INTERNATIONAL JOURNAL OF RESEARCH
In Advanced Engineering Technologies

Volume 5, Issue 2 NOV 2015

OPTIMIZED LARGE SCALE IMAGE RETRIEVAL APPROACH BASED COUPLED BINARY EMBEDDING

S. GOPAL NAIK [1] SHAIK GULAM RABBANI [2] V. GOUTHAM[3]

Department of CSE [1] Assistant Professor M.tech (CNIS)[2] Professor Ph.D[3]

Teegala Krishna Reddy Engineering College, Medbowli, Meerpet, Saroornagar, Hyderabad, Telangana, INDIA

Abstract

Advance development in Internet technologies result in generation of large-scale image data which help to decipher crisis in existing computer vision and numerous real-world applications. These large scale image dataset need competent large-scale processing techniques for efficient and effective image retrieval system. For browsing and searching image from large data, user necessitates compelling interfaces and functionalities for capturing visual attributes. Image retrieval is related to the fields of image processing, multimedia, digital libraries, remote sensing, astronomy, database applications and others related area. Visual matching could be a crucial step in image retrieval supported the bag-of-words (Bow) model. Within the baseline technique, 2 key points are thought of as an identical try if their SIFT descriptors are amount to identical visual word. However, the SIFT visual word has 2 limitations. First, it loses most of its discriminative power throughout division. Second, SIFT solely describes the native texture feature. Each drawbacks impair the discriminative power of the Bow model and result in false positive matches. To tackle this downside, multiple binary options are embedded at assortment level. To model correlation between options, a multi-IDF scheme is introduced, through that completely different binary options are coupled into the inverted file. Matching verification methods are supported binary options. The joint integration of the SIFT visual word and binary options greatly enhances the preciseness of visual matching, reducing the impact of false positive matches. The planned technique considerably improves the baseline approach. Additionally, massive scale experiments indicate that the planned technique needs acceptable memory usage and question time compared with alternative approaches.

Keywords: Feature fusion, coupled binary embedding, multi-IDF, image retrieval

1. INTRODUCTION

Image retrieval is deal with searching and retrieving digital images from a huge database. An effective image retrieval system is able to operate on the collection of images to retrieve the relevant images based on the query image which conforms as closely as possible to human perception. According to database management and computer vision communities, there are two different perspectives, text-based and content or visual based. Text based image retrieval techniques use text to describe the content of the image and content based image retrieval used image visual features to describe the content of images.

1.1 Conventional or text based image retrieval

Conventional image retrieval is text based retrieval system where keywords are used as descriptors to index an image. Text-based image retrieval techniques use text to describe the content of the image which often causes vagueness and insufficiency in performing an image database search and query processing. Weakness of text based image retrieval is difficulty in specifying exact terms and phrases in describing the content of images as keywords. Textual annotations are based on language, difference in annotation change retrieval result totally.

1.2 Content or visual based Image Retrieval

Visual Image Retrieval is content based image retrieval system use visual features such as color, texture, shape and spatial relations and or high level which extracted from the image itself.

1.3 Need of improvement in large scale image retrieval

Large scale image search is challenging search for particular image through a large database to find similar target images. It involve internet search to find similar images. Basic requirement of today's search engine is that it should be fast, accurate and scalable to large data set. As per prerequisite of many real word application such as medical application, space application etc., they need fast and accurate retrieval result. Internet is full of millions of images which increase rapidly as with growth of internet, hence, it is really challenging to measure similarity between images and image retrieval system with large and growing database. Existing image retrieval systems are slightly imprecise, time consuming and not scalable. To overcome these issues, research need to improve accuracy and speed of existing image retrieval system.

This paper focuses on the task of enormous scale partial duplicate image retrieval. Given a question image, our target is to seek out pictures containing constant options in an exceedingly massive info in real time. A picture retrieval system could be a computing system for browsing, looking and retrieving pictures from an outsized info of digital pictures. Most ancient and customary ways of image retrieval utilize some technique of adding data such as captioning, keywords, or descriptions to the pictures so retrieval will be performed over the annotation words. One of the foremost well-liked approaches to perform such a task is that the Bag-of-Words (BOW) model. The introduction of the SIFT descriptor has enabled correct partial-duplicate image retrieval supported feature matching. Specifically, the BOW model 1st constructs a codebook via unattended bunch algorithms. Then, a picture is depicted as a bar graph of visual words, created by feature division. every bin of the bar graph is weighted with tfidf score or its variants. With the inverted file arrangement, pictures area unit indexed for economical retrieval.

Essentially, one key issue of the BOW model involves visual word matching between pictures. Accurate Feature matching ends up in high image retrieval performance. However, 2 drawbacks compromise this procedure. First, in quantization, a 128-D double SIFT feature is quantal to a single integer. The discriminative power of SIFT feature is basically lost. Options that lie off from one another may very well comprise an equivalent cell, so manufacturing false positive matches. Second, the progressive systems accept the SIFT descriptor, that solely describes the native gradient distribution, with rare description of alternative characteristics, like color, of this native region. As a result, regions that area unit similar in texture area however totally different in color area might also be thought-about as a real match. Each drawbacks cause false positive matches and impair the image retrieval accuracy. Therefore, it's undesirable to require visual word index because the solely price tag to visual matching. Typically, the binary options area units extracted along side SIFT, and embedded into the inverted file. The explanation why binary feature will be used for matching verification is two-fold. First, compared with floating point vectors of an equivalent length, binary options consume a lot of less memory. As an example, for a 128-D vector, it takes 512 bytes and sixteen bytes for the floating-point and binary options, severally. Second, throughout matching verification, the playing distance between 2 binary options will be expeditiously calculated via xor operations, whereas the Euclidian distance between floating-point vectors is extremely dear to figure. Previous work of this line includes playing Embedding (HE) and its variants , that use binary SIFT options for verification. Meanwhile, binary options additionally embody abstraction context, heterogeneous feature.

In lightweight of the effectiveness of binary options, this paper proposes to refine visual matching via the embedding of multiple binary options. On one hand, binary options give complementary clues to build the discriminative power of SIFT visual word. On the opposite hand, during this feature fusion method, binary options are coupled by links derived from a virtual multi-index structure. During this structure, SIFT visual word and different binary options are combined at categorization level by taking every feature joined dimension of the virtual multi-index. Therefore, the image retrieval method votes for candidate pictures not solely similar in native texture feature, however conjointly consistent in different feature areas. With the thought of multi-index, a unique IDF theme, referred to as multi-IDF, is introduced. We have a tendency to show that binary feature verification strategies like performing Embedding, will be effectively incorporated in our framework. Moreover, we have a tendency to extend the planned framework by embedding binary color feature.

2. RELATED WORK

In this section, we discuss some recent innovative development in large scale image retrieval field such as Sentiment of image detection, image search boosted with iterative quantization hashing method, removal of noisiness in retrieval by two sage searches with query log analysis. New-fangled techniques in large scale image retrieval technique are also involved such as latent semantic analysis, query-adaptive re ranking, and dimensionality reduction methods, a novel indexing based on a graphical model or a matrix factorization.

Coupled Binary Embedding for Large-Scale Image Retrieval is introduced with successful and accurate image retrieval. Proposed system exploits multiple binary features at indexing level, multi-IDF scheme, hamming embedding and the fusion of binary color feature into image retrieval. Content-based large scale image retrieval using framework of VLAD and Product Quantization is proposed. The system is employing more efficient and discriminative local features, improving the quality of the aggregated representation; and optimizing the indexing scheme. Latent semantic analysis is applied successfully and cost effectively to large scale medical image databases. The Latent semantic analysis is applied effectively to large dataset by skipping the SVD solution of the feature matrix. Iterative quantization hashing method is successfully implemented large scale image retrieval system with query-adaptive re ranking to achieve good search performance. High Dimensionality is one of the main problems faced by large scale image retrieval. Dimensionality reduction methods are reduced the dimensions of feature vectors while maintaining high performance. These methods can be used to generate vocabulary trees based on it process large-scale image retrieval. Sentiment of image detection in Large scale image retrieval are effectively performed by using Large-scale visual sentiment ontology and detectors using adjective noun pairs with significant improvement in detection accuracy. This system understands sentiment in image and constructs a large-scale Visual Sentiment Ontology and Second concept proposed SentiBank. Many text indexing techniques have been applied in large-scale image retrieval systems to support accurate visual search. But, different indexing techniques are expected for image query and text query as there is lot of difference between these two types of queries. Large scale visual search with new indexing technique is proposed which implements decomposition achieved via a graphical model or a matrix factorization approach.

A new document image descriptor based on multi-scale run length histograms used for large-scale applications. The contextual visual vocabulary merges both spatial and semantic clues which provide best in both retrieval precision and efficiency for large-scale near-duplicated image retrieval. Large scale image retrieval is present with sketched based image feature where image search is start with similar structure, analyzing gradient orientations. The query log of a real CBIR system is designed for large scale retrieval. In this system, similarity caching system stores the results of recently/frequently submitted queries and associated results. To deal with especially noisiness in large scale image retrieval based content, a dynamic two-stage retrieval approach is proposed which improve effectiveness. Dynamic two-stage method can be significantly more effective and robust by using local feature derivatives in the visual stage instead of global.

3. SYSTEM DESIGN

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 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. A binary feature can be generated as,

$$f = (f_1, f_2, \dots, f_n)^T \xrightarrow{q(.)} b = (b_1, b_2, \dots, b_m) \quad (1)$$

where an n-D feature f is projected into an m-D binary signature b, which is stored in the inverted file. During online query, given a query feature x, its matching strength with a indexed feature y which is quantized to the same visual word with x can be written as,

$$f(x,y) = \begin{cases} exp\left(-d_b^2/\sigma^2\right) & ,if\ d_b < k, \\ 0 & otherwise \end{cases} \quad (2)$$

where db denotes the Hamming distance between the binary signatures of x and y, σ is a weighting parameter, and κ is a predefined threshold. If db exceeds κ, then x and y are viewed as a false match and rejected. In this scenario, given a query image Q, and a database image I, the similarity function between them can be formulated as,

$$sim(Q,I) = \frac{\sum_{x\in Q, y\in I} f(x,y).idf^2}{\|Q\|2\|I\|2} \quad (3)$$

where id f stands for the inverse document frequency (IDF) of the corresponding visual word. For a conventional inverted file, the IDF value of visual word wi is formulated as,

$$idf(w_i) = \log\frac{N}{n_i} \quad (4)$$

where N represents the overall number of images in the corpus, and ni denotes the number of images in which wi appears. The basic idea of IDF is to assign more weight to rare words, and less weight to frequent words. In this paper, the binary features used are not limited to those derived from the original SIFT. It also involves other heterogenous binary features, such as color feature, or potentially, the recently proposed ORB, BRISK, FREAK, etc. In essence, our task is to perform feature fusion on the indexing level bridged by the introduction of multi-IDF scheme, which differs from previous works on binary verification significantly.

B. Organization of the Inverted File

The inverted file is prevalently used to index database images in the $\mathbf{B_0W}$-based image retrieval pipeline. This data structure not only calculates the inner product between images explicitly, but, more importantly, enables efficient online retrieval process. We assume that an image collection possesses N images denoted as **D**=$\{I_i\}_{i=1}^N$ Each image **Ii** h as a set of key points. $\{\mathbf{x_j}\}_{\mathbf{j=1}}^{\mathbf{d_i}}$ Where $\mathbf{d_i}$is the number of key points in $\mathbf{I_i}$**.** Given a codebook $\{\mathbf{w_i}\}_{\mathbf{k=1}}^{\mathbf{k}}$ of size K, image I_i is quantized to a vector representation, $\mathbf{v_i} = [\mathbf{v_{i}}, \mathbf{1}, \mathbf{v_{i,2}}, \dots, \mathbf{v_{i,k}}]^{\mathbf{T}}$ where $\mathbf{v_{i,k}}$ stands for the response of visual word $\boldsymbol{\omega_k}$ in $\mathbf{I_i}$. A conventional inverted file can be denoted as **W**=$\{\boldsymbol{W_1}, \boldsymbol{W_2}, \dots, \boldsymbol{W_K}\}$**.** In W, each entry $\boldsymbol{\omega_i}$ contains a list of postings. For an image-level inverted file, each posting stores the image ID and the TF score. For a keypoint-level inverted file, each posting stores the image ID and other metadata associated with the indexed keypoint

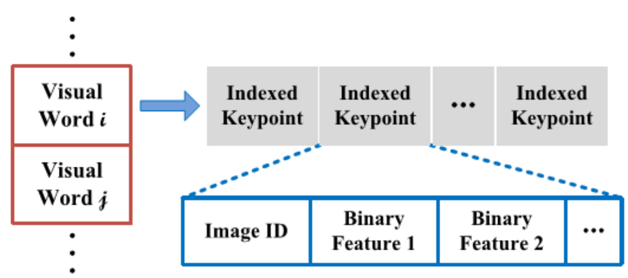

Figure 1: Structure of the key point-level inverted file.

A key point-level inverted file is organized as shown in Fig. 3. Similar to Hamming Embedding, binary features of each SIFT visual word are stored in the corresponding entry of the 1-D inverted file. However, our work differs from in two aspects.

C. A Multi-Index Illustration

With M kinds of features, the "dimension" of the multi index is M. Each dimension corresponds to a conventional inverted file of feature. F_m**,m=1,2,…,M** Then building the multi-index can be processed as follows. First, for each key point $\{\mathbf{x_i}\}$ in an image, multiple descriptors $(\mathbf{f_i^0}, \mathbf{f_i^1}, \dots, \mathbf{f_i^M})$are computed. Then, the descriptors associated with a key point are quantized into a visual word tuple $(\mathbf{W_i^0}, \mathbf{W_i^1}, \dots, \mathbf{W_i^M})$using codebooks $\{\mathbf{C^m}\}_{\mathbf{m=0}}^{\mathbf{M}}$ of each feature. Finally, for each tuple, an entry in the multi-index is identified, where the metadata of this key point can be stored.

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. 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.

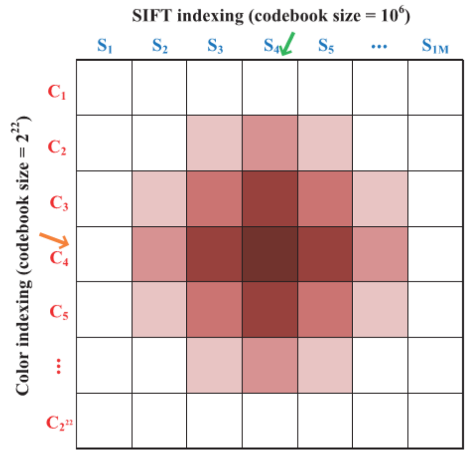

Figure 2: An example of binary multi-index fusing SIFT and color feature. The codebook sizes are 1M and 222, respectively. During online retrieval, the entry corresponding to word tuple (S4, C4) is located. Then, the neighborhood entries1 are also checked as an implementation of Multiple Assignment (MA). Color indicates the weight of these entries. A darker color signifies a larger weight.

D. Multi-IDF Formula

(1) Conventional IDF: In essence, as Robertson suggests, IDF weight can be interpreted in terms of probability, i.e., the probability that a random image I contains visual word wi, estimated as

$$p(\omega_i \; occurs \; in \; I) \approx \frac{n_i}{N} \quad (5)$$

Note that the independence assumption is also taken in the whole BoW model. Under this assumption, the IDF value can be estimated as the inverse of the fraction in Eq. 5, plus a logarithm operator,

$$idf(\omega_i) = -\log P(\omega_i) = \log \frac{N}{n_i} \quad (6)$$

(2) Proposed IDF: Without loss of generality, we illustrate the case of 2D multi-index. For each entry,

its IDF value is determined by the probability that a visual word tuple (wi0,w1 j) occurs in an arbitrary image I. For simplicity, assume features F0 and F1 are independent, i.e, at each keypoint, the value distribution of the two features does not affect each other. Therefore, the occurrence of visual words wi0 and w1 j is also independent. We can derive

$$\mathrm{p}(\omega_i^0, \omega_j^1) = p(\omega_i^0)p(\omega_j^1) \approx \frac{n_i^0}{N} \cdot \frac{n_j^1}{N} \qquad (7)$$

where n0 i and n1 j stand for the number of images containing visual word wi and w j for feature F0 and F1, respectively.2 Following Eq. 6, the IDF value of this entry can be calculated as,

$$\mathrm{Idf}(\omega_i^0, \omega_j^1)^{\mathrm{indep}} = -log\ p(\omega_i^0, \omega_j^1)$$

$$= -\left(log\ p(\omega_i^0) + log\ p(\omega_j^1)\right) = \mathrm{idf}(\omega_i^0) + \mathrm{idf}(\omega_j^1) \qquad (8)$$

Nonetheless, when the independence assumption does not hold, the multi-IDF formula would undergo some small changes. Consider the extreme case where two features are perfectly dependent, e.g, they are the same feature, the IDF formula is the same as the conventional IDF,

$$\mathrm{idf}(\omega_i^0, \omega_j^1)^{\mathrm{dep}} = \mathrm{idf}(\omega_i^0) = \mathrm{idf}\ (\omega_j^1) \qquad (9)$$

In light of Eq. 8 and Eq. 9, IDF value for two partially dependent features can be viewed as a weighted sum of each individual IDF,

$$\mathrm{idf}(\omega_i^0, \omega_j^1) = idf(\omega_i^0) + t.idf(\omega_j^1), t \in [0,1] \qquad (10)$$

E. Fusion of Color Feature

1) Color Descriptor: This paper employs the Color Names (CN) descriptor for two reasons. First, it is shown in that CN has superior performance compared with several commonly used color descriptors such as the Robust hue descriptor and Opponent derivative descriptor. Second, although colored SIFT descriptors such as HSV-SIFT and Hue SIFT provide color information, the descriptors typically lose some invariance properties and are high-dimensional.

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. We further test it in the scenario of image retrieval.

2) Feature Extraction: At each key point, 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 key point, with the area proportional to the scale of the key point. Then, we take the average CN vector as the color feature. The two descriptors of a key point are individually quantized, binaries, 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.

Suppose the CN vector is presented as (f1, f2,. . ., f11)T, and the binarization can be processed as follows

$$b_i^{(11)} = \begin{cases} 1, & if \quad f_i \ge t\hat{h}, \\ 0 & if \quad f_i < \ t\hat{h} \end{cases} \qquad (12) \quad ,$$

$$(b_i^{(22)}, b_{i+11}^{(22)}) = \begin{cases} (1,1), & if \quad f_i > t\hat{h}_1, \\ (1,0), & if \ t\hat{h}_2 < f_i \le t\hat{h}_{1,} \\ (0,0), & if \quad f_i \le t\hat{h}_2 \end{cases} \qquad (13)$$

where b_i(i = 1, 2,. . ., 11) is the ith entry of the resulting binary feature. Thresholds t$\hat{h}$= g3,t $\hat{h}_1$=g_2, t $\hat{h}_2$=g_5, where (g_1, g_2,. . . , g_{11}) is the sorted vector of (f_1, f_2,. . ., f_3) in descending order. A comparison of the two quantizers (Eq. 12 and Eq. 13) is shown in Section IV-C. Intuitively, the binarization schemes introduced in Eq. 12 and Eq. 13 represent a uniform partition (similar to the lattice partitioning in [53]) of the CN feature space. In fact, a vital difference between the CN vector and the SIFT vector is that each entry of the CN vector has explicit physical meaning. In Eq. 13, we propose to assign (1, 1) to the two most salient colors of a local region and (0, 0) to the least dominant colors. We speculate that two regions are of a similar color if they have the same dominant colors, and vice versa. Another consideration of Eq. 13 is that the minor colors may be subject to the impact of illumination or view changes. In this manner, this binarization scheme is more robust to image variations. On the other hand, one may ask why we choose $g_2$and g_5, instead of a more uniform threshold setting. In fact, region for CN extraction is quite small: in typical cases it covers tens of pixels. In such a small patch, the number of dominant colors is small (see Fig. 5 for an example). After counting the several dominant colors, approximately half of the CN dimensions are close to zero. Therefore, we use the presented thresholds which also yield satisfying performance in the experiments.

Figure 3: Sample images used to calculate the correlation coefficients. In this example, images in each row are obtained by the same text query. The text queries used to crawl these images are (from top to bottom): “Asia”, “Atlanta”, “Babados”, and “Brazil”.

Therefore, in this paper, we do not employ the binarization method proposed in Nevertheless, this paper provides a comparison with the standard Locality-Sensitive Hashing (LSH) [54] as well as the state-of-the-art Kernel zed Locality Sensitive Hashing (KLSH) methods in Section IV-C.

4) Estimation of Feature Correlation: To determine t in Eq. 10, we propose a simple scheme to measure the correlation between features. We take as an example calculating the feature correlation between SIFT visual word and binary CN. To this end, we crawled 200K high-resolution images uploaded by users from Flickr using the names of 60 countries and regions across the world. These images are generally high-resolution, with the most common size of 1024 × 768. The content of the images are very diverse, from scenes to objects, which can be viewed as a good representation of natural images. Some sample images are shown in Fig. 6. From the images, we extract over 2 × 109 (SIFT, CN) feature tuples. Then, we perform quantization on the two features: classical codebook for SIFT, and 22-bit quantize for CN. Further, two histograms are calculated using these data: For feature tuple with the same/different SIFT visual word, compute the Hamming distance of CN features, and calculate the normalized distance histogram. Finally, we calculate the correlation coefficient of the two histograms, which serves as the estimation of feature correlation parameter t.

Briefly, the intuition is that, for highly independent features, the two histograms should be very similar: whether or not the SIFT visual words of a keypoint pair are the same, the Hamming distance of the other feature is not affected. In this case, the correlation coefficient of the two histograms is close to 1, which also means a larger t. The "dependent features" case can be analysed in a similar way. Results for feature correlation calculation is presented in Fig. 7. Specifically, we present the statistical results obtained on the Flickr200K and Holidays datasets. Note that, the reason why the Holidays dataset is included is that the correlation coefficient may be subject to dataset bias, or dataset dependency. The correlation coefficients on the two datasets are 0.950 and 0.931, respectively. It suggests that the global correlation between SIFT visual word and CN feature may be independent on datasets.

4. RESULTS

Figure 4: Similarity measurement in terms of City block

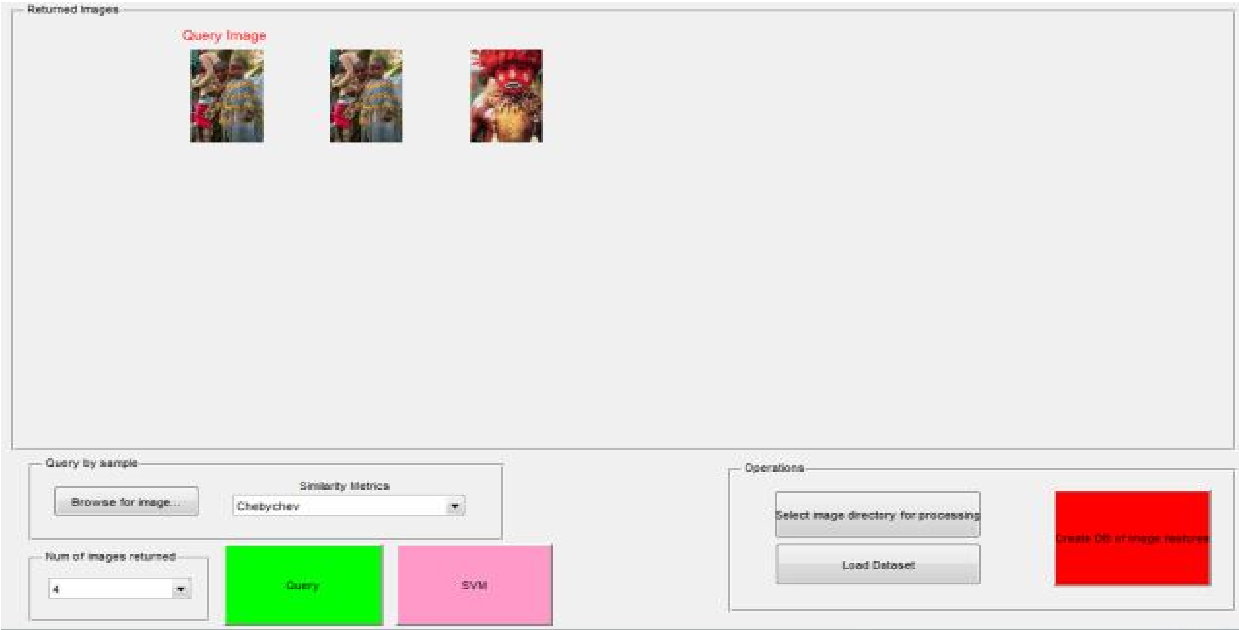

Figure 5: Similarity measurement in terms of Chebychev

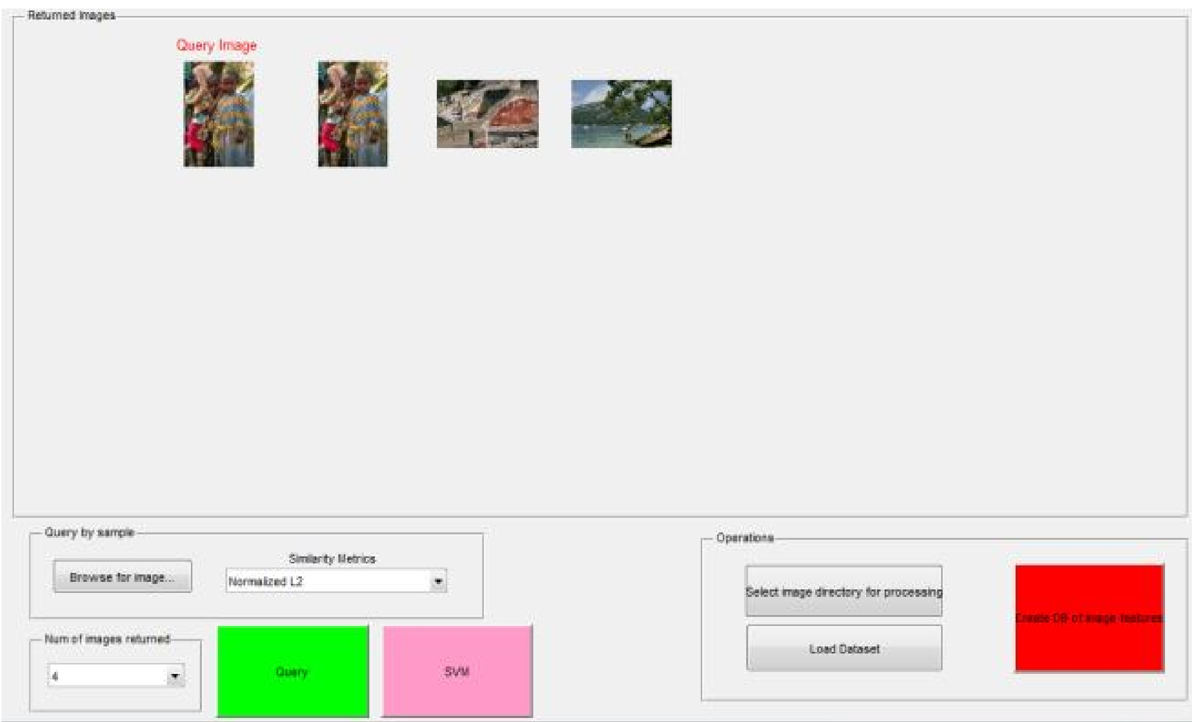

Figure 6: Similarity measurement in terms of normalized L2

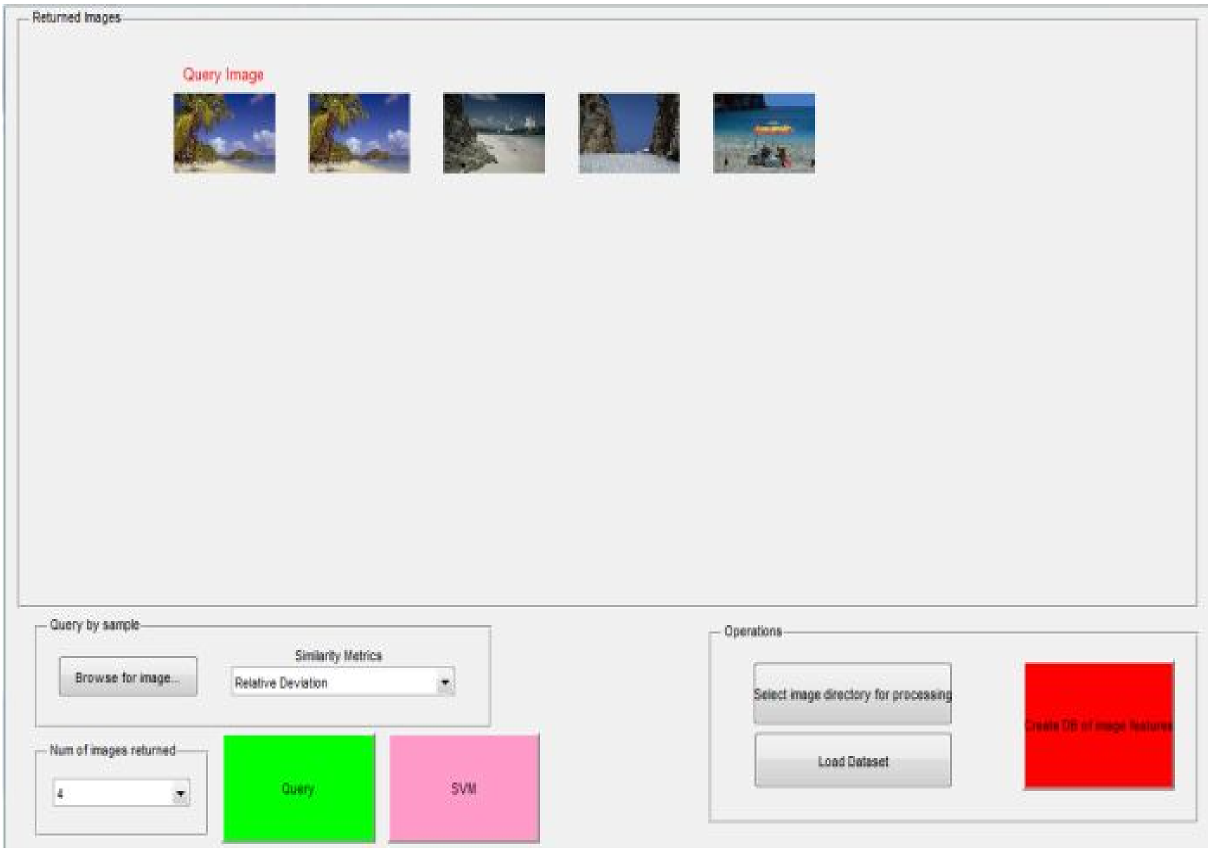

Figure 7: Similarity measurement in terms of Relative deviation

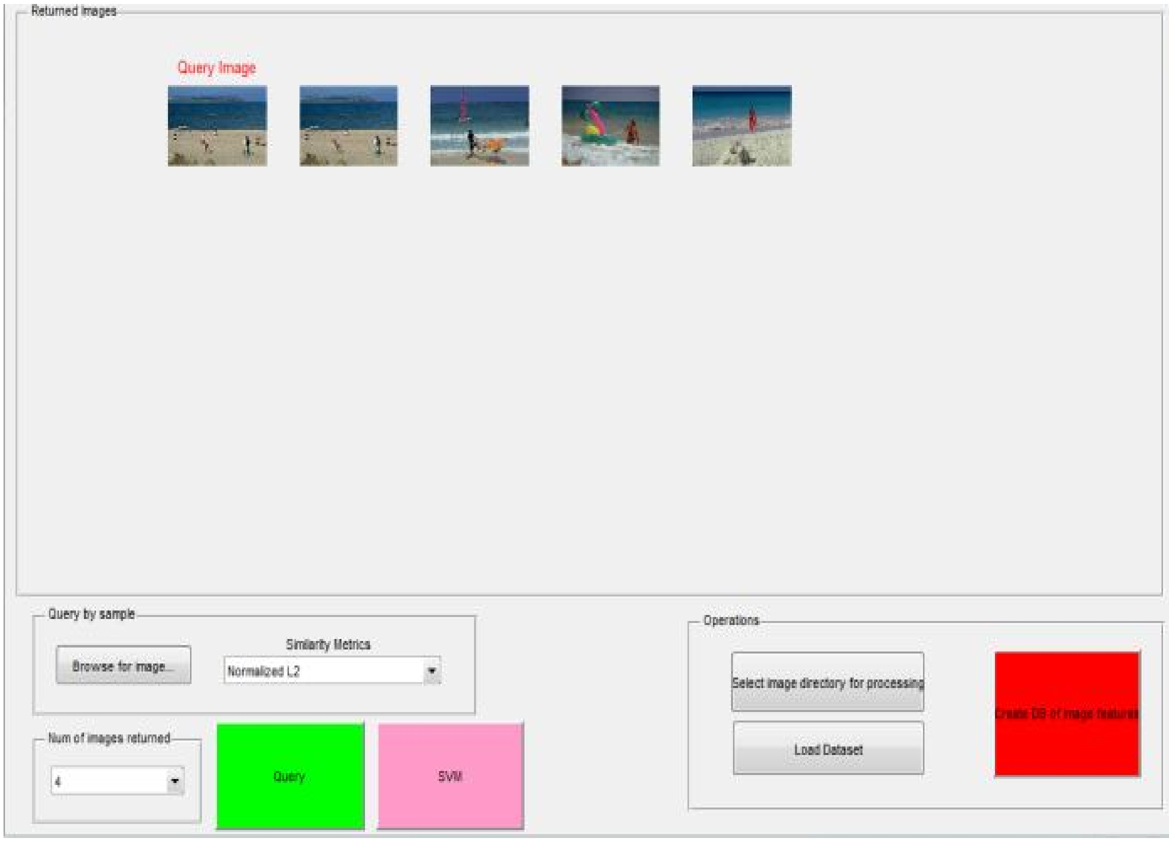

Figure 8: Similarity measurement in terms of normalized L2 Sky image

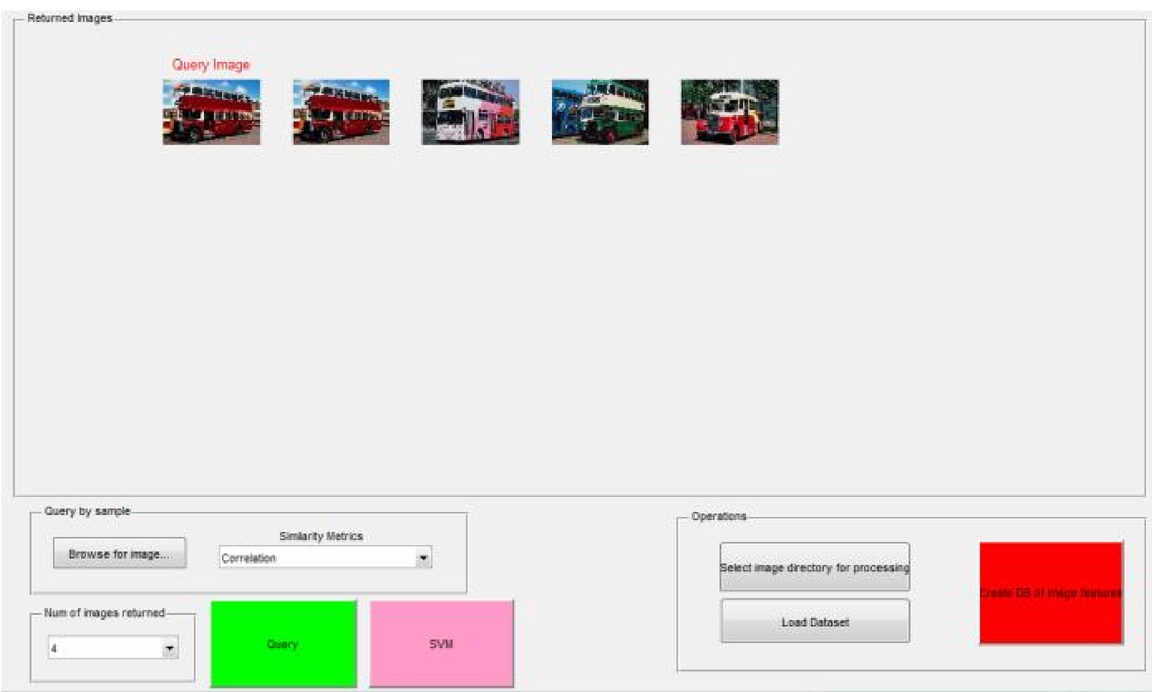

Figure 9: Similarity measurement in terms of Correlation

5. CONCLUSION

Binary Embedding methods are effective for visual matching verification. A coupled binary embedding method using a binary multi-index framework to fuse SIFT visual word with binary features at indexing level is proposed. To model the correlation between different features, a new IDF family is introduced, called the multiIDF, which can be viewed as a weighted sum of individual IDF of each fused feature. 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.

REFERENCES

1. D. G. Lowe, “Distinctive image features from scale-invariant key points,” Int. J. Comput. Vis., vol. 60, no. 2, pp. 91–110, 2004.

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

3. 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.

4. David G. Lowe “Distinctive Image Features from Scale-Invariant key points”Computer Science Department University of British Columbia Vancouver, B.C., Canada,January 5, 2004.

5. 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.

6. 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.

7. 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.

8. Liang Zheng, Shengjin Wang “Coupled Binary Embedding For Large-Scale Image Retrieval”, IEEE transactions on image processing, vol. 23, no. 8, august 2014

9. 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. 1626–1633.

10. L. Zheng, S. Wang, and Q. Tian, “Lp-norm IDF for scalable image retrieval,” IEEE Trans. Image Process., to be published.