ARTICLE IN PRESS

Neurocomputing xx (xxxx) xxxx–xxxx

Contents lists available at ScienceDirect

Neurocomputing

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

Large-scale image retrieval with supervised sparse hashing

Yan Xu[a], Fumin Shen[a,b,*], Xing Xu[a], Lianli Gao[a], Yuan Wang[c], Xiao Tan[d]

[a] School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, PR China
[b] Jiangsu Key Laboratory of Image and Video Understanding for Social Safety (Nanjing University of Science and Technology), Nanjing, PR China
[c] Department of Industrial and Systems Engineering, National University of Singapore, Singapore
[d] The University of Hong Kong, Hong Kong

ARTICLE INFO

Keywords:
Learning based hashing
Medical
Sparsity
Image retrieval

ABSTRACT

In recent years, learning based hashing becomes an attractive technique in large-scale image retrieval due to its low storage and computation cost. Hashing methods map each high-dimensional vector onto a low-dimensional hamming space by projection operators. However, when processing high dimensional data retrieval, many existing methods including hashing cost a majority of time on projection operators. In this paper, we solve this problem by implementing a sparsity regularizer. On one hand, due to the sparse property of the projection matrix, our method effectively lower both the storage and computation cost. On the other hand, we reduce the effective number of parameters involved in the learned projection matrix according to sparsity regularizer, which helps avoid overfitting problem. Without relaxing binary constraints, an iterative scheme jointly optimizing the objective function directly was given, which helps to obtain effective and efficient binary codes. We evaluate our method on three databases and compare it with some state-of-the-art hashing methods. Experimental results demonstrate that our method outperforms the comparison approaches.

1. Introduction

As the explosive growing of images on the Internet, nearest neighbor (NN) search becomes a very popular method in recent years. It has been applied in various applications such as retrieval, classification, object matching, visual tracking and related areas [10,29,13,46,30,50,9]. In modern medical field, NN search also plays an important role, such as mammogram analysis [18,19] and medical image retrieval [26,33]. When the data are high-dimensional, due to the large storage and time consumption, general NN search becomes inefficient. To solve this problem, a great number of retrieval approaches are proposed [36,32,27,59]. Hashing based method has attracted much attention because of its ability of storage and time-saving computation. hashing based method leads to an efficient search and solves scalability which is result from the growth of databases.

Hashing based methods map the high-dimensional, real-valued vector onto a low-dimensional, binary vector, and the generated binary codes are employed to efficient search. Existing hashing methods can be roughly classified into two categories: *data-independent* [2,5,16,37] and *data-dependent* [11–13,21,34,48]. Locality Sensitive Hashing (LSH) [10] is one of the most popular *data-independent* method among various existing hashing techniques. In LSH, the hash functions are generated according to random projections. Hence these hash functions are independent of any training data. The LSH family has been constantly developed based on different similarity and distance measures, including the Euclidean distance, p-norm distance [6], Mahalanobis distance [23], kernel similarity [22,37] and so on. LSH and it various achieve both good precision and recall. However, since LSH is a pure *data-independent* approach, either it needs long binary hash bits or multiple hash tables, which requires large memory consumption. Thus the LSH family may be restricted in many practical applications.

To remedy *data-independent* techniques, *data-dependent* hashing methods have been proposed to learn similarity-preserving binary codes from training data. This kind of hashing methods is also known as *learning-based* methods in the literature. *Data-dependent* hashing techniques preserve similarity between original data space and binary data space. Compared with the *data-independent* methods, *data-dependent* methods achieve equivalent results with a small code size. Representative *data-dependent* methods can be roughly divided into two parts: *supervised* methods (including *semi-supervised* methods), and *unsupervised* methods. Unsupervised methods use unlabelled data to learn binary codes. For example, Iterative Quantization (ITQ) [12], Spectral Hashing (SH) [52], Anchor Graph Hashing (AGH) [29],

* Corresponding author at: School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, PR China.
E-mail addresses: xuyan5533@gmail.com (Y. Xu), fumin.shen@gmail.com (F. Shen), xing.xu@uestc.edu.cn (X. Xu), lianli.gao@uestc.edu.cn (L. Gao), iseway@nus.edu.sg (Y. Wang), tanxchong@gmail.com (X. Tan).

http://dx.doi.org/10.1016/j.neucom.2016.05.109
Received 16 January 2016; Received in revised form 9 May 2016; Accepted 11 May 2016
Available online xxxx

Inductive Manifold Hashing(IMH) [41,42], Asymmetric Inner Product Binary Coding (AIBC) [39], Sparse Projection (SP) [53] *etc*., are some widely accepted methods. Unsupervised approaches seek to learn hashing functions from underlying data structures, distributions, or topological information. Differently, supervised methods consider the supervised information and the inherent properties of data simultaneously, *e.g.*, Semi-supervised spectral hashing [54], Kernel-based supervised hashing (KSH) [28], Ranking-based Supervised Hashing (RSH) [49], Supervised Discrete Hashing (SDH) [40], Minimal Loss Hashing (MLH) [34], *etc*. Depart from most these methods focusing generic image retrieval, a few works [17,57,58] have been devoted to tackle the large-scale medical image retrieval problem.

Recently, a lot of works related to deep neural networks (DNN) [20,55] have been proposed and show that features more than thousands of dimensions are useful for recognition tasks. Generally, many proposed hashing methods are based on traditional hand-crafted features, like GIST [35], SIFT [31], *etc*. However DNN features are learned from large-scale data and thus are not structured. On the other hand, generating binary codes, *i.e.*, binary encoding, uses a projection matrix from some d-dimensional inputs to some corresponding L-dimensional outputs. If d and L are large, the computational efficiency and storage cost of calculating projection matrix become a bottleneck.

In this paper, we propose a novel supervised hashing method which leverages sparse projection. A sparse regularizer [53] is introduced to the objective function which restricts the sparsity of projection matrix *via* controlling the number of non-zero coefficients directly. This operation not only leads to a drop of the computational cost but reduces the effective number of parameters. We also add supervised information in the objective function in order to sufficiently utilize label information which preserves semantic similarity. When implementing binary optimization, we adopt a *discrete cyclic coordinate descent* (DCC) algorithm [40] to produce binary codes bit by bit. The DCC algorithm generates the optimal binary codes in a closed form, which makes the generated hash bits effectively and efficiently. Our contributions are summarized as follows.

1. We propose a new supervised hashing method by introducing a sparse regularizer to generate sparse projection matrix which can effectively reduce memory cost and embedding time.
2. We adopt the variable-splitting to reduce each sub-optimization problem with only one constraint condition at most, which is easy to be solved. We also show that direct optimization without any relaxation achieves high quality hash codes and leads to better results.

We demonstrate our method on three databases (**ImageNet** [7], **CIFAR-10**[1] and **DDSM** [1,14]) and show that our method leads to better accuracy than some existing competitive hashing methods, including three dense projection hashing methods, *i.e.*, CCA-ITQ [12], KSH [28], SDH [40] and one sparse projection hashing method, *i.e.*, SP [53].

2. Supervised sparse projection for binary encoding

Firstly, we define the notation used in our formulation. We are given a set of training data points $X = \{x_1, \ldots, x_n\}$ in which each point is d-dimensional, *i.e.*, $x \in \mathbb{R}^d$, and we assume that all the training data points are zero-centered, *i.e.*, $\sum_{i=1}^{n} x_i = 0$. Our target is to use a sparse projection matrix $R \in \mathbb{R}^{L\times d}$ to learn a set of binary codes $B = \{b_i\}_{i=1}^{n} \in \{-1, 1\}^{L\times n}$, where L is the binary code length. We can write it as following:

$$H(X) = sgn(RX) s.\ t.\ |R|_0 \le m \tag{1}$$

[1] http://www.cs.toronto.edu/kriz/cifar.html

Here sgn $(\cdot)$ denotes sgn function, which means +1 for positive numbers and −1 for other numbers. $|\cdot|_0$ denotes the number of non-zero elements of the matrix, and m controls the sparsity of the projection matrix.

After defining the notation and hash function, we focus on proposing an objective function to learn sparse projection R. Let b_i denotes the binary representation of i-th data point, where $i = 1, \ldots, n$. The basic binary encoding scheme is to quantize the fitting error of the binary codes b_i to the continuous embedding Rx_i. We conclude above method by following:

$$\min \sum_{i=1}^{n} \|Rx_i - b_i\|^2 s.\ t.\ |R|_0 \le m \quad b_i = \{-1, 1\}^L \tag{2}$$

We can rewrite above as a matrix form. That is,

$$\min \|RX - B\|_F^2 s.\ t.\ |R|_0 \le m \quad B = \{-1, 1\}^{L\times n} \tag{3}$$

Here $\|\cdot\|_F$ denotes the Frobenius norm. This problem is similar with ITQ [12] but replacing the orthogonal constraint with a sparsity constraint.

To sufficiently utilize the label information, we use linear classification framework to solve binary codes learning problem. We adopt the multi-class classification $y = W^T b$, where $W \in \mathbb{R}^{L\times c}$, each column of the matrix W denotes the classification vector for different class and $y \in \mathbb{R}^c$ denotes label vector, which should have a maximun element in the corresponding class.

By adding the label information loss as the penalty term in (3), we can get the following problem:

$$\min_{B,R,W} \|RX - B\|_F^2 + \alpha(\|Y - W^T B\|_F^2 + \lambda\|W\|_F^2) s.\ t.\ |R|_0 \le m \quad B = \{-1, 1\}^{L\times n} \tag{4}$$

Where $\alpha > 0$ is the penalty parameter which controls the influence of label information. When $\alpha = 0$ implies that the problem (4) has degraded into unsupervised problem. With increasing of α, label information plays an important role in learning effective and efficient binary codes. λ in problem (4) is the regularization parameter. The overview of the proposed method is shown in Fig. 1.

Because of the l_0 constraint, the problem (4) is non-convex. We can't solve it directly. One alternative method is relaxing the l_0-regularization to l_1-regularization, *i.e.*, $|R|_1 \le m$, where $|\cdot|_1$ means the sum of the absolute values of the elements. In [38], the authors regard projection matrix R as a classifier that predicts the binary labels. Therefore, they rewrite their objective function as a max-margin l_1-regularized linear classifier and address the problem by using LibLinear [8]. However, in our method, we don't replace the l_0-regularization to other forms. On one hand, we consider that the l_0-regularization can directly control sparsity rate of projection matrix, while the l_1 case and other regularizations don't have this easy control, on the other hand, we will show that our method with l_0-regularization leads to a simply optimization progress without too much loss of information.

3. Optimization

It is a challenge to solve (4) directly due to the sparsity constraint. Like in [51,53], we adopt the variable-splitting and penalty techniques in optimization by introducing a new variable $\overline{R}$ and put sparsity constraint on it, and meanwhile we penalty the error between $\overline{R}X$ and RX.

We rewrite (4) as following:

$$\min_{B,R,W,\overline{R}} \|RX - B\|_F^2 + \beta\|RX - \overline{R}X\|_F^2 + \alpha(\|Y - W^T B\|_F^2 + \lambda\|W\|_F^2) s.\ t.\ |\overline{R}|_0 \le m \quad B = \{-1, 1\}^{L\times n} \tag{5}$$

Here β is a penalty term The problem is similar to Half-Quadratic Splitting [51]. We solve the problem (5) in an iterative fashion: