This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2014.2346201, IEEE Transactions on Pattern Analysis and Machine Intelligence

Query Specific Rank Fusion for Image Retrieval

Shaoting Zhang, Ming Yang, Timothee Cour, Kai Yu, Dimitris N. Metaxas *Senior Member, IEEE*

Abstract—Recently two lines of image retrieval algorithms demonstrate excellent scalability: 1) local features indexed by a vocabulary tree, and 2) holistic features indexed by compact hashing codes. Although both of them are able to search visually similar images effectively, their retrieval precision may vary dramatically among queries. Therefore, combining these two types of methods is expected to further enhance the retrieval precision. However, the feature characteristics and the algorithmic procedures of these methods are dramatically different, which is very challenging for the feature-level fusion. This motivates us to investigate how to fuse the ordered retrieval sets, *i.e.*, the ranks of images, given by multiple retrieval methods, to boost the retrieval precision without sacrificing their scalability. In this paper, we model retrieval ranks as graphs of candidate images and propose a graph-based query specific fusion approach, where multiple graphs are merged and reranked by conducting a link analysis on a fused graph. The retrieval quality of an individual method is measured on-the-fly by assessing the consistency of the top candidates' nearest neighborhoods. Hence, it is capable of adaptively integrating the strengths of the retrieval methods using local or holistic features for different query images. This proposed method does not need any supervision, has few parameters, and is easy to implement. Extensive and thorough experiments have been conducted on 4 public datasets, *i.e.*, the *UKbench*, *Corel-5K*, *Holidays* and the large-scale *San Francisco Landmarks* datasets. Our proposed method has achieved very competitive performance, including state-of-the-art results on several datasets, *e.g.*, the N-S score 3.83 for *UKbench*.

Index Terms—Large-scale image retrieval, vocabulary tree, hashing, graph-based fusion, query specific fusion

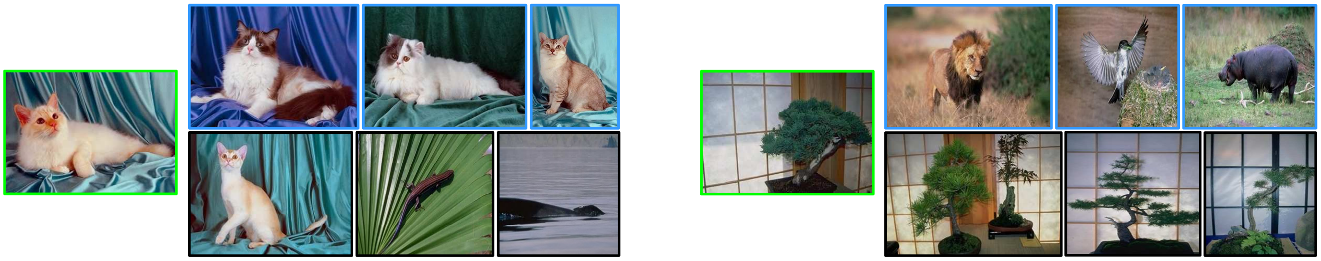
1 Introduction

Large-scale image retrieval based on visual features has long been a major research theme because of many emerging applications especially the web and mobile image search. From the perspective of image representation and methodology, most of the successful scalable image retrieval algorithms fall into two categories: 1) quantized local invariant features [24], [33] indexed by a large vocabulary tree [25]; and 2) holistic features [27], [2] indexed by compact hashing codes [35], [39]. These two approaches demonstrate *distinct* strengths in finding visually similar images. Vocabulary tree based methods are powerful in identifying near-duplicate images or regions since local features are particularly capable of attending to local image patterns or textures. On the other hand, similar textures may confuse these methods to present some candidates which appear to be irrelevant to a query. By contrast, holistic features such as color histograms or GIST features [27] delineate overall feature distributions in images, thus the retrieved candidates often appear alike at a glance but may be irrelevant.

Fig. 1 shows two illustrative cases of a success as well as a failure for either approach. The rotation invariant property of SIFT features adversely affect the retrieval precision of Fig. 1(a) by matching the curtain with the leaf, while it accurately matches the visual patterns in Fig. 1(b). On the other hand, the GIST feature considering the overall layout of the images successfully handles Fig. 1(a), but fails to deliver reasonable candidates in Fig. 1(b). Therefore, the complementary descriptive capability of local and holistic features naturally raises the question of how to integrate their strengths to yield more satisfactory retrieval results.

Although both lines of retrieval methods have been extensively studied, there is not much research effort focusing on the fusion of image retrieval methods using local and holistic features. This is due to the fact that the feature characteristics and the algorithmic procedures are dramatically different. Generally the fusion can be carried out on the feature or rank levels, *e.g.*, employing the bag-of-words (BoW) representation [33] to combine different types of features in a histogram [10], [44] or kernels [42], or combining the ordered results from different retrieval methods by rank aggregation [8], [16]. However, for a specific query image, it is quite difficult to determine online which features should play a major role in the retrieval. Moreover, it is even possible that there is no intersection among the top candidates retrieved by the local and holistic feature based methods, as shown in Fig. 1. This is very challenging for rank aggregation as it requires voting from multiple rank results. An alternative is to train a classifier to predict the retrieval quality using the similarity scores of top candidates. However, it is confronted by the issue of being sensitive to different queries and/or image databases, *e.g.*, the distributions of similarity scores may be quite different for queries with a couple or tens of relevant images. Therefore, one may need to train such a classifier for each database to somewhat "over-fit" the database statistics. These challenges prompt us to investigate a relatively principled way to evaluate *online* the quality of retrieval

Shaoting Zhang is with the Department of Computer Science at University of North Carolina at Charlotte, NC, USA.
Ming Yang is with the AI Research, Facebook Inc., CA, USA.
Timothee Cour is with Google Inc., Mountain View, CA, USA.
Kai Yu is with the Institute of Deep Learning at Baidu Inc., Beijing, China.
Dimitris N. Metaxas is with the Department of Computer Science at Rutgers University, NJ, USA.



(a) Holistic features yield more satisfactory results than local features.

(b) Local features yield more satisfactory results than holistic feature.

Fig. 1. Retrieval results of two query images (in the green boxes) in the *Corel-5K* dataset, using a holistic feature (GIST) at the first row and in the blue boxes, and BoW of local features (SIFT) at the second row and in the black boxes.

results from methods using local or holistic features and then fuse them at the rank level in an *unsupervised* way. It should also preserve the efficiency and scalability of the vocabulary tree structure and compact hashing mechanisms.

Without any supervision or relevance feedback for a retrieval set, we assume that the consensus degree among the top candidates reveals the retrieval quality. Therefore, we propose a graph-based approach to fusing and reranking retrieval results given by different methods, where the retrieval quality of an individual method is measured by the consistency of top candidates' nearest neighborhoods [45]. Given a list of ranked results by one method, *i.e.*, either the vocabulary tree-based method or the hashed holistic features, we first build a weighted graph using the constraints derived from k-reciprocal nearest neighbors [30], described later. Each edge between two nodes, *i.e.*, two candidate images, is assigned a weight based on the Jaccard similarity coefficient [12] of two neighborhoods. Such weights reflect the confidence of including the connected nodes into the retrieval results. Then, multiple graphs from different cues are fused together by appending new nodes or consolidating edge weights of existing nodes. After the candidate images from various retrieval methods are fused via graphs, we need to rank them as per the relevance and select the most similar ones. This is achieved by conducting a link analysis on the resulting graph to search for the PageRank vector [28] or the weighted maximum density subgraph. Although these two graph analysis methods are based on different assumptions, they achieve very similar and consistent results in our experiments. The precision of the resulting rank can be further improved by applying either link analysis or weighted density maximization multiple times.

The main contribution of the proposed approach is on the *unsupervised* graph-based fusion of retrieval sets given by different methods, which has three merits: 1) the retrieval quality specific to one query is effectively evaluated *online* without requiring any supervision; 2) the fusion favors the candidate images similar to a query in terms of different complementary image representations; and 3) the method can well cope with some singular cases such as little overlap of top candidates from individual cues. We have validated this method by fusing the retrieval sets based on the BoW of local features and holistic features on 4 diverse public datasets, the *UKbench*, *Corel-5K*, *Holidays* and the large-scale *San Francisco Landmarks* datasets. The evaluation shows our method consistently improves the retrieval precision and compares favorably with the recent state-of-the-art results.

The rest of the paper is organized as follows. Section 2 reviews relevant work of vocabulary trees, compact hashing, and their fusion. Section 3 presents the framework of our graph-based and query-specific fusion algorithm. Section 4 shows the experimental results on 4 public datasets and discussions. Concluding remarks are given in Section 5.

2 Related Work

Most of the scalable image retrieval algorithms fall in two threads: indexing local features by a vocabulary tree and hashing holistic features by compact binary codes. Their strengths and limitations as well as possible ways to combine them are briefly reviewed below.

2.1 Local features with vocabulary trees

Image retrieval based on the BoW of local invariant features [24], [33] has been significantly scaled up by using hierarchical vocabulary trees [25]. Such trees usually contain millions of leaf nodes attached with inverted indexes. Since this is essentially a very sparse BoW, each visual word only appears in a small number of images indexed by inverted files. Therefore, the retrieval of images containing particular visual words is very efficient. Using a tree structure demonstrates an excellent scalability in computation and precision, although it is memory consuming. For example, to utilize 10 millions of visual words, we only need a tree with 7 layers and branch factor 10, which leads to merely 7×10 inner products to quantize one descriptor. Besides vocabulary trees, product quantization [15] and its variants [26], [9]

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2014.2346201, IEEE Transactions on Pattern Analysis and Machine Intelligence

provides an alternative way to fast search approximate nearest neighbors of local invariant descriptors.

The vocabulary tree based approach has been further improved from several perspectives. Since BoW does not encode spatial information, [29] employs RANSAC as a post spatial verification, which requires the SIFT features of retrieved images to have a similar or consistent layout. [4] applies the query expansion which reissues the initial retrieval results as queries, so that the spatial constraints between the query image and each result can verify each initial return. [13] filters the local feature matching by Hamming embedding and further improves the retrieval accuracy by re-ranking with the weak geometry constraints. [40] constructs high-order features, *i.e.*, group of bundled features, and enforces robust geometric constraints within each group. [47] indexes relative spatial positions among local features in an image, which is both efficient and effective to identify false matches of local features between images. [46] quantizes spatial offsets among local features through the geometry-preserving visual phrases (GVP) and outperforms the BoW method following by a RANSAC verification.

Since images are essentially delineated by local invariant features, these methods are effective in handling image scaling, rotation, and partial occlusions, leading to a very high precision in near-duplicate image retrieval. However, if no near-duplicate image regions exist in the database, large areas of similar textures may confuse these retrieval methods and lead to irrelevant candidate images and unsatisfactory user experience. In addition, the rotation-invariant property may also cause confusion in some cases (see Fig. 1 (a)), which can be potentially corrected by considering holistic features.

2.2 Holistic features with compact hashing

Hashing-based methods focus on fast approximated nearest neighbors (ANN) search to deal with the dimensionality issue. As introduced in [35], holistic features such as color histograms, GIST [27], or image classification outcomes [18], are indexed by locality sensitive hashing (LSH) [1], which uses random projections to map data to binary codes. This method results in highly compact binary codes (*e.g.*, 128 bits), which can be efficiently compared within a large database using the Hamming distance. LSH has been extended to other similarity measures such as Mahalanobis distance [19] and p-norm distances [5]. These data-independent hashing methods may need long binary codes for a high precision in ANN, which adversely affects the efficiency. In contrast, recent research have focused on data-dependent hash functions. Many effective methods have been proposed, such as the spectral graph partitioning and hashing [39], Restricted Boltzmann Machines (RBMs) [32], semi-supervised hashing (SSH) [37] incorporating the pairwise semantic similarity and dissimilarity constraints from labeled data, and PCA hashing with iterative quantization [11]. Particularly, as suggested in [11], a random rotation on the PCA-projected features, which is optimized by iterative quantization, achieves surprisingly good performance. Supervised hashing with kernels [21] has also been proposed to leverage supervised information into hash function learning. Anchor Graph Hashing (AGH) [22], [20] has been proposed to use neighborhood graphs which reveal the underlying manifold of features, leading to a high search accuracy. Recently, fusion of features in a hashing framework has been investigated to boost the accuracy by leveraging multiple cues [43], [23], [34].

These methods leveraging compact hashing of holistic features are efficient in computation and memory usage. The computational complexity of hashing methods is usually sub-linear or even constant when using single or multiple hash tables. Even exhaustive searching is much faster than traditional methods owing to the compact binary codes and the Hamming distance metric. However, holistic features tend to be less invariant than local features, and are in general more sensitive to image transformations induced by illumination changes, scaling and pose variations. In practice, their focus on aggregated image statistics rather than fine details results in candidate images that appear roughly similar, but the retrieval precision is often lower compared to local feature based methods.

2.3 Fusion of local and holistic feature based image retrieval

Towards better retrieval performance, it is appealing to combine the strengths of complementary cues such as local and holistic features. To our best knowledge, there are not much research efforts addressing how to achieve this efficiently in the literature, although there have been several attempts combining such cues either at the feature or rank level. Combining local and holistic cues at the feature level makes it hard to preserve the efficiency and scalability induced by the vocabulary tree structure and compact hashing. Rank aggregation [8] is a straightforward solution to fusing them at the rank level, however, it requires voting from multiple rank lists and is unable to handle two lists with an empty intersection which does occasionally occur for results returned by these two distinct retrieval approaches. In either way, the key issue is how to measure and combine the cues whose effectiveness or importance varies dramatically among different query images. The closest inspiring work to ours includes [30], [17] and [41] which address different problems, *i.e.*, reranking one retrieval result by k-reciprocal nearest neighbors [30] or reranking text-based retrieval results by visual similarities employing the PageRank algorithm [28]. In contrast, we concentrate on how to fuse the retrieval results efficiently based on local and holistic features to enhance the precision.

3 Proposed Approach

3.1 Overview

To fuse the ranked retrieval results given by different methods, the critical issue is how to *automatically* measure and compare their quality, since no supervision and user relevance feedbacks are available online. The similarity scores of candidates may vary largely among queries, especially for the vocabulary tree based method, and are not comparable between different retrieval methods. Thus, a reasonable idea is to measure the consistency among the top candidates returned by one retrieval method as the retrieval quality specific to one query. Therefore, for each query image, we construct a weighted undirected graph from the retrieval results of one method, where the retrieval quality or the relevance is modeled by the weights on the edges. These weights are determined by the Jaccard similarity coefficient of two neighborhood image sets. Then we fuse multiple graphs to one and perform a localized PageRank algorithm or find the weighted maximum density subgraph centered at the query image to rerank the retrieval results. As a result, the fused retrieval results tend to be consistent in terms of different image representations.

3.2 Graph construction

Denote q the query image, d an image in the database D, and i either the query or a database image. Given a similarity function $S(\cdot,\cdot)$ between images and a retrieval method, we represent retrieval results for a query as a sorted list of candidate images with associated similarity scores $\{(d,s)\}$ where $s = S(q,d)$. We define the neighborhood of an image i as $N_k(i)$ or $N'_\epsilon(i)$, where $N_k(i)$ includes the images that are the top-k retrieved candidates using i as the query and $N'_\epsilon(i)$ includes those with $s > \epsilon$. We further define the reciprocal neighbor relation for i and i' as:

$$R_k(i,i') = i \in N_k(i') \wedge i' \in N_k(i). \tag{1}$$

As discussed in [16], [30], being the reciprocal neighbor is a reliable indication that two images are visually similar *w.r.t.* a particular image representation in a retrieval method.

For each set of retrieval results, we construct a weighted undirected graph $G = (V,E,w)$ centered at q where the nodes are the images (q and $d \in D$) and two images i, i' are linked by an edge $(i,i') \in E$ if they satisfy $R_k(i,i')$ as reciprocal neighbors. The attached edge weight w is defined as the Jaccard similarity coefficient $J(i,i')$ between the neighborhoods of i and i':

$$J(i,i') = \frac{|N_k(i) \cap N_k(i')|}{|N_k(i) \cup N_k(i')|} \tag{2}$$

$$w(i,i') = \alpha(q,i,i')J(i,i'), \tag{3}$$

where $|\cdot|$ denotes the cardinality and $\alpha(q,i,i')$ is a decay coefficient related to the number of hops to the query: let $\delta(q,i)$ be the length of the shortest path in G between q and i; we define $\alpha(q,i,i') = \alpha_0^{\max(\delta(q,i),\delta(q,i'))}$, and set $\alpha_0 = 0.8$ in all experiments. The range of edge weights is from 0 to 1, with $J(i,i') = 1$ implying that these two images share exactly the same set of neighbors, in which case we assume the two images are highly likely to be visually similar. The query q's reciprocal neighbors form the first layer in the graph whose reciprocal neighbors are expanded to the second layer *w.r.t.* q, so on so forth. The graph construction continues until either: 1) the number of nodes $|V|$ reaches a given maximum number (*i.e.*, the maximal number of images to retrieve), or 2) no more reciprocal neighbors can be found, or 3) the weights of edges become smaller than a given threshold. An illustrative example is shown in Fig. 2. Note, for holistic feature based retrieval methods, we can also employ the similarity score and the neighborhood $N'_\epsilon(i)$ in place of $N_k(i)$ to define the reciprocal neighbor relation and Jaccard similarity coefficient.

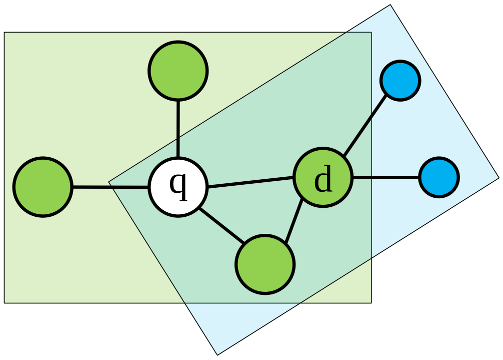


Fig. 2. An example of graph construction, where the query q links to its reciprocal neighbors (*i.e.*, q and the green discs in the green zone). d is a candidate at the first layer with its reciprocal neighbors in the blue zone, whose Jaccard coefficient to q is $3/7$ (# of nodes in the intersection divided by # of nodes in the union of the green and blue zones). The radius of the disc representing a node indicates the influence of decay coefficient α.

3.3 Graph fusion

After obtaining multiple graphs $G^m = (V^m, E^m, w^m)$ from different retrieval methods, we fuse them together to one graph $G = (V,E,w)$ with $V = \cup_m V^m$, $E = \cup_m E^m$, and $w(i,i') = \sum_m w^m(i,i')$ (with $w^m(i,i') = 0$ for $(i,i') \notin E^m$), see Fig. 3. Though the rank lists or the similarity scores in different methods are not directly comparable, their Jaccard coefficients are comparable as they reflect the consistency of two nearest neighborhoods. Without any prior, here we have to treat multiple retrieval methods equally by simply summing up the edge weights.

3.4 Graph-based ranking

Given a graph G (either obtained from a single retrieval method or by fusing multiple ones according to Sec. 3.3),

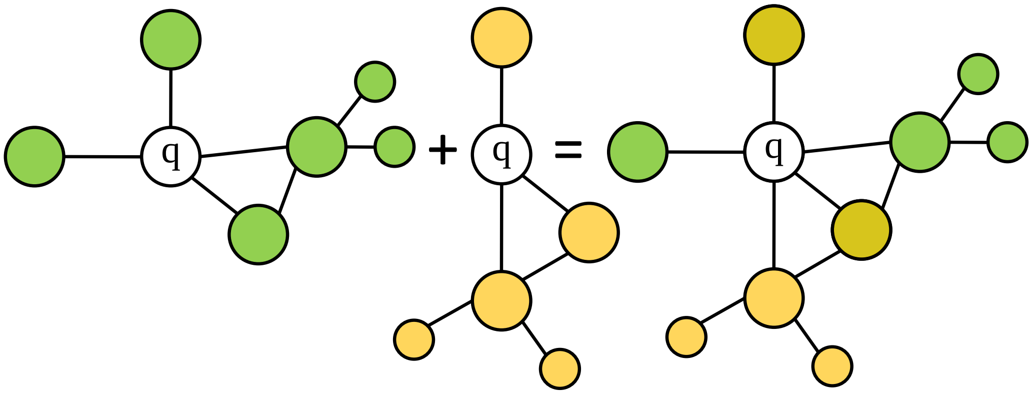


Fig. 3. Fusion of two graphs where the green and yellow graphs are derived from two different retrieval methods.

we propose two solvers to rerank the candidate images, *i.e.*, by performing the local PageRank algorithm on the edges or finding the weighted maximum density subgraph in G.

3.4.1 Ranking by the PageRank

Since the notion of well-connected nodes in G also reveals the visual consensus degree of images, we conduct a principled link analysis [28] on the whole graph G to rank according to the node connectivity. This graph G is treated as a network. Since this network is built by considering the retrieval relevance, naturally a node is more important or relevant if it has a higher probability to be visited.

We define the $|V| \times |V|$ transition matrix $\mathbf{P}$ as $P_{ii'} = w(i,i')/\deg(i)$ for $(i,i') \in E$, and 0 otherwise. It is row-stochastic, *i.e.*, each row sums to one. Consider the assumption of the *intelligent surfer model* [31], whereby a surfer probabilistically hops from node to node along the edges of G, according to the transition matrix $\mathbf{P}$. Occasionally, with a small probability $1-\beta$, the surfer jumps according to a fixed distribution over nodes π, which we set as $\pi_q = 0.99$ and uniform otherwise, where q is the index of the query node. We denote p_i^t as the probability for the surfer to be at node i at a time t and $p^t = (p_i^t)$. The equilibrium state of p, where a higher probability reflects a higher relevance to the query, is obtained by the query dependent PageRank vector as a stationary point using the power method:

$$p^{t+1} = (1-\beta)\pi + \beta \mathbf{P}^T p^t. \tag{4}$$

Once p has converged, the images are ranked according to their probabilities in p.

3.4.2 Ranking by maximizing weighted density

As the visual similarity of two images from one or more representations has been encoded in the edge weights of G, another natural idea is to search for the subgraph $G' \subset G$ containing q of a weighted maximum density, as follows:

$$G' = \underset{G'=(V',E',w)\subset G:\ q\in V'}{\operatorname{argmax}} \frac{\sum_{(i,i')\in E'} w(i,i')}{|V'|}. \tag{5}$$

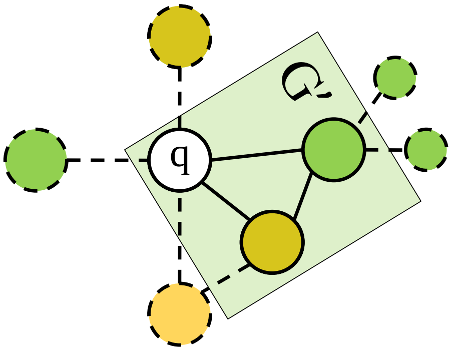


Fig. 4. Illustration of expanding G' (the green zone). Candidate nodes are connected to G', and are denoted by dash lines.

In other words, we prefer to choose nodes which can contribute more weight to the subgraph. Since edge weights are correlated with the retrieval quality, this approach selects images with potentially a higher visual similarity.

We solve Eq. (5) approximately by a greedy algorithm that grows G' iteratively, starting from $G' = (\{q\}, \emptyset, w)$. We first compute node degrees $\deg(i) = \sum_{i'} w(i,i')$ for each node i linked with q by accumulating weights from its connected edges. Then the node with the largest weight is selected to be incorporated in G'. After that, we consider all nodes connected to the current G', and select the one which can introduce the largest weight to G' (ties broken arbitrarily). Fig. 4 shows one example of determining the candidate nodes of a graph G'. G' is enlarged by applying this procedure iteratively, until a user-specified number of images is retrieved. These nodes are ranked according to their time of insertion into G'. The advantage of this ranking method is its efficiency. The computational complexity mainly depends on the connectivity (*i.e.*, the average valence of all nodes) but not the number of nodes in G, since we only check the nodes connecting to the current G'. Thus this method obtains ranking results within a similar time for different sizes of G. Although this method is not guaranteed to find a global optimum, our experiments in Sec. 4 suggest that this method achieves accurate and consistent ranking results.

3.5 Reranking for multiple times

The ranking results on the fused graph by either the PageRank or graph density maximization may still be further refined and rectified. We propose to further apply the graph-based reranking on the new retrieval result. Specifically, the new ranking of candidate images is used to build a new graph by following the same strategy as in Sec. 3.2. Then the newly-built graph is further re-ranked using either PageRank or density maximization to obtain an updated rank. The graph construction and the reranking are conducted alternately. In each iteration, the connectivity of the new graph has been further regularized and constrained by the reciprocal nearest neighbors. Therefore, the overall accuracy is expected to be improved.

This scheme has the following benefits. In the proposed graph-based reranking, we employ the Jaccard coefficient based on the reciprocal neighbor relations to measure the edge weights between images. Thus, after the reranking, if we construct a new graph based on the new rank list in the same way, the weighted graph density around the query (i.e., the sum of edge weights over the number of edges centered at the query), usually increases (it may remain the same for singular cases such as no reciprocal neighbors at all in the initial retrieval set). In addition, the images introduced as the second layer in the first round are configured as the first layer in the second round, which provides an opportunity to re-examine these second layer images by measuring the consistency among top candidate images to boost the retrieval precision. Therefore, it is intriguing to experiment whether performing the graph-based reranking again on the new graph will further improve the retrieval performance. We empirically find that conducting the graph-based reranking iteratively always improves the retrieval accuracy, even based on a single initial retrieval set. Nevertheless, the gain of the iteration over a single round of graph-based reranking depends on the initial reranking performance (or the graph density after the first reranking). For example, if the initial graph is already accurate or converged, using multiple times of our reranking may not significantly improve the accuracy compared to applying it once. In our experiments, we improved the NS-score by 0.05 on UKbench [25], precision by 1.5% on Corel-5K [7] and mAP by 0.2% on Holidays [13]. This suggests in practice we may need to perform graph-based reranking several times to reach a stable state.

3.6 Complexity and scalability

The complexity of each power method iteration in the PageRank algorithm is $O(|E|)$. In our experiments, the node valence in G is around 4-10 and the power method converges within 10 iterations. The greedy search for the maximum density subgraph is on average two times faster then the PageRank. The computational cost incurred by the proposed fusion methods is quite small given

the top candidates retrieved by different methods. In particular the running time of the proposed fusion is about 1ms regardless of the database size, and the overall query time is less than 1 second for over a million database images in our experiments. The memory overhead is determined by the number of reciprocal neighbors between images in the database which have to be pre-calculated and stored *offline* the same as [30]. The experiments in Sec. 4 demonstrate the scalability and efficiency of the original image retrieval methods are retained in the fusion method.

4 Experiments

We first describe the datasets (Sec. 4.1) and the methods (Sec. 4.2) compared in the experiments, then present the detailed evaluation results on each dataset (Sec. 4.3-Sec. 4.6), followed by the discussions about some issues and limitations (Sec. 4.7).

4.1 Datasets

We evaluate the proposed approach on 4 public datasets: the *UKbench*, *Corel-5K*, *Holidays* and *San Francisco Landmarks* (*SFLandmarks*). In the *UKbench* and *Holidays*, relevant images are near-duplicates or the same objects/scenes to the query, while, the *Corel-5K* involves category-level relevant images without any near-duplicate ones. *SFLandmarks* is a realistic large-scale dataset with a variable number of relevant images for different queries. We employ the performance measures from the original papers of these datasets and demonstrate the query specific fusion improves considerably for all these diverse datasets.

UKbench [25] includes 2,550 different objects, and each one has 4 images taken from different viewpoints and illuminations. All 10,200 images are indexed as both database images and queries. The retrieval performance is measured by 4× recall at the first 4 retrieved images, which is referred as the N-S score (maximum is 4).

Corel-5K [7] consists of 5,000 images that fall in 50 categories, such as beach, bird, jewelry, sunset, *etc.*, each containing 100 images. We use a leave-one-out method to query all 5,000 images, *i.e.*, querying every image with the remaining 4,999 images as the database images. The performance is evaluated by r-precision, *i.e.*, the precision for the top r candidates, averaged over the 5,000 queries.

Holidays [13] contains 1491 personal holiday photos undergoing various transformations. There are 500 image groups where the first image of each group is the query. The performance is measured by the mean average precision (mAP) in a leave-one-out fashion.

SFLandmarks [3] is a city-scale image database, which contains 1.06M perspective central images (PCIs) and 638K perspective frontal images (PFIs). They are generated from street-view panoramic pictures with building labels. A set of 803 images taken by camera phones is provided as queries. The performance is evaluated by the average recall rate of correct buildings *vs.* the number of candidates.

4.2 Methods

The baseline local and holistic feature based retrieval methods are denoted by the *VOC*, *GIST* and *HSV* (described below), for which we apply our graph construction (Sec. 3.2) on their retrieval results, obtaining G^{VOC}, G^{GIST} and G^{HSV}. The two proposed ranking methods are denoted by ***Graph-PageRank*** and ***Graph-density*** to generate the fused retrieval sets, which are compared with the *rank aggregation*, and a learning based fusion method, referred as *SVM-fusion*. Applying the *Graph-density* to an individual baseline obtains the *VOC-graph*, *GIST-graph* and *HSV-graph*, respectively.

VOC: We employ a variant of vocabulary tree based retrieval [25], [38] in which up to 2,500 SIFT features are detected for each image using the VLFeat library [36]. We employ a 7 layer tree with a branch factor 10. The tree is trained on 50K images in the validation set of the *ImageNet* Challenge [6] for *UKbench*, *Corel-5K* and *Holidays*, and on the PCIs and PFIs, respectively, for *SF Landmarks*, following [3].

GIST and HSV: For each image we compute the 960-dimensional GIST [27] descriptor and the 2000-dimensional HSV color histogram (using $20 \times 10 \times 10$ bins for H, S, V components). We then apply a PCA hashing method [11] to compress those to 256 bits. Retrieval is based on exhaustive search using the Hamming distance.

Rank aggregation: We use the algorithm described in [8] to combine the local and holistic retrieval results. Same as our proposed method, it does not need any supervision.

SVM-fusion: We train a linear SVM classifier that predicts which retrieval method is most appropriate for a given query, by computing a 20-dimensional input feature consisting of the top-10 normalized similarity scores for two retrieval methods. The SVM outputs binary indications about which method may achieve a higher precision. This is motivated by the observation [30] that a sharp degradation of the similarity scores may imply a confident retrieval and a long tail distribution may imply a less confident one. We employ a 5-fold cross-validation, where at the test time, we output for each query the ranked list of images from the method with a predicted higher quality.

In our graph-based fusion, the main parameter k, determining reciprocal neighborhoods, shall reflect the expected number of relevant images and the database size [30]. We set it to 5 for *UKbench* and *Holidays*, 15 for *Corel-5K*, and 30 for *SFLandmarks*, which is not sensitive to small variations. **Regarding the parameter selection, we suggest small values (e.g., 5) for datasets of nearly duplicate images, such as *UKbench*, and large values (e.g., 15) for classification datasets or large-scale datasets. The motivation is to construct graphs with sufficient nodes for fusion and reranking. With a small search range, nearly duplicate images can provide enough candidates for analysis, while othertypes of datasets usually need a large search range to ensure this.**

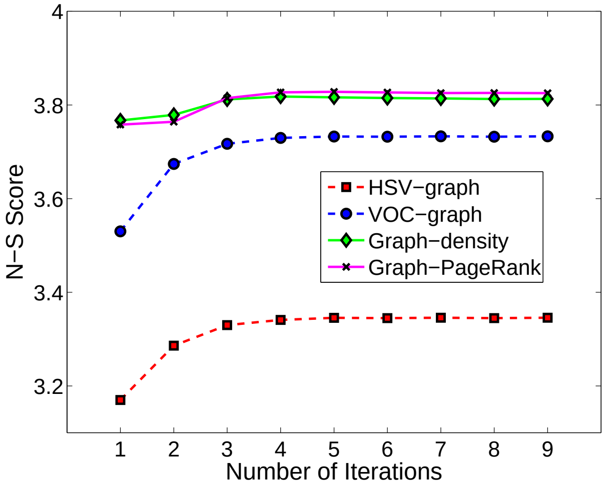


Fig. 5. N-S scores on the *UKbench* dataset, after iteratively building graphs and applying the *Graph-density* on each baseline method until converge, then the two new baseline retrieval sets are combined using the *Graph-PageRank* and *Graph-density*.

4.3 The UKbench

To show the effectiveness of our graph fusion method, we have conducted extensive experiments and provided insight analysis on this widely used *UKBench* dataset. We first compare our approach and the baselines with the state-of-the-art methods, see Table 1. We consider the fusion of the *VOC* and *HSV* retrievals, as *GIST* yields poor results here (N-S=2.21). Since the relevant images in this dataset undergo severe illumination and pose variations, *VOC* performs substantially better than holistic features. This imbalance limits the performance of rank aggregation and *SVM-fusion*. Moreover, if we employ a cross-dataset *SVM-fusion*, which is learned on the *Corel-5K* and tested on the *UKbench*, the performance (N-S=3.37) is much worse than using *VOC* only, showing that *SVM-fusion* does not generalize well across datasets. The graph-based fusion improves the baselines considerably to N-S=**3.77**, which outperforms the state-of-the-art performance N-S=3.68 in [16]. The rank aggregation was employed to combine 19 vocabulary trees [16] to achieve N-S=3.68, in contrast, we fuse just two types of features. This improvement significantly decreases the relative error rate. Indeed, this excellent performance

TABLE 1
Comparison of N-S scores on the *UKbench* dataset with recent retrieval methods and other rank fusion approaches.

Jégou *et al.* [16]	Qin *et al.* [30]	*HSV*	*VOC* [38]	*HSV graph*	*VOC graph*	*Rank aggregation*	*SVM fusion*	*Graph PageRank*	*Graph density*	*Iterative ranking*
3.68	3.67	3.17	3.54	3.28	3.67	3.47	3.56	**3.76**	**3.77**	**3.83**

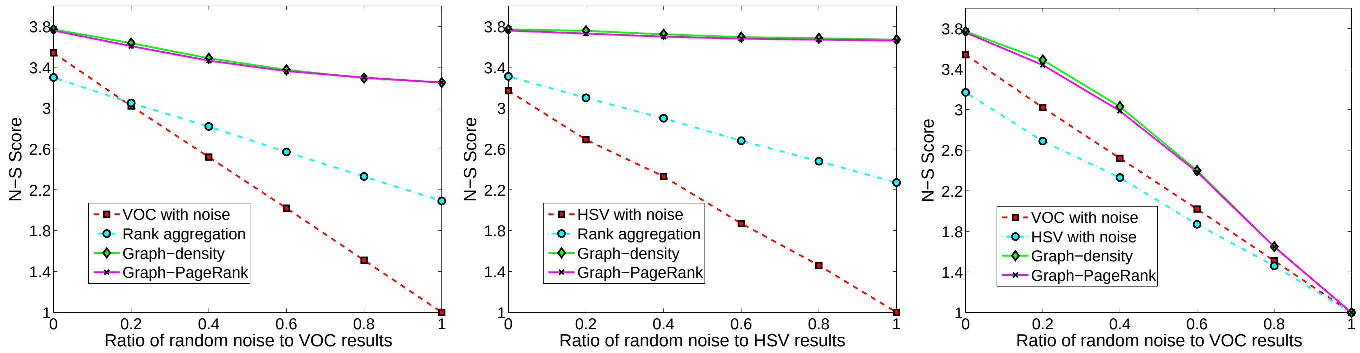


Fig. 6. **From left to right: we add random noise (from 0% to 100%) to the rank results of VOC, HSV and both, respectively. In the first two cases, the corrupted results are fused with the the other feature without noise (HSV or VOC). We compare the baselines with fusion results of the rank aggregation, the density-based and PageRank-based graph fusion methods. In the third case, we fuse two types of corrupted results.**

verifies the power of fusing local and holistic feature based retrieval methods.

A very interesting observation is that applying the *Graph-density* ranking on individual rank results can also substantially improve the retrieval precision (*i.e.*, 0.14 by the *VOC-graph* and 0.11 by the *HSV-graph*). Further, the two new ranked results can be fused by the graph-based fusion again. This iteration further improves the N-S score of each baseline method, as shown in Fig. 5. After this procedure converges to a stable state for each baseline, *i.e.*, N-S=3.73 for *VOC* and 3.35 for *HSV*, we eventually achieve the N-S score to **3.83** by the *Graph-PageRank* and **3.82** by the *Graph-density* after combining these two stable baseline ranks.

The performance of the *Graph-PageRank* and *Graph-density* are consistently close on the *UKbench*, as shown in Fig. 5. The reason is that on the *UKbench* the graphs are usually well-connected because of the near-duplicate candidates. Thus the PageRank solution by analyzing the whole graph is similar to applying the greedy search. In general, both proposed methods improve the state-of-the-art retrieval precision remarkably on this dataset, even without requiring a geometrical verification which is both time consuming and makes strong physical assumptions about near duplicates.

Fig. 6 shows that our graph fusion method is robust to the random noise. In this experiment, we add random noise to the rank results of one feature (*e.g.*, the VOC). Specifically, we replace the retrieved results with randomly assigned values. Then these corrupted rank results are fused with the results from the other feature (*e.g.*, HSV). When the ratio of the random noise increases from 0% to 100%, the N-S score of the corrupted rank results decreases to 1. Since rank aggregation is based on the voting scheme, such corrupted results adversely affect the fusion accuracy. Fig. 6 shows that rank aggregation yields much worse results than the baselines. Our graph fusion method is able to online evaluate the quality of the retrieval results from individual feature. Thus, it is very robust to such noisy retrieval results. Fig. 6 shows that its accuracy is constantly better than either of the baselines. In fact, even with 100% noise applied on HSV (VOC), the N-S score after fusing with VOC (HSV) is still around 3.54 (3.28), which is the same accuracy as applying our method on a single VOC (HSV) graph. It demonstrates that the corrupted ranks are detected and omitted automatically because of our online assessment scheme. **In addition, we also add random noise to both features, as shown in Fig. 6. In this case, the N-S score of fusion results inevitable drop to 1 when using 100% noise ratio which means both features are fully corrupted. However, our proposed fusion methods still significantly improve the retrieval accuracy until 60% noise ratio for both features, while rank aggregation fails to improve the VOC feature in such setting. This further demonstrates the efficacy of our methods in handling random noise of retrieved results.**

4.4 The Corel-5K

In this dataset, each query is associated with one hundred relevant images, so we report the precision instead of recall for the top r queries, *i.e.*, the corresponding r-precision curves in Fig. 7 and the top-1 precision in Table 2.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2014.2346201, IEEE Transactions on Pattern Analysis and Machine Intelligence

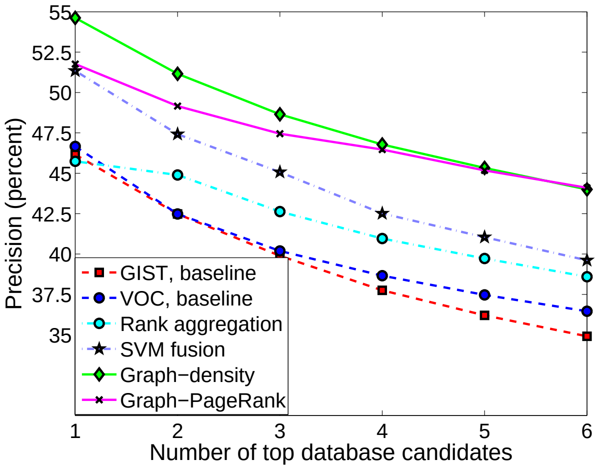


Fig. 7. The scope(r)-precision curves for the *Corel-5K* dataset.

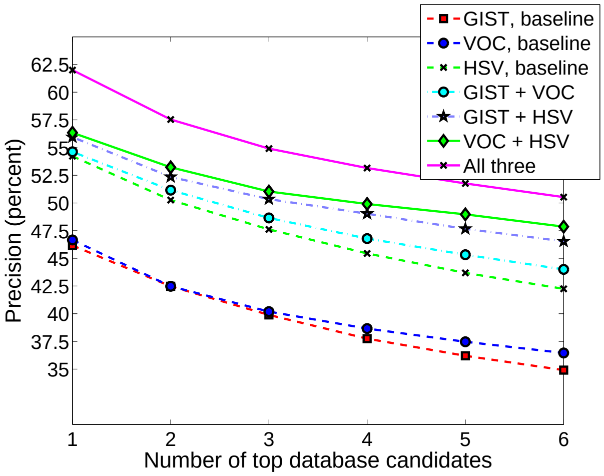


Fig. 8. The scope(r)-precision curves for the *Corel-5K* dataset when fusing results from three features.

We fuse the retrieval results of the *VOC* and *GIST* on this dataset. The top-1 precision 54.62% of the *Graph-density* is about 8% higher than either baseline method. It validates that the Jaccard similarity well reflects the retrieval quality and the graph fusion combines the strength of both baseline methods. *Graph-PageRank* does not achieve such a good precision in the top-3 retrievals. However, it becomes comparable to *Graph-density* after retrieving more images (see Fig. 7), because *Graph-PageRank* pursuits the optimization of the whole graph, while the *Graph-density* greedily finds the most relevant candidate. Thus the latter method may achieve a better performance for the first few retrievals.

The rank aggregation method improves the precision when there are some common retrieved images in both of the top candidate lists, since their ranks are promoted by the voting. However, in some cases the two rank lists may not have any overlap at all (especially for the top-1 candidate), then the aggregation cannot help.

SVM-fusion effectively improves the top-1 precision to 51.34%. However, this performance is kind of too optimistic since the number of relevant images are about the same for all the queries in the *Corel-5K* and both the *VOC* and *GIST* work equally-well, which may not hold for other databases such as the *UKbench*.

On this dataset, we also demonstrate that our method is also able to effectively fuse multiple results, i.e., *GIST*, *VOC*, and *HSV*. As shown in Fig. 8, *HSV* alone achieves top-1 precision 54.22%, which is much better than both *GIST* and *VOC*. Therefore, traditional unsupervised fusion methods may adversely affect the retrieval accuracy of *HSV*. In contrast, applying our graph fusion method significantly improves the performance of using each individual feature, *i.e.*, top-1 precision 62.0% after the fusion of these three types of features. **We also apply the iterative ranking on each baseline and then fuse them accordingly. The retrieval precision is marginally improved from 62.0% to 63.5%. Such improvement is not as significant as *UKbench*. The reason is that images in the same category of *Corel-5K* do not have strong visual similarity as *UKbench* does, since *Corel-5K* is mainly designed for the image classification instead of retrieval. Therefore, its graphs of nearest neighbors are relatively more noisy than those on *UKbench*. Nonetheless, iterative ranking still achieves the best retrieval precision among all compared methods.** As this dataset is usually used for the validation of classification methods, state-of-the-art classification methods have achieved a high accuracy using many sophisticated features and supervised learning methods (*e.g.*, relevance feedback). However, we merely employ three basic features and K-Nearest Neighbors as the baseline, since our goal is to demonstrate that this proposed graph fusion also works effectively for such a general purpose dataset, besides the datasets for near-duplicate retrieval (*e.g.*, *UKbench*).

4.5 The Holidays

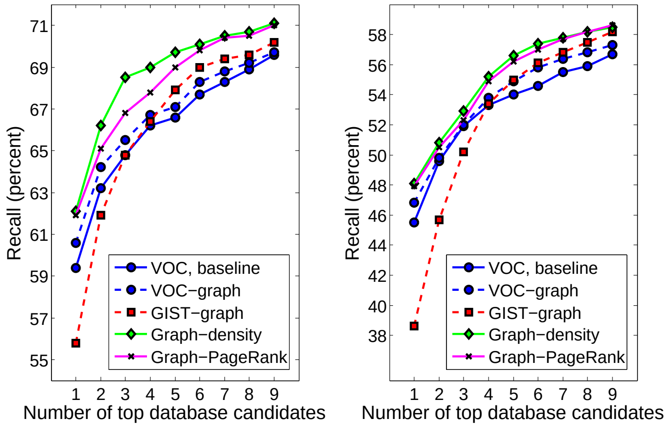
We also evaluate our proposed method on the INRIA Holidays dataset [13]. Different from the UKbench and the Corel5K datasets, each image in the INRIA Holidays may only have 1-2 relevant images. Still, we observe a consistent performance gain as on the other datasets. As shown in Table 3, the Graph-PageRank and Graph-density improve the mAP of the two baselines, *i.e.*, VOC 77.5% and HSV 62.6%, to 84.56% and 84.64%, respectively, which are also among the state-of-the-art. The improvement over VOC is more than 7% on the mAP.

In contrast, the rank aggregation and the *SVM-fusion* methods only marginally improve over the VOC by 1%. The reason is that the mAP of the HSV is about 15% lower than VOC. Such a large difference can degrade the fusion performance because the two resulted ranks may have few common retrieved images.

Again, the Graph-PageRank and Graph-density achieves very similar accuracy, *i.e.*, only 0.08% difference of the mAP. This is also consistent with the results in

TABLE 2
The top-1 precision (in %) on the *Corel-5K* dataset.

VOC	*GIST*	*VOC-graph*	*GIST-graph*	*SVM-fusion*	*Graph-PageRank*	*Graph-density*	*Three features*
46.66	46.16	51.50	50.72	51.34	**51.76**	**54.62**	**62.0**



(a) PCIs (b) PFIs

Fig. 9. Retrieval results on the *SFLandmarks*. Recall versus number of top database candidates of (a) query 803 images in the 1.06M PCIs, (b) query 803 images in the 638k PFIs.

UKbench. These two methods interpret the same graph from different perspectives, while the solutions are comparable.

On the *Holidays* dataset, iterative ranking achieves slightly better mAP than the Graph-density, i.e., 84.91%**, which indicates that applying multiple times of reranking is similar to applying it once. This is owing to the small graph structures in this dataset, i.e., 1-2 relevant images for many queries. Therefore, applying our method once already effectively discovers stable structures for these graphs, whose performance is among the state-of-the-art.**

4.6 The SFLandmarks

We study the scalability of the proposed fusion method on this real-world large-scale dataset. Concerning the online retrieval efficiency, we perform the *VOC* retrieval first, then compute the holistic feature based retrieval using the *GIST* among the top-50 candidates returned by the *VOC*. Since the query is not included in the database, we approximately determine its reciprocal neighbors based on the Jaccard similarity of the top candidates to q. Then, the two graphs of *VOC* and *GIST* are constructed and fused to generate the retrieval results. Please note that although the *GIST* graph is built upon the *VOC* results, by performing the graph fusion and ranking, the method enforces the retrieval results to be consistent in terms of different cues. Thus, this is essentially different from using the *GIST* to rank the *VOC*'s results which actually degrades *VOC*'s performance on the *SFLandmark*. In terms of memory usage, we only store the image id of the top-50 nearest neighbors in the *VOC* for the 1.7M database images which costs 340MB additional memory, a small fraction of the memory requirements for storing the inverted indexes. Although we adopt some approximations for both the *VOC* and *GIST* based retrieval, our experiments show the fusion effectively improves the performance on this large-scale problem. Moreover this is a practical setting that easily integrates with the vocabulary tree based retrieval systems.

Following the same experimental setting as in [3], we report the recall rate averaged over the 803 query images versus the number of candidates on the PCIs and PFIs separately, see Fig. 9. The recall is in terms of retrieving at least once the correct building among the top r candidates, which means multiple correct hits count as a single hit. Using the *GIST-graph* to re-rank the top-50 candidates returned by the *VOC* actually adversely affects the accuracy in the top-3 retrievals, which is probably due to the fact that local invariant features are generally more reliable than GIST in finding near-duplicate images under viewpoint changes. However, such holistic features still provide complementary information. As shown in Fig. 9, the fusion with the GIST based retrieval improves noticeably upon the *VOC*, leading to top-1 recall rates of **62.14**% for the PCIs and **48.08**% for the PFIs, which compare favorably with the method using oriented local features without GPS in [3] [1]. This validates our proposed approach as a practical retrieval method in a large-scale setting.

Illustrative fusion results on three test datasets are shown in Fig. 10, from which we observe that the query specific fusion integrates the strengths of local or holistic features adaptively.

4.7 Discussions

We discuss the running time, parameter sensitivity, implementation issues and limitations here:

1) The online query in the proposed method is very efficient, since the nearest neighborhoods are precomputed offline and the Hamming distance matching is optimized by the Intel SSE4.2 assembly. The average query time t_r in millisecond (not including the feature extraction) and the breakdown are reported in Table 4. Our *VOC* baseline takes around half second to find

[1] This statement is based on the highest recalls on the green curves in Fig.7(b) and 8(b) in [3].

TABLE 3
Comparison of the mAP (in %) on the *Holidays* dataset with recent retrieval methods and other fusion approaches.

Jégou *et al.* [13]	Jégou *et al.* [14]	*HSV*	*VOC* [38]	*Rank aggregation*	*SVM fusion*	*Graph PageRank*	*Graph density*
81.3	83.9	62.60	77.50	78.62	79.04	**84.56**	**84.64**

TABLE 4
The average query time (in *ms*) and the breakdown on the test datasets.

Dataset	# of images	*VOC*	*HSV/GIST*	*Graph-fusion*	t_r *(ms)*
UKbench	10200	85	1	< 1	87
Corel-5K	4999	76	< 1	< 1	78
Holidays	1490	72	< 1	< 1	73
PCI-SFLandmark	1,062,468	645	103	< 1	749
PFI-SFLandmark	638,090	467	64	< 1	532

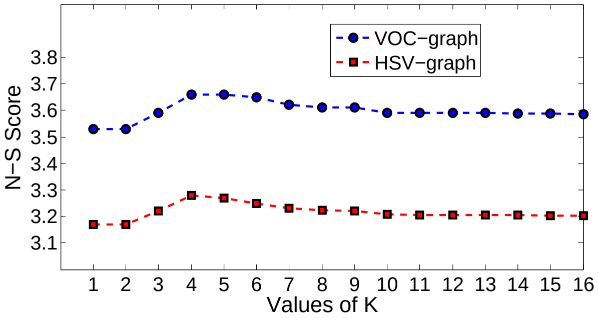


Fig. 11. The N-S scores on the *UKBench* dataset when changing parameter k.

most similar images in one million candidates. *GIST* and *HSV* need 0.1 second, without using hashing tables. Graph fusion only needs to consider the retrieved results given by the baselines, so the computational overhead is negligible.

2) This proposed graph fusion method only has one important parameter, *i.e.*, k for the k-reciprocal nearest neighbors. As mentioned in Sec. 4.2, we use smaller k values for *UKbench* and *Holidays* and larger values for *Corel5K* and *SFLandmarks*. The reason is that small-scale datasets or datasets including near-duplicates to queries usually result in a well-connected graph even with a small k, while a large k may bring some noise. In contrast, large-scale datasets or datasets containing many relevant similar images may need a large search range to build a graph. Fig. 11 shows the N-S scores on the *UKBench* dataset when changing the value of k. When $k = 1$, the method degenerates to the baseline since only the nearest neighbor of each image (*i.e.*, itself) is considered. When $k > 4$ the N-S score starts to decrease gradually. In general, the retrieval performance is not sensitive to small variations to k.

3) For certain queries, it is possible neither local nor holistic features are capable of finding relevant candidates, thus no reciprocal neighbors nor any graph can be found and built. In such cases, we just arbitrarily pick up the retrieval results given by the *VOC* or the holistic feature based retrieval without any reranking. Another corner case is that the fused graph does not have enough nodes, *i.e.*, candidates for retrieved images. In this situation, we consider the results from the relatively better baseline as additional candidates to be appended after the graph fusion results. For datasets with near-duplicate relevant images such as *UKBench*, we choose *VOC* since it is generally better than holistic feature based methods, while we can choose either one for other datasets.

4) Note that the voting-based methods are especially useful when the majority of results from different baselines are consistent, i.e., resulting ranks share many common candidates. In this case, our graph fusion and reranking may not necessarily outperform the voting scheme. However, the voting methods are not adaptive to individual queries. For example, we find it is possible that there is no intersection among the top candidates retrieved by the local and holistic feature based methods. Therefore, voting scheme is not a proper choice, which has been demonstrated in the comparison with the rank aggregation method.

5) As the nearest neighbor information is required, dynamical insertion and removal of database images require some additional treatments. One possible solution is to always keep a sufficiently large representative image set to approximate the neighborhood relations, which we leave for the future work.

6) This proposed method fuses image retrieval results in an unsupervised way. Therefore, it avoids some potential issues for supervised methods, such as over-fitting and lack of manual annotations. Moreover, this fusion method can be easily reproduced by other researchers and may serve as a plug-in in practical image retrieval systems.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2014.2346201, IEEE Transactions on Pattern Analysis and Machine Intelligence

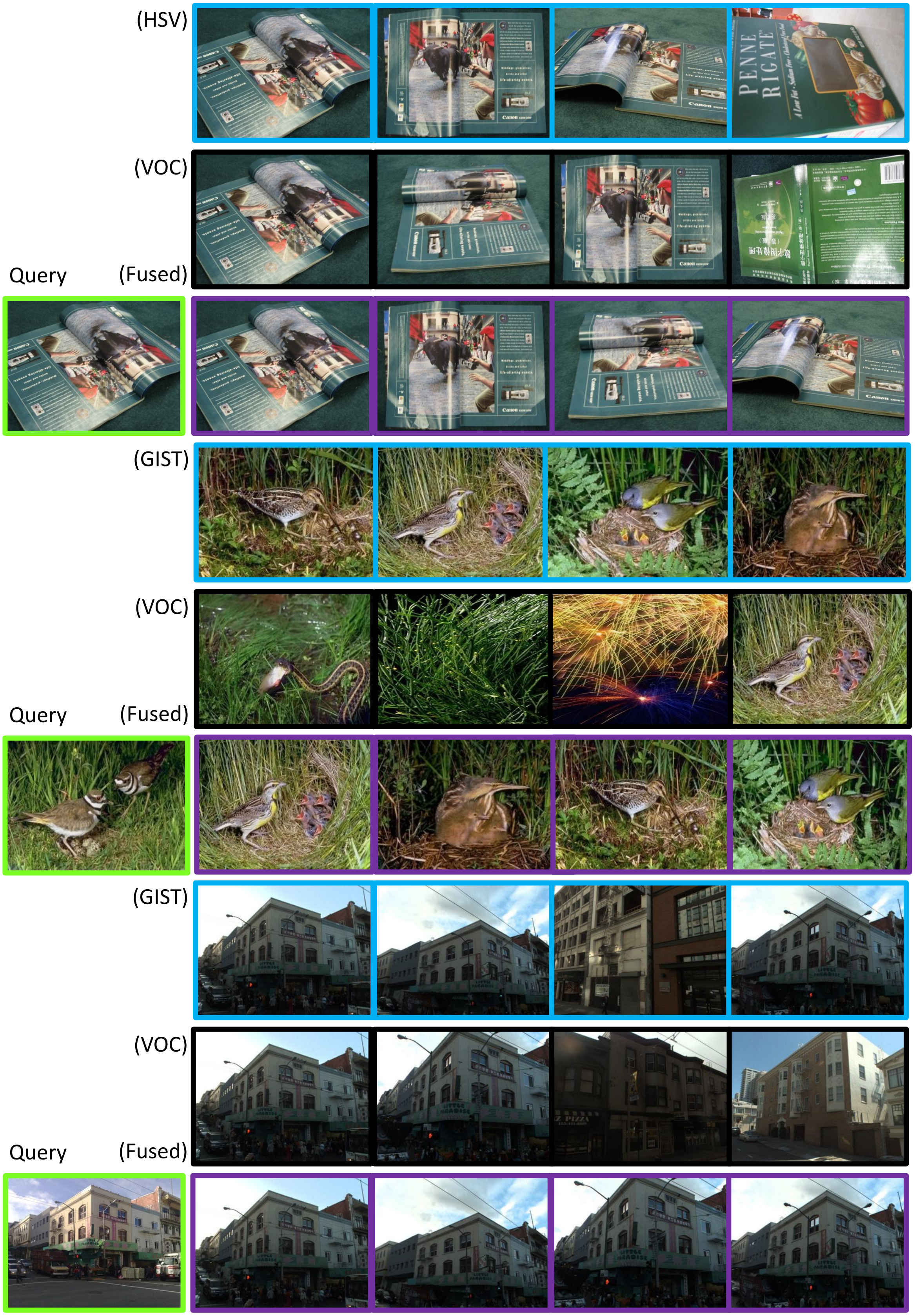


Fig. 10. Three sets of retrieval results from the *UKbench* (top), *Corel-5k* (middle), and *SFLandmarks* (bottom) datasets, respectively. Top-4 candidates are shown for the fusion results (3^{rd} row in the purple boxes) of a query (in a green box on the left), using holistic features (1^{st} row in the blue boxes), and local features (2^{nd} row in the black boxes).

5 Conclusions

In this paper, we proposed a graph-based query specific fusion of retrieval sets based on local and holistic features. In our proposed method, the retrieval quality of one set of candidate images is measured online by the consistency of the neighborhoods of top candidate images, which is specific to individual queries. Then the retrieval sets are represented as graphs and interpreted by conducting a link analysis. Such a query-specific and graph-based fusion retains the computational efficiency of the vocabulary tree based retrieval, and at the same time considerably improves the image retrieval precision on 4 diverse public datasets, including a large-scale one with over a million images. This approach does not require any supervision or relevance feedback, has few parameters and is easy to implement. These merits warrant further investigating the graph-based fusion of multiple cues for image retrieval.

6 Acknowledgement

The work was supported in part by NEC Laboratories America.

References

[1] A. Andoni and P. Indyk. Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions. In *IEEE Symposium on Foundations of Computer Science (FOCS)*, Berkeley, CA, Oct. 21-24, 2006. 3

[2] D. Cai, X. He, and J. Han. Spectral regression: a unified subspace learning framework for content-based image retrieval. In *ACM Multimedia*, Augsburg, Germany, Sept. 24-29, 2007. 1

[3] D. M. Chen, G. Baatz, K. Köser, S. S. Tsai, R. Vedantham, T. Pylvänäinen, K. Roimela, X. Chen, J. Bach, M. Pollefeys, B. Girod, and R. Grzeszczuk. City-scale landmark indentification on mobile devices. In *IEEE Conference on Computer Vision and Pattern Recognition*, Colorado Springs, CO, June 20-26, 2011. 7, 10

[4] 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 *Int'l Conference on Computer Vision*, Rio de Janeiro, Brazil, Oct. 14-17, 2007. 3

[5] M. Datar, N. Immorlica, P. Indyk, and V. S. Mirrokni. Locality-sensitive hashing scheme based on p-stable distributions. In *Annual symposium on Computational geometry*, pages 253–262. ACM, 2004. 3

[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, Miami, FL, June 20-26, 2009. 7

[7] P. Duygulu., K. Barnard, J. de Freitas, and D. Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *ECCV*, volume 4, pages 97–112, May 27 - June 2 2002. 6

[8] R. Fagin, R. Kumar, and D. Sivakumar. Efficient similarity search and classification via rank aggregation. In *ACM SIGMOD*, San Diego, CA, June 9-12, 2003. 1, 3, 7

[9] T. Ge, K. He, Q. Ke, and J. Sun. Optimized product quantization for approximate nearest neighbor search. In *IEEE Conf. on Computer Vision and Pattern Recognition*, 2013. 2

[10] P. Gehler, Sebastian, and Nowozin. On feature combination for multiclass object classification. In *Int'l Conference on Computer Vision*, Kyoto, Japan, Oct. 14-21, 2009. 1

[11] Y. Gong and S. Lazebnik. Iterative quantization: A procrustean approach to learning binary codes. In *CVPR*, Colorado Springs, CO, June 20-26, 2011. 3, 7

[12] P. Jaccard. The distribution of the flora in the alpine zone. *New Phytologist*, 11(2):37–50, 1912. 2

[13] H. Jégou, M. Douze, and C. Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *European Conference on Computer Vision*, volume 1, pages 304–317, Marseille, France, Oct. 12-18, 2008. 3, 6, 9, 11

[14] H. Jégou, M. Douze, and C. Schmid. On the burstiness of visual elements. In *IEEE Conference on Computer Vision and Pattern Recognition*, Miami, FL, June 20-25, 2009. 11

[15] H. Jégou, M. Douze, and C. Schmid. Product quantization for nearest neighbor search. *IEEE Trans. Pattern Anal. Machine Intell.*, 33(1):117–128, Jan. 2011. 2

[16] H. Jégou, C. Schmid, H. Harzallah, and J. Verbeek. Accurate image search using the contextual dissimilarity measure. *IEEE Trans. Pattern Anal. Machine Intell.*, 32(1):2–11, Jan. 2010. 1, 4, 7, 8

[17] Y. Jing and Balujia. VisualRank: Applying PageRank to large-scale image search. *IEEE Trans. Pattern Anal. Machine Intell.*, 30(11):1877–1890, Nov. 2008. 3

[18] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 26 (NIPS)*, Lake Tahoe, CA, Dec. 3-6, 2012. 3

[19] B. Kulis, P. Jain, and K. Grauman. Fast similarity search for learned metrics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(12):2143–2157, 2009. 3

[20] W. Liu, J. Wang, and S.-F. Chang. Robust and scalable graph-based semisupervised learning. *Proceedings of the IEEE*, 100(9):2624–2638, 2012. 3

[21] W. Liu, J. Wang, R. Ji, Y.-G. Jiang, and S.-F. Chang. Supervised hashing with kernels. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2074–2081. IEEE, 2012. 3

[22] W. Liu, J. Wang, S. Kumar, and S.-F. Chang. Hashing with graphs. In *International Conference on Machine Learning*, pages 1–8, 2011. 3

[23] X. Liu, J. He, and B. Lang. Multiple feature kernel hashing for large-scale visual search. *Pattern Recognition*, 47(2):748–757, 2014. 3

[24] D. G. Lowe. Distinctive image features from scale invariant keypoints. *Int'l Journal of Computer Vision*, 60(2):91–110, 2004. 1, 2

[25] D. Nistér and H. Stewénius. Scalable recognition with a vocabulary tree. In *IEEE Conference on Computer Vision and Pattern Recognition*, New York City, NY, June 17-22, 2006. 1, 2, 6, 7

[26] M. Norouzi and D. J. Fleet. Cartesian k-means. In *IEEE Conf. on Computer Vision and Pattern Recognition*, 2013. 2

[27] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *Int'l Journal of Computer Vision*, 42(3):145–175, 2001. 1, 3, 7

[28] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank citation ranking: Bringing order to the web. 1999. 2, 3, 5

[29] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *IEEE Conference on Computer Vision and Pattern Recognition*, Minneapolis, MN, June 17-22, 2007. 3

[30] D. Qin, S. Gammeter, L. Bossard, T. Quack, and L. van Cool. Hello neighbor: accurate object retrieval with k-reciprocal nearest neighbors. In *IEEE Conference on Computer Vision and Pattern Recognition*, Colorado Springs, CO, June 20-26, 2011. 2, 3, 4, 6, 7, 8

[31] M. Richardson and P. Domingos. The intelligent surfer: Probabilistic combination of link and content information in PageRank. *NIPS*, 14:1441–1448, 2002. 5

[32] R. Salakhutdinov and G. E. Hinton. Learning a nonlinear embedding by preserving class neighbourhood structure. In *International Conference on Artificial Intelligence and Statistics*, pages 412–419, 2007. 3

[33] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *Int'l Conference on Computer Vision*, Nice, France, Oct. 13-16, 2003. 1, 2

[34] J. Song, Y. Yang, Z. Huang, H. T. Shen, and R. Hong. Multiple feature hashing for real-time large scale near-duplicate video retrieval. In *ACM international conference on Multimedia*, pages 423–432, 2011. 3

[35] A. Torralba, R. Fergus, and Y. Weiss. Small codes and large image databases for recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, Anchorage, AK, June 23-28, 2008. 1, 3

[36] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. In *IEEE Conference on Computer Vision and Pattern Recognition*, Singapore, July 19-23, 2010. 7

[37] J. Wang, S. Kumar, and S.-F. Chang. Semi-supervised hashing for large-scale search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(12):2393–2406, 2012. 3

[38] X. Wang, M. Yang, T. Cour, S. Zhu, K. Yu, and T. X. Han. Contextual weighting for vocabulary tree based image retrieval. In *Int'l Conference on Computer Vision*, Barcelona, Spain, Nov. 6-13, 2011. 7, 8, 11

[39] Y. Weiss, A. Torralba, and R. Fergus. Spectral hashing. In *Advances in Neural Information Processing Systems 21 (NIPS)*, Vancouver, Canada, Dec. 8-13, 2008. 1, 3

[40] Z. Wu, Q. Ke, M. Isard, and J. Sun. Bundling feature for large scale partial-duplicated web image search. In *IEEE Conference on Computer Vision and Pattern Recognition*, Miami, FL, June 20-26, 2009. 3

[41] B. Xu, J. Bu, C. Chen, D. Cai, X. He, W. Liu, and J. Luo. Efficient manifold ranking for image retrieval. In *The International ACM SIGIR conference on Research and development in Information Retrieval*, pages 525–534. ACM, 2011. 3

[42] X.-T. Yuan and S. Yan. Visual classification with multi-task joint sparse representation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3493–3500, 2010. 1

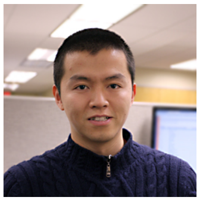
[43] D. Zhang, F. Wang, and L. Si. Composite hashing with multiple information sources. In *ACM SIGIR conference on Research and development in Information Retrieval*, pages 225–234, 2011. 3

[44] S. Zhang, J. Huang, Y. Huang, Y. Yu, H. Li, and D. N. Metaxas. Automatic image annotation using group sparsity. In *IEEE Conference on Computer Vision and Pattern Recognition*, San Francisco, CA, June 13-18, 2010. 1

[45] S. Zhang, M. Yang, T. Cour, K. Yu, and D. N. Metaxas. Query specific fusion for image retrieval. In *European Conference on Computer Vision*, pages 660–673. Springer, 2012. 2

[46] Y. Zhang, Z. Jia, and T. Chen. Image retrieval with geometry-preserving visual phrases. In *IEEE Conference on Computer Vision and Pattern Recognition*, Colorado Springs, CO, June 20-26, 2011. 3

[47] W. Zhou, Y. Lu, H. Li, Y. Song, and Q. Tian. Spatial coding for large scale partial-duplicate web image search. In *ACM Multimedia*, Florence, Italy, Oct. 25-29, 2010. 3



Shaoting Zhang is an Assistant Professor in the Department of Computer Science at the University of North Carolina at Charlotte. Before joining UNC Charlotte, he was a faculty member in the Department of Computer Science at Rutgers-New Brunswick (Research Assistant Professor, 2012-2013). He received PhD in Computer Science from Rutgers in 01/2012, M.S. from Shanghai Jiao Tong University in 2007, and B.E. from Zhejiang University in 2005. His research is on the interface of medical imaging informatics, large-scale visual understanding and machine learning.

Ming Yang is a research scientist in the AI Research at Facebook Inc. since 2013. He received the B.E. and M.E. degrees in electronic engineering from Tsinghua University, Beijing, China, in 2001 and 2004, respectively, and the PhD degree in electrical and computer engineering from Northwestern University, Evanston, Illinois, in June 2008. From 2004 to 2008, he was a research assistant in the computer vision group of Northwestern University. After his graduation, he joined NEC Laboratories America, Cupertino, California as a research staff member. His research interests include computer vision and machine learning, in particular, face recognition, large-scale image retrieval, and intelligent multimedia content analysis. He is a member of the IEEE.

Timothee Cour is a software engineer at Google, Mountain View, working on computer vision and machine learning in streetview. Prior to that he was a research scientist at NEC-Labs (Silicon Valley) in the media analytics department. He finished his postdoc in the Willow group of INRIA / Ecole Normale Superieure. He obtained his PhD in Computer Science under the supervision of Ben Taskar at the University of Pennsylvania, Philadelphia, where he also obtained his MS with Jianbo Shi. Before that he was an undergrad at the Ecole Polytechnique, France, in applied mathematics.

Kai Yu received the PhD degree in computer science from the University of Munich in 2004. He is now the deputy director of the Institute of Deep Learning at Baidu. This work was done when he was the head of the Media Analytics Department at NEC Laboratories America, where he managed an R&D division working on image recognition, multimedia search, video surveillance, sensor mining, and human-computer interaction. He has published more than 70 papers in top-tier conferences and journals in the area of machine learning, data mining, and computer vision. He has served as area chair for top-tier machine learning conferences, e.g., ICML and NIPS, and taught an AI class at Stanford University as a visiting faculty member. Before joining NEC, he was a senior research scientist at Siemens. He is a member of the IEEE.

Dimitris Metaxas is a distinguished professor (Professor II) in the Computer Science Department at Rutgers University. He is directing the Computational Biomedicine Imaging and Modeling Center (CBIM). He received the B.E. degree from the National Technical University of Athens Greece in 1986, M.S. degree from the University of Maryland in 1988, and Ph.D. from the University of Toronto in 1992. He has been conducting research toward the development of formal methods upon which computer vision, computer graphics, and medical imaging can advance synergistically.