International Journal of Computer Vision
https://doi.org/10.1007/s11263-020-01331-0

Weakly-supervised Semantic Guided Hashing for Social Image Retrieval

Zechao Li[1] · Jinhui Tang[1] · Liyan Zhang[2] · Jian Yang[1]

Received: 19 April 2019 / Accepted: 9 April 2020

Abstract
Hashing has been widely investigated for large-scale image retrieval due to its search effectiveness and computation efficiency. In this work, we propose a novel Semantic Guided Hashing method coupled with binary matrix factorization to perform more effective nearest neighbor image search by simultaneously exploring the weakly-supervised rich community-contributed information and the underlying data structures. To uncover the underlying semantic information from the weakly-supervised user-provided tags, the binary matrix factorization model is leveraged for learning the binary features of images while the problem of imperfect tags is well addressed. The uncovered semantic information enables to well guide the discrete hash code learning. The underlying data structures are discovered by adaptively learning a discriminative data graph, which makes the learned hash codes preserve the meaningful neighbors. To the best of our knowledge, the proposed method is the first work that incorporates the hash code learning, the semantic information mining and the data structure discovering into one unified framework. Besides, the proposed method is extended to one deep approach for the optimal compatibility of discriminative feature learning and hash code learning. Experiments are conducted on two widely-used social image datasets and the proposed method achieves encouraging performance compared with the state-of-the-art hashing methods.

Keywords Hashing · Image retrieval · Matrix factorization · Social image · Discrete code

1 Introduction

In recent years, we have witnessed an increasing number of high-dimensional multimedia data including images and videos, which makes the visual similarity search attract increasing attention. Due to the search effectiveness and computation efficiency, hashing has been widely studied for large-scale image retrieval (Indyk and Motwani 1998; Weiss et al. 2008; Tang et al. 2015; Liu et al. 2016, 2017; Sohn et al. 2017; Gui et al. 2018; Hu et al. 2019; Liu et al. 2017; Jin et al. 2019). The previous methods mainly explore the visual structures or the manually-annotated information, which leads to the unsatisfied performance or the expensive cost. Fortunately, social multimedia data are always associated with rich user-provided information, such as tags (Tang et al. 2017; Gong et al. 2013; Li and Tang 2017). These user-provided tags can to some extent character the corresponding semantic information of the visual content, which is helpful for hashing. Therefore, it is essential to explore the rich community-contributed information for hashing.

Hashing is to uncover a Hamming space by learning hash functions that similar images have the similar binary hash codes in the Hamming space (Liu et al. 2016). To generate hash functions, the well-known Locality Sensitive Hashing (LSH) (Indyk and Motwani 1998) is proposed by using random locality sensitive projections. Some improved methods are proposed, such as Kernelized LSH (Kulis and Grauman 2009), Shift-invariant kernel hashing (Raginsky and Lazebnik 2009) and Super-bit LSH (Ji et al. 2012).

Communicated by Li Liu, Matti Pietikäinen, Jie Qin, Jie Chen, Wanli Ouyang, Luc Van Gool.

✉ Jinhui Tang
jinhuitang@njust.edu.cn

Zechao Li
zechao.li@njust.edu.cn

Liyan Zhang
zhangliyan@nuaa.edu.cn

Jian Yang
csjyang@njust.edu.cn

[1] School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China

[2] College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China

Published online: 12 May 2020

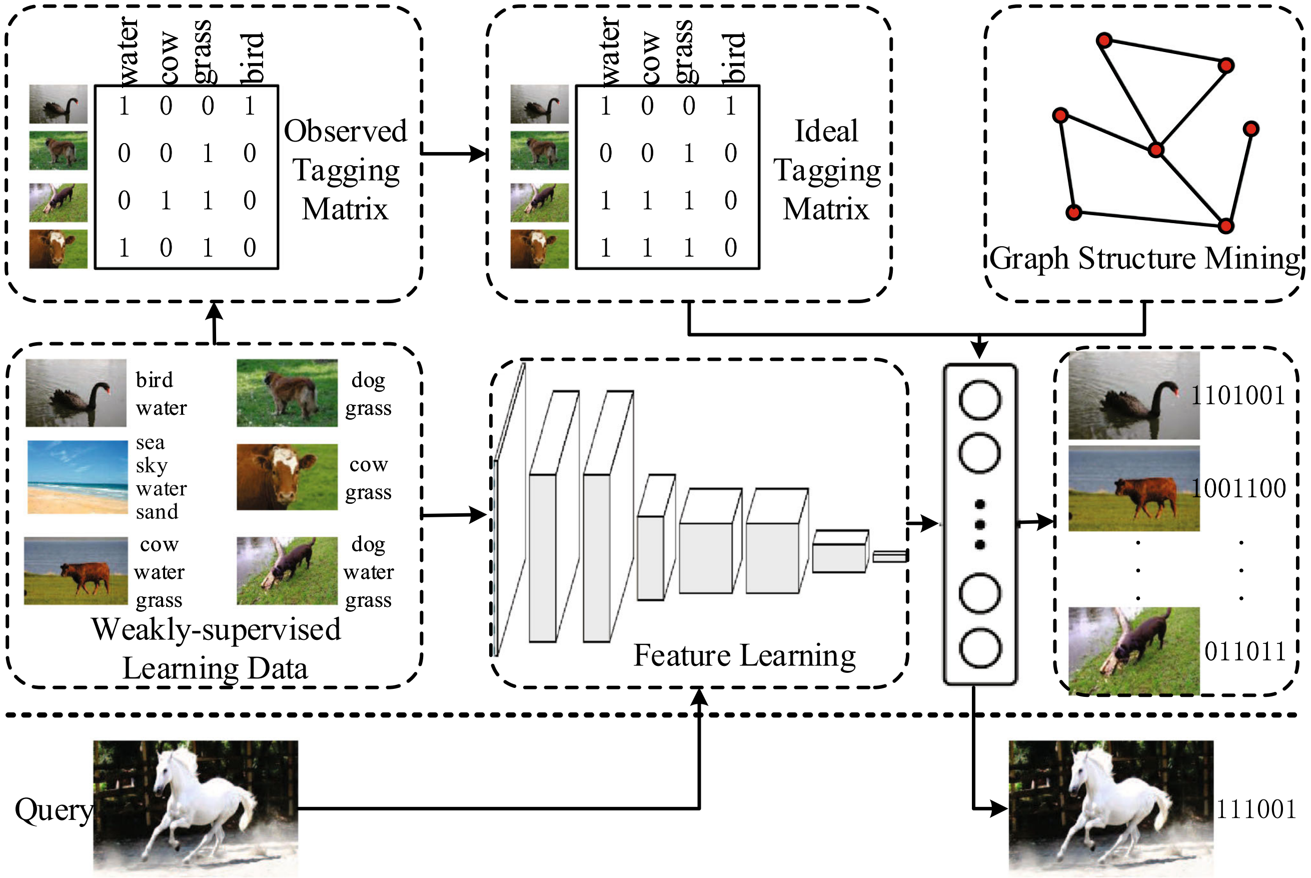


Fig. 1 The framework of the proposed hashing method for social image retrieval. The ideal binary tagging matrix learning, the graph structure mining, the feature learning and hashing are jointly incorporated into one unified framework

These methods without considering the data structure are termed as data-independent methods. To leverage the data structure, data-dependent methods are proposed, such as Spectral Hashing (SH) (Weiss et al. 2008), Neighborhood Discriminant Hashing (NDH) (Tang et al. 2015) and Spherical Hashing (Heo et al. 2015). They learn hash functions by preserving the data structure or predicting the manually-annotated labels. Deep hashing methods (Liong et al. 2015; Wang et al. 2015; Liu et al. 2016; Li et al. 2017; Cao et al. 2017, 2018; Dizaji et al. 2018; Jin et al. 2019) are proposed to leverage the deep neural networks. Recently, some hashing methods have been proposed with the help of the user-provided tags (Zhuang et al. 2011; Zhu et al. 2013; Zhang and Li 2014; Wang et al. 2015; Tang and Li 2018; Guan et al. 2018). However, the user-provided tags are often incomplete, subjective, or noisy. Thus, hash functions are learned by addressing the tag problem simultaneously. For example, in (Tang and Li 2018), the local discriminative information and the structure of the visual space are jointly leveraged to avoid overfitting the noisy tags (Li and Tang 2015). However, the underlying semantic information of the user-provided tags is not well explored, which can significantly beneficial to hash code learning. Besides, the graph structure explored by the previous works is pre-defined, which may contain some noise.

Towards this end, we propose a new hashing method by uncovering the semantic information from the weakly-supervised user-provided tags to guide the hashing learning, termed as Semantic Guided Hashing (SGH), as illustrated in Fig. 1. By well exploring the weakly-supervised tagging information, the Binary Matrix Factorization (BMF) model is leveraged to uncover the underlying semantic information. To address the problem of imperfect tags, an ideal tagging matrix is learned with the nonnegative and discrete constraints. BMF can well explore the the most important integer property of the original binary tagging matrix while hashing is to learn binary codes. Thus, they have some latent universality. This inspires us to incorporate BMF and hashing into one unified framework with a shared binary space. Therefore, the proposed method learns a unified latent space by assuming that BMF and hashing learning share a common space, which can seamlessly incorporate BMF and hashing learning. The uncovered semantic information enables to well guide the discrete hash code learning. Besides, the heterogeneous data are also seamlessly connected with this assumption. To improve the discriminative ability of hash codes, the underlying data structures are discovered by adaptively learning a discriminative data graph and the geometric structure of data is preserved. For the optimal compatibility of discriminative feature learning and hash code learning, the proposed SGH is extended to the deep model. To the best of our knowledge, the proposed method is the first work that simultaneously and seamlessly incorporates the feature learning, hash code learning, semantic information mining and data structure discovering into one unified framework. To verify the effectiveness of the proposed method for social image retrieval, extensive experiments are conducted on two widely-used social image benchmarks, which show that the proposed approach outperforms the state-of-the-art hashing methods.

Contribution. The main contributions of this work are summarized as follows.

- We propose a novel weakly-supervised Semantic Guided Hashing (SGH) method by collaboratively exploring the heterogeneous information for social image retrieval. To our best knowledge, it is the first attempt to simultaneously and seamlessly incorporate the hash code learning, the semantic information mining and the data structure discovering.
- The weakly-supervised binary matrix factorization and hashing learning are assumed to share a common space, which can well guide the discrete hash code learning.
- A discriminative data graph is adaptively learned to improve the discriminative ability of hash codes.

The structure of this paper is structured as follows. The previous methods are surveyed in Sect. 2. Section 3 explains the proposed hashing method in details. The optimization algorithm is presented in Sect. 4.The extension to the deep model is presented in Sect. 5. Experiments are conducted and discussed in Sect. 6 followed with conclusions in Sect. 7.

2 Related Work

In this section, we briefly review the related hashing methods for image retrieval. These hashing methods can be roughly divided into two groups: data-independent and data-dependent. Please refer to (Wang et al. 2018) for the survey.

As one of the most well-known data-independent hashing methods, Locality Sensitive Hashing (LSH) (Indyk and Motwani 1998) randomly generates hash codes based on the assumption that two similar images have a higher probability of collision. Some improved methods, such as Kernelized LSH (Kulis and Grauman 2009), Shift-invariant kernel hashing (Raginsky and Lazebnik 2009) and Super-bit LSH (Ji et al. 2012), are proposed to improve the performance. Due to the data distribution ignoring, the performance is still unsatisfied. And it can achieve better results by increasing the number of hash codes or hash tables.

Many data-dependent methods have been proposed to learn hash functions from data. Some unsupervised hashing methods have been developed by exploring the data structures. Spectral Hashing (SH) (Weiss et al. 2008) learns hash functions by preserving the visual similarity. Anchor Graph Hashing (AGH) (Liu et al. 2011) is proposed to address the large-scale problem and introduce the anchor graph for scalable graph-based hashing learning. Binary Reconstructive Embedding (BRE) (Kulis and Darrell 2009) is proposed for learning hash functions by minimizing the reconstruction error between the original distances and the Hamming distances. Heo et al. (2012) proposed to learn hash functions by mapping more spatially coherent data points into a binary code. Iterative Quantization (ITQ) (Gong et al. 2013) learns hash functions by maximizing the data variance of each hash bit via PCA projection. Inductive Hashing (Shen et al. 2013) is proposed to learn hash codes on the intrinsic manifold. Neighborhood Discriminant Hashing (NDH) (Tang et al. 2015) learns hash functions by preserving the neighborhood discriminative information in the Hamming space. The above methods only explore the data structures without using the supervised information. To explore the labeled and unlabeled data, semi-supervised hashing methods are proposed (Wang et al. 2012; Kan et al. 2014). To improve the discriminative ability, supervised hashing methods utilize different types of supervised information to learn hash functions (Zhang et al. 2012; Shen et al. 2015; Gui et al. 2018; Zhai et al. 2018; Gui and Li 2018; Mandal et al. 2019). In Zhang et al. (2012), Self-Taught Hashing (STH) is proposed to learn bit-specific linear Support Vector Machines (SVM) by using the labeled data. In Shen et al. (2015), a discrete supervised hashing method is proposed to learn hash functions with the discrete constraints while an improved method is proposed in Gui et al. (2018) to accelerate the training.

To utilize the powerful feature learning ability of deep learning (Gordo et al. 2017), deep hashing methods have been proposed (Xia et al. 2014; Liong et al. 2015; Wang et al. 2015; Liu et al. 2016; Lin et al. 2015; Li et al. 2016; Zhang et al. 2016; Zhu et al. 2016; Li et al. 2017; Venkateswara et al. 2017; Liong et al. 2017; Tang et al. 2018; Cao et al. 2018; Dizaji et al. 2018; Zhang et al. 2018; Jiang et al. 2018; Tang et al. 2018; Jin et al. 2019). In Xia et al. (2014), a two-stage method is proposed by first learning hash codes based on the semantic similarity matrix and then learning hash functions and neural networks based on the learned hash codes. Supervised Deep Hashing (SDH) (Liong et al. 2015) is proposed to minimize the quantization errors between the hash codes and the learned deep features. In Liu et al. (2016), Deep Supervised Hashing (DSH) preserves the pair-wise similarity based on convolution neural network for hashing learning. Deep pairwise-supervised hashing (DPSH) performs simultaneous feature learning and hash code learning for applications with pairwise labels in Li et al. (2016). In Cao et al. (2018), Deep Cauchy Hashing (DCH) learns hash codes by using a pairwise cross-entropy loss based on Cauchy distribution. HashGAN (Dizaji et al. 2018) is proposed to learn binary codes of images with three networks (a generator, a discriminator and an encoder). HashNet is proposed by using the continuation strategy for low quantization error to balance the supervised pairs (Cao et al. 2017).

Social images are always associated with rich community-contributed information, and some hashing methods are proposed to explore the community-contributed information (Zhuang et al. 2011; Tang et al. 2019; Zhu et al. 2013; Wu et al. Wu et al.; Guan et al. 2018; Tang and Li 2018). In Zhuang et al. (2011), hashing is performed by exploring the visual similarity and social relationships based on a hypergraph. A joint multi-modal dictionary learning is used to preserve the intra-modality similarity and inter-modality similarity for hashing (Wu et al. Wu et al.). In Tang and Li (2018), hash functions are learned by exploring the similarities in the visual and textual domains. A weakly-supervised deep hashing method is proposed by using two stages including weakly-supervised pre-training and supervised fine-tuning in Guan et al. (2018). The above methods mainly focus on using the user-provided tags without considering to uncover the underlying semantic information. The performance can be easily degraded by the imperfect tags.

Due to that the user-provided tagging matrix is binary and hashing is to learn binary codes to represent images, it is reasonable to jointly explore binary matrix factorization and hashing in one unified framework. Thus, this paper proposes a novel weakly-supervised hashing method by learning hashing functions coupled with binary matrix factorization. The underlying semantic information is mined by the binary matrix factorization model with the ideal tagging matrix learning. By leveraging the tagging information and image content, the uncovered semantic information enables to well guide the discrete hash code learning. The proposed method is also extended to one deep approach, which is the main focus in this paper. The proposed model is general and does not depend on specific deep architectures.

3 The Proposed SGH Method

In this section, we propose a novel weakly-supervised hashing method coupled with binary matrix factorization for social image retrieval, called Semantic Guided Hashing (SGH), which jointly explores the hash learning, the semantic information mining and the data structure discovering.

3.1 Preliminary

Throughout this paper, matrices are denoted by bold uppercase characters and vectors are dentated by bold lowercase characters. Given a matrix $\mathbf{A}$, $\mathbf{a}_i$ denotes its i-th column vector while $\mathbf{a}^j$ is its j-th row vector. The (i, j)-element of $\mathbf{A}$ is denoted by A_{ij}. $\mathbf{A}^T$ denotes the transposed matrix of $\mathbf{A}$ while $\mathrm{Tr}[\mathbf{A}]$ represents the trace of $\mathbf{A}$ if $\mathbf{A}$ is square. For the norm definition, the ℓ_2 norm of a vector is defined as $\|\mathbf{a}\|_2 = \sqrt{\sum_{i=1}^n a_i^2}$ while the Frobenius norm of $\mathbf{A} \in \mathbb{R}^{m\times n}$ is defined as $\|\mathbf{A}\|_F^2 = \sum_{i=1}^m \sum_{j=1}^n A_{ij}^2 = \mathrm{Tr}[\mathbf{A}^T\mathbf{A}]$. The ℓ_1 norm and $\ell_{2,1}$ norm of $\mathbf{A}$ are defined as follows, respectively.

$$\|\mathbf{A}\|_1 = \sum_{i=1}^m \|\mathbf{a}^i\| = \sum_{i=1}^m \sum_{j=1}^n |A_{ij}| \tag{1}$$

$$\|\mathbf{A}\|_{2,1} = \sum_{i=1}^m \|\mathbf{a}^i\|_2 = \sum_{i=1}^m \sqrt{\sum_{j=1}^n A_{ij}^2} \tag{2}$$

Given a binary matrix $\mathbf{F}$, the Binary Matrix Factorization (BMF) model (Zhang et al. 2007) is to decompose it into two latent binary matrices.

$$\mathbf{F} \approx \mathbf{V}\mathbf{U}^T \tag{3}$$

Here $\mathbf{V}$ and $\mathbf{U}$ are two binary matrices. In this work, the BMF model is improved for better hash function learning.

Given a social image set, there are n images $\{\mathbf{x}^i\}_{i=1}^n$ and the associated c user-provided tags $\mathcal{C} = \{t_1, t_2, \ldots, t_c\}$. For each image $\mathbf{x}_i$, the observed image-tag relevance is denoted by a c-dimensional binary-valued vector $\{\mathbf{y}^i\}$. Thus, we have an observed tagging matrix $\mathbf{Y} = [\mathbf{y}^1; \ldots; \mathbf{y}^n] \in \mathcal{R}^{n\times c}$. If $\mathbf{x}^i$ is associated with the j-th tag, $Y_{ij} = 1$ and $Y_{ij} = 0$ otherwise. The j-th column vector $\mathbf{y}_j$ denotes a tagging configuration with respect to the j-th tag. To handle the imperfect tags, an ideal tagging matrix $\mathbf{F} \in \mathbb{R}^{n\times c}$ is introduced. Besides, a symmetric matrix $\mathbf{S}$ denotes the image similarity. And the visual feature of the i-th image is denoted by $g(\mathbf{x}^i) \in \mathbb{R}^d$.

Hashing is to learn hash functions that map each image into the Hamming space to have compact binary representations. Suppose the dimension of the Hamming space is L, and L hash functions ought to be learned to obtain L-bit binary representations $\mathbf{B} \in \{-1, 1\}^{n\times L}$. $\mathbf{b}^i$ is the binary hash codes of the i-th image. A hash function $h(\cdot)$ is learned to generate the binary code $h : \mathbb{R} \mapsto \{-1, 1\}$. The hash functions are defined as

$$\mathbf{H}(\mathbf{x}_i) = [h_1(\mathbf{x}_i), h_2(\mathbf{x}_i), \ldots, h_L(\mathbf{x}_i)] = \mathrm{sgn}(g(\mathbf{x}_i)\mathbf{W}) \tag{4}$$

where $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_1, \ldots, \mathbf{w}_L] \in \mathbb{R}^{d\times L}$ is the transformation matrix and $\mathrm{sgn}(\cdot)$ is the signum function.

3.2 Problem Formulation

To learn good hash codes, the underlying semantic information is uncovered from the community-contributed information while the problem of imperfect user-provided tags is handled (Li et al. 2019). Because Binary Matrix Factorization (BMF) enables to preserve the most important integer property of the binary tagging matrix. We propose to leverage the binary matrix factorization model to analyze the user-provided tags. For better compatible with hashing, the binary constraints are only imposed on the image latent representations.

$$\min_{\mathbf{U},\mathbf{V}} \mathcal{L}(\mathbf{Y}, \mathbf{V}\mathbf{U}^T) + \Omega(\mathbf{U}, \mathbf{V}) \quad \text{s.t. } V_{ij} \in \{-1, 1\} \tag{5}$$

Here $\mathbf{U} \in \mathbb{R}^{c\times r}$ and $\mathbf{V} \in \mathbb{R}^{n\times r}$ are the representations of tags and images in the latent space, respectively. $\mathcal{L}(\cdot, \cdot)$ is the loss function while $\Omega(\cdot)$ denotes the regularization terms. The observed tagging matrix $\mathbf{Y}$ is corrupted by the subjective and noisy tags, which may lead to worse factorization results. To address this problem, we propose to learn an ideal tagging matrix $\mathbf{F}$ from the observed tagging matrix by a sparse model. According to the definition of the tagging matrix, each element F_{ij} is naturally nonnegative and discrete since it denotes the relevance between the i-th image and the j-th tag. Therefore, we have the following problem.

$$\min_{\mathbf{F},\mathbf{U},\mathbf{V}} \mathcal{L}(\mathbf{F}, \mathbf{V}\mathbf{U}^T) + \mathcal{L}_s(\mathbf{F}, \mathbf{Y}) + \Omega(\mathbf{U}, \mathbf{V}) \quad \text{s.t. } F_{ij} \in \{0, 1\}, V_{ij} \in \{-1, 1\} \tag{6}$$

Here $\mathcal{L}_s(\cdot, \cdot)$ is the loss function of the sparse model.

For hashing, the learned hash codes are always required to better preserve the data structures, such as the meaningful neighbor structure. Meanwhile, the hash functions should be learned in company, which can make the proposed hashing method scalable. The binary hash codes of new images can be easily obtained by the hash functions: $\mathbf{b} = \text{sgn}(g(\mathbf{x})\mathbf{W})$. Therefore, the following problem is proposed to formulate the above hashing learning.

$$\min_{\mathbf{B},\mathbf{W},g} \beta \sum_{i=1}^{n} \mathcal{L}(\mathbf{b}^i, g(\mathbf{x}_i)\mathbf{W}) + \frac{\gamma}{2} \sum_{i=1}^{n}\sum_{j=1}^{n} S_{ij} \|\mathbf{b}^i - \mathbf{b}^j\|_2^2 \quad \text{s.t. } \mathbf{B} \in \{-1, 1\}^{n\times L} \tag{7}$$

$$\Leftrightarrow \min_{\mathbf{B},\mathbf{W},g} \beta\mathcal{L}(\mathbf{B}, g(\mathbf{X})\mathbf{W}) + \gamma\text{Tr}[\mathbf{B}^T\mathbf{L}\mathbf{B}] \quad \text{s.t. } \mathbf{B} \in \{-1, 1\}^{n\times L} \tag{8}$$

β and γ are two positive trade-off parameters. $\mathbf{L} \in \mathbb{R}^{n\times n}$ is the Laplacian matrix derived from the symmetric similarity matrix, defined as $\mathbf{L} = \mathbf{E} - \mathbf{S}$, where $\mathbf{E}$ is a diagonal matrix with $E_{ii} = \sum_{j=1}^{n} S_{ij}$. The above formulation can directly learn the discrete hash codes of images and the hash functions simultaneously. The underlying Hamming space is uncovered by learning the hash functions, and the binary hash codes are the latent representations of images in the Hamming space. That is, hashing can be deemed as the latent binary space (i.e., Hamming space) learning.

For the optimal compatibility of matrix factorization and hashing, it is natural to make that the latent spaces uncovered by BMF and hashing share a common space. That is, the learned latent space by BMF is the desired Hamming space. Consequently, the mined semantic information by BMF with the ideal tagging matrix learning can be directly utilized to guide the hashing learning. We have $\mathbf{V} = \mathbf{B}$, $r = L$ and the following unified formulation.

$$\begin{aligned}\min_{\mathbf{F},\mathbf{U},\mathbf{B},\mathbf{W},g} &\mathcal{L}(\mathbf{F}, \mathbf{B}\mathbf{U}^T) + \mathcal{L}_s(\mathbf{F}, \mathbf{Y}) + \beta\mathcal{L}(\mathbf{B}, g(\mathbf{X})\mathbf{W}) \\ &+\frac{\gamma}{2}\text{Tr}[\mathbf{B}^T\mathbf{L}\mathbf{B}] + \Omega(\mathbf{U}, \mathbf{W}) \\ &\text{s.t. } \mathbf{F} \in \{0, 1\}^{n\times c}, \mathbf{B} \in \{-1, 1\}^{n\times L}\end{aligned} \tag{9}$$

The above optimization problem can learn the desired hash functions and the discrete hash codes with the fixed data graph. The data graph is constructed in advance by using the available visual features or tagging information, and kept unchanged in the whole learning process. Due to the imperfect visual features and tags, the data graph may be not good enough for the learning task. Besides, the above optimization problem can be deemed as a two stage method that first constructs the graph and then learns the model since the Laplacian matrix $\mathbf{L}$ can be treated as an operator on the data graph. It leads to that the data graph may be not compatible with the learning model.

To improve the discriminative ability of the hash codes, the data graph is proposed to be jointly learned. Through the joint learning, the data graph can be compatible with the learning model. To avoid trivial solutions, the learned graph $\mathbf{S}$ is required to be consistent with the initial graph $\mathbf{S}_0$. As the data similarity, the sum of similarities between one image and other images is constrained to be one, and $S_{ii} = 0$. Thus, the proposed formulation in Eq. 9 is rewritten as follows.

$$\begin{aligned}\min_{\mathbf{F},\mathbf{U},\mathbf{B},\mathbf{W},\mathbf{S},g} &\mathcal{L}(\mathbf{F}, \mathbf{B}\mathbf{U}^T) + \mathcal{L}_s(\mathbf{F}, \mathbf{Y}) + \alpha\mathcal{L}(\mathbf{S}, \mathbf{S}_0) \\ &+\beta\mathcal{L}(\mathbf{B}, g(\mathbf{X})\mathbf{W}) + \frac{\gamma}{2}\text{Tr}[\mathbf{B}^T\mathbf{L}\mathbf{B}] + \Omega(\mathbf{U}, \mathbf{W}) \\ &\text{s.t. } \mathbf{F} \in \{0, 1\}^{n\times c}, \mathbf{B} \in \{-1, 1\}^{n\times L} \\ &\mathbf{s}^i \geq 0, \sum_{j=1}^{n} S_{ij} = 1, \quad i = 1, \ldots, n\end{aligned} \tag{10}$$

During learning, the Laplacian matrix is updated by $\mathbf{L} = \mathbf{E} - (\mathbf{S}+\mathbf{S}^T)/2$. The above problem incorporates the hashing learning, the semantic information mining and the discriminative graph learning into one unified framework.

4 Optimization

To solve the problem in (10), we should define $\mathcal{L}(\cdot,\cdot)$, $\mathcal{L}_s(\cdot,\cdot)$ and Ω. For simplicity, the least square loss $\mathcal{L}(x,y)=\frac{1}{2}(x-y)^2$ is utilized. To address the noisy tags, $\mathcal{L}_s(x,y)=\mu|x-y|$. For the regularization terms, $\Omega(\mathbf{U},\mathbf{W})=\frac{\lambda}{2}\|\mathbf{U}\|_F^2+\frac{\eta}{2}\|\mathbf{W}\|_{2,1}$. The current focus is to develop better hashing methods. We utilize the given visual features $g(\mathbf{X})$ here and will discuss how to jointly learn features in next section. Thus, the proposed formulation becomes

$$\begin{aligned}\min_{\mathbf{F},\mathbf{U},\mathbf{B},\mathbf{W},\mathbf{S}} \ & \frac{1}{2}\|\mathbf{F}-\mathbf{B}\mathbf{U}^T\|_F^2+\mu\|\mathbf{F}-\mathbf{Y}\|_1+\frac{\alpha}{2}\|\mathbf{S}-\mathbf{S}_0\|_F^2\\ & +\frac{\beta}{2}\|\mathbf{B}-g(\mathbf{X})\mathbf{W}\|_F^2+\frac{\gamma}{2}\mathrm{Tr}[\mathbf{B}^T\mathbf{L}\mathbf{B}]\\ & +\frac{\lambda}{2}\|\mathbf{U}\|_F^2+\frac{\eta}{2}\|\mathbf{W}\|_{2,1}\\ \text{s.t.}\ & \mathbf{F}\in\{0,1\}^{n\times c},\mathbf{B}\in\{-1,1\}^{n\times L}\\ & \mathbf{s}^i\geq 0,\ \sum_{j=1}^{n}S_{ij}=1,\quad i=1,\ldots,n \end{aligned} \tag{11}$$

The above formulation is not convex over all the variables. Therefore, an iterative optimization algorithm is developed.

4.1 Updating F

With **U**, **B**, **W** and **S** fixed, the optimization problem with respect to **F** becomes the following one.

$$\min_{\mathbf{F}\in\{0,1\}^{n\times c}}\frac{1}{2}\|\mathbf{F}-\mathbf{B}\mathbf{U}^T\|_F^2+\mu\|\mathbf{F}-\mathbf{Y}\|_1 \tag{12}$$

The above problem is NP-hard due to the discrete and non-negative constraints. By introducing an auxiliary variable $\tilde{\mathbf{F}}$, we can obtain a closed-form solution for **F**.

$$\tilde{\mathbf{F}}=2\mathbf{F}-1\in\{-1,1\}^{n\times c} \tag{13}$$

Then we have the following equivalent problem.

$$\min_{\tilde{\mathbf{F}}\in\{-1,1\}^{n\times c}}\frac{1}{4}\|\tilde{\mathbf{F}}-\mathbf{F}^*\|_F^2+\frac{\mu}{2}\|\tilde{\mathbf{F}}-\tilde{\mathbf{Y}}\|_1 \tag{14}$$

Here $\mathbf{F}^*=2\mathbf{B}\mathbf{U}^T-1$, and $\tilde{\mathbf{Y}}=2\mathbf{Y}-1\in\{-1,1\}^{n\times c}$. Since $\tilde{F}_{ij}\in\{-1,1\}$ and $\tilde{Y}_{ij}\in\{-1,1\}$, we have $(\tilde{\mathbf{F}}-\tilde{\mathbf{Y}})_{ij}\in\{-2,0,2\}$. Thus we have $\|\tilde{\mathbf{F}}-\tilde{\mathbf{Y}}\|_1=\frac{1}{2}\|\tilde{\mathbf{F}}-\tilde{\mathbf{Y}}\|_F^2$. The problem (14) is equivalent to

Algorithm 1 The Proposed SGH Algorithm

```
Input:
    Tagging matrix $\mathbf{Y}$, Feature matrix $g(\mathbf{X})$;
Output:
    Converged binary codes $\mathbf{B}$ and hash function $\mathbf{H}(\mathbf{x})=\mathrm{sgn}(g(\mathbf{x})\mathbf{W})$.
Compute the image similarity $\mathbf{S}$;
Initialize $\mathbf{F}$, $\mathbf{U}$ and $\mathbf{B}$;
repeat
    Update $\mathbf{F}$ as in Eq. 18;
    Update $\mathbf{U}$ as in Eq. 20;
    Update $\mathbf{W}$ as in Eq. 22;
    Update $\mathbf{S}$ as in Eq. 26;
    Update $\mathbf{B}$ as in Eq. 29;
until Convergence criterion satisfied
```

$$\min_{\tilde{\mathbf{F}}\in\{-1,1\}^{n\times c}}\|\tilde{\mathbf{F}}-\mathbf{F}^*\|_F^2+\mu\|\tilde{\mathbf{F}}-\tilde{\mathbf{Y}}\|_F^2 \tag{15}$$

$$\Leftrightarrow\min_{\tilde{\mathbf{F}}\in\{-1,1\}^{n\times c}}-\mathrm{Tr}[\tilde{\mathbf{F}}^T(\mathbf{F}^*+\mu\tilde{\mathbf{Y}})], \tag{16}$$

which has a closed-form solution.

$$\tilde{\mathbf{F}}=\mathrm{sgn}(\mathbf{F}^*+\mu\tilde{\mathbf{Y}}) \tag{17}$$

$$\mathbf{F}=\frac{\mathrm{sgn}(2\mathbf{B}\mathbf{U}^T+2\mu\mathbf{Y}-\mu-1)+1}{2} \tag{18}$$

4.2 Updating U

For **U**, we have the following subproblem.

$$\min_{\mathbf{U}}\frac{1}{2}\|\mathbf{F}-\mathbf{B}\mathbf{U}^T\|_F^2+\frac{\lambda}{2}\|\mathbf{U}\|_F^2 \tag{19}$$

By calculating the derivative with respect to **U** and setting it to 0, we obtain the following updating rules for **U**.

$$\mathbf{U}=\mathbf{F}^T\mathbf{B}(\mathbf{B}^T\mathbf{B}+\lambda\mathbf{I}_L)^{-1} \tag{20}$$

where $\mathbf{I}_L\in\mathbb{R}^{L\times L}$ is an identity matrix.

4.3 Updating W

With other variables fixed, the subproblem with respect to **W** is as follows.

$$\min_{\mathbf{W}}\frac{\beta}{2}\|\mathbf{B}-g(\mathbf{X})\mathbf{W}\|_F^2+\frac{\eta}{2}\|\mathbf{W}\|_{2,1} \tag{21}$$

By setting the derivative with respect to **U** to 0, the updating rules for **W** are obtained.

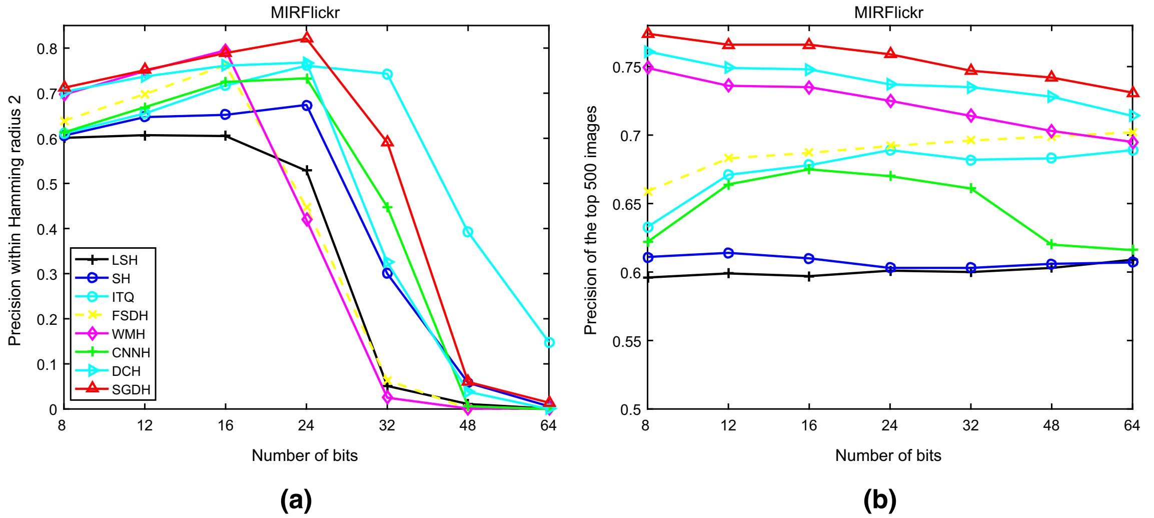
$$\mathbf{W}=(g^T(\mathbf{X})g(\mathbf{X})+\frac{\eta}{\beta}\mathbf{D})^{-1}g^T(\mathbf{X})\mathbf{B} \tag{22}$$

Here **D** is a diagonal matrix with $D_{ii}=\frac{1}{2\|\mathbf{w}^i\|_2}$.

Table 1 MAP of all the compared hashing methods for image retrieval on the MIRFlickr dataset

Method	LSH	SH	ITQ	FSDH	WMH	CNNH	DHN	HashNet	DCH	SGH	SGDH
$L = 8$	0.564	0.569	0.621	0.657	0.671	0.639	0.640	0.632	0.724	0.697	**0.741**
$L = 12$	0.563	0.569	0.627	0.654	0.668	0.640	0.631	0.635	0.729	0.698	**0.740**
$L = 16$	0.562	0.571	0.628	0.655	0.666	0.643	0.631	0.634	0.732	0.704	**0.747**
$L = 24$	0.561	0.576	0.632	0.654	0.670	0.644	0.625	0.636	0.735	0.708	**0.753**
$L = 32$	0.562	0.574	0.633	0.654	0.672	0.635	0.629	0.635	0.742	0.711	**0.765**
$L = 48$	0.557	0.575	0.635	0.653	0.672	0.631	0.635	0.636	0.729	0.710	**0.763**
$L = 64$	0.558	0.571	0.634	0.657	0.669	0.626	0.640	0.637	0.713	0.702	**0.754**

The best results are highlighted in bold



(a)

(b)

Fig. 2 The retrieval performance with different lengths of hash codes on the MIRFlickr dataset. **a** Precision within Hamming radius 2 using hash lookup. **b** Precision within the top 500 returned images using Hamming ranking

Algorithm 2 The Proposed SGDH Algorithm

```
Input:
    Tagging matrix $\mathbf{Y}$ and images $\{\mathbf{x}^i\}_{i=1}^n$;
Output:
    Converged binary codes $\mathbf{B}$, $\mathbf{W}$ and the learned network.
Pre-train the neural network;
Compute the image similarity $\mathbf{S}$;
Initialize $\mathbf{F}$, $\mathbf{U}$ and $\mathbf{B}$;
repeat
    //Forward propagation
    Do forward propagation to obtain $g(\mathbf{X})$;
    //Update other variables
    Update $\mathbf{F}$ as in Eq. 18;
    Update $\mathbf{U}$ as in Eq. 20;
    Update $\mathbf{W}$ as in Eq. 22;
    Update $\mathbf{S}$ as in Eq. 26;
    Update $\mathbf{B}$ as in Eq. 29;
    //Backpropagation
    Update the neural network;
until Convergence criterion satisfied
```

4.4 Updating S

The subproblem with respect to **S** is as follows.

$$\min_{\mathbf{S}} \frac{\alpha}{2}\|\mathbf{S}-\mathbf{S}_0\|_F^2 + \frac{\gamma}{2}\mathrm{Tr}[\mathbf{B}^T\mathbf{L}\mathbf{B}] \tag{23}$$

$$\Leftrightarrow \min_{\mathbf{S}} \sum_{i=1}^{n} \|\mathbf{s}^i - \mathbf{s}_0^i\|_2^2 + \frac{\gamma}{2\alpha}\sum_{i=1}^{n}\sum_{j=1}^{n} S_{ij}\|\mathbf{b}^i - \mathbf{b}^j\|_2^2$$

$$\text{s.t.}\ \mathbf{s}^i \geq 0,\ \sum_{j=1}^{n} S_{ij} = 1, \quad i = 1, \ldots, n \tag{24}$$

By simplify inference, the above problem is equivalent to the following one.

$$\min_{\mathbf{S}} \sum_{i=1}^{n} \|\mathbf{s}^i - \mathbf{s}_0^i\|_2^2 + \frac{\gamma}{2\alpha}\sum_{i=1}^{n} \mathbf{s}^i(\mathbf{p}^i)^T$$

$$\text{s.t.}\ \mathbf{s}^i \geq 0,\ \sum_{j=1}^{n} S_{ij} = 1, \quad i = 1, \ldots, n \tag{25}$$

Here $\mathbf{p}^i = [\|\mathbf{b}^i - \mathbf{b}^1\|_2^2, \|\mathbf{b}^i - \mathbf{b}^2\|_2^2, \ldots, \|\mathbf{b}^i - \mathbf{b}^n\|_2^2]$. The above problem is divided into n independent subproblems.

$$\min_{\mathbf{s}^i} \|\mathbf{s}^i - \mathbf{s}_0^i + \frac{\gamma}{4\alpha}\mathbf{p}^i\|_2^2$$
$$\text{s.t.}\, \mathbf{s}^i \geq 0,\ \sum_{j=1}^{n} S_{ij} = 1, \quad i = 1, \ldots, n \tag{26}$$

The above problem can be well solved by using the existing optimization strategies since it is a classic constrained quadratic programming problem.

4.5 Updating B

With **F**, **U**, **W** and **S** fixed, the optimization problem for **B** is as follows.

$$\min_{\mathbf{B}} \frac{1}{2}\|\mathbf{F} - \mathbf{B}\mathbf{U}^T\|_F^2 + \frac{\beta}{2}\|\mathbf{B} - g(\mathbf{X})\mathbf{W}\|_F^2 + \frac{\gamma}{2}\text{Tr}[\mathbf{B}^T\mathbf{L}\mathbf{B}]$$
$$\text{s.t.}\, \mathbf{B} \in \{-1, 1\}^{n\times L} \tag{27}$$

Since $\text{Tr}[\mathbf{B}\mathbf{B}^T]$ is a constant, we have the following problem.

$$\min_{\mathbf{B}} \frac{1}{2}\text{Tr}[\mathbf{B}\mathbf{U}^T\mathbf{U}\mathbf{B}^T] - \text{Tr}[(\mathbf{F}\mathbf{U} + \beta g(\mathbf{X})\mathbf{W})\mathbf{B}^T] + \frac{\gamma}{2}\text{Tr}[\mathbf{B}^T\mathbf{L}\mathbf{B}]$$
$$\text{s.t.}\, \mathbf{B} \in \{-1, 1\}^{n\times L} \tag{28}$$

By setting the derivative with respect to **B** to 0, we obtain the following sylvester function with the discrete constraints.

$$\gamma\mathbf{L}\mathbf{B} + \mathbf{B}\mathbf{U}^T\mathbf{U} = \mathbf{F}\mathbf{U} + \beta g(\mathbf{X})\mathbf{W} \tag{29}$$

The above Sylvester function can be solved to obtain $\mathbf{B}^*$ by using the existing methods, such as the Sylvester function in MATLAB. And then $\mathbf{B}^* = \text{sgn}(\mathbf{B}^*)$.

From the above analysis, it is impossible to directly obtain all the variables. As a consequence, an iterative algorithm is developed to update them, which is summarized in **Algorithm 1**. The convergence criterion used is that $|\mathcal{O}_{t-1} - \mathcal{O}_t|/\mathcal{O}_{t-1} < 10^{-6}$, where $\mathcal{O}_t$ is the value of the objective function in the t-th iteration.

5 Extension

This section will discuss how to extend the proposed method from shallow learning to deep learning. The above method learns hash functions with the given features $g(\mathbf{x})$. To integrate the feature learning and hashing learning into one unified framework, a Semantic Guided Deep Hashing (SGDH) method is proposed. That is, $g(\mathbf{x})$ is the output of the topmost layer. **W** can be deemed as a fully connected layer connected to the output layer. Several deep networks, such as Convolutional Neural Networks (CNNs) (Krizhevsky et al. 2012), can be used for extracting features from raw pixels.

The deep network is first pre-trained by using the ImageNet dataset with more than 1.2 million images. Then, **F**, **U**, **B**, **W** and **S** are learned by using the updating rules in Sect. 4. And the deep network is tuned by using the back-propagation technique. The extended deep model is summarized in **Algorithm 2**. The optimization is done by following the implementation details in the work (Jiang and Li 2018). Once the network and **W** are obtained, given a new image x, its binary hash codes are obtained: $\mathbf{b} = g(\mathbf{x})\mathbf{W}$.

6 Experiments

Experiments are conducted to evaluate the performance of the proposed hashing method for image retrieval on two widely-used social image datasets. Several presentative hashing methods are compared.

6.1 Dataset

There are massive images associated with user-provided tags available from social websites, which can be conveniently collected for experimental evaluation. In this work, experiments are conducted on two publicly-available social image datasets: MIRFlickr (Huiskes and Lew 2008) and NUS-WIDE (Tang et al. 2016).

MIRFlickr (Huiskes and Lew 2008). There are 25, 000 images associated with 1386 user-provided tags in the MIRFlickr dataset. The ground-truth annotation of 38 concepts are also provided for evaluation. Tags appearing less than 50 times are removed, which results in 457 unique tags and 18 concepts.

NUS-WIDE (Tang et al. 2016). This dataset contains 269, 648 images associated with 5018 user-provided tags. To evaluate the performance, it also provides the ground-truth annotations of 81 concepts. Those too noisy tags appearing less than 125 times are removed, resulting in 3137 unique tags.

In experiments, the performance is evaluated based on whether the query image and the returned images have at least one common semantic concept. Hence, each image ought to be related with at least one tag and one concept. Consequently, we have 18, 379 images for the MIRFlickr dataset and 207, 697 images for the NUS-WIDE dataset. To learn hash functions, the datasets are divided into two subsets: the query subset and the database subset. The query subset is constructed by randomly selecting 1000 images to evaluate the performance. The rest images are used as the database subset. To learn the hashing model, 5000 and 20, 000 images are randomly selected from the databases for the MIRFlickr and NUS-WIDE datasets, respectively.

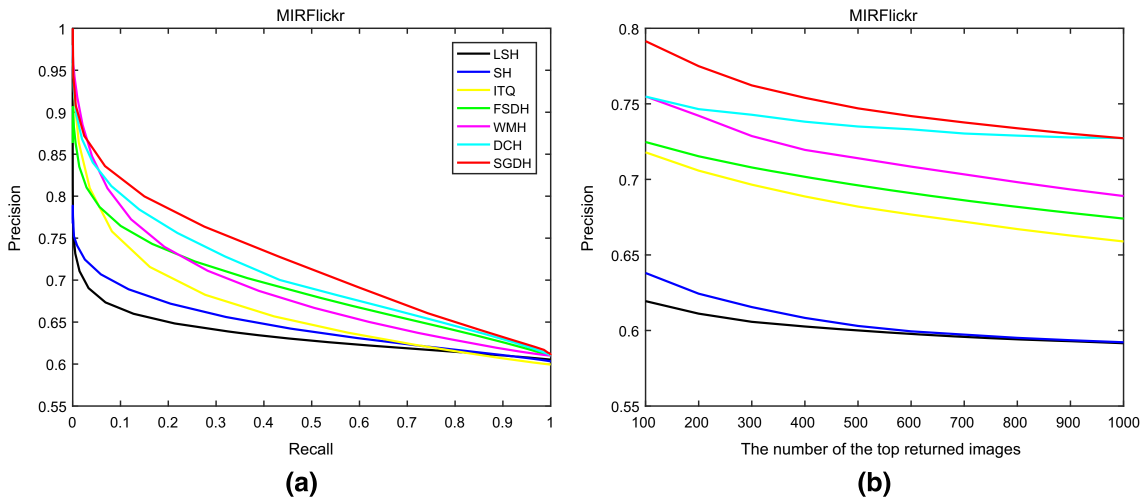


Fig. 3 The retrieval curves on the MIRFlickr dataset. **a** Precision-Recall curves with 32 hash bits **b** Precision curves of the top K returned images with 32 hash bits

6.2 Evaluation Metric

To evaluate the retrieval performance, several metrics including Precision, Recall and Mean Average Precision (MAP) are adopted. Two widely-used criteria, i.e., Hamming ranking and hash lookup, are utilized. For Hamming ranking, the Hamming distances between the query image and the images in the database are calculated and the desired neighbors are returned from the top images of the ranked list according to the Hamming distance. For hash lookup, a lookup table consists of images that are within a small radius r of the query image according to the Hamming distance. For the Hamming ranking-based evaluation, the Precision and Recall among the top M returned images are computed. For the hash lookup-based evaluation, the Hamming radius r is set to 2.

Besides, MAP is utilized to evaluate the performance of information retrieval. It is the mean of the Average Precision (AP) scores over all query images. As one standard measure used for ranked sets, AP is the average of the precision scores at each position that a relevant image appears.

$$AP = \frac{\sum_j P(j)rel(j)}{M} \tag{30}$$

$P(j)$ is the precision score of the top j results while $rel(j)$ is an indicator function equaling 1 if the j-th result is relevant, and 0 otherwise. M is the number of relevant images.

6.3 Compared Methods

To verify the effectiveness of the proposed methods, it is compared with several other popular hashing methods including shallow methods with CNN features and deep methods. The compared methods are listed as follows.

- LSH (Indyk and Motwani 1998): It generates hash functions by using the random projections.
- SH (Weiss et al. 2008): It learns hash functions by preserving neighbors in the Hamming space.
- ITQ (Gong et al. 2013): It learns hash functions by maximizing the data variance of each hash bit via PCA projection.
- FSDH (Gui et al. 2018): It learns hash functions by regressing each label to its corresponding hash codes.
- WMH (Tang and Li 2018): It learns hash functions by exploring the visual structures and the textual structures of social images.
- CNNH (Xia et al. 2014): It learns hash functions by using a two-stage strategy: hash code learning and hashing model learning.
- DHN (Zhu et al. 2016): It proposes to learn the representations tailored to hash code by controlling the quantization error.
- HashNet (Cao et al. 2017): It learns hash codes by continuation with convergence guarantees.
- DCH (Cao et al. 2018): It learns hash functions by using a pairwise cross-entropy loss based on the Cauchy distribution.
- SGH: the proposed hashing method by exploring the underlying semantic information and data structure.
- SGDH: the proposed deep hashing method by exploring the feature learning, underlying semantic information and data structure for hashing learning.

For the shallow hashing method, the used visual features are the 4096-D output of the pre-trained AlexNet (Krizhevsky et al. 2012) that is trained on the ImageNet dataset for fair comparison. For CNN-based methods, the pre-trained AlexNet is used as the base net. For the supervised deep meth-

Table 2 MAP of all the compared hashing methods for image retrieval on the NUS-WIDE dataset. The best results are highlighted in bold

Method	LSH	SH	ITQ	FSDH	WMH	CNNH	DHN	HashNet	DCH	SGH	SGDH
$L = 8$	0.430	0.445	0.457	0.461	0.462	0.449	0.456	0.422	0.473	0.468	**0.483**
$L = 12$	0.432	0.441	0.456	0.461	0.464	0.450	0.434	0.425	0.474	0.470	**0.485**
$L = 16$	0.432	0.444	0.458	0.460	0.466	0.452	0.428	0.426	0.474	0.471	**0.489**
$L = 24$	0.433	0.445	0.457	0.462	0.473	0.451	0.434	0.428	0.479	0.475	**0.493**
$L = 32$	0.432	0.446	0.458	0.469	0.477	0.453	0.442	0.432	0.484	0.479	**0.496**
$L = 48$	0.431	0.442	0.458	0.467	0.471	0.451	0.451	0.444	0.482	0.480	**0.497**
$L = 64$	0.432	0.439	0.457	0.465	0.474	0.448	0.465	0.443	0.478	0.484	**0.499**

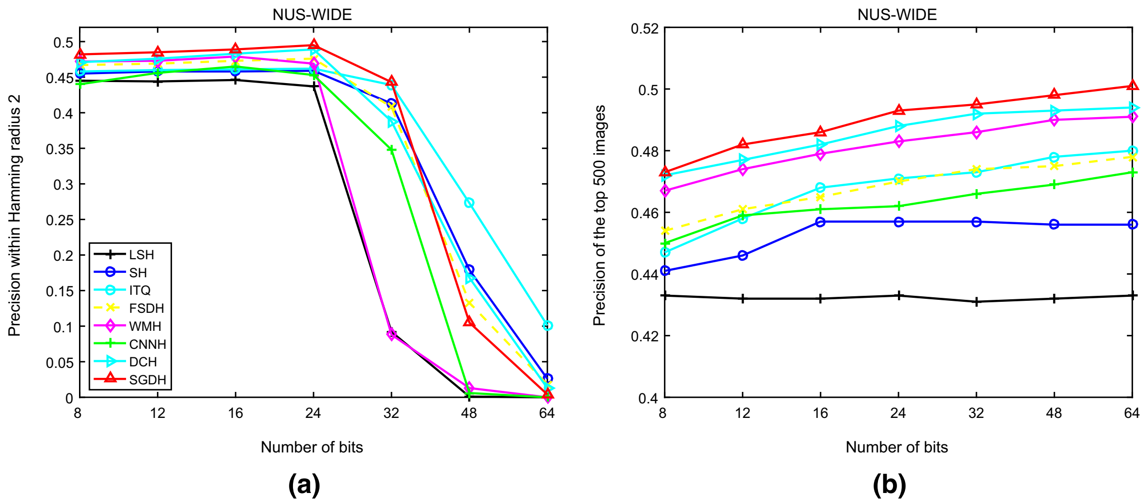


Fig. 4 The retrieval performance with different lengths of hash codes on the NUS-WIDE dataset. **a** Precision within Hamming radius 2 using hash lookup. **b** Precision within the top 500 returned images using Hamming ranking

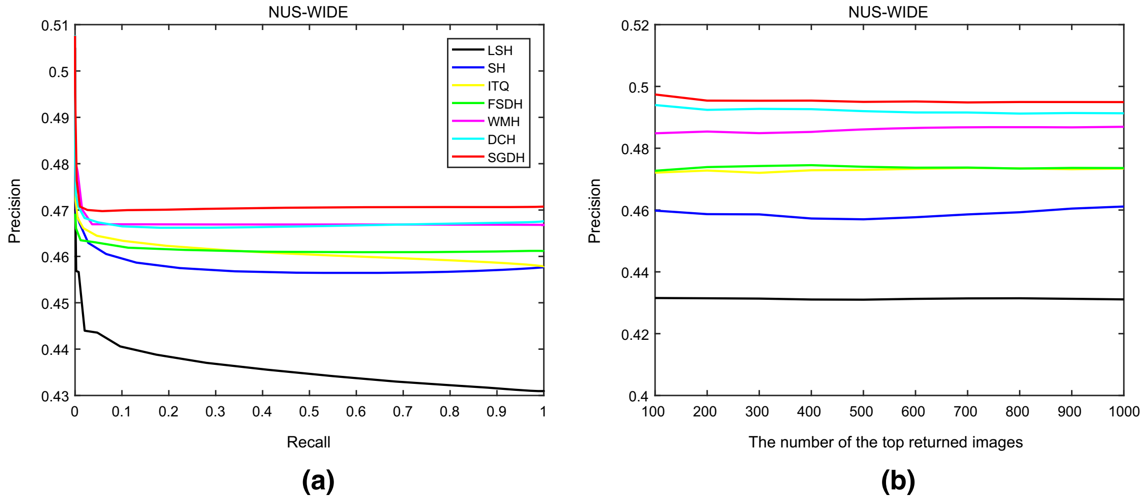


Fig. 5 The retrieval curves on the NUS-WIDE dataset. **a** Precision-Recall curves with 32 hash bits **b** Precision curves of the top K returned images with 32 hash bits

ods such as DHN (Zhu et al. 2016) and HashNet (Cao et al. 2017), they explore the supervised concept information to learn hash codes. However, there is only tagging information available for weakly supervised hashing. For comparison, the weakly-supervised tagging information is used to learn hash codes. There are some parameters to be set. The same tuning strategy is used for all the hashing method. In experiments, we set λ=0.005 and $\eta = 1$ empirically. Besides, the default values of μ, α, β and γ are set to 10, 1, 10 and 0.01, respectively. And we will discuss the sensitiveness analysis of some parameters.

6.4 Results on the MIRFlickr Dataset

This section carries out experiments on the MIRFlickr dataset to demonstrate the effectiveness of the proposed hashing method for social image retrieval. For quantitative evaluation, the length of binary hash codes is varied within {8, 12, 16, 24, 32, 48, 64}.

First, the retrieval results in terms of MAP of different hashing methods are shown in Table 1. From the compared results, it can be observed that the proposed SGH method outperforms other shallow hashing methods based on CNN features and the proposed SGDH method achieves the best performance for social image retrieval, which enable to well verify the effectiveness of the motivation of the proposed method. Besides, there are some other observations. MAP of LSH changes slightly by varying the length of hash bits since LSH randomly generates hash functions without considering the data distribution and MAP is calculated by using all the samples in the database. By considering the data distribution, the data-dependent methods are superior to LSH, which is consistent with many previous works (Tang and Li 2018). Second, the end-to-end hashing methods achieve better search results than the shallow hashing methods and the two-stage deep hashing method. Because the end-to-end learning can well leverage the powerful ability of feature learning. Third, by exploring the weakly-supervised tags that can reveal the semantic information of images, WMH and SGH achieve better results than ITQ for image retrieval. It indicates that the underlying semantic information is beneficial to learn hashing functions. Forth, the proposed method is better than WMH by mining the graph structure and uncovering the underlying semantic information. Fifth, DHN and HashNet achieve somewhat worse results. Because they are proposed to explore the supervised information and they can not well explore the weakly-supervised information. Sixth, the retrieval performance of the proposed SGDH method is better than DCH due to the graph structure mining. The learned graph structure can adaptively suitable for hashing. Finally, the proposed hashing method outperforms other hashing methods by simultaneously and seamlessly incorporating the feature learning, hash code learning, semantic information uncovering and data structure learning.

Experiments are also conducted to evaluate the performance of the hashing approaches using the Hamming ranking and hash lookup measures. The corresponding performance curves with different hash bits are presented in Fig. 2. The retrieval performance is shown by the precision curves of Hamming ranking (within top 500 results) and hash lookup (within Hamming radius 2). The results of DHN and HashNet are not shown due to that they can not well handle the noisy tagging information and it may be somewhat unfair to compare with them by using the user-provided tags. From the results, it can be seen that the proposed hashing method gains the best retrieval performance, which is consistent with the aforementioned observations. Besides, the proposed SGDH method achieves the best performance within Hamming radius 2 with 24 binary bits.

As a complementary evaluation, the precision-recall curves and the precision curves with 32 hash bits are also reported in Fig. 3. The curves of SGH and CNNH are not presented since SGDH and DCH are better than SGH and CNNH, respectively. The results demonstrate that the proposed SGDH method gains the performance improvement for social image retrieval.

The above results show that the proposed hashing method enables to learn better hash codes by jointly exploring the underlying semantic information, feature learning, data structure and hash code learning.

6.5 Results on the NUS-WIDE Dataset

This section conducts experiments on the NUS-WIDE dataset to show the effectiveness of the proposed hashing method. The length of hash codes is also set within {8, 12, 16, 24, 32, 48, 64}. The comparison scheme and setting are the same to ones on the MIRFlickr dataset. The quantitative results in terms of MAP are presented in Table 2, while Fig. 4 shows the precision curves of different hashing methods using Hamming ranking and hash lookup table. From the results, it can be easily observed that the proposed hashing method achieves the best retrieval performance on the scalable NUS-WIDE dataset. Besides, we can obtain the same conclusions from the results on the NUS-WIDE dataset to the ones on the MIRFlickr dataset, which can well validate the effectiveness of our motivation. Therefore, the proposed hashing method can learn better hash functions by uncovering the semantic information and data structure from social images.

The performance is also evaluated by using the precision-recall curves and the precision curves on the NUS-WIDE dataset. The corresponding results of the hashing methods with 32 hash bits are presented in Fig. 5. The proposed hashing method also gains the best retrieval performance. Compared with other hashing methods, the better results achieved by the proposed hashing method demonstrate that it is helpful to learn hash functions by jointly incorporating the hash code learning, the semantic information mining and the data structure discovering.

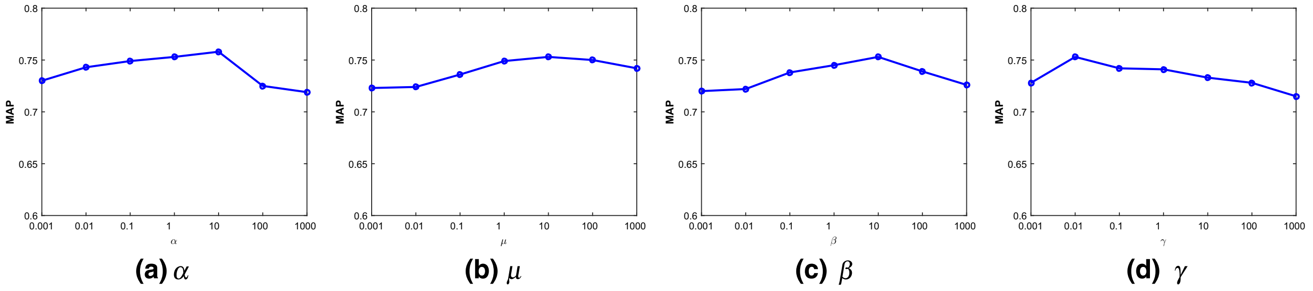


Fig. 6 The parameter sensitiveness of α, μ, β and γ in terms of MAP on the MIRFlickr dataset

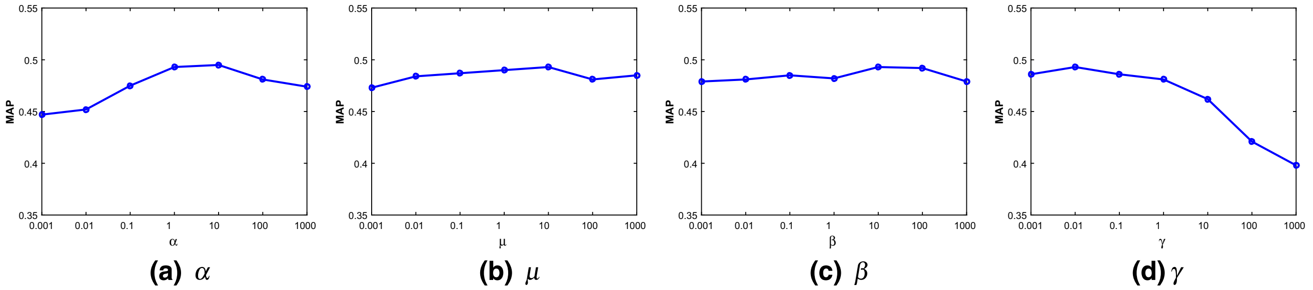


Fig. 7 The parameter sensitiveness of α, μ, β and γ in terms of MAP on the NUS-WIDE dataset

6.6 Sensitiveness Analysis

For the proposed method, there are several hyper-parameters to tradeoff each component in the proposed joint optimization formulation. This section conducts experiments to study the sensitivity analysis of the hyper-parameters. From the results, it is observed that the proposed method is robust to the parameters λ and η. Therefore, we set $\lambda = 0.005$ and $\eta = 1$ empirically as the default values. Now we study the sensitivity analysis of α, μ, β and γ.

The results by tuning α, μ, β and γ on the MIRFlickr and NUS-WIDE datasets are presented in Figs. 6 and 7, respectively. The length of hash bits is set to 24. From the results, it is observed that the proposed method has good results when α is within [1, 10]. In experiments, we set α to 1 by treating the tagging information and graph structure mining equally. Besides, the best results are achieved with $\mu = 10$, $\beta = 10$ and $\gamma = 0.01$ when α is set to 1. The performance becomes worse if we set them to be larger or smaller values. It can demonstrate that it is useful to explore the underlying semantic information and adaptively learn the data structure for hashing learning. Furthermore, the importance if the weakly supervised semantic information is controlled by μ. If μ is set to 0, that is, the weakly supervised semantic information is ignored, the results in terms of MAP with 24 bits on the MIRFlickr and NUS-WIDE datasets are 0.695 and 0.460, respectively, which are significantly worse than the performance by considering the user-provided tags. It can well verify the necessity of introducing the weakly supervised semantic information.

7 Conclusion

In this paper, we propose a new hashing method by uncovering the underlying semantic information from the weakly-supervised user-provided tags to guide the discrete hash code learning for social image retrieval. The semantic information is minded by introducing the binary matrix factorization model with the ideal tagging matrix learning. Besides, the data structure is also discovered by adaptively learning a discriminative data graph to well preserve the meaningful neighbors. The proposed method is extended to one deep approach by jointly exploring the feature learning, the hash code learning, the semantic information uncovering and the data structure mining. Experiments on two widely-used social image datasets demonstrate the effectiveness of the proposed method for social image retrieval.

Acknowledgements This work was partially supported by the National Key R&D Program of China under Grant 2018AAA0102002, the National Natural Science Foundation of China (Grant No. 61772275, 61732007, 61772268 and 61672285) and the Natural Science Foundation of Jiangsu Province (Grant BK20170033).

References

Cao, Yue, Long, Mingsheng, Liu, Bin, & Wang, Jianmin (2018). Deep cauchy hashing for hamming space retrieval. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, (pp. 1229–1237).

Cao, Zhangjie, Long, Mingsheng, Wang, Jianmin, & Yu, Philip S. (2017). Hashnet: Deep learning to hash by continuation. In *Pro-*

ceedings of IEEE International Conference on Computer Vision, (pp. 5609–5618).

Dizaji, Kamran Ghasedi, Zheng, Feng, Sadoughi, Najmeh, Yang, Yanhua, Deng, Cheng, & Huang, Heng (2018). Unsupervised deep generative adversarial hashing network. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, (pp. 3664–3673).

Gong, Yunchao, Ke, Qifa, Isard, Michael, & Lazebnik, Svetlana. (2013). A multi-view embedding space for modeling internet images, tags, and their semantics. *International Journal of Computer Vision*, *106*(2), 210–233.

Gong, Y., Lazebnik, S., Gordo, A., & Perronnin, F. (2013). Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(12), 2916–2929.

Gordo, Albert, Almazán, Jon, Revaud, Jérôme, & Larlus, Diane. (2017). End-to-end learning of deep visual representations for image retrieval. *International Journal of Computer Vision*, *124*(2), 237–254.

Guan, Ziyu, Xie, Fei, Zhao, Wanqing, Wang, Xiaopeng, Chen, Long, Zhao, Wei, & Peng, Jinye (2018). Tag-based weakly-supervised hashing for image retrieval. In *Proceedings of International Joint Conference on Artificial Intelligence*, (pp. 3776–3782).

Gui, Jie, & Li, Ping (2018). R2SDH: robust rotated supervised discrete hashing. In *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, (pp. 1485–1493).

Gui, Jie, Liu, Tongliang, Sun, Zhenan, Tao, Dacheng, & Tan, Tieniu. (2018). Fast supervised discrete hashing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *40*(2), 490–496.

Heo, Jae-Pil, Lee, Youngwoon, He, Junfeng, Chang, Shih-Fu, & Yoon, Sung eui (2012). Spherical hashing. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, (pp. 2957–2964).

Heo, Jae-Pil, Lee, Youngwoon, He, Junfeng, Chang, Shih-Fu, & Yoon, Sung-Eui. (2015). Spherical hashing: Binary code embedding with hyperspheres. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *37*(11), 2304–2316.

Hu, Haifeng, Wang, Kun, Lv, Chenggang, Wu, Jiansheng, & Yang, Zhen. (2019). Semi-supervised metric learning-based anchor graph hashing for large-scale image retrieval. *IEEE Transactions on Image Processing*, *28*(2), 739–754.

Huiskes, Mark J., & Lew, Michael S. (2008). The mir flickr retrieval evaluation. In *Proceedings of ACM International Conference on Multimedia Information Retrieval*, (pp. 39–43).

Indyk, Piotr, & Motwani, Rajeev (1998). Approximate nearest neighbors: Towards removing the curse of dimensionality. In *Proceedings of ACM Symposium on Theory of Computing*, (pp. 604–613).

Ji, Jianqiu, Li, Jianmin, Yan, Shuicheng, Zhang, Bo, & Tian, Qi (2012). Super-bit locality-sensitive hashing. In *Proceedings of Advances in Neural Information Processing Systems*, (pp. 108–116).

Jiang, Qing-Yuan, & Li, Wu-Jun (2018). Asymmetric deep supervised hashing. In *Proceedings of AAAI Conference on Artificial Intelligence*, (pp. 3342–3349).

Jiang, Qing-Yuan, Cui, Xue, & Li, Wu-Jun. (2018). Deep discrete supervised hashing. *IEEE Transactions on Image Processing*, *27*(12), 5996–6009.

Jin, Lu, Li, Kai, Li, Zechao, Xiao, Fu, Qi, Guo-Jun, & Tang, Jinhui. (2019). Deep semantic-preserving ordinal hashing for cross-modal similarity search. *IEEE Transactions on Neural Networks and Learning Systems*, *30*(5), 1429–1440.

Jin, Lu, Shu, Xiangbo, Li, Kai, Li, Zechao, Qi, Guo-Jun, & Tang, Jinhui. (2019). Deep ordinal hashing with spatial attention. *IEEE Transactions on Image Processing*, *28*(5), 2173–2186.

Kan, Meina, Xu, Dong, Shan, Shiguang, & Chen, Xilin. (2014). Semisupervised hashing via kernel hyperplane learning for scalable image search. *IEEE Transactions on Circuits and Systems for Video Technology*, *24*(4), 704–713.

Krizhevsky, Alex, Sutskever, Ilya, & Hinton, Geoffrey E. (2012). Imagenet classification with deep convolutional neural networks. In *Proceedings of Advances in Neural Information Processing Systems*, (pages 1106–1114).

Kulis, Brian, & Darrell, Trevor (2009). Learning to hash with binary reconstructive embeddings. In *Proceedings of Advances in Neural Information Processing Systems*, (pp. 1042–1050).

Kulis, Brian, & Grauman, Kristen (2009). Kernelized locality-sensitive hashing for scalable image search. In *Proceedings of IEEE International Conference on Computer Vision*, (pp. 2130–2137).

Li, Qi, Sun, Zhenan, He, Ran, & Tan, Tieniu (2017). Deep supervised discrete hashing. In *Proceedings of Advances in Neural Information Processing Systems*, (pp. 2479–2488).

Li, Wu-Jun, Wang, Sheng, & Kang, Wang-Cheng (2016). Feature learning based deep supervised hashing with pairwise labels. In *Proceedings of International Joint Conference on Artificial Intelligence*, (pp. 1711–1717).

Lin, Kevin, Yang, Huei-Fang, Hsiao, Jen-Hao, & Chen, Chu-Song (2015). Deep learning of binary hash codes for fast image retrieval. In *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, (pp. 4933–4941).

Liong, Venice Erin, Lu, Jiwen, Tan, Yap-Peng, & Zhou, Jie (2017). Cross-modal deep variational hashing. In *Proceedings of IEEE International Conference on Computer Vision*, (pp. 4097–4105).

Liong, Venice Erin, Lu, Jiwen, Wang, Gang, Moulin, Pierre, & Zhou, Jie (2015). Deep hashing for compact binary codes learning. In *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, (pp. 2475–2483).

Li, Zechao, & Tang, Jinhui. (2015). Weakly supervised deep metric learning for community-contributed image retrieval. *IEEE Transactions on Multimedia*, *17*(11), 1989–1999.

Li, Zechao, & Tang, Jinhui. (2017). Weakly supervised deep matrix factorization for social image understanding. *IEEE Transactions on Image Processing*, *26*(1), 276–288.

Li, Zechao, Tang, Jinhui, & Mei, Tao. (2019). Deep collaborative embedding for social image understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *41*(9), 2070–2083.

Liu, Haomiao, Wang, Ruiping, Shan, Shiguang, Chen, Xilin (2016). Deep supervised hashing for fast image retrieval. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, (pp. 2064–2072).

Liu, Wei, Wang, Jun, Kumar, Sanjiv, & Chang, Shih-Fu (2011). Hashing with graphs. In *Proceedings of International Conference on Machine Learning*, (pp. 1–8).

Liu, Xianglong, Deng, Cheng, Lang, Bo, Tao, Dacheng, & Li, Xuelong. (2016). Query-adaptive reciprocal hash tables for nearest neighbor search. *IEEE Transactions on Image Processing*, *25*(2), 907–919.

Liu, Xianglong, He, Junfeng, & Chang, Shih-Fu. (2017). Hash bit selection for nearest neighbor search. *IEEE Transactions on Image Processing*, *26*(11), 5367–5380.

Liu, Xianglong, Li, Zhujin, Deng, Cheng, & Tao, Dacheng. (2017). Distributed adaptive binary quantization for fast nearest neighbor search. *IEEE Transactions on Image Processing*, *26*(11), 5324–5336.

Mandal, Devraj, Chaudhury, Kunal N., & Biswas, Soma. (2019). Generalized semantic preserving hashing for cross-modal retrieval. *IEEE Transactions on Image Processing*, *28*(1), 102–112.

Raginsky, Maxim, & Lazebnik, Svetlana (2009). Locality-sensitive binary codes from shift-invariant kernels. In *Proceedings of Advances in Neural Information Processing Systems*, (pp. 1509–1517).

Shen, Fumin, Shen, Chunhua, Liu, Wei, & Shen, Heng Tao (2015). Supervised discrete hashing. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, (pp. 37–45).

Shen, Fumin, Shen, Chunhua, Shi, Qinfeng, Anton, van den Hengel, & Tang, Zhenmin (2013). Inductive hashing on manifolds. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, (pp. 1562–1569).

Sohn, Sungryull, Kim, Hyunwoo, & Kim, Junmo. (2017). Uncorrelated component analysis-based hashing. *IEEE Transactions on Image Processing*, *26*(8), 3759–3774.

Tang, Jinhui, & Li, Zechao. (2018). Weakly supervised multimodal hashing for scalable social image retrieval. *IEEE Transactions on on Circuits and Systems for Video Technology*, *28*(10), 2730–2741.

Tang, Jinhui, Lin, Jie, Li, Zechao, & Yang, Jian. (2018). Discriminative deep quantization hashing for face image retrieval. *IEEE Transactions on Neural Networks and Learning Systtems*, *29*(12), 6154–6162.

Tang, Jinhui, Li, Zechao, Wang, Meng, & Zhao, Ruizhen. (2015). Neighborhood discriminant hashing for large-scale image retrieval. *IEEE Transactions on Image Processing*, *24*(9), 2827–2840.

Tang, Jinhui, Li, Zechao, & Zhu, Xiang. (2018). Supervised deep hashing for scalable face image retrieval. *Pattern Recognition*, *75*, 25–32.

Tang, Jinhui, Shu, Xiangbo, Li, Zechao, Jiang, Yu-Gang, & Tian, Qi. (2019). Social anchor-unit graph regularized tensor completion for large-scale image retagging. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *41*(8), 2027–2034.

Tang, Jinhui, Shu, Xiaongbo, Li, Zechao, Qi, Guo-Jun, & Wang, Jingdong. (2016). Generalized deep transfer networks for knowledge propagation in heterogeneous domains. *ACM Transactions on Multimedia Computing Communications and Applications*, *12*(4s), 1–22.

Tang, Jinhui, Shu, Xiangbo, Qi, Guo-Jun, Li, Zechao, Wang, Meng, Yan, Shuicheng, et al. (2017). Tri-clustered tensor completion for social-aware image tag refinement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *39*(8), 1662–1674.

Venkateswara, Hemanth, Eusebio, Jose, Chakraborty, Shayok, & Panchanathan, Sethuraman (2017). Deep hashing network for unsupervised domain adaptation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, (pp. 5385–5394).

Wang, Daixin, Cui, Peng, Ou, Mingdong, & Zhu, Wenwu (2015). Deep multimodal hashing with orthogonal units. In *Proceedings of International Joint Conference on Artificial Intelligence*, (pp. 2291–2297).

Wang, Jun, Kumar, Sanjiv, & Chang, Shih-Fu. (2012). Semi-supervised hashing for large scale search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *34*(12), 2393–2406.

Wang, Jingdong, Zhang, Ting, Song, Jingkuan, Sebe, Nicu, & Shen, Heng Tao. (2018). A survey on learning to hash. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *40*(4), 769–790.

Weiss, Yair, Torralba, Antonio, & Fergus, Robert (2008). Spectral hashing. In *Proceedings of Advances in Neural Information Processing Systems*, (pp. 1753–1760).

Wu, Fei, Yu, Zhou, Yi, Tang, Siliang, Zhang, & Yin, & Zhuang, Yueting., (2014). Sparse multi-modal hashing. *IEEE Transactions on Multimedia*, *16*(2), 424–439.

Xia, Rongkai, Pan, Yan, Lai, Hanjiang, Liu, Cong, & Yan, Shuicheng (2014). Supervised hashing for image retrieval via image representation learning. In *Proceedings of AAAI Conference on Artificial Intelligence*, (pp. 2156–2162).

Zhai, Deming, Liu, Xianming, Ji, Xiangyang, Zhao, Debin, Satoh, Shin'ichi, & Gao, Wen. (2018). Supervised distributed hashing for large-scale multimedia retrieval. *IEEE Transactions on Multimedia*, *20*(3), 675–686.

Zhang, Dell, Wang, Jun, Cai, Deng, & Lu, Jinsong (2012). Self-taught hashing for fast similarity search. In *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, (pp. 18–25).

Zhang, Dongqing, & Li, Wu-Jun (2014). Large-scale supervised multimodal hashing with semantic correlation maximization. In *Proceedings of AAAI Conference on Artificial Intelligence*, (pp. 143–152).

Zhang, Zhongyuan, Li, Tao, Ding, Chris H. Q., & Zhang, Xiang-Sun (2007). Binary matrix factorization with applications. In *Proceedings of IEEE International Conference on Data Mining*, (pp. 391–400).

Zhang, Ziming, Chen, Yuting, & Saligrama, Venkatesh (2016). Efficient training of very deep neural networks for supervised hashing. In *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, (pp. 1487–1495).

Zhang, Haofeng, Liu, Li, Long, Yang, & Shao, Ling. (2018). Unsupervised deep hashing with pseudo labels for scalable image retrieval. *IEEE Transactions on Image Processing*, *27*(4), 1626–1638.

Zhu, Han, Long, Mingsheng, Wang, Jianmin, & Cao, Yue (2016). Deep hashing network for efficient similarity retrieval. In *Proceedings of AAAI Conference on Artificial Intelligence*, (pp. 2415–2421).

Zhu, Xiaofeng, Huang, Zi, Shen, Heng Tao, & Zhao, Xin (2013). Linear cross-modal hashing for efficient multimedia search. In *Proceedings of ACM International Conference on Multimedia*, (pp. 143–152).

Zhuang, Yueting, Liu, Yang, Wu, Fei, Zhang, Yin, & Shao, Jian (2011). Hypergraph spectral hashing for similarity search of social image. In *Proceedings of ACM International Conference on Multimedia*, (pp. 1457–1460).

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.