applied sciences

MDPI

Article

Discrete Semantics-Guided Asymmetric Hashing for Large-Scale Multimedia Retrieval

Jun Long [1,2,3], Longzhi Sun [1], Liujie Hua [1,2] and Zhan Yang [2,3,*]

[1] School of Computer Science and Engineering, Central South University, Changsha 410083, China; junlong@csu.edu.cn (J.L.); sunlongzhi@csu.edu.cn (L.S.); liujiehua@csu.edu.cn (L.H.)
[2] Network Resources Management and Trust Evaluation Key Laboratory of Hunan Province, Central South University, Changsha 410083, China
[3] Big Data Institute, Central South University, Changsha 410083, China
* Correspondence: zyang22@csu.edu.cn

Abstract: Cross-modal hashing technology is a key technology for real-time retrieval of large-scale multimedia data in real-world applications. Although the existing cross-modal hashing methods have achieved impressive accomplishment, there are still some limitations: (1) some cross-modal hashing methods do not make full consider the rich semantic information and noise information in labels, resulting in a large semantic gap, and (2) some cross-modal hashing methods adopt the relaxation-based or discrete cyclic coordinate descent algorithm to solve the discrete constraint problem, resulting in a large quantization error or time consumption. Therefore, in order to solve these limitations, in this paper, we propose a novel method, named **D**iscrete **S**emantics-Guided **A**symmetric **H**ashing (DSAH). Specifically, our proposed DSAH leverages both label information and similarity matrix to enhance the semantic information of the learned hash codes, and the $\ell_{2,1}$ norm is used to increase the sparsity of matrix to solve the problem of the inevitable noise and subjective factors in labels. Meanwhile, an asymmetric hash learning scheme is proposed to efficiently perform hash learning. In addition, a discrete optimization algorithm is proposed to fast solve the hash code directly and discretely. During the optimization process, the hash code learning and the hash function learning interact, i.e., the learned hash codes can guide the learning process of the hash function and the hash function can also guide the hash code generation simultaneously. Extensive experiments performed on two benchmark datasets highlight the superiority of DSAH over several state-of-the-art methods.

Keywords: cross-modal retrieval; discrete optimization; hashing

Citation: Long, J.; Sun, L.; Hua, L.; Yang, Z. Discrete Semantics-Guided Asymmetric Hashing for Large-Scale Multimedia Retrieval. *Appl. Sci.* **2021**, *11*, 8769. https://doi.org/10.3390/app11188769

Received: 13 July 2021
Accepted: 12 September 2021
Published: 21 September 2021

1. Introduction

In recent years, due to the rapid development of multimedia Internet of Things technologies, there has been an explosive growth in the amount of multimedia network data. Consequently, the current unimodal search methods can no longer meet the multimedia data retrieval requirements in the complex environment of the new information era. Therefore, cross-modal retrieval methods [1–3] have received increasing attention from the information retrieval community and have become a hot research topic in both academia and industry. Specifically, given a query in one modality (such as text), users expect to return its semantically related modality (text) or different modalities (such as images or videos). For decades, as a branch of **n**earest **n**eighbor **s**earch (NNS), the hashing technique has been an active research field in information retrieval community due to the following advantages: (1) Lower storage cost and (2) improved retrieval speed with the hardware-friendly bit-wise XOR and bit-count operations [4]. In the hash code learning process, the learned hash codes should meet a condition, that is, similar instances have similar hash codes in the Hamming space, and vice versa. Among the practical applications are the image retrieval [5,6] and person re-identification [7,8].

According to the learning principle, existing cross-modal hashing methods can be mainly divided into the following categories: **Unsupervised cross-modal hashing methods** [9–15]: Unsupervised hashing methods focus on discovering the intra- and inter-relations of multiple heterogeneous modalities without label information to learn the hash codes and corresponding hash functions. However, due to the lack of label information, the retrieval performance of unsupervised cross-modal hashing methods will be affected accordingly, i.e., $\mathbf{x} \xrightarrow{f(\mathbf{x};\grave{})} \mathbf{z} \xrightarrow{d(\mathbf{z};\varphi)} \hat{\mathbf{x}}$, where $f(\cdot;\theta)$ can be considered as either a hash mapping function or an encoder, $d(\mathbf{z};\varphi)$ can be regarded as a decoder. During the unsupervised learning process, the key design factor of this learning paradigm is the choice of a suitable metric that can measure the distance between $\mathbf{x}$ and $\tilde{\mathbf{x}}$, i.e., the distance between $\mathbf{x}, \tilde{\mathbf{x}}$ should be minimized: $\min_{\theta,\varphi} \mathbb{E}||\mathbf{x} - \tilde{\mathbf{x}}||_p$, where typically $p = 2$. Then, the hash codes can be computed by $sgn(\mathbf{z})$. **Supervised cross-modal hashing methods** [16–22]: Supervised hashing methods have obtained satisfactory retrieval results by using label information, and have been extensively studied, i.e., $\mathbf{x} \xrightarrow{f(\mathbf{x};\theta)} \mathbf{z} \xrightarrow{c(\mathbf{z})} \mathbf{L}$, where $f(\cdot;\theta)$ denotes a hash mapping function that selects certain latent representation $\mathbf{z}$, $c(\cdot)$ denotes a classifier, $\mathrm{sgn}(\cdot)$ denotes the element-wise sign operation, $\mathbf{L}$ denotes the labels, and $\mathbf{B}$ denotes the learned hash codes. Then, the hash codes can be computed by $sgn(\mathbf{z})$.

Although supervised cross-modal hashing methods have achieved significant success, they still have the following limitations: (1) **Limited semantic utilization**. Converting the label matrix directly to the similarity matrix will lead to a semantic loss, resulting in a large semantic gap, especially when facing multi-label datasets. (2) **Inefficient learning strategy**. Some cross-modal hashing methods are based on symmetric learning strategies, resulting in a worse retrieval performance than asymmetric learning ones. (3) **Flawed optimization strategy**. As the optimization process of the hash codes is discrete, the existing optimization strategies are mainly based on two kinds, one is to use the relaxation-based strategy, which will lead to a large quantization error; the other is to use bit-to-bit optimization strategy, such as Discrete Cyclic Coordinate (DCC) descent [23]. Although the problem of quantization error is solved, optimizing the entire hash code requires k iterations, where k is the hash code length, thus the optimization process is very time-consuming.

In order to solve the above limitations, in this paper, we proposed a novel yet simple but effective method, named **D**iscrete **S**emantics-Guided **A**symmetric **H**ashing (DSAH). Specifically, DSAH handles the nonlinear relations in different modalities with a kernelization technique, then an asymmetric learning scheme is proposed to effectively perform the hash function learning and hash code learning processes; meanwhile, our proposed DSAH considers the following aspects. First, we leverage both label information and similarity matrix to enhance the semantic information of the learned hash codes. Then, we solve the problems of matrix sparsity and outlier processing. In addition, a discrete optimization algorithm is proposed to solve the discrete problems. Our major contributions can be summarized as follows:

1. A novel supervised cross-modal hashing method, i.e., DSAH, is proposed to learn the discriminative compact hash codes for large-scale retrieval tasks. DSAH takes the label information and similarity matrix into consideration, which can improve the discriminative capability of the learned hash codes, and solves the problems of matrix sparseness and outlier processing.
2. An asymmetric learning scheme with real-valued embeddings is proposed to effectively learn the hash function and the hash codes.
3. Comprehensive experiments are conducted on two famous datasets. The experimental results demonstrate that DSAH outperforms some state-of-the-art baselines.

The remainder of this paper is organized as follows. Section 2 briefly reviews the related works. Section 3 introduces the details of the DSAH model and presents the alternative optimization algorithm. In Section 4, we give the results of experiments performed. Finally, we present the conclusions in Section 5.

2. Related Works

Cross-modal hashing retrieval has been a widely used technology in the field of information retrieval, machine intelligence, and computer vision. As mentioned above, cross-modal hashing retrieval technology can be divided into supervised and unsupervised categories. However, due to the space limitation, we refer readers to some surveys [4,24] for a more comprehensive coverage of popular hashing methods.

2.1. Unsupervised Hashing

Unsupervised hashing methods do not leverage label information for the training dataset. For example, Local Sensitive Hashing (LSH) [25] and its variants [26,27] use random projections to map instances into a Hamming space. Iterative Quantization (ITQ) [28] optimizes the projection by PCA, and then learns an orthogonal rotation matrix to bridge the quantization gap. Unsupervised semantic deep hashing (USDH) [29] uses semantic information to guide the training of hash mapping function. Unsupervised Deep Video Hashing (UDVH) [30] learns the hash codes in a self-taught manner by jointly integrating discriminative video representation with optimal code learning. Neighborhood Discriminant Hashing (NDH) [31] learns hash function by preserving the neighborhood discriminative information in Hamming space. Collective Matrix Factorization Hashing (CMFH) [32] learns a shared common latent space by a collective matrix factorization algorithm, and then adopts a thresholding operation to generate the hash codes. Fusion Similarity Hashing (FSH) [33] proposes a novel fusion similarity method for hash learning by capturing the latent relations between different heterogeneous modalities.

2.2. Supervised Hashing

Unlike unsupervised hashing methods, supervised ones utilize the labels to improve the retrieval performance. For example, Semantic Correlation Maximization (SCM) [34] preserves the pairwise similarities of the training dataset to learn the hash functions. Semantic Preserving Hashing (SePH) [35] constructs a new semantic affinity matrix into a probability distribution, and then learns the hash codes by using an approximate-based scheme. Discrete Cross-modal Hashing (DCH) [36] learns the hash codes in a bit-by-bit manner by using the discrete cyclic coordinate descent (DCC) algorithm. Label Consistent Matrix Factorization Hashing (LCMFH) [37] directly uses semantic labels to guide the hashing learning procedure. Scalable Discrete Matrix Factorization Hashing (SCRATCH) [38] is a two-step hashing method, which first generates the hash codes, and then learns the hash functions based on the learned hash codes. Nonlinear Robust Discrete Hashing (NRDH) [39] uses a nonlinear model to solve the generalization error caused by kernelization, and generates compact hash codes by constructing an asymmetric hash framework and discrete optimization algorithms. Scalable Deep Asymmetric Hashing (SDAH) [40] builds an asymmetric unequal-dimensional hash learning framework by exploring the information content of different modalities. Nonlinear Supervised Discrete Hashing (NSDH) [41] consists of two parts, the first part is a semantic enhancement descriptor, which is used to extract comprehensive latent representations of heterogeneous multimedia data, and the second part is a fast discrete optimization module, which is used to learn discriminative compact hash codes. Subspace Relation Learning for Cross-modal Hashing (SRLCH) [42] handles relationships of labels in a semantic subspace to make similar instances from different modalities closer in the binary Hamming space.

2.3. Deep Hashing

Recently, with the great success of deep learning in the field of representation learning, many deep hashing methods [13,15,40,43] have been proposed. For example, Deep Semantic-Alignment Hashing (DSAH) [44] is an unsupervised hashing method, which explores the similarity information of different modalities and proposes a semantic-alignment loss to learn the hash codes. Unsupervised Deep Cross-modal Hashing with Virtual Label Regression (UDCH-VLR) [45] proposes a novel unified learning framework to jointly

perform deep hash function training, virtual label learning, and regression. Deep Saliency Hashing (DSaH) [46] is a two-step end-to-end model, which mines salient regions and learns semantic-preserving hash codes simultaneously. Supervised Hierarchical Deep Cross-modal Hashing (SHDCH) [47] learns the hash codes by explicitly delving into the hierarchical labels. Deep Semantic cross-modal hashing with Correlation Alignment (DSCA) [48] designs two deep neural networks for image and text modality separately, and learns two hash functions. First, due to the non-smooth property of the discrete optimization causing the problem of unavailable gradient in back-propagation, these methods use the relaxation-based optimization strategies to handle the problem. Even though high retrieval performance is considered to be achieved, these methods still have a large quantization error and can only produce sub-optimal hash codes. Second, they all need large computing resources (e.g., GPUs) and a massive training dataset, which makes them fairly computationally expensive.

3. The Proposed DSAH Framework

In this section, we introduce our proposed DSAH model. The framework of DSAH is shown in Figure 1, which consists of three main parts: hash function learning, label alignment scheme and asymmetric learning framework. We demonstrate each part in the following section in detail.

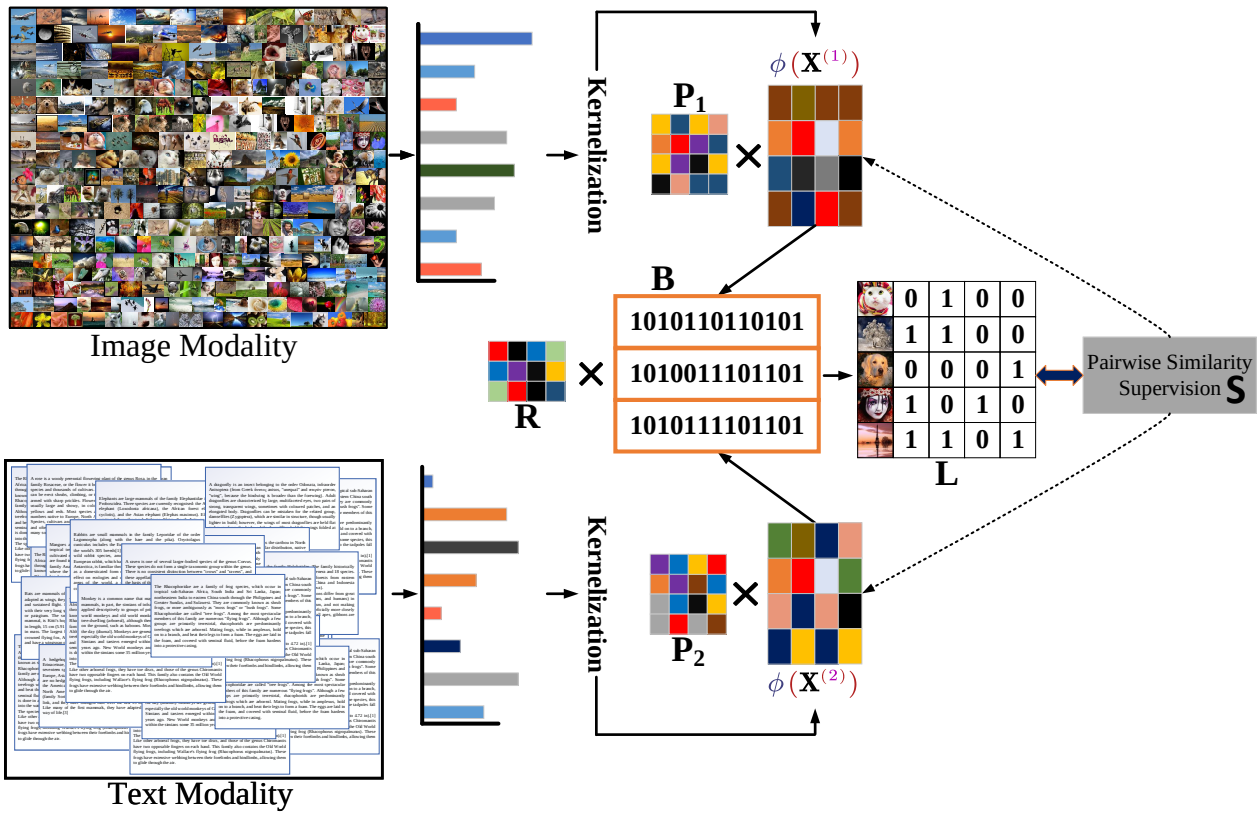

Figure 1. The framework of **D**iscrete **S**emantics-Guided **A**symmetric **H**ashing (DSAH). In our proposed DSAH, we first handle the nonlinear relations in different modalities with the kernelization, then an asymmetric learning scheme is proposed to effectively perform the hash learning process; meanwhile, our proposed method fully considers the label information to enhance the semantic information. In addition, a discrete optimization algorithm is proposed to solve the discrete problems.

3.1. Definitions

Suppose that the multimedia training data contains M modalities, represented by $\mathbf{X} = \{\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, ..., \mathbf{X}^{(M)}\}$, where $\mathbf{X}^{(m)} = \{\mathbf{x}_1^{(m)}, \mathbf{x}_2^{(m)}, ..., \mathbf{x}_n^{(m)}\} \in \mathbb{R}^{d_m \times n}$ is the m-th modality

data features and d_m is the dimension of modality $\mathbf{X}^{(m)}$. In this paper, we focus on the supervised hashing paradigm; therefore, label information $\mathbf{L} \in \{0,1\}^{c\times n}$ is available, c denotes the number of categories, and $l_i^j = 1$ indicates the j-th instance belongs to category i, and $l_{ij} = 0$ otherwise. $\mathbf{B} \in \mathbb{R}^{k\times n}$ denotes the hash codes, where k is the length of hash codes. $f(\cdot)$ denotes the hash function. The main notations used in this paper are listed in Table 1.

Table 1. Notations.

Notation	Explanations
$\mathbf{X}^{(t)} \in \mathbb{R}^{d_m\times n}$	Features of heterogeneous modalities
$\mathbf{L} \in \mathbb{R}^{c\times n}$	Label information
$\mathbf{B} \in \{-1,1\}^{k\times n}$	Hash codes
$\mathbf{P}_t \in \mathbb{R}^{k\times q}$	Hash mapping matrix
$\mathbf{D} \in \mathbb{R}^{c\times c}$	Projection matrix for label information
$\mathbf{V} \in \{-1,1\}^{k\times n}$	Auxiliary matrix
$\mathbf{J}_b \in \mathbf{R}^{k\times n}$	Auxiliary matrix
d_m	Dimension of modality $\mathbf{X}^{(m)}$
n	Number of instances
c	Number of categories
q	Number of Kernelized features

3.2. Hash Function Learning

3.2.1. Kernelization

Kernelization is a widely used technique to handle the nonlinear relations in different modalities. Therefore, in this paper, we use Radical Basis Function (RBF) kernel to express the nonlinear correlations among original high dimensional features [49–51]. Specifically, we define the RBF function $\phi(\cdot)$ as follows:

$$\phi(\mathbf{x}_i) = \begin{bmatrix} \exp\left(\frac{-||\mathbf{x}_i-\mathbf{a}_1||}{2\sigma^2}\right), \\ \ldots, \\ \exp\left(\frac{-||\mathbf{x}_i-\mathbf{a}_q||}{2\sigma^2}\right). \end{bmatrix}, \tag{1}$$

where $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_q]$ denotes the randomly chosen q anchors from the database and σ is the free parameter. Therefore, the complex original feature $\mathbf{X}^{(m)} \in \mathbb{R}^{d_m\times n}$ can be converted into a nonlinear relation representation $\phi(\mathbf{X}^{(m)}) \in \mathbb{R}^{q\times n}$.

3.2.2. Feature Mapping

The aim of DSAH is to project the original features to compact hash codes. In this paper, we adopt two linear projections as the hash mapping functions for image modality $\mathbf{X}^{(1)}$ and text modality $\mathbf{X}^{(2)}$, respectively.

$$\begin{aligned} f_1(\mathbf{X}^{(1)}) &= sgn(\mathbf{P}_1\phi(\mathbf{X}^{(1)})), \\ f_2(\mathbf{X}^{(2)}) &= sgn(\mathbf{P}_2\phi(\mathbf{X}^{(2)})), \end{aligned} \tag{2}$$

where $\mathbf{P}_1 \in \mathbb{R}^{k\times q}$ and $\mathbf{P}_2 \in \mathbb{R}^{k\times q}$ are the hash mapping matrices, which map specific kernel features into Hamming subspace, and $f_1(\cdot)$ and $f_2(\cdot)$ are hash functions for image modality and text modality, respectively.

3.3. Label Alignment Scheme

As described above, labels contain rich semantic information, directly converting the complex label vectors into binary semantic matrix will cause the loss of semantic information. The results of Gui's work [52] demonstrate that the ordinary least squares

regression is sensitive to the boundary contour. Inspired by the work in [53], we consider $\ell_{p,q}$ norm instead of ℓ_2 norm to handle the problem, the $\ell_{p,q}$ norm can be formulated as

$$\min_{\mathbf{E}} ||\mathbf{E}||_{p,q} = \min_{\mathbf{E}} (\sum_{i=1}^{c} (\sum_{j=1}^{n} |\mathbf{E}_{ij}|^p)^{q/p})^{1/q}, \tag{3}$$

where $\mathbf{E} = \mathbf{R}^\top \mathbf{B} - \mathbf{L}$ and $\mathbf{R} \in \mathbb{R}^{k \times c}$ is the semantic projection matrix. It is easy to find that when $p = q = 2$, Equation (3) is a standard Frobenius norm. However, in order to improve the robustness of the model for outliers and the sparsity of the label alignment matrix, we need to redefine the values of p, q. In general, the sparsity of the model can be guaranteed when the constraint conditions satisfy $p \geq 2$ and $0 \leq q \leq 2$. Therefore, in the paper, we set $p = 2$ and $q = 1$, as if $q = 0$, the problem is not convex. Then, we can rewrite the Equation (3) as $\min_{\mathbf{E}} ||\mathbf{E}||_{2,1}$. After some algebraic manipulations, we obtain

$$\min_{\mathbf{R},\mathbf{B}} tr(\mathbf{E}^\top \mathbf{D} \mathbf{E}), \tag{4}$$

where $\mathbf{D} \in \mathbb{R}^{c \times c}$ is the diagonal matrix, and the i-th element of $\mathbf{D}$ is defined as $d_{ii} = \frac{1}{2||e_{(i)}||_2}$, where $e_{(i)}$ is the i-th row of $\mathbf{E}$.

3.4. Asymmetric Learning Framework

We briefly review the related work Supervised Hashing with Kernels (KSH) [51], the symmetric learning framework can be formulated as

$$\begin{aligned} &\min_{\mathbf{B}} ||\mathbf{B}^\top \mathbf{B} - k\mathbf{S}||_F^2 \\ &s.t.\ \mathbf{B} \in \{-1, 1\}^{k \times n}, \end{aligned} \tag{5}$$

where $\mathbf{B}$ is the learned hash codes. However, there are two major problems of Equation (5): (1) It is very time-consuming to directly compute $\mathbf{S}$, as the similarity information $\mathbf{S}$ is a $n \times n$ matrix. (2) Some works [54,55] show that the use of an asymmetric learning framework can not only solve the problem of high time consumption, but also improves retrieval accuracy, because the value range of the asymmetric learning framework is wider than that of symmetric learning. Therefore, in this paper, we construct an asymmetric learning framework to learn the compact hash codes, that is,

$$\begin{aligned} \min_{\mathbf{B},\mathbf{P}_1,\mathbf{P}_2} &||(\mathbf{P}_1 \phi(\mathbf{X}^{(1)}))^\top \mathbf{B}_1 - k\mathbf{S}||_F^2 + ||(\mathbf{P}_2 \phi(\mathbf{X}^{(2)}))^\top \mathbf{B}_2 - k\mathbf{S}||_F^2 \\ &+ \alpha \sum_{i=1}^{2} ||\mathbf{B}_i - \mathbf{P}_i \phi(\mathbf{X}^{(i)})||_F^2, \\ &s.t.\ \mathbf{B}_i \in \{-1, 1\}^{k \times n} \end{aligned} \tag{6}$$

The advantages of Equation (6) are as follows:

1. The learning mode uses an efficient asymmetric learning architecture instead of a time-consuming symmetric one.
2. The use of the real-valued embeddings instead of the binary embeddings produces a close semantic similarity relation, and the value of the objective function is smaller.
3. The last term is used to reduce the quantization errors, which leads to a better retrieval performance.

However, there is a limitation of Equation (6) that is for the purpose of cross-modal retrieval tasks, we need to obtain a unified hash codes. Therefore, we need to consider another discrete constraint, i.e., $\min_{\mathbf{B}_1,\mathbf{B}_2} ||\mathbf{B}_1 - \mathbf{B}_2||_F^2$. In order to make the optimization easy, we

set a unified hash code $\mathbf{B} = \mathbf{B}_1 = \mathbf{B}_2$ instead of minimizing the constraint $\min_{\mathbf{B}_1,\mathbf{B}_2} ||\mathbf{B}_1 - \mathbf{B}_2||_F^2$, then Equation (6) can be rewritten as

$$\begin{aligned} \min_{\mathbf{B},\mathbf{P}_1,\mathbf{P}_2} & ||(\mathbf{P}_1\phi(\mathbf{X}^{(1)}))^\top \mathbf{B} - k\mathbf{S}||_F^2 + ||(\mathbf{P}_2\phi(\mathbf{X}^{(2)}))^\top \mathbf{B} - k\mathbf{S}||_F^2 \\ & + \alpha||\mathbf{B} - 0.5\Big(\mathbf{P}_1\phi(\mathbf{X}^{(1)}) + \mathbf{P}_2\phi(\mathbf{X}^{(2)})\Big)||_F^2, \\ & s.t.\ \mathbf{B} \in \{-1,1\}^{k\times n} \end{aligned} \tag{7}$$

where α is the balance parameter.

3.5. The Joint Framework

Combining the above constraints and individual objective function, we obtain

$$\begin{aligned} \min_{\mathbf{B},\mathbf{R},\mathbf{P}_1,\mathbf{P}_2} & ||(\mathbf{P}_1\phi(\mathbf{X}^{(1)}))^\top \mathbf{B} - k\mathbf{S}||_F^2 + ||(\mathbf{P}_2\phi(\mathbf{X}^{(2)}))^\top \mathbf{B} - k\mathbf{S}||_F^2 + tr(\mathbf{E}^\top \mathbf{D}\mathbf{E}) \\ & + \alpha||\mathbf{B} - 0.5\Big(\mathbf{P}_1\phi(\mathbf{X}^{(1)}) + \mathbf{P}_2\phi(\mathbf{X}^{(2)})\Big)||_F^2 + \gamma\, Re(\mathbf{P}_1^*, \mathbf{P}_2^*, \mathbf{R}), \\ & s.t.\ \mathbf{B} \in \{-1,1\}^{k\times n} \end{aligned} \tag{8}$$

where γ is the trade-off parameter, $\mathbf{P}_i^* = \mathbf{P}_i\phi(\mathbf{X}^{(i)})$, $Re(\cdot) = ||\cdot||_F^2$ is the Frobenius norm regularization term, which is used to prevent overfitting.

3.6. Optimization

In this part, we use an alternating strategy to solve the four variables $\mathbf{B}, \mathbf{R}, \mathbf{P}_1, \mathbf{P}_2$ in Equation (8), as the four variables are coupled with each other. The problem is split into four steps as follows.

Fix $\mathbf{R}, \mathbf{P}_1, \mathbf{P}_2$, *update* $\mathbf{B}$. The sub-problem of Equation (8) related to $\mathbf{B}$ can be formulated as

$$\begin{aligned} \min_{\mathbf{B}} & ||(\mathbf{P}_1\phi(\mathbf{X}^{(1)}))^\top \mathbf{B} - k\mathbf{S}||_F^2 + ||(\mathbf{P}_2\phi(\mathbf{X}^{(2)}))^\top \mathbf{B} - k\mathbf{S}||_F^2 + tr(\mathbf{E}^\top \mathbf{D}\mathbf{E}) \\ & + \alpha||\mathbf{B} - 0.5\Big(\mathbf{P}_1\phi(\mathbf{X}^{(1)}) + \mathbf{P}_2\phi(\mathbf{X}^{(2)})\Big)||_F^2, \\ & s.t.\ \mathbf{B} \in \{-1,1\}^{k\times n} \end{aligned} \tag{9}$$

In the next step, we need to solve the following problem:

$$\begin{aligned} \min_{\mathbf{B}} & tr\Big(-2k\mathbf{S}\mathbf{B}^\top \mathbf{P}_1\phi(\mathbf{X}^{(1)}) - 2k\mathbf{S}\mathbf{B}^\top \mathbf{P}_2\phi(\mathbf{X}^{(2)}) + \mathbf{B}^\top \mathbf{R}\mathbf{D}\mathbf{R}^\top \mathbf{B} \\ & -2\mathbf{B}^\top \mathbf{R}\mathbf{D}\mathbf{L} - 2\alpha\mathbf{C}\mathbf{B}^\top\Big) \\ & s.t.\ \mathbf{B} \in \{-1,1\}^{k\times n} \end{aligned} \tag{10}$$

where $\mathbf{C} = 0.5(\mathbf{P}_1\phi(\mathbf{X}^{(1)}) + \mathbf{P}_2\phi(\mathbf{X}^{(2)}))$. As the $\mathbf{B}$ is the discrete value, it is challenging to directly solve the value of $\mathbf{B}$. In this solution, we propose an augmented Lagrangian multiplier (ALM) [39] to compute $\mathbf{B}$. Specifically, we introduce an auxiliary value $\mathbf{V} \in \{-1,1\}^{k\times n}$ to replace the $\mathbf{B}$ of second term, i.e., $\mathbf{B}^\top \mathbf{R}\mathbf{D}\mathbf{R}^\top \mathbf{B}$. Then, we obtain the following formula:

$$\begin{aligned} \min_{\mathbf{B}} & tr\Big(-2k\mathbf{S}\mathbf{B}^\top \mathbf{P}_1\phi(\mathbf{X}^{(1)}) - 2k\mathbf{S}\mathbf{B}^\top \mathbf{P}_2\phi(\mathbf{X}^{(2)}) - 2\alpha\mathbf{C}\mathbf{B}^\top \\ & +\mathbf{B}^\top \mathbf{R}\mathbf{D}\mathbf{R}^\top \mathbf{V} - 2\mathbf{B}^\top \mathbf{R}\mathbf{D}\mathbf{L}\Big) + \frac{\xi}{2}||\mathbf{B} - \mathbf{V} + \frac{\mathbf{J}_b}{\xi}||_F^2, \\ & s.t.\ \{\mathbf{B}, \mathbf{V}\} \in \{-1,1\}^{k\times n} \end{aligned} \tag{11}$$

where $\mathbf{J}_b$ measures the gap between $\mathbf{B}$ and $\mathbf{V}$.

Then, the value of $\mathbf{B}$ can be solved with a closed-form solution:

$$\begin{aligned}\mathbf{B} = sgn(&2k\mathbf{P}_1\phi(\mathbf{X}^{(1)})\mathbf{S} + 2k\mathbf{P}_2\phi(\mathbf{X}^{(2)})\mathbf{S} + 2\alpha\mathbf{C} - \mathbf{R}\mathbf{D}\mathbf{R}^\top\mathbf{V} \\ &+ 2\mathbf{R}\mathbf{D}\mathbf{L} + \xi\mathbf{V} - \mathbf{J}_b).\end{aligned} \tag{12}$$

However, the computational complexity of $\mathbf{P}_*\phi(\mathbf{X}^{(*)})\mathbf{S}|_{*=1}^2$ is $\mathcal{O}(n^2)$, which is not suitable for large-scale retrieval tasks. To address this problem, we use the label matrix $\mathbf{L} \in \mathbb{R}^{c\times n}$ to replace the similarity matrix $\mathbf{S} \in \mathbb{R}^{n\times n}$. Specifically, we let $\tilde{\mathbf{L}}_{ij} = l_{ij}/||\mathbf{l}_i||_2$, as the element at the i-th row and the j-th column in the matrix $\mathbf{L}$. Then, the similarity matrix $\mathbf{S}$ can be rewritten as

$$\mathbf{S} = 2\tilde{\mathbf{L}}^\top\tilde{\mathbf{L}} - \mathbf{1}_n\mathbf{1}_n{}^\top, \tag{13}$$

where $\mathbf{1}_n$ is a vector with all elements being 1. Therefore, we can rewrite the term $\mathbf{P}_*\phi(\mathbf{X}^{(*)})\mathbf{S}|_{*=1}^2$ as

$$\mathbf{P}_*\phi(\mathbf{X}^{(*)})\mathbf{S}|_{*=1}^2 = 2\mathbf{P}_*\phi(\mathbf{X}^{(*)})\tilde{\mathbf{L}}^\top\tilde{\mathbf{L}} - \mathbf{P}_*\phi(\mathbf{X}^{(*)})\mathbf{1}_n\mathbf{1}_n{}^\top|_{*=1}^2, \tag{14}$$

which consumes $\mathcal{O}((q+c)kn)$.

Fix $\mathbf{B}$, *update* $\mathbf{V}$. The sub-problem related to $\mathbf{V}$ can be formulated as

$$\begin{aligned}&\min_{\mathbf{V}} tr(\mathbf{B}^\top\mathbf{R}\mathbf{D}\mathbf{R}^\top\mathbf{V}) + \frac{\xi}{2}||\mathbf{B} - \mathbf{V} + \frac{\mathbf{J}_b}{\xi}||_F^2. \\ &\quad s.t.\ \{\mathbf{B},\mathbf{V}\} \in \{-1,1\}^{k\times n}\end{aligned} \tag{15}$$

Then, the value of $\mathbf{V}$ can be solved with a closed-form solution,

$$\mathbf{V} = sgn(-\mathbf{R}\mathbf{D}^\top\mathbf{R}^\top\mathbf{B} + \xi\mathbf{B} + \mathbf{J}_b). \tag{16}$$

Update $\mathbf{J}_b$. The sub-problem related to $\mathbf{J}_b$ can be updated as

$$\mathbf{J}_b = \mathbf{J}_b + \xi(\mathbf{B} - \mathbf{V}),\ \xi = \rho\xi, \tag{17}$$

where ρ is a parameter to control the convergence speed.

Fix $\mathbf{B}, \mathbf{P}_1, \mathbf{P}_2$, *update* $\mathbf{R}$. The sub-problem of Equation (8) related to $\mathbf{R}$ can be formulated as

$$\min_{\mathbf{R}} tr(\mathbf{E}^\top\mathbf{D}\mathbf{E}) + \gamma\ Re(\mathbf{R}). \tag{18}$$

In the next step, we need to solve the following problem:

$$\min_{\mathbf{R}} tr(\mathbf{B}^\top\mathbf{R}\mathbf{D}\mathbf{R}^\top\mathbf{B} - 2\mathbf{B}\mathbf{L}^\top\mathbf{D}\mathbf{R}^\top) + \gamma tr(\mathbf{R}\mathbf{R}^\top). \tag{19}$$

Setting the derivative Equation (20) w.r.t $\mathbf{R}$ to 0, we obtain

$$\mathbf{B}\mathbf{B}^\top\mathbf{R}\mathbf{D} + \gamma\mathbf{R} = \mathbf{B}\mathbf{L}^\top\mathbf{D}. \tag{20}$$

We transform Equation (20) into

$$\mathbf{B}\mathbf{B}^\top\mathbf{R} + \gamma\mathbf{R}\mathbf{D}^{-1} = \mathbf{B}\mathbf{L}^\top. \tag{21}$$

Then, it can be seen that Equation (21) is a Sylvester equation. Therefore, the value of $\mathbf{R}$ can be easily solved. Due to the space limitation, the detail about the solution is not given here.

Fix $\mathbf{B}, \mathbf{R}, \mathbf{P}_2$, *update* $\mathbf{P}_1$. The sub-problem of Equation (8) related to $\mathbf{P}_1$ can be formulated as

$$\begin{aligned} \min_{\mathbf{P}_1} & ||(\mathbf{P}_1\phi(\mathbf{X}^{(1)}))^\top \mathbf{B} - k\mathbf{S}||_F^2 + \alpha||\mathbf{B} - 0.5\Big(\mathbf{P}_1\phi(\mathbf{X}^{(1)}) + \mathbf{P}_2\phi(\mathbf{X}^{(2)})\Big)||_F^2 \\ & + \gamma\, Re(\mathbf{P}_1^*), \\ & s.t.\ \mathbf{B} \in \{-1, 1\}^{k\times n} \end{aligned} \tag{22}$$

Setting the derivative Equation (22) w.r.t $\mathbf{P}_1$ to 0, we obtain

$$\begin{aligned} & 2\mathbf{P}_1\phi(\mathbf{X}^{(1)})\phi(\mathbf{X}^{(1)})^\top - 2k\mathbf{B}\mathbf{S}^\top\phi(\mathbf{X}^{(1)})^\top - 4\alpha\mathbf{B}\phi(\mathbf{X}^{(1)})^\top + 2\alpha\mathbf{P}_1\phi(\mathbf{X}^{(1)})\phi(\mathbf{X}^{(1)})^\top \\ & + 2\alpha\mathbf{P}_2\phi(\mathbf{X}^{(2)})\phi(\mathbf{X}^{(1)})^\top + 2\gamma\mathbf{P}_1(\mathbf{X}^{(1)})\phi(\mathbf{X}^{(1)})^\top = 0 \end{aligned} \tag{23}$$

Then, the value of $\mathbf{P}_1$ can be solved with a closed-form solution:

$$\begin{aligned} \mathbf{P}_1 = & (k\mathbf{B}\mathbf{S}^\top\phi(\mathbf{X}^{(1)})^\top + 2\alpha\mathbf{B}\phi(\mathbf{X}^{(1)})^\top - \alpha\mathbf{P}_2\phi(\mathbf{X}^{(2)})\phi(\mathbf{X}^{(1)})^\top)\cdot \\ & ((1+\alpha+\gamma)\phi(\mathbf{X}^{(1)})\phi(\mathbf{X}^{(1)})^\top)^{-1} \end{aligned} \tag{24}$$

where $\mathbf{S}$ is also transformed using Equation (13); then, we have

$$\mathbf{B}\mathbf{S}^\top\phi(\mathbf{X}^{(1)})^\top = 2(\mathbf{B}\tilde{\mathbf{L}}^\top)(\phi(\mathbf{X}^{(1)})\tilde{\mathbf{L}}^\top)^\top - (\mathbf{B}\mathbf{1}_n)(\phi(\mathbf{X}^{(1)})\mathbf{1}_n)^\top, \tag{25}$$

which consumes $\mathcal{O}((q+k)cn)$.

Fix $\mathbf{B}, \mathbf{R}, \mathbf{P}_1$, *update* $\mathbf{P}_2$. The sub-problem of Equation (8) related to $\mathbf{P}_2$ can be formulated as

$$\begin{aligned} \min_{\mathbf{P}_2} & ||(\mathbf{P}_2\phi(\mathbf{X}^{(2)}))^\top \mathbf{B} - k\mathbf{S}||_F^2 + \alpha||\mathbf{B} - 0.5\Big(\mathbf{P}_1\phi(\mathbf{X}^{(1)}) + \mathbf{P}_2\phi(\mathbf{X}^{(2)})\Big)||_F^2 \\ & + \gamma\, Re(\mathbf{P}_2^*), \\ & s.t.\ \mathbf{B} \in \{-1, 1\}^{k\times n} \end{aligned} \tag{26}$$

It is easy to find that the optimization of $\mathbf{P}_2$ is almost identical to $\mathbf{P}_1$-subproblem. Then, the value of $\mathbf{P}_2$ can be solved with a closed-form solution:

$$\begin{aligned} \mathbf{P}_2 = & (k\mathbf{B}\mathbf{S}^\top\phi(\mathbf{X}^{(2)})^\top + 2\alpha\mathbf{B}\phi(\mathbf{X}^{(2)})^\top - \alpha\mathbf{P}_1\phi(\mathbf{X}^{(1)})\phi(\mathbf{X}^{(2)})^\top)\cdot \\ & ((1+\alpha+\gamma)\phi(\mathbf{X}^{(2)})\phi(\mathbf{X}^{(2)})^\top)^{-1} \end{aligned} \tag{27}$$

Moreover, the terms of $\phi(\mathbf{X}^{(1)})\phi(\mathbf{X}^{(1)})^\top$ and $\phi(\mathbf{X}^{(2)})\phi(\mathbf{X}^{(2)})^\top$ are constants and can be computed once before the iterative optimization.

The objective function is solved by iteratively updating four variables until the objective function converges or reaches the preset maximum number of iterations. The iterative optimization for solving the Equation (8) is summarized in Algorithm 1.

Algorithm 1 DSAH

Input: Training modalities $\mathbf{X} = \{\mathbf{X}^{(1)}, ..., \mathbf{X}^{(M)}\}$, labels $\mathbf{L}$, hash code length k, parameter α, γ, maximum iteration number T.

Output: Hash mapping functions $\mathbf{P}_1$ and $\mathbf{P}_2$.

Procedure:
1. Centralize $\mathbf{X}$ by means.
2. Computing Kernelized features $\phi(\mathbf{X})$.
3. Initialize $\mathbf{V}, \mathbf{B}, \mathbf{R}, \mathbf{P}_1, \mathbf{P}_2, \mathbf{J}_b$ by random initialization.
4. **Repeat**
   $\mathbf{B}$-Step: Update $\mathbf{B}$ according to (12).
   $\mathbf{V}$-Step: Update $\mathbf{V}$ according to (16).
   $\mathbf{J}_b$-Step: Update $\mathbf{J}_b$ according to (17).
   $\mathbf{R}$-Step: Update $\mathbf{R}$ according to (21).
   $\mathbf{P}_1$-Step: Update $\mathbf{P}_1$ according to (24).
   $\mathbf{P}_2$-Step: Update $\mathbf{P}_2$ according to (27).
   Until up to T.

3.7. Out-of-Sample Extension

In the query phase, the proposed DSAH can easily map the original high-dimensional instances into compact hash codes. Specifically, given a new query $\mathbf{x}_q^{(t)} \notin \mathbf{X}^{(m)}$, DSAH learns its corresponding hash codes by

$$\mathbf{b}_q^{(t)} = sgn(\mathbf{P}_t \phi(\mathbf{x}_q^{(t)})), \tag{28}$$

where $\phi(\mathbf{x}_q^{(t)})$ is the nonlinear kernelized embedding of $\mathbf{x}_q^{(t)}$.

3.8. Complexity Analysis

For each iteration, the time complexity is analyzed as follows. The time computational complexity of $\mathbf{B}$ is $\mathcal{O}(k^2c + kc^2 + (q + c + k)kn)$, $\mathbf{V}$ is $\mathcal{O}(kc^2 + k^2c + k^2n)$, $\mathbf{R}$ is $\mathcal{O}((k^2 + kc)n + k^2c + kc^2 + c^3)$, $\mathbf{P}_1$ and $\mathbf{P}_2$ are all $\mathcal{O}(q^3 + kq^2 + (kq + kc + qc + q^2)n)$. As $\{k, c, q\} \ll n$, the training complexity is $\mathcal{O}((kq + k^2 + kc + q^2)n)$. Given the iteration T, the overall training complexity for DSAH is $\mathcal{O}((kq + k^2 + kc + q^2)nT)$, where $T \ll \{k, c, q\}$ is very small , which is linear to the training set size. Therefore, DSAH is highly scalable for large-scale cross-modal retrieval tasks.

4. Experiments

4.1. Datasets

To evaluate the performance of DSAH, we conducted experiments on two widely used datasets, i.e., MIRFlickr [56] and NUS-WIDE [57] datasets.

4.1.1. MIRFlickr

It contains 25,000 instances collected from open website, which are annotated by at least one of 24 tags. Similar to the work in [38], we ignored the instances that textual tags appear less than 20 times and finally selected 20,015 instances. We randomly selected $2k$ instances as the query set and the rest as the retrieval set. Each image is represented as a 512-D GIST feature and each text is represented as a 1386-D bag-of-word (BOW) vector.

4.1.2. NUS-WIDE

It contains 269,648 instances collected from Flick with 5018 unique tags and 81 ground-truth concepts that can be used for evaluation. Similar to the work in [38], we selected the ten most frequent tags and corresponding 186,577 instances. We randomly selected 2000

instances as the query set and the rest as the retrieval set. Each image is represented as a 500-D SIFT feature and each text is represented as a 1000-D bag-of-word (BOW) vector.

4.2. Methodology

To verify the effectiveness of our proposed DSAH method, seven state-of-the-art cross-modal hashing methods are compared. Among them, CMFH [32] and FSH [33] are unsupervised cross-modal hashing methods, and SCM-Seq [34], SePH-km [35], DCH [36], LCMFH [37], and SRLCH [42] are supervised ones. We have briefly introduced the compared baselines in Section 2. For fair comparison, the experimental results with citations are copied from the corresponding works.

To evaluate our proposed DSAH, we conducted two cross-modal retrieval tasks: (1) "Image2Text" using an image query to retrieve texts; (2) "Text2Image" using a text query to retrieve images. In this paper, three widely-used evaluation measures are used to evaluate the retrieval performance, i.e., mean average precision (mAP), precision-recall curves (PR) and precisions w.r.t top-k returned image (P@k).

4.3. Implementation Details

DSAH consists of several parameters, i.e., α and γ. We tune the balance parameters, i.e., α and γ using grid search, and the best performance is achieve when $\{\alpha = 10^{-1}, \gamma = 10^{-3}\}$ and $\{\alpha = 10^{-2}, \gamma = 10^{-3}\}$ on MIRFlickr and NUS-WIDE datasets, respectively. q is the number of kernel and optimal performance is obtained when $q = 2000$. ξ and ρ are used for ALM algorithm and the best performance is obtained when $\{\xi = 10^{-2}, \rho = 1.5\}$ and $\{\xi = 10^{-1}, \rho = 1.5\}$ on MIRFlickr and NUS-WIDE datasets, respectively. All our experiments are conducted on a workstation with a Intel Xeon Silver 4210 CPU@2.20 GHz of 10 cores and 128 G RAM.

4.4. Results

Tables 2 and 3 show the mAP scores of different compared cross-modal hashing methods at 8 bits, 16 bits, 32 bits, 64 bits, and 128 bits on MIRFlickr and NUS-WIDE datasets, respectively. Note that the mAP metric is one of the comprehensive evaluation criterions used to measure the effectiveness of the proposed method. From these tables, it can be observed that the mAP scores of DSAH are higher than most compared baselines with different code lengths on the two datasets. In the seven compared baselines, only SRLCH, LCMFH, and DCH obtain satisfactory retrieval performance. The main reason is that they learn the common latent representation across different modalities through matrix factorization operations, thus the common latent representation can be used as a bridge to solve the heterogeneous gap between different modalities. However, they ignore the use of an asymmetric learning framework to enhance the semantic similarity of different modalities and the noises contained in the labels. In contrast, our proposed DSAH leverages both the similarity matrix and label information to enhance the semantic information of the learned hash codes, and solves the problem of noises contained in the labels. Specifically, on the MIRFlickr dataset, compared to the best baselines, i.e., SRLCH, the mAP scores of DSAH have an increase of 2.7% on average, and on the NUS-WIDE dataset, DSAH obtains the highest mAP scores of all compared baselines, which demonstrates the efficacy of DSAH. Meanwhile, by comparing supervised cross-modal hashing methods and unsupervised ones on the two datasets, we find that the supervised hashing methods, i.e., SCM-Seq, SePH-km, DCH, LCMFH, and SRLCH, can always outperform the unsupervised hashing methods, i.e., CMFH and FSH, as the supervised information can improve the ability of hash learning process. In addition, the mAP scores of most cross-modal hashing methods increase as the length of the hash codes becomes longer, revealing that the longer codes can handle more discriminative information. The performance on the T2I task, i.e., the use of text modality to retrieve image modality is better than that on the I2T task, i.e., the use of image modality to retrieve text modality. The reason is that the semantic information in text modality is more than that in image modality.

Table 2. Performance comparison on MIRFlickr dataset measured by mAP.

Task	Method	8 Bits	16 Bits	32 Bits	64 Bits	128 Bits
I→T	CMFH	0.5599	0.5687	0.5680	0.5685	0.5687
	FSH	0.5911	0.6016	0.6149	0.6194	0.6242
	SCM-seq	0.6235	0.6373	0.6478	0.6537	0.6611
	SePH-km	0.6641	0.6685	0.6818	0.6830	0.6873
	DCH	0.6659	0.6738	0.6859	0.6897	0.7030
	LCMFH	0.6821	0.6812	0.6887	0.6909	0.7034
	SRLCH	0.7092	0.7113	0.7241	0.7276	0.7359
	DSAH	**0.7156**	**0.7236**	**0.7412**	**0.7498**	**0.7556**
T→I	CMFH	0.5615	0.5615	0.5606	0.5606	0.5608
	FSH	0.5869	0.5979	0.6114	0.6186	0.6251
	SCM-seq	0.6103	0.6206	0.6298	0.6372	0.6427
	SePH-km	0.7033	0.7076	0.7212	0.7293	0.7348
	DCH	0.7256	0.7511	0.7585	0.7681	0.7909
	LCMFH	0.7351	0.7308	0.7544	0.7689	0.7806
	SRLCH	0.7467	0.7613	0.7798	0.7899	0.8071
	DSAH	**0.7526**	**0.7782**	**0.8041**	**0.8163**	**0.8180**

Table 3. Performance comparison on NUS-WIDE dataset measured by mAP.

Task	Method	8 Bits	16 Bits	32 Bits	64 Bits	128 Bits
I→T	CMFH	0.3406	0.3437	0.3399	0.3409	0.3440
	FSH	0.3620	0.3732	0.3894	0.4014	0.4084
	SCM-seq	0.5013	0.5120	0.5422	0.5488	0.5483
	SePH-km	0.5256	0.5537	0.5627	0.5622	0.5698
	DCH	0.5840	0.5808	0.5907	0.5932	0.5843
	LCMFH	0.5955	0.6113	0.6286	0.6337	0.6412
	SRLCH	0.5789	0.5932	0.6378	0.6398	0.6529
	DSAH	**0.6231**	**0.6432**	**0.6517**	**0.6667**	**0.6791**
T→I	CMFH	0.3456	0.3498	0.3435	0.3486	0.3529
	FSH	0.3623	0.3717	0.3835	0.3973	0.4007
	SCM-seq	0.4709	0.4836	0.5067	0.5141	0.5161
	SePH-km	0.6102	0.6407	0.6515	0.6608	0.6651
	DCH	0.7106	0.7103	0.7098	0.7260	0.7223
	LCMFH	0.6765	0.7198	0.7389	0.7614	0.7667
	SRLCH	0.6874	0.6989	0.7567	0.7581	0.7875
	DSAH	**0.7324**	**0.7596**	**0.7756**	**0.7834**	**0.8024**

Figure 2 plots the precision–recall and P@*k* curves in the cases of 64-bit code length for all compared baselines on two datasets. From the figure, we can draw the following observations.

1. From the precision–recall curves, we can observe that the area under the precision–recall curves of DSAH is larger than the compared baselines, which shows the effectiveness of DSAH.
2. From the P@*k* curves, we can observe that DSAH outperforms the compared baselines in most cases, which further demonstrate its superiority.

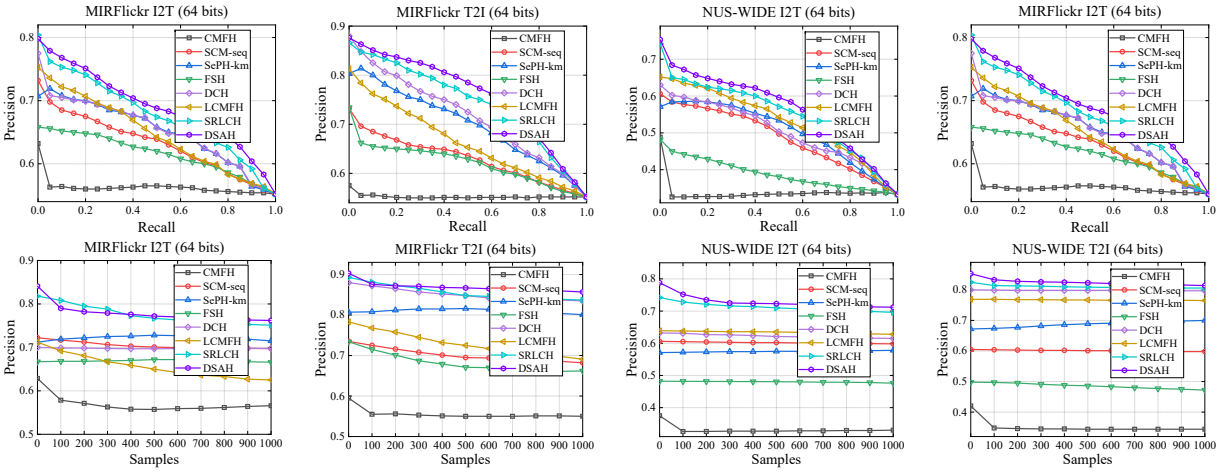

Figure 2. Precision–Recall and P@*k* curves obtained by different baselines and tested on MIRFlickr and NUS-WIDE datasets.

4.5. Further Study

4.5.1. Effects of Discrete Optimization

To validate the effects of the proposed discrete optimization strategy, we denote a variant of DSAH, named DSAH-Re. Specifically, we first relax the discrete constraints, then Equation (9) can be solved as

$$\min_{\mathbf{B}} \alpha \sum_{i=1}^{2} ||\mathbf{B} - \mathbf{P}_i \phi(\mathbf{X}^{(i)})||_F^2 + tr(\mathbf{E}^\top \mathbf{D}\mathbf{E}). \tag{29}$$

Setting the derivative Equation (29) w.r.t. $\mathbf{B}$ to 0, and the value of $\mathbf{B}$ can be solved with a closed-form solution:

$$\mathbf{B} = (\mathbf{R}\mathbf{D}\mathbf{R}^\top)^{-1}\Big(\alpha(\mathbf{P}_1\phi(\mathbf{X}^{(1)}) + \mathbf{P}_2\phi(\mathbf{X}^{(2)})) + \mathbf{R}\mathbf{D}\mathbf{L}\Big). \tag{30}$$

Then, we obtain the hash codes by mean-thresholding operation. The mAP results of DSAH and DSAH-Re on two datasets are shown in Table 4 and 5. From the table, we can observe that the performance of DSAH is better than that of DSAH-Re on two datasets. These results demonstrate that our proposed discrete optimization algorithm performs well in avoiding quantization errors and improving the performance of cross-modal retrieval tasks.

Table 4. mAP scores of different ablated versions of DSAH on MIRFlickr dataset.

Task	Method	8 Bits	16 Bits	32 Bits	64 Bits	128 Bits
I→T	DSAH-Re	0.5982	0.6076	0.6081	0.5911	0.5784
	DSAH-Ke	0.6832	0.6872	0.6898	0.6913	0.6906
	DSAH-Nm	0.6991	0.7212	0.7245	0.7289	0.7310
	DSAH	**0.7156**	**0.7236**	**0.7412**	**0.7498**	**0.7556**
T→I	DSAH-Re	0.5543	0.5521	0.5760	0.5773	0.5801
	DSAH-Ke	0.7362	0.7690	0.7772	0.7887	0.7921
	DSAH-Nm	0.7111	0.7304	0.7297	0.7358	0.7439
	DSAH	**0.7526**	**0.7782**	**0.8041**	**0.8163**	**0.8180**

Table 5. mAP scores of different ablated versions of DSAH on NUS-WIDE dataset.

Task	Method	8 Bits	16 Bits	32 Bits	64 Bits	128 Bits
I→T	DSAH-Re	0.4558	0.4611	0.4732	0.4798	0.4832
	DSAH-Ke	0.5842	0.5911	0.6287	0.6334	0.6499
	DSAH-Nm	0.6069	0.6093	0.6421	0.6429	0.6551
	DSAH	**0.6231**	**0.6432**	**0.6517**	**0.6667**	**0.6791**
T→I	DSAH-Re	0.4568	0.4650	0.4717	0.4823	0.4804
	DSAH-Ke	0.7287	0.7452	0.7676	0.7704	0.7799
	DSAH-Nm	0.7225	0.7407	0.7589	0.7621	0.7697
	DSAH	**0.7324**	**0.7596**	**0.7756**	**0.7834**	**0.8024**

4.5.2. Effects of Kernelization

In this paper, DSAH adopts a kernelization technique to handle the nonlinear relations between different heterogeneous modalities to improve the retrieval accuracy and efficiency. To demonstrate the effects of kernelization, we denote a variant of DSAH, named DSAH-ke, which directly uses the original features to learn the hash codes. We conduct experiments on two datasets with the code length varying from 8 bits to 128 bits to evaluate the performance of DSAH-ke. The mAP results of DSAH-ke are reported in Tables 4 and 5. From the tables, we can observe that the lack of using kernelization will reduce the retrieval performance.

4.5.3. Effects of $\ell_{2,1}$ Norm

As shown in Section 3.3, the $\ell_{p,q}$ norm, i.e., $\ell_{2,1}$, is used to improve the robustness for outliers. Therefore, in this section, to verify its effectiveness, we denote a variant of DSAH, named DSAH-Nm, which replaced the term $||\mathbf{R}^\top\mathbf{B} - \mathbf{L}||_{2,1}$ in Equation (8) with $||\mathbf{R}^\top\mathbf{B} - \mathbf{L}||_F^2$. The mAP results on two datasets with the code length varying from 8 bits to 128 bits are reported in Table 4 and 5. From the table, we can observe that the $\ell_{2,1}$ norm is effective to improve the performance of DSAH, the reason may be that the label information often inevitably contains some noises or subjective factors.

4.5.4. Effects of Word Embeddings

In order to verify the impact of different word embeddings on the performance of cross-modal retrieval. We denote a *Bidirectional Encoder Representations from Transformers* (BERT)-based [58] variant of DSAH, named DSAH-BERT. The BERT-based word embeddings are generated by summing the 786-D features from the last 4-hidden layers of a 12 layers BERT trained in an uncased way (https://github.com/huggingface/transformers, accessed on 13 July 2021). We conduct experiments on NUS-WIDE dataset to evaluate the effects of word embeddings. The mAP scores on NUS-WIDE dataset with the code length varying from 8 bits to 128 bits are reported in Table 6. From the table, we can find that BERT-based word embedding provides a slightly higher average mAP scores than those with bag-of-word embedding.

Table 6. mAP scores on NUS-WIDE dataset using BERT and bag-of-word embeddings of DSAH.

Task	Method	8 Bits	16 Bits	32 Bits	64 Bits	128 Bits
I→T	DSAH	**0.6231**	0.6432	0.6517	**0.6667**	**0.6791**
	DSAH-BERT	0.6212	**0.6491**	**0.6521**	0.6634	0.6747
T→I	DSAH	**0.7324**	**0.7596**	0.7756	0.7834	0.8024
	DSAH-BERT	0.7311	0.7558	**0.7801**	**0.7907**	**0.8071**

4.5.5. Effects of Deep Learning Based Representation

More recently, deep neural networks have achieved promising performance in the field of representation learning. To validate the effectiveness of DSAH, we conduct experiments on

NUS-WIDE dataset to evaluate the effects of deep learning based representations. The corresponding variant of DSAHis named DSAH-Deep. Specifically, each image is represented as a 4096-D vector extracted by the fc7-layer of VGG-16 net [59]. The mAP scores on NUS-WIDE dataset with the code length varying from 8 bits to 128 bits are shown in Table 7. From the table, we can observe that the use of deep learning based representation for cross-modal retrieval improves the accuracy of retrieving text through images, but reduces the accuracy of retrieving images through text. The reason may be that deep learning-based representation improves the semantics of the image representation, but the difficulty of retrieving images is increased due to the increase of the dimensionality simultaneously.

Table 7. mAP scores on NUS-WIDE dataset using deep and shallow representations of DSAH.

Task	Method	8 Bits	16 Bits	32 Bits	64 Bits	128 Bits
I→T	DSAH	0.6231	0.6432	0.6517	0.6667	0.6791
	DSAH-Deep	**0.7232**	**0.7421**	**0.7519**	**0.7630**	**0.7793**
T→I	DSAH	**0.7324**	**0.7596**	**0.7756**	**0.7834**	**0.8024**
	DSAH-Deep	0.6533	0.6982	0.7106	0.7218	0.7295

4.5.6. Effects of Parameters

In this section, we conduct parameter sensitivity analysis experiments to observe the variation of mAP scores under different α and γ. In this experiment, by prefixing the code length as 64 bits, we vary the parameters α and γ in the range of $\{10^{-5}, 10^{-4}, ..., 10^{4}, 10^{5}\}$. Figures 3 and 4 report the results. From these results, we observe that the performance of our proposed DSAH is relatively stable on a wide range of α and γ values. Specifically, on the MIRFlickr dataset, when $\alpha < 10^{0}$ and $\gamma < 10^{1}$, the retrieval performance becomes stable. On the NUS-WIDE dataset, when $\alpha < 10^{2}$ and $\gamma < 10^{1}$, the scores of mAP have a very small fluctuation. Therefore, our proposed method can be easily tuned for practical implementations.

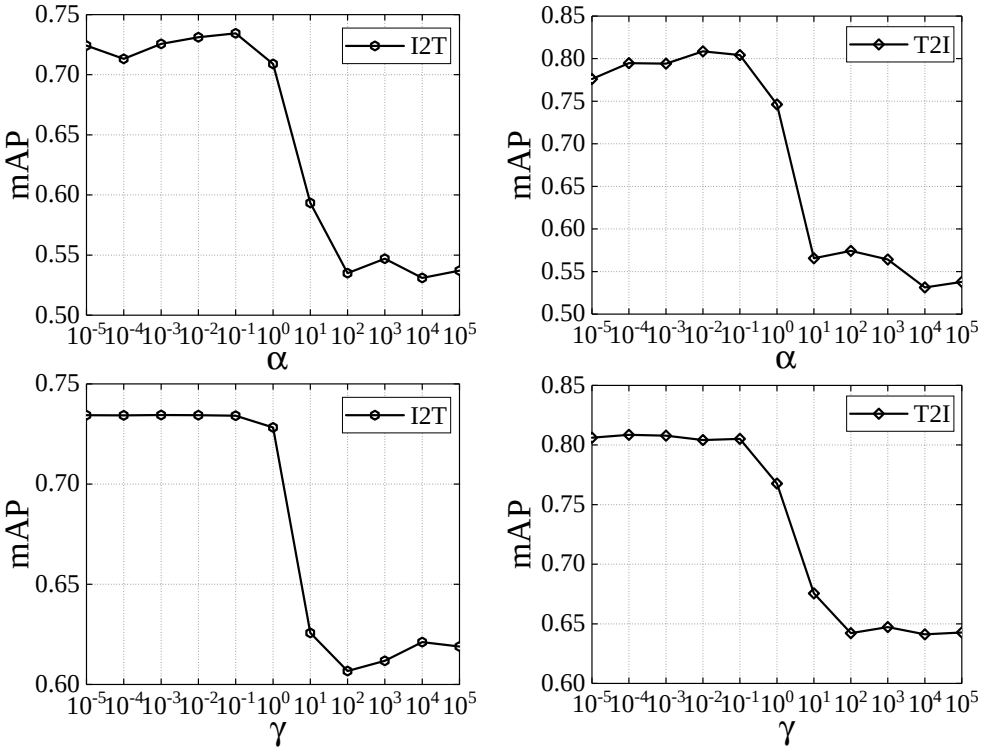

Figure 3. Parameter sensitivity analysis of α and γ on MIRFlickr dataset with 64 bits.

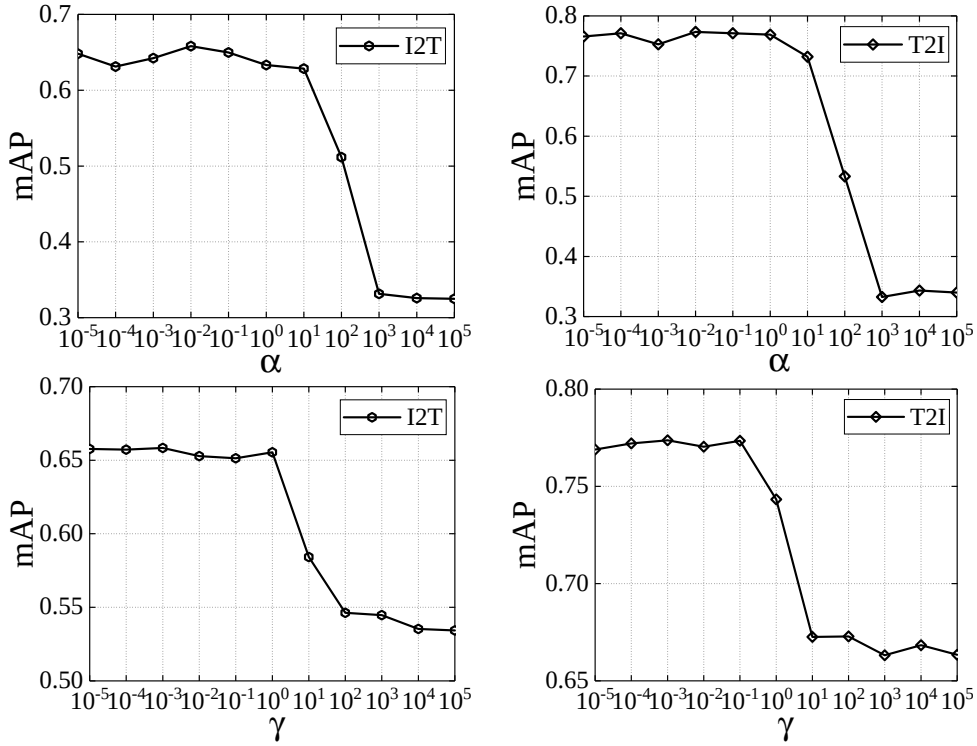

Figure 4. Parameter sensitivity analysis of α and γ on NUS-WIDE dataset with 64 bits.

4.5.7. Convergence Analysis

In order to show the convergence of our proposed DSAH, we conduct the experiments on two datasets with the codes length fixed as 64 bits. Similar results can be obtained on other lengths of hash codes. The results are shown in Figure 5. Note that, in order to visually represent the convergence of the objective function, the value can be normalized by dividing by the maximum value on each dataset. From the figure, we can easily see that the values of objective function can converge very fast, i.e., less than 12 iterations, which demonstrates the efficiently of the closed-form solutions of the optimization algorithm.

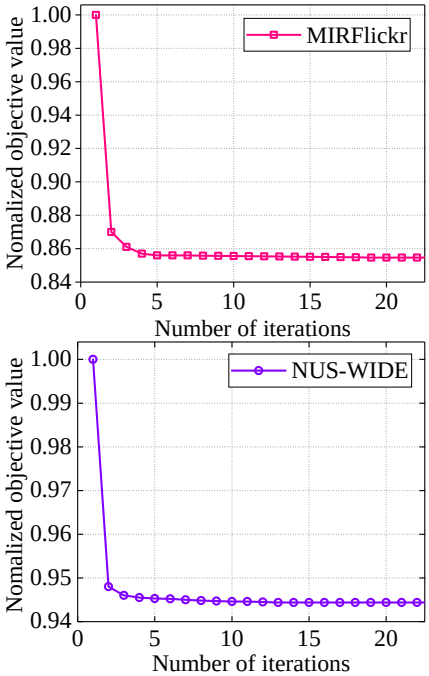

Figure 5. Convergence curves of DSAH on two datasets with 64 bits.

4.6. Limitations

The main potential limitation of our proposed DSAH is that the time complexity of constructing the pairwise similarity matrix is $\mathcal{O}(n^2)$. Although the method we proposed uses label matrices instead of a pairwise similarity matrix for matrix decomposition, it cannot effectively solve the large time complexity problem. Therefore, compared with the hash methods that only use label information for learning, the time cost of DSAH is slightly high. In addition, without point-to-point label information, there is no general algorithm to process similarity matrices on all datasets.

5. Conclusions

In this paper, we present a novel cross-modal hashing method, named DSAH, for large-scale cross-modal retrieval. In detail, to enhance the feature representation in the linear model, we handle the nonlinear relations with a kernelization technique. Meanwhile, DSAH incorporates the label information and semantic matrix into the learning process. Therefore, DSAH can obtain more semantic information to improve the discriminative capability of the learned hash codes. However, due to the inevitable noise and subjective factors in labels for large-scale dataset, the $\ell_{2,1}$ norm is used to sparse the matrix and effectively deal with outliers. In addition, a discrete optimization algorithm is proposed to solve the quantization errors and improve the optimization efficiency. Extensive experiments on two datasets demonstrate the superiority of DSAH on cross-modal retrieval tasks.

Author Contributions: Conceptualization, L.S.; methodology, Z.Y.; software, L.S.; validation, L.S., Y.Z. and J.L.; formal analysis, L.S., Y.Z. and J.L.; investigation, L.S.; resources, L.S. and J.L.; data curation, L.S.; writing—original draft preparation, L.S.; writing—review and editing, Z.Y. and L.H.; visualization, L.S.; supervision, Z.Y.; project administration, J.L.; funding acquisition, J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China under Grant U2003208, in part by the Science and Technology Plan of Hunan under Grant No. 2016TP1003, and in part by the Key Technology R & D Program of Hunan Province under Grant No. 2018GK2052.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Deng, C.; Chen, Z.; Liu, X.; Gao, X.; Tao, D. Triplet-Based Deep Hashing Network for Cross-Modal Retrieval. *IEEE Trans. Image Process.* **2018**, *27*, 3893–3903.
2. Li, C.; Deng, C.; Li, N.; Liu, W.; Gao, X.; Tao, D. Self-Supervised Adversarial Hashing Networks for Cross-Modal Retrieval. In Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, 18–22 June 2018; pp. 4242–4251.
3. Yang, E.; Deng, C.; Liu, W.; Liu, X.; Tao, D.; Gao, X. Pairwise Relationship Guided Deep Hashing for Cross-Modal Retrieval. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; pp. 1618–1625.
4. Wang, J.; Zhang, T.; Song, J.; Sebe, N.; Shen, H.T. A Survey on Learning to Hash. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 769–790.
5. Wang, G.; Hu, Q.; Cheng, J.; Hou, Z. Semi-supervised Generative Adversarial Hashing for Image Retrieval. In Proceedings of the Computer Vision—ECCV 2018—15th European Conference, Munich, Germany, 8–14 September 2018; pp. 491–507.
6. Xia, R.; Pan, Y.; Lai, H.; Liu, C.; Yan, S. Supervised Hashing for Image Retrieval via Image Representation Learning. In Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, Québec City, QC, Canada, 27–31 July 2014; pp. 2156–2162.
7. Li, D.; Gong, Y.; Cheng, D.; Shi, W.; Tao, X.; Chang, X. Consistency-Preserving deep hashing for fast person re-identification. *Pattern Recognit.* **2019**, *94*, 207–217.
8. Li, M.; Jiang, Q.; Li, W. Deep Multi-Index Hashing for Person Re-Identification. *arXiv preprint* **2019**, arXiv:abs/1905.10980.
9. Deng, C.; Yang, E.; Liu, T.; Li, J.; Liu, W.; Tao, D. Unsupervised Semantic-Preserving Adversarial Hashing for Image Search. *IEEE Trans. Image Process.* **2019**, *28*, 4032–4044.

10. Song, J.; Yang, Y.; Yang, Y.; Huang, Z.; Shen, H.T. Inter-media hashing for large-scale retrieval from heterogeneous data sources. In Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data, New York, NY, USA, 22–27 June 2013; pp. 785–796.
11. Zhou, J.; Ding, G.; Guo, Y. Latent semantic sparse hashing for cross-modal similarity search. In Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval, 2014; pp. 415–424.
12. He, K.; Wen, F.; Sun, J. K-Means Hashing: An Affinity-Preserving Quantization Method for Learning Binary Compact Codes. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 2938–2945.
13. Shen, F.; Xu, Y.; Liu, L.; Yang, Y.; Huang, Z.; Shen, H.T. Unsupervised Deep Hashing with Similarity-Adaptive and Discrete Optimization. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 3034–3044.
14. Weiss, Y.; Torralba, A.; Fergus, R. Spectral Hashing. In Proceedings of the Advances in Neural Information Processing Systems 21, Proceedings of the Twenty-Second Annual Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 8–11 December 2008; pp. 1753–1760.
15. Zhang, H.; Liu, L.; Long, Y.; Shao, L. Unsupervised Deep Hashing With Pseudo Labels for Scalable Image Retrieval. *IEEE Trans. Image Process.* **2018**, *27*, 1626–1638.
16. Mandal, D.; Chaudhury, K.N.; Biswas, S. Generalized Semantic Preserving Hashing for Cross-Modal Retrieval. *TIP* **2019**, *28*, 102–112.
17. Bronstein, M.M.; Bronstein, A.M.; Michel, F.; Paragios, N. Data fusion through cross-modality metric learning using similarity-sensitive hashing. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010; pp. 3594–3601.
18. Liu, X.; Nie, X.; Zeng, W.; Cui, C.; Zhu, L.; Yin, Y. Fast Discrete Cross-modal Hashing with Regressing from Semantic Labels. In Proceedings of the 26th ACM international conference on Multimedia, 2018; pp. 1662–1669.
19. Shen, F.; Shen, C.; Liu, W.; Shen, H.T. Supervised Discrete Hashing. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, 7–12 June 2015; pp. 37–45.
20. Luo, X.; Zhang, P.; Wu, Y.; Chen, Z.; Huang, H.; Xu, X. Asymmetric Discrete Cross-Modal Hashing. In Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval, 2018; pp. 204–212.
21. Liu, H.; Wang, R.; Shan, S.; Chen, X. Deep Supervised Hashing for Fast Image Retrieval. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, 27–30 June 2016; pp. 2064–2072.
22. Yang, Z.; Yang, L.; Huang, W.; Sun, L.; Long, J. Enhanced Deep Discrete Hashing with semantic-visual similarity for image retrieval. *Inf. Process. Manag.* **2021**, *58*, 102648.
23. Yang, R.; Shi, Y.; Xu, X. Discrete Multi-view Hashing for Effective Image Retrieval. In *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval, ICMR 2017, Bucharest, Romania, 6–9 June 2017*; Ionescu, B., Sebe, N., Feng, J., Larson, M.A., Lienhart, R., Snoek, C., Eds.; ACM: New York, NY, USA, 2017; pp. 175–183.
24. Wang, J.; Liu, W.; Kumar, S.; Chang, S. Learning to Hash for Indexing Big Data—A Survey. *Proc. IEEE* **2016**, *104*, 34–57.
25. Andoni, A.; Indyk, P. Near-Optimal Hashing Algorithms for Approximate Nearest Neighbor in High Dimensions. In Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2006), Berkeley, CA, USA, 21–24 October 2006; pp. 459–468.
26. Datar, M.; Immorlica, N.; Indyk, P.; Mirrokni, V.S. Locality-sensitive hashing scheme based on p-stable distributions. In Proceedings of the 20th ACM Symposium on Computational Geometry, Brooklyn, New York, NY, USA, 8–11 June 2004; pp. 253–262.
27. Huang, Q.; Feng, J.; Fang, Q.; Ng, W.; Wang, W. Query-aware locality-sensitive hashing scheme for l_p norm. *VLDB J.* **2017**, *26*, 683–708.
28. Gong, Y.; Lazebnik, S.; Gordo, A.; Perronnin, F. Iterative Quantization: A Procrustean Approach to Learning Binary Codes for Large-Scale Image Retrieval. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 2916–2929.
29. Jin, S.; Yao, H.; Sun, X.; Zhou, S. Unsupervised semantic deep hashing. *Neurocomputing* **2019**, *351*, 19–25.
30. Wu, G.; Han, J.; Guo, Y.; Liu, L.; Ding, G.; Ni, Q.; Shao, L. Unsupervised Deep Video Hashing via Balanced Code for Large-Scale Video Retrieval. *IEEE Trans. Image Process.* **2019**, *28*, 1993–2007.
31. Tang, J.; Li, Z.; Wang, M.; Zhao, R. Neighborhood Discriminant Hashing for Large-Scale Image Retrieval. *IEEE Trans. Image Process.* **2015**, *24*, 2827–2840.
32. Ding, G.; Guo, Y.; Zhou, J.; Gao, Y. Large-Scale Cross-Modality Search via Collective Matrix Factorization Hashing. *TIP* **2016**, *25*, 5427–5440.
33. Liu, H.; Ji, R.; Wu, Y.; Huang, F.; Zhang, B. Cross-Modality Binary Code Learning via Fusion Similarity Hashing. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017, pp. 6345–6353.
34. Zhang, D.; Li, W. Large-Scale Supervised Multimodal Hashing with Semantic Correlation Maximization. In Proceedings of the AAAI Conference on Artificial Intelligence, 2014; pp. 2177–2183.
35. Lin, Z.; Ding, G.; Hu, M.; Wang, J. Semantics-preserving hashing for cross-view retrieval. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3864–3872.
36. Xu, X.; Shen, F.; Yang, Y.; Shen, H.T.; Li, X. Learning Discriminative Binary Codes for Large-scale Cross-modal Retrieval. *TIP* **2017**, *26*, 2494–2507.

37. Wang, D.; Gao, X.; Wang, X.; He, L. Label Consistent Matrix Factorization Hashing for Large-Scale Cross-Modal Similarity Search. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *41*, 2466–2479.
38. Chen, Z.; Li, C.; Luo, X.; Nie, L.; Zhang, W.; Xu, X. SCRATCH: A Scalable Discrete Matrix Factorization Hashing Framework for Cross-Modal Retrieval. *IEEE Trans. Circuits Syst. Video Technol.* **2020**, *30*, 2262–2275.
39. Yang, Z.; Long, J.; Zhu, L.; Huang, W. Nonlinear Robust Discrete Hashing for Cross-Modal Retrieval. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, 25–30 July 2020*; Huang, J., Chang, Y., Cheng, X., Kamps, J., Murdock, V., Wen, J., Liu, Y., Eds.; ACM: New York, NY, USA, 2020; pp. 1349–1358.
40. Yang, Z.; Raymond, O.I.; Huang, W.; Liao, Z.; Zhu, L.; Long, J. Scalable deep asymmetric hashing via unequal-dimensional embeddings for image similarity search. *Neurocomputing* **2020**, *412*, 262–275.
41. Yang, Z.; Yang, L.; Raymond, O.I.; Zhu, L.; Huang, W.; Liao, Z.; Long, J. NSDH: A Nonlinear Supervised Discrete Hashing framework for large-scale cross-modal retrieval. *Knowl. Based Syst.* **2021**, *217*, 106818.
42. Shen, H.T.; Liu, L.; Yang, Y.; Xu, X.; Huang, Z.; Shen, F.; Hong, R. Exploiting subspace relation in semantic labels for cross-modal hashing. *IEEE Trans. Knowl. Data Eng.* **2020**, *10*, 3351–3365. doi:10.1109/TKDE.2020.2970050.
43. Jiang, Q.; Li, W. Deep Cross-Modal Hashing. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, 21–26 July 2017; pp. 3270–3278.
44. Yang, D.; Wu, D.; Zhang, W.; Zhang, H.; Li, B.; Wang, W. Deep Semantic-Alignment Hashing for Unsupervised Cross-Modal Retrieval. In *Proceedings of the 2020 on International Conference on Multimedia Retrieval, ICMR 2020, Dublin, Ireland, 8–11 June 2020*; Gurrin, C., Jónsson, B.Þ., Kando, N., Schöffmann, K., Chen, Y.P., O'Connor, N.E., Eds.; ACM: New York, NY, USA, 2020; pp. 44–52.
45. Wang, T.; Zhu, L.; Cheng, Z.; Li, J.; Gao, Z. Unsupervised Deep Cross-modal Hashing with Virtual Label Regression. *Neurocomputing* **2020**, *386*, 84–96.
46. Jin, S.; Yao, H.; Sun, X.; Zhou, S.; Zhang, L.; Hua, X. Deep Saliency Hashing for Fine-Grained Retrieval. *IEEE Trans. Image Process.* **2020**, *29*, 5336–5351.
47. Zhan, Y.; Luo, X.; Wang, Y.; Xu, X. Supervised Hierarchical Deep Hashing for Cross-Modal Retrieval. In *Proceedings of the MM '20: The 28th ACM International Conference on Multimedia, Virtual Event/Seattle, WA, USA, 12–16 October 2020*; Chen, C.W., Cucchiara, R., Hua, X., Qi, G., Ricci, E., Zhang, Z., Zimmermann, R., Eds.; ACM: New York, NY, USA, 2020; pp. 3386–3394.
48. Zhang, M.; Li, J.; Zhang, H.; Liu, L. Deep semantic cross modal hashing with correlation alignment. *Neurocomputing* **2020**, *381*, 240–251.
49. Kulis, B.; Darrell, T. Learning to Hash with Binary Reconstructive Embeddings. In Proceedings of the Workshop on Learning from Multiple Sources with Applications to Robotics (NIPS), Whistler, BC, Canada, 11 December 2009; pp. 1042–1050.
50. Kulis, B.; Grauman, K. Kernelized locality-sensitive hashing for scalable image search. In Proceedings of the IEEE 12th International Conference on Computer Vision, ICCV 2009, Kyoto, Japan, 27 September–4 October 2009; 2009; pp. 2130–2137.
51. Liu, W.; Wang, J.; Ji, R.; Jiang, Y.; Chang, S. Supervised hashing with kernels. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 2074–2081.
52. Gui, J.; Li, P. R 2 SDH: Robust Rotated Supervised Discrete Hashing. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, 19–23 August 2018; Guo, Y., Farooq, F., Eds.; ACM: New York, NY, USA, 2018; pp. 1485–1493.
53. Cheng, M.; Jing, L.; Ng, M.K. Robust Unsupervised Cross-modal Hashing for Multimedia Retrieval. *ACM Trans. Inf. Syst.* **2020**, *38*, 1–25.
54. Jiang, Q.; Li, W. Asymmetric Deep Supervised Hashing. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, LA, USA, 2–7 February 2018; pp. 3342–3349.
55. Zhang, Z.; Lai, Z.; Huang, Z.; Wong, W.K.; Xie, G.; Liu, L.; Shao, L. Scalable Supervised Asymmetric Hashing With Semantic and Latent Factor Embedding. *IEEE Trans. Image Process.* **2019**, *28*, 4803–4818.
56. Huiskes, M.J.; Lew, M.S. The MIR flickr retrieval evaluation. In Proceedings of the 1st ACM International Conference on Multimedia Information Retrieval, 2008; pp. 39–43.
57. Chua, T.; Tang, J.; Hong, R.; Li, H.; Luo, Z.; Zheng, Y. NUS-WIDE: a real-world web image database from National University of Singapore. In Proceedings of the 8th ACM International Conference on Image and Video Retrieval, CIVR 2009, Santorini Island, Greece, 8–10 July 2009.
58. Devlin, J.; Chang, M.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, 2–7 June 2019; 2019; Volume 1 (Long and Short Papers), pp. 4171–4186.
59. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, 7–9 May 2015.