Large Scale Image Retrieval by Coupled Binary Embedding

[1]S.Raguvel,[2]Miss D.Dharini
[1]M.E Communication Systems, K. Ramakrishnan College of Engineering
Trichy, India
[2]Assistant Professor, K. Ramakrishnan College of Engineering
Trichy, India

Abstract

Visual matching is a crucial step in image retrieval based on the bag-of-words (BoW) model. In the baseline method, two key points are reconsidered as a matching pair if their SIFT descriptors are quantized to the same visual word. However, t he SIFT visual word has two limitations. First, it loses most of its discriminative power during quantization. Second, SIFT only describes the local texture feature. Both drawbacks impair the discriminative power of the BoW model and lead to false positive matches. To tackle this problem, this paper proposes to embed multiple binary features at indexing level. To model correlation between features, a multi-IDF scheme is introduced, through which different binary features are coupled into the inverted file. We show that matching verification methods based on binary features, such as Hamming embedding, can be effectively incorporated in our framework. As, we explore the fusion of binary color feature into image retrieval. The joint integration of the SIFT visual word and binary features greatly enhances the precision of visual matching, reducing the impact of false positive matches. Our method is evaluated through extensive experiments on four benchmark datasets (Ukbench, Holidays, Dup Image and MIR Flickr 1M). We show that our method significantly improves the baseline approach. In addition, large-scale experiments indicate that the proposed method requires acceptable memory usage and query time compared with other approaches. Further, when global color feature is integrated, our method yields competitive performance with the state-of-the-arts.

Index Terms—*Feature fusion, coupled binary embedding, multi IDF, image retrieval.*

I. INTRODUCTION

This paper focuses on the task of large scale partial duplicate image retrieval. Given a query image, our target is to find images containing the same object or scene in a large database in real time. Due to the low descriptive power of texts or tags [2], [3], content based image retrieval (CBIR) has been a hot topic in computer vision community. One of the most popular approaches to perform such a task is the Bag-of-Words (BoW) model [4]. The introduction of the SIFT descriptor [5] has enabled accurate partial-duplicate image retrieval based on feature matching. Specifically, the BoW model first constructs a codebook via unsupervised clustering algorithms [6], [7]. Then, an image is represented as a histogram of visual words, produced by feature quantization. Each bin of the histogram is weighted with tf-idf score [4] or its variants [1], [8], [9]. With the inverted file data structure, images are indexed for efficient retrieval. Essentially, one key issue of the BoW model involves visual word matching between images. Accurate feature matching leads to high image retrieval performance. However, two drawbacks compromise this procedure. First, in quantization, a 128-D double SIFT feature is quantized to a single integer. Although it enables efficient online retrieval, the discriminative power of SIFT feature is largely lost. Features that lie away from each other may actually fall into the same cell, thus producing false positive matches. Second, the state-of-the-art systems rely on the SIFT descriptor, which only describes the local gradient distribution, with rare description of other characteristics, such as color, of

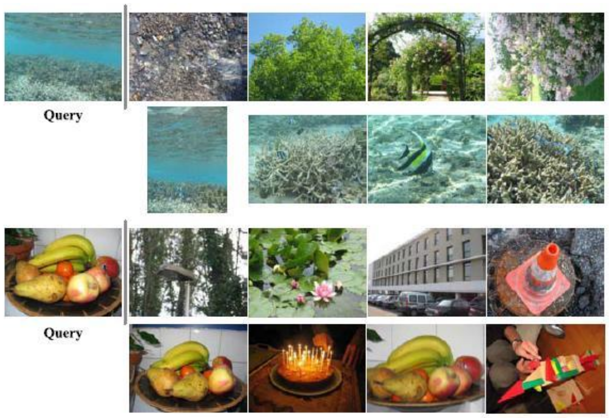

Fig.1 Two Sample Retrieval from Holiday dataset. For each query, baseline results and results with color fusion is demonstrated

this local region. As a result, regions which are similar in texture space but different in color space may also be considered as a true match. Both drawbacks lead to false positive matches and impair the image retrieval accuracy. Therefore, it is undesirable to take visual word index as the only ticket to visual matching. Instead, the matching procedure should be further checked by other cues, which should be efficient in terms of both memory and time. A reasonable choice to address the above problem involves the usage of binary features. Typically, the binary features are extracted along with SIFT, and embedded into the inverted file. The reason why binary feature can be employed for matching verification is two-fold. First, compared with floating-point vectors of the same length, binary features consume much less memory. For example, for a 128-D vector, it takes 512 bytes and 16 bytes for the floating-point and binary features, respectively. Second, during matching verification, the Hamming distance between two binary features can be efficiently calculated via *xor* operations, while the Euclidean distance between floating-point vectors is very expensive to compute. Previous work of this line includes Hamming Embedding (HE) [1] and its variants [10], [11], which use binary SIFT features for verification. Meanwhile, binary features also include spatial context [12], heterogeneous feature such as color [13], etc. In light of the effectiveness of binary features, this paper proposes to refine visual matching via the embedding of multiple binary features. On one hand, binary features provide complementary clues to rebuild the discriminative power of SIFT visual word. On the other hand, in this feature fusion process, binary features are coupled by links derived from a virtual multi-index structure. In this structure, SIFT visual word and other binary features are combined at indexing level by taking each feature as one dimension of the virtual multiindex. Therefore, the image retrieval process votes for candidate images not only similar in local texture feature, but also consistent in other feature spaces. With the concept of multiindex, a novel IDF scheme, called multi-IDF, is introduced. We show that binary feature verification methods such as Hamming Embedding, can be effectively incorporated in our framework. Moreover, we extend the proposed framework by embedding binary color feature. This paper argues that feature fusion by coupled binary feature embedding significantly enhances the discriminative power of SIFT visual word. First, SIFT binary feature retains more information from the original feature, providing effective check for visual word matching. Second, color binary feature gives complementary clues to SIFT feature (see Fig. 2 for the effects of color fusion). Both aspects serve to improve feature matching accuracy. Extensive experiments on four image retrieval datasets confirm that the proposed method dramatically improves image retrieval accuracy, while remaining efficient as well. Fig. 1 gives some examples where our method returns challenging images candidates while the conventional SIFT-based model fails. The rest of the paper is organized as follows. After a brief review of related work in Section II, we introduce the proposed binary feature embedding method based on virtual multi-index in Section III. Section IV presents the experimental results on four benchmark datasets for image retrieval applications. Finally, conclusions are given in Section V.

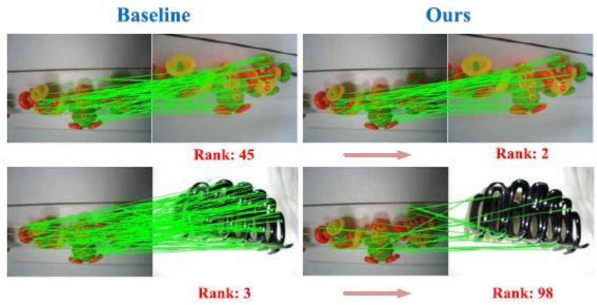

Fig.2 An example of image matching using baseline and image fusion method

II. RELATED WORK

This paper aims at improving BoW-based image retrieval via indexing-level feature fusion. So we briefly review four Fig. 2. An example of image matching using the baseline (left) and the proposed fusion (right) method. For each image pair, the query image is on the left. The first row represents matching between relevant images, while the second row contains irrelevant ones. We also show the ranks of the candidate images. We can see that the fusion of color information improves performance significantly closely related aspects, i.e, feature quantization, spatial constraint encoding, feature fusion, and indexing strategy.

A. *Feature Quantization*

Usually, hundreds or thousands of local features, e.g, SIFT [5] or its variants [14], [15] are extracted in an image. To reduce memory cost and speed up image matching, each SIFT feature is assigned to one or a few nearest centroids in the codebook via approximate nearest neighbor (ANN) algorithms [6], [7]. This process is featured by a significant information loss from a 128-D double vector to a 1-D integer. To reduce quantization error, multiple assignment [16] or soft quantization [17] is employed, which instead increase the query time and memory overload. Another choice includes the Fisher Vector (FV) [18]. In FV, the Gaussian Mixture Model (GMM) is used to train a codebook. The quantization process is performed softly by estimating the probability that a given feature falls into each Gaussian mixture. Quantization error can also be tackled using binary features. Hamming embedding [1] generates binary signatures coupling SIFT visual word for matching verification. These binary features provide information to filter out false matches, rebuilding the discriminative power of SIFT visual word. Quantization artifact can also be addressed by modeling spatial constraints among local features [12], [19], [20]. Another recent trend includes designing codebook-free methods [21], [22] for efficient feature quantization.

B. Feature Fusion

The combination of multiple features has been demonstrated to obtain superior performance in various tasks, such as Topic modeling [23], [24], boundary detection [25], character recognition [26], object classification [27] and detection [28], [29] tasks. Typically, early and late fusions are the two main approaches. Early fusion [27] refers to fusing multiple features at pixel level, while late fusion [30] learns semantic concepts directly from unimodal features. In the field of instance retrieval, feature fusion is no easy task due to the lack of sufficient training data. Douze et al. [31] combine fisher vector and attributes in a manner equivalent to early fusion. Zhang et al. [32] perform late fusion by combining rank lists of BoW and global features by graph fusion. In [33], co-indexing is employed to augment the inverted index with globally similar images. In [34], multi-level features including the popular CNN feature [35] are integrated, and state-of the-art results on benchmarks are reported on benchmarks. Wengert et al. [13] use global and local color features to provide complementary information. Their work is similar to ours in that binary color feature is also integrated into the inverted file. However, in their work the trade-off between color and SIFT hamming embeddings is heuristic and dependent on the dataset. Instead this paper focuses on the indexing-level feature fusion by modeling correlations between features, which could be generalized on different datasets, providing a different view from previous works.

C. Indexing Strategy

The inverted file structure [4] greatly promotes the efficiency of large scale image retrieval. In essence, the inverted file stores image IDs where the corresponding visual word appears. Modified inverted file may also include other cues for further visual match check, such as binary Hamming codes [1] feature position, scale, and orientation etc. For example, Zhou *et al.* perform on-the-fly spatial coding with the metadata stored in the inverted file. Zheng *et al.* employ indexing-level feature fusion with a 2D inverted file, and greatly improve retrieval efficiency. A Bayes probabilistic model is proposed to merge multiple inverted indices, while cross-indexing [41] traverses two inverted indices in an iterative manner. The closest inspiring work to ours includes [42], which addresses ANN problem via inverted multi-indices built on product quantization (PQ) [43]. In their work, each dimension of the multi-index corresponds to a segment of the SIFT feature vector, so the multi-index is a product of the "de-composition" of the SIFT feature. Opposite to [42], in this work, we "compose" SIFT and color features into the multi-index, implicitly performing feature fusion at indexing level. The multi-index used in this paper serves as an illustration of the coupling mechanism and derives the multi-IDF formula, which is a bridge between features.

III. PROPOSED APPROACH

This section provides a formal description of the proposed framework for binary feature embedding.

A. Binary Feature Verification Revisit

The SIFT visual word is such a weak discriminator that false positive matches occur prevalently: dissimilar SIFT features are assigned to the same visual word, and vice versa. To rebuild its discriminative power, binary features are employed to provide further verification for visual word matching pairs. The Hamming Embedding (HE) proposed in [1] suggests a way to inject SIFT binary feature into the retrieval system. This paper, however, exploits the embedding of multiple binary signatures from heterogeneous features.

B. Organization of the Inverted File

The inverted file is prevalently used to index database images in the BoW-based image retrieval pipeline. This data structure not only calculates the inner product between images explicitly, but, more importantly, enables efficient online retrieval process.

C. A Multi-Index Illustration

In this section, we provide an alternative explanation of the proposed method from the perspective of multi-index structure, which stands as the foundation of the multi-IDF formula. During online retrieval, each image is represented by a bag of word tuple as in the offline phase. In this manner, every keypoint is described by multiple features. Then, for each word tuple, we find and vote for the candidate images from the corresponding entry in the multi-index.

1. Embedding Binary Features:

This paper embeds binary features into the SIFT visual word framework. Due to its bitwise nature, each binary feature equals to a decimal number. So the binary feature itself can be viewed as a visual word: there is no need to train a codebook explicitly [21]. The reason why we use binary features instead of traditional visual words is that a coarser-to-fine mechanism is implied in binary features. Basically, the Hamming distance between two binary features represents their similarity, while the traditional visual word only allows a “hard” matching mode. A binary feature of *k* bits corresponds to a codebook of size 2*k*. Consequently, we can adapt each binary feature to the virtual multi-index easily. Specifically, the SIFT binary features involved in [1] can be coupled with the SIFT visual word in the virtual multi-index as well.

D. Fusion of Color Feature

In this paper, we embed color feature with SIFT at indexing level.

1) Color Descriptor

This paper employs the Color Names (CN) descriptor [28] for two reasons. First, it is shown in [28] that CN has superior performance compared with several commonly used color descriptors such as the Robust hue descriptor [49] and Opponent derivative descriptor [49]. Second, although colored SIFT descriptors such as HSV-SIFT [50] and Hue SIFT [51] provide color information, the descriptors typically lose some invariance properties and are high-dimensional [52]. Basically, the CN descriptor assigns to each pixel a 11-D vector, of which each dimension encodes one of the eleven basic colors: black, blue, brown, grey, green, orange, pink, purple, red, white and yellow. The effectiveness of CN has been validated in image classification and detection applications [27], [28]. We further test it in the scenario of image retrieval.

2) Feature Extraction

At each keypoint, two descriptors are extracted, *i.e.*, a SIFT descriptor and a CN descriptor. In this scenario, SIFT is extracted with the standard algorithm [5].As with CN, we first compute CN vectors of pixels surrounding the keypoint, with the area proportional to the scale of the keypoint. Then, we take the average CN vector as the color feature. The two descriptors of a keypoint are individually quantized, binarized, and fed into our model, respectively.

3) Binarization

Because the CN descriptor has explicit semantic meaning in each dimension, we do not adopt the classical clustering method to perform quantization. Instead, we directly convert a CN vector into a binary feature, which itself can be viewed as a distinct visual word [21]. Specifically, we try two binarization schemes, producing 11-bit vector *b (*11*)* and 22-bit vector *b (*22*)*, respectively.

IV. EXPERIMENTS

To evaluate the effectiveness of our method, we conducted experiments on four public benchmark datasets: the Ukbench [7], the Holidays [1], the DupImage [38], and the MIR Flickr 1M [56] datasets.

A. Datasets

1) Ukbench

The Ukbench dataset consists of 10200 images of 2550 groups. Each group contains four images of the same object or scene, taken from different viewpoints. Each of the 10200 images is taken as query image. The performance is measured by the recall for the top-4 candidates, referred to as N-S score (maximum 4).

2) Holidays

The Holidays dataset is composed of 500 queries from 1491 annotated personal holiday photos. mAP(mean Average Precision) is used to evaluate the performance.

3) DupImage

The DupImage dataset contains 1104 images from 33 annotated groups. From this ground truth dataset, 108 representative queries are selected, and mAP is employed to evaluate image retrieval performance.

4) MIR Flickr 1M

This dataset contains 1 million images retrieved from Flickr. We add this dataset to the Holidays, Ukbench, and DupImage datasets to test the scalability of the proposed method.

V. CONCLUSION

Binary Embedding methods are effective for visual matching verification. In this paper, we propose a coupled binary embedding method using a binary multi-index framework to fuse SIFT visual word with binary features at indexing level. To model the correlation between different features, we introduce a new IDF family, called the multi-IDF, which can be viewed as a weighted sum of individual IDF of each fused feature. Specifically, we explore the integration of the local color descriptor in the retrieval framework. Through extensive experiments on three benchmark datasets, we show that significant improvement can be achieved when multiple binary features are fused. Moreover, we demonstrate the effectiveness of multi-IDF in coupling different binary features. Further, when merged with global color feature by graph fusion, we are capable of outperforming the state of-the-art methods. In large-scale settings, by storing binary features in the inverted file, the proposed method consumes acceptable memory usage and query time compared with other approaches.

In the future work, more investigation will be focused on the mechanism of how features complement each other and promote visual matching accuracy. Since our method can be easily extended to include other binary features such as recently proposed ORB [44], BRISK [45], FREAK [46], etc, various feature fusion and selection strategies will also be explored to further improve performance.

REFERENCES

[1] H.Jégou, M. Douze, and C.Schmid" Hamming embedding and weak geometric consistency for large scale image search," in Proc. 10th Eur.Conf. Comput. Vis. ECCV, 2008, pp. 304–317.

[2] W. Lu, J. Wang, X.-S. Hua, S.Wangand S. Li "Contextual image search," in Proc. 19th ACM Multimedia, 2011, pp. 513–522.

[3] X. Li, Y.-J.Zhang, B.Shen and B.-D. Liu, "Image tag completion by low-rank factorization with dual reconstruction structure preserved," in Proc. IEEE Int. Conf, Image Process. (ICIP), Oct. 2014.

[4] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in Proc. IEEE Int. Conf. Comput. Vis.,(ICCV), Oct. 2003, pp. 1470–1477.

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

[6] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in Proc.Comput. Vis. Pattern Recognit. (CVPR), 2007, pp. 1–8.

[7] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in Proc. Comput. Vis. Pattern Recognit. (CVPR),vol.2. 2006, pp. 2161–2168.

[8] L.Zheng, S.Wang, Z. Liu, and Q. Tian, "Lp-norm IDF for large scale image search,"in Proc. IEEE Comput. Vis. Pattern Recognit. (CVPR),Jun.2013, pp. 1626–633.

[9] L. Zheng,S. Wang, and Q. Tian, "Lp norm IDF for scalable image retrieval,"IEEE Trans. Image Process., to be published.

[10] D. Qin and C. W. L. van Gool, "Query adaptive similarity for large scale object retrieval," in Proc. IEEE Comput. Vis. Pattern Recognit. (CVPR),Jun. 2013, pp. 1610–1617.

[11] G. Tolias, Y. Avrithis, and H. Jégou, "To aggregate or not to aggregate: Selective match kernels for image search," in Proc. IEEE Int. Conf.Comput. Vis., (ICCV), Dec. 2013, pp. 1401–1408.

[12] Z. Liu, H. Li, W. Zhou, R. Zhao, and Q. Tian, "Contextual hashing forlarge-scale image search," IEEE Trans. Image Process., vol. 23, no. 4,pp. 1606–1614, Apr. 2014.

[13] C.Wengert, M. Douze, and H. Jégou, "Bag-of-colors for improved image search," in Proc. 19th ACM Multimedia, 2011, pp. 1437–1440.

[14] R. Arandjelovic and A. Zisserman, "Three things everyone should know to improve object retrieval," in Proc. IEEE Comput. Vis. Pattern Recognit. (CVPR), Jun. 2012, pp. 2911–2918.

[15] K.Simonyan, A. Vedaldi, and A. Zisserman, "Descriptor learning using convex optimisation," in Proc. 12th Eur. Conf. Comput. Vis. (ECCV),Oct. 2012, pp. 243–256.

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

[17] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Lost in quantization: Improving particular object retrieval in large scale image databases," in Proc. IEEE Comput. Vis. Pattern Recognit. (CVPR), Jun. 2008, pp. 1–8.

[18] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in Proc. 11th Eur. Conf. Comput.Vis. (ECCV), 2010, pp. 143–156.

[19] L.Xie, Q. Tian, and B. Zhang, "Spatial pooling of heterogeneous features for image classification," IEEE Trans. Image Process., vol. 23, no. 5, pp. 1994–2008, May 2014.

[20] W. Zhou, H. Li, Y. Lu, and Q. Tian, "Principal visual word discovery for automatic license plate detection," IEEE Trans. Image Process., vol. 21, no. 9, pp. 4269–4279, Sep. 2012.