Pattern Recognition 47 (2014) 748–757

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Multiple feature kernel hashing for large-scale visual search

Xianglong Liu [a,*], Junfeng He [b,c], Bo Lang [a]

[a] State Key Laboratory of Software Development Environment, Beihang University, Beijing 100191, China
[b] Department of Electrical Engineering, Columbia University, New York, NY 10027, USA
[c] Facebook, 1601 Willow Rd, Menlo Park, CA 94025, USA

ARTICLE INFO

Article history:
Received 26 December 2012
Received in revised form
1 July 2013
Accepted 27 August 2013
Available online 7 September 2013

Keywords:
Locality-sensitive hashing
Multiple features
Compact hashing
Multiple kernels

ABSTRACT

Recently hashing has become attractive in large-scale visual search, owing to its theoretical guarantee and practical success. However, most of the state-of-the-art hashing methods can only employ a single feature type to learn hashing functions. Related research on image search, clustering, and other domains has proved the advantages of fusing multiple features. In this paper we propose a novel multiple feature kernel hashing framework, where hashing functions are learned to preserve certain similarities with linearly combined multiple kernels corresponding to different features. The framework is not only compatible with general types of data and diverse types of similarities indicated by different visual features, but also general for both supervised and unsupervised scenarios. We present efficient alternating optimization algorithms to learn both the hashing functions and the optimal kernel combination. Experimental results on three large-scale benchmarks CIFAR-10, NUS-WIDE and a-TRECVID show that the proposed approach can achieve superior accuracy and efficiency over state-of-the-art methods.

1. Introduction

Recent years have witnessed the explosive growth of the vision data, which brings great challenges to the scalable visual search. Among the vast solutions like KD-Tree [1], hash-based approximate nearest neighbor (ANN) search has drawn much attention, owing to its promising performance in many applications like pose estimation [2], image search [3], active learning [4], etc. Locality-Sensitive Hashing (LSH) [5] pioneered the hash-based ANN research by introducing the important concept named "locality sensitivity" that similar data points are hashed into similar codes. With the embedded hash codes, search can be achieved in a constant or sub-linear time. Following the idea, Charikar [6] proposed the random projection based LSH that efficiently produces hash bits and preserves cosine similarities. Datar et al. [7] further studied p-stable distribution based LSH for similarity metric like l_p-norm ($p \in (0, 2]$). In the past few years, many well-learned hashing methods have been proposed to solve the visual search problems under various scenarios like unsupervised [3,8], (semi-)supervised [9–11], kernelized [12], multi-probes [13], and multi-bits [14].

Despite the aforementioned progress in hash-based visual search, most of the state-of-the-art hashing methods suffer from the limitation that hashing functions are usually learned only from a single feature. However, in practice multiple representations exist for the same visual object: images can be described by various different visual descriptors such as SIFT [15], GIST [16], and so on. It has been proved in many applications that adaptively combining a set of diverse and complementary features gives better performance than using a single one. For instance, the content-based image retrieval systems gain significant performance improvements by fusing multiple features like color and texture [17–19]. Feature combination is also very helpful in other domains like image classification [20] and clustering [21].

Existing multiple feature hashing approaches [17,22] also have achieved encouraging performance gains by fusing different features. However, to handle multiple features, these methods either equally pre-concatenate all features as one or post-combine the linear outputs of each feature type. On one side, equally treating all features will not fully exploit the correlation and importance of each feature type. It is well-known that different features may convey unbalanced and different information, and some of them are complementary to each other under different similarity measures [22]. On the other hand, feature concatenation expands the feature dimension, and thus brings expensive computation at both training and searching stages (see detailed discussion in Section 6.2). Therefore, it is impractical to apply these approaches to large-scale image search with a number of high-dimensional visual features.

In this paper, we propose a novel hashing framework to learn informative hashing functions that incorporate multiple features and preserve neighbor relations under a certain similarity metric. In this framework, different features are non-linearly, yet implicitly mapped into and concatenated in high dimensional spaces, and then the hashing functions with multiple features can be formulated as projections using linearly combined multiple

* Corresponding author. Tel.: +86 1082338094; fax: +86 1082316736.
E-mail addresses: xlliu@nlsde.buaa.edu.cn, xlliuchina@gmail.com (X. Liu).

0031-3203/$ - see front matter
http://dx.doi.org/10.1016/j.patcog.2013.08.022

kernels. Kernel tricks are often more natural to gauge the similarity of general data types, where the underlying data embedding to the high-dimensional space is not known explicitly. Many hashing methods benefit from the use of domain specific kernel functions [9,12,23]. In our formulation, a set of kernels corresponding to different features are automatically weighted and linearly combined as one. This is very similar to the popular method in computer vision named Multiple Kernel Learning (MKL) [24], which has shown to be able to learn a better reproduced kernel space and improve the performance.

Owing to the kernel representation, the combination of multiple embedded features in corresponding kernel spaces does not bring more computation than a single feature type. Given the similarities among data, we efficiently and alternately optimize the hashing functions and the kernel combination using eigen-decomposition and quadratic programming respectively. It is worth highlighting the main contributions of this paper:

1. We formulate the multiple feature hashing as a similarity preserving problem in a generic framework for both supervised and unsupervised cases. With all features mapped into and concatenated in kernel spaces, each learned hashing function is formed as a linear projection with respect to a combination of multiple kernels. The formulation is compatible with general types of data with any kernel function and helps archive fast computation.
2. For the supervised scenario, the proposed method incorporates different features to preserve the semantic similarity by exploiting the complementarity among features. Without any supervised information, it can also be adapted to learn the discriminant hashing functions by recovering the low-rank affinity behind diverse similarities of different features.
3. For both scenarios, we respectively propose an alternating optimization way to efficiently learn both the hashing functions and the kernel combination coefficients.
4. Extensive experiments show the superior performance and efficiency of the proposed approach with compact hash codes (usually less than 64 for fast computation and storage efficiency) over state-of-the-art methods.

Note that the whole paper extends upon a previous conference publication [25] with additional exploration on the algorithm generalization (both supervised and unsupervised cases), detailed analysis of the algorithmic properties (connections with other algorithms, the computational complexity and the parameter sensitivity), and amplified experimental results. The rest of the paper is organized as follows: we provide a short review on related work in Section 2. Section 3 introduces the basic idea and the formulation for our multiple feature kernel hashing. Then in Sections 4 and 5 we will respectively present the algorithms for the supervised and unsupervised hashing based on the given formulation. The computational problem evolved in our method is analyzed in Section 6. Section 7 describes our evaluation settings and the experimental results. Finally, we conclude in Section 8.

2. Related work

In the past few years, hashing has become a promising approach to address the approximate nearest neighbor search in many applications like large-scale visual search. Locality-sensitive hashing (LSH) [5] as the well-known pioneering work embeds similar data under specific metric into similar binary codes in Hamming space with high probabilities. With these binary codes, multiple hash tables can be built to accomplish sub-linear search. Since the projection vectors are randomly generated in LSH, much following research has been devoted to learning compact projections under different scenarios, including unsupervised hashing [3,8,12], (semi-)supervised hashing [10,11], and nonlinear hashing with kernels [9,12,23,26] for general data types.

Spectral Hashing (SH) [3] attempts to encode data into compact binary codes, preserving the original similarities in Hamming space. It formulates the hashing problem as a graph partition problem, with the basic requirements of good binary codes: the bit balance and uncorrelation

$$\min_{Y} \sum_{i,j} S_{ij} \| Y_i - Y_j \|^2 \quad \text{s.t.} \quad Y_i \in \{-1,1\}^P, \ \sum_i Y_i = 0, \ \frac{1}{N}\sum_{i,j} Y_i Y_j^T = I. \tag{1}$$

Here S_{ij} is the similarity between the i-th and j-th samples, and N and P are respectively the numbers of data samples and bits for each sample. Y_i, the i-th column of binary matrix Y, is the hash code of the i-th sample.

The original problem with discrete constraints, equivalent to the balanced graph partitioning problem, is NP-hard. However, after relaxing the constraints, it turns to be an eigen-decomposition problem of a graph Laplacian matrix, which can be solved efficiently. For a novel sample, SH depends on the uninform distribution assumption which may be not true in real world. Moreover, its data similarity is strictly defined in the original feature space, which limits its power. To address these problems, He et al. [12] propose an optimal kernel hashing (OKH) that explicitly defines the hash functions based on kernels. In their formulation, there is no data distribution assumption, and the learned hash functions can be directly applied to novel input samples of generic types.

In many vision problems multiple feature fusion has been proved very helpful [18,20]. But there is no much research incorporating multiple features except [22] and [17]. Both methods formulate the hashing problem with multiple features based on SH. In [22] the hash output is the convex output combination of all linear hash functions on different sources, while [17] concatenates all features as one and projects it using the learned hashing hyperplane. The feature concatenation leads to expensive computation costs and the poor generalization for various data types and similarity measures.

Another related work is the multiple kernel learning proposed recently [27]. It combines multiple kernels, corresponding to different similarities or information from different sources, instead of using a single one, and tries to find which works best by different learning methods. In previous research, several combination ways have been

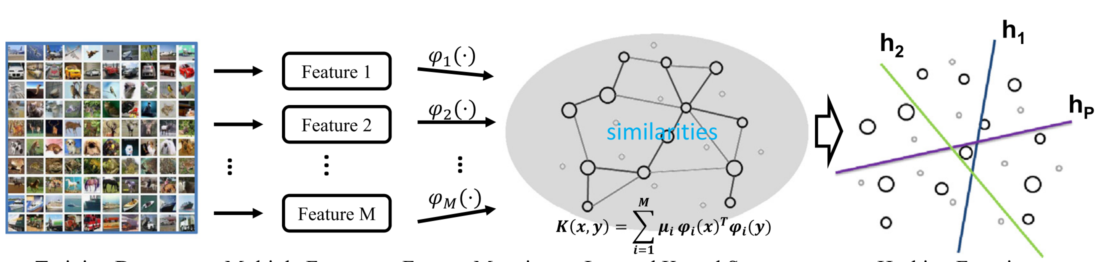

Fig. 1. Illustration of the proposed multiple feature kernel hashing framework for both supervise and unsupervised hashing.