To Project More or to Quantize More: Minimizing Reconstruction Bias for Learning Compact Binary Codes

Zhe Wang[1,2], **Ling-Yu Duan**[1], **Junsong Yuan**[2], **Tiejun Huang**[1], **Wen Gao**[1]
[1]Institute of Digital Media, Peking University, Beijing, China
[2]Rapid-Rich Object Search (ROSE) Lab, Nanyang Technological University, Singapore.
{zhew,lingyu,tjhuang,wgao}@pku.edu.cn, {jsyuan}@ntu.edu.sg

Abstract

We present a novel approach called Minimal Reconstruction Bias Hashing (MRH) to learn similarity preserving binary codes that jointly optimize both projection and quantization stages. Our work tackles an important problem of how to elegantly connect optimizing projection with optimizing quantization, and to maximize the complementary effects of two stages. Distinct from previous works, MRH can adaptively adjust the projection dimensionality to balance the information loss between projection and quantization. It is formulated as a problem of minimizing reconstruction bias of compressed signals. Extensive experiment results have shown the proposed MRH significantly outperforms a variety of state-of-the-art methods over several widely used benchmarks.

1 Introduction

Approximate nearest neighbour (ANN) search [Gionis *et al.*, 1999a] plays an important role in machine learning, computer vision and information retrieval. Using similarity preserving binary codes to represent original data points is of particular interest for ANN search [Weiss *et al.*, 2008; Norouzi and Fleet, 2011]. The binary codes can bring about low memory cost as well as fast similarity distance computing speed. This is particular useful when dealing with large scale database [Torralba *et al.*, 2008; Gong and Lazebnik, 2011; Weiss *et al.*, 2008; Duan *et al.*, 2016].

A common binary coding approach, often called Hashing, is to develop similarity preserving hashing functions for mapping data points into a Hamming space. As it is NP-hard to directly learn the optimal binary codes [Weiss *et al.*, 2008], hashing methods typically work on a two-stage strategy: projection and quantization [Kong and L, 2012; Kong *et al.*, 2012]. Specifically, given a data point $\mathbf{x} \in \mathbb{R}^d$, they first project $\mathbf{x}$ into a low dimensional vector

$$\mathbf{y} = [\mathrm{f}_1(\mathbf{x}), \mathrm{f}_2(\mathbf{x}), ..., \mathrm{f}_k(\mathbf{x})] \in \mathbb{R}^k,$$

where real-valued functions $\{\mathrm{f}_i(\cdot)\}_{i=1}^k$ are called projection functions. Then they utilize Single Bit Quantization (SBQ) to quantize each projection element $\mathrm{f}_\mathrm{i}(\mathbf{x})$ into a single bit by thresholding [Kong and L, 2012; Wang *et al.*, 2016].

Lots of research efforts have been devoted to the first stage, with an aim to learn powerful projections to maintain the similarity structure of the original data points. Local Sensitive Hashing (LSH) [Andoni and Indyk, 2006] adopts a random projection which is independent of training data. Similarly, Shift Invariant Kernel Hashing (SIKH) [Raginsky and Lazebnik, 2009] chooses random projection and applies shifted cosine function to generate binary codes. Both LSH and SIKH are data independent and flexible since they do not rely on any training data. However, long codes are often required to achieve satisfactory performance [Gong and Lazebnik, 2011; Raginsky and Lazebnik, 2009].

To build up more effective projection, many promising data dependent methods have been proposed. Through learning the projection functions over training data, data dependent methods usually outperform data independent methods at relatively shorter codes [Liu *et al.*, 2010]. Representative methods include Spectral Hashing [Weiss *et al.*, 2008], Binary Reconstructive Embedding Hashing [Kulis and Darrell, 2009], Semi-Supervised Hashing [Wang *et al.*, 2010], Anchor Graph Hashing [Liu *et al.*, 2010], Iterative Quantization [Gong and Lazebnik, 2011], Minimal Loss Hashing [Norouzi and Fleet, 2011], Kernal Supervised Hashing [Liu *et al.*, 2012], Isotropic Hashing [Kong and Li, 2012], K-means Hashing [He *et al.*, 2013], Inductive Hashing on Manifolds [Shen *et al.*, 2013], Harmonious Hashing [Xu *et al.*, 2013], Discrete Graph Hashing [Liu *et al.*, 2014], Sparse Projection Hashing [Xia *et al.*, 2015], etc.

Moreover, recent works have reported the impact of quantization on Hashing performance. Single Bit Quantization (SBQ) in most hashing methods incurs lots of quantization errors, which would seriously degrade the performance [Kong and L, 2012; Kong *et al.*, 2012]. Thus, promising Multiple Bits Quantization (MBQ) methods have been proposed. Double Bits Quantization [Kong and L, 2012] divides each projection dimension into three regions and uses double bits code to represent each element region. Manhattan Quantization [Kong *et al.*, 2012] proposes natural binary code (NBC) and adopts Manhattan distance to compute the distance between NBC codes. Hamming Compatible Quantization [Wang *et al.*, 2015] aims to minimize the distance error function to preserve the capability of similarity metric between Euclidean space and Hamming space. Overall, MBQ methods do facilitate the reduction of information

loss in quantization. Experiment results have demonstrated the functionality of high quality quantization in improving Hashing performance [Kong and L, 2012; Kong *et al.*, 2012; Wang *et al.*, 2015].

Hence, both projection and quantization are important. Optimal binary codes rely on the joint optimization of projection and quantization. However, how to elegantly connect optimizing projection with optimizing quantization, and to maximize the complementary effects of two stages, remains a challenging problem in practice. Given a specified binary code length k, shall we project more or quantize more?

The projection dimensionality in existing hashing methods or quantization methods are all fixed, which is not flexible. An intuitive thought is, when the original data points inherently lie in a low dimensional space, we may project data points to a lower-dimensional space (project more), while performing finer quantization for each element by using multiple bits instead of a single bit (quantization less). For example, we may project original data points into space $\mathbb{R}^{\frac{k}{2}}$ and assign 2 bits to quantize the values of each projection element, while the target code length remains as k. Through adaptively adjusting the projection dimensionality, we may minimize the information loss over the whole course of Hashing, in which the balance between projection and quantization can be addressed by the optimal projection dimensionality.

In this paper, we propose a novel approach Minimal Reconstruction Bias Hashing (MRH) to tackle the joint optimization of projection and quantization. We summarize the main contributions of this paper as follows:

- We present a novel approach to learn similarity preserving binary codes which jointly optimizes both projection and quantization stages with adjustable projection dimensionality. To the best of our knowledge, this is the first work that can adaptively adjust the projection dimensionality to balance the information loss between projection and quantization. Our practice of jointly optimizing projection dimensionality, projection matrix, as well as quantization functions, has achieved the state-of-the-art performance over several benchmarks.
- We reinterpret the problem of maximizing the similarity preserving in Hashing from the perspective of minimal reconstruction bias of signals. By introducing a lower bound analysis, we establish the relationship between the information loss from projection and quantization, and the Hamming approximation errors, which justifies the learning objective of our proposed MRH method.
- By analyzing the unimodal characteristics of the MRH objective function with respect to projection dimensionality, we propose an effective solution to resolve the joint optimization problem of MRH. In particular, we have reduced the complexity of searching optimal projection dimensionality from $\mathrm{O(N)}$ to $\mathrm{O(log(N))}$.

2 Preliminaries

We firstly introduce the basic notations. Let matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$ denote the samples of data points, and k denote the length of target codes. The goal is to learn a binary string $\mathbf{b}_i \in \{0,1\}^k$ for each data point $\mathbf{x}_i \in \mathbb{R}^d$ with the maximized similarity preservation in Hamming space.

The proposed binary code learning approach involves both projection and quantization stages. At the projection stage, we adopt linear projection to transform $\mathbf{x}_i \in \mathbb{R}^d$ into a subspace

$$\mathbf{y}_i = \mathrm{T}(\mathbf{x}_i) \in \mathbb{R}^{\frac{k}{c}},$$

where $\mathrm{T}(\mathbf{x}_i) = \mathbf{R}\mathbf{x}_i$ and $\mathbf{R} \in \mathbb{R}^{\frac{k}{c}\times d}$. The projection matrix $\mathbf{R}$ is required to be orthogonal, i.e., $\mathbf{R}\mathbf{R}^\top = \mathbf{I}$, to make the projections of a vector in different dimensions independent of each other. At the quantization stage, we quantize projection vector $\mathbf{y}_i$ into

$$\widehat{\mathbf{y}}_i = \mathrm{Q}(\mathbf{y}_i) \in \mathcal{H}^{\frac{k}{c}},$$

where $\mathcal{H}$ is the set of quantization centroid(s), and each element in $\mathbf{y}_i$ is quantized to a value in $\mathcal{H}$. Finally, we use c bits to encode each element of $\widehat{\mathbf{y}}_i$ to obtain a binary string

$$\mathbf{b}_i = \mathrm{B}(\widehat{\mathbf{y}}_i) \in \{0,1\}^k.$$

Note that we introduce variable c to adjust the projection dimensionality. If the given code length k is indivisible by c, the target code length will round down to $\lfloor \frac{k}{c} \rfloor \times c$ bits. In this paper, we will figure out an optimal c value to make a joint optimization of projection $\mathrm{T}(\cdot)$ and quantization $\mathrm{Q}(\cdot)$.

The range of Hamming distance is limited to the length of a binary code. The maximum Hamming distance of c bits codes is only c. When we use c bits to encode 2^c values, the distance consistency in the Hamming space cannot maintained. Let us take the example of $c = 2$. We quantize the projection values into $2^2 = 4$ centroid(s) with $\sigma_0 < \sigma_1 < \sigma_2 < \sigma_3$, namely, $(00)_2$, $(01)_2$, $(10)_2$ and $(11)_2$. We have $\|\sigma_1 - \sigma_2\| < \|\sigma_1 - \sigma_3\|$, but $\mathrm{d_H}(01, 10) > \mathrm{d_H}(01, 11)$ where $\mathrm{d_H}(.)$ denotes the Hamming distance.

To address the issue of inconsistent measurements, we may resort to other distance measurer like Manhattan distance [Kong *et al.*, 2012]. But this would seriously degrade the retrieval efficiency [Wang *et al.*, 2015]. By contrast, Hamming distance measurement is extremely fast, and more than 10^9 operations can be done per second [He *et al.*, 2013; Weiss *et al.*, 2008], so that Hamming distance computing is still the priority of effective and efficient ANN search. In this work, we employ an incomplete encoding strategy to keep the distance consistency in Hamming space, in which we only quantize projection values into $c + 1$ equidistant centroid(s) $\mathcal{H} = \{\sigma_i\}_{i=0}^{c}$, where $\sigma_i - \sigma_{i-1} = \Delta$ for $1 \le i \le c$. Then, we apply unary representation [Gionis *et al.*, 1999b] to encode each centroid $\mathrm{B}(\sigma_i) = \mathrm{U}_c(i)$ for $\sigma_i \in \mathcal{H}$, where unary representation $\mathrm{U}_c(i)$ is defined as a c bits binary string with i ones followed by $c - i$ zeros, e.g. $\mathrm{U}_2(1) = 10$, $\mathrm{U}_3(0) = 000$, $\mathrm{U}_4(2) = 1100$. With the unary representation, the Hamming distance between binary codes is proportional to the distance of the centroid(s),

$$\mathrm{d_H}(\mathrm{B}(\sigma_i), \mathrm{B}(\sigma_j)) = \|\sigma_i - \sigma_j\|/\Delta.$$

Clearly, unary representation is an incomplete encoding method, as a c bits code can represent 2^c states. To make the Hamming distance consistent with the distance of quantization centroid(s), we have to discard parts of code space.

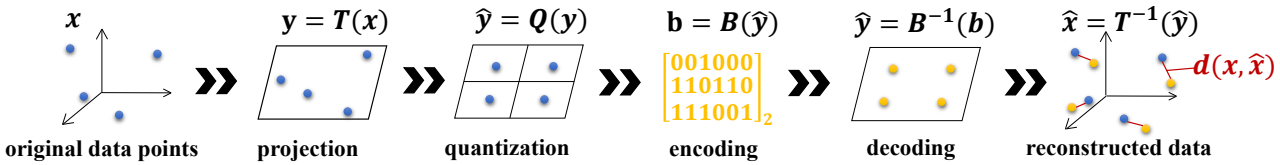


Figure 1: Illustration of reconstruction bias from projection and quantization. Towards optimal binary coding, the aim is to minimize this bias. The red lines indicate the reconstruction bias. This figure is best viewed in color version.

3 Minimal Reconstruction Bias Hashing

We formulate the problem of optimal binary coding (i.e., optimal hashing) from the perspective of minimizing the reconstruction bias of signals [Allen and Gray, 2012]. The relationship between minimal reconstruction bias and Hamming approximation errors will be studied as well.

3.1 Reconstruction Bias

The reconstructed data points are recovered from the compressed codes. Given the hashing code $\mathbf{b}_i$ of data point $\mathbf{x}_i$, we obtain the reconstructed data by first decoding $\mathbf{b}_i$ to quantization centroid(s) and then transforming the quantization vector back to the original space (see Fig.1). Specifically, we first decode $\mathbf{b}_i$ and get $\widehat{\mathbf{y}}_i = \mathrm{B}^{-1}(\mathbf{b}_i)$. Quantization function $\mathrm{Q}(.)$ is not invertible, so $\mathbf{y}_i$ can't be recovered. Then, we directly apply the inverse projection transformation to $\widehat{\mathbf{y}}_i$ and get

$$\widehat{\mathbf{x}}_i = \mathrm{T}^{-1}(\mathrm{B}^{-1}(\mathbf{b}_i)) = \mathbf{R}^\top \widehat{\mathbf{y}}_i. \tag{1}$$

Here, $\widehat{\mathbf{x}}_i$ is called the reconstructed data of $\mathbf{x}_i$. The reconstruction bias is defined as the distance between $\widehat{\mathbf{x}}_i$ and $\mathbf{x_i}$

$$\mathrm{d}(\mathbf{x}_i, \widehat{\mathbf{x}}_i) = \|\mathbf{x}_i - \widehat{\mathbf{x}}_i\|_2 = \|\mathbf{x}_i - \mathbf{R}^\top \widehat{\mathbf{y}}_i\|_2 \tag{2}$$

where $\mathrm{d}(\cdot)$ denotes the Euclidean distance, $\|\cdot\|_2$ denotes the L2 norm of a vector.

3.2 Learning Objective

The reconstruction bias indicates the information loss incurred by mapping the data points into Hamming space. To preserve the similarity structure of original data points, we aim to minimize the reconstruction bias. Directly optimizing the objective function in Eqn.2 is intractable due to a large number of free parameters in $\widehat{\mathbf{y}}_i$ and the orthogonal constraint $\mathbf{R}$. Hence, we propose to relax $\mathrm{d}(\mathbf{x}_i, \widehat{\mathbf{x}}_i)$ as

$$\|\mathbf{x}_i - \mathbf{R}^\top \widehat{\mathbf{y}}_i\|_2 \leq \|\mathbf{x}_i - \mathbf{R}^\top \mathbf{y}_i\|_2 + \|\mathbf{R}^\top(\mathbf{y}_i - \widehat{\mathbf{y}}_i)\|_2. \tag{3}$$

The first term indicates the distortions by transforming $\mathbf{y}_i$ back to $\mathbf{x}_i$. When $\mathbf{R}$ is orthogonal, for any vector $\mathbf{a}$, we have[1] $\|\mathbf{R}^\top \mathbf{a}\|_2 = \|\mathbf{a}\|_2$. So the second term actually indicates the mean square error (MSE) of $\widehat{\mathbf{y}}_i$. Let $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_n]$ denotes the projection matrix and $\widehat{\mathbf{Y}} = \mathrm{Q}(\mathbf{Y})$. To resolve optimal binary coding, we formulate the problem of jointly minimizing the projection distortions and the quantization errors, in which the projection dimensionality may be variable as well. Specifically, the learning objective is formulated as

$$\begin{aligned} &\min_{\mathrm{c},\mathbf{R},\widehat{\mathbf{Y}}} \|\mathbf{X} - \mathbf{R}^\top \mathbf{Y}\|_F^2 + \|\mathbf{Y} - \widehat{\mathbf{Y}}\|_F^2, \\ &s.t.\ 1 \leq c \leq k,\ \mathbf{R} \in \mathbb{R}^{\frac{k}{c} \times d},\ \mathbf{R}\mathbf{R}^\top = \mathbf{I} \\ &\mathbf{Y} = \mathbf{R}\mathbf{X},\ \widehat{\mathbf{Y}} \in \mathcal{H}^{\frac{k}{c} \times n}, \|\mathcal{H}\| = c + 1. \end{aligned} \tag{4}$$

where $\|\cdot\|_F$ denotes the Frobenius norm, $\|\mathbf{X} - \mathbf{R}^\top \mathbf{Y}\|_F^2$ denotes the sum of projection distortions, indicating the information loss in the projection stage, and $\|\mathbf{Y} - \widehat{\mathbf{Y}}\|_F^2$ denotes the sum of mean square error (MSE), indicating the information loss in the quantization stage. In particular, variable c is to adjust the projection dimensionality to adaptively balance the information loss between the projection and quantization stages in a joint optimization.

3.3 Relationship to Hamming Approximation

Similarity preserving hashing methods aim to map close data points to near binary codes [Gionis *et al.*, 1999a; Andoni and Indyk, 2006]. Conversely, if two data points are far away in the original space, their binary codes should produce a large Hamming distance. We will show that the distance approximation error between the original distance and the root mean square Hamming distance is a lower bound of the learning objective in Eqn.4. Since the Hamming approximation quality, as a critical indicator, can significantly impact the performance of ANN search [Kulis and Darrell, 2009; He *et al.*, 2013], this lower bound analysis may justify the rationale of the proposed learning objective.

In hashing methods, the similarity of two data points $\mathbf{x}_i$ and $\mathbf{x}_j$ is defined by the Hamming distance of their hashing codes, $\mathrm{d}_H(\mathbf{b}_i^k, \mathbf{b}_j^k)$, where $\mathbf{b}_i^k = \mathrm{B}(\widehat{\mathbf{y}}_i^k)$, $\mathbf{b}_i^k$ and $\widehat{\mathbf{y}}_i^k$ denote the k-th element in vector $\mathbf{b}_i$ and $\widehat{\mathbf{y}}_i$, respectively. Let l denote the projection dimensionality, s_k the Hamming distance of the k-th hashing codes where $s_k = \mathrm{d}_H(\mathbf{b}_i^k, \mathbf{b}_j^k)$. The root mean square Hamming distance $\mathrm{f}(\mathbf{b}_i, \mathbf{b}_j)$ of two binary strings $\mathbf{b}_i$ and $\mathbf{b}_j$ is defined as $\mathrm{f}(\mathbf{b}_i, \mathbf{b}_j) = (\sum_{i=1}^{l} s_i^2 / l)^{\frac{1}{2}}$. Consider the distance approximation error between the original distance $\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j)$ and the root mean squared Hamming distance $\mathrm{f}(\mathbf{b}_i, \mathbf{b}_j)$. We have the theorem

Theorem 1. *The distance approximation error between the original distance and the root mean squared Hamming distance is a lower bound of objective function* G,

$$\sum_{i,j} \left(\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j) - \lambda \mathrm{f}(\mathbf{b}_i, \mathbf{b}_j)\right)^2 \leq \mu \mathrm{G}$$

where parameter λ *and* μ *are two constant factors, which* $\lambda = \Delta\sqrt{l}$ *and* $\mu = 32\mathrm{n}$.

[1] $\|\mathbf{R}^\top \mathbf{a}\|_2^2 = (\mathbf{R}^\top \mathbf{a})^\top \mathbf{R}^\top \mathbf{a} = \mathbf{a}^\top \mathbf{R}\mathbf{R}^\top \mathbf{a} = \|\mathbf{a}\|_2^2$

Proof. According to the triangle inequality, we have

$$\sum_{i,j} \|\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j) - \mathrm{d}(\widehat{\mathbf{x}}_i, \widehat{\mathbf{x}}_j)\|$$
$$\leq \sum_{i,j} \|\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j) - \mathrm{d}(\mathbf{x}_i, \widehat{\mathbf{x}}_j)\| + \sum_{i,j} \|\mathrm{d}(\mathbf{x}_i, \widehat{\mathbf{x}}_j) - \mathrm{d}(\widehat{\mathbf{x}}_i, \widehat{\mathbf{x}}_j)\|$$
$$\leq 2\sum_{j} \mathrm{d}(\mathbf{x}_i, \widehat{\mathbf{x}}_j) + 2\sum_{i} \mathrm{d}(\mathbf{x}_i, \widehat{\mathbf{x}}_i) = 4\mathrm{d}(\mathbf{x}_i, \widehat{\mathbf{x}}_i).$$

Relax the right side of the above inequality, according to Cauchy-Schwarz Inequality, we have $\sum_i \mathrm{d}(\mathbf{x}_i, \widehat{\mathbf{x}}_i)$

$$\leq \sum_{i} \|\mathbf{x}_i - \mathbf{R}^\top \mathbf{y}_i\|_2 + \|\mathbf{R}^\top(\mathbf{y}_i - \widehat{\mathbf{y}}_i)\|_2$$
$$\leq (1^2 + ... + 1^2)^{\frac{1}{2}} \Big(\sum_{i} \|\mathbf{x}_i - \mathbf{R}^\top \mathbf{y}_i\|_2^2 + \|\mathbf{R}^\top(\mathbf{y}_i - \widehat{\mathbf{y}}_i)\|_2^2\Big)^{\frac{1}{2}}$$
$$= \sqrt{2\mathrm{n}} \Big(\|\mathbf{X} - \mathbf{R}^\top \mathbf{Y}\|_F^2 + \|\mathbf{Y} - \widehat{\mathbf{Y}}\|_F^2\Big)^{\frac{1}{2}}.$$

Then, with the inequality transitive property, we obtain

$$\sum_{i,j} \|\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j) - \mathrm{d}(\widehat{\mathbf{x}}_i, \widehat{\mathbf{x}}_j)\| \leq 4\sqrt{2\mathrm{n}}\mathrm{G}^{\frac{1}{2}}. \tag{a1}$$

On the other hand, as $\|\mathbf{R}^\top \mathbf{a}\|_2 = \|\mathbf{a}\|_2$, we have $\mathrm{d}(\widehat{\mathbf{x}}_i, \widehat{\mathbf{x}}_j)$

$$= \|\widehat{\mathbf{y}}_i - \widehat{\mathbf{y}}_j\|_2 = \sqrt{\sum_k \mathrm{d}_H^2(\mathbf{b}_i^k, \mathbf{b}_j^k)} = \Delta\sqrt{l}\mathrm{f}(\mathbf{b}_i, \mathbf{b}_j).$$

By substituting $\mathrm{d}(\widehat{\mathbf{x}}_i, \widehat{\mathbf{x}}_j) = \Delta\sqrt{l}\mathrm{f}(\mathbf{b}_i, \mathbf{b}_j)$ to (a1) and squaring both sides of the inequality, we obtain Theorem 1. □

4 Optimization

The goal is to minimize the objective of overall information loss in Eqn.4 with variables c, $\mathbf{R}$ and $\widehat{\mathbf{Y}}$.

4.1 Update $\widehat{\mathbf{Y}}$ and $\mathbf{R}$

To resolve $\mathbf{R}$ and $\widehat{\mathbf{Y}}$, we first fix the variable c, which is meant to fix the projection dimensionality over the course of alternating optimization of $\mathbf{R}$ and $\widehat{\mathbf{Y}}$. The alternating fashion works by updating $\mathbf{R}$ or $\widehat{\mathbf{Y}}$ with the other fixed.

Update $\widehat{\mathbf{Y}}$

When updating $\widehat{\mathbf{Y}}$, the learning objective reduces to the second term $\|\mathbf{Y} - \widehat{\mathbf{Y}}\|_F^2$. We simply assume that the data distribution is zero-centered. If c is an odd number where $\mathrm{c} = 2\mathrm{t} + 1$, the quantization centroid set $\mathcal{H}$ is represented as $\mathcal{H} = \{0, \pm\Delta, ..., \pm\mathrm{t}\Delta\}$. Without loss of generality, we just show the odd case. We quantize each element $\mathrm{y} \in \mathbf{Y}$ to the nearest value in $\mathcal{H}$

$$\mathrm{Q}(\mathrm{y}) = \arg\min_{\sigma \in \mathcal{H}} \|\sigma - \mathrm{y}\|^2. \tag{5}$$

The only variable in quantization function $\mathrm{Q}(.)$ is Δ. We solve Δ by minimizing the sum of MSE $\|\mathbf{Y} - \widehat{\mathbf{Y}}\|_F^2$,

$$\Delta^* = \arg\min_{\Delta} \sum_{\mathrm{y} \in \mathbf{Y}} \|\mathrm{y} - \mathrm{Q}(\mathrm{y})\|^2. \tag{6}$$

The above optimization problem is a multi chromatic pair problem in computational geometry [Berg *et al.*, 2000], which can be solved by Expectation Maximization (EM) algorithm [Moon and K, 1996].

Update $\mathbf{R}$

Updating $\mathbf{R}$ is a typical orthogonality constraint optimization problem. We apply the optimization procedure in [Z. and W., 2013] to update $\mathbf{R}$. Let $\mathbf{U}$ be the partial derivative of the objective function with respect to $\mathbf{R}$. We have $\mathbf{U} = -2\widehat{\mathbf{Y}}\mathbf{X}^\top$. We first define the skew-symmetric matrix [Z. and W., 2013]

$$\mathbf{M} = \mathbf{R}^\top \mathbf{U} - \mathbf{U}^\top \mathbf{R}. \tag{7}$$

Then, we adopt Crank Nicolson like [Smith, 1965] scheme to update the orthogonal matrix $\mathbf{R}$

$$\mathbf{R}^{(t+1)} = \mathbf{R}^{(t)} - \frac{\tau}{2}(\mathbf{R}^{(t)} + \mathbf{R}^{(t+1)})\mathbf{M} \tag{8}$$

where τ denotes the step size, we empirically set $\tau = 0.5$.

Convergence

We alternatively update $\widehat{\mathbf{Y}}$ and $\mathbf{R}$ in several iterations until convergence. In practice, we have found the algorithm converges in about 50-100 iterations. A typical behavior of the information loss (4) is shown in Fig.5.

4.2 Update c to Balance the Information Loss

Given a specified target code length, setting c as a large value can improve quantization quality but would degrade projection quality, and vice versa. There exists a trade-off between projection and quantization. To balance the information loss between projection and quantization, we aim to optimize c to minimize the objective of overall reconstruction bias. The value of c ranges from 1 to the target code length k (say hundreds or thousands). The brute-force enumeration method would be costly.

Rather than exhaustive search, we propose a fast approach to search for the optimal c, as our empirical findings have shown that the objective function with respect to c is unimodal. To explain this important findings, we have derived the theorem 2 by assuming a moderate distribution function (i.e. uniform or Gaussian) of projection values.

Theorem 2. *Function* $\mathrm{G}(\mathrm{c})$ *can be well-approximated by an unimodal function, which only has a single local minimum point* c^*. $\mathrm{G}(\mathrm{c})$ *is monotonically decreasing for* $\mathrm{c} \leq \mathrm{c}^*$ *and monotonically increasing for* $\mathrm{c} > \mathrm{c}^*$.

Proof. According to the orthogonal constraint in $\mathbf{R}$, we have

$$\|\mathbf{X} - \mathbf{R}^\top \mathbf{Y}\|_F^2 = \mathrm{Tr}\left((\mathbf{X} - \mathbf{R}^\top \mathbf{Y})(\mathbf{X} - \mathbf{R}^\top \mathbf{Y})^\top\right)$$
$$= \mathrm{Tr}\left(\mathbf{X}\mathbf{X}^\top - \mathbf{R}^\top \mathbf{Y}\mathbf{X}^\top - \mathbf{X}\mathbf{Y}^\top \mathbf{R} + \mathbf{R}^\top \mathbf{Y}\mathbf{Y}^\top \mathbf{R}\right)$$
$$= \mathrm{Tr}\left(\mathbf{X}\mathbf{X}^\top - \mathbf{Y}\mathbf{Y}^\top\right) = \|\mathbf{X}\|_F^2 - \|\mathbf{Y}\|_F^2.$$

Let matrix $\mathbf{E} = \mathbf{Y} - \hat{\mathbf{Y}}$. $\mathrm{G}(\mathrm{c})$ can be simplified as

$$\mathrm{G}(\mathrm{c}) = \|\mathbf{X}\|_F^2 - \|\mathbf{Y}\|_F^2 + \|\mathbf{E}\|_F^2.$$

The first term is independent of c. Each term of $\|\mathbf{Y}\|_F^2$ or $\|\mathbf{E}\|_F^2$ contains $\mathbf{n} \times \lfloor \frac{\mathbf{d}}{\mathbf{c}} \rfloor$ elements. The expression of $\mathrm{G}(\mathrm{c})$ depends on the distribution of projection values. We adopt

statistical expectation for sample estimation. Without loss of generalization, assuming d is divisible by c, we have

$$\|\mathbf{Y}\|_F^2 = \sum_{\mathrm{y}\in\mathbf{Y}} \mathrm{y}^2 \approx \frac{\mathrm{nd}}{\mathrm{c}}\mathrm{E}(\mathrm{y}^2),$$

$$\|\mathbf{E}\|_F^2 = \sum_{\mathrm{y}\in\mathbf{Y}} (\mathrm{y}-\mathrm{Q}(\mathrm{y}))^2 \approx \frac{\mathrm{nd}}{\mathrm{c}}\mathrm{E}((\mathrm{y}-\mathrm{Q}(\mathrm{y}))^2).$$

When projection values Y are subject to an uniform distribution, the probability density function is given by

$$\mathrm{f}(\mathrm{y}) = 1/(\mathrm{p}_2-\mathrm{p}_1),\ \mathrm{y}\in[\mathrm{p}_1,\mathrm{p}_2],\ \mathrm{p}_1<\mathrm{p}_2.$$

Accordingly, we have [Allen and Gray, 2012]

$$\mathrm{E}(\mathrm{y}^2) = \frac{(\mathrm{p}_1^2+\mathrm{p}_2^2+\mathrm{p}_1^2\mathrm{p}_2^2)}{3},\ \mathrm{E}(\mathrm{y}-\mathrm{Q}(\mathrm{y})^2) = \frac{\Delta^2}{12}.$$

In our method, step size $\Delta = \frac{\mathrm{p}_2-\mathrm{p}_1}{\mathrm{c}+1}$. Then,

$$\|\mathbf{Y}\|_F^2 = \frac{\mathrm{nd}(\mathrm{p}_1^2+\mathrm{p}_2^2+\mathrm{p}_1^2\mathrm{p}_2^2)}{3\mathrm{c}},$$

and

$$\|\mathbf{E}\|_F^2 = \frac{\mathrm{nd}(\mathrm{p}_2-\mathrm{p}_1)^2}{12\mathrm{c}(\mathrm{c}+1)^2}.$$

Let $\lambda = \mathrm{nd}(\mathrm{p}_1^2+\mathrm{p}_2^2+\mathrm{p}_1^2\mathrm{p}_2^2)/3$, $\mu = \mathrm{nd}(\mathrm{p}_2-\mathrm{p}_1)^2/12$ and $\eta = \|\mathbf{X}\|_F^2$, $\mathrm{G}(\mathrm{c})$ can be represented as

$$\mathrm{G}(\mathrm{c}) = \frac{\lambda}{\mathrm{c}(\mathrm{c}+1)^2} - \frac{\mu}{\mathrm{c}} + \eta.$$

Take the derivative of G with respect to c. We have

$$\frac{\partial \mathrm{G}}{\partial \mathrm{c}} = -\lambda\frac{3\mathrm{c}^2+4\mathrm{c}+1}{(\mathrm{c}^3+2\mathrm{c}^2+\mathrm{c})^2} + \frac{\mu}{\mathrm{c}^2},$$

and

$$\frac{\partial \mathrm{G}}{\partial \mathrm{c}} = 0 \ \Leftrightarrow\ \frac{\lambda}{\mu} = \frac{(\mathrm{c}+1)^4}{3\mathrm{c}^2+4\mathrm{c}+1}.$$

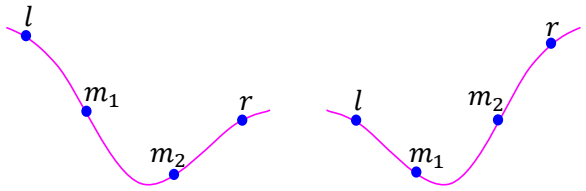
Let $\mathrm{H}(\mathrm{c})$ denote the function of right side in above equation,

$$\mathrm{H}(\mathrm{c}) = \frac{(\mathrm{c}+1)^4}{3\mathrm{c}^2+4\mathrm{c}+1} = \frac{(\mathrm{c}+1)^3}{3\mathrm{c}+1}.$$

$\mathrm{H}(\mathrm{c})$ is monotonically increasing for $\mathrm{c}\geq 1$. Assume c^* is the minimal point where $\mathrm{G}'(\mathrm{c}^*)=0$ and $\mathrm{H}(\mathrm{c}) = \frac{\lambda}{\mu}$. Then, for $\mathrm{c}>\mathrm{c}^*$ we have $\mathrm{H}(\mathrm{c}) > \frac{\lambda}{\mu}$ i.e. $\mathrm{G}'(\mathrm{c}^*)>0$, and for $\mathrm{c}<\mathrm{c}^*$ we have $\mathrm{H}(\mathrm{c}) < \frac{\lambda}{\mu}$ i.e. $\mathrm{G}'(\mathrm{c}^*)<0$. Thus, $\mathrm{G}(\mathrm{c})$ is unimodal. □

Likewise, for Gaussian distribution $\mathrm{f}(\mathrm{y}) = \frac{1}{\sigma\sqrt{2\pi}}\mathrm{e}^{-\frac{(\mathrm{y}-\mu)^2}{2\delta^2}}$, $\mathrm{y}\in[\mu-\mathrm{p},\mu+\mathrm{p}]$, we can still derive the expression of G by computing the expectation of y^2 and $(\mathrm{y}-(\mathrm{Q}(\mathrm{y})))^2$. We employ the second order Taylors expansion to represent $\mathrm{f}(\mathrm{y})$ and calculate $\mathrm{E}(\mathrm{y}^2)$ and $\mathrm{E}((\mathrm{y}-(\mathrm{Q}(\mathrm{y})))^2)$ by computing the integral of the Taylors expansion, followed by the derivative and monotonic analysis for the proof.

$\mathrm{G}(\mathrm{c})$ is defined in discrete domain $\mathrm{c}\in\{1,2,...,k\}$. According to the unimodal property in Theorem.2, we can apply ternary search to find out the optimal c^*. We have



If $G(m_1) > G(m_2)$, then update $l = m_1$.

If $G(m_1) \leq G(m_2)$, then update $r = m_2$.

Figure 2: A tool example of using ternary search algorithm to update l and r for an unimodal function.

Theorem 3. *Let* c^* *denote the minimum point of objective function* $\mathrm{G}(\mathrm{c})$. *Assume we have already known* $\mathrm{c}^*\in[l,r]$. *Let* $s=\frac{r-l}{3}$, $m_1=\lfloor l+\frac{1}{3}s\rfloor$ *and* $m_2=\lfloor l+\frac{2}{3}s\rfloor$. *We have*

- *if* $\mathrm{G}(\mathrm{m}_1)\leq\mathrm{G}(\mathrm{m}_2)$, *then* $l\leq c^*\leq \mathrm{m}_2$.
- *if* $\mathrm{G}(\mathrm{m}_1)>\mathrm{G}(\mathrm{m}_2)$, *then* $\mathrm{m}_1\leq \mathrm{c}^*\leq r$.

Proof. Consider $\mathrm{m}_1\neq\mathrm{m}_2$. For $\mathrm{G}(\mathrm{m}_1)<\mathrm{G}(\mathrm{m}_2)$, we have $\mathrm{c}^*\leq\mathrm{m}_2$. If not, then $\mathrm{m}_1<\mathrm{m}_2<\mathrm{c}^*$. As function $\mathrm{G}(c)$ is monotonically decreasing for $\mathrm{c}<\mathrm{c}^*$, then $\mathrm{G}(\mathrm{m}_1)>\mathrm{G}(\mathrm{m}_2)$. This is contradictory with the assumption. Thus, $\mathrm{c}^*\leq\mathrm{m}_2$ and $l\leq c^*\leq\mathrm{m}_2$. For $\mathrm{G}(\mathrm{m}_1)>\mathrm{G}(\mathrm{m}_2)$, the same procedure can be adapted to obtain $\mathrm{m}_1\leq\mathrm{c}^*\leq\mathrm{r}$. If $\mathrm{m}_1=\mathrm{m}_2$, we have $l=r$. Obviously, Theorem 3 still holds. □

The ternary search algorithm works as follows. We initialize $l=1$ and $r=k$. For each iteration, we set $c=\mathrm{m}_1$ and $c=\mathrm{m}_2$ respectively, and solve $\mathrm{G}(\mathrm{m}_1)$ and $\mathrm{G}(\mathrm{m}_2)$ by alternatively updating $\mathbf{R}$ and $\widehat{\mathbf{Y}}$. If $\mathrm{G}(\mathrm{m}_1)\leq\mathrm{G}(\mathrm{m}_2)$, we update $r=m_2$. Otherwise, we update $l=m_1$. The algorithm terminates when $l=r$. As we cut out $1/3$ search scope after each iteration, the run time order is

$$\mathrm{T}(\mathrm{k}) = \mathrm{T}(2\mathrm{k}/3)+1 = \mathrm{O}(\log k). \tag{9}$$

The reduced complexity benefits the fast search of optimal c, especially when learning long binary codes. Figure 2 shows an example of ternary search.

Algorithm 1 The algorithm to optimize Equation (4).

Input: original data points $\{\mathbf{x}_i\}_{i=1}^l$ and target code length k.
Output: a binary string $\mathbf{b}_i$ for each data point $\mathbf{x}_i$.

```
Initialize l = 1 and r = k.
while l < r do
    Let s = (r−l)/3, m1 = ⌊l + (1/3)s⌋ and m2 = ⌊l + (2/3)s⌋.
    Set c = m1, c = m2, and calculate G(m1), G(m2) by
    alternatively updating R and Ŷ.
    If G(m1) ≤ G(m2), then set r = m2.
    Otherwise, if G(m1) > G(m2), set l = m1.
end while
Set c = l. Project each x_i into y_i ∈ R^{k/c} and quantize each
projection value into c bits to obtain a binary string b_i with
⌊k/c⌋ × c bits.
```

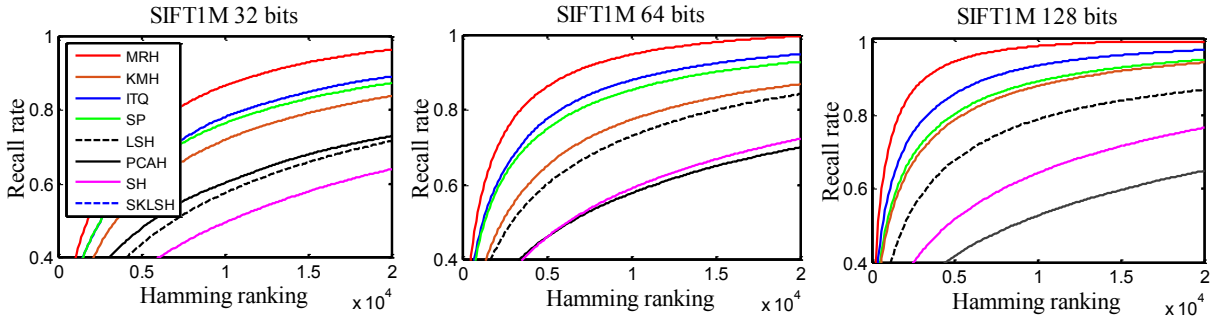


Figure 3: Results of recall rate of state-of-the-art hashing methods at code length 32, 64 and 128 bits on SIFT1M.

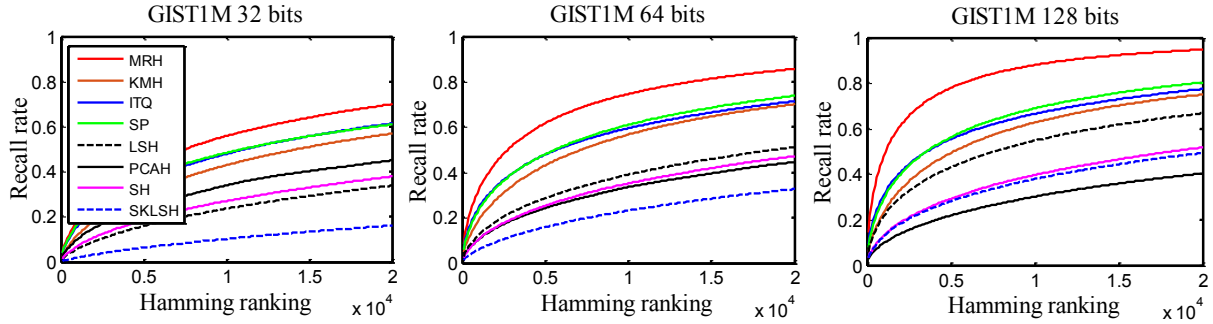


Figure 4: Results of recall rate of state-of-the-art hashing methods at code length 32, 64 and 128 bits on GIST1M.

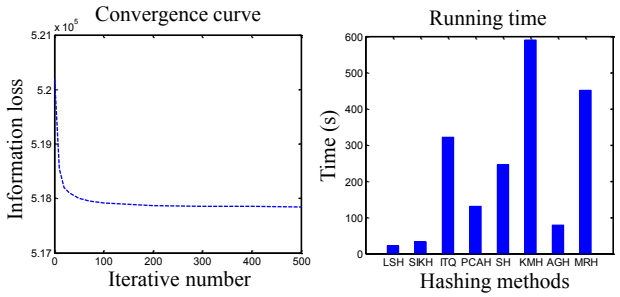


Figure 5: Left: a convergence curve over CIFAR10. Right: training time cost of baseline methods over ImageNet1M.

4.3 Complexity Analysis

We need $O(\log k)$ recursions to find out the optimal c. In each recursion, we iteratively update $\widehat{\mathbf{Y}}$ and $\mathbf{R}$. Let t denote the number of iterations for the alternatively updating, t_1 the iteration number in EM algorithm and t_2 the iteration number in Crank Nicolson like scheme [Smith, 1965]. It takes $O(t_1 c^2 n)$ to update $\widehat{\mathbf{Y}}$ and $O(nl + d^2 + t_2 ld)$ to update $\mathbf{R}$. The overall time complexity is $O(\log k(t_1 c^2 n + ln + d^2 + t_2 ld))$. We will show the running time cost in the next section. Algorithm 1 shows the pseudo-code of our MRH algorithm.

5 Experiments

We evaluate and compare the approaches over five benchmarks SIFT1M [Jegou *et al.*, 2011], GIST1M [Jegou *et al.*, 2011], CIFAR10 [Krizhevsky, 2009], LableMe22k [Torralba *et al.*, 2008] and ImageNet1M [Deng *et al.*, 2009]. SIFT1M [Jegou *et al.*, 2011] and GIST1M [Jegou *et al.*, 2011] are the feature datasets containing 1 million 128-D SIFT [Lowe, 2004] and 960-D GIST [Aude and Torralba, 2001] features, respectively. The CIFAR10 dataset contains 60,000 images. The LabelMe22K dataset contains 22,019 images. Each image in CIFAR-10 and LabelMe22K is represented by a 512 dimensional GIST feature [Aude and Torralba, 2001]. ImageNet1M is another large-scale benchmark with 1 million images. For each image, we extract a 4096-D fisher vector [Perronnin and Dance, 2006] to evaluate the performance in a high dimensional space.

5.1 Baseline Methods

We perform extensive comparison with 7 state-of-the-art methods: Local sensitive hashing (LSH) [Andoni and Indyk, 2006], Iterative quantization (ITQ) [Gong and Lazebnik, 2011], Shift invariant kernels hashing (SIKH) [Raginsky and Lazebnik, 2009], Principal component analysis hashing (PCAH) [Wang *et al.*, 2006], Spectral hashing (SH) [Weiss *et al.*, 2008], K-means hashing (KMH) [He *et al.*, 2013] and Sparse Projection Hashing (SP) [Xia *et al.*, 2015]. All the methods are run with released source codes in default settings.

5.2 Setting and Configuration

We follow most of previous hashing works to adopt the Hamming distance ranking for ANN search. Recall rate and mean average precision (mAP) are used for performance comparison. For each benchmark, we select 1000 data points as

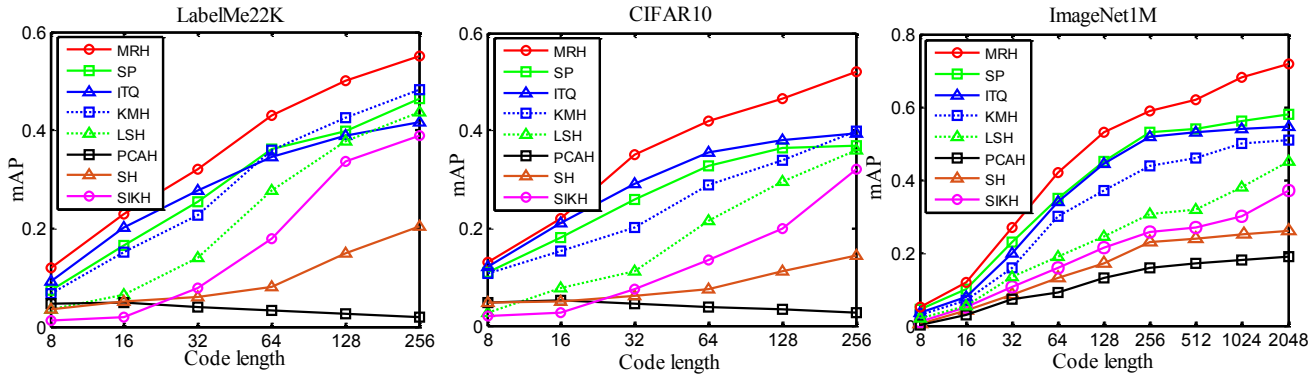


Figure 6: Results of mAP of state-of-the-art hashing methods on LableMe-22K, CIFAR10,and ImageNet1M dataset.

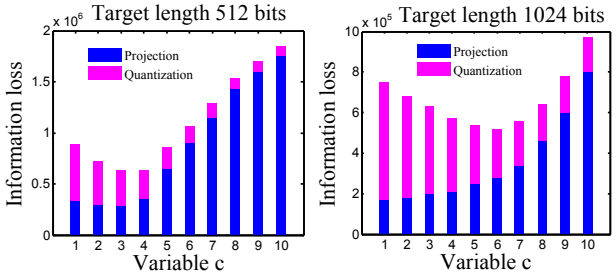


Figure 7: Results of information loss derived from the objective function when setting different c values.

queries and leave the rest as database. For each query, the top 100 nearest data points in Euclidean distance are used as the ground truth. We random select $10K$ data points from each dataset for training. We empirically set $t = 100$, $t_1 = 50$ and $t_2 = 50$. Figure 5(b) shows the training time. Except for KMH [He *et al.*, 2013], other baseline methods are run in a single thread. All experiments are carried out over a server with Intel i5 CPU at 3.20GHz and 64Gb memory cache.

5.3 Performance

Figure 4 and Figure 3 shows the results of recall rate of baseline methods over datasets GIST1M and SIFT1M. The proposed MRH consistently outperforms the state-of-the-art methods. Over dataset GIST1M, the performance gains of MRH are with 9.1%, 13.4% and 17.8% recall rate when ranking 20,000 in code length 32 bits, 64 bits and 128 bits, respectively, and with 7.1%, 5.4% and 3.2% recall rate on SIFT1M.

Figure 6 shows the mAP results of baseline methods. We report the results over LabelMe, CIFAR and ImageNet1M. MRH outperforms the baselines on all settings even at small codes. As the code length increases, the performance gap becomes more significant. For example, MRH outperforms the competitive method SP by 2.1%, 2.9%, 4.2%, 7.4%, 8.5% and 7.1% at code length 8 bits to 256 bits over LabelMe, respectively. Considerable improvements are also obtained over CIFAR10 as well as ImageNet1M.

Figure 7 shows the impact of projection dimensionality for learning 512 and 1024 bits codes over ImageNet1M. Variable c produces balancing effects on projection and quantization.

Table 1: The optimal value of c for learning binary codes with different length on dataset CIFAR, LabelMe and ImageNet.

Dataset	Code Length 2^n							
	$\leq 2^4$	2^5	2^6	2^7	2^8	2^9	2^{10}	2^{11}
LabelMe	1	2	2	3	3	-	-	-
CIFAR	1	1	2	3	4	-	-	-
ImageNet	1	1	2	3	4	4	6	8

Increasing c reduces the quantization error but it incurs more projection distortions, and vice versa. There exists a trade-off between projection and quantization. From figure 7, the best setting is $c = 4$ for 512 bits and $c = 6$ for 1024 bits. We notice that MRH tends to set c to a large value for long size codes, which means that more bits are allowed for quantizing the values of each projection element. By adaptively adjusting the projection dimensionality, MRH obtains discriminative codes with overall minimal information loss. Table 1 lists the optimal settings of c on three datasets.

6 Conclusion

The joint optimization of projection and quantization impacts the similarity preserving of binary codes, which is important for generating discriminative binary codes. To optimize the projection dimensionality has tackled the problem of balancing the information loss between the projection and quantization stages. The practice of jointly optimizing projection dimensionality, projection matrix, as well as quantization functions is expected to facilitate the state-of-the-art Hashing methods.

Acknowledgments

This work was supported by the National Hightech R&D Program of China (863 Program): 2015AA016302, and Chinese Natural Science Foundation: 61271311, 61390515, 61421062. Part of this work was carried out at the Rapid-Rich Object Search (ROSE) Lab at the Nanyang Technological University, Singapore. The ROSE Lab is supported by the National Research Foundation, Singapore, under its Interactive Digital Media (IDM) Strategic Research Programme. Ling-Yu Duan is the corresponding author.

References

[Allen and Gray, 2012] Gersho Allen and Robert M. Gray. Vector quantization and signal compression. *Springer Science and Business Media*, 2012.

[Andoni and Indyk, 2006] Alexandr Andoni and Piotr Indyk. Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions. *In IEEE FOCS*, 2006.

[Aude and Torralba, 2001] Oliva Aude and Antonio Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 2001.

[Berg *et al.*, 2000] De Berg, Mark, Marc Van Kreveld, Mark Overmars, and Otfried Cheong Schwarzkopf. Computational geometry. *Springer Berlin Heidelberg*, 2000.

[Deng *et al.*, 2009] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. *CVPR*, 2009.

[Duan *et al.*, 2016] Ling-Yu Duan, Vijay Chandrasekhar, Jie Chen, Jie Lin, Zhe Wang, Tiejun Huang, Bernd Girod, and Wen Gao. Overview of the mpeg-cdvs standard. *TIP*, 2016.

[Gionis *et al.*, 1999a] Gionis, Aristides, Piotr Indyk, and Rajeev Motwani. Similarity search in high dimensions via hashing. *VLDB*, 1999.

[Gionis *et al.*, 1999b] Aristides Gionis, Piotr Indyk, and Rajeev Motwani. Similarity search in high dimensions via hashing. *VLDB*, 1999.

[Gong and Lazebnik, 2011] Yunchao Gong and Svetlana Lazebnik. Iterative quantization: A procrustean approach to learning binary codes. *CVPR*, 2011.

[He *et al.*, 2013] Kaiming He, Fang Wen, and Jian Sun. K-means hashing: an affinity-preserving quantization method for learning binary compact codes. *CVPR*, 2013.

[Jegou *et al.*, 2011] Herve Jegou, Matthijs Douze, and Cordelia Schmid. Product quantization for nearest neighbor search. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2011.

[Kong and L, 2012] Weihao Kong and Wu-Jun L. Double-bit quantization for hashing. *In Proceedings of the Twenty-Sixth AAAI*, 2012.

[Kong and Li, 2012] Weihao Kong and Wu-Jun Li. Isotropic hashing. *In NIPS*, 2012.

[Kong *et al.*, 2012] Weihao Kong, Wu-Jun Li, and Minyi Guo. Manhattan hashing for large-scale image retrieval. *In Proceedings of the 35th international ACM SIGIR*, 2012.

[Krizhevsky, 2009] Alex Krizhevsky. Learning multiple layers of features from tiny images. *Tech report, University of Torontos*, 2009.

[Kulis and Darrell, 2009] Brian Kulis and Trevor Darrell. Learning to hash with binary reconstructive embeddings. *NIPS*, 2009.

[Liu *et al.*, 2010] Wei Liu, Jun Wang, Sanjiv Kumar, and Shih-Fu Chang. Hashing with graphs. *ICML*, 2010.

[Liu *et al.*, 2012] Wei Liu, Jun Wang, Rongrong Ji, Yu-Gang Jiang, and Shih-Fu Chang. Supervised hashing with kernels. *CVPR*, 2012.

[Liu *et al.*, 2014] Wei Liu, Cun Mu, Sanjiv Kumar, and Shih-Fu Chang. Discrete graph hashing. *In NIPS*, 2014.

[Lowe, 2004] David G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004.

[Moon and K, 1996] Moon and Tood K. The expectation-maximization algorithm. *Signal processing magazine*, 1996.

[Norouzi and Fleet, 2011] Mohammad Norouzi and David Fleet. Minimal loss hashing for compact binary codes. *In ICML*, 2011.

[Perronnin and Dance, 2006] Florent Perronnin and Christopher Dance. Fisher kenrels on visual vocabularies for image categorizaton. *CVPR*, 2006.

[Raginsky and Lazebnik, 2009] Maxim Raginsky and Svetlana Lazebnik. Locality-sensitive binary codes from shift-invariant kernels. *In NIPS*, 2009.

[Shen *et al.*, 2013] F. Shen, C. Shen, Q. Shi, A. van den Hengel, and Z. Tang. Inductive hashing on manifolds. *CVPR*, 2013.

[Smith, 1965] Gordon D Smith. Numerical solution of partial differential equations. *math*, 1965.

[Torralba *et al.*, 2008] Antonio Torralba, Robert Fergus, and Yair Weiss. Small codes and large image databases for recognition. *In Proceedings of CVPR*, 2008.

[Wang *et al.*, 2006] Xin-Jing Wang, Lei Zhang, Feng Jing, and Wei-Ying Ma. Annosearch: Image auto-annotation by search. *CVPR*, 2006.

[Wang *et al.*, 2010] Jun Wang, Sanjiv Kumar, and Shih-Fu Chang. Semi-supervised hashing for scalable image retrieval. *CVPR*, 2010.

[Wang *et al.*, 2015] Zhe Wang, Ling-Yu Duan, Jie Lin, Xiaofang Wang, Tiejun Huang, and Wen Gao. Hamming compatible quantization for hashing. *IJCAI*, 2015.

[Wang *et al.*, 2016] Zhe Wang, Ling-Yu Duan, Tiejun Huang, and Wen Gao. Affinity preserving quantization for hashing: A vector quantization approach to learning compact binary codes. *AAAI*, 2016.

[Weiss *et al.*, 2008] Yair Weiss, Antonio Torralba, and Rob Fergus. Spectral hashing. *In NIPS*, 2008.

[Xia *et al.*, 2015] Yan Xia, Kaiming He, Pushmeet Kohli, and Jian Sun. Sparse projections for high-dimensional binary codes. *CVPR*, 2015.

[Xu *et al.*, 2013] Bin Xu, Jiajun Bu, Yue Lin, Chun Chen, Xiaofei He, and Deng Cai. Harmonious hashing. *IJCAI*, 2013.

[Z. and W., 2013] Wen Z. and Yin W. A feasible method for optimization with orthogonality constraints. *Mathematical Programming 138*, 2013.