoCWs computation in queries from UKbench (a) and LSVRC10 (b). (b) shows the top four estimated category labels of queries from LSVRC10.

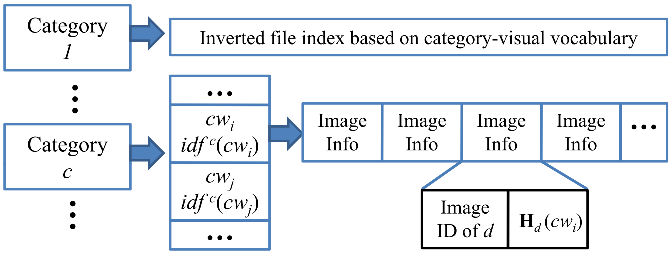


Fig. 6. Illustration of the proposed hierarchical index structure.

B. Hierarchical Indexing

Based on the category-visual representation, we propose a hierarchical index structure illustrated in Fig. 6. As shown in the figure, we first estimate the category labels of database images to classify them into relevant categories, and then build inverted file index in each category.

Because we loosely annotate the images, some category labels might be fuzzy and many images may be relevant to multiple categories. Indexing such images into one category would result in poor recall rate. However, repeatedly indexing them in multiple categories increases the size of index file. To seek a trade-off, we index an image d into the most relevant category c, if the weight of c in d's label vector $\mathbf{L}_d$, *i.e.*, $\mathbf{L}_d(c)$ is larger than a threshold h. Otherwise, we index d into more relevant categories till the total weight of these categories in $\mathbf{L}_d$ is larger than h. This threshold largely decides the size of index file, *e.g.*, larger h means higher probability to index one image into multiple categories; on the contrary smaller h reduces the redundancy, but may result in lower recall rate, especially with poor image annotation accuracy. Our experiments show that $h = 0.5$ is a reasonable tradeoff between memory cost and accuracy.

On two datasets, we count the number of category each image is indexed into when $h = 0.5$, and show the statistical results in Fig. 7. It can be seen that, most of images, *i.e.*, about

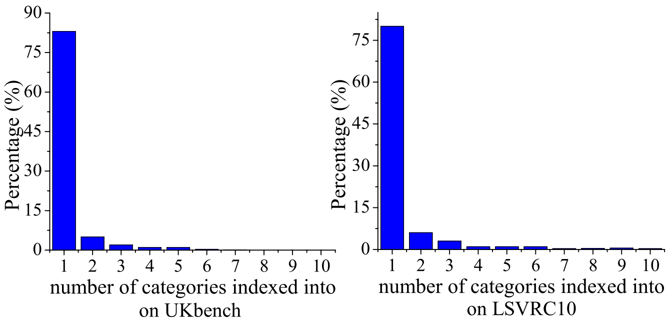


Fig. 7. Histograms of the number of category each image is indexed into on UKbench and LSVRC10 if the threshold h is set as 0.5.

80%, are indexed into one category. Note that, we also filter many noisy local features during BoCWs model generation. Consequently, the generated index still could be more compact than the traditional index in [19]. The effects of h will be discussed in Sec. V.

After indexing the images into different categories, in each category we build inverted file index illustrated in Fig. 6. In inverted file index, each category-visual word is taken as a keyword and is connected to a list of images containing it. For an image d recorded in the list of cw, we record two types of information: the image ID of d and the TF of cw in d, *i.e.*, the $\mathbf{H}_d(cw)$. $idf^c(cw)$ denotes the Inverse Document Frequency (IDF) of cw in category c. IDF reflects the discriminative power of a keyword in retrieval. For instance, keywords like “a” and “the” have small IDF values, because they commonly appear in any document. More details of TF and IDF could be found in [19].

C. Cascade Retrieval

Based on the above index structure, we build a cascade retrieval algorithm, which includes three steps: 1) compute the BoCWs model $\{cw\}_q$ of query q with Eq. (16) to filter noisy local features, 2) compute the category-visual representation $(\mathbf{H}_q, \mathbf{L}_q)$ with Eq. (22) and filter irrelevant categories, 3) search images with visual feature $\mathbf{H}_q$ in inverted file indexes of relevant categories.

In the second step, we reject α percent categories with minimum weights in $\mathbf{L}_q$ and keep the rest $(100 - \alpha)$ percent for visual search. This further improves the recall rate. The parameter α will be discussed in Sec. V. In the third step, the similarity between q and a database image d indexed in category c is computed as

$$sim(q,d) = \sum_{v} idf^c(v) \times \min\left(\mathbf{H}_q(v), \mathbf{H}_d(v)\right)$$
$$v \in \{cw\}_q \cap \{cw\}_d, \tag{23}$$

where $\mathbf{H}_d(v)$ and $idf^c(v)$ are recorded in the inverted file index, v is a category-visual word that appears in the BoCWs models of both q and d. Because a database image may be indexed into different categories, multiple similarity values between query and this image could be computed. In such case, we take the maximum value as the final similarity.

After computing and sorting the similarities of database images, relevant images could be returned. Because only images in relevant categories are considered, Eq. (23) implicitly combines category and visual clues to compute the image similarity. Hence, our approach encourages the semantic consistency in retrieved images.

V. Experiments

A. Experimental Setup

We evaluate the proposed method in three retrieval tasks: object search, landmark search, and large-scale similar image search. We use the UKbench [19] dataset, which contains 2,550 objects under 4 different viewpoints for object search. For landmark search, we collect a dataset from the Google Image Search by searching images with landmarks such as “Eiffel Tower”, “Big Ben”, *etc*. From the downloaded images, we select 50 categories, within each we keep 150 relevant images for indexing and 50 relevant images as queries. Finally, we get a landmark dataset containing 7500 images and 2500 queries. As for the large-scale similar image search, to make our experimental implementation and evaluation feasible, we use the LSVRC10 dataset [1]. As a subset of the ImageNet [4], LSVRC10 contains 1000 categories, about 1.2 million training images, and 150K test images. It covers different types of categories including scenes, man-made objects, plants, animals, *etc*.

Our method is not restricted to the type of local descriptor. We extract SIFT descriptors from interest points detected by Difference-of-Gaussian (DoG) [17] for object and landmark search. As for similar image search, we first densely sample 32 × 32 image patches with step size 16. Then from each patch, we fuse three types of descriptors, *i.e.*, 128 dimensional SIFT, 256 dimensional color histogram, and 256 dimensional LBP texture [2] as the local feature. During clustering and feature quantization, we equally combine three corresponding similarities, each of which is normalized between 0 and $1/3$, as the similarity measurement for the fused local features. All experiments are conducted on a PC with a 4-core 3.4GHz processor and 16GB memory.

Our method is related to the following parameters, *i.e.*, ω: the similarity threshold in clustering and feature quantization, α: the percentage of discarded categories during retrieval, h: which decides the number of categories each image may be indexed into, and B. B controls the branch number in the hierarchical k-means clustering. Larger B generates flatter vocabulary tree with more branches, which helps to decrease the quantization error on each layer of the tree. However, flatter tree also makes feature quantization along the vocabulary tree and the hierarchical k-means clustering more expensive [8]. Detailed discussions about the effects of B can be found in [8]. In our experiments, we simply set B as 20, a reasonable tradeoff between accuracy and efficiency. In the followings, we will test our method, as well as discuss the effects of ω, α, and h.

B. Object Search

Each category in UKbench contains 4 images of the same object. To exclude the queries from training set of category-visual vocabulary, we repeat 4 groups of experiment.

TABLE I

PERFORMANCE ON UKBENCH WITH DIFFERENT VALUES OF ω

ω	0.7	0.74	0.78	**0.82**	0.86	0.9
N-S Score	1.65	1.93	2.23	**2.44**	2.41	2.24
Vocabulary Size (M)	0.05	0.12	0.25	0.32	0.45	0.52
Portion of kept local feature (%)	80.3	78.5	75.5	71.9	68.6	63.3

TABLE II

PERFORMANCE ON UKBENCH WITH DIFFERENT VALUES OF α

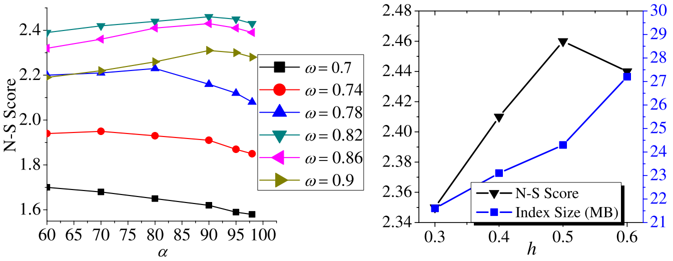
α	0	60	70	80	**90**	95	98
N-S Score	2.17	2.39	2.42	2.44	**2.46**	2.45	2.43

Each group uses 3 images for category-visual vocabulary generation and indexing, and the rest image as query. For each group, we compute the N-S score that counts the number of positive images *in the first 3 returned images*. The averaged N-S score of the four groups is hence taken for performance evaluation. Because ω controls category-visual vocabulary generation and α controls the irrelevant category rejection, they are relatively independent of each other. Hence we first test one of them by fixing the other. In the followings, we test ω and set α as 80. The results are illustrated in Table I.

It can be seen from Table I that, ω substantially influences the number of generated visual words and the retrieval performance. Smaller ω means less strict similarity constraint in clustering. Larger ω partitions the feature space in finer scale and generates larger tree. Large ω also reduces the number of local features that could pass the similarity constraint in BoCWs computation in Eq. (16). Consequently, in Table I, the portion of kept local features after BoCWs computation constantly drops if ω is larger. From this experiment, we find that larger ω is helpful to improve the retrieval accuracy, however too large ω degrades the accuracy. This might be because too large ω also discards many discriminative local features and generates many category-visual words with low repeatability by over-partitioning the feature space. From the experiment, we found that setting ω as 0.82 generates 0.32 M category-visual words and gets the best N-S score. In the followings, we fix ω as 0.82 and test α.

The influence of α is illustrated in Table II. It can be observed that, discarding irrelevant categories substantially improves the performance. This clearly demonstrates the advantage of the cascade category-aware visual retrieval since narrowing down the search space not only improves the efficiency but also gets better accuracy. Besides that, it also validates the calculated category labels of images. We also observe that discarding too many categories degrades the performance. This is reasonable because we use weak category clues to loosely annotate images. Hence many relevant categories may be discarded if α is too large. From this experiment, we can conclude that, although the category clue introduced is weak, it still remarkably improves the retrieval accuracy.

Note that, ω affects the generated category-visual vocabulary, which relates to the accuracy of image annotation. With more accurate category clue, the retrieval algorithm could discard more irrelevant categories, *i.e.*, setting larger α, without degrading the recall rate. Therefore, α and ω are not strictly independent of each other. In previous experiments, we concisely show the individual effects of these two parameters. The joint effects of them are shown in Fig. 8(a). The effects of h are illustrated in Fig. 8(b). It can be seen that small h indexes images into less categories hence generates more compact index. Because the retrieve algorithm considers a portion of categories, too small h would results in poor recall rate. It also can be observed that too large h produces large index file, but does not constantly improve the retrieval accuracy. This might be because many irrelevant images are also indexed into the same category with too loose constraint. From the Fig. 8, we conclude that setting ω as 0.82, α as 90, and h as 0.5 gets the best N-S score, *i.e.*, 2.46.



(a) (b)

Fig. 8. Illustration of the effects of α, ω, and h on UKbench.

TABLE III

COMPARISON WITH THE BASELINE ON UKBENCH

Method	Vocabulary Size (M)	N-S Score	Index Size (MB)
Proposed	0.32	**2.46**	**24.3**
Baseline [19]	0.33	1.98	30.5

To demonstrate the advantage of our method over existing ones, we use visual vocabulary based retrieval [19] as the baseline. We implement the baseline with the same experimental setting, *i.e.*, repeat the retrieval for four times and each time uses three images for indexing and training a vocabulary tree that has 4 layers and 24 branches. Hence, the vocabulary size of the baseline is about 0.33 M, which is similar to the size of category-visual vocabulary. We compare the average N-S score and memory consumption between our method and the baseline in Table III.

In Table III, our method shows substantially better accuracy and memory consumption than the baseline. Decreasing 20% index size (30.5MB to 24.3MB) and improving the N-S score from 1.98 to 2.46 (max: 3.0) are significant progresses. This is mainly because we utilize extra category clue to filter noisy local features and irrelevant categories. Some examples of retrieval result are shown in Fig. 9. Obviously in Fig. 9, the images retrieved by our method are more consistent with the query in semantics than the results of baseline. Since the retrieval time on small-scale dataset is mostly decided by the feature extraction and quantization, we evaluate the efficiency in large-scale image retrieval task in Sec. V-D.

Besides the baseline, we also make comparison with several state-of-the-art algorithms in Table IV. Note that, these works index all the 4 images in each category and compute N-S score in the first 4 returned images. Because queries are also

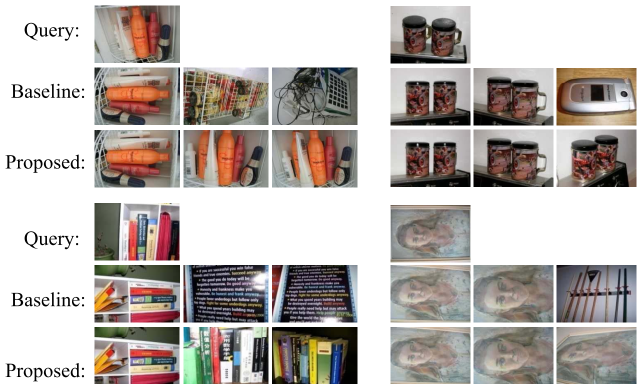


Fig. 9. Examples of retrieval result based on proposed algorithm and baseline on UKbench.

TABLE IV
COMPARISON WITH STATE-OF-THE-ART ON UKBENCH

Methods	Proposed	[21]	[11]	[36]	[26]	[22]
N-S Score[1]	**2.46**	2.45	2.42	2.26	2.56	2.52

indexed, the first returned image would be the query. Hence, to make fair comparison, we reasonably subtract their reported N-S scores by 1 to remove the influence of query. As shown in Table IV, our method outperforms [21], [11], and [36] and achieves slightly worse performance than [26] and [22].

It is necessary to note that, these works compared in Table IV effectively improve the retrieval accuracy of baseline [19], but also introduce extra computations or memory costs. For instance, [11] records an extra 64-bit Hamming code for each visual word to decrease the quantization error. The method in [26] doubles the memory cost compared to [19] because of the recorded contextual clues among visual words. As illustrated in Table III, our algorithm consumes fewer memory than baseline [19], since many noisy local features are discarded and won't be indexed. In the retrieval stage, [21], [36], and [26] need extra computations to extract and verify the spatial consistency among visual words. The system of [22] also needs to compute and fuse the KNN relationships among images. However, our approach can be more efficient than baseline, because it discards irrelevant categories and noisy local descriptor thus needs to scan fewer images and inverted lists. We do not compare the retrieval efficiency on this small-scale dataset, since the extraction and quantization of local features dominate the online computation. The comparison of efficiency between our approach and [19] on large-scale dataset in Sec. V-D does manifest this conclusion. Therefore, it is reasonable to claim our approach also shows better memory cost and efficiency than the recent works in Table IV.

C. Landmark Search

In this experiment, we test our method on the collected landmark dataset. Because we also extract sparse SIFT descriptors from landmark images, we refer to the setting of ω in the previous experiment. By setting ω as 0.82, and discarding 20% candidates, our algorithm finally generates a vocabulary containing 27K category-visual words. Besides that, we also generate another two category-visual vocabularies containing 30K and 24K category visual words by discarding 10% and 30% candidates, respectively. In this retrieval task, we compare following five algorithms,

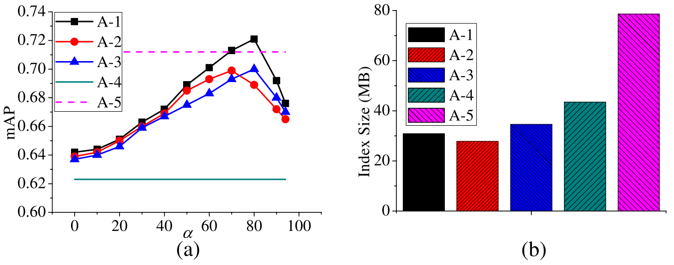


Fig. 10. Comparisons of mAP and memory consumption in landmark image search.

- A-1: our approach with 27K category-visual words.
- A-2: our approach with 30K category-visual words.
- A-3: our approach with 24K category-visual words.
- A-4: visual vocabulary tree based image retrieval with 28K classic visual words, *i.e.*, 4 layers, 21 branches [19].
- A-5: Bundled Feature [27] with the 28K visual words trained in A-4.

We use mean Average Precision (mAP) which computes the average precision across different recall levels in the first 150 returned images as the performance measurement. To make the comparison fair, we also train the vocabulary tree in A-4 on the 7500 database images. The comparisons of mAP and index size among five algorithms are illustrated in Fig. 10.

As shown in Fig. 10(a), our method constantly shows better performance than the baseline. Meanwhile, after rejecting 80% irrelevant categories, A-1 also outperforms the Bundled Feature, which records extra spatial clues between local features in index file and verifies the spatial consistency in online retrieval to identify the mismatched visual words [27].

Fig. 10(b) shows the comparison of the size of index file. Since retrieval systems need to load the index file in memory, this figure intuitively compares the memory consumptions. It can be observed that, A-1 only consumes 32MB to load the index file, which is substantially better than Bundled Feature (80MB) and classic visual vocabulary (45MB). This is mainly because lots of the noisy local descriptors are discarded in BoCWs model computation, hence are not indexed. Bundled Feature needs to record extra spatial clues in index file, hence is the most memory consuming algorithm in the figure. Examples of retrieval results based on our method are illustrated in Fig. 11.

From this experiment, we also observe the effects of removing different portions of candidates with small STDs of category labels. Removing either too many or too few candidates would degrade the best performance. Besides that, keeping more candidates produces larger index because larger vocabulary introduces more visual words and builds more inverted lists. We conclude that discarding 20% candidates in category-visual vocabulary generation achieves a reasonable tradeoff between accuracy and memory cost.

[1] The reported N-S scores are subtracted by 1 to make fair comparison

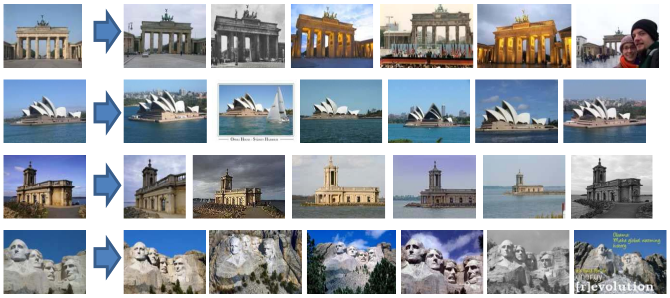
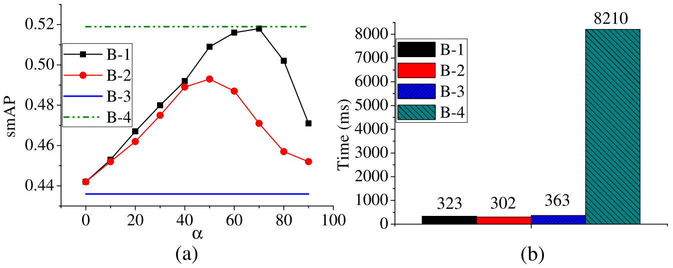


Fig. 11. Examples of retrieval result in landmark image search. The leftmost image is the query.



(a) (b)

Fig. 12. Comparisons of smAP and efficiency in large-scale similar image search.

D. Large-Scale Similar Image Search

In this experiment, we test our method on the large-scale LSVRC10 dataset. The 1.2 million training images are used for training category-visual vocabulary and indexing. The 150K test images are used as queries. In LSVRC10, images show different levels of semantic relevance with queries. Therefore, we use semantic mean Average Precision (smAP) computed with Eq. (24) instead of mAP to reflect such semantic similarity in the performance measurement, *i.e.*,

$$smAP = \frac{1}{150}\sum_{i=1}^{150}\left(\frac{1}{i}\sum_{k=1}^{i}\max\left(0, 1 - \frac{\mathbb{D}(tag_q, tag_k)}{8}\right)\right) \tag{24}$$

where tag_q and tag_k are the groundtruth tags of query and the k-th returned image, respectively. $\mathbb{D}\left(tag_q, tag_k\right)$ ranges between [0,16] reflects the semantic distance between the two images. It is computed as the depth of the lowest common ancestor of tag_q and tag_k in the WordNet [7]. More details about the similarity defined by WordNet could be found in [4] and [1]. In Eq. (24), if $\mathbb{D}\left(tag_q, tag_k\right)$ is larger than 8, the two images will be considered as irrelevant.

We set the parameter ω as 0.85, which generates about 2.1 million category-visual words. To test our method, we compare the following algorithms,

- B-1: cascade category-aware visual search *with* LabelReRank.
- B-2: cascade category-aware visual search *without* LabelReRank.
- B-3: visual vocabulary tree based image retrieval with 2.2 million visual words, *i.e.*, 5 layers and 19 branches. The training set and local descriptor are identical with the ones in B-1 and B-2.
- B-4: 2048 dimensional binary semantic attribute vector extracted with the algorithm of [25]. Hamming distance is utilized as the similarity measurement. Linear search is utilized as the retrieval strategy.

The comparisons of smAP are illustrated in Fig. 12(a). From Fig. 12(a), we observe that B-1 and B-2 constantly outperform the baseline, *i.e.*, B-3. By rejecting more irrelevant categories, the performances of both B-1 and B-2 are improved. After rejecting more than 50% irrelevant categories, the performance of B-2 begins to drop. However, the performance of B-1 keeps increasing till α achieves to 70. It is reasonable to infer that this is because the computed category labels in B-1 are more accurate than the ones in B-2. This shows the validity of our LabelReRank algorithm.

By extracting high-dimensional image features and training strong classification models, the semantic attribute vector of B-4 shows strong discriminative power and substantially outperforms the baseline. We found that by rejecting 70% irrelevant categories, B-1 shows similar performance with B-4. This strongly demonstrates the validity of our method in similar image search, *i.e.*, our algorithm extracts efficient local features but shows comparable performance with the expensive state-of-the-art attribute features. In the followings, we set the α of B-1 and B-2 as 70 and 50, respectively and compare the efficiency of these algorithms.

The average time needed for retrieving one image, *i.e.*, including feature extraction, quantization, searching, and ranking, is compared in Fig. 12(b). Note that, we use the executable software package for Windows platform provided by the authors of [25] to extract features for B-4. The other features in B-1, B-2, and B-3 are implemented by ourselves with Visual C++ platform. All the experiments are executed on a single CPU core. Obviously, the semantic attribute (B-4) is the most time consuming algorithm. This is mainly because computing semantic attributes needs to extract high dimensional features and load, compute thousands of object classifiers. The most time consuming operations in B-1, B-2, and B-3 are local feature extraction and quantization, each typically takes only about 200ms on average. Besides that, the linear search in B-4 also consumes more time than the online search strategy in B-1, B-2, and B-3, which only needs to search part of the database with inverted indexes.

Because B-1, B-2, and B-3 use the same feature, similar index structure, and visual vocabularies with similar sizes, they show similar efficiency. Note that our approach does not save feature computation time compared with B-3, because they extract identical local features. B-1 and B-2 are more efficient than B-3 mainly because they discard irrelevant categories and noisy local features, hence need to scan fewer images and fewer inverted lists during retrieval. It also can be seen that, B-1 consumes more time than B-2 because it needs extra LabelReRank to update the category clue. From this experiment, we conclude that our algorithm shows both decent accuracy and efficiency in large-scale image search task. Examples of retrieval results based on B-1 are illustrated in Fig. 13.

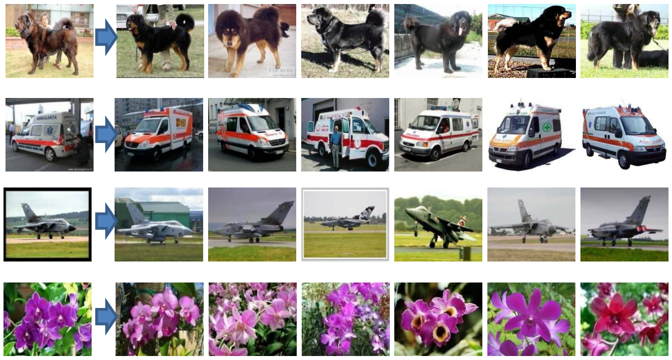


Fig. 13. Examples of retrieval result on LSVRC10. The leftmost image is the query.

E. Discussions and Future Work

From the above experiments, we can conclude that the accuracy of image annotation largely influences the memory cost, efficiency, and accuracy of our approach. With more accurate image category clue, we can reduce the redundancy of index file and scan fewer categories during retrieval. The optimal α is closely related to the accuracy of image annotation. Because of the relatively low accuracy of image annotation, we have to scan the top $(100 - \alpha)$ percent most relevant categories to ensure the recall rate. Different datasets show different difficulties for annotation, hence they reasonably have different optimal values of α. For instance, the images in UKbench are easier to annotate than the ones in LSVRC10, which are taken under different environments with larger intra-class variances. Consequently, the optimal α, *i.e.*, 0.9 in UKbench is larger than the optimal α, *i.e.*, 0.7 in LSVRC10. It also can be seen that in Fig. 12, the optimal α for B-1 is larger than the one of B-2 because B-1 gets more accurate image annotation result. Therefore, the optimal α could be inferred from the accuracy of image annotation. An interesting topic in our future work would be learning the optimal α from the accuracy of image annotation.

To improve the accuracy of image annotation, we need a category-visual vocabulary with high discriminative power. To achieve this, we remove a portion of category-visual word candidates with small STDs of the category labels. This strategy is inspired by the stop words removal strategy in traditional information retrieval, which is also commonly used in many BoWs based image retrieval works [19]. However, there is no theoretical guideline for the optimal portion of stop words. The adopted parameter in our experiments, *i.e.*, 20%, is mainly decided by our experience and experiments. Therefore, a topic that needs further study is to decide the optimal portion of stop words automatically for different applications. A possible solution is to learn this parameter on a subset of test images.

A possible strategy to further improve our approach is using supervised strategy to classify the database images for offline indexing, but still using category-visual vocabulary to estimate the category clues during online retrieval. This does not sacrifice the online retrieval efficiency but could largely improve the semantic relevance among images indexed into the same category. Hence, this strategy is potential to reduce the index file size and improve the retrieval accuracy.

Another issue that needs to discuss is the recall rate of our retrieval system. Although we discard many irrelevant categories during online retrieval, our retrieval system still gets satisfactory recall rate. This is mainly because 1) we index many images in multiple categories, making them easier to be retrieved, and 2) rejecting an appropriate portion of irrelevant categories eliminates the negative effects of irrelevant images, and makes the relevant images easier to stand out. The mAP computes the precision rate at different recall levels. Thus large mAP reasonably reflects both high precision and high recall rates. The N-S score used for UKbench is the recall rate of the top 3 returned images. Therefore, our experimental results reasonably support this conclusion.

VI. Conclusions

In this paper, we study the cascade category-aware visual search, which to the best of our knowledge, is one of the first efforts that embeds weak category clue to achieve better retrieval accuracy, efficiency, and memory consumption. In cascade category-aware visual search, we first compute category-visual image representation with category-visual words, which are discriminative and repeatable local features labeled with categories. Then, we build a hierarchical index structure to record the category clue and visual clue of database images, respectively. In the retrieval stage, the retrieval system narrows down the search space gradually by 1) filtering the noisy local features on cluttered backgrounds of query, 2) rejecting the irrelevant categories in database, and 3) discriminative visual search in relevant categories. Object search, landmark search, and similar image search on a large-scale dataset all manifest the advantages of the proposed algorithm. In addition, this cascade category-aware image search method can be easily reproduced by other researchers. These warrant further investigation of the incorporation of image categorization and large-scale visual search.

References

[1] (2010). *Large Scale Visual Recognition Challenge* [Online]. Available: http://www.image-net.org/challenges/LSVRC/2010

[2] T. Ahonen, A. Hadid, and M. Pietikinen, "Face description with local binary patterns: Application to face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 12, pp. 2037–2041, Dec. 2006.

[3] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," in *Proc. Int. World Wide Web Conf.*, 1998.

[4] J. Deng, A. C. Berg, and L. Fei-Fei, "Hierarchical semantic indexing for large scale image retrieval," in *Proc. IEEE Int. Conf. CVPR*, Jun. 2011, pp. 785–792.

[5] M. Douze, A. Ramisa, and C. Schmid, "Combining attributes and Fisher vectors for efficient image retrieval," in *Proc. IEEE Int. Conf. CVPR*, Jun. 2011, pp. 745–752.

[6] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth, "Describing objects by their attributes," in *Proc. IEEE Int. Conf. CVPR*, Jun. 2009, pp. 1778–1785.

[7] C. Fellbaum, "Wordnet: An electronic lexical database," in *Bradford Books*. Cambridge, MA, USA: MIT Press, 1998.

[8] J. Gemert, C. Veenman, A. Smeulders, and J. Geusebroek, "Visual word ambiguity," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 7, pp. 1271–1283, Jul. 2010.

[9] A. Globerson and S. Roweis, "Metric learning by collapsing classes," in *Proc. NIPS*, vol. 18. 2006, pp. 451–458.

[10] Y. Gong and S. Lazebnik, "Iterative quantization: A procrustean approach to learning binary codes," in *Proc. IEEE Int. Conf. CVPR*, Jun. 2011, pp. 817–824.

[11] H. Jégou, M. Douze, and C. Schmid, "Improving bag-of-feature for large scale image search," *Int. J. Comput. Vis.*, vol. 87, no. 3, pp. 316–336, 2010.

[12] H. Jégou, C. Schmid, H. Harzallah, and J. Verbeek, "Accurate image search using the contextual dissimilarity measure," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 1, pp. 2–11, Jan. 2010.

[13] Y. Jing and S. Baluja, "VisualRank: Applying pagerank to large-scale image search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 11, pp. 1877–1890, Nov. 2008.

[14] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. NIPS*, 2012, pp. 1106–1114.

[15] Y. Lin *et al.*, "Large-scale image classification: Fast feature extraction and SVM training," in *Proc. IEEE Int. Conf. CVPR*, Jul. 2011, pp. 1689–1696.

[16] D. Liu, G. Hua, P. Viola, and T. Chen, "Integrated feature selection and higher-order spatial feature extraction for object categorization," in *Proc. IEEE Int. Conf. CVPR*, Jun. 2008, pp. 1–8.

[17] D. G. Lowe, "Distinctive image features from scale invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[18] P. K. Mallapragada, R. Jin, and A. K. Jian, "Online visual vocabulary pruning using pairwise constraints," in *Proc. IEEE Int. Conf. CVPR*, Jun. 2010, pp. 3073–3080.

[19] D. Nistér and H. Stewénius, "Scalable recognition with a vocabulary tree," in *Proc. IEEE Int. Conf. CVPR*, vol. 2. Jun. 2006, pp. 2161–2168.

[20] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the Fisher kernel for large-scale image classification," in *Proc. ECCV*, 2010, pp. 143–156.

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

[22] X. Shen, Z. Lin, J. Brandt, S. Avidan, and Y. Wu, "Object retrieval and localization with spatially-constrained similarity measure and k-NN reranking," in *Proc. IEEE Int. Conf. CVPR*, Jun. 2012, pp. 3013–3020.

[23] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *Proc. 9th IEEE ICCV*, vol. 2. Oct. 2003, pp. 1470–1477.

[24] Q. Tian, S. Zhang, W. Zhou, R. Ji, B. Ni, and N. Sebe, "Building descriptive and discriminative visual codebook for large-scale image applications," *Int. J. Multimedia Tools Appl.*, vol. 51, no. 2, pp. 441–477, 2011.

[25] L. Torresani, M. Szummer, and A. Fitzgibbon, "Efficient object category recognition using classemes," in *Proc. ECCV*, 2010, pp. 776–789.

[26] X. Wang, M. Yang, T. Cour, S. Zhu, K. Yu, and T. X. Han, "Contextual weighting for vocabulary tree based image retrieval," in *Proc. IEEE ICCV*, Nov. 2011, pp. 209–216.

[27] Z. Wu, Q. Ke, M. Isard, and J. Sun, "Bundling feature for large scale partial-duplicated web image search," in *Proc. IEEE Int. Conf. CVPR*, Jun. 2009, pp. 25–32.

[28] G. Ye, D. Liu, I.-H. Jhuo, and S.-F. Chang, "Robust late fusion with rank minimization," in *Proc. IEEE Int. Conf. CVPR*, Jun. 2012, pp. 3021–3028.

[29] S. Zhang, Q. Huang, G. Hua, S. Jiang, W. Gao, and Q. Tian, "Building contextual visual vocabulary for large-scale image applications," in *Proc. Int. Conf. Multimedia*, 2010, pp. 501–510.

[30] S. Zhang, Q. Huang, Y. Lu, W. Gao, and Q. Tian, "Building pair-wise visual word tree for efficient image re-ranking," in *Proc. ICASSP*, 2010, pp. 794–797.

[31] S. Zhang, Q. Tian, G. Hua, Q. Huang, and W. Gao, "Descriptive visual words and visual phrases for image applications," in *Proc. 17th ACM Int. Conf. Multimedia*, 2009, pp. 75–84.

[32] S. Zhang, Q. Tian, Q. Huang, and Y. Rui, "Embedding multi-order spatial clues for scalable visual matching and retrieval," *IEEE J. Emerging Sel. Topics Circuits Syst.*, vol. 4, no. 1, pp. 130–141, Mar. 2014.

[33] S. Zhang, Q. Tian, K. Lu, Q. Huang, and W. Gao, "Edge-SIFT: Discriminative binary descriptor for scalable partial-duplicate mobile search," *IEEE Trans. Image Process.*, vol. 22, no. 7, pp. 2889–2902, Jul. 2013.

[34] S. Zhang, M. Yang, T. Cour, K. Yu, and D. N. Metaxas, "Query specific fusion for image retrieval," in *Proc. 12th ECCV*, 2012.

[35] S. Zhang, M. Yang, X. Wang, Y. Lin, and Q. Tian, "Semantic-aware co-indexing for image retrieval," in *Proc. IEEE ICCV*, Dec. 2013, pp. 1673–1680.

[36] Y. Zhang, Z. Jia, and T. Chen, "Image retrieval with geometry-preserving visual phrases," in *Proc. IEEE Int. Conf. CVPR*, Jun. 2011, pp. 809–816.

[37] W. Zhou, Y. Lu, H. Li, Y. Song, and Q. Tian, "Spatial coding for large scale partial-duplicate web image search," in *Proc. Int. Conf. Multimedia*, 2010, pp. 511–520.

Shiliang Zhang received the Ph.D. degree in computer science from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2012. He was a Visiting Researcher with the Media Analysis Group, NEC Laboratories America, Cupertino, CA, USA. He is currently a Post-Doctoral Research Fellow with the Department of Computer Science, University of Texas at San Antonio, San Antonio, TX, USA.

Dr. Zhang's research interests include large-scale image and video retrieval and multimedia content affective analysis. He was a recipient of the CCF Excellent Doctoral Dissertation from the China Computer Federation, the Outstanding Doctoral Dissertation Award from the Chinese Academy of Sciences, the President Scholarship from the Chinese Academy of Sciences, the ACM Multimedia Student Travel Grants, and the Microsoft Research Asia Fellowship in 2010. He has authored and co-authored over 20 papers in journals and conferences, including the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON MULTIMEDIA, COMPUTER VISION AND IMAGE UNDERSTANDING, the IEEE JOURNAL ON EMERGING AND SELECTED TOPICS IN CIRCUITS AND SYSTEMS, *ACM Multimedia*, and the International Conference on Computer Vision. He was a recipient of the Top 10% Paper Award at the IEEE Multimedia Signal Processing in 2011.

Qi Tian (M'96–SM'03) received the B.E. degree in electronic engineering from Tsinghua University, Beijing, China, in 1992, the M.S. degree in electrical and computer engineering from Drexel University, Philadelphia, PA, USA, in 1996, and the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana-Champaign, Champaign, IL, USA, in 2002. He is currently a Professor with the Department of Computer Science, University of Texas at San Antonio (UTSA), San Antonio, TX, USA. He took one-year faculty leave at Microsoft Research Asia, Beijing, from 2008 to 2009.

Dr. Tian's research interests include multimedia information retrieval and computer vision. He has authored over 210 refereed journal and conference papers. He received faculty research awards from Google, NEC Laboratories of America, FXPAL, Akiira Media Systems, and HP Labs. He was a recipient of the Best Paper Awards at the 2013 Pacific-Rim Conference on Multimedia (PCM), the 2013 International Workshop on Multimedia Signal Processing (MMSP), and the 2012 International Conference on Internet Multimedia Computing and Service, the Top 10% Paper Award at 2011 MMSP, the Best Student Paper at the 2006 IEEE International Conference on Acoustics, Speech, and Signal Processing, the Best Paper Candidate at 2007 PCM, and the 2010 ACM Service Award. He is the Guest Editor of the IEEE TRANSACTIONS ON MULTIMEDIA, *Journal of Computer Vision and Image Understanding*, *Pattern Recognition Letter*, *EURASIP Journal on Advances in Signal Processing*, and *Journal of Visual Communication and Image Representation*, and is on the Editorial Board of the IEEE TRANSACTIONS ON CIRCUIT AND SYSTEMS FOR VIDEO TECHNOLOGY, *Multimedia Systems Journal*, *Journal of Multimedia*, and *Journal of Machine Visions and Applications*.

Qingming Huang (M'04–SM'08) received the Ph.D. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 1994.

He was a Post-Doctoral Fellow with the National University of Singapore, Singapore, from 1995 to 1996, and was with the Institute for Infocomm Research, Singapore, as a Research Staff Member, from 1996 to 2002. He joined the Chinese Academy of Sciences, Beijing, China, under Science100 Talent Plan in 2003, and is currently a Professor with the Graduate University of Chinese Academy of Sciences, Beijing. His current research areas are image and video analysis, video coding, pattern recognition, and computer vision.

Wen Gao (M'92–SM'05–F'08) received the M.S. and Ph.D. degrees in computer science from the Harbin Institute of Technology, Harbin, China, in 1985 and 1988, respectively, and the Ph.D. degree in electronics engineering from the University of Tokyo, Tokyo, Japan, in 1991.

He was a Research Fellow with the Institute of Medical Electronics Engineering, University of Tokyo, in 1992, and a Visiting Professor with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA, in 1993. From 1994 to 1995, he was a Visiting Professor with the AI Laboratory, Massachusetts Institute of Technology, Cambridge, MA, USA. He is currently a Professor with the School of Electronic Engineering and Computer Science, Peking University, Beijing, China, and a Professor of Computer Science with the Harbin Institute of Technology.

Yong Rui (F'10) received the B.S. degree from Southeast University, Nanjing, China, the M.S. degree from Tsinghua University, Beijing, China, and the Ph.D. degree from the University of Illinois at Urbana-Champaign, Champaign, IL, USA. He is currently a Senior Director with Microsoft Research Asia, Beijing.

Dr. Rui is a fellow of the International Association for Pattern Recognition and the International Society of Optics and Photonics, and a Distinguished Scientist of the Association for Computing Machinery. He is an Associate Editor-in-Chief of the IEEE MULTIMEDIA MAGAZINE, an Associate Editor of the ACM TRANSACTIONS ON MULTIMEDIA COMPUTING, COMMUNICATION AND APPLICATIONS, and a Founding Editor of the *International Journal of Multimedia Information Retrieval*. He was an Associate Editor of the IEEE TRANSACTIONS ON MULTIMEDIA from 2004 to 2008, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGIES from 2006 to 2010, the *ACM/Springer Multimedia Systems Journal* from 2004 to 2006, and the *International Journal of Multimedia Tools and Applications* from 2004 to 2006.