

# SIFT Meets CNN: A Decade Survey of Instance Retrieval

Liang Zheng, Yi Yang, and Qi Tian, *Fellow, IEEE*

**Abstract**—The Bag-of-Words (BoW) model has been predominantly viewed as the state of the art in Content-Based Image Retrieval (CBIR) systems since 2003. The past 13 years has seen its advance based on the SIFT descriptor due to its advantages in dealing with image transformations. In recent years, image representation based on the Convolutional Neural Network (CNN) has attracted more attention in image retrieval, and demonstrates impressive performance. Given this time of rapid evolution, this article provides a comprehensive survey of image retrieval methods over the past decade. In particular, according to the feature extraction and quantization schemes, we classify current methods into three types, *i.e.*, SIFT-based, one-pass CNN-based, and multi-pass CNN-based. This survey reviews milestones in BoW image retrieval, compares previous works that fall into different BoW steps, and shows that SIFT and CNN share common characteristics that can be incorporated in the BoW model. After presenting and analyzing the retrieval accuracy on several benchmark datasets, we highlight promising directions in image retrieval that demonstrate how the CNN-based BoW model can learn from the SIFT feature.

**Index Terms**—Bag-of-Words model, instance retrieval, Convolutional Neural Networks, SIFT feature, literature survey.

## 1 INTRODUCTION

CONTENT-Based Image Retrieval (CBIR) has been a long-standing research topic in the computer vision society. Featuring an exponentially increasing number of web images, the era of big data calls for scalable systems which allow efficient indexing, retrieval, re-ranking, and browsing. CBIR has come to prominence because natural language is limited in describing the content of images. “A picture is worth a thousand words” depicts the dilemma faced by text-based systems. Currently, CBIR largely focuses on leveraging the visual content in images and designing efficient matching algorithms in terms of both memory and speed.

Text-based image retrieval has been popular since the 1970s. At that time, images had manually annotated texts, and image retrieval was reduced to text retrieval. We refer readers to several survey papers [1], [2] in this field for an overview of the algorithms of the time. There are two drawbacks to the application of text-based algorithms. First, manual annotation is expensive for the database images. Second, text is not fully descriptive of the image content, and even with web annotations, the text may be very noisy and compromise retrieval quality.

The study of Content-Based Image Retrieval truly started in the early 1990s, due to the emergence of large-scale image collections for which manual annotation is infeasible. To overcome this difficulty, visual content of images is used instead of textual annotations. Images are indexed by their visual features, such as texture and color. Since then, a myriad of algorithms and image retrieval systems have been proposed. A straightforward strategy in CBIR is to employ

global descriptors to retrieve images that are similar to the query. This method dominated the field of image retrieval in the 1990s and early 2000s. A well-known problem of many such methods is that global signatures may fail the invariance expectation to image changes such as illumination, occlusion, intersection, and truncation. Image variance limits the application scope of image retrieval and lowers retrieval accuracy. This problem has given rise to image retrieval methods based on local features.

This survey focuses on instance-level image retrieval. In this task, given a query image depicting a specific object/scene/architecture, we aim to search for images containing the same object/scene/architecture that may be captured under different views, illumination, or with occlusions. Here we contrast instance retrieval with class retrieval [3], [4]. The latter aims at retrieving images of the same class with the query. In the following, if not specified, we use “image retrieval” and “instance retrieval” interchangeably.

In 2000, Smeulders *et al.* [5] presented a comprehensive survey of CBIR “at the end of the early years”; three years later (2003) the BoW model was adopted in image retrieval [6], and in 2005 was applied to image classification [7]. The community witnesses the rise of the BoW model for about 10 years until it reached a peak in 2012 when Krizhevsky *et al.* [8] achieved the state-of-the-art result in ILSRVC 2012 which improved over previous best results by a large margin. Since then, the research focus has transferred to deep learning methods [9], especially to the Convolutional Neural Networks (CNN). Milestones of image retrieval in the last decade are listed in Fig. 1.

The BoW model was originally proposed for document modeling, because texts can be parsed into words naturally. Basically, this model builds a word histogram for each document by accumulating local responses into a global representation, which text retrieval engines can draw on.

- L. Zheng and Y. Yang are with the Centre for Quantum Computation and Intelligent Systems, University of Technology at Sydney, NSW, Australia. E-mail: liang.zheng@uts.edu.au, yi.yang@uts.edu.au
- Q. Tian is with the Department of Computer Science, University of Texas at San Antonio, TX, 78256 USA. E-mail: qitian@cs.utsa.edu

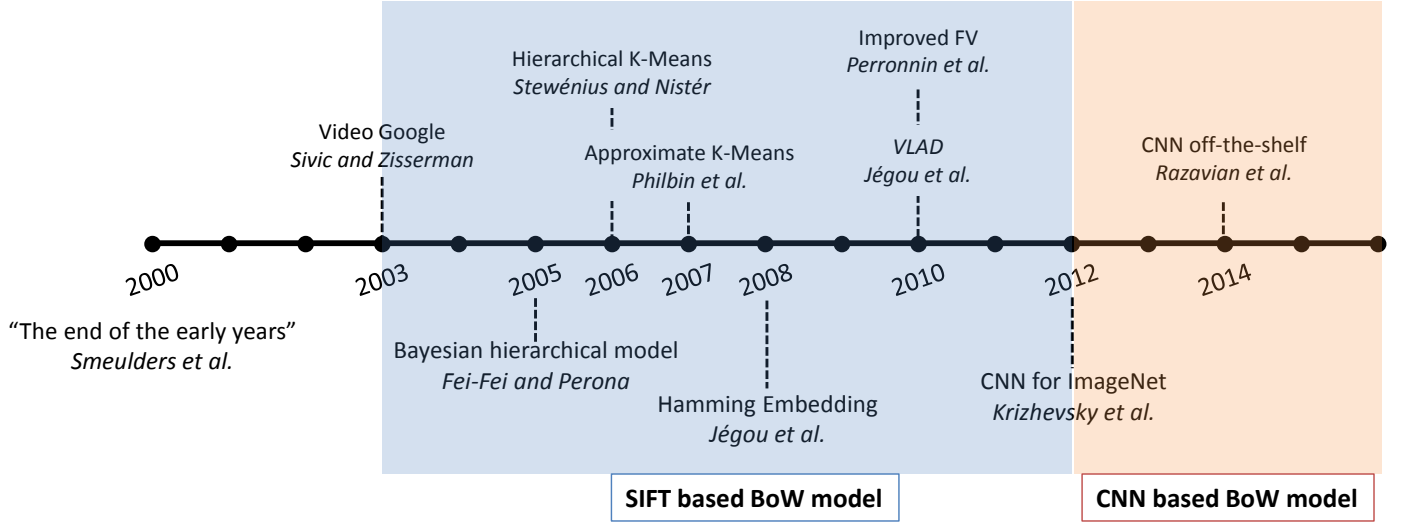


Fig. 1: Milestones [5], [6], [7], [8], [9], [10], [11], [12], [13], [14] of BoW based image retrieval. Times for BoW models based on SIFT or CNN features are covered in blue and red, respectively, marked by the pioneering work of Krizhevsky *et al.* [8].

However, a shortcoming of images is that words are not explicitly available, and the BoW model cannot be applied directly. Fortunately, the introduction of the Scale-Invariant Feature Transform (SIFT) [15] has made this model possible in the image domain. In its original definition, SIFT is comprised of feature detector and descriptor, which is invariant to image translation, scaling, and rotation, as well as partially invariant to illumination changes, and is robust to local geometric distortion. Later, SIFT usually refers to the 128-D descriptor. With this feature, an image can be transformed into a collection of local feature vectors, which can be viewed as prototypes of words in text. In the field of image retrieval, Sivic *et al.* [6] first employed the BoW model using the SIFT feature. With a pre-trained codebook composed of feature vectors of the same dimension, local features of an image are quantized to visual words. An image can thus be represented in the same manner as a document, and classic weighting schemes such as TF-IDF and the inverted index can be incorporated. In essence, the similarity measurement implied in the BoW model with the inverted index is equivalent to the cosine distance between two  $\ell_2$  normalized vectors. While the cosine distance is symmetric, recent works have also proposed asymmetric measurement to account for binarization loss [16] or small queries [17]. In the years that followed, the BoW model was further extended to scene recognition [7] and object categorization [18], [19], in which the inverted index is replaced by the Support Vector Machine (SVM).

The BoW model has been the state of the art among many others [20], [21], [22]. It has greatly advanced the research of instance retrieval in the past ten years, and many improvements have been proposed. In recent years, however, the popularity of SIFT-based models seems to be overtaken by the Convolutional Neural Network (CNN), which is a hierarchical structure that has been shown to outperform hand-crafted features in a number of vision tasks, such as object detection [23], image segmentation [24], and classification [8]. The power of CNN mainly comes from the huge number of free parameters [8] and the use of large-

scale datasets with rich annotations [25]. Using the features extracted from CNN models [26], [27], [28], [29], researchers have reported competitive performance compared to the classic BoW model.

### 1.1 Organization of This Paper

Recognizing that this is a time of change, this paper will provide a comprehensive literature survey of both the SIFT-based and CNN-based BoW retrieval models in the task of instance retrieval. Several closely related aspects will be discussed. We first present the overall retrieval pipeline and method categorization in Section 2. In Section 3, features and their combination schemes will be described. In section 4 and Section 5, we will provide analysis into the quantization process and the data structures for efficiency, respectively. On several benchmark datasets, Section 6 summarizes the comparisons between SIFT- and CNN-based methods. We will provide insights learned from the SIFT feature and point out future directions in Section 7, and conclude this survey in Section 8.

## 2 PIPELINE OF BOW IMAGE RETRIEVAL

### 2.1 Formulations

In this section, we introduce the pipeline formulation of the BoW-based image retrieval [10], [11], [12], [14]. The pipeline is illustrated in Fig. 2. Suppose we have an gallery  $\mathcal{G}$  consisting of  $N$  images. Given a set of  $n$  feature detectors  $\{d_i\}_{i=1}^n$ , we obtain a number of  $n$  heat maps denoted as  $\{\mathcal{M}\}_{i=1}^n$ . A heat map  $\mathcal{M}_i, i = 1, 2, \dots, n$  reflects the response of the input image to the detector  $d_i, i = 1, 2, \dots, n$ . From these heat maps, local descriptors can be extracted from local regions of the sparse interest points or dense patches. Let  $\{f_i\}_{i=1}^D, f_i \in \mathbb{R}^p$  be the local descriptors of  $D$  detected regions in an image.

BoW image retrieval trains a codebook during the offline procedure. A pool of  $p$ -dim local descriptors  $\mathcal{F} \equiv \{f_i\}_{i=1}^M$  are computed from an unsupervised training set. The baseline approach, *i.e.*, k-means, partitions the  $M$  points into  $k$  clusters; the center points of the clusters can be viewed

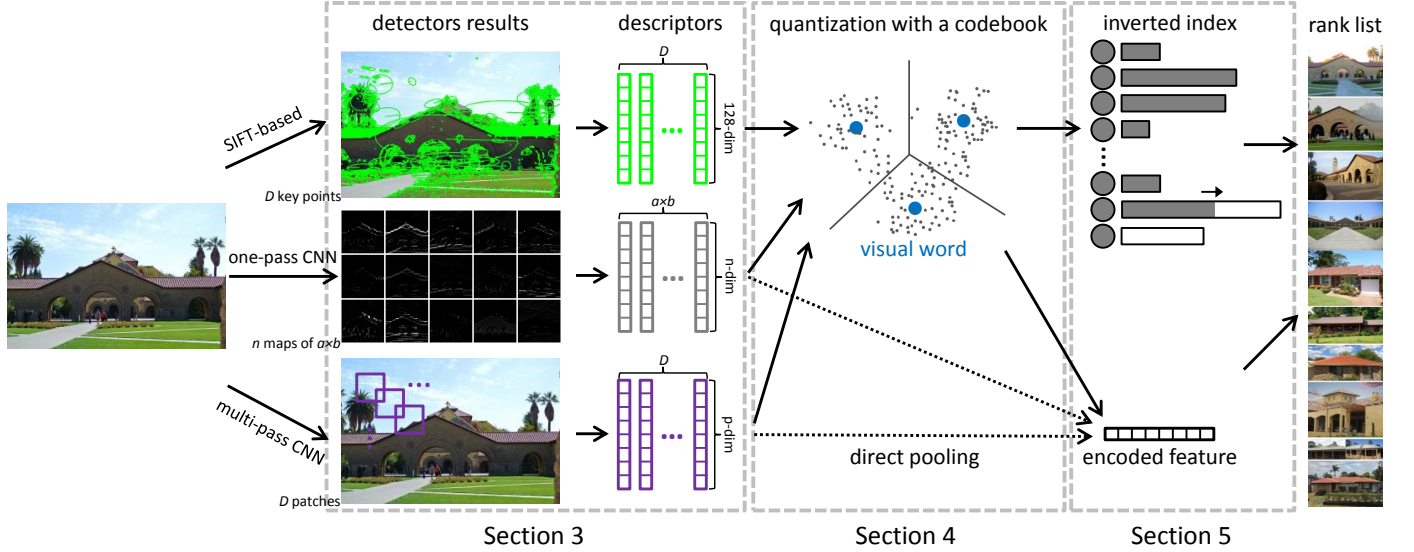


Fig. 2: Pipeline of the Bag-of-Words (BoW) model. For the three method types, feature detection and description are performed in different manners which will be described in Section 3. These local features are quantized to visual words pre-defined in a codebook (Section 4). The inverted index or feature encodings are used for retrieval efficiency, to be covered in Section 5.

as visual words, forming the codebook. Let  $\mathcal{C} \in \mathbb{R}^{p \times k}$  be the codebook matrix of size  $k$  consisting of the  $k$  centers, i.e.,  $\mathcal{C} = [c_1, c_2, \dots, c_k]$ . The objective function of k-means is to minimize the intra-cluster squared distance:

$$\ell(\mathcal{C}) = \sum_{f \in F} \min_i \|f - c_i\|_2^2 = \sum_{f \in F} \min_{b \in \mathcal{B}^k} \|f - \mathcal{C}b\|_2^2, \quad (1)$$

where  $\mathcal{B}^k \equiv \{b \mid b \in \{0, 1\}^k \text{ and } \|b\| = 1\}$  so that  $b$  is a binary vector with 1 entry of value one, and  $k-1$  zeros. By iterative and alternative optimization of Eq. 1 w.r.t  $\mathcal{C}$  and  $b$ , a local minimum can be found, resulting in a codebook for the BoW model. Since  $p$  and  $k$  are fixed, and when a pre-defined number of iterations  $l$  is used, the running time of k-means is often written as  $\mathcal{O}(Mkpl)$ , where  $M$  is the number of training points, and  $k$  and  $p$  denote the number of clustering centers and feature dimension, respectively.

The visual words in codebook  $\mathcal{C}$  are typically assigned specific weights, called the Term Frequency and Inverse Document Frequency (TF-IDF). TF is defined as:

$$\text{TF}(c_i^j) = o_i^j, \quad (2)$$

where  $o_i^j$  is the number of occurrences of a visual word  $c_i$  within an image  $j$ . TF is thus a local weight. IDF, on the other hand, determines the contribution of a given visual word through global statistics. The classic IDF weight of visual word  $c_i$  is calculated as:

$$\text{IDF}(c_i) = \log \frac{N}{n_i}, \text{ where } n_i = \sum_{j \in \mathcal{G}} \mathbb{1}(o_i^j > 0), \quad (3)$$

where  $N$  is the number of gallery images, and  $n_i$  encodes the number of images in which word  $c_i$  appears. Considering both TF and IDF, the baseline weight for visual word  $c_i$  in image  $j$  is:

$$w(c_i^j) = \text{TF}(c_i^j) \text{IDF}(c_i). \quad (4)$$

Given an input local descriptor  $f \in \mathbb{R}^p$ , quantization is performed by mapping  $f$  to an integer index using the pre-defined codebook:

$$q(f) = \min_i \|f - c_i\|_2^2, \quad (5)$$

where  $q(\cdot)$  is the quantization function, through which  $f$  is quantized to the index of its nearest visual word in the codebook.

The baseline BoW retrieval model scores the two local features  $f_1$  and  $f_2$  by the matching function defined as:

$$s(f_1, f_2) = \mathbb{1}(q(f_1), q(f_2)) \cdot \text{IDF}(q(f_1)) \cdot \text{IDF}(q(f_2)), \quad (6)$$

where positive scores will be produced iff.  $q(x) = q(y)$ . For two images  $I_1$  and  $I_2$  comprising of descriptors  $\{f_i^1\}_{i=1}^{D_1}$  and  $\{f_i^2\}_{i=1}^{D_2}$  descriptors, their BoW vectors can be written as  $[w(c_1^1), w(c_2^1), \dots, w(c_k^1)]$  and  $[w(c_1^2), w(c_2^2), \dots, w(c_k^2)]$ , respectively, where  $w(\cdot)$  is defined in Eq. 4. Their image-level similarity is calculated as the sum of all the feature-level matching scores:

$$S(I_1, I_2) = \frac{1}{\ell_1} \frac{1}{\ell_2} \sum_{i=1}^{D_1} \sum_{j=1}^{D_2} s(f_i^1, f_j^2), \quad (7)$$

$$\text{where } \ell_1 = \left( \sum_{i=1}^k w(c_i^1)^2 \right)^{\frac{1}{2}}, \ell_2 = \left( \sum_{i=1}^k w(c_i^2)^2 \right)^{\frac{1}{2}}.$$

In Eq. 7,  $S(\cdot, \cdot)$  is the similarity function of two images, and  $\ell_1$  and  $\ell_2$  are the  $\ell_2$  norms of the two images, respectively. Equation 7 is equivalent to the cosine distance between the BoW vectors of  $I_1$  and  $I_2$ :

$$S(I_1, I_2) = \frac{1}{\ell_1} \frac{1}{\ell_2} \sum_{i=1}^k w(c_i^1) w(c_i^2) \quad (8)$$

where image norms  $\ell_1$  and  $\ell_2$  are defined as Eq. 7. In practice, when large codebooks are used, the BoW vectors are sparse. To reduce memory consumption, the inverted file is usually employed which preserves the cosine similarity.

Method Type	Detector	# Detector ( $n$ )	Descriptor	Dim. ( $p$ )	Quantization
SIFT-based	DoG, Hessian-Affine, <i>etc.</i>	1	SIFT	128	yes
One-pass CNN	Conv. filters of CaffeNet, VGGNet, <i>etc.</i>	>1	FC/column features	$n$	yes/no
Multi-pass CNN	Dense sampling, region proposals, <i>etc.</i>	1	FC/pooled features	not fixed	yes/no

TABLE 1: Major differences between three types of the BoW image retrieval models. For SIFT and multi-pass CNN models, one type of detector is used, while one-pass CNN model uses multiple detectors (conv. filters). The descriptor dimension of SIFT and one-pass CNN models is typically 128 and  $n$ , respectively, but can be high for the multi-pass CNN model.

## 2.2 Three Types of Retrieval Methods

According to the difference in feature extraction procedure, this survey classifies previous methods on BoW image retrieval into three types: SIFT-based, one-pass CNN-based, and multi-pass CNN-based. Their major differences are summarized in Table 1.

**The SIFT-based retrieval model** refers to what had been predominantly studied before 2012 [8]. This line of methods usually use one type of detector ( $n = 1$ ), *e.g.*, Hessian-Affine, coupled with one type of descriptor, *e.g.*, mostly the 128-dim SIFT descriptor ( $p = 128$ ). Since the SIFT descriptor does not have semantic interpretations for its dimensions, quantization (instead of direct pooling) is always performed.

**The one-pass CNN-based model**, as its name implies, passes the image through the deep network only once, producing response maps from the convolutional layers to the Fully Connected (FC) layers. The convolutional filters in CNN can be viewed as local feature detectors. In this model, the number of filters in each layer typically equals to the descriptor dimension ( $n = p$ ), and the number of descriptors is  $a \times b$ , where  $a$  and  $b$  are the height and width of the heap maps in the corresponding layer. Since the CNN features have semantic meaning and stronger discriminative ability, direct pooling methods have also been proven effective.

**The multi-pass CNN-based model** extracts CNN-related features from multiple local regions. This type of methods resembles the SIFT-based model except that a different set of detectors and descriptors are usually used. The dimension of the resulting FC or pooled column features is not fixed and can be high. They are then quantized or directly pooled to yield compact representations (PCA is usually employed). In the following sections, we will review important works falling into the individual modules shown in Fig. 2. In Section 3 and Section 4, previous literature of different methods types will be introduced separately, while Section 5 will review methods in together because these method types usually employ similar indexing and encoding schemes. In Table 2, we summarize the details of some representative methods for each type.

## 3 FEATURES IN THE BOW MODEL

### 3.1 Feature Extraction

#### 3.1.1 SIFT-based

The feature is the engine of computer vision tasks. In image retrieval based on the Bag-of-Words model, local features are employed, composed of both feature detector  $d$  and descriptor  $f \in \mathbb{R}^p$ . The local features can be viewed as prototypes of the words in the text, without which the BoW model would not hold. Since images constantly undergo variations in scale, rotation, illumination, truncation, *etc.*, it

is of vital importance that the local features are invariant to these transformations. We refer readers to several survey papers on local features [30], [31], [32].

In the original work by Lowe *et al.* [15], the DoG (Difference of Gaussians) detector is proposed. In [33], Ge *et al.* further test the performance of LOG and Harris detectors, as well as their combination. Wu *et al.* [34] employ the Maximally Stable Extremal Region (MSER) detector as well as the DoG detector. In recent years, however, most retrieval works [12], [35] have employed the Hessian-affine detector [36], which outputs an elliptical affine invariant region around the centers. The accuracy of the Hessian-affine detector is superior to the DoG detector on several image retrieval benchmarks, given the same descriptor. In [37], Simonyan *et al.* dismiss the orientation estimation, *i.e.*, using gravity assumption, in the Hessian-affine detector, and demonstrate consistent improvement on architecture datasets.

For feature description, on the other hand, SIFT [15] has been used as the default descriptor. A detected region is represented by a 128-dimensional SIFT vector, which has been shown in [38] to outperform other competing descriptors such as moment invariants [39], Shape context [40], and Steerable filters [41], in terms of matching accuracy. In [42], Arandjelović and Zisserman propose the RootSIFT variant by two steps: 1)  $\ell_1$  normalize the SIFT descriptor, 2) square root each element. It is shown in [42] that comparing RootSIFT with Euclidean distance is equivalent to using the original SIFT with the Hellinger kernel. RootSIFT has been shown to yield superior performance to original SIFT and is now used as a routine in the SIFT-based retrieval models. Ge *et al.* also test the DAISY descriptor [43] in retrieval. In [44], Tolias *et al.* propose not to align the detected patches by dominant orientations, but to jointly encode the angle in a continuous manner in the aggregation stage.

#### 3.1.2 One-pass CNN

Image filters are typically referred to when the feature detectors of the one-pass CNN-based models are considered. The Convolutional Neural Network has a hierarchical structure. From bottom to top layers, the image undergoes convolution with filters, and the receptive fields of these image filters increase from the bottom layers to the top layers. Generally speaking, filters in the same layer have the same size but different parameters. For example, in AlexNet [8], there are 96 filters in the first layer of sizes  $11 \times 11 \times 3$ , while in the 5th layer, 256 filters exist and are of size  $3 \times 3 \times 192$ . In [45], Zeiler *et al.* observe that the filters are sensitive to certain visual patterns, and that these patterns evolve from low level bars in bottom layers to high level objects in top layers. For low-level and simple visual stimulus, the CNN filters act as the detectors in the local hand-crafted features, but for



the high-level and complex stimulus, the CNN filters have distinct characteristics that depart from SIFT-like detectors.

The most commonly used detector in the one-pass CNN model has a global receptive field [9], [46]. The resulting global FC feature is used to perform a linear scan of the image database. Many retrieval works [29], [47], [48] have recently focused on intermediate level detectors, *e.g.*, the lower-level filters are used to detect local visual patterns. Compared with the global receptive field, local or mid-level detectors are more robust to image transformations such as truncation, occlusion, and illumination changes, in ways that are similar to the invariant local detectors. Typically a pre-trained CNN model is used, so the filters are fixed too. But in [49], the Landmarks dataset containing various architectures is collected and used to fine-tune the ImageNet pre-trained AlexNet model. The re-trained network (filters) produces superior features on Landmark related retrieval datasets such as Oxford5k [11] and Holidays [12], but has decreased performance on the Ukbench dataset [10] where common objects are presented. In [50], Radenovic *et al.* employ the retrieval and Structure-From-Motion methods to build 3D landmark models to select positive and hard negative training samples. The Siamese Network is then employed to train a CNN model. The fine-tuned model again exhibits good accuracy on landmark retrieval datasets, but its generalization ability on non-landmark datasets remains unknown.

In the one-pass CNN model, the descriptor is tightly coupled with the detectors; the resulting activation maps of the convolution between the image and CNN filters can be viewed as a feature ensemble, which is called the “column feature” in Table 1 following the HyperColumn representation proposed by Hariharan *et al.* [51]. For example in AlexNet [8], there are  $n = 96$  detectors (convolutional filters) in the 1st convolutional layer. These filters produces  $n = 96$  heat maps of size  $27 \times 27$  (after max pooling). Each pixel in the maps has a receptive field of  $19 \times 19$  and records the response of the image w.r.t the corresponding filter [29], [47], [48]. The column feature is therefore of size  $1 \times 1 \times 96$  ( $p = n = 96$  in Fig. 2) and can be viewed as a description of a certain patch in the original image. Each dimension of this descriptor denotes the level of activation of the corresponding detector and resembles the SIFT descriptor to some extent. As mentioned above, the global detector and descriptor are most commonly used, and the dimension of the descriptor is 4,096 in standard AlexNet when extracted from the fully connected layers (FC6 or FC7) [9], [46]. In [49], [50], the CNN models are fine-tuned on newly collected instance-level dataset, and the resulting network is shown to outperform the pre-trained CNN model in related test sets.

### 3.1.3 Multi-pass CNN

The multi-pass CNN model is similar to the SIFT-based model with a different set of detector and descriptor. With respect to the formulations in Section 2, the multi-pass model is featured by using a single feature detector ( $n = 1$ ) which produces multiple image regions and requires to access to the CNN model for multiple times. The local detectors can be multi-scale image patches [26], the Difference of Gaussian feature points [52], regions generated by MSER [53], selective search [54], [55], edgebox [56], [57], [58] or the



Fig. 3: False match removal by (A) Hamming Embedding [12], (B) local-local feature fusion, and (C) local-global feature fusion.

Region Proposal Network (RPN) [59], [60]. In [26], a two-scale sliding window strategy is employed to generate patches. In [9], the dataset images are first cropped and rotated, and then divided into patches of different scales, the union of which covers the whole image. It is also a good practice to use region proposals to generate potential object regions [54], [56], [60].

For the region descriptors, the multi-pass CNN retrieval method typically employs the FC or pooled intermediate CNN features. Fischer *et al.* [53] evaluate the image matching accuracy of CNN and SIFT and demonstrate that CNN is superior except on blurred images. The CNN features used in [53] are extracted from layers 1-4 in AlexNet which is trained either on ImageNet or self-collected image patches. In [26], the 4,096-dim FC features are extracted from the multi-scale image regions. These features are subsequently aggregated to form a compact vector. The same FC descriptor is employed in [54] to describe the region proposals before a max-pooling aggregation step.

## 3.2 Feature Fusion

### 3.2.1 SIFT-based

**Local-local fusion.** A problem with the SIFT feature is that only local gradient description is provided. Other discriminative information encoded in an image is still not leveraged. For example, two detected regions may be very similar in SIFT space, but they can be very different in other feature spaces, which may determine that they belong to a false match pair to be rejected. A good example is to couple SIFT with color descriptors. For example, the Bag-of-Colors (BoC) method [61] extracts a 256-dim color histogram from the elliptical region of each SIFT descriptor in the CIE-Lab color space. It is binarized to combine with the SIFT binary codes [12]. A similar idea is reflected in [35] where the local Color Names descriptor [62] is extracted by coupling with SIFT. Both methods achieve competitive accuracy on several benchmark datasets, but a potential problem is that intensive variation in illumination may compromise the effectiveness of colors. Also, in some cases, objects with smooth surfaces [63] tend to have fewer keypoints than highly textured objects. The Bag-of-Boundary approach [63]

performs foreground segmentation and employs a set of boundary features for sculpture retrieval. In Fig. 3 (B), a pair of false match cannot be rejected by Hamming Embedding due to their similarity in the SIFT space, but the fusion of other local (or regional) features may correct this problem.

**Local-global fusion.** Local and global features describe images from different aspects. Due to the intense image variations in retrieval, local features may be less effective. For example, color feature may be more effective than SIFT when a smooth object is present. Therefore, how to combine local and global information remains an interesting topic. In Zhang *et al.*'s work [64], an offline graph is built for each type of feature, which is subsequently fused during query. In an improvement of [64], Deng *et al.* [65] add weakly supervised anchors to aid graph fusion. In both methods, global features such as GIST [66] and HSV histogram are extracted from images. Another concurrent work in local-global fusion consists of co-indexing [67]. The global features such as the 1000-dim activation extracted from fully-connected layer 8 in AlexNet [8] are fused in the inverted index built with SIFT. In [68], automatically learned category-specific attributes are combined with pre-trained category-level information on the score level for retrieval. A recent work, *i.e.*, query-adaptive late fusion by Zheng *et al.* [69], extracts a number of features (local or global, good or bad) and weights them in a query-adaptive manner. In Fig. 3 (C), when local (and regional) cues are not enough to reject a false match pair, it would be effective to further incorporate visual information from a larger context scale.

### 3.2.2 One-pass CNN

For the one-pass CNN model, fusion strategy is relatively straightforward. Since the image is passed through the network only once, a natural idea is to fuse CNN activations from multiple layers. This is in essence a multi-scale fusion strategy because different CNN layers have different sizes of the receptive field. In [70], Hariharan *et al.* propose the "Hypercolumn" by resizing convolutional maps in each layer and concatenating the column features on each pixel position. In [48], pooled multi-layer CNN vectors are concatenated for holistic image recognition and retrieval. It is shown in both works the fusing multi-layer column features brings benefit over the single-layer column features.

### 3.2.3 Multi-pass CNN

Fusion methods in Multi-pass CNN is richer compared with the one-pass case. Two possible strategies are widely adopted in literature, *i.e.*, multi-scale CNN feature fusion and heterogeneous fusion of CNN and other descriptors. Both strategies take advantage of the complementary nature of the features to be fused. In the multi-scale fusion strategy, image patches of multiple scales are generated, from which the CNN features are extracted and subsequently pooled [26]. Yoo *et al.* [71] use Fisher Vector in image classification to encode the FC layers extracted from multiple image scales. In semantic segmentation, Farabet *et al.* [72] directly concatenate the CNN features of the same layer but with multiple scales of the image.

A typical case in the fusion of heterogeneous descriptors is the fusion with SIFT. Although CNN resembles SIFT in

many aspects, fundamental differences nevertheless exist. It is known that SIFT is robust to image variances [15] in rotation, scale, illumination, *etc.* It had also been shown that CNN has some limited variance to rotation and illumination [73], but good invariance to scale changes [74]. Another major difference is that SIFT does not capture semantic information, while the CNN feature from top layers is more sensitive to high-level semantics such as parts or objects. Chandrasekhar *et al.* [74] find that unlike image classification, CNN does not necessarily always outperform SIFT in image retrieval. Instead, mixing both approaches seems to be superior. In [75], regional and global CNN features are fused with SIFT under the BoW framework with a probabilistic model, so that accurate visual matching can be achieved.

## 3.3 Geometric feature combination

A frequent concern with the BoW model is the lack of geometric constraints among local features. For example, the word sequence in the text follows a 1-dim structure, while the pattern of local features in the image have 2-dim structure. In object recognition, spatial pyramid pooling [76], [77] (or the better-known Spatial Pyramid Matching, SPM [76]) has been identified as one of the most successful methods. It quantizes the image space from coarse to fine, from which local features are aggregated and concatenated to incorporate rigid spatial information. The GIST feature [66] shares a similar idea with SPM. These methods lack invariance to image transformations.

### 3.3.1 SIFT-based

In image retrieval, global spatial verification based on RANSAC [11], [78] is effective in re-ranking a subset of top-ranked images, but is prone to efficiency problems. As a result, how to efficiently and accurately incorporate spatial cues in the BoW model has been extensively studied. For example, visual phrases [79], [80], [81], [82], [83] are generated among individual visual words to provide more strict matching criterion. In [84], visual word co-occurrences in the entire image are estimated, while in [34], [85], [86] visual word clusters within local neighborhoods are discovered. Zhang *et al.* [82] propose the Geometry-preserving Visual Phrases (GVP) by projecting both long-range and short-range feature combinations onto an offset space and calculating the matching scores based on the aggregation statistics in the space. In [83], descriptive visual phrases (DVP) are generated for effective visual object representation by visual word combinations. Visual phrases can also be constructed from adjacent image patches [79], random spatial partitioning [81], and localized stable regions [34] such as Maximally Stable Extremal Region (MSER) [87]. Hao *et al.* [88] formulate a 3D visual phrase as a triangular facet on the surface of a reconstructed 3D landmark model which explicitly describes the spatial structure of a 3D object. Evolving from visual word to visual phrase, Zheng *et al.* [89] introduce the concept of the visual phraselet for even finer visual matching by grouping spatially consistent visual phrases. This type of methods has been popular since visual words, analogies to words in the text, have been aggregated to those parallel to phrases in the text.

Spatial information can be stored in the inverted index [90], [91] and will be introduced briefly in Section 5.1. For

example, weak geometric consistency (WGC) [12] uses an additional 12 bits per feature for its orientation and scale, and verifies the consistency of angle and scale between the matched local features. Shen *et al* [92] and Lampert *et al* [93] measure the similarity of images by matching the localized sub-windows of a gallery image, and apart from the ranking results, provide the localization of the query pattern. In [94], pairwise matched keypoints are used to estimate global orientation and scale variances, and the two parameters validate each other for spatial verification.

### 3.3.2 One-pass CNN

In the one-pass CNN model, the intermediate column features are viewed as special formulation of the local features, and each filter generates a heat map on the image that denotes the extent of the activation of the image filters. Regardless of whether quantization is performed on these column features, similar geometric issues exist with these CNN activations as with those of SIFT. A feasible strategy is to use direct pooling to gain invariance to image translations. For example, Tolias *et al.* [29] and Zheng *et al.* [48] propose to pool (max or average pooling) each feature map into a single activation, recording the matching strength of the image and the corresponding filter. As an improvement, Razavian *et al.* [95] propose the application of spatial pooling with a grid of size  $2 \times 2$  on the feature maps and to produce a longer vector, but one that is more sensitive to spatial constraints. While the above works does not include a quantization step, Mohedano *et al.* [96] propose to use quantized feature maps to locate objects by spatial re-ranking, which is employed for the query expansion process characterized by sum pooling.

### 3.3.3 Multi-pass CNN

Pooling is also a good option to improve the robustness to object translations in the multi-pass CNN model. For example, the CNN features from image patches cropped from various scales and positions in [26] are extracted and subsequently pooled to gain invariance to object scale and translation changes. In [54], selective search [55] is leveraged to detect the likely positions of objects from which the CNN features are extracted and pooled. In [60], the CNN features are extracted from the RPN [59]. After a filtering stage using image-level pooled descriptors, the region-level descriptors are employed to calculate the regional similarities with the query region, and a spatial-aware re-ranking module can be achieved. In SIFT-based retrieval, if a visual word appears in an image several times, max pooling is equal to the situation in which the occurrence of the word is counted as 1, and average pooling is more like the TF weights that reflect the total number of occurrences. On the direction of geometric-aware feature design, the CNN feature still has a long way to go when compared with the number of works associated with SIFT features based on the inverted index [82], [83], [90]. How to take advantage of the inverted index to embed spatial clues and yet enable efficient retrieval is a very promising research topic.

**Summary.** From the perspective of feature extraction, current instance retrieval methods can be classified into three types: SIFT-based, one-pass CNN, and multi-pass CNN. All three method types exhibit the “detector + descriptor” mode. For

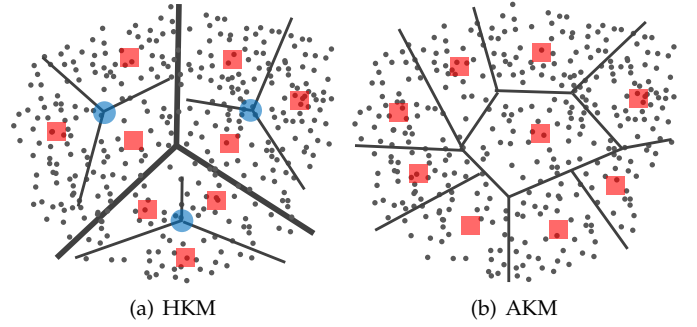


Fig. 4: Two milestone clustering methods (a) Hierarchical K-Means (HKM) [10] and (b) Approximate K-Means (AKM) [11] for BoW codebook generation. Bold borders and blue discs are the clustering boundaries and centers of the first layer of HKM. Slim borders and red squares are the final clustering results in both methods.

SIFT-based methods, the most commonly used combination is Hessian-affine detector + SIFT descriptor due to its superior retrieval accuracy. For the other two retrieval models, the convolutional filters and image patches are used as feature detector, respectively; the column/FC features and different forms of pooled features are used as descriptor, respectively. Since single descriptors may have limited descriptive or scale ability, heterogeneous or multi-scale feature fusion can be leveraged to improve the overall performance. The geometric-aware methods are also effective in filtering out false local matches or improve feature robustness to object translations. This type of methods is required to calculate efficient spatial signatures and avoid a matching procedure that is too rigid due to object deformation.

## 4 QUANTIZATION

The core component in the BoW model is called the “visual word”. Typically, in the offline procedures, a codebook is obtained. Given a local feature (belonging to a database or query image), the BoW model locates certain entries in the codebook, the index of which is taken as the quantization result. In this survey, previous methods of two important steps will be described.

### 4.1 Codebook Generation

The codebook can be viewed as a set of basis vectors in the high dimensional feature space. Each visual word in the codebook lies in the center of a sub-space, called the “Voronoi cell”. A larger codebook corresponds to a finer partitioning, resulting in more discriminative visual words. In contrast, when the codebook is small, Voronoi cells are typically large, so such a coarse space partitioning leads to less discriminative visual words. To partition a pool of features into clusters, a theoretical choice of clustering method is k-means [104], also called Lloyd’s algorithm (Eq. 1). It starts from a set of  $k$  centers obtained by random initialization or priors,  $\{c_1^{(1)}, \dots, c_1^{(k)}\}$ , and then alternatives the following two steps:



Method Type	Methods	Detector	Descriptor	Quantization	Index	Indexed Content
SIFT-based	HKM [10]	MSER	SIFT	Voc. tree	Inv. Index	Image ID, TF
	AKM [11]	Hessian-Affine	SIFT	ANN	Inv. Index	Image ID, TF
	HE [12]	Hessian-Affine	SIFT	ANN	Inv. Index	SIFT bin. codes
	Bundling [97]	DoG & MSER	SIFT	Soft [98]	Inv. Index	Bundle info.
	VLAD [14]	Hessian-Affine	SIFT	NN	PQ [14]	-
	FV [99]	Hessian-Affine	SIFT	NN	PCA	-
	GVP [82]	Hessian-Affine	SIFT	ANN	Inv. Index	Quan. offset
	SQ [100]	DoG	SIFT	Scalar	Inv. Index	SIFT bin. codes
	c-MI [35]	Hessian-Affine	SIFT & color	ANN & NN	2-D Index [101]	SIFT & color bin. codes
Multi-pass CNN	Off the Shelf [9]	Dense patches	FC7 (AlexNet)	-	Linear	-
	MOP [26]	Dense patches	FC7 (AlexNet)	NN	VLAD	-
	CKN [102]	Hessian-Affine	Patch-CKN	NN	VLAD	-
	OLDFP [54]	Selective search [55]	FC7 (AlexNet)	-	MaxPooling, ITQ [103]	-
	MSS [95]	Dense patches	Conv5 (VGG-16)	-	Linear	-
One-pass CNN	BLCF [96]	VGG-16	Conv5	ANN	Inv. Index	-
	Neural Codes [49]	AlexNet ft. Landmarks	Layer 5, 6, 7	-	PCA, DDR	-
	SPoC [27]	VGG	Conv5	-	PCA <sub>w</sub>	-
	MAC [29]	VGG	Conv5	-	PCA <sub>w</sub>	-
	Siamese [50]	VGG ft. new data	Conv5	-	PCA <sub>w</sub>	-

TABLE 2: Categorization and details of representative image retrieval works.

- Assignment step: given the current set of cluster centers  $\{c_1^{(t)}, \dots, c_j^{(t)}\}$ , assign each point  $f_i$  to its closest cluster center:

$$s_i^t = \arg \min_j \|f_i - c_j^{(t)}\|_2^2. \quad (9)$$

- Update step: the points in each cluster is  $\mathcal{S}_j^{(t)} = \{f_i | s_i^{(t)} = j\}$ , and compute the new center for each cluster as,

$$c_j^{(t+1)} = \frac{1}{|\mathcal{S}_j^{(t)}|} \sum_{f_i \in \mathcal{S}_j^{(t)}} f_i. \quad (10)$$

The computational complexity of the assignment step and the update step is  $\mathcal{O}(Mk)$  and  $\mathcal{O}(M)$ , respectively, where  $M$  is the total number of training points.

#### 4.1.1 SIFT Features

In image retrieval, the size of the codebook varies from several hundred to several million [11], [14], [67], [99], [105]. In VLAD (Vector of Locally Aggregated Descriptors) and FV (Fisher Vector) based works, the codebook sizes are typically around several hundred, *e.g.*, 64, 128, 256. With such small codebooks, first-order [14] and second-order [99] residuals can be encoded into a descriptor with acceptable dimension (Principal Component Analysis is still necessary for dimension reduction). In VLAD, exact K-Means is used, while in FV, the Gaussian Mixture Model (GMM) is trained using Maximum Likelihood (ML) estimation. This line of method is popular because the generated signature is straightforward to use (Euclidean distance) when further acceleration is available [106]. However, when small codebooks are used, the visual words are not discriminative enough. The incorporation of SIFT residuals, such as VLAD [14] and Triangulation Embedding [107], is beneficial but this compensation is limited and the retrieval accuracy on some benchmarks is inferior to larger codebooks [107].

At the other end of the scale, some works in the literature use large codebooks containing 1 million [10], [11] or more [67], [105] visual words. In such cases, the visual words are more discriminative, so the baseline BoW model has a higher accuracy on benchmarks [11], [67], [105], [108], [109]. Two representative works are Hierarchical K-Means (HKM) [10]

and Approximate K-Means (AKM) [11], as illustrated in Fig. 4 and mentioned Fig. 1. Proposed in 2006, HKM applies standard k-means on the training features hierarchically. It first partitions the points into a few clusters (*e.g.*,  $\bar{k}$ ), and then recursively partitions each cluster into further clusters. In every recursion, each point should be assigned to one of the  $\bar{k}$  clusters, with the depth of the cluster tree being  $\mathcal{O}(\log k)$ , where  $k$  is the target cluster number. The computational cost of HKM is therefore  $\mathcal{O}(\bar{k}M \log k)$ , much smaller than the complexity of standard k-means  $\mathcal{O}(Mk)$  when  $k$  is large.

Another milestone work in BoW codebook generation is Approximate K-Means (AKM) [11]. This method indexes the  $k$  cluster centers using a forest of random  $k$ -d trees, so that the assignment step can be performed efficiently with Approximate Nearest Neighbor (ANN) search. In AKM, the cost of assignment can be written as  $\mathcal{O}(k \log k + vn \log k) = \mathcal{O}(vn \log k)$ , where  $v$  is the number of nearest cluster candidates to be accessed in the  $k$ -d trees. We can find that the computational complexity of AKM is on par with HKM, and is significantly smaller than standard k-means when  $k$  is large (a large codebook). Nevertheless, experiments on the retrieval benchmarks clearly show that AKM yields superior accuracy to HKM. The reason is that in HKM, when quantizing (assigning) a feature descriptor, it is possible that quantization error occurs at an initial level of the tree, so both the quantization and codebook training processes can be compromised.

In other examples, larger codebooks [67], [105] of size over 10 million are constructed. Here, Hierarchical K-Means (HKM) is employed, which builds a tree structure for fast clustering. In [110], Zhou *et al.* propose a scalar quantization scheme, in which no explicit codebook is trained. Instead, local features are transformed to binary features, and the codebook space is thus defined by the binary space spanned by the binary numbers. For 256-bit binary features, the spanned space contains  $2^{256}$  visual words. In Product quantization (PQ) [106], the theoretical codebook size can be also large as to  $2^{64}$  considering the Cartesian product. The small sub-codebooks used in PQ are trained by standard k-means. It should be noted that PQ is a quantization method for ANN search, not for the quantization step in BoW retrieval.

Higher retrieval performance can be obtained by median-sized codebooks. For example, Jégou *et al.* [12] construct a



codebook of size 20k while Tolias *et al.* [111] find that a 64k codebook yields superior accuracy when combined with a modified Hamming Embedding matching procedure. For heterogeneous codebooks where information from multiple domains is utilized, Zheng *et al.* separately train the SIFT and Color Names (CN) codebooks with size of 20k and 200, respectively, yielding a visual word space of 4 million visual word tuples. In the Edgel Index [112], the codebook is composed of a position channel and an orientation channel, and a codebook of  $40k \times 6$  entries is constructed.

#### 4.1.2 CNN Features

The codebook generation methods for CNN features are similar for both the one-pass and multi-pass models (Table 1), given the pool of extracted local features. When extracting CNN as local features on local patches, the codebook generation process is similar to that of the SIFT feature. Nevertheless, the practice of extracting hundreds or thousands of CNN features from image patches has not been sufficiently popular, so codebook generation methods have not been extensively studied. Of the few that have, Gong *et al.* [26] adopt VLAD encoding to pool CNN features from image patches. In this process, a small codebook (*e.g.*,  $k = 100$ ) is trained by classic K-Means. While a small codebook favors the generation of compact vectors, large codebooks can provide more discriminative representations [11], [105] and enable time efficiency by the inverted index, at the cost of higher memory consumption compared to compact vectors [113]. Therefore, a possible future direction is to design discriminative large vocabularies sheltered by the inverted index.

## 4.2 Visual word assignment

### 4.2.1 SIFT-based

With a pre-defined codebook, local features can be quantized to visual words (Eq. 5). This is the key step in the BoW model, as it explicitly transforms an image into a representation that resembles the text. The biggest problem associated with this process is the information loss from a full 128-dim vector to one or several integers. The conventional method is hard quantization [11], in which one visual word is assigned to each input local feature. A number of approaches have been explored to reduce quantization error. Philbin *et al.* [114] propose quantizing a feature into several nearest visual words in the codebook; the weight of each assigned visual word relates negatively to its distance from the feature by  $\exp(-\frac{d^2}{2\sigma^2})$ , where  $d$  is the distance of the descriptor to the cluster center. The idea of “soft quantization” has also been employed in Fisher Vector, in which the weights of each Mixture are determined by the possibility that a feature belongs to the  $i$ th Gaussian Mixture. Quantization with soft-like schemes also includes sparse coding [33], in which the sparse coefficients reconstruct the input local feature with constrained error bound. In [105], Mikulik *et al.* propose to learn some alternative visual words for each visual word which are likely to contain descriptors of matching features, and take up fixed memory consumption. For medical image retrieval, Wang *et al.* [115] solve the reconstruction weights with Quadratic Programming (QP) on the neighboring visual words. In [116], Cai *et al.* suggest that when a local feature

is far away from even the nearest visual word, this feature can be discarded without performance drop. In [100], [117], [118], [119], local features are quantized without an explicitly trained codebook, which is called “Scalar Quantization”. A 128-dim floating point vector is first binarized and the first dimensions of the resulting binary vector are directly converted to a decimal number as a visual word. In the case of large quantization error and low recall, Scalar Quantization uses bit-flop to generate hundreds of visual words for a local feature. To address the recall problem, Liu *et al.* [120] propose cross-indexing to take advantage of both traditional and scalar quantization.

### 4.2.2 CNN-based

For both the one-pass CNN and multi-pass CNN models, the quantization process is similar. When aggregating local descriptors into VLAD [14], nearest neighbor search is usually used due to the small codebook size [26], [47], [102]. The quantization error is reduced by the accumulating residual vectors in VLAD. In [96], an assignment map is produced after the quantization of all the local CNN descriptors with a codebook of size 25k, which is further used for spatial re-ranking. In some other cases, direct pooling is used, such as MAC (Maximum Activations of Convolutions) [29], [48]. To our knowledge, works on large codebooks and approximate quantization methods have not been extensively studied in the CNN-based BoW model. Future investigation is therefore in need on how to effectively quantize local and patch level CNN features for efficient retrieval.

## 4.3 Feature weighting

### 4.3.1 SIFT-based

In the BoW model, visual words are assigned weights in various ways. In the *de facto* standard, the Term Frequency (TF) (Eq. 2) and the Inverse Document Frequency (IDF) (Eq. 3) are both used to define the weight of each feature. They have also been applied in the field of natural language processing for explicitly extracting key phrases from the text [121]. TF counts the number of occurrences of a feature within each image, and thus is informative about textures. IDF, on the other hand, determines the contribution of a given visual word in a global manner. The presence of a less common visual word in an image may be a better discriminator than a more common one. The TF and IDF formulas are presented in Eq. 2 and Eq. 3, respectively.

In text retrieval, the TF-IDF scheme is predominantly used, and several effective variants have been proposed to effectively estimate word weighting. For example, Okapi BM25 [122] additionally takes into account the average document length and the current document length, as well as other hyper-parameters. The heuristic improved TW-IDF scheme [123] integrates a graph-based TF-like function. Both methods are variants of the local weighting scheme, *i.e.*, TF, and use the same global term weighting, *i.e.*, IDF as defined in Eq. 3. Many other alternatives exist, *e.g.*,  $x^I$  [124] and Residual IDF [125]. In text stream analysis and text classification, task-specific global term weighting schemes such as Inverse Corpus Frequency (ICF) [126] and Relevant Frequency (RF) [127] have also proven to be effective. Despite

the simplicity of the IDF calculation in Eq. 3, its robustness has been backed up by many justifications [128], [129].

Although extensive studies on TF-IDF have been conducted in text analysis, relatively fewer are witnessed in image retrieval. An important problem associated with visual word weighting is burstiness [130], [131], [132]. This refers to the phenomenon whereby repetitive structures appear in an image: “if a word appears once, it is more likely to appear again” [130]. This problem tends to dominate image similarity and affects retrieval quality to a large extent. In the image domain, Jégou *et al.* [132] propose several TF variants to deal with burstiness. An effective strategy consists of exerting a square operation on TF, which is in spirit inspired by text retrieval [131]. In [133], Revaud *et al.* propose learning the keypoint groups that frequently correlate with one another to learn from incorrect detections, and to down-weight these groups in the scoring function. Shi *et al.* [134] propose the explicit detection of bursty groups, from which meta-feature is formed and fed into the BoW model. For the VLAD and Fisher Vector representations, an improved version [113] employs the power-law normalisation [99] to tackle with the multiple match and burstiness problem. It is a simple post-processing method that transforms an input feature vector  $\mathbf{f} = (f_1, \dots, f_p)$  into  $f_i := \text{sign}(f_i) \times \|f_k\|^\alpha$ , where  $0 \leq \alpha < 1$  is a constant. Zheng *et al.* [108] propose the  $\mathcal{L}_p$ -norm IDF to tackle burstiness while Murata *et al.* [135] design the exponential IDF which is later incorporated into the BM25 formula. Both methods are shown to be effective in image and video retrieval, respectively. While most works view burstiness as a negative problem, Akihiko *et al.* [136] make use of this phenomenon, which appears more frequently in architectures, and then design new similarity measurement following burstiness detection.

#### 4.3.2 One-pass CNN

In one-pass CNN representations, the feature maps within each layer can be assigned specific weights before pooling. In [27], Babenko *et al.* propose the injection of the prior knowledge that objects tend to be located toward image centers, and impose a 2-D Gaussian mask on the feature maps before sum pooling. In [28], Kalantidis *et al.* propose performing both feature map-wise and channel-wise weighing, which aims to highlight the highly active spatial responses while reducing burstiness effects. In [29] and [48], activations of the last convolutional layer undergo sum or max pooling, with additional prior of object localization [29]. Xie *et al.* [137] improve the MAC representation [29], [48] by propagating the high-level semantics and spatial context to low-level neurons for improving the descriptive ability of these bottom-layer activations.

#### 4.3.3 Multi-pass CNN

For multi-pass CNN features, when compact vectors are employed, the quantization weights are assigned through the Gaussian Mixture Models. Current literature focus more on the feature detection and description parts, so open issues are exist on effective weight assignment to the CNN visual words or descriptors. For example, IDF-like schemes may be useful to down-weight image patches of low discriminative ability and vice versa. It is also interesting to train CNN models tuned for instance similarity. Previous works such as [49], [50]

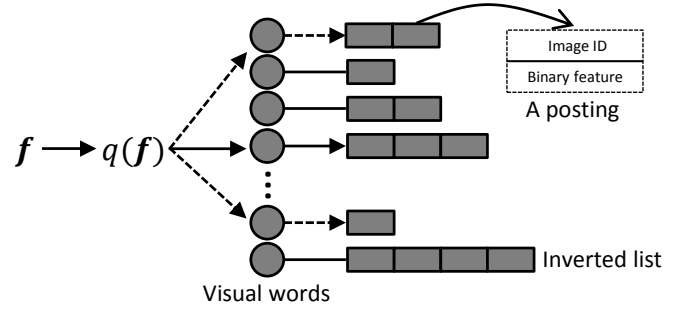


Fig. 5: Data structure of the inverted index. It physically contains  $k$  inverted lists, where  $k$  is the codebook size. Each inverted list consists of a number of postings, which index the image ID and some binary signatures. During retrieval, a quantized feature will traverse the inverted list corresponding to its assigned visual word. Dashed line denotes soft quantization, in which multiple inverted lists are visited.

collect instance level datasets by supervised/unsupervised methods and fine tune the existing CNN models. The resulting models are, however, effective in landmark-specific retrieval which is suggested by the content of the training data. It is therefore necessary that more generic instance retrieval models can be trained from new large-scale datasets preferably with less amount of human labor.

## 5 DATA STRUCTURES FOR EFFICIENT RETRIEVAL

Given the quantization results of both the SIFT and CNN features, previous works apply either the inverted index (Section 5.1) or compact feature embeddings (Section 5.2) to achieve efficient retrieval. The two strategies are based on large and small codebooks, respectively.

### 5.1 Inverted Index

#### 5.1.1 Structure and Indexed Data

The inverted index is designed to enable efficient storage and retrieval. Typically, it is useful when a relatively large codebook is employed. Its structure is illustrated in Fig. 5. As a baseline method, the inverted index is a one-dimensional structure in which each entry corresponds to a visual word in the codebook. An inverted list is attached to each word entry, and those which are indexed in the each inverted list are called indexed features or postings.

For local features, the inverted index is an efficient structure which takes advantages of the sparse nature of the visual word histogram given a large codebook. For example, if the codebook size is 1 million and an image has 1,000 local features, the visual word histogram will at most have  $10^{-3}$  non-zeros entries (assuming that all the local features are quantized to entirely different visual words). Under such circumstances, the memory usage of a  $10^6$ -dim vector is about 1M bytes (unsigned char precision); when using the inverted index, the memory cost should be 4K bytes, assuming that each indexed feature takes 4 bytes, which is a 250-fold reduction. Therefore, when it comes to large codebooks, the inverted index is a necessary component of the retrieval engine.

In literature, new algorithms are designed to be adjustable to the inverted index, which in turn generates a number of index variants. In the baseline method [10], [11], the image ID and Term Frequency (TF) are stored in a posting. In [138], Wang *et al.* additionally incorporate quantized descriptor contextual weight, descriptor density, mean relative log scale, and the mean orientation difference in each posting, to enhance the discriminative power of visual words along the vocabulary tree [10]. Zhang *et al.* [82] propose storing the quantized spatial information in the postings. The spatial information is then used for online offset space computation and Geometry-preserving Visual Phrase (GVP) generation. An improved version of GVP appears in [89]. In a similar manner, Zhou *et al.* [90] additionally store the quantized orientation of each feature in the postings, and generate a spatial map for each image out of the relative positions of local features. In another example, Liu *et al.* [91] generate binary spatial context signatures and index them within each posting for post-matching verification. Another important line of index modification is to inject heterogeneous features which are in complementary to SIFT. For example, in [35], [61], a binary color signature is generated with the local SIFT descriptor, and stored in the postings for verification. These binary signatures are memory efficient and largely improve the matching precision of SIFT visual words. In [67], Zhang *et al.* expand the inverted index with globally consistent neighbors to the indexed images; the increase in memory consumption is alleviated by the option of semantic isolated image deletion. Liu *et al.* [120] proposes the cross-index strategy by building two inverted indices, *i.e.*, by classic BoW and by scalar quantization [100]; the two indices interact with each other to boost retrieval recall in a soft quantization manner.

In an expansion of the traditional inverted index, Zheng *et al.* [35] propose the coupled Multi-Index (c-MI) for feature fusion. This is an extended version of the Multi-Index proposed by Babenko *et al.* [101] which is applied in the Approximate Nearest Neighbor (ANN) search of local descriptors. The Multi-Index is capable of increasing ANN accuracy by enforcing the same pair of visual words quantized by the two codebooks, but in image retrieval, where a query feature may have a number of ground truths matches, the harm caused to recall cannot be ignored. Therefore, it is proved in [139] that a SIFT-SIFT Multi-Index is inferior to a single codebook of the same size. The c-MI structure is also used in medical image retrieval [140]. Zheng *et al.* [141] demonstrate that when multiple codebooks are constructed, down-weighting the indexed features located within the intersection set of multiple candidate pools brings consistent benefit. In [142], Xia *et al.* introduce the joint inverted index by constructing several evenly distributed codebooks to improve the recall of ANN search.

Previous works on CNN features typically use compact vectors for the sake of memory and speed efficiency [9], [26], [29], [49]. Indexing such vectors is similar to the retrieval methods based on global representations, which has been a focus in hashing algorithms. We refer readers to a recent survey of hashing in [143]. On the other hand, the inverted index also has advantages in improving retrieval accuracy. To our knowledge, two works exploit the usage of the inverted index on CNN features. In [75], an inverted index

based on the SIFT feature is built, and binarized CNN signatures from regional and global patches are stored for matching strength estimation. In [144], CNN features from two-scale image patches are extracted and quantized by a medium-sized codebook, based on which an inverted index is constructed and refined with additionally stored global signatures. It should be noted that the inverted index has not been extensively studied in CNN-based image retrieval. We mention it as an important future work.

### 5.1.2 Hamming Embedding and its improvements

Hamming Embedding (HE) [12] proposed by Jégou *et al.* is a milestone work that improves the retrieval accuracy to a large extent. HE first maps a SIFT descriptor  $f \in \mathbb{R}^p$  from the  $p$ -dimensional space to a  $p_b$ -dimensional space:

$$x = P \cdot f = (x_1, \dots, x_{p_b}), \quad (11)$$

where  $P \in \mathbb{R}_b^p \times p$  is a projecting matrix, and  $x$  is a low-dimensional vector. By creating a matrix of random Gaussian values and applying a QR factorization to it, matrix  $P$  is taken as the first  $p_b$  rows of the resulting orthogonal matrix. To binarize  $x$ , Jégou *et al.* propose to compute the median vector  $\bar{x}_i = (\bar{x}_{1,i}, \dots, \bar{x}_{p_b,i})$  of the low-dimensional vector using descriptors falling in each Voronoi cell  $c_i$ . Given descriptor  $f$  and its projected vector  $x$ , Hamming Embedding computes the its visual word  $c_t$ , and the HE binary vector is computed as:

$$b_j(x) = \begin{cases} 1 & \text{if } x_j > \bar{x}_{j,t}, \\ 0 & \text{otherwise} \end{cases}, \quad (12)$$

where  $b(x) = (b_1(x), \dots, b_{p_b}(x))$  is the resulting HE vector of dimension  $p_b$ . The binary feature  $b(x)$  serves as a further check for matching strength. That is, when matching a pair of keypoints, a true match should satisfy two criteria: 1) identical visual words and 2) small Hamming distance between the Hamming Embeddings of their corresponding SIFT descriptors. A natural extension of HE [145] is to calculate the matching strength between feature  $f_1$  and  $f_2$  reversely to the Hamming distance in an exponential function:

$$w_{\text{HE}}(f_1, f_2) = \exp\left(-\frac{\mathcal{H}(b(x_1), b(x_2))}{2\gamma^2}\right), \quad (13)$$

where  $b(x_1)$  and  $b(x_2)$  are the HE binary vector of  $f_1$  and  $f_2$ , respectively,  $\mathcal{H}(\cdot, \cdot)$  computes the Hamming distance between two binary vectors, and  $\gamma$  is a weighting parameter. As shown in Fig. 6, Hamming Embedding [12] and its weighted version [145] improves retrieval performance considerably in 2008 and 2010.

Applications of HE include video copy detection [146], image classification [147], and re-ranking [148]. For example, Jain *et al.* [147] use HE to efficiently estimate patch matching similarity which is integrated in linear kernel-based SVM for image classification. In [148], Tolias *et al.* apply HE in image re-ranking. They use lower HE thresholds to find strict correspondences which resemble those found by RANSAC, and the resulting image subset is more likely to contain true positives for query re-formulation. For a detailed description of re-ranking methods, we refer readers to [149] for a comprehensive survey.



In a broad sense, improvement over HE has been observed in two aspects, *i.e.*, SIFT-based and heterogeneous feature-based. More effective HE formulations have been designed [111], [150]. For example, Jain *et al.* [16] propose an Asymmetric Hamming Embedding (AHE) method based on vector-to-binary distance comparison, in order to reduce the information loss on the query side. AHE works by exploiting the vector-to-hyperplane distances and retains the efficiency of the inverted index. Qin *et al.* [150] propose to adaptively estimate the local feature distance w.r.t the distance distribution of false matches, and in turn design a higher-order similarity measurement within a probabilistic framework. With a similar matching function, Tolias *et al.* combine the advantages of VLAD [113] and BoW, resulting in a selective match kernel that is later approximated to binary cases for large-scale usage. Tolias *et al.* also demonstrate that using more bits (*e.g.*, 128) in HE is superior to the original 64 bits scheme w.r.t the cost of efficiency. In Scalar Quantization [100], [117], Zhou *et al.* propose the use of the first 32 bits as the code word, and employ the rest of the bits of a 256-dim feature for binary verification. This method is efficient, but may be prone to relatively low recall.

The idea of SIFT Hamming Embedding is extended to incorporate multiple heterogeneous features, on the other hand. The idea is that when the matching strength between two keypoints is estimated, heterogeneous features provide complementary insights while retaining computational efficiency. The Bag-of-Colors [61] concatenates a binary color descriptor with SIFT HE to incorporate complementary information, similar to the coupled Multi-Index [35]. Zheng *et al.* [75] propose embedding the floating-point CNN features onto the binary space, and multi-scale matching verification is performed under a probabilistic framework. In [91], the binary signature of the spatial distribution around a keypoint is encoded and employed for spatial verification.

## 5.2 Feature Encodings

Feature encodings aggregate local descriptors into a single vector, which can be used in the subsequent linear scan (after dimension reduction or hashing for efficiency). The most popular feature encoding methods in image retrieval are the Vector of Locally Aggregated Descriptors (VLAD) [14] and the Fisher Vector (FV) [13], [151]. A comprehensive study of encoding methods can be accessed in [152]. Since both encodings are high-dimensional, Principle Component Analysis (PCA) is usually adapted to reduce the vectors to lower dimensions, and it has been shown that retrieval accuracy even increases after PCA [153].

The encoding of the Fisher Vector starts with training a GMM model using the expectation maximization (EM) algorithm [154], which can be viewed as a codebook with soft quantization. FV thus describes the averaged first and second order difference between local features and the GMM centers. The dimension of FC is  $2DK$ , where  $D$  is the dimension of the local descriptors, and  $K$  is the codebook size of GMM. The FV undergoes power normalisation (or signed square normalization) [13], [99] to suppress the burstiness problem. In this step, each component of FC undergoes non-linear transformation featured by parameter  $\alpha$  as,  $x_i := \text{sign}(x_i) \|x_i\|^\alpha$ . Then the L2 normalization is employed

Name	# images	# queries	Content
Holidays [12]	1,491	500	scene
Ukbench [10]	10,200	10,200	common objects
Paris6k [98]	6,412	55	buildings
Oxford5k [11]	5,062	55	buildings
Flickr100k [98]	99,782	-	from Flickr's popular tags

TABLE 3: Statistics of popular instance-level datasets

to produce FV. In [155], larger codebooks (up to 4,096) are generated and demonstrate superior classification accuracy to smaller codebooks, but at the cost of computational efficiency. In [156], Koniusz *et al.* propose the augmentation of each descriptor with its spatial coordinates and associated tunable weights. To correct the assumption that local regions are identically and independently distributed (iid), Cinbis *et al.* [157] propose non-iid models that discount the burstiness effect and yield improvement over the power normalisation. To further improve FV, Douze *et al.* [158] combine FV with the concatenation of the scores of attribute classifiers to integrate semantic information.

The VLAD encoding scheme proposed by Jégou *et al.* [14] can be thought of as a simplified version of the FV with several modifications. It considers the first order difference between the local features and the codebook centers learned by K-Means. The dimension of VLAD is  $DK$ . The PCA reduced vectors can be efficiently indexed and searched by Product Quantization (PQ) [106] which has been demonstrated to produce superior results to other popular methods such as FLANN [159]. A more detailed discussion of VLAD and PQ can be viewed in [160]. To improve this method, Tolias *et al.* [44] propose jointly encoding the descriptor angles during aggregation to achieve orientation covariance of the residual statistics. In [161], Arandjelovic *et al.* extend the VLAD representation in three aspects: 1) intra-normalization to suppress burstiness, 2) vocabulary adaptation to address the problem of dataset transfer, and 3) multi-VLAD for small object discovery. In [153], Jégou and Chum suggest the usage of PCA and whitening to de-correlate visual word co-occurrences, and propose the training of multiple codebooks to reduce quantization loss. Jégou *et al.* [107] further introduce triangulation embedding which considers only the direction and not the magnitude of the input vectors, and presents a democratic aggregation that limits the interference between the mapped vectors. The relationship between VLAD and the BoW framework is discussed in [111]. For a systematic comparison of various encoding methods, we refer readers to [162].

## 6 EXPERIMENTAL COMPARISONS

### 6.1 Image Retrieval Datasets

Five popular instance retrieval datasets are used in this survey. Statistics of these datasets can be accessed in Table 3.

**Holidays** [12] is collected by Jégou *et al.* from personal holiday albums, so most of the images are of various scene types. The database has 1491 images composed of 500 groups of similar images. Each image group has 1 query, totaling

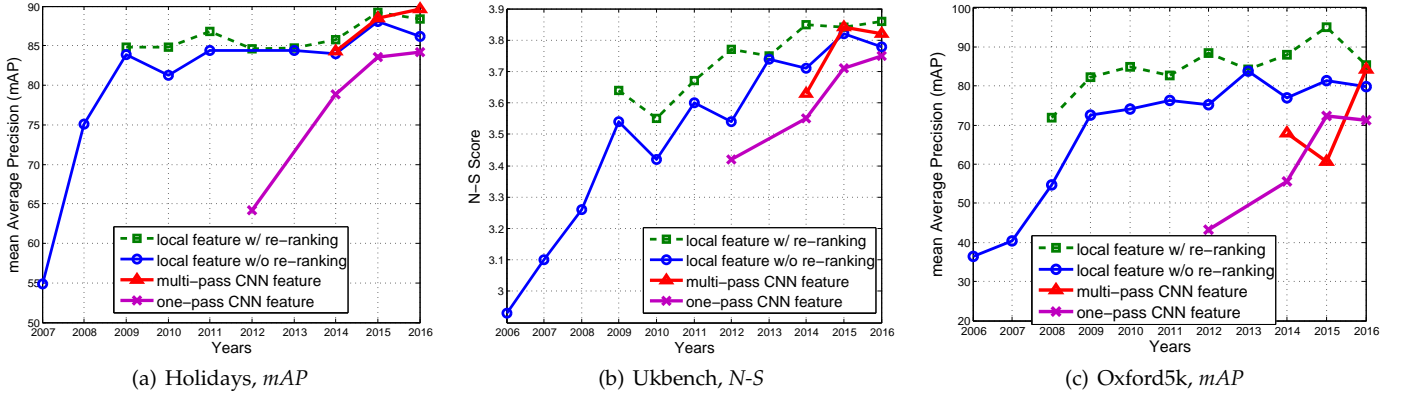


Fig. 6: The state of the art over the years on the (a) Holidays, (b) Ukbentch, and (c) Oxford5k datasets. Four types of works are summarized, *i.e.*, using local features with re-ranking, using local features without re-ranking, using multi-pass CNN features, and using one-pass CNN features. Research on these datasets basically started in 2006 or 2007. It is evident that retrieval accuracy of the CNN-based methods is swiftly catching up.

500 query images. Unlike Oxford5k and Paris6k, queries on Holidays are entire images instead of ROIs.

**Ukbentch** [10] consists of 10,200 images of various content, such as objects, scenes, and CD covers. All the images are divided into 2550 groups. Each group has 4 images depicting the same object/scene, under various angles, illuminations, translations, *etc.* Each image in this dataset is taken as the query in turn, so there are 10,200 queries.

**Oxford5k** [11] is collected by crawling images from Flickr using the names of 11 different landmarks in Oxford. A total of 5062 images form the image database. The dataset defines 5 queries for each landmark by hand-drawn bounding boxes, so that 55 query Regions of Interest (ROI) exist in total. Each database image is assigned one of four labels, *good*, *OK*, *junk*, or *bad*. The first two labels are true matches to the query ROIs, while “bad” denotes the distractors. In junk images, less than 25% of the objects are visible, or they undergo severe occlusion or distortion, so these images have zero impact on retrieval accuracy.

**Flickr100k** [114] contains 99,782 high resolution images crawled from Flickr’s 145 most popular tags. In literature, this dataset is typically added to Oxford5k to test the scalability of retrieval algorithms.

**Paris6k** [114] dataset is created in conjunction with the Oxford5k dataset. It featured 6412 images crawled from 11 queries on specific Paris architecture. Like Oxford5k, each landmark has 5 queries, so there are also 55 queries for this dataset. The database images are annotated with the same four types of labels as the Oxford5k dataset.

## 6.2 Evaluation Metrics

**Precision-Recall.** According to traditional definition, recall denotes the ratio of returned true matches to the total number or true matches in the database, while precision refers to the fraction of true matches in the returned images. For example, when considering a subset of  $n$  images as the returned images, assuming there are  $n_p$  true matches among them, and a total of  $N_p$  true matches exist in the whole database, then  $\text{recall}@n$  ( $r@n$ ) and  $\text{precision}@n$  ( $p@n$ ) are calculated as  $\frac{n_p}{N_p}$  and  $\frac{n_p}{n}$ , respectively. A perfect image

retrieval system should be able to locate all the true matches to a query in the database, and return them in the top ranks in the rank list, *i.e.*, 100% recall and precision. In image retrieval, given a query image and its rank list, a precision-recall curve can be drawn on the (precision, recall) points  $(r@1, p@1)$ ,  $(r@2, p@2)$ , ...,  $(r@N, p@N)$ , where  $N$  is the number of images in the database.

**Average Precision and Mean Average Precision.** When measuring the retrieval performance of a single query, the precision-recall curve will suffice: a higher curve means superior retrieval performance. To more clearly record the retrieval performance, Average Precision (AP) is used, which amounts to the area under the precision-recall curve. Typically, a larger AP means a higher precision-recall curve, and thus better retrieval performance. Since image retrieval datasets typically have multiple query images, their respective APs are averaged to produce a final retrieval performance evaluation, *i.e.*, the mean Average Precision (mAP), the value of which varies from 0 to 1. Conventionally, we use mAP to evaluate retrieval accuracy on the Oxford5k, Paris6k, and Holidays datasets.

**N-S Score.** The N-S score is specifically used on the Ukbentch dataset and is named after David Nistér and Henrik Stewénius [10]. It is equivalent to  $\text{precision}@4$  or  $\text{recall}@4$  because every query in Ukbentch has 4 true matches in the database. The N-S score is calculated as the average number of true matches in the top-4 ranks across all the rank lists, and thus the value of N-S score ranges from 0 to 4.

## 6.3 Comparison and Analysis

We present the improvement in retrieval accuracy over the past 10 years in Fig. 6 and Table 4. The numbers are computed using codebooks trained on independent datasets [12]. Several observations can be drawn. First, it is evident that the field of instance retrieval has been constantly improving. For example, the baseline approach (HKM) only yields a retrieval accuracy of 59.7%, 2.85, 44.3%, 26.6%, and 46.50% on Holidays, Ukbentch, Oxford5k, Oxford5k+Flickr100k, and Paris, respectively. Starting from these baselines, a noticeable improvement can be seen in the years 2008-2010 with the

Method Type	Methods	Voc. Size/Dim.	Holidays	Ukbench	Oxford5k	+100k	Paris6k	Mem./Img (bytes)
SIFT-based	HKM [10]	1M	59.7	2.85	44.3 <sup>†</sup>	26.6 <sup>†</sup>	46.5 <sup>†</sup>	3.5×1,000*
	AKM [11]	1M	64.1 <sup>†</sup>	3.02 <sup>†</sup>	49.3	34.3	50.2 <sup>†</sup>	3.5×1,000
		20k	46.9	2.88	33.8	22 <sup>§</sup>	35.6 <sup>†</sup>	3.5×1,000
	HE [12]	200k	77.5	3.38	50.7	37 <sup>§</sup>	49.0 <sup>†</sup>	10.5×1,000
		20k	73.5	3.42	51.7	39 <sup>§</sup>	49.9 <sup>†</sup>	10.5×1,000
	Fine Voc. [105]	16M	75.8**	-	84.9	79.5	82.4	3.5×1,000
	Burst [132]	20k	83.9	3.54	64.7	49 <sup>§</sup>	62.8 <sup>†</sup>	11.5×1,000
	Three Things [42]	1M	-	-	80.9	72.2	76.5	3.5×1,000
	ASMK [163]	65k	81.0	-	80.4	75.0	77.0	18.5×0.8×1,000
	c-MI [35]	20k×200	84.0	3.71	58.2 <sup>†</sup>	35.2 <sup>†</sup>	55.1 <sup>†</sup>	14.25×1,000
	Co-index [67]	1M	81.2	3.74	72.7	-	-	7.42×1,000
	Q.ada [150]	1M	78.0	-	82.1	72.8	73.6	36×1,000
	VLAD [14]	64/4,096	55.6	3.18 <sup>†</sup>	37.8	27.2 <sup>†</sup>	38.6 <sup>†</sup>	16k
	FV [99]	64/4,096	59.5	3.09 <sup>†</sup>	41.8	33.1 <sup>†</sup>	43.0	16k
	Triangular [107]	64/8,064	77.1	3.56 <sup>†</sup>	67.6	61.1	-	31.5k
Multi-pass CNN	Off the Shelf [9]	-/4k	84.3	3.64	68.0	-	79.5	16k
	MOP [26]	100/2,048	80.2	-	-	-	-	8k
	CKN [102]	256/65,536	79.3	3.76	56.5	-	-	256k
	OLDFP [54]	-/512	88.5	3.81	60.7	-	66.2	2k
	MSS [95]	-/256	71.6	3.37	53.3	48.9	67.0	1k
One-pass CNN	BLCF [96]	25k	-	-	78.8	65.1	84.8	3.5×14×14
	Neural Codes [49]	-/4,096	79.3	3.29	54.5	51.2	-	4k
	SPoC [27]	-/256	78.4	3.66	65.7	64.2	-	1k
	MAC [29]	-/256	76.7	-	58.3 <sup>‡</sup>	49.2 <sup>‡</sup>	72.6 <sup>‡</sup>	1k
	Siamese [50]	-/512	82.5	-	80.1 <sup>‡</sup>	74.1 <sup>‡</sup>	85.0 <sup>‡</sup>	2k
	NetVLAD [164]	64/2048	82.8	-	70.8	-	78.3	8k

TABLE 4: Performance Summarization of Representative Methods on Three Datasets. The addition of Flickr100k into Oxford5k is denoted as “+100k”. \* we assume there are 1,000 key-points per image. \*\* manually rotate images in Holidays to be upright. <sup>†</sup> numbers are reported by our own implementation. <sup>‡</sup> keep all activations falling into the bounding box [27]. <sup>§</sup> numbers are estimated from the curves.

introduction of Hamming Embedding [12], [145]. From then on, major improvements come from the strength of feature fusion [35], [75], [165], such as color and FC features. The benefit of re-ranking methods is also clear from Fig. 6. Since this paper mainly focuses on the BoW model, we refer readers to [149] for a survey on re-ranking models.

Second, CNN-based retrieval models have quickly demonstrated their strengths in instance retrieval. In the year 2012 when the AlexNet [8] was introduced, the performance of the off-the-shelf FC features is still far from satisfactory compared with SIFT models during the same period. For example, the use of ImageNet pre-trained CNN feature yields 64.2%, 3.42, and 43.3% in mAP, N-S score, and mAP, respectively, on the Holidays, Ukbench, and Oxford5k datasets. These numbers are lower than Graph Fusion [64] by 20.64%, 0.12 on Holidays and Ukbench, and lower than Spatially-Constrained Similarity Measure (SCSM) by 31.90% on Oxford5k, respectively. However, with the advance in feature encoding and CNN architectures, the performance of CNN features is improving fast. On the benchmark datasets, there is a clear trend that the multi-pass CNN models [166], [167] are reporting competitive accuracy.

Third, on the benchmark datasets, we observe that the multi-pass CNN-based and the SIFT-based methods generally have superior performance to the one-pass CNN approaches. The former two models, as mentioned in literature reviews, are similar in nature, and the CNN features are usually more discriminative in visual matching [53]. Since the multi-pass CNN model extracts more visual information

from an image (at cost of more time for feature extraction), and more sophisticated indexing and pooling methods can be employed, the multi-pass CNN model outperforms the one-pass model. That said, the one-pass CNN model still has good potential in future improvement especially in geometry integration and heterogeneous feature fusion.

Finally, while competitive accuracy is achieved by CNN-based models, their memory consumption is on par with the SIFT-based methods (*e.g.*, use several kbytes per image). This is because Table 1 and Fig. 6 does not include results after dimension reduction or binary code generation. It is generally considered that using short codes is beneficial towards very large galleries, at a cost of accuracy decrease.

## 7 FUTURE RESEARCH DIRECTIONS

We can see from the above review that the SIFT-based retrieval model has been extensively studied in the past ten years. Nevertheless, its popularity currently seems to have been overtaken by the Convolutional Neural Network (CNN) features. Fortunately, it has been found that both the SIFT and CNN features are analogous in nature and the Bag-of-Words (BoW) model can accommodate both. Lessons learned from SIFT will create a number of new research directions for the CNN-based retrieval framework. In the following, we will describe these open issues.

### 7.1 CNN Feature Extraction

Learning a discriminative CNN model for instance retrieval is the key to an effective CNN-based retrieval system. A



local feature, as mentioned above, consists of detector and descriptor, both of which are implicitly contained in the available CNN network (especially for the one-pass CNN retrieval model). Nevertheless, there is a lack of large-scale instance-level datasets in image retrieval, and currently available datasets such as ImageNet [25] only provide images with class labels. This is problematic for instance retrieval because it retrieves images of the same object/scene rather than the same class. Babenko *et al.* [49] collect the Landmark dataset by text queries in Flickr before the disposal of irrelevant images, and Radenovic *et al.* [50] propose an unsupervised method of constructing 3D landmark models from which the truth matches can be grouped. Both collected datasets focus on landmarks, so the generalization ability is open to doubt. In fact, Babenko *et al.* [49] report that fine-tuning a CNN model on the Landmark dataset compromises retrieval accuracy on Ukbench which consists of common objects instead of buildings. Training an effective CNN model that works on generic image retrieval is therefore especially challenging due to the lack of training data.

It is clear from the above analysis that the image retrieval community is in great need of a large-scale instance-level dataset, or efficient methods for generating such a dataset in either a supervised or unsupervised manner. For different retrieval tasks, such as landmark, or logos, specific training sets can be collected as is done in [49], [50]. Research can be conducted on understanding which factors in a re-trained network are important for instance-level image retrieval. Novel insights are needed on how filters help instance matching, and in which aspects these filters resemble the SIFT detector.

An even more challenging retrieval problem is finding a small query object in an large image [168]. The CNN feature should have robust invariance to large scale differences. It would be beneficial to construct datasets in which objects vary extensively in scale, so that the network does not only focus on the global characteristics, but places more emphasis on the image details. This problem requires more discriminative local pattern detection based on CNN filters, which should fire at small objects with high response. The discriminative aggregation of local features, which tend to preserve the filter response to such local patterns, is of great value.

Previous works employ standard classification [49], Siamese-loss [50], or Triplet-loss [169] CNN models for fine-tuning. It is worth designing novel networks for the retrieval task. For example, our preliminary experiment taking advantage of both the pairwise loss and the classification loss has shown very promising results. For specific tasks in which objects are well-structured such as pedestrian retrieval, sequence models such as the Recurrent Neural Networks can be employed to pool the local (regional) descriptors.

It would be also interesting to investigate the performance of CNN descriptors extracted from hand-crafted detectors, *e.g.*, DoG (Difference of Gaussians) [15], Hessian Affine [36], or from region proposals *e.g.*, selective search [55], edgebox [170], *etc.* In fact, it remains unknown whether these off-the-shelf are optimal for instance retrieval, because they typically have competitive results on object localization under certain (20) classes. Salvador *et al.* [60] demonstrate that fine-tuning Faster R-CNN using the query instances

substantially improves retrieval performance, but it is not feasible to fine-tune a CNN model every time a query arises. It is therefore critical that new local detectors that are effective for generic object/scene discovery be proposed and evaluated in combination with CNN descriptors.

## 7.2 Geometric Constraints in CNN Activations

Imposing geometric constraints have been proven to be an effective strategy for improving matching accuracy over the orderless SIFT-based BoW histogram. For the one-pass CNN retrieval model, the 2-D distribution (feature map) of a certain filter on the input image is well structured, and the stack of the feature maps of all filters in a layer resembles the SIFT features extracted from densely sampled patches. The difference is that each dimension of the CNN local descriptor has an explicit meaning: the activation strength of the filter on the corresponding region. While pooling methods such as VLAD or FV achieve considerable invariance to spatial misalignment, such methods do not provide more precise spatial codes. This survey suggests two possible solutions to this problem. First, after the quantization of local CNN descriptors, it is feasible to explicitly model the visual word co-occurrence by encoding the spatial context [90], [91], or generating higher order visual elements like the visual phrase [82], [83]. Although the resulting aggregated feature vector may have a higher dimension, its discriminative ability can be largely improved. Second, since each dimension of the CNN local descriptor has its semantic representation, an alternative is to discover spatial constraints without codebook quantization. For example, simple thresholding can then be employed to spot the fire position and its amplitude, and CNN activations from multiple layers can be integrated from which pattern combinations can be discovered. This is distinct from the SIFT feature because the dimensions of SIFT do not have explicit semantic meanings, so the discovered patterns can hardly have spatial discriminability.

When considering the multi-pass CNN model, in which CNN features (either the pooled intermediate layers or the fully connected layer) are extracted from hand-crafted (or learned) image patches, the resulting bag-of-words model is also accessible to spatial regularization. It is promising to learn from previous experiences using the SIFT feature [12], [79], [81], [82]. Both strict or loose spatial constraints can be utilized to improve the matching accuracy among CNN visual words. Since the current literature largely neglects this aspect, it remains a worthy task for future research.

## 7.3 Feature Fusion with Complementary Features

The CNN features exhibit an increasing level of semantics along the hierarchy, while hand-crafted features such as SIFT, color histogram, and GIST, focus on certain aspects of an image. While CNN achieves good performance on image retrieval benchmarks, it might still fail in many cases where severe truncation and occlusion occur; but SIFT may be well suited to facing such image distortions. Take advantage of the complementary nature of these features is a promising prospect. On the fusion of global vectors, Zheng *et al.* [69] show that by adaptively assigning larger weights to good features and lower weights to bad features, the complementarity of heterogeneous features can be effectively

exploited. Douze *et al.* [158] also demonstrate the benefits of concatenating FV and image attributes. Therefore, in the field of global feature fusion, it is important that the CNN codes can be combined with the previously hand-crafted state of the art, considering the popularity of encoding methods such as VLAD and FV.

Another possible solution concerns local feature fusion. In the one-pass CNN model, since each filter covers a local region of an image, it is feasible that complementary coupled features are also extracted from the same region. This is essentially similar to the fusion schemes between SIFT and other local features such as color [61], [69]. Early fusion concatenates heterogeneous features at an early stage following their extraction, and the BoW can be subsequently constructed. Another option is to build an inverted index through CNN, and store binary embeddings of the features to be fused in the index. The binary codes are in turn used for matching verification. Since the inverted index may incur the problem of memory and time inefficiency, it is important that the inserted information be small and efficient for computation. Nevertheless, a concurrent problem of feature fusion in image retrieval is how to optimally determine the contribution of each feature. Given a query image, it is not known in advance what the query is about, so it is not trivial to pre-define weights for individual features. This problem can preferably be addressed by query difficulty estimation [171], [172]. In addition, another guideline for feature fusion is that offline steps should be independent of the database, allowing for dynamic database update.

The situation when extracting CNN features from image patches is similar to the SIFT case. In the Bag-of-Boundary method [63], a number of features are extracted from boundary fragments and subsequently concatenated before quantization. Multiple features can be extracted from the image patches with CNN. Dimension reduction may be of vital importance for preserving the discriminative ability of individual features while keeping it sufficiently low dimensional to fit in the BoW model. Apart from region proposals, it is unknown how semantic segmentation or superpixels contribute to patch descriptions.

## 7.4 CNN Quantization

It has been mentioned above that CNN quantization has not yet been extensively studied; instead, a number of works have been introduced for SIFT quantization in both retrieval and classification. When generating FV or VLAD from CNN local descriptors, it is feasible to adopt more sophisticated methods to reduce quantization error, such as sparse coding [173] or Locality-constrained Linear Coding [174], *etc.* For multi-pass CNN features, the major difference between SIFT and CNN is that the latter is typically high-dimensional. This is problematic for quantization, because the codebook may not be large enough to provide a discriminative partitioning of the high-dimensional feature space. One solution is to construct large codebooks after PCA, as is done in a number of image retrieval works based on SIFT [10], [11], [67], [105]; another strategy is to use hashing techniques for efficient quantization. The two solutions do not contradict each other, and can be combined for usage. When training a large codebook, approximate K-Means methods are good

options for efficient convergence. Similar approximate Nearest Neighbor search algorithms can then be used for feature quantization. On the use of Hashing methods, Zhou *et al.* propose a codebook-free scalar quantization method, which is beneficial towards large codebooks without extensive off-line training or inaccurate quantization. Its disadvantage is the relatively low matching recall incurred by long binary vectors, so a promising direction is to design short binary codes that ensure high recall to be used for codebook generation and quantization.

Another aspect that makes CNN quantization distinct from SIFT is that each dimension of the column CNN descriptor has explicit low/high-level semantic meanings. It might therefore not be optimal to put the entire vector in codebook clustering and quantization. Instead, quantization within each activation map may be an effective strategy. The quantization results in this case can be concatenated to form the final representation.

## 7.5 CNN and the Inverted Index

The inverted index is typically used when codebook size is large and the BoW histogram is sparse, so that considerable memory cost can be reduced. To our knowledge, current research on CNN based retrieval rarely generates large codebooks [96] possibly in concern of memory cost and system complexity. Nevertheless, previous success in SIFT based image retrieval still suggests promising performance of this classic data structure. For example, both one-dimensional and multi-dimensional inverted index [35], [101] can be built with CNN to improve matching accuracy. It is also feasible to index binary signatures ranging from spatial context, complementary features, to semantic embeddings designed for CNN features.

## 8 CONCLUDING REMARKS

The BoW model has been extensively studied in instance retrieval over the last 13 years. It was not until 2012 that SIFT, due to its advantage in dealing with image transformation, was employed as the *de facto* visual feature, and a myriad works have since been proposed to improve its discriminative power in the field of instance retrieval. Recent years have witnessed the rise of the Convolutional Neural Network feature. In this survey, we classify previous methods into three types according to the feature extraction and quantization process, *i.e.*, SIFT-based, one-pass CNN-based, and multi-pass CNN-based. We demonstrate that the two features have common properties, and can be interpreted in the framework of BoW (Fig. 2). A comprehensive survey of the basic components of BoW, *i.e.*, features, quantization, and data structures for efficiency, has been conducted in this paper *w.r.t* the use of both the SIFT and CNN features.

From the collected experimental results on five benchmark datasets, it is observed that the state of the art has consistently been pushed forward with both features. The retrieval accuracy of the SIFT feature is at a high level, but the performance of the CNN feature is swiftly catching up, thanks to learning from experiences with SIFT. Given the overwhelming performance of CNN in image recognition, detection, segmentation, *etc.*, the research focus of image

retrieval has now turned to the CNN-based model. At this time of change, it is beneficial to provide an on-the-run summary and highlight the lessons learned from previous experience.

As we have pointed out, important research directions still exist for the CNN-based model. For example, an important direction concerns the collection of large-scale instance level datasets, and learning a generic CNN model for generic retrieval problems that is especially tuned towards small object discovery. Another topic that is under-exploited is to encode geometric constraints among CNN activations, which is extensively studied in SIFT-based retrieval models. Due to the limitation of CNN features *w.r.t* severe image transformation, it is practical to fuse complementary features on both global and local levels with the CNN representation. Since current CNN methods tend to use small codebooks, discriminative codebook learning and quantization methods are possible topics. Also, large vocabularies coupled with the inverted index and various binary embeddings can be explored to improve retrieval accuracy, although this may result in memory overload, though. While various feature coding schemes are proposed in both retrieval and classification based on SIFT, the introduction of novel encoding methods and improvement for the CNN model should be encouraged. As the last remark, although SIFT is losing its popularity in the retrieval community, we should not completely forget it. We speculate that some combination of CNN and SIFT may still produce a successful image retrieval system.

## ACKNOWLEDGMENTS

The authors would like to thank the pioneer researchers in image retrieval and other related fields.

## REFERENCES

- [1] H. Tamura and N. Yokoya, "Image database systems: A survey," *Pattern Recognition*, vol. 17, no. 1, pp. 29–43, 1984.
- [2] S.-K. Chang and A. Hsu, "Image information systems: where do we go from here?" *IEEE transactions on Knowledge and Data Engineering*, vol. 4, no. 5, pp. 431–442, 1992.
- [3] B. Siddiquie, R. S. Feris, and L. S. Davis, "Image ranking and retrieval based on multi-attribute queries," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 801–808.
- [4] L. Torresani, M. Szummer, and A. Fitzgibbon, "Efficient object category recognition using classemes," in *European conference on computer vision*, 2010, pp. 776–789.
- [5] A. W. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1349–1380, 2000.
- [6] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *Proceedings of the IEEE International Conference on Computer Vision*, 2003, pp. 1470–1477.
- [7] L. Fei-Fei and P. Perona, "A bayesian hierarchical model for learning natural scene categories," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005, pp. 524–531.
- [8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.
- [9] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "Cnn features off-the-shelf: an astounding baseline for recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2014, pp. 806–813.
- [10] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2006, pp. 2161–2168.
- [11] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.
- [12] H. Jegou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *European conference on computer vision*, 2008, pp. 304–317.
- [13] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *European Conference on Computer Vision*, 2010, pp. 143–156.
- [14] H. Jegou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 3304–3311.
- [15] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [16] M. Jain, H. Jegou, and P. Gros, "Asymmetric hamming embedding: taking the best of our bits for large scale image search," in *ACM Multimedia*, 2011, pp. 1441–1444.
- [17] C.-Z. Zhu, H. Jegou, and S. Satoh, "Query-adaptive asymmetrical dissimilarities for visual object retrieval," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1705–1712.
- [18] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *European Conference on Computer Vision Workshops*, 2004, pp. 1–2.
- [19] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman, "Discovering object categories in image collections," *Technical Report A. I. Memo 2005-005, Massachusetts Institute of Technology*, 2005.
- [20] Y. Yang, F. Nie, D. Xu, J. Luo, Y. Zhuang, and Y. Pan, "A multimedia retrieval framework based on semi-supervised ranking and relevance feedback," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 4, pp. 723–742, 2012.
- [21] Y. Cai, L. Yang, W. Ping, F. Wang, T. Mei, X.-S. Hua, and S. Li, "Million-scale near-duplicate video retrieval system," in *ACM Multimedia*, 2011, pp. 837–838.
- [22] Y. Yang, Y.-T. Zhuang, F. Wu, and Y.-H. Pan, "Harmonizing hierarchical manifolds for multimedia document semantics understanding and cross-media retrieval," *IEEE Transactions on Multimedia*, vol. 10, no. 3, pp. 437–446, 2008.
- [23] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 580–587.
- [24] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.
- [25] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.
- [26] Y. Gong, L. Wang, R. Guo, and S. Lazebnik, "Multi-scale orderless pooling of deep convolutional activation features," in *European Conference on Computer Vision*, 2014, pp. 392–407.
- [27] A. Babenko and V. Lempitsky, "Aggregating local deep features for image retrieval," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1269–1277.
- [28] Y. Kalantidis, C. Mellina, and S. Osindero, "Cross-dimensional weighting for aggregated deep convolutional features," *arXiv:1512.04065*, 2015.
- [29] G. Tolias, R. Sivic, and H. Jegou, "Particular object retrieval with integral max-pooling of cnn activations," *arXiv:1511.05879*, 2015.
- [30] Y. Li, S. Wang, Q. Tian, and X. Ding, "A survey of recent advances in visual feature detection," *Neurocomputing*, vol. 149, pp. 736–751, 2015.
- [31] T. Tuytelaars and K. Mikolajczyk, "Local invariant feature detectors: a survey," *Foundations and Trends in Computer Graphics and Vision*, vol. 3, no. 3, pp. 177–280, 2008.
- [32] A. S. Mian, M. Bennamoun, and R. Owens, "Keypoint detection and local feature matching for textured 3d face recognition," *International Journal of Computer Vision*, vol. 79, no. 1, pp. 1–12, 2008.
- [33] T. Ge, Q. Ke, and J. Sun, "Sparse-coded features for image retrieval," in *British Machine Vision Conference*, 2013.



- [34] Z. Wu, Q. Ke, M. Isard, and J. Sun, "Bundling features for large scale partial-duplicate web image search," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 25–32.
- [35] L. Zheng, S. Wang, and Q. Tian, "Coupled binary embedding for large-scale image retrieval," *IEEE Transactions on Image Processing*, vol. 23, no. 8, pp. 3368–3380, 2014.
- [36] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool, "A comparison of affine region detectors," *International journal of computer vision*, vol. 65, no. 1-2, pp. 43–72, 2005.
- [37] K. Simonyan, A. Vedaldi, and A. Zisserman, "Learning local feature descriptors using convex optimisation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 8, pp. 1573–1585, 2014.
- [38] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005.
- [39] L. Van Gool, T. Moons, and D. Ungureanu, "Affine/photometric invariants for planar intensity patterns," in *European Conference on Computer Vision*, 1996, pp. 642–651.
- [40] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 509–522, 2002.
- [41] W. T. Freeman and E. H. Adelson, "The design and use of steerable filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 9, pp. 891–906, 1991.
- [42] R. Arandjelović and A. Zisserman, "Three things everyone should know to improve object retrieval," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2911–2918.
- [43] E. Tola, V. Lepetit, and P. Fua, "A fast local descriptor for dense matching," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.
- [44] G. Tolias, T. Furon, and H. Jégou, "Orientation covariant aggregation of local descriptors with embeddings," in *European Conference on Computer Vision*, 2014, pp. 382–397.
- [45] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *European Conference on Computer Vision*, 2014, pp. 818–833.
- [46] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deepconvolutional activation feature for generic visual recognition," *arXiv:1310.1531*, 2013.
- [47] J. Ng, F. Yang, and L. Davis, "Exploiting local features from deep networks for image retrieval," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015, pp. 53–61.
- [48] L. Zheng, Y. Zhao, S. Wang, J. Wang, and Q. Tian, "Good practice in cnn feature transfer," *arXiv:1604.00133*, 2016.
- [49] A. Babenko, A. Slesarev, A. Chigorin, and V. Lempitsky, "Neural codes for image retrieval," in *European Conference on Computer Vision*, 2014, pp. 584–599.
- [50] F. Radenović, G. Tolias, and O. Chum, "Cnn image retrieval learns from bow: Unsupervised fine-tuning with hard examples," *arXiv:1604.02426*, 2016.
- [51] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv:1409.1556*, 2014.
- [52] S. Zagoruyko and N. Komodakis, "Learning to compare image patches via convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 4353–4361.
- [53] P. Fischer, A. Dosovitskiy, and T. Brox, "Descriptor matching with convolutional neural networks: a comparison to sift," *arXiv:1405.5769*, 2014.
- [54] K. Mopuri and R. Babu, "Object level deep feature pooling for compact image representation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015, pp. 62–70.
- [55] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders, "Selective search for object recognition," *International journal of computer vision*, vol. 104, no. 2, pp. 154–171, 2013.
- [56] T. Uricchio, M. Bertini, L. Seidenari, and A. Bimbo, "Fisher encoded convolutional bag-of-windows for efficient image retrieval and social image tagging," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 9–15.
- [57] S. D. Bhattacharjee, J. Yuan, Y.-P. Tan, and L.-Y. Duan, "Query-adaptive small object search using object proposals and shape-aware descriptors," *IEEE Transactions on Multimedia*, vol. 18, no. 4, pp. 726–737, 2016.
- [58] S. Das Bhattacharjee, J. Yuan, Y.-P. Tan, and L. Duan, "Query-adaptive logo search using shape-aware descriptors," in *ACM Multimedia*, 2015, pp. 1155–1158.
- [59] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, 2015, pp. 91–99.
- [60] A. Salvador, X. Giró-i Nieto, F. Marqués, and S. Satoh, "Faster r-cnn features for instance search," *arXiv:1604.08893*, 2016.
- [61] C. Wengert, M. Douze, and H. Jégou, "Bag-of-colors for improved image search," in *ACM Multimedia*, 2011, pp. 1437–1440.
- [62] F. S. Khan, R. M. Anwer, J. van de Weijer, A. D. Bagdanov, M. Vanrell, and A. M. Lopez, "Color attributes for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3306–3313.
- [63] R. Arandjelović and A. Zisserman, "Smooth object retrieval using a bag of boundaries," in *International Conference on Computer Vision*, 2011, pp. 375–382.
- [64] S. Zhang, M. Yang, T. Cour, K. Yu, and D. N. Metaxas, "Query specific fusion for image retrieval," in *European Conference on Computer Vision*, 2012, pp. 660–673.
- [65] C. Deng, R. Ji, W. Liu, D. Tao, and X. Gao, "Visual reranking through weakly supervised multi-graph learning," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 2600–2607.
- [66] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International journal of computer vision*, vol. 42, no. 3, pp. 145–175, 2001.
- [67] S. Zhang, M. Yang, X. Wang, Y. Lin, and Q. Tian, "Semantic-aware co-indexing for image retrieval," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1673–1680.
- [68] R. Tao, A. W. Smeulders, and S.-F. Chang, "Attributes and categories for generic instance search from one example," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 177–186.
- [69] L. Zheng, S. Wang, L. Tian, F. He, Z. Liu, and Q. Tian, "Query-adaptive late fusion for image search and person re-identification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1741–1750.
- [70] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Hypercolumns for object segmentation and fine-grained localization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 447–456.
- [71] D. Yoo, S. Park, J.-Y. Lee, and I. Kweon, "Multi-scale pyramid pooling for deep convolutional representation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015, pp. 71–80.
- [72] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, "Learning hierarchical features for scene labeling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1915–1929, 2013.
- [73] Y. LeCun, F. J. Huang, and L. Bottou, "Learning methods for generic object recognition with invariance to pose and lighting," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2004, pp. 90–97.
- [74] V. Chandrasekhar, J. Lin, O. Morère, H. Goh, and A. Veillard, "A practical guide to cnns and fisher vectors for image instance retrieval," *arXiv:1508.02496*, 2015.
- [75] L. Zheng, S. Wang, J. Wang, and Q. Tian, "Accurate image search with multi-scale contextual evidences," *International Journal of Computer Vision*, pp. 1–13, 2016.
- [76] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2006, pp. 2169–2178.
- [77] K. Grauman and T. Darrell, "The pyramid match kernel: Discriminative classification with sets of image features," in *Proceedings of the IEEE International Conference on Computer Vision*, 2005, pp. 1458–1465.
- [78] O. Chum and J. Matas, "Matching with prosac-progressive sample consensus," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005, pp. 220–226.
- [79] Q.-F. Zheng, W.-Q. Wang, and W. Gao, "Effective and efficient object-based image retrieval using visual phrases," in *ACM Multimedia*, 2006, pp. 77–80.

- [80] M. A. Sadeghi and A. Farhadi, "Recognition using visual phrases," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 1745–1752.
- [81] Y. Jiang, J. Meng, and J. Yuan, "Randomized visual phrases for object search," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3100–3107.
- [82] Y. Zhang, Z. Jia, and T. Chen, "Image retrieval with geometry-preserving visual phrases," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 809–816.
- [83] S. Zhang, Q. Tian, G. Hua, Q. Huang, and S. Li, "Descriptive visual words and visual phrases for image applications," in *ACM Multimedia*, 2009, pp. 75–84.
- [84] L. Torresani, M. Szummer, and A. Fitzgibbon, "Learning query-dependent prefilters for scalable image retrieval," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 2615–2622.
- [85] J. Yuan, Y. Wu, and M. Yang, "Discovery of collocation patterns: from visual words to visual phrases," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.
- [86] C. L. Zitnick, J. Sun, R. Szeliski, and S. Winder, "Object instance recognition using triplets of feature symbols," *Technical Report MSR-TR-200753*, Microsoft Research, 2007.
- [87] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide-baseline stereo from maximally stable extremal regions," *Image and vision computing*, vol. 22, no. 10, pp. 761–767, 2004.
- [88] Q. Hao, R. Cai, Z. Li, L. Zhang, Y. Pang, and F. Wu, "3d visual phrases for landmark recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3594–3601.
- [89] L. Zheng and S. Wang, "Visual phraselet: Refining spatial constraints for large scale image search," *IEEE Signal Processing Letters*, vol. 20, no. 4, pp. 391–394, 2013.
- [90] W. Zhou, Y. Lu, H. Li, Y. Song, and Q. Tian, "Spatial coding for large scale partial-duplicate web image search," in *ACM Multimedia*, 2010, pp. 511–520.
- [91] Z. Liu, H. Li, W. Zhou, and Q. Tian, "Embedding spatial context information into inverted file for large-scale image retrieval," in *ACM Multimedia*, 2012, pp. 199–208.
- [92] X. Shen, Z. Lin, J. Brandt, S. Avidan, and Y. Wu, "Object retrieval and localization with spatially-constrained similarity measure and k-nn re-ranking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3013–3020.
- [93] C. H. Lampert, "Detecting objects in large image collections and videos by efficient subimage retrieval," in *Proceedings of the IEEE International Conference on Computer Vision*, 2009, pp. 987–994.
- [94] X. Li, M. Larson, and A. Hanjalic, "Pairwise geometric matching for large-scale object retrieval," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5153–5161.
- [95] A. S. Razavian, J. Sullivan, A. Maki, and S. Carlsson, "A baseline for visual instance retrieval with deep convolutional networks," *arXiv:1412.6574*, 2014.
- [96] E. Mohedano, A. Salvador, K. McGuinness, F. Marques, N. E. O'Connor, and X. Giró-i Nieto, "Bags of local convolutional features for scalable instance search," *arXiv:1604.04653*, 2016.
- [97] Z. Wu, Q. Ke, M. Isard, and J. Sun, "Bundling features for large scale partial-duplicate web image search," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 25–32.
- [98] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Lost in quantization: Improving particular object retrieval in large scale image databases," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.
- [99] F. Perronnin, Y. Liu, J. Sánchez, and H. Poirier, "Large-scale image retrieval with compressed fisher vectors," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 3384–3391.
- [100] W. Zhou, Y. Lu, H. Li, and Q. Tian, "Scalar quantization for large scale image search," in *ACM Multimedia*, 2012, pp. 169–178.
- [101] A. Babenko and V. Lempitsky, "The inverted multi-index," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 6, pp. 1247–1260, 2015.
- [102] M. Paulin, M. Douze, Z. Harchaoui, J. Mairal, F. Perronin, and C. Schmid, "Local convolutional features with unsupervised training for image retrieval," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 91–99.
- [103] Y. Gong and S. Lazebnik, "Iterative quantization: A procrustean approach to learning binary codes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 817–824.
- [104] S. Lloyd, "Least squares quantization in pcm," *IEEE transactions on information theory*, vol. 28, no. 2, pp. 129–137, 1982.
- [105] A. Mikulík, M. Perdoch, O. Chum, and J. Matas, "Learning a fine vocabulary," in *European Conference on Computer Vision*, 2010, pp. 1–14.
- [106] H. Jégou, M. Douze, and C. Schmid, "Product quantization for nearest neighbor search," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 1, pp. 117–128, 2011.
- [107] H. Jégou and A. Zisserman, "Triangulation embedding and democratic aggregation for image search," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3310–3317.
- [108] L. Zheng, S. Wang, Z. Liu, and Q. Tian, "Lp-norm idf for large scale image search," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 1626–1633.
- [109] G. Zeng, "Fast approximate k-means via cluster closures," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 3037–3044.
- [110] W. Zhou, Y. Lu, H. Li, and Q. Tian, "Scalar quantization for large scale image search," in *ACM Multimedia*, 2012, pp. 169–178.
- [111] G. Toliás, Y. Avrithis, and H. Jégou, "To aggregate or not to aggregate: Selective match kernels for image search," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1401–1408.
- [112] Y. Cao, C. Wang, L. Zhang, and L. Zhang, "Edgel index for large-scale sketch-based image search," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 761–768.
- [113] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, and C. Schmid, "Aggregating local image descriptors into compact codes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 9, pp. 1704–1716, 2012.
- [114] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Lost in quantization: Improving particular object retrieval in large scale image databases," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.
- [115] J. Wang, Y. Li, Y. Zhang, C. Wang, H. Xie, G. Chen, X. Gao et al., "Bag-of-features based medical image retrieval via multiple assignment and visual words weighting," *IEEE Transactions on Medical Imaging*, vol. 30, no. 11, pp. 1996–2011, 2011.
- [116] Y. Cai, W. Tong, L. Yang, and A. G. Hauptmann, "Constrained keypoint quantization: towards better bag-of-words model for large-scale multimedia retrieval," in *Proceedings of the ACM International Conference on Multimedia Retrieval*, 2012, p. 16.
- [117] W. Zhou, M. Yang, H. Li, X. Wang, Y. Lin, and Q. Tian, "Towards codebook-free: Scalable cascaded hashing for mobile image search," *IEEE Transactions on Multimedia*, vol. 16, no. 3, pp. 601–611, 2014.
- [118] W. Zhou, M. Yang, X. Wang, H. Li, Y. Lin, and Q. Tian, "Scalable feature matching by dual cascaded scalar quantization for image retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 1, pp. 159–171, 2016.
- [119] W. Zhou, H. Li, R. Hong, Y. Lu, and Q. Tian, "Bsift: Toward data-independent codebook for large scale image search," *IEEE Transactions on Image Processing*, vol. 24, no. 3, pp. 967–979, 2015.
- [120] Z. Liu, H. Li, L. Zhang, W. Zhou, and Q. Tian, "Cross-indexing of binary sift codes for large-scale image search," *IEEE Transactions on Image Processing*, vol. 23, no. 5, pp. 2047–2057, 2014.
- [121] K. S. Hasan and V. Ng, "Conundrums in unsupervised keyphrase extraction: making sense of the state-of-the-art," in *Proceedings of the International Conference on Computational Linguistics*, 2010, pp. 365–373.
- [122] M. Li and P. Vitányi, *An introduction to Kolmogorov complexity and its applications*, 2009.
- [123] F. Rousseau and M. Vazirgiannis, "Graph-of-word and tw-idf: new approach to ad hoc ir," in *Proceedings of the ACM international conference on information and knowledge management*, 2013, pp. 59–68.
- [124] A. Bookstein and D. R. Swanson, "Probabilistic models for automatic indexing," *Journal of the American Society for Information science*, vol. 25, no. 5, pp. 312–316, 1974.
- [125] K. W. Church and W. A. Gale, "Poisson mixtures," *Natural Language Engineering*, vol. 1, no. 02, pp. 163–190, 1995.

- [126] J. W. Reed, Y. Jiao, T. E. Potok, B. A. Klump, M. T. Elmore, and A. R. Hurson, "Tf-icf: A new term weighting scheme for clustering dynamic data streams," in *International Conference on Machine Learning and Applications*, 2006, pp. 258–263.
- [127] M. Lan, C. L. Tan, J. Su, and Y. Lu, "Supervised and traditional term weighting methods for automatic text categorization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 4, pp. 721–735, 2009.
- [128] A. Aizawa, "An information-theoretic perspective of tf-idf measures," *Information Processing and Management*, vol. 39, no. 1, pp. 45–65, 2003.
- [129] W. R. Greiff, "A theory of term weighting based on exploratory data analysis," in *Proceedings of the international ACM SIGIR conference on Research and development in information retrieval*, 1998, pp. 11–19.
- [130] R. E. Madsen, D. Kauchak, and C. Elkan, "Modeling word burstiness using the dirichlet distribution," in *Proceedings of the international conference on Machine learning*, 2005, pp. 545–552.
- [131] J. D. Rennie, L. Shih, J. Teevan, D. R. Karger *et al.*, "Tackling the poor assumptions of naive bayes text classifiers," in *Proceedings of the International Conference on Machine Learning*, 2003, pp. 616–623.
- [132] H. Jégou, M. Douze, and C. Schmid, "On the burstiness of visual elements," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 1169–1176.
- [133] J. Revaud, M. Douze, and C. Schmid, "Correlation-based burstiness for logo retrieval," in *ACM Multimedia*, 2012, pp. 965–968.
- [134] M. Shi, Y. Avrithis, and H. Jégou, "Early burst detection for memory-efficient image retrieval," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 605–613.
- [135] M. Murata, H. Nagano, R. Mukai, K. Kashino, and S. Satoh, "Bm25 with exponential idf for instance search," *IEEE Transactions on Multimedia*, vol. 16, no. 6, pp. 1690–1699, 2014.
- [136] A. Torii, J. Sivic, T. Pajdla, and M. Okutomi, "Visual place recognition with repetitive structures," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 883–890.
- [137] L. Xie, L. Zheng, J. Wang, A. Yuille, and Q. Tian, "Interactive: Inter-layer activeness propagation," *arXiv:1605.00052*, 2016.
- [138] X. Wang, M. Yang, T. Cour, S. Zhu, K. Yu, and T. X. Han, "Contextual weighting for vocabulary tree based image retrieval," in *Proceedings of the IEEE International Conference on Computer Vision*, 2011, pp. 209–216.
- [139] L. Zheng, S. Wang, and Q. Tian, "Coloring image search with coupled multi-index," in *IEEE China Summit and International Conference on Signal and Information Processing*, 2015, pp. 137–141.
- [140] M. Jiang, S. Zhang, R. Fang, and D. N. Metaxas, "Leveraging coupled multi-index for scalable retrieval of mammographic masses," in *IEEE International Symposium on Biomedical Imaging*, 2015, pp. 276–280.
- [141] L. Zheng, S. Wang, W. Zhou, and Q. Tian, "Bayes merging of multiple vocabularies for scalable image retrieval," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1955–1962.
- [142] Y. Xia, K. He, F. Wen, and J. Sun, "Joint inverted indexing," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 3416–3423.
- [143] J. Wang, T. Zhang, J. Song, N. Sebe, and H. T. Shen, "A survey on learning to hash," *arXiv:1606.00185*, 2016.
- [144] Y. Liu, Y. Guo, S. Wu, and M. S. Lew, "Deepindex for accurate and efficient image retrieval," in *Proceedings of the International Conference on Multimedia Retrieval*, 2015, pp. 43–50.
- [145] H. Jégou, M. Douze, and C. Schmid, "Improving bag-of-features for large scale image search," *International Journal of Computer Vision*, vol. 87, no. 3, pp. 316–336, 2010.
- [146] M. Douze, H. Jégou, and C. Schmid, "An image-based approach to video copy detection with spatio-temporal post-filtering," *IEEE Transactions on Multimedia*, vol. 12, no. 4, pp. 257–266, 2010.
- [147] M. Jain, R. Benmokhtar, H. Jégou, and P. Gros, "Hamming embedding similarity-based image classification," in *Proceedings of the 2nd ACM International Conference on Multimedia Retrieval*. ACM, 2012, p. 19.
- [148] G. Tolias and H. Jégou, "Visual query expansion with or without geometry: refining local descriptors by feature aggregation," *Pattern Recognition*, vol. 47, no. 10, pp. 3466–3476, 2014.
- [149] T. Mei, Y. Rui, S. Li, and Q. Tian, "Multimedia search reranking: A literature survey," *ACM Computing Surveys*, vol. 46, no. 3, p. 38, 2014.
- [150] D. Qin, C. Wengert, and L. Gool, "Query adaptive similarity for large scale object retrieval," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 1610–1617.
- [151] F. Perronnin and C. Dance, "Fisher kernels on visual vocabularies for image categorization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.
- [152] K. Chatfield, V. S. Lempitsky, A. Vedaldi, and A. Zisserman, "The devil is in the details: an evaluation of recent feature encoding methods," in *British Machine Vision Conference*, vol. 2, no. 4, 2011, p. 8.
- [153] H. Jégou and O. Chum, "Negative evidences and co-occurrences in image retrieval: The benefit of pca and whitening," in *European Conference on Computer Vision*, 2012, pp. 774–787.
- [154] G. McLachlan and D. Peel, *Finite mixture models*, 2004.
- [155] P.-H. Gosselin, N. Murray, H. Jégou, and F. Perronnin, "Revisiting the fisher vector for fine-grained classification," *Pattern Recognition Letters*, vol. 49, pp. 92–98, 2014.
- [156] P. Koniusz, F. Yan, and K. Mikolajczyk, "Comparison of mid-level feature coding approaches and pooling strategies in visual concept detection," *Computer vision and image understanding*, vol. 117, no. 5, pp. 479–492, 2013.
- [157] R. G. Cinbis, J. Verbeek, and C. Schmid, "Approximate fisher kernels of non-iid image models for image categorization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 6, pp. 1084–1098, 2015.
- [158] M. Douze, A. Ramisa, and C. Schmid, "Combining attributes and fisher vectors for efficient image retrieval," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 745–752.
- [159] M. Muja and D. G. Lowe, "Scalable nearest neighbor algorithms for high dimensional data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 11, pp. 2227–2240, 2014.
- [160] E. Spyromitros-Xioufis, S. Papadopoulos, I. Y. Kompatsiaris, G. Tsoumakas, and I. Vlahavas, "A comprehensive study over vlad and product quantization in large-scale image retrieval," *IEEE Transactions on Multimedia*, vol. 16, no. 6, pp. 1713–1728, 2014.
- [161] R. Arandjelovic and A. Zisserman, "All about vlad," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 1578–1585.
- [162] Y. Huang, Z. Wu, L. Wang, and T. Tan, "Feature coding in image classification: A comprehensive study," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 3, pp. 493–506, 2014.
- [163] G. Tolias, Y. Avrithis, and H. Jégou, "Image search with selective match kernels: aggregation across single and multiple images," *International Journal of Computer Vision*, vol. 116, no. 3, pp. 247–261, 2016.
- [164] R. Arandjelović, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: Cnn architecture for weakly supervised place recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.
- [165] S. Zhang, X. Wang, Y. Lin, and Q. Tian, "Cross indexing with grouplets," *IEEE Transactions on Multimedia*, vol. 17, no. 11, pp. 1969–1979, 2015.
- [166] A. S. Razavian, J. Sullivan, S. Carlsson, and A. Maki, "Visual instance retrieval with deep convolutional networks," *ITE Transactions on Media Technology and Applications*, vol. 4, no. 3, pp. 251–258, 2016.
- [167] O. Morère, A. Veillard, J. Lin, J. Petta, V. Chandrasekhar, and T. Poggio, "Group invariant deep representations for image instance retrieval," *arXiv:1601.02093*, 2016.
- [168] O. Chum, M. Perdoch, and J. Matas, "Geometric min-hashing: Finding a (thick) needle in a haystack," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 17–24.
- [169] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu, "Learning fine-grained image similarity with deep ranking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1386–1393.
- [170] C. L. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *European Conference on Computer Vision*, 2014, pp. 391–405.



- [171] X. Tian, Y. Lu, and L. Yang, "Query difficulty prediction for web image search," *IEEE Transactions on Multimedia*, vol. 14, no. 4, pp. 951–962, 2012.
- [172] X. Tian, Y. Lu, L. Yang, and Q. Tian, "Learning to judge image search results," in *ACM Multimedia*, 2011, pp. 363–372.
- [173] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 1794–1801.
- [174] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 3360–3367.



**Liang Zheng** received the Ph.D degree in Electronic Engineering from Tsinghua University, China, in 2015, and the B.E. degree in Life Science from Tsinghua University, China, in 2010. He was a postdoc researcher in University of Texas at San Antonio, USA. He is currently a postdoc researcher in Quantum Computation and Intelligent Systems, University of Technology Sydney, Australia. His research interests include image retrieval, classification, and person re-identification.



**Yi Yang** received the Ph.D. degree in computer science from Zhejiang University. He was a Post-Doctoral Research Fellow with the School of Computer Science, Carnegie Mellon University, from 2011 to 2013. He is currently an Associate Professor with the Centre for Quantum Computation and Intelligent Systems, University of Technology at Sydney, Sydney. His research interests include multimedia and computer vision.



**Qi Tian** (M'96-SM'03-F'16) received the B.E. degree in electronic engineering from Tsinghua University, China, the M.S. degree in electrical and computer engineering from Drexel University, and the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana-Champaign, in 1992, 1996, and 2002, respectively. He is currently a Professor with the Department of Computer Science, University of Texas at San Antonio (UTSA). He took a one-year faculty leave at Microsoft Research Asia

from 2008 to 2009. His research interests include multimedia information retrieval and computer vision. He has authored more than 230 refereed journal and conference papers. His research projects were funded by NSF, ARO, DHS, SALS, CIAS, and UTSA, and he also received faculty research awards from Google, NEC Laboratories of America, FXPAL, Akiira Media Systems, and HP Laboratories. He received the Best Paper Awards in PCM 2013, MMM 2013, and ICIMCS 2012, the Top 10% Paper Award in MMSP 2011, the Best Student Paper in ICASSP 2006, and the Best Paper Candidate in PCM 2007. He received 2010 ACM Service Award. He is the Guest Editor of the *IEEE Transactions on Multimedia*, *Journal of Computer Vision and Image Understanding*, *Pattern Recognition Letter*, *EURASIP Journal on Advances in Signal Processing*, *Journal of Visual Communication and Image Representation*, and is on the Editorial Board of the *IEEE Transactions on Multimedia*, *IEEE Transactions on Circuit and Systems for Video Technology*, *Multimedia Systems Journal*, *Journal of Multimedia*, and *Journal of Machine Vision and Applications*.