Minimizing Reconstruction Bias Hashing via Joint Projection Learning and Quantization

Ling-Yu Duan, *Member, IEEE*, Yuwei Wu, Yicheng Huang, Zhe Wang, Junsong Yuan, *Senior Member, IEEE*, and Wen Gao, *Fellow, IEEE*

***Abstract*— Hashing, a widely studied solution to the approximate nearest neighbor search, aims to map data points in the high-dimensional Euclidean space to the low-dimensional Hamming space while preserving the similarity between original points. As directly learning binary codes can be NP-hard due to discrete constraints, a two-stage scheme, namely, "projection and quantization", has already become a standard paradigm for learning similarity-preserving hash codes. However, most existing hashing methods typically separate these two stages and thus fail to investigate complementary effects of both stages. In this paper, we systematically study the relationship between "projection and quantization", and propose a novel minimal reconstruction bias hashing (MRH) method to learn compact binary codes, in which the projection learning and quantization optimizing are jointly performed. By introducing a lower bound analysis, we design an effective ternary search algorithm to solve the corresponding optimization problem. Furthermore, we conduct some insightful discussions on the proposed MRH approach, including the theoretical proof, and computational complexity. Distinct from previous works, the MRH can adaptively adjust the projection dimensionality to balance the information loss between the projection and quantization. The proposed framework not only provides a unique perspective to view traditional hashing methods, but also evokes some other researches, e.g., guiding the design of the loss functions in deep networks. Extensive experiment results have shown that the proposed MRH significantly outperforms a variety of state-of-the-art methods over eight widely used benchmarks.**

***Index Terms*— Bias hashing, quantization error, joint optimization, image retrieval.**

I. Introduction

APPROXIMATE nearest neighbor (ANN) search is a fundamental problem that appears in many applications in computer vision, machine learning and information retrieval [1]–[3]. Its goal is to find some approximate nearest neighbors for a query from a collection of data points by encoding high-dimensional feature vectors to short binary codes while preserving similarities between original data. Using similarity preserving binary codes to represent original data points can significantly reduce the memory storage cost and boost similarity distance computing speed, hence is of particular interest for the fast ANN search [4], especially when dealing with large scale databases [5]–[7].

A common binary coding approach, often called hashing, aims at learning similarity preserving hash functions for mapping data points to a low-dimensional Hamming space. As discussed in [4], directly learning the optimal binary codes is equivalent to a graph partitioning problem which is typically NP-hard. Therefore, existing hashing methods often adopt a two-stage strategy: projection[1] and quantization [8], [9]. Concretely, the original data points $\mathbf{x} \in \mathbb{R}^d$ is first projected to a low-dimensional space by discarding the discrete constraints, given by

$$\mathbf{y} = [f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_k(\mathbf{x})] \in \mathbb{R}^k,$$

where $\{f_i(\cdot)\}_{i=1}^k$ are projection functions. Then each element $f_i(\mathbf{x})$ in $\mathbf{y}$ is quantized into a single bit or multiple bits to generate binary codes [10]. Most existing hashing methods attempt to rectify a single stage and can be naturally classified to either projection-centered or quantization-centered hashing. Unfortunately, such an approximate solution often makes the resulting hash functions less effective due to the accumulated quantization error. In this paper, we concentrate on *how to elegantly connect the projection with the quantization, and to maximize the positive complementary effects of two stages instead of heavily relying on only one of them.*

[1]The term "projection" is not restricted to linear mapping, non-linear dimension reduction techniques are also applicable.

Manuscript received August 9, 2017; revised January 18, 2018; accepted March 14, 2018. Date of publication March 21, 2018; date of current version April 6, 2018. This work was supported by the National Natural Science Foundation of China under Grant 61661146005, Grant U1611461, and Grant 61390515, and in part by the National Key Research and Development Program of China under Grant 2016YFB1001501, and in part by the PKU-NTU Joint Research Institute through a donation from the Ng Teng Fong Charitable Foundation. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Ling Shao. *(Corresponding author: Ling-Yu Duan.)*

L.-Y. Duan is with the Institute of Digital Media, School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China, and also with the PKU-NTU Joint Research Institute, Beijing 100871, China (e-mail: lingyu@pku.edu.cn).

Y. Wu is with the Beijing Laboratory of Intelligent Information Technology, School of Computer Science, Beijing Institute of Technology, Beijing 100081, China, and also with the PKU-NTU Joint Research Institute, Beijing 100871, China (e-mail: wuyuwei@bit.edu.cn).

Y. Huang and Z. Wang are with the Institute of Digital Media, School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China (e-mail: anorange0409@pku.edu.cn; zwang@pku.edu.cn).

J. Yuan is with the Department of Computer Science and Engineering, The State University of New York, Buffalo, NY 14260-2500 USA (e-mail: jsyuan@buffalo.edu).

W. Gao is with the Institute of Digital Media, School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China, and also with the PKU-NTU Joint Research Institute, Beijing 100871, China (e-mail: wgao@pku.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2018.2818008

Many research efforts have been devoted to the **projection stage**, aiming at learning powerful projections to maintain the similarity structure of the original data. Locality Sensitive Hashing (LSH) [11] and its kernel version, *i.e.,* Kernelized Locality Sensitive Hashing (KLSH) [12] adopt random projections followed by simple thresholding to map data points close in the Euclidean space to similar codes. Although random projection based hashing methods are data independent and flexible, long hash codes are often required to meet the desired performance, which increases computation and memory consumption [13]. To build up a more effective projection which better captures the underlying geometry of specific datasets, there has been increasing research efforts devoted to data-dependent hashing methods. By fine-tuning the projection mapping over the training data, data-dependent methods typically outperform random projection based methods with shorter codes [14]. Typical methods include Spectral Hashing [4], Isotropic Hashing [9], and Harmonious Hashing [15] *etc.* These methods preserve the similarity between data points by keeping the points that are close in the original Euclidean space as neighbors in the Hamming space. It has also been shown that harnessing bilinear projection [16]–[19], sparse projection [20], nonlinear manifold embedding [14], [21]–[24], and deep nonlinear projection [25], [26] will help produce neighborhood-preserving binary codes.

Moreover, recent works have reported the significant impact of **quantization** on hashing performance [8], [27]–[30]. Single bit quantization in most hashing methods incurs lots of quantization errors, which could seriously degrade the ANN search performance [9]. Actually, it is unreasonable to suppose that the projection dimensionality must be equal to the length of targe codes [4]. If the data inherently lies in a low subspace, we could further reduce the dimensionality to remove the redundancy in data points, making the projection values more discriminative. When we project the data point into lower dimensions (*e.g.,* $\lfloor \frac{k}{c} \rfloor$), we would have more bits (c bits) to encode each projection value to reduce the quantization error.

Fig.1 shows the results by using different projection dimensionality to learn a binary code with 256 bits. In this scenario, we employ c bits to encode each projection value and thus the projection dimensionality will be $\lfloor \frac{256}{c} \rfloor$. The blue line indicates the retrieval performance of the projection values (prior to quantization), and the red dashed line indicates the performance after the quantization. When we project data points to 256-D ($c = 1$), the performance dramatically degrades on the quantization stage. A better way is to project data points into 64-D and then use 4 bits to quantize each projection value. Although the 64-D feature is not as discriminative as the 256-D feature, but $c = 4$ bits coding could effectively reduce the performance drops in single bit coding, and the final results of $c = 4$ is better than others.

The aforementioned observation demonstrates that both the projection and quantization stage can significantly contribute to the quality of the learned hash codes in terms of ANN search performance. However, the non-individual or joint behaviors, as well as complementary effects of the projection and quantization stages has not been systematically analyzed in the literatures, especially in the context where the pairwise similarity of samples should be preserved. It is thus desirable that the joint optimization of both stages should be established for ANN search. That is, given a set of data points and the target code length k, if the data points lie in a low dimensional manifold, we may project the data points into a lower-dimensional space (project more) while still maintaining the geometry structure of the original data, and using more bits to quantize each projected dimension instead of single bit (quantization less). For instance, we may project the 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 to be k. Therefore, we should adaptively adjust the projection dimensionality to better balance the information loss between two stages and minimize the total loss over the whole process of hashing. As a result, a natural question arises here: given a target code length k, *shall we project more or quantize more?*

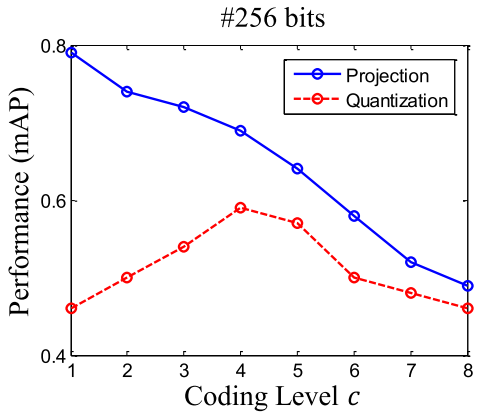


Fig. 1. The retrieval performance of mean Average Precision (mAP) on different projection dimensionality to learn 256 bits binary code. The blue line indicates the performance when data points are projected into subspace $\lfloor \frac{k}{c} \rfloor$ (without quantization). The dotted red line indicates the performance after quantization stage with c bits coding for each projection value. Experiments are evaluated on 1 million Fisher vector (FV) [31] feature extracted from ImageNet1M [32].

To this end, designing a unified learning objective in line with both quality assessments is crucial not only to establish a concrete mathematical implementation for the expected joint optimization model, but also to justify the widely applied two-stage hashing paradigm by revealing the inherent relationship between two stages. In this paper, we propose a novel hashing method called Minimal Reconstruction-bias Hashing (MRH) to tackle the problem of jointly optimizing projection and quantization stages with a unified learning objective function. Our contributions are three-fold:

- We present a novel approach to learning 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 systematically studies the interrelationship 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 consistently over several benchmarks.

- We come up with a unified learning objective function to resolve the joint optimization problem from the perspective of minimal reconstruction bias of signals. By introducing a lower bound analysis, we establish the relationship between the information loss from both projection and quantization, and the Hamming approximation errors, which to some extent justified the widely adopted two-stage hashing paradigm theoretically.
- By analyzing the unimodal characteristics of the MRH objective function with respect to projection dimensionality, we propose an effective ternary search algorithm to solve the joint optimization problem of MRH. In particular, we have reduced the complexity of searching optimal projection dimensionality from $\mathcal{O}(N)$ to $\mathcal{O}(log(N))$.

This paper is an extended version of the work previously published in [33]. Apart from the substantially extended introduction, related work and in-depth discussion, this submission differs from [33] in the following major aspects: (1) We have further given the detailed exploration on the proposed algorithm from different point of views including the rigorous derivations, theoretical proof, convergence analysis, and computational complexity. (2) More extensive experiments have been conducted on eight benchmark datasets to demonstrate the effectiveness of our method.

The remaining sections are organized as follows. Section II gives a brief review of the ANN search. Section III, as the preliminary, introduces the motivation of this paper. In Section IV, we formulate the problem of optimal binary coding from the perspective of minimizing the reconstruction bias of signals. The detailed optimization of our formulation is introduced in Section V. Section VI demonstrates that the distance approximation error between the original distance and the root mean squared Hamming distance is a lower bound of our objective function. We conduct comprehensive experiments to demonstrate the superiority of the proposed framework in Section VII, and conclude the paper in Section VIII.

II. Related Work

Nearest neighbor (NN) search is the optimization problem of finding the point in a given set that is closest (or most similar) to a given point, which is widely used in various applications, such as computer vision, multimedia, and machine learning. Our work is related to approximate nearest neighbor (ANN) search methods, which can be roughly grouped into two categories: Vector Quantization based methods [34]–[41] and Hashing based methods [11], [16], [17], [20], [23], [26], [42]–[44]. Readers are referred to [45] and [46] for the comprehensive review.

In this section, we mainly review the closely related work of hashing. In general, hashing methods aim to map the original data usually denoted by the high-dimensional floating point number representations into the low-dimensional binary Hamming space while preserving the similarities among original data to some extent. It is well known that the binary constraints imposed on the objective function make the optimization problem NP-hard. A two-stage scheme, namely "projection and quantization", has already become a standard paradigm for learning similarity-preserving hash codes.

A. Hashing on Projection

There are two categories of mainstream hashing approaches based on the type of hashing functions, called data-independent method and data-dependent method. Data-independent methods do not rely on any training data, which are flexible, but often require long codes to achieve satisfactory performance. Local Sensitive Hashing [11] (LSH) adopts a random projection which is independent of training data. Similarly, Shift Invariant Kernel Hashing [47] (SIKH) chooses the random projection and applies a shifted cosine function to generate binary codes. The kernelized version of LSH, Kernelized Locality-Sensitive Hashing (KLSH) [12], has also been proposed for large-scale image retrieval. Theoretically, the Hamming distance between hash codes can progressively approximate the Euclidean distance between original features. Nevertheless, fairly long hash codes (*e.g.,* more than 1024 bits) are often required to achieve satisfactory retrieval performance.

Different from data-independent methods, data-dependent methods learn the hashing function from training data and outperform data-independent methods. Weiss *et al.* [4] proposed spectral hashing (SH) by using spectral partitioning for the graph constructed from the data similarity relationships. Kong and Li [9] proposed an isotropic hashing method to learn projection functions with isotropic variances such that larger-variance dimensions will carry more information. Xia *et al.* [20] presented an effective sparse regularizer to reduce the effective number of parameters involved in the learned projection operator, which effectively decreases the computational cost for computing long codes. Moreover, several works are devoted to handling high-dimensional data by the bilinear form of hash functions [16], [17]. Along this line, Liu *et al.* [18] presented a binary projection bank method which can effectively reduce the high-dimensional representations to medium-dimensional binary codes without sacrificing accuracies. Yu *et al.* [19] employed a circulant matrix to generate binary codes, in which the circulant structure enables the use of Fast Fourier Transformation to speed up the computation. In addition, many efforts are devoted to studying the problem of learning hash functions in the context of multi-modal data for cross-modal similarity search [48]–[50].

Notwithstanding the effectiveness of preserving data similarity in the original space by the aforementioned hashing methods, these methods may fail to preserve the non-linear manifold structure of data due to the linear or bilinear projections employed by them. Accordingly, methods exploiting non-linear projections have gained increasing popularity. Liu *et al.* [14], [21] leveraged the anchor graph to approximate the similarity matrix for efficient nearest neighbor search. Through feature transformation, Jiang and Li [22] effectively approximated the whole graph without explicitly computing the similarity graph matrix, and proposed a sequential learning method to learn the hash functions in a bit-wise manner. Shen *et al.* [24] proposed an Inductive Manifold Hashing scheme which generates nonlinear coding

functions by exploiting the non-parametric manifold learning approach. Guo *et al.* [23] discovered that the low-dimensional embedding of anchors has a significant impact on the hash function, and resolved the optimized anchor embedding by solving the orthogonality constrained maximization problem. Liu *et al.* [51] proposed an unsupervised hashing approach which exploits the ordinal information between data points, and learns the optimal hashing functions with a graph-based approximation to embed the ordinal relations.

It is noted that great success in deep neural networks for representation learning has inspired deep hashing algorithms [25], [26], [42], [43]. Xia *et al.* [52] computed hash codes by minimizing the similarity difference where image representation and hash function are jointly learnt through a deep network. Dai *et al.* [53] proposed a novel generative approach to learn hash functions, in which the discrete optimization is replaced by the maximization over the negative Helmholtz free energy. Distinct from most previous hashing methods which seek a single linear or non-linear projection to map each sample into a binary code, deep hashing can seek multiple hierarchical non-linear transformations to learn binary codes, so that the nonlinear relationship of samples is well exploited [54]. However, deep neural networks often require hundreds of megabytes of storage, which make them inconvenient to deploy in mobile applications or memory light-weight hardware. In this paper, we focus on developing an effective hashing algorithm in a hand-crafted but light-weight manner to jointly optimize the projection distortions and quantization errors. In addition, most deep hashing methods often employ single bit quantization to encode the projected vector, which results in the information loss to some extent. We may consider to applying the projection distortion and quantization error as learning objectives to facilitate the training of deep networks.

B. Hashing on Quantization

The methods mentioned above are restricted to the fixed projection dimensionality, *i.e.,* the number of projection dimensionality is equal to the target code size. Specifically, to learn k bits length code, they would project a data point into k dimensionality subspace, and then obtain the binary codes by taking a sign function. This coding strategy can not maximally utilize the allocated bits to encode the most valuable information. The single bit coding for each projection value which may incur lots of information loss, as a projection value can be any real number but it becomes a single bit after quantization. Thus, promising multiple bits quantization (MBQ) methods have been proposed. Double bits quantization [8], [27] divides each projection dimension into three regions and uses double bits code to represent each element region. Overall, MBQ methods do facilitate the reduction of information loss in quantization. Recently, Since clustering is a powerful quantization method to model the complex relationships of data points, several hashing methods exploit the clustering structure among data in the binary quantization, *e.g.,* spherical hashing (SPH) [28], K-means hashing (KMH) [29], and adaptive binary quantization (ABQ) [30]. Experimental results have demonstrated the functionality of high quality quantization in improving hashing performance [8], [9], [27].

The two-stage scheme of "projection and quantization" often makes the resulting hash functions less effective due to the accumulated quantization error. Gong *et al.* [6] proposed Iterative Quantization (ITQ) which figures out an orthogonal rotation matrix to refine the initial projection matrix. Although ITQ can decrease the quantization distortion, it learns orthogonal rotations over pre-computed mappings which usually makes ITQ suboptimal. Wang *et al.* [27] introduced a hamming compatible quantization method to minimize the distance error function to preserve the capability of similarity metric between the Euclidean space and Hamming space. Liu *et al.* [21] developed a discrete graph hashing method to directly solve the binary code without discarding discrete constraints. To further obtain high-quality hash codes, Shen *et al.* [55] directly handled the discrete constraints by using the discrete proximal linearized minimization algorithm. Our method differs from the closely related works [6], [27], [55]. We learn similarity preserving binary codes which jointly optimizes both projection and quantization stages with adjustable projection dimensionality. In addition, a rigorous lower bound analysis of the information loss between the projection and quantization is introduced to support our model.

III. Preliminaries

We first introduce the basic notations. Let matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$ 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 $\mathbf{x}_i \in \mathbb{R}^d$ that maximizes similarity preservation in the Hamming space.

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

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

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

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

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

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

Note that we introduce a 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 perform a joint optimization of projection $T(\cdot)$ and quantization $Q(\cdot)$.

The range of Hamming distance is limited to the length of binary codes. The maximum Hamming distance of c-bit codes is only c. When we use c bits to encode 2^c values, the distance consistency in the Hamming space cannot be maintained. Let us take the example of $c = 2$. We quantize the projection values into $2^2 = 4$ centroids with $\sigma_0 < \sigma_1 < \sigma_2 < \sigma_3$, namely,

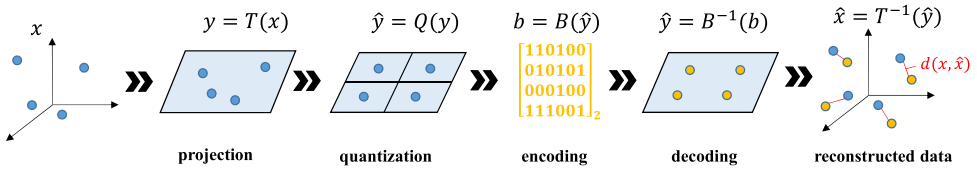


Fig. 2. Illustration of reconstruction bias from projection and quantization. Towards optimal binary coding, the aim is to minimize this bias. The red lines indicate the reconstruction bias. This figure is best viewed in color version.

$(00)_2, (01)_2, (10)_2$ and $(11)_2$. We have $\|\sigma_1 - \sigma_2\| < \|\sigma_1 - \sigma_3\|$, but $d_H(01, 10) > d_H(01, 11)$ where $d_H(\cdot)$ denotes the Hamming distance.

To address the issue of inconsistent measurements, we may resort to other distance measurements like Manhattan distance [56]. But it would seriously degrade the retrieval efficiency [27]. In contrast, Hamming distance measurement is extremely fast, and more than 10^9 operations can be done per second [29] [4], so Hamming distance computing is still the priority of the 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 centroids $\mathcal{H} = \{\sigma_i\}_{i=0}^{c}$, where $\sigma_i - \sigma_{i-1} = \Delta$ for $1 \leq i \leq c$. Then, we apply a unary representation [57] to encode each centroid $B(\sigma_i) = U_c(i)$ for $\sigma_i \in \mathcal{H}$, where unary representation $U_c(i)$ is defined as a c-bit binary string with i ones followed by $c - i$ zeros, *e.g.*, $U_2(1) = 10$, $U_3(0) = 000$, $U_4(2) = 1100$. With the unary representation, the Hamming distance between binary codes is proportional to the distance of the centroids,

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

Clearly, the unary representation is an incomplete encoding strategy, as a c-bit code is capable of representing 2^c states. To make the Hamming distance consistent with the distance between quantization centroids, we have to discard parts of the coding space.

IV. 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. The relationship between minimal reconstruction bias and Hamming approximation errors will be studied theoretically in Section VI.

A. 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. 2). Specifically, we first decode $\mathbf{b}_i$ and get $\widehat{\mathbf{y}}_i = B^{-1}(\mathbf{b}_i)$. $B^{-1}(\cdot)$ denotes the inverse quantization. It maps an integer quantization index $\mathbf{b}_i$ to the reconstruction value $\widehat{\mathbf{y}}_i$ that is the output approximation of the input value. Quantization function $Q(\cdot)$ 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

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

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 $\mathbf{x}_i$

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

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

B. Learning Objective

The reconstruction bias indicates the information loss incurred by mapping the data points to the Hamming space. To preserve the similarity structure of original data points, we aim to minimize the reconstruction bias. Directly optimizing the objective function in Eq. (2) is intractable due to a large number of free parameters in $\widehat{\mathbf{y}}_i$ and the orthogonal constraint of $\mathbf{R}$. Here, we can turn to minimize the square of reconstruction bias in Eq. (2) according to the Theorem 1.

Theorem 1: The squared reconstruction bias function can be decomposed into the sum of projection distortions and quantization errors

$$\|\mathbf{x}_i - \mathbf{R}^\top \widehat{\mathbf{y}}_i\|_2^2 = \|\mathbf{x}_i - \mathbf{R}^\top \mathbf{y}_i\|_2^2 + \|\mathbf{y}_i - \widehat{\mathbf{y}}_i\|_2^2. \tag{3}$$

Proof: First we separate projection distrotion and quantization error terms inside the norm operator

$$\|\mathbf{x}_i - \mathbf{R}^\top \widehat{\mathbf{y}}_i\|_2^2 = \|(\mathbf{x}_i - \mathbf{R}^\top \mathbf{y}_i) + \mathbf{R}^\top(\mathbf{y}_i - \widehat{\mathbf{y}}_i)\|_2^2.$$

Notice that the quantization error vector $\mathbf{R}^\top(\mathbf{y}_i - \widehat{\mathbf{y}}_i)$ lies in the projection subspace spanned by row vectors of $\mathbf{R}$, while the projection error vector $\mathbf{x}_i - \mathbf{R}^\top \mathbf{y}_i$ is orthogonal to the projection space, and hence orthogonal to the quantization error vector. Using Pythagoras theorem gives us

$$\|\mathbf{x}_i - \mathbf{R}^\top \widehat{\mathbf{y}}_i\|_2^2 = \|(\mathbf{x}_i - \mathbf{R}^\top \mathbf{y}_i)\|_2^2 + \|\mathbf{R}^\top(\mathbf{y}_i - \widehat{\mathbf{y}}_i)\|_2^2. \tag{4}$$

Finally, notice that $\mathbf{R}$ is orthogonal, we have that for any $\mathbf{a} \in \mathbb{R}^{\frac{k}{c}}$

$$\|\mathbf{R}^\top \mathbf{a}\|_2^2 = (\mathbf{R}^\top \mathbf{a})^\top \mathbf{R}^\top \mathbf{a} = \mathbf{a}^\top \mathbf{R}\mathbf{R}^\top \mathbf{a} = \|\mathbf{a}\|_2^2.$$

Plugging it back to Eq. (4) finishes the proof. ■

Let $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n]$ denote the projection matrix and $\widehat{\mathbf{Y}} = Q(\mathbf{Y})$. To figure out the optimal binary coding, we formulate the problem as joint minimization of the projection

distortions and the quantization errors, in which the projection dimensionality c may be variable as well. Specifically, the learning objective is formulated as

$$\begin{aligned}&\arg\min_{c,\mathbf{R},\widehat{\mathbf{y}}} \|\mathbf{X}-\mathbf{R}^\top\mathbf{Y}\|_F^2+\|\mathbf{Y}-\widehat{\mathbf{Y}}\|_F^2.\\ &s.t.\ 1\le c\le k, \mathbf{R}\in\mathbb{R}^{\frac{k}{c}\times d}, \mathbf{R}\mathbf{R}^\top=\mathbf{I}\\ &\quad \mathbf{Y}=\mathbf{R}\mathbf{X}, \widehat{\mathbf{Y}}\in\mathcal{H}^{\frac{k}{c}\times n}, \|\mathcal{H}\|=c+1. \end{aligned} \tag{5}$$

In Eq. (5), $\|\cdot\|_F$ denotes the Frobenius norm. The first term $\|\mathbf{X}-\mathbf{R}^\top\mathbf{Y}\|_F^2$ denotes the sum of projection distortions, indicating the information loss in the projection stage, and the second term $\|\mathbf{Y}-\widehat{\mathbf{Y}}\|_F^2$ denotes the sum of mean square error (MSE), indicating the information loss in the quantization stage. In particular, the variable c is to adjust the projection dimensionality to adaptively balance the information loss between the projection and quantization stages in a joint optimization.

V. Optimization

The goal is to minimize the objective of overall reconstruction errors defined in Eq. (5) with respect to c, $\mathbf{R}$ and $\widehat{\mathbf{Y}}$.

A. Update $\widehat{\mathbf{Y}}$ and $\mathbf{R}$

To resolve $\mathbf{R}$ and $\widehat{\mathbf{Y}}$, we first fix the variable c, which means 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.

1) Update $\widehat{\mathbf{Y}}$: When updating $\widehat{\mathbf{Y}}$, the learning objective reduces to the quantization error term $\|\mathbf{Y}-\widehat{\mathbf{Y}}\|_F^2$. In the theory of the signal quantization, as an example, rounding a real number to the nearest integer value forms a very basic type of quantizer, *i.e.,* a uniform one. We simply assume that the data distribution is zero-centered. In this case, a typical (mid-tread) uniform quantizer with a quantization step size Δ can be applied, because the mid-tread quantizer has zero as one of its quantized values. It is useful for situations where it is necessary for the zero value to be represented.

Due to the midtreading of zero, the number of quantizing level is odd if a symmetric sample value range is to be covered [58]. Assume that c is an odd number where $c=2t+1$, the quantization centroid set $\mathcal{H}$ is represented as $\mathcal{H}=\{\pm\Delta/2,\pm3\Delta/2,\ldots,\pm(2t+1)\Delta/2\}$. We quantize each element $\mathbf{y}\in\mathbf{Y}$ to the nearest value in $\mathcal{H}$

$$Q(y)=\operatorname*{argmin}_{\sigma\in\mathcal{H}}\|\sigma-\mathbf{y}\|^2.$$

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

$$\begin{aligned}\Delta^*=&\operatorname*{argmin}_{\Delta}\sum_{\mathbf{y}\in\mathbf{Y}}\|\mathbf{y}-Q(\mathbf{y})\|^2,\\ &s.t.\ Q(\mathbf{y})\in\mathcal{H},\ \|\sigma_i-\sigma_{i-1}\|=\Delta.\end{aligned} \tag{6}$$

We observed that the function in Eq. (6) is a segmented quadratic function of Δ which can be solved by separately enumerating each quadratic segment. Let $\mathrm{f}_{\mathbf{y}}(\Delta)$ denotes the quantization error of the element $\mathbf{y}$. $\mathrm{f}_{\mathbf{y}}(\Delta)$ is a function of Δ which is defined as

$$\mathrm{f}_{\mathbf{y}}(\Delta)=\min\Big\{(\mathbf{y}\pm\frac{1}{2}\Delta)^2,\ldots,(\mathbf{y}\pm\frac{2t+1}{2}\Delta)^2\Big\}.$$

Optimizing Eq. (6) equals to minimize function $\sum\mathrm{f}_{\mathbf{y}}(\Delta)$ of Δ.

For any $a\in\mathbb{R}$, based on the symmetry in the definition, $\mathrm{f}_a(\cdot)$ is an even function where $\mathrm{f}_a(\Delta)=\mathrm{f}_a(-\Delta)$. Consider the function expression of $\mathrm{f}_a(\Delta)$ for $\Delta\in[0,\infty)$. Let

$$\begin{cases}(a-\frac{1}{2}\Delta)^2=(a-\frac{3}{2}\Delta)^2 & \Rightarrow\Delta=a\\ (a-\frac{3}{2}\Delta)^2=(a-\frac{5}{2}\Delta)^2 & \Rightarrow\Delta=\frac{a}{2}\\ \cdots\\ (a-\frac{2t-1}{2}\Delta)^2=(a-\frac{2t+1}{2}\Delta)^2 & \Rightarrow\Delta=\frac{a}{t}.\end{cases}$$

$\mathrm{f}_a(\Delta)$ is a segmented function which is segmented by split points $\{\frac{a}{l}\}_{l=1}^t=\{a,\frac{a}{2},\ldots,\frac{a}{t}\}$. The function expression of $\mathrm{f}_a(\Delta)$ can be written as

$$\mathrm{f}_a(\Delta)=\begin{cases}\mathrm{f}_a^t=(a-\frac{2t+1}{2}\Delta)^2, & 0\le\Delta\le\frac{a}{t}\\ \cdots\\ \mathrm{f}_a^l=(a-\frac{2l+1}{2}\Delta)^2, & \frac{a}{l+1}\le\Delta\le\frac{a}{l}\\ \cdots\\ \mathrm{f}_a^0=(a-\frac{1}{2}\Delta)^2 & a\le\Delta.\end{cases}$$

where $l=1,2,\ldots,t-1$.

Let $\mathbf{F}=\sum\mathrm{f}_{\mathbf{y}}^{o_i}(\Delta)$, where o_i is the o_i-th segment function in $\mathrm{f}_a(\Delta)$. $\mathbf{F}$ is segmented by the split points in $S=\bigcup\{\frac{a}{l}\}_{l=1}^t$ which has totally $N=\frac{k}{c}nt$ split points. We first sort the elements in S, *i.e.,* $s_1\le s_2\le\ldots\le s_N$, then enumerate each segment one by one, started at $\Delta\in[0,s_1)$ and ended at $\Delta\in[s_N,\infty)$. We initialize all μ_i to t. Once we are in interval $\Delta\in[s_l,s_{l+1})$, we update each μ_i by

$$o_i=o_i-1,\ if\ \ \frac{\mathbf{y}}{o_i}<s_l\ \ and\ \ o_i\neq0$$

and then we solve the minimal value in this interval. The overall minimal value of all segments is selected as the solution. Algorithm 1 shows the pseudo-code for solving Eq. (6).

The other uniform quantizer does not have zero as one of its quantized values, so is called midrise. Its number of decision intervals is even if a symmetric sample value range is to be covered. Assume that c is an even number where $c=2t$, the quantization centroid set $\mathcal{H}$ is represented as $\mathcal{H}=\{\pm\Delta,\pm\Delta,\ldots,\pm t\Delta\}$. Let $\mathrm{f}_{\mathbf{y}}(\Delta)$ denotes the quantization error of the element $\mathbf{y}$. $\mathrm{f}_{\mathbf{y}}(\Delta)$ is a function of Δ which is defined as

$$\mathrm{f}_{\mathbf{y}}(\Delta)=\min\Big\{(\mathbf{y}\pm\Delta)^2,\ldots,(\mathbf{y}\pm t\Delta)^2\Big\}.$$

Optimizing Eq. (6) equals to minimize function $\sum\mathrm{f}_{\mathbf{y}}(\Delta)$ of Δ. The subsequent procedures are similar with the case of $c=2t+1$, which will not be described in detail.

Algorithm 1 The Algorithm to Optimize Eq. (6)

Input: projected matrix $\mathbf{Y}$, and coding level $c = 2t + 1$.
Output: quantization step Δ.
Let $\mathbf{F} = \sum \mathrm{f}_{\mathbf{y}}^{o_i}(\Delta)$ and sort the elements in S.
Initialize $o_i = t$ and set the mean square error as $\mathrm{MSE} = \infty$.
Define $s_0 = 0$ and $s_N = \infty$.
for l from 0 to N **do**
For o_i, if $\frac{\mathbf{y}}{o_i} < s_l$ and $o_i \neq 0$, then $o_i = o_i - 1$.
Compute $\mathbf{F}(\Delta^*) = \min\{\mathrm{f}_{\mathbf{y}}^{\mu_i}\}$ for $\Delta \in [s_{l-1}, s_l)$.
If $\mathbf{F}(\Delta^*) < \mathrm{MSE}$, update $\mathrm{MSE} = \mathbf{F}(\Delta^*)$ and $\Delta = \Delta^*$.
end for

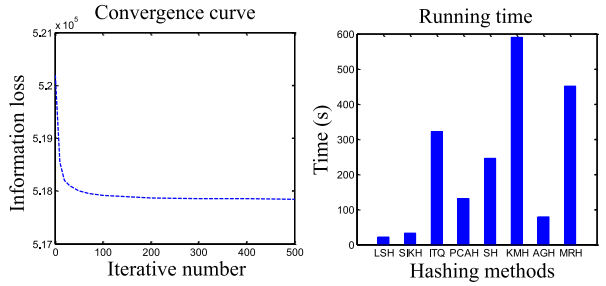


Fig. 3. Left: a convergence curve on the CIFAR10 dataset. Right: training time cost of baseline methods on the ImageNet1M dataset.

2) Update $\mathbf{R}$*:* Updating $\mathbf{R}$ is a typical optimization problem with orthogonality constraints. We apply the optimization procedure in [59] to update $\mathbf{R}$. Let $\mathbf{U}$ be the partial derivative of the objection function with respect to $\mathbf{R}$. We have

$$\mathbf{U} = \frac{\partial G}{\partial \mathbf{R}} = \frac{\partial \|\mathbf{X} - \mathbf{R}^\top \widehat{\mathbf{Y}}\|_F^2}{\partial \mathbf{R}} = -2\widehat{\mathbf{Y}}\mathbf{X}^\top. \tag{7}$$

To preserve the orthogonality constraint on $\mathbf{R}$, we first define the skew-symmetric matrix [59]:

$$\mathbf{M} = \mathbf{R}^\top \mathbf{U} - \mathbf{U}^\top \mathbf{R}. \tag{8}$$

Then, we adopt Crank Nicolson like scheme [60] to update the orthogonal matrix $\mathbf{R}$:

$$\mathbf{R}^{(t+1)} = \mathbf{R}^{(t)} - \frac{\tau}{2}\left(\mathbf{R}^{(t+1)} + \mathbf{R}^{(t)}\right)\mathbf{M}, \tag{9}$$

where τ denotes the step size. We empirically set $\tau = 0.5$. By solving Eq. (9), we can get

$$\mathbf{R}^{(t+1)} = \mathbf{R}^{(t)}\mathbf{M}, \tag{10}$$

and

$$\mathbf{M}^{(t+1)} = \left(\mathbf{I} + \frac{\tau}{2}\mathbf{M}\right)^{-1}\left(\mathbf{I} - \frac{\tau}{2}\mathbf{M}\right). \tag{11}$$

We iteratively update $\mathbf{R}$ several times based on Eq. (10) with the Barzilai-Borwein (BB) method [59] and alternatively update $\widehat{\mathbf{Y}}$ and $\mathbf{R}$ in several iterations until convergence. In practice, the algorithm usually converges within 50 iterations. A typical behavior of the learning objective function Eq. (5) is shown in Fig. 3.

B. Update c

Given a specified code length, setting c to a large value can improve quantization quality but could degrade projection quality, and vice versa. To balance the information loss between the projection and quantization, we aim to find the optimal c to minimize the objective of overall reconstruction bias. Since the value of c ranges from 1 to the target code length k (say hundreds or thousands). Undoubtedly, the brute-force enumeration method will be time-consuming.

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

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

Proof: According to the orthogonal constraint on $\mathbf{R}$, we have

$$\begin{aligned}\|\mathbf{X} - \mathbf{R}^\top\mathbf{Y}\|_F^2 &= tr\left((\mathbf{X} - \mathbf{R}^\top\mathbf{Y})(\mathbf{X} - \mathbf{R}^\top\mathbf{Y})^\top\right)\\ &= tr\left(\mathbf{X}\mathbf{X}^\top - \mathbf{R}^\top\mathbf{Y}\mathbf{X}^\top - \mathbf{X}\mathbf{Y}^\top\mathbf{R} + \mathbf{R}^\top\mathbf{Y}\mathbf{Y}^\top\mathbf{R}\right)\\ &= tr\left(\mathbf{X}\mathbf{X}^\top - \mathbf{Y}\mathbf{Y}^\top\right) = \|\mathbf{X}\|_F^2 - \|\mathbf{Y}\|_F^2.\end{aligned}$$

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

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

The first term is independent of c. Each term of $\|\mathbf{Y}\|_F^2$ or $\|\mathbf{E}\|_F^2$ contains $n \times \lfloor \frac{k}{c} \rfloor$ elements. The expression of $G(c)$ depends on the distribution of projection values. We adopt statistical expectation for sample estimation. Without loss of generality, assuming k is divisible by c, we have

$$\begin{aligned}\|\mathbf{Y}\|_F^2 &= \sum_{\mathbf{y}\in\mathbf{Y}} \mathbf{y}^2 \approx \frac{nk}{c} E(\mathbf{y}^2),\\ \|\mathbf{E}\|_F^2 &= \sum_{\mathbf{y}\in\mathbf{Y}} (\mathbf{y} - Q(\mathbf{y}))^2 \approx \frac{nk}{c} E((\mathbf{y} - Q(\mathbf{y}))^2).\end{aligned}$$

When projection values Y are subject to a uniform distribution on close interval $[p_1, p_2]$, the probability density function is given by

$$f(\mathbf{y}) = 1/(p_2 - p_1),\ \mathbf{y} \in [p_1, p_2],\ p_1 < p_2$$

Accordingly, we have [61]

$$E(\mathbf{y}^2) = \frac{(p_1^2 + p_2^2 + p_1^2 p_2^2)}{3},\ E((\mathbf{y} - Q(\mathbf{y}))^2) = \frac{\Delta^2}{12}$$

In our method, step size $\Delta = (p_2 - p_1)/(c + 1)$. Then,

$$\|\mathbf{Y}\|_F^2 = \frac{nk(p_1^2 + p_2^2 + p_1^2 p_2^2)}{3c},$$

[2]A function is said to be "unimodal" if it only has one local extremum.

and

$$\|\mathbf{E}\|_F^2 = \frac{nk(p_2 - p_1)^2}{12c(c+1)^2}.$$

Let $\mu = nk(p_1^2 + p_2^2 + p_1^2 p_2^2)/3$, $\lambda = nk(p_2 - p_1)^2/12$ and $\eta = \|\mathbf{X}\|_F^2$,$G(c)$ can be represented as

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

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

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

and

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

Let $H(c)$ denote the funtion of right hand side in the above equation. $H(c)$ is monotonically increasing for $c > 1$. Assume c^* is the optimal point where $G'(c^*) = 0$ and $H(c^*) = \frac{\lambda}{\mu}$. Then, for $c > c^*$ we have $H(c) > \frac{\lambda}{\mu}$ and $G'(c) > 0$,and for $c < c^*$ we have $H(c) < \frac{\lambda}{\mu}$ and $G'(c) < 0$. Thus $G(c)$ is indeed unimodal. ■

Likewise, for the Gaussian distribution,

$$f(\mathbf{y}) = \frac{1}{\sigma\sqrt{2\pi}} \exp(-\frac{(\mathbf{y} - \mu)^2}{2\sigma^2}), \mathbf{y} \in [\mu - p, \mu + p]$$

we can still derive the expression of G by computing the expectation of $\mathbf{y}^2$ and $(\mathbf{y} - Q(\mathbf{y}))^2$. We employ the second order Taylors expansion to represent $f(\mathbf{y})$ and calculate $E(\mathbf{y}^2)$ and $E((\mathbf{y} - Q(\mathbf{y}))^2)$ by computing the integral of the Taylors expansion, followed by the derivative and monotonic analysis for the proof.

C. A Ternary Search Algorithm

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

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

- if $G(m_1) \leq G(m_2)$, then $l \leq c^* \leq m_2$.
- if $G(m_1) > G(m_2)$, then $m_1 \leq c^* \leq r$.

Proof: Consider $m_1 \neq m_2$. For $G(m_1) < G(m_2)$, we have $c^* \leq m_2$. If not, then $m_1 < m_2 < c^*$. As the function $G(c)$ is monotonically decreasing for $c < c^*$, then $G(m_1) > G(m_2)$. This is contradictory with the assumption. Thus, $c^* \leq m_2$ and $l \leq c^* \leq m_2$. For $G(m_1) > G(m_2)$, the same procedure can be adopted to obtain $m_1 \leq c^* \leq r$. If $m_1 = m_2$, 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 $c = m_1$ and $c = m_2$ respectively, and solve $G(m_1)$ and $G(m_2)$ by alternatively updating $\mathbf{R}$ and $\widehat{\mathbf{Y}}$. If $G(m_1) \leq G(m_2)$, 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

$$T(k) = T(2k/3) + 1 = O(\log k) \tag{12}$$

The reduced complexity benefits the fast search of c^*, especially when learning long binary codes. Fig. 4 shows an example of ternary search.

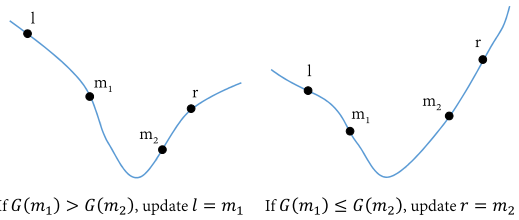


Fig. 4. A toy example of using ternary search algorithm to update l and r for an unimodal function.

Algorithm 2 The Algorithm for Optimizing Equation (5)

Input: Original data points $\{\mathbf{x}_i\}_{i=1}^n$ 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 $s = \frac{r-l}{3}, m_1 = \lfloor l + \frac{s}{3} \rfloor$ and $m_2 = \lfloor l + \frac{2s}{3} \rfloor$.
  Set $c = m_1, c = m_2$. and calculate $G(m_1), G(m_2)$ by alternatively updating $\mathbf{R}$ and $\widehat{\mathbf{Y}}$.
  if $G(m_1) \leq G(m_2)$ **then**
    Set $r = m_2$.
  else
    Set $l = m_1$.
  end if
end while
Set $c = l$, Project each $\mathbf{x}_i$ into $\mathbf{y}_i \in \mathbb{R}^{\frac{k}{c}}$ and quantize each projection value into c bits to obtain a binary string $\mathbf{b}_i$ with $\lfloor \frac{k}{c} \rfloor \times c$ bits.

D. 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_1 denote the number of iterations for the alternatively updating and t_2 the iteration number in Crank Nicolson like scheme [60]. 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 $O(\log k(t_1 c^2 n + nl + d^2 + t_2 ld))$. Algorithm 2 shows the pseudo-code of our MRH algorithm.

VI. Relationship Between Our Model and Hamming Approximation

Similarity preserving hashing methods aim to map close data points to near binary codes [11], [57]. 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 Eq. (5). Since the

Hamming approximation quality, as a critical indicator, can significantly impact the performance of ANN search [29], [62], this lower bound analysis may justify 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, $d_H(\mathbf{b}_i^k, \mathbf{b}_j^k)$, where $\mathbf{b}_i^k = B(\widehat{\mathbf{y}}_i^k)$, $\mathbf{b}_i^k$ 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 $s_k = d_H(\mathbf{b}_i^k, \mathbf{b}_j^k)$. The root mean square Hamming distance $f(\mathbf{b}_i, \mathbf{b}_j)$ of two binary strings $\mathbf{b}_i$ and $\mathbf{b}_j$ is defined as $f(\mathbf{b}_i, \mathbf{b}_j) = (\sum_{i=1}^{l} s_i^2/l)^{\frac{1}{2}}$. Consider the distance approximation error between the original distance $d(\mathbf{x}_i, \mathbf{x}_j)$ and the root mean squared Hamming distance $f(\mathbf{b}_i, \mathbf{b}_j)$, we have the theorem

Theorem 4: The distance approximation error between the original distance and the root mean squared Hamming distance is a lower bound of objective function G in Eq. (5),

$$\sum_{i,j} \left(d(\mathbf{x}_i, \mathbf{x}_j) - \lambda f(\mathbf{b}_i, \mathbf{b}_j)\right)^2 \leq \mu G. \tag{13}$$

where parameter $\lambda = \Delta\sqrt{l}$ and $\mu = 32n$ are constant factors.

Proof: According to the triangle inequality, we have

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

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

$$\begin{aligned}
&\leq \sum_{i} \|\mathbf{x} - \mathbf{R}^\top \mathbf{y}_i\|_2 + \|\mathbf{y}_i - \widehat{\mathbf{y}}_i\|_2 \\
&\leq (1^2 + \cdots + 1^2)^{\frac{1}{2}} \left(\sum_{i} \|\mathbf{x}_i - \mathbf{R}^\top \mathbf{y}_i\|_2^2 + \|\mathbf{y}_i - \widehat{\mathbf{y}}_i\|_2^2\right)^{\frac{1}{2}} \\
&= \sqrt{2n}\left(\|\mathbf{X} - \mathbf{R}^\top \mathbf{Y}\|_F^2 + \|\mathbf{Y} - \widehat{\mathbf{Y}}\|_F^2\right)^{\frac{1}{2}}.
\end{aligned}$$

Then, with the inequality transitive property, we obtain

$$\sum_{i,j} |d(\mathbf{x}_i, \mathbf{x}_j) - d(\widehat{\mathbf{x}}_i, \widehat{\mathbf{x}}_j)| \leq 4\sqrt{2n} G^{\frac{1}{2}}. \tag{14}$$

On the other hand, as $\|\mathbf{R}^\top \mathbf{a}\|_2 = \|\mathbf{a}\|_2$, we have

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

By substituting $d(\widehat{\mathbf{x}}_i, \widehat{\mathbf{x}}_j) = \Delta\sqrt{l} f(\mathbf{b}_i, \mathbf{b}_j)$ to Eq. (14) and squaring both sides of the inequality, we obtain Theorem 4. ∎

VII. Experiments

In this section, extensive experiments are conducted to evaluate the proposed method in terms of ANN search accuracy and recall rate. Our approach is implemented in Matlab 2016b. The experiments are performed on an DELL desktop computer with 3.40GHz Intel Core(TM) i7−6700 CPU and 16GB RAM.

A. Datasets

We evaluate and compare the state-of-the-art approaches over eight benchmark datasets SIFT1M [34], GIST1M [34], CIFAR10 [63], LableMe22K [5], MNIST [64], NUS-WIDE [65], MPEG CDVS [7] and ImageNet1M [32]. SIFT1M and GIST1M are popular large-scale dataset to evaluate hash models. They contain one million unlabeled data with each data represented by a 128-dimensional SIFT feature vector and a 960-dimensional GIST feature vector, respectively. The CIFAR10 dataset is a labeled subset of the 80M Tiny Images collection [63]. It consists of 10 classes with each class containing 6K 32×32 color images, leading to 60K images in total. The LabelMe22K dataset contains 22, 019 images. In the experiments, each image in CIFAR10 and LabelMe22K is represented by a 512-dimensional GIST feature. The MNIST dataset contains 70, 000 images of handwritten digits numbers $0 \sim 9$. Each image was resized to 28×28 by computing the center of mass of the pixels and vectorized into a 784 dimensional gray scale feature. NUS-WIDE is a web image dataset including 269, 468 images along with six types of low-level features extracted from these images. We use the 500 dimensional BOW (bag-of-visual-words) features based on SIFT local descriptors. MPEG CDVS is a benchmark for evaluating compact descriptors in visual search, containing 28, 590 images of five classes: graphics, paintings, frames, landmarks and common objects. For each image, we extract a 512 dimensional Nested Invariant Pooling (NIP), which is adopted by MPEG CDVA standardization [66]. ImageNet1M is a large-scale benchmark with 1 million images. For each image, we extract a 4096 dimensional Fisher vector [31] to evaluate the performance in a high dimensional space.

B. Configuration and Evaluation

To evaluate the effectiveness of our method, we perform extensive comparisons with 8 methods: Locality sensitive hashing (LSH) [11], Iterative quantization (ITQ) [6], Scalable graph hashing (ScGH) [22], Sparse projection hashing (SP) [20] , Adaptive binary quantization (ABQ) [30], Binary autoencoders (BA) [44], Ordinal constraint hashing (OCH) [51], and Stochastic generative hashing (StGH) [53]. All the methods are run with released source codes in default settings.

We follow most previous hashing works to adopt the Hamming distance ranking for ANN search. Recall and mean average precision (mAP) are widely employed to test the accuracy of approximate nearest search. Specifically, recall@R is a fraction of the true nearest neighbor (ground truth) is found in the first R items. Average precision (AP) is the average proportion that the K true nearest neighbors are retrieved in the most relevant R samples considering the order of samples. The mAP is the mean of the AP for all queries. For each benchmark, we randomly select 1000 data points as queries and leave the rest as database. For each query, the top

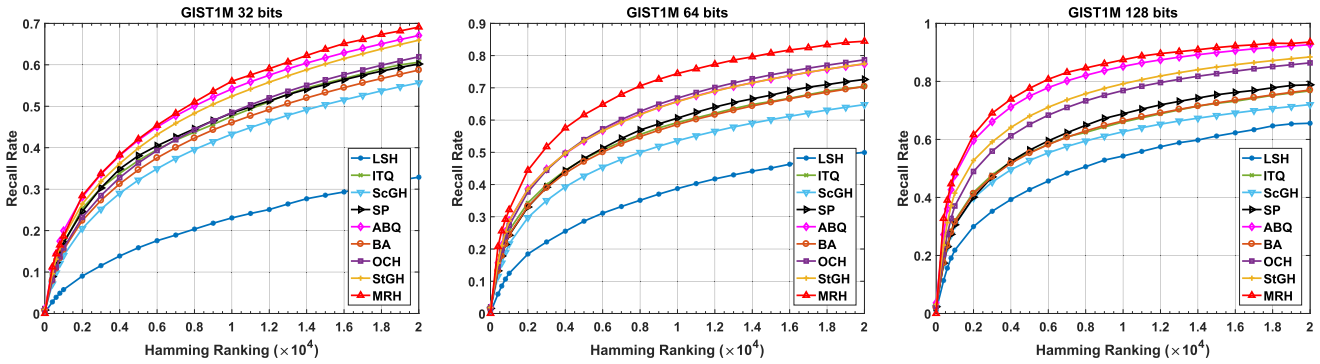


Fig. 5. Results of recall rate of state-of-the-art hashing methods at code length 32, 64 and 128 bits on GIST1M.

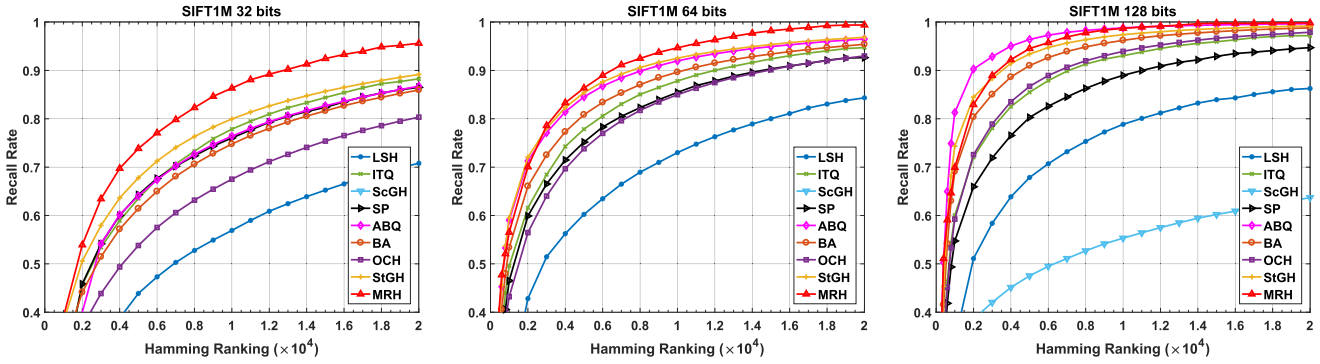


Fig. 6. Results of recall rate of state-of-the-art hashing methods at code length 32, 64 and 128 bits on SIFT1M.

100 nearest data points in Euclidean distance are used as the ground truth. We randomly select $10K$ data points from each dataset for training. We empirically set $t = 100$, $t_1 = 50$ and $t_2 = 50$. All baseline methods are run in a single thread.

C. Comparisons With Competing Methods

To comprehensively demonstrate the efficiency and effectiveness of the proposed MRH, we further compare it with several related hashing algorithms in terms of mAP and recall. Fig. 5 and Fig. 6 show the results of recall rate of baseline methods over the GIST1M and SIFT1M datasets. The proposed MRH consistently outperforms the state-of-the-art methods (only except the result on the SIFT1M 128 bits compared with ABQ). The performance gains of MRH are with 2%, 5.7%, 6.4%, 2.5% recall rate when ranking 20, 000 on the dataset GIST1M 32 bits, GIST1M 64 bits, SIFT1M 32 bits and SIFT1M 64 bits, respectively. This is beneficial from our joint optimization of "projection and quantization". MRH can learn an optimal quantization bit number c serving as the resulting hashing codes. On the dataset GIST1M 128 bit and SIFT1M 128 bit, MRH performs comparable with the ABQ method.

Fig. 7 shows the mAP results of both MRH and baseline methods. We report the results over the LabelMe22K, CIFAR10, MNIST, NUS-WIDE, MPEG CDVS and ImageNet1M datasets. For each dataset, we evaluate all methods at different bitrates, starting from 16 $= 2^4$ and keep multiplying the current bitrate by 2 until it exceeds the dimension of the original features. For example, for the NUS-WIDE dataset where the dimensionality of original features is 500, we evaluate at 16, 32, 64, 128 and 256 bits. For the ImageNet1M dataset with 4096-D Fisher vector, we evaluate up to 2048 bits. The results demonstrate significant advantage of MRH over the baseline methods on the most settings, even at short codes. As the code length increases, the performance gap generally becomes more significant. The MRH outperforms the competitive method ITQ by 0.4%, 1.6%, 3.1%, 16.8% and 28.5% at the code length from 16 bits to 256 bits over the LabelME22K dataset, respectively. Considerable improvements are also obtained on other datasets. Furthermore, the performance of MRH on the different datasets are stable. As shown in Fig. 7, the MRH always maintain a high performance over different datasets with different bitrates, while other baseline methods typically suffer from low performance on a few datasets due to inapplicable data distribution. For example, ABQ works badly on the MNIST dataset. SP, ITQ, BA and OCH have poor performance on the NUS-WIDE dataset. StGH fails on the MPEG CDVS dataset. In particular, several methods even performs worse as the code length increases, such as the ABQ on the CIFAR10 and LableMe22K datasets, the SP on the NUS-WIDE dataset.

It is noted that on MPEG CDVS dataset, ITQ and SP methods perform slightly better at small bitrates, but MRH still has a clear advantage when the bitrate reaches 512 bits. This is reasonable since it can be shown that the optimal projection dimensionality always equals to the code length

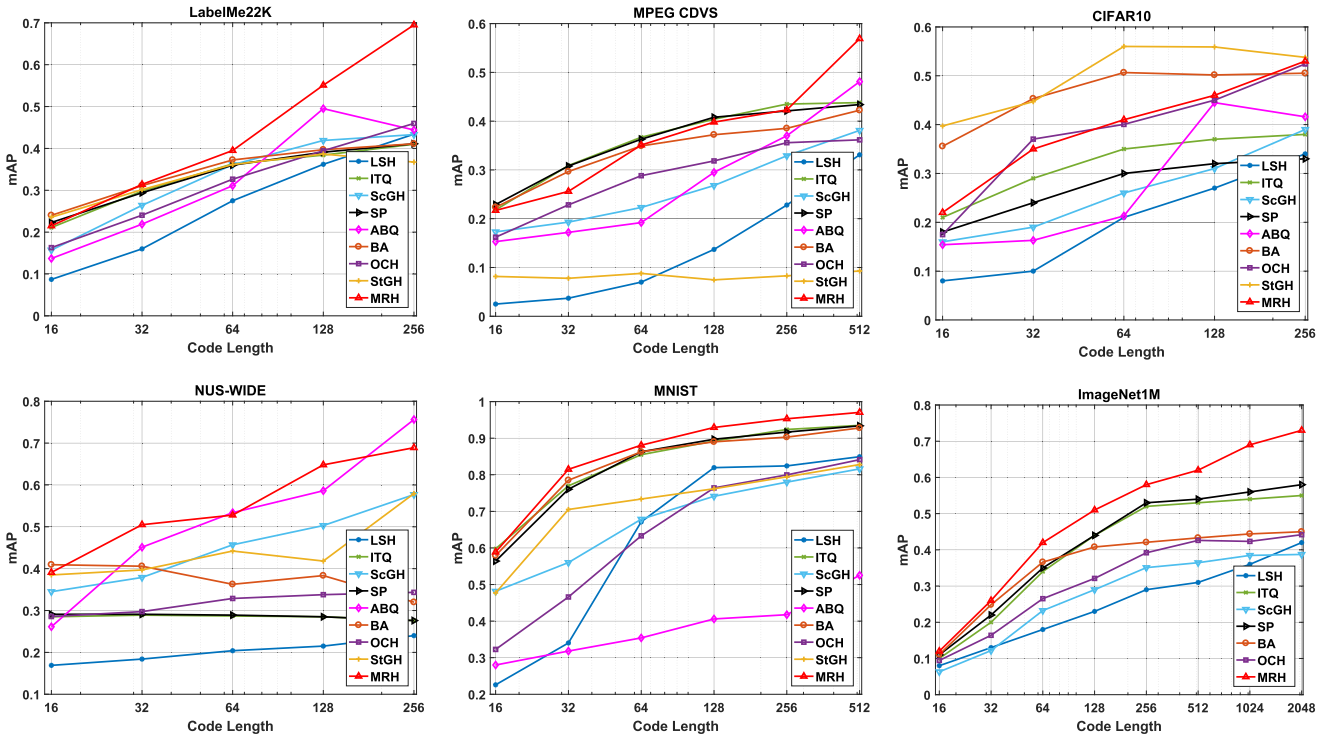
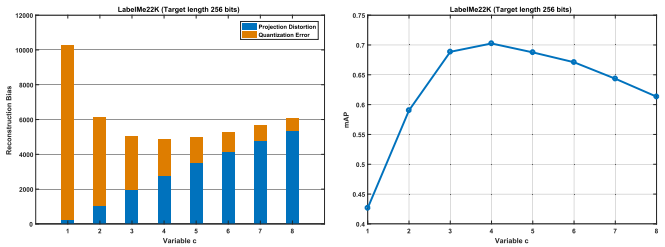


Fig. 7. Results of mAP of state-of-the-art hashing methods on various benchmark datasets (best viewed in high-resolution color display).

at low bitrates (as illustrated in Table I), in which the MRH method degenerates into PCA plus sign hashing function, and the complementary effects of quantization and projection is inhibited. However, the optimal c at 512 bits was set to 2, leading to a significant reduction of quantization error with double bit hashing enhanced by MRH, and thus causing a rapid performance growth from 256 to 512 bits. BA and StGH outperform MRH at small bitrates on CIFAR10 dataset, but their performances stops growing when the bitrate is greater than 64. In contrast, MRH's performance constantly increases as the code length increases, and is comparable with BA and StGH when the bitrate reaches 256 bits.



(a) (b)

Fig. 8. Results of reconstruction bias derived from (5) and the corresponding mAP results when setting different c values. The experiment is conducted on the LabelMe22K dataset at target code length of 256 bits. (a) Reconstruction Bias. (b) Search mAP.

D. Discussion

1) The Rationale of the Learning Objective: Fig. 8 shows the overall reconstruction error derived in Eq.(5) and the corresponding mAP results for learning 256 bits codes on the LabelMe22K dataset. Notice that the variation of search mAP at different quantization bit number c is perfectly synchronized with that of our learning objective. Thus it is reasonable to claim that the reconstruction bias is indeed a good indicator of actual ANN search performance, which further supported the rationale of our objective function along with the lower-bound analysis in Theorem 4.

2) The Impact of Projection Dimensionality: Fig. 9 shows the impact of projection dimensionality for learning 512 and 1024 bits codes over ImageNet1M. Variable c produces balancing effects on the projection and quantization stages. Increasing c reduces the quantization error but incurs more projection distortions, and vice versa. There does exist a trade-off between projection and quantization. From Fig. 9, the best

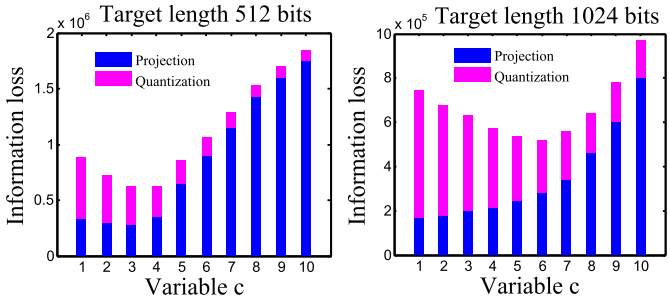


Fig. 9. Results of reconstruction bias derived from the learning objective function Eq. (5) when setting different c values.

setting is $c = 4$ for 512 bits and $c = 6$ for 1024 bits. We notice that the MRH tends to set c to a large value for long size codes, which means that more bits are allowed for quantizing

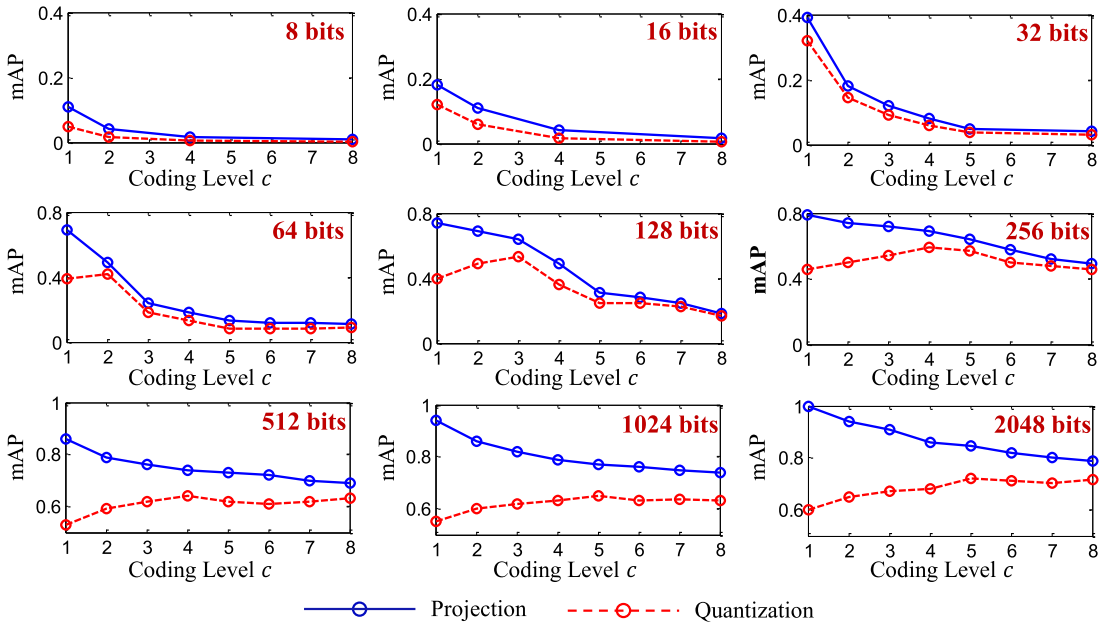


Fig. 10. The intermediate results of our MRH with the different coding level c for learning binary codes from 8 bits to 2048 bits on the ImageNet1M dataset. The blue line indicates the performance when data points are projected into subspace $\lfloor \frac{k}{c} \rfloor$ (prior to quantization). The dotted red line indicates the performance after quantization stage with c bits coding for each projection value.

TABLE I

THE OPTIMAL VALUE OF c FOR LEARNING BINARY CODES WITH DIFFERENT LENGTH ON THE CIFAR10, LABELME22K, MNIST, NUS-WIDE, MNIST AND IMAGENET1M DATASETS

Dataset	Code Length 2^n							
	$\leq 2^4$	2^5	2^6	2^7	2^8	2^9	2^{10}	2^{11}
LabelMe22K	1	2	2	3	4	-	-	-
CIFAR10	1	1	2	3	4	-	-	-
NUS-WIDE	1	1	1	2	2	-	-	-
MPEG CDVS	1	1	1	1	1	2	-	-
MNIST	1	1	1	1	2	3	-	-
ImageNet1M	1	1	2	3	4	4	6	8

the values of each projection element. By adaptively adjusting the projection dimensionality, the MRH obtains discriminative codes with overall minimal information loss. Table I lists the optimal settings of c on all six datasets.

To better visualize the impact of projection dimensionality, we show the intermediate results of our MRH with the different coding level c on the ImageNet1M dataset, as depicted in Fig. 10. We can see that single bit coding largely degrades the performance and higher coding level can consistently reduce the performance drop in the quantization stage. For the short codes such as 8 bits, 16 bits and 32 bits, the MRH achieves the best performance when $c = 1$. For relatively long codes ($b > 32$), $c = 1$ is not the best choice. For example, at code length 256 bits, when $c = 1$ the data points would be projected to 256-D (with 0.791 mAP). After the quantization, the performance drops down to 0.462. A better way is first to project data points to 64-D and then use $c = 4$ bits to quantize each projection. Our method can adaptively adjust the projection dimensionality by balancing the information loss during projection and quantization stage.

3) The Relationship Between Projection and Quantization: To further analyze the characteristics of the proposed joint optimization framework, we fix the projection dimension and observe the mAP results on the challenging MPEG CDVS dataset by setting different number of quantization bits c. The results are shown in Table II. Note that when the projection dimension is fixed, the search performance first increases together with the variable c, but starts dropping beyond a certain threshold c_0 that becomes larger as projection dimension grows. In our experiments, $c_0 = 2, 3, 5, 6, 6$ at projection dimension range from 100 to 500. However, since the projection dimension is fixed, the quantization error always decreases whenever c grows, which means the total reconstruction error is also constantly decreasing.

A possible explanation to this abnormality could be that the data suffers from severe distortions at low projection dimensions, while, over the course of MRH optimization, the "over" refinement of the quantization stage actually amplifies two contradicting factors: the preservation of original information and the aggravation of distortions. The observed threshold c_0 can be interpreted as the "critical point" at which the marginal influences of these two factors cancel out, and for any $c > c_0$, quantize the distorted data at c bits per dimension would lead to negative impacts on the search accuracy with such "over-quantized" codes.

Moreover, we argue that the projection and quantization stages are inherently correlated regardless of the explicit bitrate constraints (although the bitrate constraints lead to nice unimodal property that aids fast ternary search), and to simply optimize quantization stage alone can sometimes leads to negative results as opposed to expectations. This further justifies the proposed framework in which we seek the balance between projection distortions and quantization errors by adaptively adjusting the projection dimensionality under different bitrate constraints. In addition, it should be noted that the mutual relationship between projection and quantization, and their impact on each other can provide a new perspective to view traditional hashing methods but also

TABLE II
RESULTS OF MAP ON MPEG CDVS DATASET UNDER DIFFERENT PROJECTION DIMENSIONS AND QUANTIZATION BIT NUMBERS

Projection Dimension	Quantization bits per dimension					
	$c = 1$	2	3	4	5	6
100	0.2218	**0.2516**	0.2354	0.2187	0.1995	0.1876
200	0.2676	0.3709	**0.3929**	0.3750	0.3588	0.3509
300	0.2795	0.4338	0.5117	0.5271	**0.5364**	0.5286
400	0.2908	0.4673	0.5761	0.6353	0.6651	**0.6850**
500	0.2893	0.4725	0.5893	0.6654	0.7149	**0.7513**

open up research issues in hashing. For example, the joint optimization framework or its idea of leveraging positive complementary effects can be integrated into deep learning based hashing methods, so that more robust loss functions could be explored to better characterize and utilize these inherent relationships.

VIII. CONCLUSION

In this paper, we have proposed a novel hashing method called Minimal Reconstruction Bias Hashing (MRH) for learning compact binary codes. We interpreted the problem of maximizing similarity preservation of binary codes from the perspective of minimizing the reconstruction error, and presented a joint optimization framework to balance the trade-off between projection and quantization stages with flexible projection dimensionality. Moreover, we have introduced a lower-bound analysis to establish the relationship between the reconstruction bias and Hamming approximation error, justifying the learning objective of our MRH method. By analyzing the unimodality of the objective function with respect to projection dimensionality, a fast ternary search algorithm was introduced to determine the optimal solution in the sub-linear time. Extensive experimental results over eight benchmark datasets demonstrate the superiority of our method against the state-of-the-art methods in terms of ANN search accuracy.

REFERENCES

[1] Y. Liu, F. Wu, Y. Yang, Y. Zhuang, and A. G. Hauptmann, "Spline regression hashing for fast image search," *IEEE Trans. Image Process.*, vol. 21, no. 10, pp. 4480–4491, Oct. 2012.

[2] Y. Gao, M. Shi, D. Tao, and C. Xu, "Database saliency for fast image retrieval," *IEEE Trans. Multimedia*, vol. 17, no. 3, pp. 359–369, Mar. 2015.

[3] J. Yu, D. Tao, M. Wang, and Y. Rui, "Learning to rank using user clicks and visual features for image retrieval," *IEEE Trans. Cybern.*, vol. 45, no. 4, pp. 767–779, Apr. 2015.

[4] Y. Weiss, A. Torralba, and R. Fergus, "Spectral hashing," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2009, pp. 1753–1760.

[5] A. Torralba, R. Fergus, and Y. Weiss, "Small codes and large image databases for recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2008, pp. 1–8.

[6] Y. Gong, S. Lazebnik, A. Gordo, and F. Perronnin, "Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 12, pp. 2916–2929, Dec. 2013.

[7] L.-Y. Duan *et al.*, "Overview of the MPEG-CDVS standard," *IEEE Trans. Image Process.*, vol. 25, no. 1, pp. 179–194, Jan. 2016.

[8] W. Kong and W.-J. Li, "Double-bit quantization for hashing," in *Proc. Conf. Artif. Intell. (AAAI)*, 2012, vol. 1. no. 2, p. 5.

[9] W. Kong and W.-J. Li, "Isotropic hashing," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2012, pp. 1646–1654.

[10] Z. Wang, L.-Y. Duan, T. Huang, and W. Gao, "Affinity preserving quantization for hashing: A vector quantization approach to learning compact binary codes," in *Proc. Conf. Artif. Intell. (AAAI)*, 2016, pp. 1102–1108.

[11] A. Andoni and P. Indyk, "Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions," in *Proc. IEEE Symp. Found. Comput. Sci.*, Oct. 2006, pp. 459–468.

[12] B. Kulis and K. Grauman, "Kernelized locality-sensitive hashing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 6, pp. 1092–1104, Jun. 2012.

[13] X. Liu, J. He, and S.-F. Chang, "Hash bit selection for nearest neighbor search," *IEEE Trans. Image Process.*, vol. 26, no. 11, pp. 5367–5380, Nov. 2017.

[14] W. Liu, J. Wang, S. Kumar, and S.-F. Chang, "Hashing with graphs," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2011, pp. 1–8.

[15] B. Xu, J. Bu, Y. Lin, C. Chen, X. He, and D. Cai, "Harmonious hashing," in *Proc. Int. Joint Conf. Artif. Intell. (IJCAI)*, 2013, pp. 1820–1826.

[16] W. Liu, J. Wang, Y. Mu, S. Kumar, and S.-F. Chang, "Compact hyperplane hashing with bilinear functions," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2012, pp. 17–24.

[17] Y. Gong, S. Kumar, H. A. Rowley, and S. Lazebnik, "Learning binary codes for high-dimensional data using bilinear projections," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 484–491.

[18] L. Liu, M. Yu, and L. Shao, "Projection bank: From high-dimensional data to medium-length binary codes," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2821–2829.

[19] F. X. Yu, S. Kumar, Y. Gong, and S.-F. Chang, "Circulant binary embedding," in *Proc. 31st Int. Conf. Mach. Learn. (ICML)*, 2014, pp. 946–954.

[20] Y. Xia, K. He, P. Kohli, and J. Sun, "Sparse projections for high-dimensional binary codes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3332–3339.

[21] W. Liu, C. Mu, S. Kumar, and S.-F. Chang, "Discrete graph hashing," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2014, pp. 3419–3427.

[22] Q. Y. Jiang and W. J. Li, "Scalable graph hashing with feature transformation," in *Proc. Int. Joint Conf. Artif. Intell. (IJCAI)*, 2015, pp. 2248–2254.

[23] Y. Guo, G. Ding, L. Liu, J. Han, and L. Shao, "Learning to hash with optimized anchor embedding for scalable retrieval," *IEEE Trans. Image Process.*, vol. 26, no. 3, pp. 1344–1354, Mar. 2017.

[24] F. Shen, C. Shen, Q. Shi, A. V. D. Hengel, Z. Tang, and H. T. Shen, "Hashing on nonlinear manifolds," *IEEE Trans. Image Process.*, vol. 24, no. 6, pp. 1839–1851, Jun. 2015.

[25] H. Liu, R. Wang, S. Shan, and X. Chen, "Deep supervised hashing for fast image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2064–2072.

[26] Q. Li, Z. Sun, R. He, and T. Tan. (2017). "Deep supervised discrete hashing." [Online]. Available: https://arxiv.org/abs/1705.10999

[27] Z. Wang, L.-Y. Duan, J. Lin, X. Wang, T. Huang, and W. Gao, "Hamming compatible quantization for hashing," in *Proc. Int. Joint Conf. Artif. Intell. (IJCAI)*, 2015, pp. 2298–2304.

[28] J.-P. Heo, Y. Lee, J. He, S.-F. Chang, and S.-E. Yoon, "Spherical hashing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 2957–2964.

[29] K. He, F. Wen, and J. Sun, "K-means hashing: An affinity-preserving quantization method for learning binary compact codes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 2938–2945.

[30] Z. Li, X. Liu, J. Wu, and H. Su, "Adaptive binary quantization for fast nearest neighbor search," in *Proc. Eur. Conf. Artif. Intell. (ECAI)*, 2016, pp. 64–72.

[31] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2010, pp. 143–156.

[32] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2009, pp. 248–255.

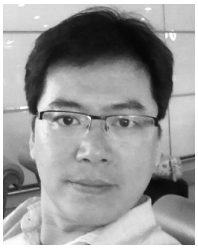
[33] Z. Wang, L.-Y. Duan, J. Yuan, T. Huang, and W. Gao, "To project more or to quantize more: Minimizing reconstruction bias for learning compact binary codes," in *Proc. Int. Joint Conf. Artif. Intell. (IJCAI)*, 2016, pp. 2181–2188.

[34] H. Jegou, M. Douze, and C. Schmid, "Product quantization for nearest neighbor search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 1, pp. 117–128, Jan. 2011.

[35] T. Ge, K. He, Q. Ke, and J. Sun, "Optimized product quantization for approximate nearest neighbor search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 2946–2953.

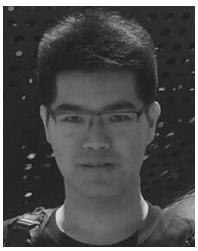
[36] J. Wang, J. Wang, J. Song, X.-S. Xu, H. T. Shen, and S. Li, "Optimized Cartesian k-means," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 1, pp. 180–192, Jan. 2015.

[37] Q. Ning, J. Zhu, Z. Zhong, S. C. H. Hoi, and C. Chen, "Scalable image retrieval by sparse product quantization," *IEEE Trans. Multimedia*, vol. 19, no. 3, pp. 586–597, Mar. 2017.
[38] Y. Cao, M. Long, J. Wang, and S. Liu, "Deep visual-semantic quantization for efficient image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 916–925.
[39] J. Li, X. Lan, X. Li, J. Wang, N. Zheng, and Y. Wu, "Online variable coding length product quantization for fast nearest neighbor search in mobile retrieval," *IEEE Trans. Multimedia*, vol. 19, no. 3, pp. 559–570, Mar. 2017.
[40] X. Wang, T. Zhang, G.-J. Qi, J. Tang, and J. Wang, "Supervised quantization for similarity search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2018–2026.
[41] M. Long, Y. Cao, J. Wang, and P. S. Yu, "Composite correlation quantization for efficient multimodal retrieval," in *Proc. 39th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2016, pp. 579–588.
[42] K. Lin, J. Lu, C.-S. Chen, and J. Zhou, "Learning compact binary descriptors with unsupervised deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1183–1192.
[43] X. Yan, L. Zhang, and W.-J. Li, "Semi-supervised deep hashing with a bipartite graph," in *Proc. Int. Joint Conf. Artif. Intell. (IJCAI)*, 2017, pp. 3238–3244.
[44] M. Á. Carreira-Perpiñán and R. Raziperchikolaei, "Hashing with binary autoencoders," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 557–566.
[45] J. Wang, W. Liu, S. Kumar, and S. F. Chang, "Learning to hash for indexing big data—A survey," *Proc. IEEE*, vol. 104, no. 1, pp. 34–57, Jun. 2016.
[46] J. Wang *et al.*, "A survey on learning to hash," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 769–790, Apr. 2018.
[47] M. Raginsky and S. Lazebnik, "Locality-sensitive binary codes from shift-invariant kernels," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2009, pp. 1509–1517.
[48] Y. Wang, X. Lin, L. Wu, W. Zhang, and Q. Zhang, "LBMCH: Learning bridging mapping for cross-modal hashing," in *Proc. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2015, pp. 999–1002.
[49] Y. Wang, X. Lin, L. Wu, W. Zhang, Q. Zhang, and X. Huang, "Robust subspace clustering for multi-view data by exploiting correlation consensus," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3939–3949, Nov. 2015.
[50] L. Wu and Y. Wang, "Robust hashing for multi-view data: Jointly learning low-rank kernelized similarity consensus and hash functions," *Image Vis. Comput.*, vol. 57, pp. 58–66, Jan. 2017.
[51] H. Liu, R. Ji, Y. Wu, and F. Huang, "Ordinal constrained binary code learning for nearest neighbor search," in *Proc. Conf. Artif. Intell. (AAAI)*, 2017, pp. 2238–2244.
[52] R. Xia, Y. Pan, H. Lai, C. Liu, and S. Yan, "Supervised hashing for image retrieval via image representation learning," in *Proc. Conf. Artif. Intell. (AAAI)*, vol. 1. 2014, p. 2.
[53] B. Dai, R. Guo, S. Kumar, N. He, and L. Song. (2017). "Stochastic generative hashing." [Online]. Available: https://arxiv.org/abs/1701.02815
[54] L. Liu, M. Yu, and L. Shao, "Learning short binary codes for large-scale image retrieval," *IEEE Trans. Image Process.*, vol. 26, no. 3, pp. 1289–1299, Mar. 2017.
[55] F. Shen, X. Zhou, Y. Yang, J. Song, H. T. Shen, and D. Tao, "A fast optimization method for general binary code learning," *IEEE Trans. Image Process.*, vol. 25, no. 12, pp. 5610–5621, Dec. 2016.
[56] W. Kong, W.-J. Li, and M. Guo, "Manhattan hashing for large-scale image retrieval," in *Proc. 35th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2012, pp. 45–54.
[57] A. Gionis, P. Indyk, and R. Motwani, "Similarity search in high dimensions via hashing," in *Proc. Int. Conf. Very Large Data Bases*, 1999, pp. 518–529.
[58] H. G. Musmann, "Predictive image coding," in *Image Transmission Techniques*, W. K. Pratt, Ed. New York, NY, USA: Academic, 1979, pp. 73–112.
[59] Z. Wen and W. Yin, "A feasible method for optimization with orthogonality constraints," *Math. Program.*, vol. 142, no. 1, pp. 397–434, 2013.
[60] G. D. Smith, *Numerical Solution of Partial Differential Equations*, 3rd ed. Oxford, U.K.: Clarendon, Dec. 1985.
[61] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. New York, NY, USA: Springer, 2001.
[62] B. Kulis and T. Darrell, "Learning to hash with binary reconstructive embeddings," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2009, pp. 1042–1050.
[63] A. Krizhevsky, "Learning multiple layers of features from tiny images," Univ. Toronto, Toronto, ON, Canada, Tech. Rep., Apr. 2009.
[64] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
[65] T. S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. Zheng, "Nus-wide: A real-world Web image database from national university of Singapore," in *Proc. ACM Int. Conf. Image Video Retr.*, 2009, p. 48.
[66] Y. Lou *et al.*, "Compact deep invariant descriptors for video retrieval," in *Proc. Data Compress. Conf. (DCC)*, Apr. 2017, pp. 420–429.



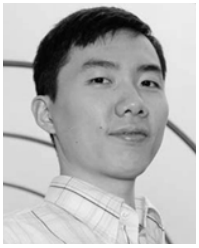
Ling-Yu Duan (M'06) has been serving as the Associate Director of the Rapid-Rich Object Search Laboratory, a joint laboratory between Nanyang Technological University, Singapore, and Peking University (PKU), China, since 2012. He is currently a Full Professor with the National Engineering Laboratory of Video Technology, PKU. His research interests include multimedia indexing, search, and retrieval, mobile visual search, visual feature coding, and video analytics. He received the *EURASIP Journal on Image and Video Processing* Best Paper Award in 2015, the Ministry of Education Technology Invention Award (first prize) in 2016, the National Technology Invention Award (second prize) in 2017, the China Patent Award for Excellence in 2017, and the National Information Technology Standardization Technical Committee Standardization Work Outstanding Person Award in 2015. He was a Co-Editor of MPEG Compact Descriptor for Visual Search Standard (ISO/IEC 15938-13). He is serving as a Co-Chair of MPEG Compact Descriptor for Video Analytics. He is currently an Associate Editor of the *ACM Transactions on Intelligent Systems and Technology* and the *ACM Transactions on Multimedia Computing, Communications, and Applications*.

Yuwei Wu received the Ph.D. degree in computer science from the Beijing Institute of Technology (BIT), Beijing, China, in 2014. From 2014 to 2016, he was a Post-Doctoral Research Fellow with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. He is currently an Assistant Professor with the School of Computer Science, BIT. He has strong research interests in computer vision and information retrieval. He received the Outstanding Ph.D. Thesis Award from BIT and a Distinguished Dissertation Award Nominee from the China Association for Artificial Intelligence.



Yicheng Huang received the bachelor's degree in computer science and technology from Peking University, Beijing, China, in 2015, where he is currently pursuing the M.S. degree with the School of Electrical Engineering and Computer Science. His current research interests include large-scale image retrieval and fast nearest neighbor search.

Zhe Wang received the bachelor's degree in software engineering from Beijing Jiaotong University, China, in 2012, and the master's degree in computer science from the School of Electrical Engineering and Computer Science, Peking University, China, in 2015. He was with the ROSE Laboratory, Nanyang Technological University, Singapore. He is currently affiliated with the Institute of Digital Media, Peking University. His research interests include large-scale image retrieval and fast approximate nearest neighbor search.



Junsong Yuan (M'08–SM'14) received Ph.D. from Northwestern University in 2009 and M.Eng. from National University of Singapore in 2005. Before that, he graduated from the Special Class for the Gifted Young of Huazhong University of Science and Technology (HUST), Wuhan, China, in 2002. He is currently an associate professor at Department of Computer Science and Engineering (CSE), the State University of New York, Buffalo, USA. Before that he was an associate professor at School of Electrical and Electronics Engineering (EEE), Nanyang Technological University (NTU), Singapore. He received 2016 Best Paper Award from IEEE Transactions on Multimedia, Doctoral Spotlight Award from IEEE Conference on Computer Vision and Pattern Recognition (CVPR'09), Nanyang Assistant Professorship from NTU, and Outstanding EECS Ph.D. Thesis award from Northwestern University. He is currently Senior Area Editor of *Journal of Visual Communication and Image Representation* (JVCI), Associate Editor of IEEE TRANSACTIONS ON IMAGE PROCESSING (T-IP), IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY (T-CSVT), and served as Guest Editor of *International Journal of Computer Vision* (IJCV). He is Program Co-chair of ICME'18 and VCIP'15, and Area Chair of ACM MM'18, ACCV'18'14, ICPR'18'16, CVPR'17, ICIP'18'17 etc.

Wen Gao (M'92–SM'05–F'09) received the Ph.D. degree in electronics engineering from THE University of Tokyo, Tokyo, Japan, in 1991. He was a Professor of computer science with the Harbin Institute of Technology, Harbin, China, from 1991 to 1995, and a Professor with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. He is currently a Professor of computer science with the Institute of Digital Media, School of Electronic Engineering and Computer Science, Peking University, Beijing. He has chaired a number of prestigious international conferences on multimedia and video signal processing, such as IEEE ICME and ACM Multimedia, and served on the advisory and technical committees of numerous professional organizations. He served or serves on the Editorial Boards for several journals, such as the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, the IEEE TRANSACTIONS ON MULTIMEDIA, the IEEE TRANSACTIONS ON AUTONOMOUS MENTAL DEVELOPMENT, the *EURASIP Journal of Image Communications*, and the *Journal of Visual Communication and Image Representation*.