Democratic Diffusion Aggregation for Image Retrieval

Zhanning Gao, Jianru Xue, *Member, IEEE*, Wengang Zhou, Shanmin Pang, and Qi Tian, *Fellow, IEEE*

***Abstract*—Content-based image retrieval is an important research topic in the multimedia field. In large-scale image search using local features, image features are encoded and aggregated into a compact vector to avoid indexing each feature individually. In the aggregation step, sum-aggregation is wildly used in many existing works and demonstrates promising performance. However, it is based on a strong and implicit assumption that the local descriptors of an image are identically and independently distributed in descriptor space and image plane. To address this problem, we propose a new aggregation method named *democratic diffusion aggregation* (DDA) with weak spatial context embedded. The main idea of our aggregation method is to re-weight the embedded vectors before sum-aggregation by considering the relevance among local descriptors. Different from previous work, by conducting a diffusion process on the improved kernel matrix, we calculate the weighting coefficients more efficiently without any iterative optimization. Besides considering the relevance of local descriptors from different images, we also discuss an efficient *query fusion* strategy which uses the initial top-ranked image vectors to enhance the retrieval performance. Experimental results show that our aggregation method exhibits much higher efficiency (about $\times$ 14 faster) and better retrieval accuracy compared with previous methods, and the *query fusion* strategy consistently improves the retrieval quality.**

***Index Terms*—Democratic diffusion aggregation (DDA), image retrieval, query fusion.**

I. Introduction

With the explosive growth of Web images, large-scale content-based image retrieval (CBIR) has been an important research focus for both multimedia and computer vision communities [6], [42], [45], [50]–[53]. As shown in Fig. 1, the general CBIR pipeline usually consists of the off-line stage and the on-line stage. Local feature extraction, embedding and aggregation steps are shared for both stages to build the image representation. After that, at the off-line stage, an indexing structure is constructed to apply efficient searching at the on-line stage. To further refine the initial searching result, a re-ranking strategy is often employed after searching step. In this paper, we propose a novel aggregation method to build a compact image representation for large-scale image retrieval. In addition, we also discuss a simple yet effective re-ranking strategy based on our new image representation.

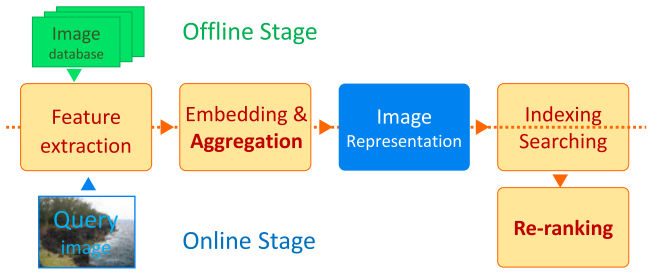


Fig. 1. General framework of the CBIR system. In this paper, we focus on the aggregation step. In addition, we also discuss a simple yet effective re-ranking strategy based on our compact image representation.

Image representation is one of the key issues for large-scale CBIR. Bag-of-visual-word (BOW) representations based on local descriptors such as SIFT [32] are wildly used in CBIR systems [8], [20], [41]. However, there are two nontrivial issues led by the BOW based methods, i.e., computational efficiency and memory cost [22], [31], [39], especially in the retrieval scenario with beyond billion scale images and limited resources [24]. Recently, several encoding methods are proposed to aggregate local descriptors into a compact vector, such as Fisher vectors [37], Vector of locally aggregated descriptors (VLAD) [24], and T-embedding method [25]. In addition, a compact descriptors for visual search (CDVS) standard (formally known as MPEG-7, Part 13) was published by ISO on August 25th, 2015 [13]. The CDVS standard defines a framework for compact and format independent image representation construction. Enhanced by approximate nearest neighbor search algorithms, e.g., product quantization [23], inverted multi-index [3], or hashing method [14], [28], these methods can significantly improve the efficiency of the memory consumption and computational load towards Web-scale image search based on visual content. We focus on the compact image representation in this paper.

Generally, based on local features (e.g. SIFT [32]), there are three key steps in compact image representation construction: (i) feature extraction, (ii) embedding/coding and (iii) aggregation/pooling [17], [24], [25]. In the feature extraction step,

Manuscript received July 08, 2015; revised February 22, 2016 and May 01, 2016; accepted May 05, 2016. Date of publication May 13, 2016; date of current version July 15, 2016. This work was supported by National Basic Research Program of China (973 Program) Project 2012CB316400 and by the National Natural Science Foundation of China (NSFC) under Contract 61273252. The work of W. Zhou was supported in part by the NSFC under Contract 61472378 and in part by the Anhui Provincial Natural Science Fundation under Contract 1508085MF109. The work of Q. Tian was supported in part by the NSFC under Contract 61429201, in part by the ARO under Grant W911NF-15-1-0290 and Grant W911NF-12-1-0057, and in part by the Faculty Research Gift Awards by NEC Laboratories of America and Blippar. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Adrian Munteanu.

Z. Gao, J. Xue, and S. Pang are with the Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: gaozn1990@stu.xjtu.edu.cn; jrxue@mail.xjtu.edu.cn; pangshanminn@sina.com).

W. Zhou is with the Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei 230027, China (e-mail: zhwg@ustc.edu.cn).

Q. Tian is with the Department of Computer Science, University of Texas at San Antonio, San Antonio, TX 78249 USA (e-mail: qitian@cs.utsa.edu).

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

Digital Object Identifier 10.1109/TMM.2016.2568748

the image is represented with a set of local descriptors. In the embedding step, each local descriptor is mapped into a high-dimensional vector. The aggregation step integrates all the embedded vectors of an image into a single vector which obtains a compact representation for image retrieval. After the popular Fisher vectors and VLAD, many efforts focus on the feature extraction and the embedding steps [2], [25], [43] to gain more discriminative embedded vectors. For instance, Ge *et al.* [17] explore different local descriptors based on sparse coding method. Tolias *et al.* [43] encode the orientation of local descriptor in the embedding step to achieve a geometric-aware embedding strategy. Different from VLAD [24], T-embedding [25] computes the residual vectors with each visual word instead of the nearest one and localizes them via a triangulation strategy. However, in the aggregation step, most methods simply sum the embedded vectors to a single vector, i.e., sum-aggregation, which ignore the relevance among local descriptors. Thus, the frequently occurring local descriptors[1] will dominate in the final representation than rarely occurring local descriptors [34]. However, the frequently occurring local descriptors are not necessarily the most informative ones. Instead, these descriptors may reduce the discrimination of the image representation because of over count. Therefore, it is crucial to balance the influence of frequent and rare local descriptors in the aggregation step.

In this paper, we present a novel aggregation method named *democratic diffusion aggregation* (DDA) to address the influence of frequent and rare local descriptors in the aggregation step. In order to balance the influence of local descriptors in image representation, the main idea of our aggregation method is to re-weight embedded vectors of an image before sum-aggregation. The weighting coefficients are calculated efficiently from pairwise similarities between local descriptors of an image. We also embed weak spatial context to depress visual co-occurrence [19] caused by local feature detector. Previous work [25], [34] usually involve an iterative optimization problem to calculate the weighting coefficients. However, by conducting a diffusion process, our method can calculate the weighting coefficients efficiently via a simple closed-form solution without any iterative optimization. Furthermore, with this diffusion process and weak spatial context of local descriptors, we can obtain much more reliable pairwise similarities between local descriptors. Hence, besides the high efficiency, the image representation with DDA also achieves better retrieval performance comparing with previous work.

Besides the exploration of local descriptors from one image, the relevance of descriptors between query image and initial top-ranked images can also improve the image retrieval accuracy at the re-ranking step. In BOW based image search techniques [7], [22], [36], [39], query expansion (QE) [8] significantly boosts the retrieval performance by adding relevant visual words to the query. To suppress the impact of false positives, geometric verification is usually involved in QE. However, for the compact image representations, the spatial context information of local features is no longer available. Therefore, it is infeasible to apply geometric verification to our new image representation. To leverage the idea of QE for the compact image representation, we also discuss a re-ranking strategy, i.e., query fusion. It is performed by averaging the query vector and the top-ranked vectors to generate a new query. Experimental results demonstrate the steady improvement with our re-ranking strategy.

This work is an extension of our previous paper [15]. The key difference lies in the following aspects. Firstly, we propose a new aggregation method named DDA in Section III. By conducting a graph diffusion process on the modified kernel matrix, we obtain a simple closed-form solution to estimate the weighting coefficients. Secondly, we add an experimental study on the influence of the local features number per image. We also add evaluation of our methods on a large scale dataset. At last, some state-of-the-art encoding methods are added in the comparison part. We also present the complexity analysis of our aggregation methods compared with other encoding methods which further exhibit the advantages of our aggregation methods.

The paper is organized as follows. Section II discusses related work about the compact image representation and re-ranking methods. Then, we present our aggregation methods in Section III, and *query fusion* scheme in Section IV. Experimental results are then provided in Section V. Finally, we conclude the paper in Section VI.

II. Related Work

Image representation is one of the key issues for the CBIR. With the explosive growth of Web images, the compact image representation has attracted great attention recently. To construct more discriminative image representation, Gong *et al.* employ multi-scale deep convolutional activation features instead of hand-crafted feature combined with VLAD [24] to produce the compact image representation. Tolias *et al.* [43] encode the orientation of local descriptor in the embedding step to achieve a geometric-aware embedding strategy. Different from VLAD [24], T-embedding [25] computes the residual vectors with each visual word instead of the nearest one and localizes them via a triangulation strategy. The contributions of these methods mainly focus on the feature extraction and the embedding steps. In the aggregation step, the sum-aggregation is applied for most of the embedding methods, including the original VLAD [24], Fisher vectors [37], and the CDVS standard based on scalable compressed Fisher vector representation [29]. However, sum-aggregation is based on a strong assumption that local descriptors of an image are identically and independently distributed (i.i.d.) [5], [9]. Thus, the frequently occurring local descriptors will be more influential than rarely occurring local descriptors and may reduce the discrimination of the image representation because of over count.

To address this problem, many strategies are proposed for specific embedding methods. For example, based on BOW model, Jégou *et al.* [21] use IDF-like weighting method to address the burstiness problem. Zheng *et al.* [49] re-estimate the visual word frequency by l_p-norm pooling in an offline manner. For VLAD-like representations, power normalization and PCA rotation [24], [37] are effective methods to tackle frequent descrip-

[1] In BOW model, frequently occurring local descriptors usually lead to burstiness [21] and co-occurrence [19] of visual words.

tors. All of these strategies can be categorized as post-methods for sum-aggregation to modify the i.i.d. assumption of local descriptors. In this paper, we propose an alternative aggregation method to balance the influence of local descriptors in the aggregation step. Different with these post-methods after sum-aggregation, we propose to re-weight the embedded vectors before sum-aggregation to suppress the frequent descriptors. In addition, the experiments show that combining our aggregation method with power normalization and PCA rotation can further improve the retrieval performance.

Recent relevant work [25], [34] have explored similar idea in image retrieval and image classification tasks, respectively. The generalized max pooling (GMP) [34] and the democratic aggregation (DA) [25] propose to weight the embedded vectors based on the similarity among embedded vectors to equalize the influence of frequent and rare descriptors. However, our aggregation method achieves better performance and high-efficiency. We summarize the difference with previous work and our contributions in the aggregation step as follows. (i) Instead of embedded vectors, we employ the original local descriptors, i.e., RootSIFT [1], to estimate the weighting coefficients. Thus, the weighting coefficients can be estimated independent of the embedding step. (ii) In addition, we also embed weak spatial context to depress visual co-occurrence [19] during the coefficients estimation. (iii) Previous work [25], [34] usually involve an iterative optimization problem to calculate the weighting coefficients, while our method can calculate the weighting coefficients efficiently without any iterative optimization by conducting a diffusion process.

Motivated from the success in the information retrieval, Chum *et al.* [8] translate the QE principle to the CBIR. QE and its variants are wildly used for the BOW based image representation [7], [22], [36], [39]. Geometric verification is usually adopted before expanding features to the new query due to its sensitivity to false positives. Therefore, it is not suited to our compact representation since no geometric information of local descriptors is saved after the aggregation step. A notable exception is Hamming query expansion (HQE) [44] which is effective without using any geometric information. Instead of geometric verification, HQE use a much lower Hamming embedding threshold to filter out most of the false matches when expanding features to the new query. It is much faster than geometric verification but still needs the binary signature of each local descriptors which is not available for our compact representation either. As we can see, for QE and HQE, the key point to make it effective is to reduce false matches. Fortunately, the similarity metric of T-embedding vectors [25] is much robust for false positive. Therefore, the verification part is no longer needed for our compact representation.

III. Efficient DA Methods

In this section, we first briefly introduce the background of T-embedding and DA and analyze its computational complexity in Section A. Then, we present our efficient aggregation method based on T-embedding vectors in Section B and C. In Section B, we propose a simpler yet more effective strategy to obtain the kernel matrix. After that, in Section C, we develop a simple closed-form solution without any iterative optimization to compute the weighting coefficients more efficiently, i.e., the DDA method, which achieves much higher efficiency and better retrieval performance comparing with previous work [25].

A. T-embedding and DA

1) T-embedding: Let $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}_i, \mathbf{x}_i \in \mathbb{R}^d, \|\mathbf{x}_i\| = 1$, be a set of local descriptors from an image to be described. T-embedding maps each local descriptor into a high dimensional vector $\phi_\triangle(\mathbf{x}_i)$ with codebook $\mathcal{C} = \{\mathbf{c}_1, \ldots, \mathbf{c}_{|\mathcal{C}|}\}_j, \mathbf{c}_j \in \mathbb{R}^d$, which is learned by k-means from an independent training set [25]. The T-embedding function can be derived as

$$\phi_\triangle(\mathbf{x}_i) = \Sigma^{-1/2}(\mathbf{R}(\mathbf{x}_i) - \mathbf{R}_0) \tag{1}$$

$$\mathbf{R}(\mathbf{x}_i) = \left[\frac{\mathbf{x}_i - \mathbf{c}_1}{\|\mathbf{x}_i - \mathbf{c}_1\|}^\top, \ldots, \frac{\mathbf{x}_i - \mathbf{c}_{|\mathcal{C}|}}{\|\mathbf{x}_i - \mathbf{c}_{|\mathcal{C}|}\|}^\top \right]^\top \tag{2}$$

where $\mathbf{R}_0 = \mathbb{E}_Z[\mathbf{R}(\mathcal{Z})]$ and $\Sigma^{-1/2}$ are trained by an independent set $\mathcal{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_h\}_i, \mathbf{z}_i \in \mathbb{R}^d, \|\mathbf{z}_i\| = 1$.

$$\Sigma^{-1/2} = \mathrm{diag}({\gamma_1}^{-1/2}, \ldots, {\gamma_{|\mathcal{C}|\times d}}^{-1/2})\mathbf{P}^\top \tag{3}$$

where $\mathbf{P}$ are the eigenvectors of the covariance matrix, which is computed by the centered training set $\mathbf{R}(\mathcal{Z}) - \mathbf{R}_0$. γ_i is the eigenvalue of the ith eigenvector. In fact, (1) performs a whitening operation [19], [30] on $\mathbf{R}(\mathbf{x}_i)$. To reduce the variance of the cosine similarity between unrelated T-embedding vectors, as suggested in [25], we discard the first d components of $\phi_\triangle(\mathbf{x}_i)$ associated with the largest eigenvalues. The final T-embedding vector is l_2-normalized and its dimensionality is $\mathrm{D} = (|\mathcal{C}| - 1) \times d$. Since T-embedding based encoding methods indicate satisfactory performance based on hand-crafted local feature [25], we employ T-embedding in our embedding step. In feature extraction step, we use RootSIFT [1] detected by Hessian-affine detector as the local descriptor of the image.

2) Sum and DA: Let $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ and $\mathcal{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_m\}$ be two sets of local descriptors from two images. After T-embedding step, the similarity of these two images can be computed by a match kernel $\mathcal{K}$ [4] of the form

$$\mathcal{K}(\mathcal{X}, \mathcal{Y}) = \mathbf{1}_n \mathbf{K}(\mathcal{X}, \mathcal{Y}) \mathbf{1}_m^\top \tag{4}$$

where $\mathbf{1}_m = \underbrace{[1, \ldots, 1]}_{m}$ and

$$\mathbf{K}(\mathcal{X}, \mathcal{Y}) = \begin{bmatrix} \phi_\triangle(\mathbf{x}_1)^\top \phi_\triangle(\mathbf{y}_1) & \ldots & \phi_\triangle(\mathbf{x}_1)^\top \phi_\triangle(\mathbf{y}_m) \\ \vdots & \ddots & \vdots \\ \phi_\triangle(\mathbf{x}_n)^\top \phi_\triangle(\mathbf{y}_1) & \ldots & \phi_\triangle(\mathbf{x}_n)^\top \phi_\triangle(\mathbf{y}_m) \end{bmatrix}. \tag{5}$$

For sum-aggregation, the final vector is derived from the set $\{\phi_\triangle(\mathbf{x}_1), \ldots, \phi_\triangle(\mathbf{x}_n)\}$ by

$$\psi_s(\mathcal{X}) = \sum_{\mathbf{x} \in \mathcal{X}} \phi_\triangle(\mathbf{x}). \tag{6}$$

Thus, the match kernel can be computed by

$$\mathcal{K}(\mathcal{X}, \mathcal{Y}) = \sum_{\mathbf{x} \in \mathcal{X}} \sum_{\mathbf{y} \in \mathcal{Y}} \phi_\triangle(\mathbf{x})^\top \phi_\triangle(\mathbf{y}) = \psi_s(\mathcal{X})^\top \psi_s(\mathcal{Y}). \tag{7}$$

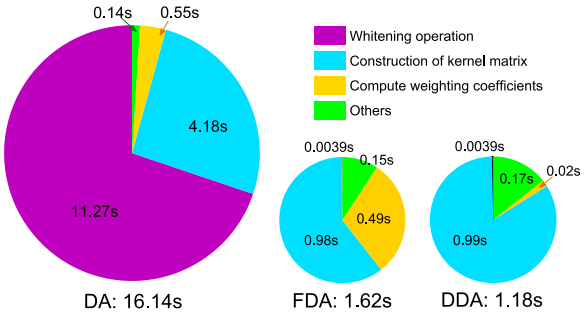


Fig. 2. CPU timings of each stage in original DA, fast democratic aggregation (FDA), and DDA. The results are reported on the Holidays dataset. $|\mathcal{C}| = 64$, $\mathrm{D} = (|\mathcal{C}| - 1) \times d = 8064$.

Instead of computing the summation of the $\phi_{\triangle}(\mathbf{x})$, a DA method [25] is proposed to weight the summation

$$\psi_d(\mathcal{X}) = \sum_{\mathbf{x}_i \in \mathcal{X}} \lambda_{\mathbf{x}_i} \phi_{\triangle}(\mathbf{x}_i). \tag{8}$$

The set of weighting coefficients $\{\lambda_{\mathbf{x}_1}, \ldots, \lambda_{\mathbf{x}_n}\}$ is used to equalize the contribution of T-embedding vectors, i.e., modifying the original match kernel $\mathcal{K}$ to a *democratic* kernel. The weighting coefficients are obtained by solving the following equation:

$$\mathbf{\Lambda}\mathbf{K}(\mathcal{X}, \mathcal{X})\mathbf{\Lambda}\mathbf{1}_n^\top = c \cdot \mathbf{1}_n^\top \tag{9}$$

where $\mathbf{\Lambda} = \mathrm{diag}(\lambda_{\mathbf{x}_1}, \ldots, \lambda_{\mathbf{x}_n})$. After setting all negative elements of $\mathbf{K}$ to 0 and $c = 1$, (9) can be solved by the Sinkhorn scaling algorithm [26].

Despite promising retrieval accuracy, DA suffers high computational complexity. Fig. 2 presents the timings (we present CPU time which accumulates all active threads) of different stages in DA step. The results are obtained by the MATLAB code released by [25] running on the Holidays [20] dataset. The DA combined with T-embedding vectors can be derived by (1) and (8) as

$$\psi_d(\mathcal{X}) = \sum_{\mathbf{x}_i \in \mathcal{X}} \lambda_{\mathbf{x}_i} \Sigma^{-1/2} \left(\mathrm{R}(\mathbf{x}_i) - \mathrm{R}_0\right). \tag{10}$$

$\{\lambda_{\mathbf{x}_1}, \ldots, \lambda_{\mathbf{x}_n}\}$ relies on the kernel matrix $\mathbf{K}$ that is computed by (5). Different from sum-aggregation which can be computed efficiently by

$$\psi_s(\mathcal{X}) = \Sigma^{-1/2} \sum_{\mathbf{x}_i \in \mathcal{X}} \left(\mathrm{R}(\mathbf{x}_i) - \mathrm{R}_0\right). \tag{11}$$

(10) cannot be optimized because of the coupling between T-embedding step and construction of $\mathbf{K}$. Moreover, the construction of $\mathbf{K}$ is also time consuming considering the high dimensionality of the T-embedding vector. As shown in Fig. 2, the bottlenecks of the DA are the whitening operation and the construction of $\mathbf{K}$, especially the whitening operation of projecting each residual vector separately with the matrix $\Sigma^{-1/2}$.

B. Construction of the Kernel Matrix

When weighting the T-embedding vectors of an image, intuitively, we have to use those vectors to calculate the kernel matrix and estimate the weighting coefficients. However, essentially, the kernel matrix $\mathbf{K}$ reflects the interaction between each pair of local descriptors of an image, which is similar to the affinity matrix [35] constructed in spectral clustering algorithm. It inspires us to employ the original local descriptors (after whitening) to re-construct the kernel matrix. In addition, to further depress the co-occurrence, especially the artificial visual co-occurrences [19], we also embed the weak spatial context of local descriptors to the kernel. Motivated by the above discussion, our new kernel matrix is formulated as

$$\mathbf{K}_{\mathrm{new}} = (1 - \rho) \cdot \mathbf{K}_{\mathrm{SIFT}} + \rho \cdot \mathbf{K}_{\mathrm{SP}}. \tag{12}$$

The construction of $\mathbf{K}_{\mathrm{SIFT}}$ and $\mathbf{K}_{\mathrm{SP}}$ will be presented in the following. The influence of mixing coefficient ρ will be discussed in experiment section.

1) The Kernel Matrix $\mathbf{K}_{\mathrm{SIFT}}$ With Whitened RootSIFT: We first explain why the original local descriptors can be employed to calculate the kernel matrix $\mathbf{K}_{\mathrm{SIFT}}$, and then discuss the necessity of the whitening operation on local descriptors.

A key property [25] of T-embedding vector is that the inner product between the embedding vectors is an excellent similarity metric, considering the fact that the inner product value is close to zero with high probability for unrelated embedding vectors while much greater than zero for related embedding vectors from matched local descriptors. Although this property is crucial to obtain the kernel matrix, mapping local descriptors to T-embedding vectors cannot enhance the distinctiveness of local descriptors. Therefore, it is unnecessary to map local descriptors to high-dimensional vectors (T-embedding vectors). Considering the fact that the kernel matrix $\mathbf{K}$ is used to reflect the interaction between each pair of local descriptors of an image, we may use the original descriptors to calculate the matrix $\mathbf{K}$.

However, as evaluated in [25] and illustrated in Fig. 3, the inner product between two unrelated original descriptors (RootSIFT [1]) is also much larger than zero. Therefore, the inner product of the original RootSIFT cannot construct the kernel directly. If we can modify the original descriptors to approximate the key property of the T-embedding vector without increasing the dimensionality, we can obtain the kernel matrix $\mathbf{K}$ efficiently before the aggregation step. Fortunately, the experimental result (see Fig. 3 and details in the following) shows that the excellent similarity metric of T-embedding is mainly given by the whitening operation [19], [30] [see (1) and (3)]. Therefore, we propose to map the original RootSIFT $\mathbf{x}_i$ to whitened RootSIFT $\mathbf{d}_i$ by a whitening operation to approximate this similarity metric

$$\mathbf{d}_i = \mathrm{diag}({\zeta_1}^{-1/2}, \ldots, {\zeta_d}^{-1/2})\mathbf{Q}^\top(\mathbf{x}_i - \mathbf{z}_0) \tag{13}$$

where $\mathbf{z}_0 = \mathbb{E}_Z[\mathcal{Z}]$, and $\mathbf{Q}$ are the eigenvectors of the covariance matrix, which is computed by the centered training set $\mathcal{Z}$. ζ_i is the eigenvalue of the ith eigenvector, and $\mathbf{d}_i$ is l_2-normalized. In the following, we prove that similarity metric between whitened RootSIFT approximately satisfies the property of T-embedding vector.

To evaluate the similarity metric between whitened RootSIFTs and show the influence of whitening operation for T-embedding vector, the datasets *Liberty harris* and *Notredame*

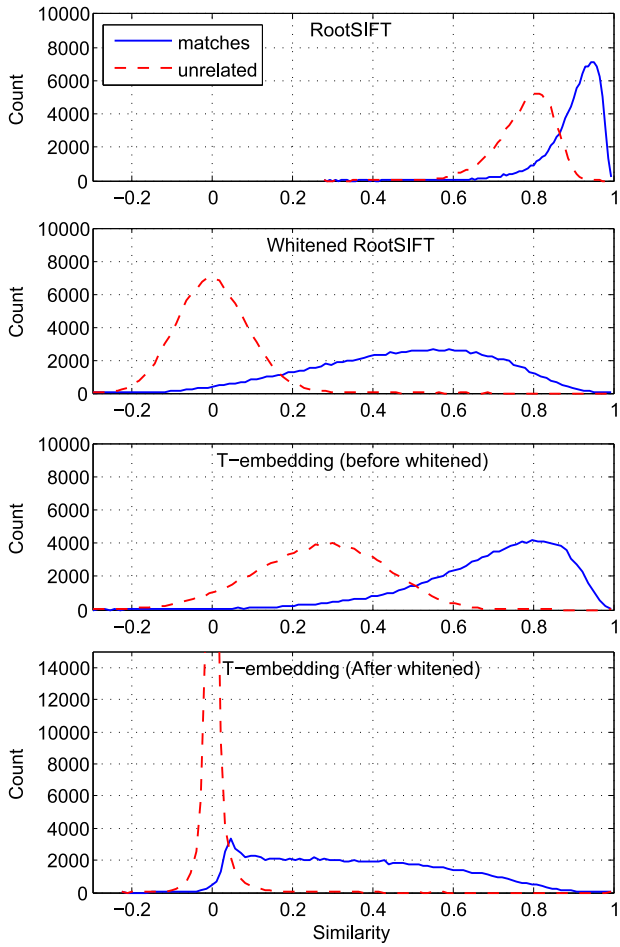


Fig. 3. Histogram of the cosine similarity between related and unrelated image patches described by RootSIFT (1st row), whitened RootSIFT (2nd row), T-embedding vector before whitened (3rd row), and T-embedding vector after whitened (4th row). $|\mathcal{C}| = 16$ in T-embedding step.

harris[2] [46] are employed to compare the statistics of the cosine similarity for related and unrelated patches. We use about 110k pairs of matched image patches and the same number of pairs of unrelated patches from *Notredame harris* to compute the cosine similarity between different descriptors [RootSIFT (1st row), Whitened RootSIFT (2nd row), T-embedding vector before whitening (3rd row), T-embedding vector after whitening (4th row)]. The training stage of T-embedding and whitening operation are performed with *Liberty harris*. We illustrate the different distribution of cosine similarity between different descriptors in Fig. 3. It shows that the whitening operation increases the contrast between related and unrelated descriptors. Moreover, the whitened RootSIFT descriptors approximately satisfy the property of T-embedding vector mentioned above, i.e., the cosine similarity between two unrelated descriptors is close to zero yet much larger than zero for related descriptors.

[2][Online]. Available: http://www.cs.ubc.ca/~mbrown/patchdata/patchdata.html

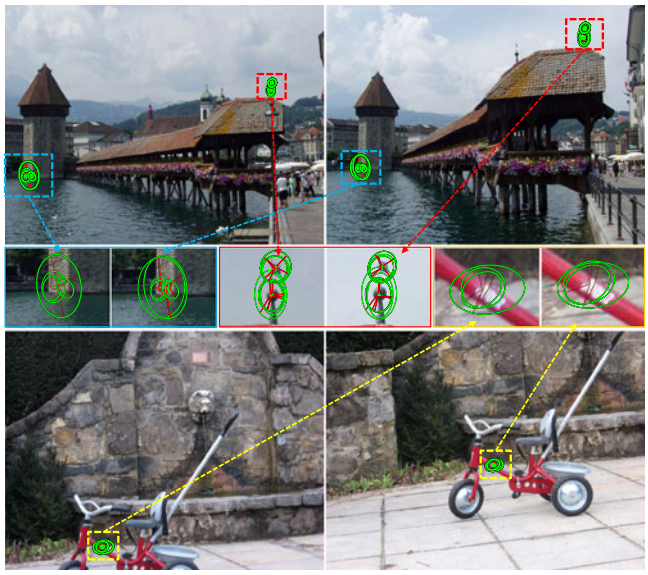


Fig. 4. Examples of the co-occurrences caused by the feature detector. Local features are extracted with the Hessian-affine detector. Green ellipses denote the scale of local features. Red lines denote the main orientation of local features. Different orientations (red lines) also produce different descriptors. However, these descriptors sharing similar spatial location are extracted at the same time.

The kernel matrix with whitened RootSIFT can be constructed as follows:

$$\mathbf{K}_{\mathrm{SIFT}} = \begin{bmatrix} \mathrm{sim}\left(\mathbf{d}_1, \mathbf{d}_1\right) & \dots & \mathrm{sim}\left(\mathbf{d}_1, \mathbf{d}_n\right) \\ \vdots & \ddots & \vdots \\ \mathrm{sim}\left(\mathbf{d}_n, \mathbf{d}_1\right) & \dots & \mathrm{sim}\left(\mathbf{d}_n, \mathbf{d}_n\right) \end{bmatrix} \tag{14}$$

where

$$\mathrm{sim}\left(\mathbf{d}_i, \mathbf{d}_j\right) = \begin{cases} \langle \mathbf{d}_i | \mathbf{d}_j \rangle & \text{if } \langle \mathbf{d}_i | \mathbf{d}_j \rangle > t \\ 0 & \text{if } \langle \mathbf{d}_i | \mathbf{d}_j \rangle \leq t. \end{cases} \tag{15}$$

With the benefit from the low dimensionality of the whitened RootSIFT, we can calculate the kernel matrix more efficiently then (5). Note most of the cosine similarity $\langle \mathbf{d}_i | \mathbf{d}_j \rangle$ between unrelated descriptors, see Fig. 3, is less than 0.2. Therefore, we employ $t = 0.2$.

2) Embedding Weak Spatial Context: We consider the spatial context between local features in an image to further depress the co-occurrence, especially the artificial visual co-occurrences [19]. Either the T-embedding vectors or the whitened RootSIFT descriptors can only measure the similarity between descriptors of local features, whereas the co-occurrences [7], [19] of different local features in an image is also an important factor to measure the interaction between local features. Besides some visual patterns which produce co-occurrences between different images, the feature detector may also introduce some artificial co-occurrences [19]. Fig. 4 illustrates some co-occurrences examples caused by the feature detector. [25] employs RN (rotation and normalization) to address the co-occurrence issue. Since RN operation is conducted on the final aggregated vector, it aims to depress the co-occurrences between the bins of the final vector. As a supplement of RN, we exploit the spatial context of local

features to capture the artificial co-occurrences in aggregation step.

We define a dependence matrix based on the spatial context among local features as follows:

$$\mathbf{K}_{\mathrm{SP}} = \begin{bmatrix} \mathrm{dep}(\mathbf{l}_1, \mathbf{l}_1) & \dots & \mathrm{dep}(\mathbf{l}_1, \mathbf{l}_n) \\ \vdots & \ddots & \vdots \\ \mathrm{dep}(\mathbf{l}_n, \mathbf{l}_1) & \dots & \mathrm{dep}(\mathbf{l}_n, \mathbf{l}_n) \end{bmatrix} \tag{16}$$

where

$$\mathrm{dep}(\mathbf{l}_i, \mathbf{l}_j) = \begin{cases} 1 - \frac{1}{\beta}\sqrt{||\mathbf{l}_i - \mathbf{l}_j||}, & \text{if } \sqrt{||\mathbf{l}_i - \mathbf{l}_j||} < \beta \\ 0, & \text{if } \sqrt{||\mathbf{l}_i - \mathbf{l}_j||} \geq \beta \end{cases} \tag{17}$$

and $\mathbf{l}_i$ denotes the location of each local feature on the image plane. In our experiments, we set $\beta = 5$ to depress the influence of local descriptors pairs with large spatial intervals.

C. Calculation of the Weighting Coefficients

When the kernel matrix $\mathbf{K}_{\mathrm{new}}$ is ready, we derive the weighting coefficients from it. In the following, we discuss two solutions to achieve this goal.

1) Fast Democratic Aggregation: Before discussing our DDA method, with the benefit from our new kernel matrix, we first demonstrate a fast version of DA method [15].

After constructing the kernel matrix, similar with the original DA, the weighting coefficients $\{\lambda_{\mathbf{x}_1}, \dots, \lambda_{\mathbf{x}_n}\}$ can be determined by our new kernel matrix with

$$\mathbf{\Lambda}\mathbf{K}_{\mathrm{new}}\mathbf{\Lambda}\mathbf{1}_n = \mathbf{1}_n \tag{18}$$

where $\mathbf{\Lambda} = \mathrm{diag}(\lambda_{\mathbf{x}_1}, \dots, \lambda_{\mathbf{x}_n})$, and (18) can be solved by the Sinkhorn scaling algorithm [26]. However, since the new kernel matrix is calculated with local descriptors, there is no coupling between T-embedding step and the calculation of the matrix $\mathbf{K}_{\mathrm{new}}$. The weighting coefficients for DA can be computed once the local features are extracted before the T-embedding step. Therefore, the DA can be efficiently achieved by

$$\psi_{\mathrm{Fd}}(\mathcal{X}) = \Sigma^{-1/2} \sum_{\mathbf{x}_i \in \mathcal{X}} \lambda_{\mathbf{x}_i} \left(\mathrm{R}(\mathbf{x}_i) - \mathrm{R}_0\right). \tag{19}$$

Considering its high efficiency, we refer to the DA with our new kernel matrix as *FDA*.

2) Democratic Diffusion Aggregation: The weighting coefficients are derived from the kernel matrix, which usually involves an iterative optimization problem [25], [34]. To further accelerate the aggregation step, we propose a simple closed-form solution without any iterative optimization by conducting a graph diffusion process on the modified kernel matrix.

It is well known that a graph diffusion process is able to reveal the intrinsic relation between local features [10], [47]. Considering the feature space of the local descriptors, the diffusion process can reduce the noise of the feature space and reveal relevant geometric structures of the space at different scales (as shown in the following, the iteration times t plays the role of a scale parameter) [10]. Considering the characteristics of the graph diffusion process, to estimate a better set of weighting coefficients, we conduct a diffusion process on the kernel matrix before calculating the weighting coefficients. It is notable that this scheme can also avoid the iterative operation when calculating the weighting coefficients.

Let $G = (\mathcal{X}, \mathbf{K})$ represent a set of local descriptors from an image as an edge-weighted graph, where $\mathcal{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ is the set of local descriptors and $\mathbf{K}$ represent pairwise similarities between local descriptors, i.e., the kernel matrix. A simple realization of diffusion process on a graph is to compute powers of the graph matrix $\mathbf{K}$, i.e., to run the random walk forward in time. In order to make the graph diffusion process independent of the iteration times T (or time), we consider the graph diffusion process [27] defined as

$$\mathbf{K}^{(T)} = \sum_{i=0}^{T} \left(\eta \frac{\mathbf{K}}{||\mathbf{K}||_1}\right)^i. \tag{20}$$

When $0 < \eta < 1$, (20) converges to a fixed and nontrivial solution given by

$$\mathbf{K}^{\infty} = \lim_{T \to \infty} \mathbf{K}^{(T)} = \left(\mathbf{I} - \eta \frac{\mathbf{K}}{||\mathbf{K}||_1}\right)^{-1} \tag{21}$$

where $\mathbf{I}$ is the identity matrix.

In aggregation step, [25] and [34] compute the kernel matrix by $\mathbf{K} = \mathbf{\Phi}^{\top}\mathbf{\Phi}$, where $\mathbf{\Phi} = [\phi(\mathbf{x}_1), .., \phi(\mathbf{x}_n)]$ denotes the set of embedded vectors. Our FDA method uses (12) to compute the kernel matrix, i.e., $\mathbf{K} = \mathbf{K}_{\mathrm{new}}$. All of them estimate the weighting coefficients with kernel matrix which involves an iterative optimization. Considering the characteristics of the graph diffusion process, different from previous work, we first propose to estimate the weighting coefficients with $\mathbf{K}^{\infty}$ computed by (21). To equalize the contribution of each local descriptor, inspired by GMP [34], our DDA's criterion can be written as

$$\mathbf{K}^{\infty}\boldsymbol{\lambda}^{\top} = \mathbf{1}_n^{\top} \tag{22}$$

where $\boldsymbol{\lambda} = [\lambda_{\mathbf{x}_1}, \dots, \lambda_{\mathbf{x}_n}]$ denotes the weighting coefficients vector and $\mathbf{1}_n = \underbrace{[1, \dots, 1]}_{n}$. $\boldsymbol{\lambda}$ is used to modify $\mathbf{K}^{\infty}$ to a democratic kernel matrix. The weighting coefficients can be computed by

$$\boldsymbol{\lambda}^{\top} = (\mathbf{K}^{\infty})^{-1}\mathbf{1}_n^{\top} = \left(\mathbf{I} - \eta \frac{\mathbf{K}}{||\mathbf{K}||_1}\right)\mathbf{1}_n^{\top}. \tag{23}$$

(23) shows that, the weighting coefficients can be directly calculated by the kernel matrix. Compared with FDA, DDA method introduces an extra parameter η. In addition, when $\eta \to 0$, the DDA degenerates to sum-aggregation. Fig. 5 illustrates the difference between our aggregation methods (include DDA and FDA) and original DA. Table I shows the computational complexity comparing original and our FDA. In addition, we also visualize the different influence of DA, FDA and DDA in the aggregation step in Fig. 6. It shows that similar key descriptors can be detected around the sink by the three aggregation methods. In addition, some repeated structures on the wall can be suppressed in FDA and DDA with the benefit from weak context.

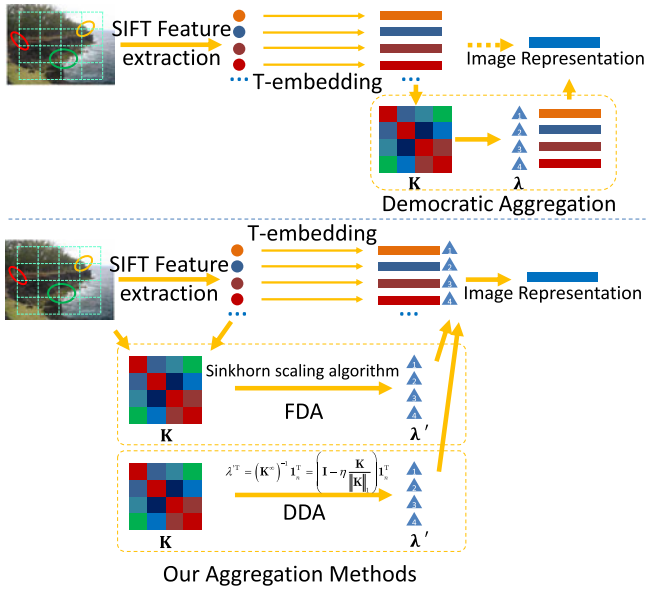


Fig. 5. Framework of our aggregation methods and DA. Due to the independence between embedding step and weighting coefficients computation, our methods can achieve higher computational efficiency. FDA: fast democratic aggregation, DDA: democratic diffusion aggregation.

TABLE I
COMPLEXITY COMPARISON WITH MULTIPLICATIONS TIMES

method	Calculation of $\psi(\mathcal{X})$	Calculation of $\mathbf{K}/\mathbf{K}_{\mathrm{new}}$
DA	$O(n\mathrm{D}^2)$	$O(n^2\mathrm{D})$
FDA/DDA	$O(\mathrm{D}^2)$	$O(n^2 d)$

DA – Original democratic aggregation, FDA – Fast democratic aggregation, DDA – Democratic Diffusion aggregation. $card(\mathcal{X}) = n, \mathrm{D} = (|\mathcal{C}| - 1) \times d$.

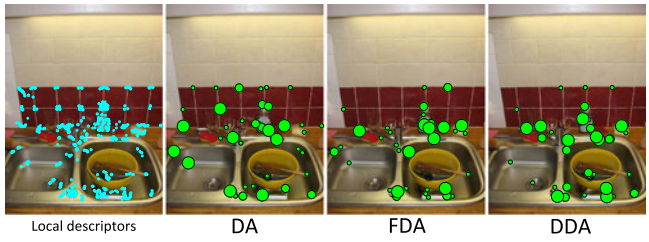


Fig. 6. Influence of weighting coefficients calculated by different aggregation methods. The left image shows the location of all the extracted local descriptors. After calculating the weighting coefficients with DA, FDA, and DDA, 50 local descriptors with highest weights are illustrated on the right three images, respectively. A larger circle denotes higher weights. Similar key descriptors are detected around the sink by the three aggregation methods. In addition, with the benefit from embedding weak context in FDA and DDA, some repeated structures on the wall are further suppressed. DA: original DA, FDA: fast democratic aggregation, DDA: democratic diffusion aggregation.

IV. QUERY FUSION

To further enhance the retrieval performance of our compact image representation, at the re-ranking step, we discuss a new query retrieval strategy – *query fusion* in this section. As we know, QE [8] can improve the retrieval accuracy of the BOW based image search techniques [7], [22], [36], [39]. Since QE is very sensitive to false positive feature matches, spatial verification, which works like a feature filter, is adopted before expanding the local feature to the new query. The baseline of QE [8] shows even worse retrieval result which simply sums the BOW vectors computed from the resulting image and the query image without spatial verification. In a word, spatial verification is vital for QE.

For the compact image representation, since no spatial information of local features is preserved after embedding and aggregation steps, the QE is not compatible with the compact image representation. However, as discussed in [25], the T-embedding method demonstrates an important metric property that the cosine similarity between two unrelated T-embedding vectors is close to zero yet much larger than zero for related vectors. That is to say, the similarity metric of T-embedding vectors is much robust for false positives. This key property shows similar influence as spatial verification. Thus, the query vector and the resulting vectors can be simply averaged to obtain the new refined query, i.e., *query fusion*. An outline of our strategy is as follows:

1) search the dataset vectors with the query vector, and select the top N ranked vectors as fusion vector;

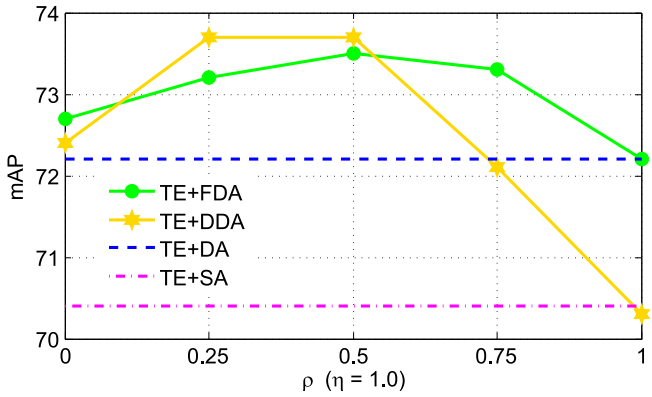


Fig. 7. Influence of ρ on the Holidays dataset. We set $\eta = 1$ for DDA. TE: T-embedding, FDA: fast democratic aggregation, DDA: democratic diffusion aggregation, DA: original democratic aggregation, SA: sum-aggregation.

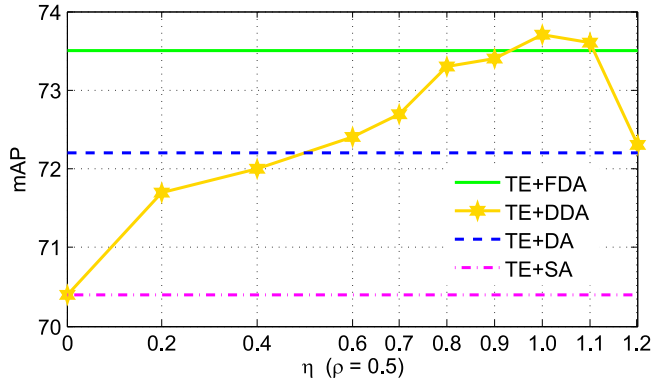


Fig. 8. Influence of η on the Holidays dataset. We set mixing coefficient $\rho = 0.5$ for FDA and DDA. TE: T-embedding, SA: sum-aggregation, DA: original democratic aggregation, FDA: fast democratic aggregation, DDA: democratic diffusion aggregation.

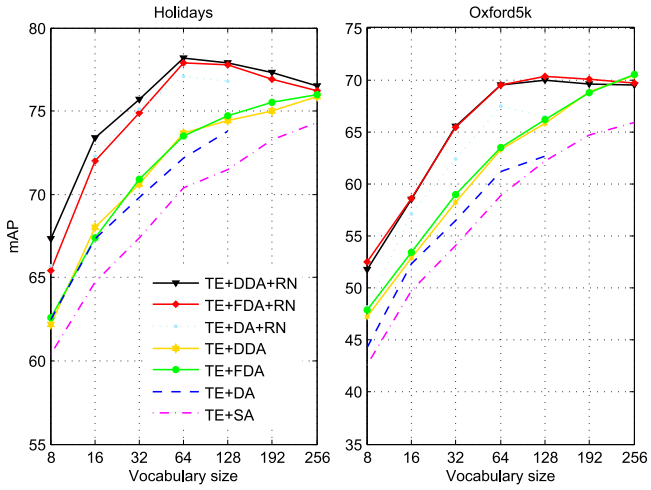


Fig. 9. Retrieval performance with different vocabulary size on two datasets. TE: T-embedding, SA: sum-aggregation, DA: original DA, FDA: fast democratic aggregation, DDA: democratic diffusion aggregation, RN: rotation and normalization.

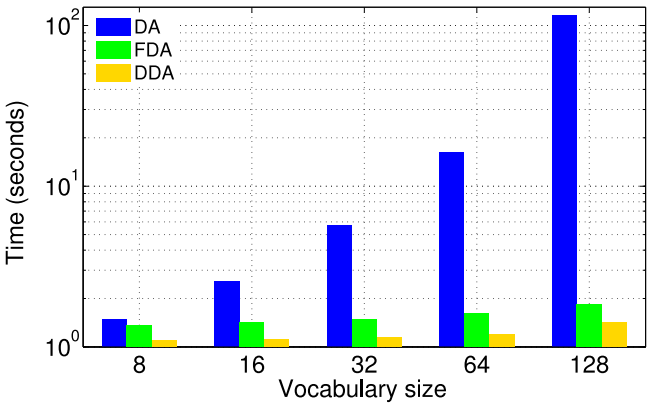


Fig. 10. Average CPU timings for FDA, DDA, and the original DA on Holidays. The vertical axis is shown in log-scale.

TABLE II
PERFORMANCE COMPARISON BETWEEN ORIGINAL AGGREGATION (DA) AND OUR PROPOSED FDA AND DDA

Method	Dim.	mAP Holidays	mAP Oxford5K	mAP Ox105K
TE+SA		70.4	58.9	52.3
TE+DA		72.2	62.0	54.1
TE+FDA		73.5	63.5	60.7
TE+DDA	8064	73.7	63.3	60.9
TE+DA+RN		77.1	67.6	61.1
TE+FDA+RN		78.0	69.5	62.5
TE+DDA+RN		**78.2**	**69.5**	**62.6**
TE+DA+RN	↓	72.0	56.2	49.2
TE+FDA+RN	1024	72.3	**58.3**	50.1
TE+DDA+RN		**73.0**	58.2	**50.5**
TE+DA+RN	↓	70.0	52.8	45.1
TE+FDA+RN	512	70.2	53.4	45.4
TE+DDA+RN		**70.4**	**53.4**	**45.7**

We also present the results after dimensionality reduction to short vectors. TE – T-embedding, SA – Sum-aggregation, RN – Rotation and normalization. $|\mathcal{C}| = 64$.

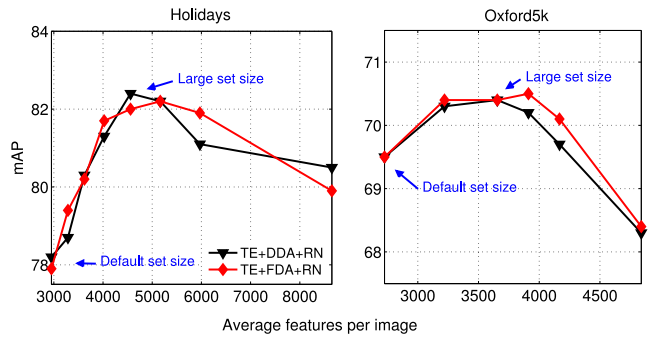


Fig. 11. Influence of different feature set size based on our image representations. TE: T-embedding, FDA: fast democratic aggregation, DDA: democratic diffusion aggregation, RN: rotation and normalization.

2) average current query vector with N fusion vectors to form a new query;
3) re-query the dataset with the new query, and get new retrieval results; and
4) repeat step 1) - 3) M times.

The influence of N and M will be discussed in experimental section. Experimental results show that the *query fusion* exhibits a significant improvement in retrieval accuracy.

V. EXPERIMENTS

In this section, we present the experimental validation about our aggregation methods and query fusion. We employ T-embedding method in the embedding step. The original DA method [25] is adopted as baseline to evaluate the FDA and DDA.

A. Datasets and Evaluation Protocol

We evaluate our methods on two public datasets, INRIA Holidays [20] and Oxford5K [38]. The performance on Oxford105K is also reported which combines Oxford5K with 100K distractor images. The Holidays+Flickr1M [20] dataset is used to evaluate the retrieval quality on a larger scale.

INRIA Holidays dataset contains 1491 images which consist of different locations and objects. The dataset selects 500 images as queries associated with the 500 partitioning groups of the image set. The retrieval performance is measured by mean average precision (mAP) which is computed from the ranked list with the query removed. All training stages are performed on a independent dataset, i.e., Flickr60K [20].

Oxford5K dataset consists of 5062 images of Oxford famous buildings. 55 query images are selected in the dataset associated with 11 distinct buildings. For each query, a bounding box is provided to denote the area of the query image depicting a landmark building. We only employ the local descriptors inside the bounding box to construct the compact image representation [25]. The retrieval performance is also measured by mAP, which is computed from the ranked list with the *junk* images removed. For the experiments on Oxford5K and Oxford105K,

TABLE III
PERFORMANCE COMPARISON BETWEEN DEFAULT FEATURE SETS AND THE LARGE FEATURE SETS ON HOLIDAYS AND OXFORD5K

	Default		Large	
Method(dim=8064)	Holidays	Oxford5K	Holidays	Oxford5K
TE+DA	72.2	62.0	73.8	62.7
TE+DA+RN	77.1	67.6	81.6	69.5
TE+DA+RN+QF	79.2	74.8	83.0	73.2
TE+FDA	73.5	63.5	74.3	64.6
TE+FDA+RN	78.0	69.5	82.0	70.4
TE+FDA+RN+QF	79.5	75.1	83.3	74.9
TE+DDA	73.7	63.3	74.9	64.4
TE+DDA+RN	78.2	69.5	82.4	70.4
TE+DDA+RN+QF	**80.5**	**75.3**	**84.4**	**75.7**

TE – T-embedding, DA – Original democratic aggregation, A – Fast democratic aggregation, DDA – Democratic diffusion aggregation, RN – Rotation and normalization, QF – Query fusion. $|\mathcal{C}| = 64$.

all the training stages are performed on the independent dataset Paris6K [39].

B. Implementation Details

Local descriptors. Hessian-Affine local feature detector [33] is employed to extract the regions of interest. The extracted local features are described by RootSIFT descriptors [1], [32], i.e., the same descriptors provided in previous work [2], [25]. Most of our experiments use the default detector threshold value of Hessian-Affine detector [33]. We also consider the use of different threshold values to include larger sets of features, and show the corresponding benefit in search quality.

Post-processing. The same post-processing method is used as [25]. After aggregation step, the final vector is power-law normalized [24], [37]. This process improves the performance by depressing the visual bursts [21]. We set the power-law exponent α to 0.5 standardly to achieve the best or close-to-best performance. RN (Rotation and Normalization) operation is employed to address the co-occurrences issue between the bins of the final vector. It is conducted with a second power-law normalization ($\alpha = 0.5$) after rotating the aggregated vectors with a PCA rotation matrix [40].

Dimension reduction. We preserve only the first D' components of the aggregated vectors after post-processing to evaluate the performance of short vector representation.

Search scheme. In the search step, we evaluate our image representation by linearly scanning the aggregated vectors of all database images.

Query fusion. Since the query fusion is compatible with power-law normalization and the RN operation, it is performed after all these post-processing methods.

C. Aggregation Methods Evaluation

1) Impact of the Parameters: Compared with original aggregation method, FDA introduces three extra parameters, i.e., mixing coefficient ρ, threshold t and β. Moreover, DDA method also introduces an extra parameter η. We empirically set $\beta = 5$

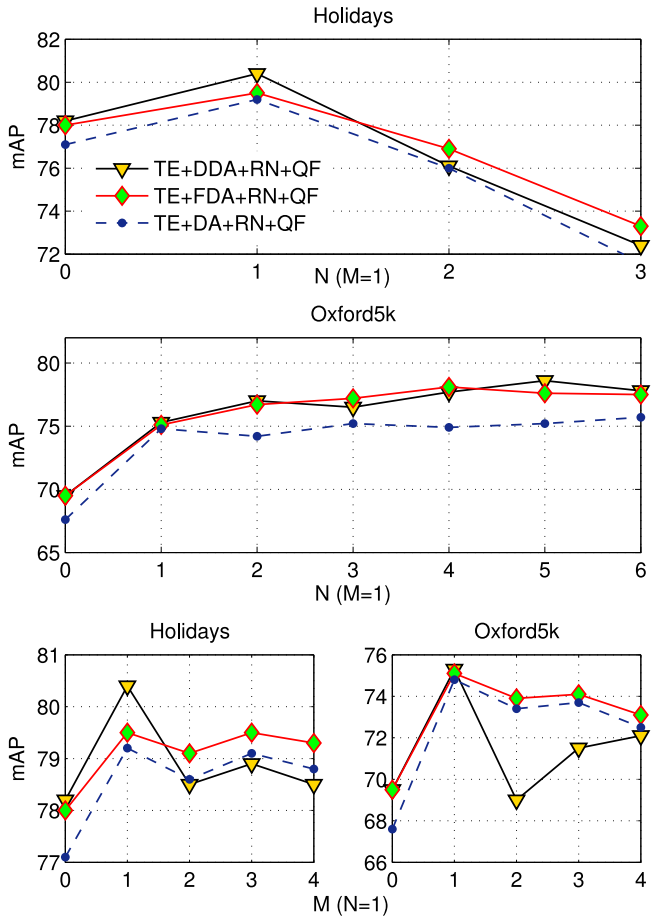


Fig. 12. Performance of query fusion with different M and N on default feature set. Note, we fix $M = 1$(or $N = 1$) to evaluate the influence of N(or M). $|\mathcal{C}| = 64$. TE: T-embedding, DA: original democratic aggregation, FDA: fast democratic aggregation, DDA: democratic diffusion aggregation, RN: rotation and normalization, QF: query fusion.

to depress the influence of local descriptors pairs with large spatial intervals, and $t = 0.2$ to obtain a sparse kernel matrix.

Influence of mixing coefficient. If $\rho = 0$, the spatial context is ignored in aggregation step. If $\rho = 1$, only spatial context is taken into account without feature descriptors. We select different ρ in democratic aggregating step with $|\mathcal{C}| = 64$ on Holidays dataset to evaluate the influence of ρ. The experimental results are shown in Fig. 7.

As shown in Fig. 7, both the whitened RootSIFT descriptors and spatial context can improve the performance compared with sum-aggregation (TE+SA). For DDA, whitened RootSIFT descriptors may have dominant effect in aggregation step. However, combining two of them achieves better performance. Therefore, we set $\rho = 0.5$ in the following experiments.

Influence of η. As mentioned in Section III, when $\eta \to 0$, the DDA smoothly degenerates to sum-aggregation. That is to say, the democracy of aggregation method can be adjusted by η. Fig. 8 shows that DDA exhibits comparable or better retrieval performance compared with FDA when $\eta > 0.8$. Since the best retrieval performance is obtained around $\eta = 1$, we simply set $\eta = 1$ in the following experiments. In addition, when combined

TABLE IV
PERFORMANCE COMPARISON BETWEEN EXHAUSTED SEARCH AND EXHAUSTED SEARCH COMBINED WITH QUERY FUSION (QF)

		Default feature set			Large feature set		
Method	Dim.	Holidays	Oxford5K	Ox105K	Holidays	Oxford5K	Ox105K
TE+DA+RN		77.1	67.6	61.1	81.6	69.5	65.7
TE+DA+RN+QF		79.2	74.8	69.0	83.0	73.2	71.5
TE+FDA+RN	8064	78.0	69.5	62.5	82.0	70.4	65.5
TE+FDA+RN+QF		79.5	75.1	**71.8**	83.3	74.8	72.9
TE+DDA+RN		78.2	69.5	62.6	82.4	70.4	65.9
TE+DDA+RN+QF		**80.5**	**75.3**	71.2	**84.4**	**75.7**	**73.8**
TE+FDA+RN		72.3	58.3	50.1	76.5	60.5	54.2
TE+FDA+RN+QF	↓	73.5	67.3	58.5	77.6	67.0	**63.4**
TE+DDA+RN	1024	73.0	58.2	50.5	77.1	61.2	54.7
TE+DDA+RN+QF		**74.0**	**67.4**	**62.0**	**78.4**	**67.6**	62.7
TE+FDA+RN		70.2	53.4	45.4	73.9	55.7	49.4
TE+FDA+RN+QF	↓	**71.4**	61.0	**55.2**	74.8	61.3	57.6
TE+DDA+RN	512	70.4	53.4	45.7	74.8	56.1	49.9
TE+DDA+RN+QF		71.2	**62.1**	54.7	**75.7**	**61.5**	**57.9**

We also present the results after dimensionality reduction to short vectors. TE – T-embedding, DA – Original democratic aggregation, FDA – Fast democratic aggregation, DDA – Democratic diffusion aggregation, RN – Rotation and normalization. $|\mathcal{C}| = 64$.

with RN operation (see Fig. 9), the DDA achieves better search quality.

Vocabulary size $|\mathcal{C}|$. For different vocabulary sizes, Fig. 9 shows that our two aggregation methods both obtain comparable or even better retrieval accuracy. Due to the computational overhead, the original DA method is only evaluated under $|\mathcal{C}| \leq 128$. With the benefit from high efficiency, we can evaluate our methods with larger vocabulary size. As shown in Fig. 9, the retrieval accuracy is consistently improved in the aggregation step. However, when $|\mathcal{C}| > 64$, the capability of RN to improve retrieval performance is declining. When $|\mathcal{C}| = 256$, even worse performance is observed in the Oxford5k dataset. Therefore, it is not suitable to apply larger vocabulary, i.e., $|\mathcal{C}| > 128$, in the RN operation, considering the limited training images and higher dimensionality of image representation.

2) Comparison With DA: Fig. 10 reports the computational time for our aggregation methods and the original DA. The evaluation is carried out on a Xeon E5-2680v2/2.8GHz (10 cores) on the Holidays dataset. Both the original DA and our aggregation methods are implemented with MATLAB code. Note, the CPU time is larger than elapsed time because CPU time accumulates all active threads. Our FDA is an order of magnitude faster than the original DA when $|\mathcal{C}| = 64$. The speedup is due to the fact that the kernel matrix is calculated efficiently and the weighting coefficients for DA can be pre-computed without embedded vectors. Thus, like sum-aggregation, the final compact vector can be computed efficiently with (19). Furthermore, due to its non-iterative solution, DDA is more efficient compared with FDA (about 30% speedup).

As shown in Table II, besides efficient computation, our efficient aggregation methods also improve the retrieval accuracy. The performance after dimensionality reduction is also presented in Table II. It shows that our aggregation methods work well with RN operation and dimensionality reduction.

3) Performance on Large Set of Features: To compare with the baseline [25], the evaluation above is conducted with the same local descriptors detected by the default threshold on the response of local key points. In this section, we evaluate the influence of different sizes of feature set.

Fig. 11 shows that the default setting of feature detector does not produce the best performance of our image representations, yet using a little more features yields superior retrieval performance. Based on the performance of different feature set, we derive two larger feature sets from Holidays (about 4.5k features per image) and Oxdord5k (about 3.7k features per image), respectively. Table III reports the comparison of retrieval performance on default feature sets and the large feature sets.

D. Query Fusion Evaluation

Impact of parameters. We first discuss the influence of iteration number M and fusion vector number N to the retrieval performance. The performance of query fusion with different M and N on the default datasets is presented in Fig. 12. We can see that no further improvement can be obtained for larger M. The number of fusion vectors N shows different effect on Oxford5K and Holidays. This is expected, because Holidays dataset contains only a few images of the same object. Therefore, previous work about QE [7], [8], [22], [36], [39] usually only evaluate the performance of QE only with Oxford5K and Paris6K, which have quite a lot related images for each query image. Our query fusion improves the retrieval accuracy on both Holiday and Oxfor5K with $N = 1$. On Oxford5K, the best result mAP $= 78.6\%$ is obtained with $N = 5$ and $M = 1$ by DDA (TE+DDA+RN+QF). To make a trade off between accuracy and efficiency, $N = 1$ and $M = 1$ are our recommend parameters for query fusion. Thus, we need search the dataset twice.

Performance. Table IV presents the retrieval accuracy improvement obtained by query fusion. The short vector after dimensionality reduction is also compatible with query fusion. Comparing the results on two different datasets, we must admit that our query fusion strategy is more suitable for Oxford5K-like

Fig. 13. Example results on Oxford5K. The first row of each query shows retrieval results without query fusion, and the second row shows results with query fusion. False positives are marked with red bounding boxes.

dataset, i.e., the dataset that contains plenty related images for a query. Fig. 13 shows some examples of image retrieval results on Oxford5K employed the query fusion strategy. We can see that the query fusion strategy can suppress the false positives and improve the retrieval accuracy by merging the representation of the first returned image from the initial rank list.

E. Performance on Large Scale Dataset

To evaluate the performance of our image representations on large scale dataset, we merge the Holidays dataset with another set of 1 million images retrieved from Flickr [20]. Fig. 14 gives the mAP performance as the function of dataset size for our methods. Due to the high computational complexity of DA method (when $|\mathcal{C}| = 64$), it [25] only presented the results of DA method when $|\mathcal{C}| = 16$, i.e., mAP = 51.9 (dim = 1920) and 49.4 when vector dimensionality reduces to 1024. With the benefit from efficient computation, we present the performance of our aggregation methods with $|\mathcal{C}| = 64$ on the large scale dataset. Our aggregated vectors achieve significant improvement compared with pervious work (mAP = 56.7 for DDA and 57.0 for FDA when dim = 1024 and $|\mathcal{C}| = 64$).

F. Comparison With Other Methods

Table V shows that our methods outperform most of existing encoding methods. Although the orientation covariant aggregation [43] achieves similar mAP performance on Holidays dataset with ours, the orientation covariant aggregation [43] leads to higher dimensionality of vector representation and sensitivity to the orientation of image. In fact, our efficient aggregation methods are compatible with orientation covariant aggregation and can be enhanced with it if necessary. Note, with the benefit from query fusion, our method outperforms all of previous methods on Oxford5K and Oxford105K with a large margin.

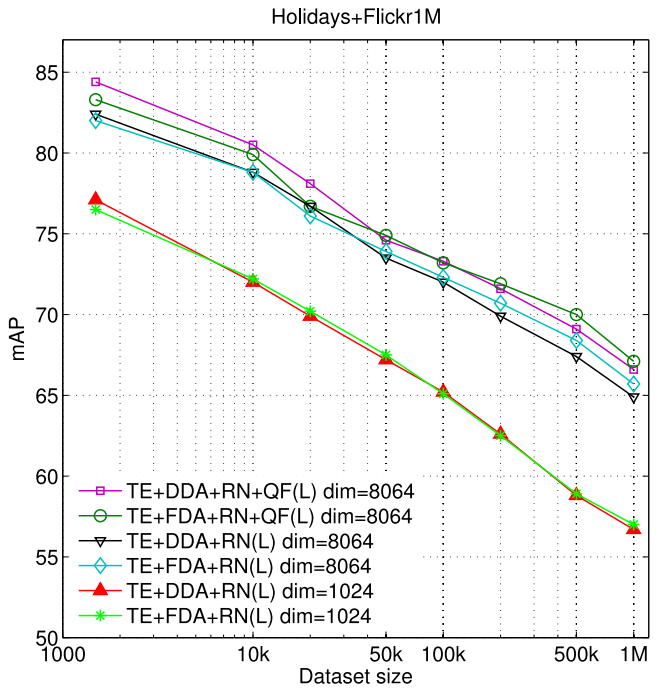


Fig. 14. Performance of our aggregation methods with large feature set on large scale dataset (up to 1 million). $|\mathcal{C}| = 64$. TE: T-embedding, FDA: fast democratic aggregation, DDA: democratic diffusion aggregation, RN: rotation and normalization, QF: query fusion.

TABLE V
PERFORMANCE COMPARISON WITH OTHER ENCODING METHODS

Method	Dim.	mAP		
		Holidays	Oxford5K	Ox105K
VLAD [24]	4096	55.6	37.8	–
Fisher [24]	4096	59.5	41.8	–
VLAD-intra [2]	32 536	65.3	55.8	–
VLAD+ [11]	8192	65.8	51.7	45.6
CVLAD [48]	32 768	68.8	42.7	–
FV+Attr. [12]	6755	69.9	–	–
SC [17]	11 024	76.7	–	–
TE+DA+RN [25]	8064	77.1	**67.6**	**61.1**
MOP-CNN [18]	12 288	78.8	–	–
MOP-CNN+PCA [18]	2048	80.2	–	–
CVLAD(ODense) [48]	32 768	80.4	45.9	–
Fisher⊗ ($\|\mathcal{C}\| = 32$) [43]	17 920	81.2	60.7	52.2
Fisher⊗ ($\|\mathcal{C}\| = 64$) [43]	35 840	**84.1**	64.8	–
TE+FDA+RN	8064	78.0	69.5	62.5
TE+FDA+RN+QF		79.5	75.1	**71.8**
TE+DDA+RN		78.2	69.5	62.6
TE+DDA+RN+QF		**80.5**	**75.3**	71.2
TE+FDA+RN(L)	8064	82.0	70.4	65.5
TE+FDA+RN+QF(L)		83.3	74.8	72.9
TE+DDA+RN(L)		82.4	70.4	65.9
TE+DDA+RN+QF(L)		**84.4**	**75.7**	**73.8**
TE+FDA+RN+QF(L)	↓	77.6	67.0	63.4
TE+DDA+RN+QF(L)	1024	78.4	67.6	62.7

Results are compared with full vector representation. (L) denotes the performance on large feature sets. TE – T-embedding, DA – Original democratic aggregation, FDA – Fast democratic aggregation, DDA – Democratic diffusion aggregation , RN – Rotation and normalization, QF – Query fusion.

TABLE VI
COMPUTATION COMPLEXITY COMPARISON WITH OTHER ENCODING METHODS

Method	Dim.	Embedding	Aggregation	Search	Remark
VLAD [24]	4096	$O(ndk)$	$O(1)$	$O(dkL)$	$n \approx 3000, d = 64, k = 64$
Fisher [24]	4096	$O(ndk)$	$O(1)$	$O(dkL)$	$n \approx 3000, d = 64, k = 64$
MOP-CNN [18]	12 288	$O(ndk)$	$O(1)$	$O(DL)$	$n \approx 100, d = 500, k = 100, D = 12288$
Fisher$\otimes$ ($\|\mathcal{C}\| = 64$) [43]	35 840	$O(ndka)$	$O(1)$	$O(rdkL)$	$n \approx 3000, d = 64, k = 64, a = 7, r = 8$
CVLAD(ODense) [48]	32 768	$O(ndkr)$	$O(1)$	$O(r^2dkL)$	$n \approx 10^5, d = 128, k = 32, r = 8$
SC [17]	11 024	$O(nd(K + S))$	$O(1)$	$O(KL)$	$n \approx 3000, d = 104 + 128, K = 10000, S - sparsity$
TE+SA+RN [25]	8064	$O(d^2k^2 + ndk)$	$O(1)$	$O(dkL)$	$n \approx 3000, d = 128, k = 64$
TE+DA+RN [25]	8064	$O(nd^2k^2)$	$O(n^2dk)$	$O(dkL)$	$n \approx 3000, d = 128, k = 64$
TE+DDA+RN	8064	$O(d^2k^2 + ndk)$	$O(n^2d)$	$O(dkL)$	$n \approx 3000, d = 128, k = 64$
TE+DDA+RN+QF	8064	$O(d^2k^2 + ndk)$	$O(n^2d)$	$O(MdkL)$	$n \approx 3000, d = 128, k = 64, M = 2$

n: number of feature per image, d: feature dimension, k: codebook size, L: dataset size,
r: number of orientation [48], a: angle vector size [43], M: iteration times of query fusion
The searching complexity is based on linearly scanning the dataset. TE – T-embedding, SA – Sum-aggregation, DA – Original democratic aggregation, FDA – Fast democratic aggregation, DDA – Democratic diffusion aggregation, RN – Rotation and normalization, QF – Query fusion.

We present the complexity analysis of some above methods in Table VI. The early encoding methods, such as VLAD and Fisher Vector [24], show high computational efficiency but poor retrieval accuracy. Although the encoding (embeding+aggregation) complexity of MOP-CNN [18] is similar with VLAD, it suffers high computational complexity in feature extraction step. MOP-CNN requires about 10^{10} multiplications to extract CNN features from an image. SC [17] is time-consuming (about $\times 30$ slower [16] than VLAD) due to large codebook size ($K = 10\,000$ in SC $vs. k = 64$ in VLAD and T-emdding). For T-embedding (TE) based encoding methods, if combined with sum-aggregation (SA) [25], it achieves high computation efficiency close to VLAD. For DA [25], due to extra whitening operation and construction of kernel matrix, the computation is very expensive (see Fig. 2). However, considering our DDA for T-embedding vectors, Table VI shows that it requires the same computation cost as TE+SA in embedding step, and less computation cost compared with DA in aggregation step.

VI. CONCLUSION

In this paper, we target on compact image representation for large-scale CBIR. Based on T-embedding vectors, firstly, we propose a FDA method embedded with weak spatial context. Compared with the original DA, our approach significantly improves the efficiency with over ten times speedup while achieving comparable or even better retrieval accuracy. Secondly, by conducting a graph diffusion process on the kernel matrix, we also propose a closed-form solution named DDA to estimate the weighting coefficients. Due to its non-iterative solution, DDA is more efficient (about $\times \mathbf{14}$ faster) compared with the original DA. However, the RN operation is unsuitable for large vocabularies ($|\mathcal{C}| > 128$). In addition, constructing the kernel matrix is also time and memory consuming for larger set of local descriptors, e.g., densely extracted local descriptors. Future work will be focused on improving the performance with large vocabularies and apply the DDA on the densely extracted local descriptors.

At the re-ranking step, we discuss a simple yet effective retrieval strategy, query fusion, to boost the retrieval performance of the compact image representation. Experimental results demonstrate consistent improvement in accuracy with this strategy. Our query fusion is verified based on exhausted search which needs to keep all original vectors in memory. For large scale image search equipped with approximate nearest neighbor search algorithms, we will investigate the query fusion in the case that the original feature vectors are discarded to save memory.

REFERENCES

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

[2] R. Arandjelovic and A. Zisserman, "All about VLAD," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 1578–1585.

[3] A. Babenko and V. Lempitsky, "The inverted multi-index," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2012, pp. 3069–3076.

[4] L. Bo and C. Sminchisescu, "Efficient match kernel between sets of features for visual recognition," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, pp. 135–143.

[5] Y.-L. Boureau, J. Ponce, and Y. LeCun, "A theoretical analysis of feature pooling in visual recognition," in *Proc. Int. Conf. Mach. Learn.*, 2010, pp. 111–118.

[6] L. Chu, S. Jiang, S. Wang, Y. Zhang, and Q. Huang, "Robust spatial consistency graph model for partial duplicate image retrieval," *IEEE Trans. Multimedia*, vol. 15, no. 8, pp. 1982–1996, Jun. 2013.

[7] O. Chum and J. Matas, "Unsupervised discovery of co-occurrence in sparse high dimensional data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2010, pp. 3416–3423.

[8] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman, "Total recall: Automatic query expansion with a generative feature model for object retrieval," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2007, pp. 1–8.

[9] R. G. Cinbis, J. Verbeek, and C. Schmid, "Image categorization using fisher kernels of non-IID image models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2012, pp. 2184–2191.

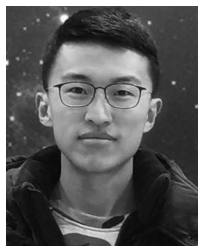
[10] R. R. Coifman and S. Lafon, "Diffusion maps," *Appl. Comput. Harmonic Anal.*, vol. 21, no. 1, pp. 5–30, 2006.

[11] J. Delhumeau, P.-H. Gosselin, H. Jégou, and P. Pérez, "Revisiting the VLAD image representation," in *Proc. ACM Int. Conf. Multimedia*, 2013, pp. 653–656.

[12] M. Douze, A. Ramisa, and C. Schmid, "Combining attributes and Fisher vectors for efficient image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2011, pp. 745–752.

[13] L.-Y. Duan *et al.*, "Overview of the MPEG-CDVS standard," *IEEE Trans. Image Process.*, vol. 25, no. 1, pp. 179–194, Jan. 2016.

[14] L.-Y. Duan, J. Lin, Z. Wang, T. Huang, and W. Gao, "Weighted component hashing of binary aggregated descriptors for fast visual search," *IEEE Trans. Multimedia*, vol. 17, no. 6, pp. 828–842, Jun. 2015.
[15] Z. Gao, J. Xue, W. Zhou, S. Pang, and Q. Tian, "Fast democratic aggregation and query fusion for image search," in *Proc. ACM Int. Conf. Multimedia Retrieval*, 2015, pp. 35–42.
[16] T. Ge, K. He, and J. Sun, "Product sparse coding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2014, pp. 939–946.
[17] T. Ge, Q. Ke, and J. Sun, "Sparse-coded features for image retrieval," in *Proc. Brit. Mach. Vis. Conf.*, Sep. 2013, pp. 132.1–132.11.
[18] Y. Gong, L. Wang, R. Guo, and S. Lazebnik, "Multi-scale orderless pooling of deep convolutional activation features," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 392–407.
[19] H. Jégou and O. Chum, "Negative evidences and co-occurences in image retrieval: The benefit of PCA and whitening," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 774–787.
[20] H. Jégou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 304–317.
[21] H. Jégou, M. Douze, and C. Schmid, "On the burstiness of visual elements," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2009, pp. 1169–1176.
[22] H. Jégou, M. Douze, and C. Schmid, "Improving bag-of-features for large scale image search," *Int. J. Comput. Vis.*, vol. 87, no. 3, pp. 316–336, 2010.
[23] H. Jégou, M. Douze, and C. Schmid, "Product quantization for nearest neighbor search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 1, pp. 117–128, Jan. 2011.
[24] H. Jégou *et al.*, "Aggregating local image descriptors into compact codes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 9, pp. 1704–1716, Sep. 2012.
[25] H. Jégou and A. Zisserman, "Triangulation embedding and democratic aggregation for image search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2014, pp. 3310–3317.
[26] P. A. Knight, "The sinkhorn-knopp algorithm: Convergence and applications," *J. Matrix Anal. Appl.*, vol. 30, no. 1, pp. 261–275, 2008.
[27] S. Lafon and A. B. Lee, "Diffusion maps and coarse-graining: A unified framework for dimensionality reduction, graph partitioning, and data set parameterization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 9, pp. 1393–1403, Sep. 2006.
[28] P. Li, M. Wang, J. Cheng, C. Xu, and H. Lu, "Spectral hashing with semantically consistent graph for image indexing," *IEEE Trans. Multimedia*, vol. 15, no. 1, pp. 141–152, Jan. 2013.
[29] J. Lin *et al.*, "Rate-adaptive compact fisher codes for mobile visual search," *IEEE Signal Process. Lett.*, vol. 21, no. 2, pp. 195–198, Feb. 2014.
[30] W. Liu, J. Wang, S. Kumar, and S.-F. Chang, "Hashing with graphs," in *Proc. Int. Conf. Mach. Learn.*, Jun. 2011, pp. 1–8.
[31] Z. Liu, H. Li, W. Zhou, R. Hong, and Q. Tian, "Uniting keypoints: Local visual information fusion for large-scale image search," *IEEE Trans. Multimedia*, vol. 17, no. 4, pp. 538–548, Apr. 2015.
[32] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.
[33] K. Mikolajczyk and C. Schmid, "Scale and affine invariant interest point detectors," *Int. J. Comput. Vis.*, vol. 60, no. 1, pp. 63–86, 2004.
[34] N. Murray and F. Perronnin, "Generalized max pooling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2014, pp. 2473–2480.
[35] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," *Advances Neural Inf.*, vol. 2, pp. 849–856, 2002.
[36] M. Perďoch, O. Chum, and J. Matas, "Efficient representation of local geometry for large scale object retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2009, pp. 9–16.
[37] F. Perronnin, Y. Liu, J. Sánchez, and H. Poirier, "Large-scale image retrieval with compressed fisher vectors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2010, pp. 3384–3391.
[38] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2007, pp. 1–8.
[39] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Lost in quantization: Improving particular object retrieval in large scale image databases," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2008, pp. 1–8.
[40] B. Safadi and G. Quénot, "Descriptor optimization for multimedia indexing and retrieval," in *Proc. Workshop Content Based Multimedia Indexing*, 2013, pp. 65–71.
[41] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2003, pp. 1470–1477.
[42] X. Tian, Q. Jia, and T. Mei, "Query difficulty estimation for image search with query reconstruction error," *IEEE Trans. Multimedia*, vol. 17, no. 1, pp. 79–91, Jan. 2015.
[43] G. Tolias, T. Furon, and H. Jégou, "Orientation covariant aggregation of local descriptors with embeddings," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 382–397.
[44] G. Tolias and H. Jégou, "Visual query expansion with or without geometry: Refining local descriptors by feature aggregation," *Pattern Recog.*, vol. 47, no. 10, pp. 3466–3476, 2014.
[45] S. Wang and S. Jiang, "INSTRE: A new benchmark for instance-level object retrieval and recognition," *ACM Trans. Multimedia Comput., Commun. Appl.*, vol. 11, no. 3, p. 37, 2015.
[46] S. A. Winder and M. Brown, "Learning local image descriptors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2007, pp. 1–8 .
[47] X. Yang, L. Prasad, and L. J. Latecki, "Affinity learning with diffusion on tensor product graph," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 28–38, Jan. 2013.
[48] W. L. Zhao, H. Jégou, and G. Gravier, "Oriented pooling for dense and non-dense rotation-invariant features," in *Proc. Brit. Mach. Vis. Conf.*, 2013, pp. 99.1–99.11.
[49] L. Zheng, S. Wang, Z. Liu, and Q. Tian, "LP-norm IDF for large scale image search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 1626–1633.
[50] W. Zhou, H. Li, R. Hong, Y. Lu, and Q. Tian, "BSIFT: Towards data-independent codebook for large scale image search," *IEEE Trans. Image Process.*, vol. 24, no. 3, pp. 967–979, Mar. 2015.
[51] W. Zhou, Y. Lu, H. Li, Y. Song, and Q. Tian, "Spatial coding for large scale partial-duplicate web image search," in *Proc. ACM Int. Conf. Multimedia*, 2010, pp. 511–520.
[52] W. Zhou *et al.*, "Towards codebook-free: Scalable cascaded hashing for mobile image search," *IEEE Trans. Multimedia*, vol. 16, no. 3, pp. 601–611, Apr. 2014.
[53] W. Zhou *et al.*, "Scalable feature matching by dual cascaded scalar quantization for image retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 1, pp. 159–171, Jan. 2016.



Zhanning Gao received the B.S. degree in automatic control engineering from the Xi'an Jiaotong University, Xi'an, China, in 2012, and is currently working toward the Ph.D. degree at the Institute of Artificial Intelligence and Robtics, Xi'an Jiaotong University.

His research interests include compact image representation and large scale content-based image retrieval.

Jianru Xue (M'06) received the M.S. and Ph.D. degrees from Xi'an Jiaotong University (XJTU), Xi'an, China, in 1999 and 2003, respectively.

He was with FujiXerox, Tokyo, Japan, from 2002 to 2003, and visited the University of California at Los Angeles, Los Angeles, CA, USA, from 2008 to 2009. He is currently a Professor with the Institute of Artificial Intelligence and Robotics, XJTU. His research interests include computer vision, visual navigation, and video coding based on analysis.

Dr. Xue served as a the Co-Organization Chair of the 2009 Asian Conference on Computer Vision and 2006 Virtual System and Multimedia Conference. He also served as a PC Member of the 2012 Pattern Recognition Conference, and the 2010 and 2012 Asian Conference on Computer Vision.

Wengang Zhou received the B.E. degree in electronic information engineering from Wuhan University, Wuhan, China, in 2006, and the Ph.D. degree in electronic engineering and information science from the University of Science and Technology of China (USTC), Hefei, China, in 2011.

He was a Research Intern with the Internet Media Group, Microsoft Research Asia, Beijing, China, from December 2008 to August 2009. From September 2011 to 2013, he was a Postdoc Researcher with the Computer Science Department, University of Texas at San Antonio, San Antonio, TX, USA. He is currently an Associate Professor with the Department of Electronic Engineering and Information Science, USTC. His research interest includes computer vision and multimedia content analysis and retrieval.

Shanmin Pang received the Ph.D. degree from Xi'an Jiaotong University (XJTU), Xi'an, China, in 2015.

He is currently a Lecturer with the School of Software Engineering, XJTU. His research interests include pattern recognition, computer vision, and image processing.

Mr. Pang was the recipient of the Best Application Paper Award at the ACCV 2012 conference.

Qi Tian (S'95-M'96-SM'03-F'16) received the Ph.D. degree in ECE from the University of Illinois at Urbana-Champaign (UIUC), Champaign, IL, USA, in 2002, the B.E. degree in electronic engineering from Tsinghua University, Beijing, China, in 1992, and the M.S. degree in ECE from Drexel University, Philadelphia, PA, USA, in 1996.

He is currently a Full Professor with the Department of Computer Science, the University of Texas at San Antonio (UTSA), San Antonio, TX, USA. He was a tenured Associate Professor from 2008 to 2012 and a tenure-track Assistant Professor from 2002 to 2008. During 2008 and 2009, he took one-year faculty leave with Microsoft Research Asia as Lead Researcher in the Media Computing Group. He has authored or coauthored more than 320 refereed journal and conference papers. His research interests include multimedia information retrieval, computer vision, pattern recognition, and bioinformatics.

Prof. Tian is the Associate Editor of the IEEE TRANSACTIONS ON MULTIMEDIA, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, and the *Multimedia System Journal*, and is a member of the Editorial Board for the Journal of Multimedia and the *Journal of Machine Vision and Applications*. He is the Guest Editor of IEEE TRANSACTIONS ON MULTIMEDIA and the *Journal of Computer Vision and Image Understanding*, among others. He was the recipient or corecipient of the Best Paper Award in ACM ICMR 2015, the Best Paper Award in PCM 2013, the Best Paper Award in MMM 2013, the Best Paper Award in ACM ICIMCS 2012, the Top 10% Paper Award in MMSP 2011, the Best Student Paper Award in ICASSP 2006, and was the coauthor of the Best Student Paper Candidate in ICME 2015 and the Best Paper Candidate in PCM 2007. He was the recipient of 2014 Research Achievement Awards from the College of Science, UTSA. He was the recipient of the 2010 ACM Service Award.