ISSN 2278 – 0211 (Online)

Class Based Image Search with Hash Codes

Ruksanamol T. A.
Information Technology, Ilahia College of Engineering and Technology, Ernakulam, India
Femithamol A. M.
Information Technology, Ilahia College of Engineering and Technology, Ernakulam, India

Abstract:
With the proliferation of images on the internet, there is a strong need to develop an efficient image search system. In this paper proposes an efficient image retrieval system using hash codes..Hash codes reduce the memory computation during the search. With the hash codes image matching can be efficiently measured. The hashing method import high level image features in to hamming space performing search based on hamming distance. In this paper proposes mainly two tasks such as offline processing and online processing .In the offline processing bitwise weight of the hash codes for the predefined classes are generated. In online processing computing query adaptive weight. Then the images can be ranked by weighted hamming distance at a finer grained hash code level.

Keywords*: hash codes; weighted hamming distance; image search*

1. Introduction

Images play an important role in various fields such as art gallery, medical, journalism and entertainment. It is necessary to develop an efficient image search system to retrieve images from large database collections.

Generally a large scale image retrieval system consists of two components. First an effective image feature representation and then an effective image search mechanism. These two factors are important because it is well known that the quality of search results heavily depends on the image feature representation and also an efficient search mechanism is critical since most of the existing image features are high dimensions and current image databases are huge, on top of which exhaustively comparing a query with every database sample is computationally prohibitive.

In this work, images are represented using bag-of-visual-words (BOW) framework [2], where local invariant image features (eg.,SIFT[3]) are extracted and quantized based on a group of visual words. Then these features are embedded into hash codes for efficient search.

For this, we consider a hashing technique including semi-supervised hashing and semantic hashing with deep belief networks. Hashing is preferable over tree-based indexing structures as it generally requires greatly reduced memory and also works better for high-dimensional samples. With the hash codes, image matching can be efficiently measured.

In this paper we compute query adaptive weight for each bit of the hash codes. This is the main contribution of this work which has mainly two advantages. Firstly, images can be efficiently ranked on a finer grained hash code level. Second more suitable set of weight is assigned to each query instead of using single set of weight for all the query.

2. Related Works

There are many good surveys of general image search task. Many people choose basic features such as color and texture in the early years [4], while more effective features such as GIST [5] and SIFT [3] have been popular recently [2], [6].

Lowe introduced the Scale-Invariant Feature Transform (SIFT) descriptor [Lowe 1999] in 1999. The basic idea is to extract interesting features from an image sample and be able to compare them to template features, regardless of a change in scale or orientation.

Embedding high-level image features into hash codes has become very popular recently. Hashing satisfies both query time and memory requirements as the binary hash codes are compact in memory and efficient in search via hash table lookup or bitwise operations. Locality Sensitive Hashing (LSH) [10] is one of the most well-known unsupervised hashing methods. Recently, Kulis and Grauman [2] extended LSH to work in arbitrary kernel space, and Chum et al. [10] proposed min-Hashing to extend LSH for sets of features. In [3], Kulis and Darrell proposed a supervised hashing method to learn hash functions by minimizing reconstruction error between original feature distances and Hamming distances of hash codes. In [12], Salakhutdinov and Hinton proposed a method called semantic hashing, which uses deep belief networks [8] to learn hash codes. All these hashing methods (either unsupervised or supervised) have one limitation when applied to image search. The Hamming distance of hash codes cannot offer fine-grained ranking of search results, which is very important.

3. Proposed System

The proposed class based image search system is depicted in fig.1.In this work we design group of classes, each with set of representative images. Low level features of all the images are embedded into hash codes, then we separately compute bitwise weights for each classes.

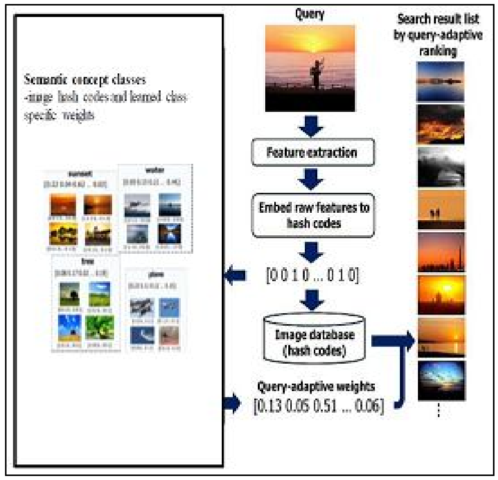

Figure 1:System Architecture

The proposed system works in two parts:

- Offline processing
- Online processing

The flowchart of offline processing is as shown in figure

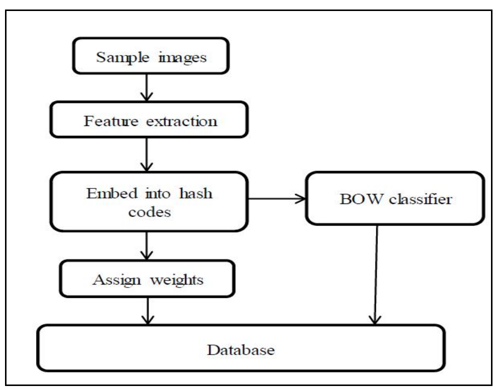

Figure 2: Offline Processing

In offline processing, we have a database that contain set of sample images. The first step in offline processing is to extract image features such as color, texture, saliency, SIFT. Then these features are embedded into hash codes. Images are then assigned tags as per the features and are classified into different classes using clustering. These classes together form Bag-Of-Visual Words (BoW). All this data is stored in the database.

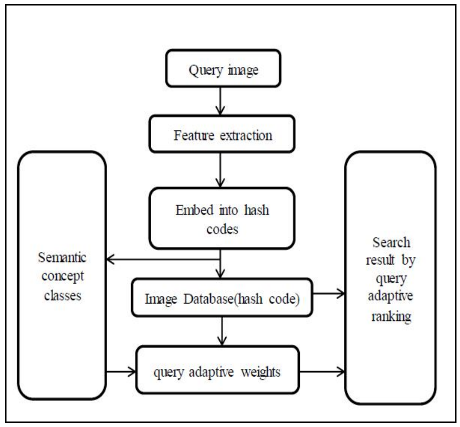

Figure 3: Online processing

In online processing, first step is to extract features from the query or input image. Then these features are embedded into hash code. These hash code along with the assigned weight are compared with the data stored in the database and list of similar images is produced. These images are ranked based on the hamming distance. And thus we get an efficient search result.

In this system first step is to extract features from the images. we use features such as color, texture, saliency and SIFT. Next step is the hash code generation. The hash code is generally smaller than the features itself, and is generated by formula. Therefore, it requires less memory storage. The image features mapped into hamming space using hashing method and then quantized into hash codes. Next is to find the similarity between the query image and the database images. A similarity measure is finding the distance between two images. The distance between two images is calculated using feature vectors that are extracted from the images. Therefore, search result is not a single image, but many images will be retrieved similar to the query image. In this, uses hamming distance for similarity measure. With hash codes of feature vectors, similarity measure can be performed in hamming space using hamming distance. By definition, hamming distance between two hash codes is the total number of bits at which the binary values are different [5].

4. Conclusion

In this paper a novel framework for class based image search with hash code is presented. By harnessing a large set of predefined semantic concept classes, the approach is able to predict query-adaptive bitwise weights of hash codes in real-time, with which search results can be rapidly ranked by a finer-grained hash code level. One can further extend this system by adopting a filter for removing noise from the image which may improve the search quality.

5. References

1. A. Torralba, R. Fergus, and W. Freeman, “80 million tiny images: A large data set for nonparametric object and scene recognition,” IEEETrans. Pattern Anal.Mach. Intell., vol. 30, no. 11, pp. 1958–1970, Nov.2008.
2. J. Sivic and A. Zisserman, “Video Google: A text retrieval approach to object matching in videos,” in Proc. IEEE Int. Conf. Computer Vision, 2003.
3. D. Lowe, “Distinctive image features from scale-invariant keypoints,” Int. J. Comput. Vision, vol. 60, no. 2, pp. 91–110, 2004.
4. J. R. Smith and S.-F. Chang, “VisualSEEk: A fully automated contentbased image query system,” in Proc. ACM Int. Conf.Multimedia, 1996.
5. A. Oliva and A. Torralba, “Modeling the shape of the scene: A holistic representation of the spatial envelope,” Int. J. Comput. Vision, vol. 42, pp. 145–175, 2001.
6. D. Nister and H. Stewenius, “Scalable recognition with a vocabulary tree,” in Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2006.
7. G. Hinton and R. Salakhutdinov, “Reducing the dimensionality of data with neural networks,” Science, vol. 313, no. 5786, pp. 504–507, 2006.
8. R. Datta, D. Joshi, J. Li, and J. Z.Wang, “Image retrieval: Ideas, influences,and trends of the new age,” ACM Comput. Surveys, vol. 40, no.2, 2008.
9. J. L. Bentley, “Multidimensional binary search trees used for associative searching,” Commun. ACM, vol. 18, no. 9, pp. 509–517, 1975.
10. A. Torralba, R. Fergus, and Y. Weiss, “Small codes and large image databases for recognition,” in Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2008.
11. Y. Weiss, A. Torralba, and R. Fergus, “Spectral hashing,” in Adv. Neural Inf. Process. Syst., 2008.
12. J. Wang, S. Kumar, and S.-F. Chang, “Sequential projection learning for hashing with compact codes,” in Proc. Int. Conf.Machine Learning, 2010