Cross-Modal Hashing Method With Properties of Hamming Space: A New Perspective

Zhikai Hu, Yiu-Ming Cheung, *Fellow, IEEE*, Mengke Li, and Weichao Lan

***Abstract*—Cross-modal hashing (CMH) has attracted considerable attention in recent years. Almost all existing CMH methods primarily focus on reducing the modality gap and semantic gap, i.e., aligning multi-modal features and their semantics in Hamming space, without taking into account the space gap, i.e., difference between the real number space and the Hamming space. In fact, the space gap can affect the performance of CMH methods. In this paper, we analyze and demonstrate how the space gap affects the existing CMH methods, which therefore raises two problems: solution space compression and loss function oscillation. These two problems eventually cause the retrieval performance deteriorating. Based on these findings, we propose a novel algorithm, namely Semantic Channel Hashing (SCH). First, we classify sample pairs into fully semantic-similar, partially semantic-similar, and semantic-negative ones based on their similarity and impose different constraints on them, respectively, to ensure that the entire Hamming space is utilized. Then, we introduce a semantic channel to alleviate the issue of loss function oscillation. Experimental results on three public datasets demonstrate that SCH outperforms the state-of-the-art methods. Furthermore, experimental validations are provided to substantiate the conjectures regarding solution space compression and loss function oscillation, offering visual evidence of their impact on the CMH methods.**

***Index Terms*—Cross-modal retrieval, hashing, Hamming space, loss oscillation, solution space compression.**

I. Introduction

CROSS-MODAL retrieval [1], [2], [3], which seeks to retrieve information across different modalities, has received increasing attention in the literature. Existing methods primarily concentrate on addressing modality gap [4] and semantic gap [5] in this task, i.e., achieving the alignment of multi-modal features and their semantics. Consequently, these approaches typically map multi-modal data into a common semantic space, with the anticipation that samples possessing similar semantics cluster in this space. Among them, real number-based cross-modal (RCM) methods are a popular branch that employs a real number space as this semantic space. Although these methods [6], [7], [8] have achieved promising results in their application domains, storing multi-modal data with real numbers is inefficient and cumbersome for storage and retrieval.

To achieve fast retrieval with low storage, hash techniques [9], [10], [11] have been developed, forming a new promising branch of cross-modal hashing (CMH) methods. In CMH methods, the common semantic space is defined as a Hamming space, where all samples are stored as discrete hash codes, and their similarity can be easily calculated by the XOR operation, reducing storage costs and improving retrieval efficiency. Owing to these advantages, CMH methods have gained popularity and emerged as the preferred choice for cross-modal retrieval task [12]. However, since multi-modal data are initially stored in the real number space but later mapped to the Hamming space, an additional space gap, namely the difference between these two spaces, needs to be bridged. Regretfully, almost all existing CMH methods focus on bridging the aforementioned modality gap and semantic gap but ignore the space gap. Moreover, they usually simplify the space gap into discrete attribute of hash codes and have proposed corresponding solutions such as discretely updating hash codes [13], [14], [15], [16] and minimizing quantization error [17], [18]. However, they overlook the key differences between these two spaces. Specifically, different from the real number space, Hamming space holds three special properties. *Property 1*: *The number of points is finite in Hamming space. Property 2*: *The similarity values between points in Hamming space are discrete and finite.* Let us take the commonly used cosine similarity as an example. The set of values of the cosine similarity between two points in a k-dimensional Hamming space is a discrete finite set, denoted as $\mathcal{C} = \{1 - 2d_H/k | d_H = 0, 1, 2, \ldots, k\}$, where d_H is the Hamming distance between the two points. The size of $\mathcal{C}$ is $|\mathcal{C}| = k + 1$. *Property 3*: *For any anchor in Hamming space, the Hamming distances between it and all points in the space follow the Binomial distribution* $\mathcal{B}(k, 1/2)$. As a result, such overlook limits their performance on cross-modal retrieval.

In this paper, we will analyze the impact of these three properties on the CMH methods. We focus primarily on the supervised learning paradigm as labels can often indicate more precise semantic information. As previously mentioned, the existing

Manuscript received 14 June 2023; revised 9 April 2024; accepted 21 April 2024. Date of publication 23 April 2024; date of current version 5 November 2024. This work was supported in part by the NSFC/Research Grants Council (RGC) Joint Research Scheme under Grant N_HKBU214/21, in part by the General Research Fund of RGC under Grant 12201321, Grant 12202622 and Grant 12201323, in part by the RGC Senior Research Fellow Scheme under Grant SRFS2324-2S02, in part by the National Natural Science Foundation of China (NSFC) under Grant 62306181, and in part by the Natural Science Foundation of Guangdong Province under Grant 2024A1515010163. Recommended for acceptance by N. Sebe. *(Corresponding author: Yiu-Ming Cheung.)*

Zhikai Hu, Yiu-Ming Cheung, and Weichao Lan are with the Department of Computer Science, Hong Kong Baptist University, Hong Kong, China (e-mail: cszkhu@comp.hkbu.edu.hk; ymc@comp.hkbu.edu.hk; cswclan@comp.hkbu.edu.hk).

Mengke Li is with the Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ) and Visual Computing Research Center, Shenzhen University, Shenzhen 518060, China (e-mail: limengke@gml.ac.cn).

Codes are available at https://github.com/hutt94/SCH.

Digital Object Identifier 10.1109/TPAMI.2024.3392763

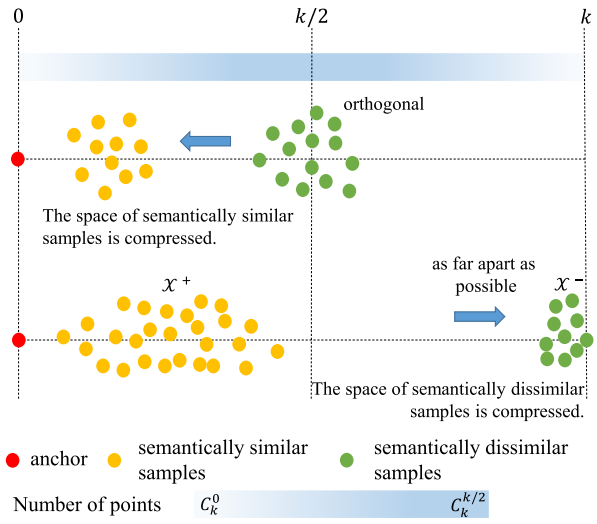


Fig. 1. There are two commonly used constraints in CMH methods to distribute semantically dissimilar samples, i.e., keeping them orthogonal or apart as far as possible. However, the special properties of Hamming spaces make these two constraints problematic in CMH methods, as they can lead to compression of the solution space.

CMH methods primarily address the semantic gap by gathering semantically similar samples and separating semantically dissimilar ones. It turns out that more similar samples can be retrieved from the retrieval set. Generally, there are two common strategies for separating semantically dissimilar samples: making them orthogonal [19], [20] or keeping them apart as far as possible [21], [22], [23]. Unfortunately, both of them will raise the problem of solution space compression, i.e., samples in the retrieval set are forced to be distributed in a confined space. For example, orthogonal constraint, as proposed in [24], [25], aims at enforcing semantically dissimilar samples to have orthogonal hash codes. That is, their corresponding hash codes should have a Hamming distance of $k/2$ in a k-dimensional Hamming space. Conversely, samples with similar semantics should have hash codes that are as close as possible, preferably less than $k/2$ away from each other. In this way, given any sample x in the training set $\mathcal{T}$ as an anchor, the hash codes of the rest samples in set $\mathcal{T}/x$ are expected to be distributed in the Hamming space within $k/2$ of its corresponding hash code, which we denote as $\mathcal{S}(x)$. Unfortunately, nearly half of the Hamming space is not utilized in $\mathcal{S}(x)$. Finally, the overall solution space for the training set $\mathcal{T}$ is the intersection of solution spaces of all samples, i.e., $\cap_{x\in\mathcal{T}}\mathcal{S}(x)$. As pointed out by Property 1, the number of points in the Hamming space is finite, and as a result, the more training samples in $\mathcal{T}$, the more compressed the overall solution space is, which may result in insufficient space for separating semantically dissimilar samples. Additionally, the space of semantically similar samples is also compressed, as illustrated in Fig. 1.

Another commonly used constraint in CMH methods is to keep semantically dissimilar samples apart as far as possible [12], [26], [27], which, however, can also lead to a compressed solution space. As pointed out by Property 3, the Hamming distances between two hash codes follow the Binomial distribution $\mathcal{B}(k, 1/2)$, implying that the corresponding space has fewer points as the distance from an anchor increases when $d_H > k/2$. To achieve this constraint, all semantically dissimilar samples of a given anchor x, denoted as the set $\mathcal{X}^-$, are expected to be as far away as possible from x. However, it will force the samples in $\mathcal{X}^-$ to be compressed into a small space, which ultimately prevent the separation of semantically dissimilar samples within the set $\mathcal{X}^-$. Although in practical training procedure, semantically similar samples set $\mathcal{X}^+$ and semantically dissimilar samples set $\mathcal{X}^-$ can balance each other, this balance only works when the number of classes is small and training samples of each class are balanced. As the number of classes and samples increases, this balance may fail, and the side with more samples usually prevails, finally compresses the space of its opposite side, as shown in Fig. 1. Overall, excessively stringent constraints imposed on semantically dissimilar samples will result in solution space compression. More detailed analyses of this issue will be provided in Section IV-C-1.

In addition to the problem of solution space compression, the existing CMH methods face another issue of loss function oscillation, wherein the gradient at the optimal solution is non-zero, preventing the loss function from consistently decreasing. Specifically, in the prevailing supervised CMH paradigm, the similarity s between the labels of a sample pair is commonly employed as the supervisory information to guide the learning of their corresponding hash codes, whose similarity is denoted as c. To accomplish this, we usually define a loss function $f(s, c)$, e.g., $f(s, c) = (s - c)^2$ [24], [28], [29], and learn hash functions and hash codes by minimizing it, i.e., $\min_c f(s, c)$. Notably, since f is often continuous while the similarity values of hash codes are discrete, as pointed out by Property 2, this setup can lead to the problem of loss function oscillation. Specifically, once the $\operatorname{argmin}_c f(s, c) \notin \mathcal{C}$, we can only select a value $c' \in \mathcal{C}$ that is the closest to s to obtain the optimal solution. Obviously, the corresponding gradient is not zero at this point, i.e., $\nabla f(s, c') \neq 0$ (see Section III-C-2). That is, the gradient of the optimal solution can be non-zero. This implies that even if the optimal solution has been found, the corresponding sample pairs will continue to contribute to the gradient and may jump out of the optimal solution in the next epoch, leading to the problem of loss function oscillation.

To address the aforementioned two issues, this paper proposes a novel CMH method, namely Semantic Channel Hashing (SCH). Specifically, to avoid the problem of solution space compression, we classify sample pairs into three categories based on the similarity calculated by their labels: fully semantic-positive samples, partially semantic-positive samples, and semantic-negative samples. Different constraints are then imposed on their corresponding hash codes, ensuring that the entire Hamming space is utilized effectively. Specifically, in Hamming space, the fully semantic-positive samples are expected to be as close as possible to each other, and the partially semantic-positive samples are distributed in a relatively close range in an orderly manner, while the distances between semantic-negative samples need to be farther than $k/2$. Moreover, to alleviate the oscillation of the loss function, we extend the Hamming distances assigned

to different samples in the first step into semantic channels, ensuring that the gradient of the loss function is zero at the optimal solution. In summary, the key contributions of this paper are as follows:

- We present an analysis of three characteristics of space gap in CMH task and explain how they hinder existing CMH methods. Subsequently, it is found that current CMH methods suffer from the problems of solution space compression and loss function oscillation. This insight would inspire future researchers in the design of CMH algorithms.
- To address the identified problems, we propose a novel algorithm SCH. By handling sample pairs with different similarities differently, we enable the full utilization of the Hamming space. Moreover, the introduction of the semantic channel helps alleviate the problem of loss function oscillation.
- Experimental results on three public datasets demonstrate the superior retrieval performance of the proposed SCH. Furthermore, we have designed experiments to validate our analyses of solution space compression and loss function oscillation. By comparing SCH with two commonly used loss functions, we provide a better understanding of the motivation and effectiveness of SCH.

The remainder of this paper is organized as follows. In Section II, we make an overview of cross-modal retrieval methods and efforts to cope with discrete attribute of hash codes in recent years. Section III details the proposed algorithm, and Section IV presents the experimental results and analyses. Finally, we draw a conclusion in Section V.

II. Related Work

A. Real Number-Based Cross-Modal Methods

Real number-based cross-modal methods aim to project multi-modal data into a common real number space. One of the most classical approaches is Canonical Correlation Analysis (CCA) [30], which learns a common representation of multi-modal data by maximizing the pairwise correlation between them. Many extensions, such as RCCA [31] and ml-CCA [32], have been subsequently proposed. Another typical method is Semantic Correlation Match (SCM) [5], which proposes three hypotheses: correlation matching for unsupervised methods, semantic matching for supervised ones, and their combination, semantic correlation matching, laying the foundational framework for subsequent cross-modal retrieval methods. Recently, deep neural networks have demonstrated exceptional abilities in various tasks [33], [34] and have been introduced into cross-modal retrieval tasks. For example, Feng et al. [6] and Hu et al. [35] used two auto-encoders to extract common features of multi-modal data, and Wei et al. [36] proposed a new baseline for cross-modal retrieval based on convolutional neural networks (CNN) visual features. Inspired by generative adversarial nets (GAN) [37], He et al. [38] and Wang et al. [7] employed adversarial learning to minimize the modality gap. Although they have achieved satisfactory performance, they still suffer from huge storage costs and low retrieval speed.

B. Cross-Modal Hashing Methods

Cross-modal hashing methods can be roughly divided into unsupervised [39], [40], [41] and supervised ones [42], [43], [44], [45]. Unsupervised methods typically leverage paired data to learn the correlations between modalities. For instance, Ding et al. [46] proposed Collective Matrix Factorization Hashing (CMFH) to learn unified hash codes from the original multi-modal data features. Su et al. [47] and Liu et al. [48] proposed different algorithms for constructing similarity matrices to guide hash code learning. Other approaches enhance CMH methods by drawing inspiration from the application of deep learning in other domains. For instance, Hu et al. [49] proposed a new method to automatically learn useful information from unsupervised CMH methods through knowledge distillation to supervise the learning of hash codes. Hu et al. [50] and Zhang et al. [51] respectively integrated the contrastive learning and graph neural network into the CMH methods, which have achieved promising results.

Supervised methods usually achieve better performance due to the utilization of label information. The mainstream approach, e.g., [19], [22], [23], [52], [53], is to construct a similarity matrix by using labels and preserve these similarity relations in Hamming space. Traditional matrix factorization-based CMH methods [52], [54], [55], [56] often employ orthogonal constraints to preserve relationships in the similarity matrix, aiming to ensure that semantically dissimilar samples are orthogonal. Building upon this foundation, various methods have been proposed. For example, Mandal et al. [24] used matrix factorization to obtain hash codes that can be adaptable to a wide range of scenarios. Liu et al. [20] proposed to use matrix tri-factorization decomposition to learn hash codes of varying lengths for data of different modalities. Wang et al. [57] were the first to propose replacing traditional dense hash codes with high-dimensional sparse hash codes. In contrast, deep hashing methods [12], [21], [22], [23] often adopt strategies that aim to keep semantically dissimilar samples as far as possible. In addition to this aim, they introduced other modules to facilitate the learning of hash codes. For example, Jiang et al. [21] were the pioneers in employing a deep model for CMH and devised a scheme for discretely updating hash codes. Inspired by GAN and self-supervised learning, Li et al. [22] proposed a self-supervised framework to learn hash codes and minimize the modality gap by adversarial learning. Zhang et al. [23] utilized two intra-modal and inter-modal asymmetric networks to further align the semantics of different modalities. Furthermore, inspired by metric learning, some methods combine triplet loss [58], [59] with hashing techniques. By leveraging the similarity relationships within triplets, these approaches, e.g., [60], [61], aim to enlarge the distance between semantically similar and dissimilar samples, yielding promising results.

More recently, several new deep learning models, such as Transformer [62], Bidirectional Encoder Representations from Transformers (BERT) [63], and Vision Transformer (ViT) [64], have gained popularity in the multi-modal domain. These models have also been adopted by some CMH methods, e.g., [26], [27], to extract highly expressive image and text features, which subsequently enhances their retrieval performance.

C. Bridging the Space Gap

Existing methods usually simplify the space gap to the discrete attribute of the hash code. As hash codes are discrete, optimizing loss functions is usually challenging. Some methods, e.g., [46], [65], attempt to simplify the discrete constraint into continuous constraint, which inevitably results in quantization errors and negatively impacts retrieval performance. To address this issue, Gong et al. [17] proposed a method that learns a rotation of zero-centered data to minimize quantization errors. Specifically, an orthogonal matrix is learned to rotate the hash code, which has been used in some subsequent work [16], [66]. In contrast, Shen et al. [13] retained the discrete constraints and proposed a discrete cyclic coordinate (DCC) descent method that can update hash codes in a discrete manner row by row. Although this method has achieved good performance in the CMH methods [15], [20], it is often time-consuming due to the row-by-row update. To this end, Gui et al. [14] proposed a closed-form solution of hash codes by reformulating the regression loss. This approximation solution only requires one operation to update the entire hash code matrix in a discrete manner, significantly improving training efficiency. This optimization method has been widely adopted in subsequent traditional CMH methods [16], [53], [57], [67] and its efficiency has been extensively verified. Due to the requirement of function continuity for automatic differentiation in deep learning, it is challenging to apply the discrete update strategy to the deep learning-based CMH methods. To address this, Cao et al. [68] proposed to use $\tanh(\beta\cdot)$ as activation function in the last hash layer, which can gradually approximate the sign function as the value of β increases. It aims to reduce the quantization error while maintaining function continuity for automatic differentiation.

Some works [69], [70] also mention the issue of solution space compression that differs from the definition in this paper. Specifically, they aim to narrow the solution space of *parameters* in the models to make the models converge to the optimal solution more rapidly. Besides, loss oscillation can arise from various factors, such as mini-batch training [71], non-differentiable regularization terms [72], and noisy labels [73]. Momentum [71] is designed to dampen loss oscillation caused by mini-batch training, and its mechanism has been extensively discussed [74], [75]. Wu et al. [72] proposed a smoothing regularization to alleviate oscillation. In this paper, we focus more on loss oscillation caused by the discreteness of hash codes.

III. Proposed Method

A. Notation and Problem Definition

In this paper, we focus on the cross-modal retrieval task of image and text modalities. Specifically, the dataset consists of n image-text pairs $\{x_i^I, x_i^T\}_{i=1}^n$, where $x_i^I \in \mathbb{R}^{d_1}$ and $x_i^T \in \mathbb{R}^{d_2}$ represent the i-th image and text samples, respectively. The original image and text data are denoted as $\mathbf{X}^I = \{x_i^I\}_{i=1}^n$ and $\mathbf{X}^T = \{x_i^T\}_{i=1}^n$, respectively. Their corresponding labels are denoted as $\mathbf{L} = \{l_i\}_{i=1}^n$, and each label $l_i \in \{0,1\}^c$, where c is the number of classes. In the training stage, our goal is to learn two hash functions $\mathbf{f}^I$ and $\mathbf{f}^T$ that project the image and text data into k-bit hash codes $\mathbf{B}^I = \{b_i^I\}_{i=1}^n$ and $\mathbf{B}^T = \{b_i^T\}_{i=1}^n$, respectively, where $b_i^{I,T} = \mathbf{f}^{I,T}(x_i^{I,T}) \in \{-1,1\}^k$. Afterward, the whole multi-modal dataset will be projected into the Hamming space to form a retrieval set. In the test stage, given a query q, it is first mapped into its corresponding hash code by $\mathbf{f}^{I,T}$ and then, the Hamming distance is utilized to search for the most relevant hash codes in the retrieval set, which are then returned as retrieval results.

B. Overview

For clear illustration, in Section III-B and III-C, we will use x_i and b_i to represent the i-th sample and its hash code of both modalities, respectively, i.e., x_i^I and x_i^T, and b_i^I and b_i^T.

The core idea of SCH involves the allocation of an appropriate Hamming distance to the hash codes b_i and b_j of samples x_i and x_j in the training stage, based on their similarity $\mathbf{S}_{ij}$, as illustrated in Fig. 2. In this paper, this Hamming distance is referred to as λ_{ij}. The cosine distance of their labels is initially utilized to establish the similarity $\mathbf{S}_{ij}$. Using $\mathbf{S}_{ij}$, we then assign a semantic channel, comprising of an appropriate Hamming distance λ_{ij} and a channel width τ, to samples x_i and x_j. Subsequently, this semantic channel is utilized to steer the learning of their corresponding hash codes b_i and b_j as well as hash functions $\mathbf{f}^{I,T}$.

C. Semantic Channel Hashing

1) Appropriate Hamming Distance: First, we estimate the relationships between samples by utilizing the cosine similarity of labels, i.e., $\mathbf{S}_{ij} = \cos(l_i, l_j) \in [0,1]$. Based on different $\mathbf{S}_{ij}$, the relationships between samples (x_i, x_j) can be grouped into three categories:

1) $\mathbf{S}_{ij} = 0$ if the labels of samples x_i and x_j are totally different, we term them semantic-negative samples and denote them as (x_i, x_j^-);
2) $\mathbf{S}_{ij} \in (0,1)$ if samples x_i and x_j share some of their labels, we term them partially semantic-positive samples and denote them as (x_i, x_j^+);
3) $\mathbf{S}_{ij} = 1$ if the labels of samples x_i and x_j are the same, we term them fully semantic-positive samples and denote them as $(x_i, x_j^\oplus)$.

It can be seen that, while the similarity $\mathbf{S}$ is informative enough to capture the nuances of semantic-positive samples, it fails to differentiate among semantic-negative samples where $\mathbf{S}_{ij} = 0$. However, it is important to acknowledge that the semantic-negative samples set $\mathcal{X}^- = \{x_j^-\}$ associated with any sample x_i also encompasses different semantic relationships. If we were to impose the same strict constraint, e.g., orthogonal constraint, on all semantic-negative samples in the limited Hamming space, it would inevitably lead to the solution space compression. In this regard, this paper proposes to adopt a differential strategy to handle these three types of samples. Specifically, we impose relatively strict constraint on both fully and partially semantic-positive samples, while constraining semantic-negative samples with relatively loose constraint.

To achieve this, we first utilize $\mathbf{S}$ to estimate the appropriate Hamming distance between different samples. The Hamming

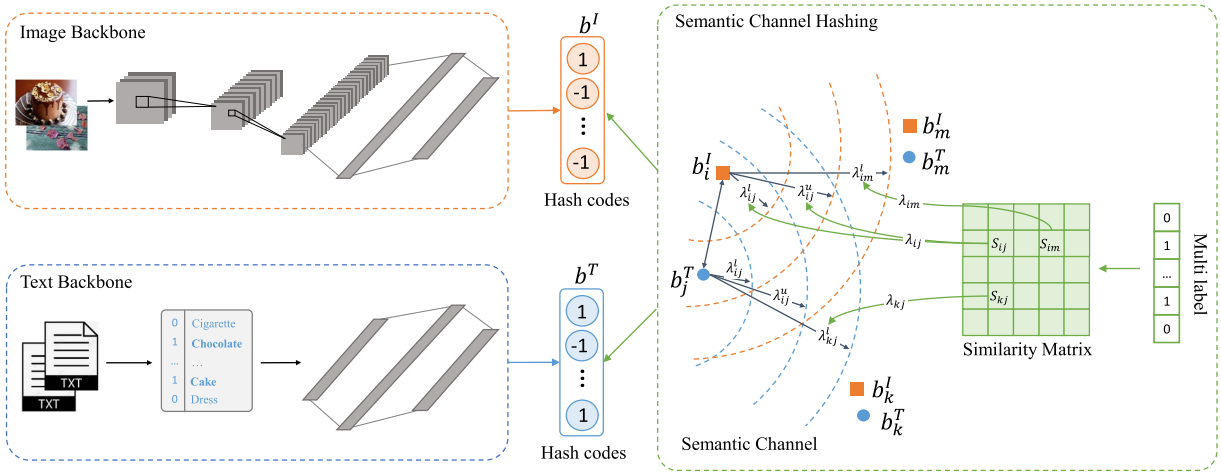


Fig. 2. Semantic channel hashing aims to determine the appropriate Hamming distance based on known information $\mathbf{S}_{ij}$ and use this distance to supervise the learning of hash codes b_i and b_j. First, the similarity $\mathbf{S}_{ij}$ between two samples is calculated by their corresponding labels. Then, the appropriate Hamming distances, including lower bound λ_{ij}^l and upper bound λ_{ij}^u, are estimated by $\mathbf{S}_{ij}$. The Hamming distance between corresponding hash codes b_i and b_j is expected to stay in the range $[\lambda_{ij}^l, \lambda_{ij}^u]$, i.e., $\lambda_{ij}^l \leq d_H(b_i, b_j) \leq \lambda_{ij}^u$.

distance between the hash codes b_i and b_j of the samples x_i and x_j can be computed by

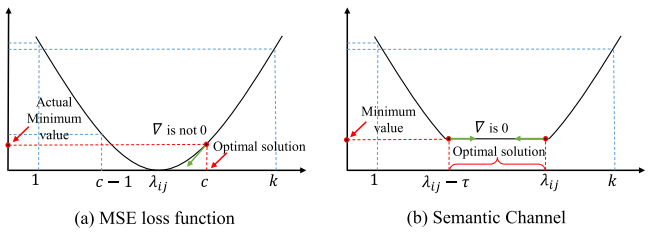
$$d_H(b_i, b_j) = \frac{k}{2}(1 - \cos(b_i, b_j)). \tag{1}$$

Intuitively, if two samples share a similar semantic relationship, we expect $\cos(b_i, b_j)$ to be larger, and vice versa. Thus, $\cos(b_i, b_j)$ is proportional to $\mathbf{S}_{ij}$. Subsequently, we replace $\cos(b_i, b_j)$ in Eq. (1) with $\mathbf{S}_{ij}$ to estimate the appropriate Hamming distance between them as follows:

$$\lambda_{ij} = \frac{k}{2}(1 - \mathbf{S}_{ij}). \tag{2}$$

Obviously, λ_{ij} is bounded by the interval $[0, k/2]$ due to the restriction of $\mathbf{S}_{ij} \in [0, 1]$. This implies that half of the Hamming space remains unused. The underlying cause of this issue is the lack of detailed description for the relationship among semantic-negative samples, which are crudely characterized by $\mathbf{S}_{ij} = 0$. To optimize the utilization of the remaining Hamming space, for any sample x_i, we propose to allocate the space within $k/2$ for semantic-positive samples x_j^+ and $x_j^\oplus$, and reserve the space beyond $k/2$ for semantic-negative samples x_j^-, which can greatly alleviate the problem of solution space compression. The specific implementation will be described in Section III-D.

2) Semantic Channel: It is apparent that λ_{ij} is not always an integer in Eq. (2) and would cause the loss function to oscillate if it is directly utilized to constrain the Hamming distance between hash codes, such as through Mean Squared Error (MSE) loss $\mathcal{L}_{MSE} = (\lambda_{ij} - d_H(b_i, b_j))^2$. This is because the gradient at the optimal solution of the loss function is not always zero in discrete cases. As shown in Fig. 3(a), when λ_{ij} is not an integer, the optimal solution of the loss function $\mathcal{L}_{MSE}$ can only be achieved at point c, where the gradient is not zero. To avoid this situation, we expand λ_{ij} into a channel to ensure that the optimal solution of the loss function falls within the set $\mathcal{D} = \{0, 1, 2, \ldots, k\}$. Specifically, we introduce the concept of a semantic channel with



(a) MSE loss function (b) Semantic Channel

Fig. 3. Gradients at optimal solution of MSE loss and the proposed semantic channel, under discrete situation.

upper and lower bounds denoted as λ_{ij}^u and λ_{ij}^l, respectively, which can be obtained from the following formula:

$$\begin{cases} \lambda_{ij}^l = \lambda_{ij} - \tau \\ \lambda_{ij}^u = \lambda_{ij} \end{cases}, \tag{3}$$

where $\tau \in \mathbf{Z}^+$ is the width of the target semantic channel. We aim for the optimal solution $d_H(b_i, b_j)$ to fall within the semantic channel. That is,

$$\lambda_{ij}^l \leq d_H(b_i, b_j) \leq \lambda_{ij}^u. \tag{4}$$

It needs to guarantee that τ is an integer greater than 1, so that $[\lambda_{ij}^l, \lambda_{ij}^u] \cap \mathcal{D} \neq \emptyset$ is always established. This way, to prevent oscillations in the loss function, we only need to ensure that the loss function achieves its minimum value when $d_H(b_i, b_j)$ satisfies Eq. (4), as shown in Fig. 3(b).

D. Loss Function

1) Semantic-Negative Samples: For semantic-negative samples (x_i, x_j^-), it is desired that their corresponding hash codes (b_i, b_j^-) are distant from each other in the Hamming space. However, as discussed in Section III-C, the similarity $\mathbf{S}_{ij} = 0$ does not provide specific information about the exact distance range for $d_H(b_i, b_j^-)$. Therefore, in the absence of the semantic channel

and setting $\lambda_{ij}^{l} = k/2$, our focus is solely on the lower bound λ_{ij}^{l} of (x_i, x_j^-). Specifically, we aim for the Hamming distance $d_H(b_i, b_j^-)$ to exceed the lower bound λ_{ij}^{l}, i.e., $\lambda_{ij}^{l} \leq d_H(b_i, b_j^-)$, ensuring their semantic negativity, without requiring a precise magnitude. Consequently, we can define the loss function for semantic-negative samples as follows:

$$\mathcal{L}_{neg} = \sum_{*}^{\{I,T\}} \sum_{i,j}^{n} \max\{0, \lambda_{ij}^{l} - d_H(b_i^*, b_j^{*-})\}. \tag{5}$$

This loss function comprises four components. The terms $d_H(b_i^I, b_j^{I-})$ and $d_H(b_i^T, b_j^{T-})$ represent the Hamming distances between hash codes from the same modality, ensuring intra-modal similarity. On the other hand, the terms $d_H(b_i^I, b_j^{T-})$ and $d_H(b_i^T, b_j^{I-})$ represent the Hamming distances between hash codes from different modalities, ensuring inter-modal similarity.

2) Partially Semantic-Positive Samples: The loss function for partially semantic-positive samples involves ensuring the proper distribution of their corresponding hash codes (b_i, b_j^+). Since the similarity value $\mathbf{S}_{ij}$ provides specific information about how close they should be, we use the upper and lower bounds from Eq. (3) simultaneously to constrain the distance of (b_i, b_j^+). The goal is to have their distance be close, but not excessively close. Thus, we define the distance of (b_i, b_j^+) to satisfy the condition $\lambda_{ij}^{l} \leq d_H(b_i, b_j^+) \leq \lambda_{ij}^{u}$, where the upper bound λ_{ij}^{u} maintains the proximity of the distance, while the lower bound λ_{ij}^{l} prevents the distance from being too small. This leads to the following formulation for the loss function of partially semantic-positive samples:

$$\sum_{*}^{\{I,T\}} \sum_{i,j}^{n} max\{0, \lambda_{ij}^{l} - d_H(b_i^*, b_j^{*+}), d_H(b_i^*, b_j^{*+}) - \lambda_{ij}^{u}\}. \tag{6}$$

In Eq. (6), only one of the three terms can be positive at most. When $d_H(b_i, b_j^+) < \lambda_{ij}^{l}$, only the second term $\lambda_{ij}^{l} - d_H(b_i, b_j^+)$ is positive, indicating that b_j^+ is too close to b_i. Similarly, when $\lambda_{ij}^{u} < d_H(b_i, b_j^+)$, only the third term $d_H(b_i, b_j^+) - \lambda_{ij}^{u}$ is positive, indicating that b_j^+ is relatively far from b_i. Only when $\lambda_{ij}^{u} \leq d_H(b_i, b_j^+) \leq \lambda_{ij}^{l}$, all three terms are non-positive, indicating that the distance between (b_i, b_j^+) is appropriate, close but not excessively so. Therefore, Eq. (6) can be reformulated as follows:

$$\begin{aligned}\mathcal{L}_{ppos} = &\sum_{*}^{\{I,T\}} \sum_{i,j}^{n} max\{0, \lambda_{ij}^{l} - d_H(b_i^*, b_j^{*+})\} \\ &+ \sum_{*}^{\{I,T\}} \sum_{i,j}^{n} max\{0, d_H(b_i^*, b_j^{*+}) - \lambda_{ij}^{u}\}.\end{aligned} \tag{7}$$

3) Fully Semantic-Positive Samples: For fully semantic-positive samples $(x_i, x_j^{\oplus})$, their similarity $\mathbf{S}_{ij} = 1$ represents the strongest constraint between the two samples, indicating that they should have highly similar hash codes $(b_i b_j^{\oplus})$. From Eq. (3), the lower bound λ_{ij}^{l} for fully semantic-positive samples is always negative, i.e., $-\tau$. As a result, the second term $\lambda_{ij}^{l} - d_H(b_i, b_j^{\oplus})$ in Eq. (6) is always non-positive. Therefore, we only need to consider the upper bound λ_{ij}^{u}, which ensures that the hash codes of fully semantic-positive samples are kept close to each other in the Hamming space. Consequently, for fully semantic-positive samples, Eq. (6) can be simplified as follows:

$$\mathcal{L}_{fpos} = \sum_{*}^{\{I,T\}} \sum_{i,j}^{n} max\{0, d_H(b_i^*, b_j^{*\oplus}) - \lambda_{ij}^{u}\}. \tag{8}$$

4) Overall Loss Function: Combining these three losses $\mathcal{L}_{neg}$, $\mathcal{L}_{ppos}$, and $\mathcal{L}_{fpos}$, the overall loss function can be derived by

$$\mathcal{L} = \mathcal{L}_{ppos} + \alpha \mathcal{L}_{fpos} + \beta \mathcal{L}_{neg}, \tag{9}$$

where α and β are two hyper-parameters. To address the time-consuming construction of positive and negative sample pairs when implementing the loss function in Eq. (9), we introduce an equivalent matrix form of the loss function

$$\mathcal{L} = ||\mathbf{W}^l \odot \sigma(\Lambda^l - \mathbf{B}\mathbf{B}^\top)||_F + ||\mathbf{W}^u \odot \sigma(\mathbf{B}\mathbf{B}^\top - \Lambda^u)||_F, \tag{10}$$

where $\Lambda^l = [\lambda_{i,j}^{l}]^{n \times n}$, $\Lambda^u = [\lambda_{i,j}^{u}]^{n \times n}$, $\sigma(\cdot) = max\{0, \cdot\}$, and $\odot$ is matrix dot product. $\mathbf{W}^l$ and $\mathbf{W}^u$ are two weight matrices that are defined as follow:

$$\mathbf{W}_{ij}^{l} = \begin{cases} 1, & \mathbf{S}_{ij} \in (0,1) \\ \beta, & \mathbf{S}_{ij} = 1 \\ 0, & \mathbf{S}_{ij} = 0 \end{cases}, \mathbf{W}_{ij}^{u} = \begin{cases} 1, & \mathbf{S}_{ij} \in (0,1) \\ 0, & \mathbf{S}_{ij} = 1 \\ \alpha, & \mathbf{S}_{ij} = 0 \end{cases}. \tag{11}$$

It is worth noting that in Eq. (10), we opt to use the $||\cdot||_F$ instead of $||\cdot||_1$. This choice is made because when deriving Eq. (10), the additional denominator in the gradient, i.e., $\frac{1}{||\cdot||_F}$, allows for the consideration of the global distribution between positive and negative samples during the update process. Algorithm 1 summarizes the overall training procedure of the proposed SCH.

5) Relation With Triplet-Margin Loss: The triplet-margin loss is a commonly used loss function in rank-based cross-modal hashing methods [60], [61]. The objective of the

Algorithm 1: SCH.

Require: Training dataset $\mathcal{T} = \{\mathbf{X}^I, \mathbf{X}^T, \mathbf{L}\}$;
An image backbone network $\mathbf{f}^I(\mathbf{X}^I; \omega^I)$ which is parameterized by ω^I and learning rate γ^I;
A text backbone network $\mathbf{f}^T(\mathbf{X}^T; \omega^T)$ which is parameterized by ω^T and learning rate γ^T.
Two hyper-parameters α and β.

1: **for** $iter = 1$ to maximum iteration **do**
2: Sample batch samples $\mathcal{B}$ from $\mathcal{T}$;
3: Estimate the similarities for samples in $\mathcal{B}$;
4: Estimate the appropriate Hamming distances for samples in $\mathcal{B}$ via Eq. (2);
5: Assign appropriate semantic channels for samples in $\mathcal{B}$ via Eq. (3);
6: Update the parameters of image backbone via $\omega^I = \omega^I - \gamma^I \nabla_{\omega^I} \mathcal{L}((\mathbf{X}^I, \mathbf{X}^T, \mathbf{L}); \omega^I)$;
7: Update the parameters of text backbone via $\omega^T = \omega^T - \gamma^T \nabla_{\omega^T} \mathcal{L}((\mathbf{X}^I, \mathbf{X}^T, \mathbf{L}); \omega^T)$;
8: **end for**

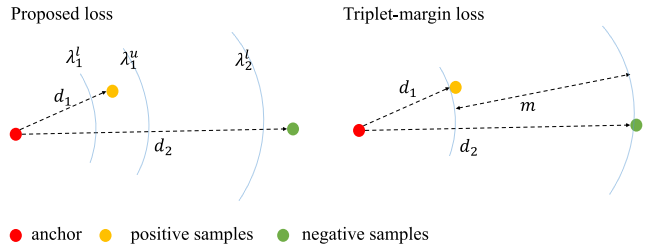


Fig. 4. Difference between the proposed loss and triplet-margin loss. In the proposed loss, absolute distance is considered, i.e., $\lambda_1^l < d_1 < \lambda_1^u$ and $\lambda_2^l < d_2$. In contrast, triplet-margin loss cares more about relative distance, i.e., $d_2 - d_1 - m > 0$.

triplet-margin loss is to increase the distance between semantic-negative samples compared to the distance between semantic-positive samples. The typical formulation of the triplet-margin loss is:

$$\sum_{*}^{\{I,T\}} \sum_{i,j}^{n} \max\{0, m + d_H(b_i^*, b_j^{*+}) - d_H(b_i^*, b_j^{*-})\}, \quad (12)$$

where m controls the desired distance threshold in the triplet-margin loss. Our proposed loss function shares certain similarities with the triplet-margin loss in terms of the underlying idea. The difference between them is that our loss is based on absolute distances, while the triplet-margin loss primarily focuses on relative distances. Consider three samples: an anchor, a positive sample, and a negative sample, as illustrated in Fig. 4. Let d_1 denote the distance between the anchor and the positive sample, and d_2 denote the distance between the anchor and the negative sample. The triplet-margin loss aims to ensure that d_2 is larger than d_1 by a margin of m, i.e., $d_2 - d_1 - m > 0$. However, the specific value of d_1 is less concerned. In contrast, our proposed loss simultaneously constrains both d_1 and d_2, with the conditions $\lambda_1^l < d_1 < \lambda_1^u$ and $\lambda_2^l < d_2$. This constraint can be approximated as $d_2 - d_1 - (\lambda_2^l - \lambda_1^u) > 0$, which is consistent with the formulation of the triplet-margin loss.

The proposed loss offers two advantages over the triplet-margin loss. First, it eliminates the need for a hyper-parameter. In the triplet-margin loss, the parameter m plays a critical role, but its selection is often based on empirical observation. Additionally, adjusting m becomes necessary when the dimension of the Hamming space changes. In contrast, the proposed loss does not rely on any hyper-parameter, which simplifies the training process. Second, in the triplet-margin loss, constructing triplet tuples can be time-consuming and computationally intensive. This process involves identifying suitable anchor-positive-negative combinations, which can impede the efficiency of the model in terms of training time and computational resources [76]. The proposed loss, on the other hand, does not require the explicit construction of triplets, making it more efficient.

IV. Experiment

A. Experiment Setting

1) Datasets: To validate the effectiveness of the proposed method and its competitors, a series of experiments were conducted on three widely-used datasets, including MIRFlickr [77], NUS-WIDE [78], and IAPR TC-12 [79].

MIRFlickr comprises 25,000 image-text pairs. It is composed of 24 different concepts. The images are represented using raw RGB features, while each text is represented by a 1,386-dimensional BoW vector. Following [21], we excluded samples whose textual tags appear less than 20 times, leaving us with a total of 20,015 image-text pairs. Out of these pairs, we used 2,000 pairs as the test set and reserved the remaining 18,015 pairs for retrieval purposes. We randomly selected 10,000 samples from the retrieval set as our training set.

NUS-WIDE comprises 260,648 image-text pairs. Each image-text pair is assigned at least one label from 81 possible concepts. The images are represented using raw RGB features, while each text is represented by a 1,000-dimensional BoW vector. We selected only the samples belonging to the top 10 most frequent concepts, resulting in a total of 186,577 pairs. Out of these pairs, 2,000 pairs were used as the test set, while the remaining 184,577 pairs were used as the retrieval set. Similar to our protocol with MIRFlickr, we randomly selected 10,000 samples from the retrieval set as our training set.

IAPR TC-12 contains 20,000 image-text pairs, each of which is annotated with multi-labels from a set of 255 semantic categories. Each image is represented by a 4,096-dimensional vector that was extracted by the pretrained CNN-F [80], while each text is represented by a 2,912-dimensional BoW vector. Following [50], we randomly selected 2,000 pairs as the test set, while the remaining pairs were used as both the training and retrieval sets.

Since only a portion of the MIRFlickr and NUS-WIDE datasets are used as training samples, to ensure they are representative of all categories, we uniformly select an equal number of samples from all categories, forming a comprehensive training set that covers all categories, same as the settings in [47], [48]. Similarly, we also select an equal number of samples from all categories to form the test set.

2) Implement Details: As the focus of this paper is not on model design, we employ the commonly used twin-tower model, which employs different backbones for each modality. Specifically, we utilize VGG-19 [81] as the backbone for image modality, and a 3-layer MLP for text modality. We utilize the last layer of HashNet [68] as the output layer to ensure discrete attribute of hash codes. The hyper-parameters α and β are set to 1. The semantic channel width τ is set to 3. It is generally accepted that more complex backbones can learn more expressive features. Thus, we also provide a comparison of our method with some state-of-the-art methods using different backbones in Section IV-B-4 to demonstrate the effectiveness of our approach. Additionally, we utilize the SGD optimizer with 0.9 momentum and 5×10^{-4} weight decay. The learning rates of both image and text net are 0.005 and the training batch size is set to 32.

TABLE I
MEAN AVERAGE PRECISION (MAP) SCORE COMPARISONS OF ALL APPROACHES ON TWO DATASETS MIRFLICKR AND NUS-WIDE

Methods	MIRFlickr								NUS-WIDE							
	$I \rightarrow T$				$T \rightarrow I$				$I \rightarrow T$				$T \rightarrow I$			
	16	32	64	128	16	32	64	128	16	32	64	128	16	32	64	128
CVH	0.561	0.561	0.562	0.567	0.561	0.562	0.562	0.566	0.386	0.399	0.409	0.413	0.393	0.400	0.408	0.415
STMH	0.566	0.581	0.599	0.602	0.560	0.578	0.597	0.599	0.542	0.554	0.557	0.561	0.526	0.556	0.553	0.556
CMSSH	0.586	0.584	0.572	0.571	0.566	0.569	0.561	0.562	0.532	0.532	0.529	0.536	0.407	0.416	0.405	0.407
SCM	0.655	0.670	0.673	0.678	0.665	0.679	0.687	0.694	0.560	0.583	0.581	0.582	0.497	0.510	0.510	0.511
SePH	0.671	0.674	0.675	0.682	0.691	0.697	0.700	0.706	0.527	0.541	0.543	0.551	0.562	0.566	0.579	0.586
DCMH	0.721	0.728	0.741	0.740	0.753	0.759	0.773	0.775	0.590	0.627	0.622	0.628	0.556	0.614	0.618	0.624
ATFH-N	0.702	0.721	0.746	0.750	0.737	0.753	0.762	0.764	0.605	0.649	0.646	0.645	0.591	0.622	0.611	0.617
CHN	0.763	0.779	0.789	0.791	0.764	0.781	0.791	0.792	0.638	0.652	0.670	0.673	0.636	0.641	0.648	0.652
SSAH	0.761	0.784	0.792	0.801	0.768	0.775	0.784	0.791	0.648	0.665	0.680	0.687	0.634	0.660	0.667	0.674
EGDH	0.757	0.772	0.796	0.799	0.761	0.773	0.790	0.796	0.650	0.668	0.681	0.684	0.638	0.660	0.664	0.670
AGAH	0.762	0.785	0.796	0.805	0.789	0.790	0.805	0.812	0.663	0.680	0.682	0.689	0.656	0.667	0.671	0.679
MSSPQ	0.787	0.801	0.817	-	0.795	0.789	0.802	-	0.635	0.648	0.662	-	0.631	0.663	0.688	-
HMAH	0.756	0.760	0.771	0.783	0.751	0.762	0.773	0.794	0.642	0.649	0.658	0.662	0.638	0.642	0.650	0.657
MAFH	0.741	0.756	0.768	0.780	0.732	0.746	0.758	0.760	0.632	0.641	0.649	0.657	0.634	0.651	0.662	0.665
MIAN	**0.801**	**0.812**	0.817	**0.826**	0.805	0.817	0.826	0.830	0.663	0.673	0.684	0.681	0.676	0.692	0.698	0.704
SCH	0.794	0.810	**0.820**	0.822	**0.827**	**0.842**	**0.845**	**0.851**	**0.683**	**0.701**	**0.706**	**0.699**	**0.718**	**0.733**	**0.736**	**0.739**

The best performance is boldfaced.

The proposed SCH is implemented with Pytorch on an NVIDIA Tesla V100-32G.

3) Compared Methods: We compare the proposed SCH with some classical shallow-feature based baselines, including CVH [82], STMH [83], CMSSH [84], SCM [85], SePH [19], and several state-of-the-art end-to-end cross-modal hashing methods, including: DCMH [21], ATFH-N [86], CHN [87], SSAH [22], EGDH [12], AGAH [61], MSSPQ [45], HMAH [88], MAFH [89], MIAN [23]. To ensure fair comparisons with shallow-feature based baselines, we utilize 4096-dimensional image features extracted by the pre-trained VGG-19 network as the input.

4) Evaluation Metrics: We evaluate the retrieval performance using two retrieval tasks: Image to Text retrieval ($I \rightarrow T$) and Text to Image retrieval ($T \rightarrow I$). In the former, we use images as queries to retrieve corresponding texts from the retrieval set, while in the latter, we use texts as queries to retrieve corresponding images. Considering that all compared methods are hamming ranking approaches, following the suggestion in [90], we employ three commonly used criteria to assess retrieval performance: mean Average Precision (mAP), precision-recall curve, and top-K precision curve. For Hamming ranking approaches, retrieval time and index memory are solely dependent on the length of the hash codes. Once hash code lengths are identical, the retrieval time and storage space on the same device remain consistent. Therefore, these aspects are not specifically presented within this paper.

B. Results and Analysis

1) Retrieval Performance: The mAP results on the three datasets are presented in Tables I and II. The competitors encompass three typical approaches: 1) SePH is a classic method that employs orthogonal constraints to restrict semantic-negative samples; 2) Most deep methods, such as DCMH, SSAH, and MIAN, incorporate constraints to keep semantic-negative samples as far apart as possible; 3) AGAH utilizes triplet-margin loss. In most cases, SCH outperforms all the compared methods.

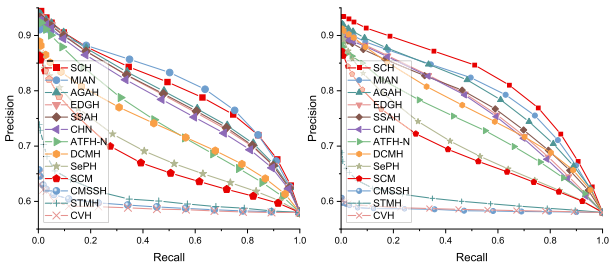
TABLE II
THE MEAN AVERAGE PRECISION (MAP) SCORES OF ALL APPROACHES ON THE IAPR TC-12 DATASET ARE COMPARED

Methods	IAPR-TC12					
	$I \rightarrow T$			$T \rightarrow I$		
	32	64	128	32	64	128
CVH	0.384	0.385	0.372	0.395	0.389	0.368
STMH	0.481	0.487	0.495	0.479	0.482	0.484
CMSSH	0.487	0.485	0.487	0.482	0.485	0.491
SCM	0.512	0.520	0.534	0.507	0.524	0.538
SePH	0.534	0.561	0.578	0.527	0.545	0.569
DCMH	0.527	0.546	0.567	0.520	0.547	0.565
ATFH-N	0.524	0.548	0.564	0.514	0.532	0.557
CHN	0.554	0.579	0.601	0.537	0.559	0.582
SSAH	0.561	0.581	0.605	0.544	0.561	0.597
EGDH	0.549	0.574	0.597	0.532	0.558	0.579
AGAH	0.571	0.589	0.611	0.557	0.574	0.596
HMAH	0.561	0.582	0.597	0.542	0.561	0.579
MAFH	0.568	0.597	0.601	0.551	0.574	0.591
MIAN	0.582	0.591	0.605	0.553	0.564	0.585
SCH	**0.586**	**0.608**	**0.630**	**0.571**	**0.607**	**0.623**

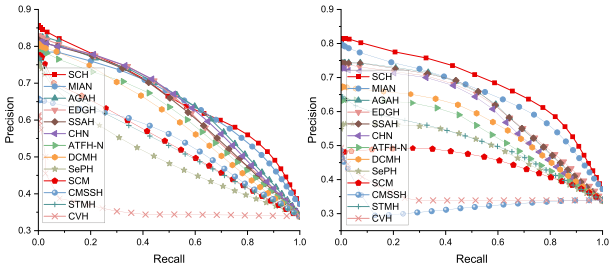
Best performance is highlighted in bold.

Besides, SCH utilizes a basic twin-tower model and straightforward network architectures without placing significant emphasis on the alignment of hash codes across different modalities. In contrast, competing methods such as SSAH and MIAN incorporate specialized network structures for text data to extract more informative text representations. Moreover, ATFH-N, SSAH, and AGAH introduce adversarial networks to align hash codes, while SSAH additionally incorporates a network for label information to enhance the supervision signals. SCH outperform these competitors even with the simple two-tower model, which confirms the effectiveness of bridging the space gap.

In addition, we present precision-recall curves on MIRFlickr and NUS-WIDE datasets with 128-bit hash code length in Fig. 5 and top-K precision curves in Fig. 6. Notably, SCH consistently achieves the best or second best precision results at the same recall rates, particularly in the $T \rightarrow I$ tasks. This observation indicates that the distribution of hash codes in the retrieved set has been significantly optimized in terms of Hamming distance,

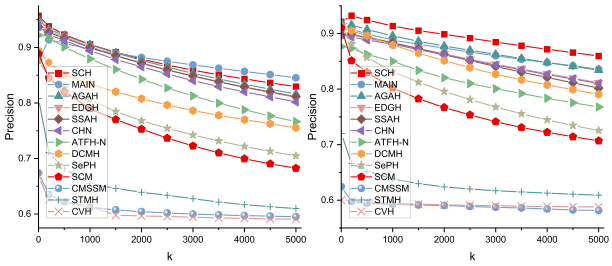


(a) MIR@128 bits $I \rightarrow T$ (b) MIR@128 bits $T \rightarrow I$

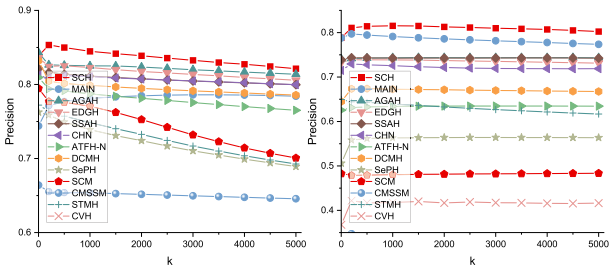


(c) NUS@128 bits $I \rightarrow T$ (d) NUS@128 bits $T \rightarrow I$

Fig. 5. Precision-recall curves of all methods on MIRFlickr and NUS-WIDE datasets. The code length is 128.



(a) MIR@128 bits $I \rightarrow T$ (b) MIR@128 bits $T \rightarrow I$



(c) NUS@128 bits $I \rightarrow T$ (d) NUS@128 bits $T \rightarrow I$

Fig. 6. Top-k precision curves of all methods on MIRFlickr and NUS-WIDE datasets. The code length is 128.

resulting in the aggregation of samples with similar semantics. As a result, with the same recall rate, more correct results are ranked at the top of the retrieval results. This finding is further supported by the results in Fig. 6, where SCH consistently demonstrates the highest precision among the top k retrieved samples (where $k \leq 5000$), particularly in the $T \rightarrow I$ task.

2) Ablation Study: In this section, we investigate the influence of three types of samples, i.e., semantic-negative samples (x_i, x_j^-), partially semantic-positive samples (x_i, x_j^+), and fully semantic-positive samples $(x_i, x_j^{\oplus})$, on the retrieval performance. We implement three variants of SCH: SCH w./o. $\mathcal{L}_{neg}$, SCH w./o. $\mathcal{L}_{ppos}$, and SCH w./o. $\mathcal{L}_{fpos}$. These variants correspond to the loss function Eq. (9) without $\mathcal{L}_{neg}$, $\mathcal{L}_{ppos}$, and $\mathcal{L}_{fpos}$, respectively. The hash code length is set to 128, and experiments are conducted on the MIRFlickr and NUS-WIDE datasets. The results are presented in Table III. It can be observed that

TABLE III
ABLATION STUDY ON MIRFLICKR AND NUS-WIDE DATASETS

	MIRFlickr		NUS-WIDE	
	$I \rightarrow T$	$T \rightarrow I$	$I \rightarrow T$	$T \rightarrow I$
SCH	**0.8220**	**0.8506**	**0.6991**	**0.7388**
SCH w./o. $\mathcal{L}_{neg}$	0.7384	0.7368	0.5677	0.5499
SCH w./o. $\mathcal{L}_{ppos}$	0.7724	0.7611	0.6621	0.6814
SCH w./o. $\mathcal{L}_{fpos}$	0.8201	0.8473	0.6920	0.7322

The code length is 128. The best performance is boldfaced.

- The performance of SCH w./o. $\mathcal{L}_{neg}$ shows a significant drop, with more than 8% and 10% decrease on the MIRFlickr and NUS-WIDE datasets, respectively. This result emphasizes the importance of properly distributing semantic-negative samples. Specifically, it is crucial to ensure that the distance between semantic-negative samples is greater $k/2$. It helps prevent semantic-negative samples from being located near the query during retrieval stage.
- The average performance of both tasks of SCH w./o. $\mathcal{L}_{ppos}$ also shows a significant drop, with approximately 7% and 5% decrease on the MIRFlickr and NUS-WIDE datasets, respectively. This finding highlights the importance of partially semantic-positive samples, as they contain valuable and informative data. If only the separation of semantic-negative samples is ensured, while the positioning of partially semantic-positive samples is neglected, the overall retrieval performance will still be greatly compromised.
- The performance of SCH w./o. $\mathcal{L}_{fpos}$ shows a drop of approximately 1% on both tasks and datasets. One possible reason for this observation is that fully semantic-positive samples naturally have similar representations and are prone to clustering together, even without the explicit constraint $\mathcal{L}_{fpos}$. Furthermore, the organized arrangement of partially semantic-positive samples may indirectly encourage the gathering of fully semantic-positive samples.

3) Parameter Sensitive: We also investigate the sensitivity of the parameters α and β. We set their ranges to $\{0.001, 0.01, 0.1, 1, 1.5, 2\}$ and report the results in Fig. 7. From Fig. 7(a) and (c), it can be observed that the proposed SCH is not highly sensitive to changes in α. This aligns with the analysis presented in Section IV-B-2, indicating that even with small values of α, fully semantic-positive samples tend to cluster effectively in the Hamming space. Conversely, when using small values of β (e.g., $\beta = \{0.001, 0.01, 0.1\}$), as shown in Fig. 7(b) and (d), the performance of SCH experiences a significant drop. These results also corroborate the earlier analysis, i.e., maintaining a sufficient separation among semantic-negative samples is crucial. Therefore, small values of β weaken the effectiveness of $\mathcal{L}_{neg}$ and consequently lead to inferior performance.

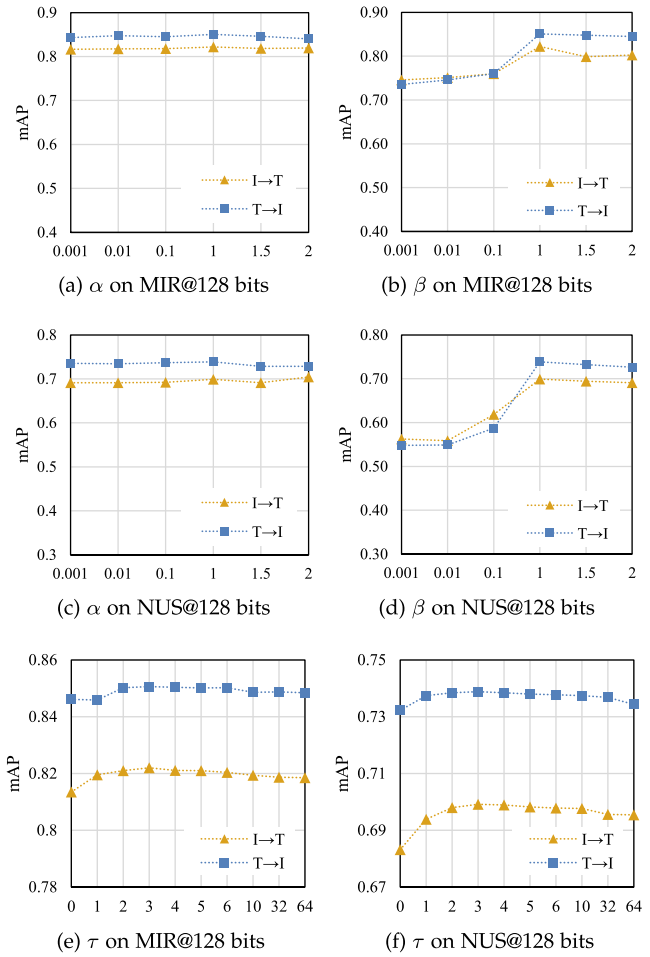


Fig. 7. Parameter sensitive of α and β on MIRFlickr and NUS-WIDE datasets. The code length is 128.

We further conduct experiments on the MIRFlickr and NUS-WIDE datasets to assess the impact of different values of semantic channel width τ, and the results are presented in Fig. 7(e) and (f). It is evident that the performance with $\tau > 0$ is higher than that with $\tau = 0$, providing evidence for the effectiveness of the semantic channel. Additionally, excessively large values of τ do not yield significant improvements in retrieval performance. Considering that τ in Eq. (3) is utilized to determine the lower bound for semantic-positive samples, the results in Fig. 7 may suggest that, as long as the upper bound is appropriately determined to ensure that semantic-positive samples are not too distant, the choice of channel width does not have stringent constraints, provided that $\tau > 0$ is maintained.

4) Various Backbones: With the widespread use of attention-based models such as Transformer, BERT, and ViT in the multi-modal field, some recent cross-modal retrieval methods have used them as backbones to extract image and text features. For instance, CMGCAH [27] uses ViT as the backbone of the image modality, while UniHash [26] uses BERT as the backbone of the both modalities. To compare with these methods, we replace the image backbone VGG of SCH with image encoder (ViT-B/32, ViT-B/16, and ViT-L/14) of CLIP [91], [92]. The

TABLE IV
COMPARISON OF SOTA METHODS UNDER DIFFERENT BACKBONES ON NUSWIDE DATASET

Method	Img. B.	Txt. B.	32	64
CMGCAH	ViT	MLP	0.6440/0.6801	0.6560/0.6844
UniHash	BERT	BERT	0.7027/0.7261	0.7051/0.7270
SCH	VGG	MLP	0.7009/0.7330	0.7057/0.7363
	ViT-B/32		0.6973/0.7341	0.7137/0.7373
	ViT-B/16		**0.7182/0.7386**	**0.7204/0.7412**
	ViT-L/14		0.7081/0.7382	0.7121/0.7378

/ indicates mAP scores of I → T/T → I.

experimental results are shown in Table IV. It can be observed that employing multimodal pretrained backbones can further enhance the performance of SCH. However, this improvement does not always persist with an increase in model size. For instance, despite ViT-L/14 having the most parameters, its performance is not superior to ViT-B/16. A plausible reason is that, although ViT-L/14 learns features superior to ViT-B/16 in the real-valued domain, these differences are attenuated after encoding through the hash functions.

C. Analyses of Space Gap

In Section I, we analyze how the space gap undermines existing CMH methods, resulting in the problems of compressed solution space and loss function oscillation. In this section, we experimentally verify these two claims.

1) Solution Space Compression: We first introduce two commonly used loss functions. The first is the mean square error (MSE) loss function

$$\mathcal{L}_{MSE} = \sum_{i,j}(\mathbf{S}_{ij} - \Phi_{ij})^2, \tag{13}$$

where Φ_{ij} is the similarity between hash codes, generally $\Phi_{ij} = \cos(b_i, b_j)$ is used. Here the modal information is omitted for brevity. Under this constraint, semantic-negative samples are required to be orthogonal. The second is the negative log-likelihood (NLL) loss function

$$\mathcal{L}_{NNL} = \sum_{i,j}(\log(1 + e^{\Phi_{ij}}) - \mathbf{S}_{ij}\Phi_{ij}), \tag{14}$$

which requires semantic-negative samples to be as far away as possible. We design three models M1-M3 with consistent backbones, but using Eq. (10), MSE, and NNL as loss functions, respectively. Their loss functions and aims are summarized in Table V.

Next, we create a subset of the NUS-WIDE dataset called NUS-sub. This is done by selecting the 4 most frequent single labels from NUS-WIDE dataset and assigning them new labels [1, 0, 0], [0, 1, 0], [0, 0, 1], and [0, 0, 0]. Furthermore, we select the multi-labels obtained from the combination of the first three single labels and assigned them new labels [1, 0, 1], [0, 1, 1], [1, 1, 0], and [1, 1, 1]. As a result, we create a dataset consisting of 8 new classes, and for each class, we select 150 samples for training. The details of the NUS-sub dataset are presented in Table VI and some samples are presented in Fig. 8. It is worth noting that samples in classes C1-C7 have both

TABLE V
THREE DIFFERENT CONSTRAINTS IMPOSED ON HASH CODES IN CMH METHODS

No.	Loss function	Aims
M1	Eq. (10)	Ensure that semantic-positive samples are in close proximity while keeping semantic-negative samples at a distance *greater than* $k/2$.
M2	Eq. (13)	Ensure that semantic-positive samples are in close proximity while keeping semantic-negative samples *orthogonal*.
M3	Eq. (14)	Ensure that semantic-positive samples are in close proximity while keeping semantic-negative samples being *as far apart as possible*.

TABLE VI
SUMMARY OF THE NUS-SUB DATASET

Class	Label	#Train	Semantic-Pos.	Semantic-Neg.
C1	100	150	C4,C5,C7	C2,C3,C6,C8
C2	010	150	C4,C6,C7	C1,C3,C5,C8
C3	001	150	C5,C6,C7	C1,C2,C4,C8
C4	110	150	C1,C2,C5,C6,C7	C3,C8
C5	101	150	C1,C3,C4,C6,C7	C2,C8
C6	011	150	C2,C3,C4,C5,C7	C1,C8
C7	111	150	C1-C6	C8
C8	000	150	-	C1-C7

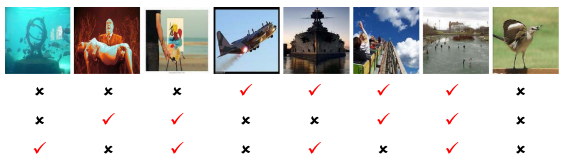


Fig. 8. Some samples of NUS-sub dataset.

semantic-positive and semantic-negative relationships, while samples in class C8 only have a semantic-negative relationship with the other 7 classes. We utilize the NUS-sub dataset to train models M1-M3 and obtained their corresponding hash codes. The average distance between each class is visualized in Fig. 9. Comparing Fig. 9(a)–(c), we observe that

- All three models are able to preserve the correct semantic relationships. For instance, the distances between class C8 and the other classes are the farthest, while class C1 is farther away from classes C2, C3, and C6 but relatively closer to classes C4, C5, and C7.
- The distances between each class are compressed in model M2. Specifically, class C8 remain orthogonal to the other classes, resulting in a distance of about 8. On the other hand, compared to model M1, the distances between classes C1-C7 in M2 are generally reduced by about 1. This phenomenon suggests that the orthogonal constraint does lead to compression of the solution space.
- The model M3 exhibits an overemphasis on separating semantic-negative samples. This is due to the fact that in Eq. (14), when $\mathbf{S}_{ij} = 0$, $\min(\log(1 + e^{\Phi_{ij}}))$ causes Φ_{ij} to converge towards -1 indefinitely, resulting in all semantic-negative samples continuing to contribute to the gradient during training.

Overall, despite presenting different biases, all three models successfully preserve the correct semantic relationships. To further evaluate their performance, we introduce additional single-label classes C9-C13 to NUS-sub. As a result, each class in C8-C13 forms a semantic-negative relationship with all other classes. We train three models on this extended dataset and visualize the average distance between hash codes among classes in Fig. 9. When comparing Fig. 9(d)–(f), it is evident that M1 still successfully preserves the correct semantic relationships. However, due to the orthogonal constraint on M2, the space encompassing C1-C7 undergoes further compression and some semantic-negative samples, e.g., C10 and C11, become inseparable. M3 also faces this problem, some semantic-negative samples are also forced to be squeezed together, as seen with C10 and C11. As mentioned previously, although the semantic-positive and semantic-negative samples could balance each other, as the number of categories and samples increases, this balance will eventually be disrupted, leading to a compressed space on the side with a small number of samples. Under this experiment setting, compared with the close relationship among C1-C7, which can be regarded as a large group, the connection between remaining single-label classes C8-C13 are weaker. As a consequence, the space allocated to C8-C13 are compressed to accommodate C1-C7. Specifically in Fig. 9(e) and (f), C1-C7 finally compresses the space of C10-C11.

2) Loss Function Oscillation: To investigate the influence of the semantic channel on loss function oscillation, two variants of SCH were implemented: SCH1 with the semantic channel width τ set to 0, and SCH2 with orthogonal constraints replacing the constraints on semantic negative samples and $\tau = 0$. These two models and SCH are evaluated on the NUS-sub dataset (13 classes). The loss function and average mAP on the two tasks are depicted in Fig. 10. To mitigate the impact of large learning rates on the loss function, the learning rate was reduced to one-tenth of its previous value every 10 epochs after first 100 epochs. We calculate mAP score every 20 epochs.

From Fig. 10, it is evident that the loss function of SCH exhibits the smoothest behavior. Comparing SCH with SCH1, we can observe the role of the semantic channel in alleviating the oscillation in the loss function. In contrast, the loss function of SCH2 still shows noticeable oscillation even after convergence. This phenomenon can be attributed to two reasons. First, in theory, the absence of a semantic channel means that the gradient at the optimal solution of the loss function cannot be guaranteed to be zero. Second, the strict orthogonal constraint leads to the compression of the solution space, causing certain squeezed samples to continuously contribute to the gradient. For instance, in Classes C10 and C11 (as shown in Fig. 9(e)), which should ideally be orthogonal, they are instead squeezed together, resulting in a continued contribution of gradients that counteract this compression. This effect is more pronounced in terms of the mAP score, as SCH2 exhibits an unstable mAP value even after convergence. In contrast, both SCH and SCH1 exhibit relative stability after convergence. It is worth noting that despite setting $\tau = 0$ in SCH1, it still reaches a relatively stable state. One possible reason for the observation is related to

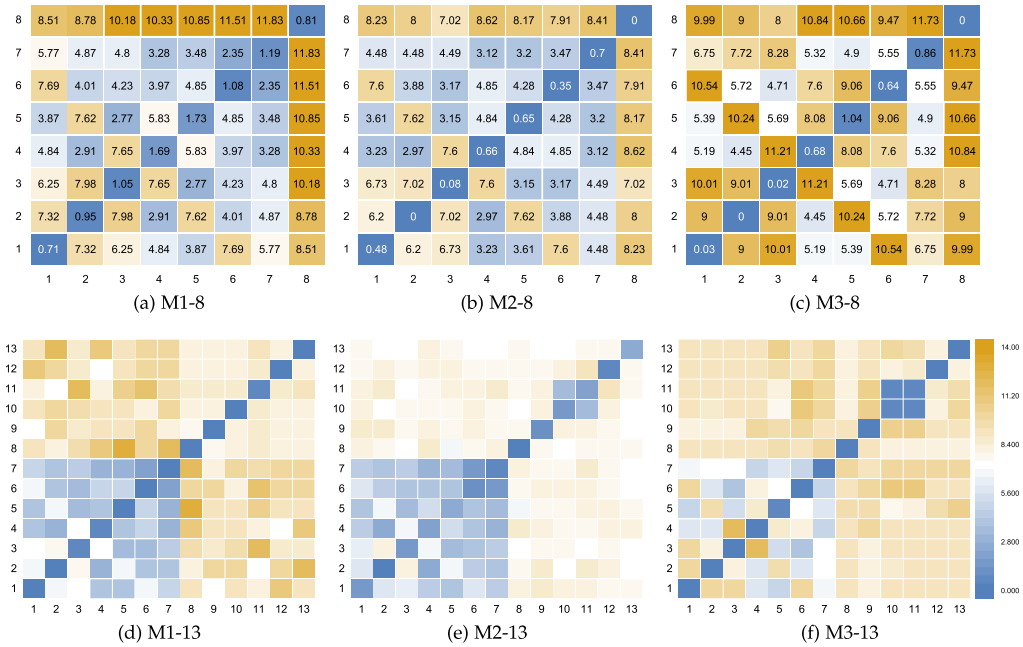


Fig. 9. The average distance between all classes of models M1-M3 is computed for a hash code length of 16. (a)–(c) Display the results on the NUS-sub dataset with 8 classes (C1-C7), while (d)–(f) depict the results on the NUS-sub dataset with 13 classes (C1-C13).

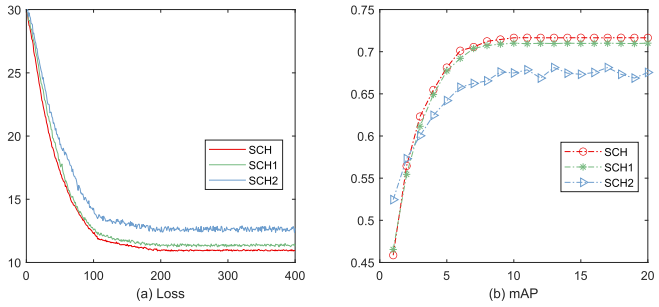


Fig. 10. Loss functions and average retrieval mAP scores of the three models on the NUS-sub (13 classes) dataset.

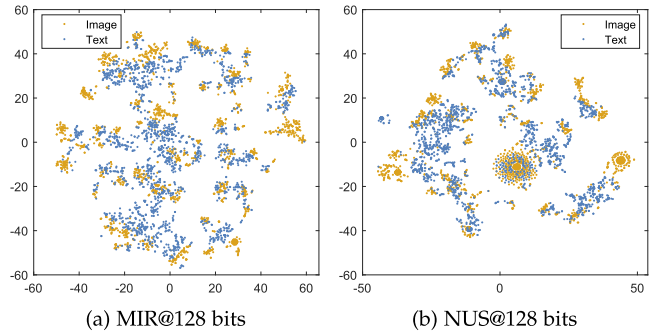


Fig. 11. T-sne visualization of hash codes of image and text modalities on MIRFlickr and NUS-WIDE datasets. The code length is 128.

the activation function $\tanh(\beta\cdot)$ used during training. Although $\lim_{\beta\to+\infty}\tanh(\beta\cdot) = \text{sign}(\cdot)$, the hash codes are still not fully guaranteed to be discrete, and their distances may not strictly fall within the set $\mathcal{D}$. Comparing SCH1 and SCH2, it can be found that this loss oscillation can also be alleviated as long as there is no significant compression of the solution space.

V. Conclusion

In this paper, we have addressed the issue of the space gap in current CMH methods. We have analyzed that the space gap can lead to two main problems: solution space compression and loss function oscillation. To address these problems, we have proposed a novel SCH, where we can exploit the entire Hamming space by classifying pairs of samples into fully semantic-positive samples, partially semantic-positive samples and semantic-negative samples, and assigning them different Hamming distances. In addition, we have also introduced the concept of semantic channel to alleviate the loss function oscillation. The experimental results on three public datasets demonstrated the effectiveness of SCH. Furthermore, we have also designed experiments to demonstrate the impact of the space gap on the current CMH methods, which helps better understand the proposed SCH.

While SCH has achieved significant improvement by bridging the space gap, we have still identified areas for further enhancement: 1) Although we have employed HashNet to obtain discrete hash codes, this does not guarantee the complete discreteness of the learned hash codes, which potentially diminishes the effectiveness of SCH. Minimizing the quantization error of hash codes during the training process may enhance the performance of SCH. 2) SCH primarily addresses the space gap and lacks emphasis on modality alignment, as shown in Fig. 11. Further aligning modalities on the basis of SCH should be beneficial for improving retrieval performance.

REFERENCES

[1] J. Chi and Y. Peng, "Dual adversarial networks for zero-shot cross-media retrieval," in *Proc. Int. Joint Conf. Artif. Intell.*, 2018, pp. 663–669.

[2] L. Zhen, P. Hu, X. Wang, and D. Peng, "Deep supervised cross-modal retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 10394–10403.

[3] X. Liu, Y. He, Y.-M. Cheung, X. Xu, and N. Wang, "Learning relationship-enhanced semantic graph for fine-grained image–text matching," *IEEE Trans. Cybern.*, vol. 54, no. 2, pp. 948–961, Feb. 2024.

[4] V. W. Liang, Y. Zhang, Y. Kwon, S. Yeung, and J. Y. Zou, "Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 17612–17625.

[5] J. C. Pereira et al., "On the role of correlation and abstraction in cross-modal multimedia retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 3, pp. 521–535, Mar. 2014.

[6] F. Feng, X. Wang, and R. Li, "Cross-modal retrieval with correspondence autoencoder," in *Proc. ACM Int. Conf. Multimedia*, 2014, pp. 7–16.

[7] B. Wang, Y. Yang, X. Xu, A. Hanjalic, and H. T. Shen, "Adversarial cross-modal retrieval," in *Proc. ACM Int. Conf. Multimedia*, 2017, pp. 154–162.

[8] S. Chun, S. J. Oh, R. S. de Rezende, Y. Kalantidis, and D. Larlus, "Probabilistic embeddings for cross-modal retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 8415–8424.

[9] A. Andoni and P. Indyk, "Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions," *Commun. ACM*, vol. 51, no. 1, pp. 117–122, 2008.

[10] R. Salakhutdinov and G. Hinton, "Semantic hashing," *Int. J. Approx. Reasoning*, vol. 50, no. 7, pp. 969–978, 2009.

[11] W. Liu, J. Wang, R. Ji, Y.-G. Jiang, and S.-F. Chang, "Supervised hashing with kernels," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 2074–2081.

[12] Y. Shi, X. You, F. Zheng, S. Wang, and Q. Peng, "Equally-guided discriminative hashing for cross-modal retrieval," in *Proc. Int. Joint Conf. Artif. Intell.*, 2019, pp. 4767–4773.

[13] F. Shen, C. Shen, W. Liu, and H. T. Shen, "Supervised discrete hashing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 37–45.

[14] J. Gui, T. Liu, Z. Sun, D. Tao, and T. Tan, "Fast supervised discrete hashing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 2, pp. 490–496, Feb. 2018.

[15] X. Xu, F. Shen, Y. Yang, H. T. Shen, and X. Li, "Learning discriminative binary codes for large-scale cross-modal retrieval," *IEEE Trans. Image Process.*, vol. 26, no. 5, pp. 2494–2507, May 2017.

[16] D. Zhang, X.-J. Wu, T. Xu, and J. Kittler, "WATCH: Two-stage discrete cross-media hashing," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 6, pp. 6461–6474, Jun. 2023.

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

[18] X. Wang, X. Liu, Z. Hu, N. Wang, W. Fan, and J.-X. Du, "Semi-supervised semantic-preserving hashing for efficient cross-modal retrieval," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2019, pp. 1006–1011.

[19] Z. Lin, G. Ding, M. Hu, and J. Wang, "Semantics-preserving hashing for cross-view retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3864–3872.

[20] X. Liu, Z. Hu, H. Ling, and Y.-M. Cheung, "MTFH: A matrix tri-factorization hashing framework for efficient cross-modal retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 3, pp. 964–981, Mar. 2021.

[21] Q.-Y. Jiang and W.-J. Li, "Deep cross-modal hashing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3232–3240.

[22] C. Li, C. Deng, N. Li, W. Liu, X. Gao, and D. Tao, "Self-supervised adversarial hashing networks for cross-modal retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4242–4251.

[23] Z. Zhang, H. Luo, L. Zhu, G. Lu, and H. T. Shen, "Modality-invariant asymmetric networks for cross-modal hashing," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 5, pp. 5091–5104, May 2023.

[24] D. Mandal, K. N. Chaudhury, and S. Biswas, "Generalized semantic preserving hashing for n-label cross-modal retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4076–4084.

[25] E. Yang, D. Yao, T. Liu, and C. Deng, "Mutual quantization for cross-modal search with noisy labels," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 7551–7560.

[26] H. Wu et al., "Contrastive label correlation enhanced unified hashing encoder for cross-modal retrieval," in *Proc. ACM Int. Conf. Inf. Knowl. Manage.*, 2022, pp. 2158–2168.

[27] W. Ou, J. Deng, L. Zhang, J. Gou, and Q. Zhou, "Cross-modal generation and pair correlation alignment hashing," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 3, pp. 3018–3026, Mar. 2023.

[28] J. Yu, H. Zhou, Y. Zhan, and D. Tao, "Deep graph-neighbor coherence preserving network for unsupervised cross-modal hashing," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 4626–4634.

[29] Y. Wang, Z.-D. Chen, X. Luo, and X.-S. Xu, "A high-dimensional sparse hashing framework for cross-modal retrieval," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 12, pp. 8822–8836, Dec. 2022.

[30] H. Hotelling, "Relations between two sets of variates," in *Breakthroughs in Statistics*, Berlin, Germany: Springer, 1992, pp. 162–190.

[31] T. Yao, T. Mei, and C.-W. Ngo, "Learning query and image similarities with ranking canonical correlation analysis," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 28–36.

[32] V. Ranjan, N. Rasiwasia, and C. V. Jawahar, "Multi-label cross-modal retrieval," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 4094–4102.

[33] M. Li, Y.-M. Cheung, and Z. Hu, "Key point sensitive loss for long-tailed visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 4, pp. 4812–4825, Apr. 2023.

[34] W. Lan, Y.-M. Cheung, J. Jiang, Z. Hu, and M. Li, "Compact neural network via stacking hybrid units," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 1, pp. 103–116, Jan. 2024.

[35] Z. Hu, X. Liu, A. Li, B. Zhong, W. Fan, and J. Du, "Efficient cross-modal retrieval via discriminative deep correspondence model," in *Proc. 2nd CCF Chin. Conf. Comput. Vis.*, Tianjin, China, Springer, 2017, pp. 662–673.

[36] Y. Wei et al., "Cross-modal retrieval with CNN visual features: A new baseline," *IEEE Trans. Cybern.*, vol. 47, no. 2, pp. 449–460, Feb. 2017.

[37] I. Goodfellow et al., "Generative adversarial networks," *Commun. ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[38] L. He, X. Xu, H. Lu, Y. Yang, F. Shen, and H. T. Shen, "Unsupervised cross-modal retrieval through adversarial learning," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2017, pp. 1153–1158.

[39] J. Song, Y. Yang, Y. Yang, Z. Huang, and H. T. Shen, "Inter-media hashing for large-scale retrieval from heterogeneous data sources," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2013, pp. 785–796.

[40] M. Long, Y. Cao, J. Wang, and P. S. Yu, "Composite correlation quantization for efficient multimodal retrieval," in *Proc. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2016, pp. 579–588.

[41] H. Liu, R. Ji, Y. Wu, F. Huang, and B. Zhang, "Cross-modality binary code learning via fusion similarity hashing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 7380–7388.

[42] J. Wang, G. Li, P. Pan, and X. Zhao, "Semi-supervised semantic factorization hashing for fast cross-modal retrieval," *Multimedia Tools Appl.*, vol. 76, no. 19, pp. 20197–20215, 2017.

[43] Z. Hu, X. Liu, X. Wang, Y.-M. Cheung, N. Wang, and Y. Chen, "Triplet fusion network hashing for unpaired cross-modal retrieval," in *Proc. ACM Int. Conf. Multimedia Retrieval*, 2019, pp. 141–149.

[44] G. Wu et al., "Unsupervised deep hashing via binary latent factor models for large-scale cross-modal retrieval," in *Proc. Int. Joint Conf. Artif. Intell.*, 2018, pp. 2854–2860.

[45] L. Zhu, L. Cai, J. Song, X. Zhu, C. Zhang, and S. Zhang, "MSSPQ: Multiple semantic structure-preserving quantization for cross-modal retrieval," in *Proc. Int. Conf. Multimedia Retrieval*, 2022, pp. 631–638.

[46] G. Ding, Y. Guo, and J. Zhou, "Collective matrix factorization hashing for multimodal data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 2075–2082.

[47] S. Su, Z. Zhong, and C. Zhang, "Deep joint-semantics reconstructing hashing for large-scale unsupervised cross-modal retrieval," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 3027–3035.

[48] S. Liu, S. Qian, Y. Guan, J. Zhan, and L. Ying, "Joint-modal distribution-based similarity hashing for large-scale unsupervised deep cross-modal retrieval," in *Proc. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2020, pp. 1379–1388.

[49] H. Hu, L. Xie, R. Hong, and Q. Tian, "Creating something from nothing: Unsupervised knowledge distillation for cross-modal hashing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 3123–3132.

[50] P. Hu, H. Zhu, J. Lin, D. Peng, Y.-P. Zhao, and X. Peng, "Unsupervised contrastive cross-modal hashing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 3, pp. 3877–3889, Mar. 2023.

[51] P.-F. Zhang, Y. Li, Z. Huang, and X.-S. Xu, "Aggregation-based graph convolutional hashing for unsupervised cross-modal retrieval," *IEEE Trans. Multimedia*, vol. 24, pp. 466–479, 2021.

[52] Y. Wang, X. Luo, L. Nie, J. Song, W. Zhang, and X.-S. Xu, "BATCH: A scalable asymmetric discrete cross-modal hashing," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 11, pp. 3507–3519, Nov. 2021.

[53] H. Li, C. Zhang, X. Jia, Y. Gao, and C. Chen, "Adaptive label correlation based asymmetric discrete hashing for cross-modal retrieval," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 2, pp. 1185–1199, Feb. 2023.
[54] Y. Wang, Z.-D. Chen, X. Luo, R. Li, and X.-S. Xu, "Fast cross-modal hashing with global and local similarity embedding," *IEEE Trans. Cybern.*, vol. 52, no. 10, pp. 10064–10077, Oct. 2022.
[55] Z. Hu, Y.-M. Cheung, M. Li, W. Lan, D. Zhang, and Q. Liu, "Joint semantic preserving sparse hashing for cross-modal retrieval," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 4, pp. 2989–3002, Apr. 2024.
[56] Z. Hu, Y.-M. Cheung, M. Li, W. Lan, and D. Zhang, "Key points centered sparse hashing for cross-modal retrieval," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2024, pp. 8431–8435.
[57] Y. Wang, Z.-D. Chen, X. Luo, and X.-S. Xu, "High-dimensional sparse cross-modal hashing with fine-grained similarity embedding," in *Proc. Web Conf.*, 2021, pp. 2900–2909.
[58] W. Chen, X. Chen, J. Zhang, and K. Huang, "Beyond triplet loss: A deep quadruplet network for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 403–412.
[59] W. Ge, "Deep metric learning with hierarchical triplet loss," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 269–285.
[60] C. Deng, Z. Chen, X. Liu, X. Gao, and D. Tao, "Triplet-based deep hashing network for cross-modal retrieval," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 3893–3903, Aug. 2018.
[61] W. Gu, X. Gu, J. Gu, B. Li, Z. Xiong, and W. Wang, "Adversary guided asymmetric hashing for cross-modal retrieval," in *Proc. ACM Int. Conf. Multimedia Retrieval*, 2019, pp. 159–167.
[62] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6000–6010.
[63] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Annu. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, 2019, pp. 4171–4186.
[64] A. Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Representations*, 2021, pp. 1–21.
[65] J. Tang, K. Wang, and L. Shao, "Supervised matrix factorization hashing for cross-modal retrieval," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3157–3166, Jul. 2016.
[66] X. Wang, X. Liu, S. Peng, Y.-M. Cheung, Z. Hu, and N. Wang, "Fast semantic preserving hashing for large-scale cross-modal retrieval," in *Proc. IEEE Int. Conf. Data Mining*, 2019, pp. 1348–1353.
[67] Z.-D. Chen, C.-X. Li, X. Luo, L. Nie, W. Zhang, and X.-S. Xu, "SCRATCH: A scalable discrete matrix factorization hashing framework for cross-modal retrieval," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 7, pp. 2262–2275, Jul. 2020.
[68] Z. Cao, M. Long, J. Wang, and P. S. Yu, "HashNet: Deep learning to hash by continuation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 5608–5617.
[69] R. J. W. James and J. T. Buchanan, "A neighbourhood scheme with a compressed solution space for the early/tardy scheduling problem," *Eur. J. Oper. Res.*, vol. 102, no. 3, pp. 513–527, 1997.
[70] E. Dupuis, D. Novo, I. O'Connor, and A. Bosio, "A heuristic exploration of retraining-free weight-sharing for CNN compression," in *Proc. 27th Asia South Pacific Des. Automat. Conf.*, 2022, pp. 134–139.
[71] N. Qian, "On the momentum term in gradient descent learning algorithms," *Neural Netw.*, vol. 12, no. 1, pp. 145–151, 1999.
[72] W. Wu et al., "Batch gradient method with smoothing L1/2 regularization for training of feedforward neural networks," *Neural Netw.*, vol. 50, pp. 72–78, 2014.
[73] Q. Teng, K. Wang, L. Zhang, and J. He, "The layer-wise training convolutional neural networks using local loss for sensor-based human activity recognition," *IEEE Sensors J.*, vol. 20, no. 13, pp. 7265–7274, Jul. 2020.
[74] S. Ruder, "An overview of gradient descent optimization algorithms," 2016, *arXiv:1609.04747*.
[75] G. Goh, "Why momentum really works," *Distill*, vol. 2, no. 4, 2017, Art. no. *e6*.
[76] H. Liu, R. Ji, Y. Wu, and F. Huang, "Ordinal constrained binary code learning for nearest neighbor search," in *Proc. AAAI Conf. Artif. Intell.*, 2017, pp. 1–7.
[77] M. J. Huiskes and M. S. Lew, "The MIR flickr retrieval evaluation," in *Proc. ACM Int. Conf. Multimedia Inf. Retrieval*, 2008, pp. 39–43.
[78] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. Zheng, "NUS-WIDE: A real-world web image database from National University of Singapore," in *Proc. ACM Int. Conf. Image Video Retrieval*, 2009, pp. 1–9.
[79] H. J. Escalante et al., "The segmented and annotated IAPR TC-12 benchmark," *Comput. Vis. Image Understanding*, vol. 114, no. 4, pp. 419–428, 2010.
[80] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," in *Proc. British Mach. Vis. Conf.*, 2014, pp. 1–12.
[81] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Representations*, 2015, pp. 1–14.
[82] S. Kumar and R. Udupa, "Learning hash functions for cross-view similarity search," in *Proc. Int. Joint Conf. Artif. Intell.*, 2011, pp. 1360–1365.
[83] D. Wang, X. Gao, X. Wang, and L. He, "Semantic topic multimodal hashing for cross-media retrieval," in *Proc. Int. Joint Conf. Artif. Intell.*, 2015, pp. 3890–3896.
[84] M. M. Bronstein, A. M. Bronstein, F. Michel, and N. Paragios, "Data fusion through cross-modality metric learning using similarity-sensitive hashing," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 3594–3601.
[85] D. Zhang and W.-J. Li, "Large-scale supervised multimodal hashing with semantic correlation maximization," in *Proc. AAAI Conf. Artif. Intell.*, 2014, pp. 1–7.
[86] X. Liu, Y.-M. Cheung, Z. Hu, Y. He, and B. Zhong, "Adversarial tri-fusion hashing network for imbalanced cross-modal retrieval," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 5, no. 4, pp. 607–619, Aug. 2021.
[87] Y. Cao, M. Long, J. Wang, and P. S. Yu, "Correlation hashing network for efficient cross-modal retrieval," in *Proc. British Mach. Vis. Conf.*, 2017, pp. 1–8.
[88] W. Tan, L. Zhu, J. Li, H. Zhang, and J. Han, "Teacher-student learning: Efficient hierarchical message aggregation hashing for cross-modal retrieval," *IEEE Trans. Multimedia*, vol. 25, pp. 4520–4532, 2022.
[89] X. Li, J. Yu, H. Lu, S. Jiang, Z. Li, and P. Yao, "MAFH: Multilabel aware framework for bit-scalable cross-modal hashing," *Knowl.-Based Syst.*, vol. 279, 2023, Art. no. 110922.
[90] D. Cai, "A revisit of hashing algorithms for approximate nearest neighbor search," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 6, pp. 2337–2348, Jun. 2021.
[91] A. Radford et al., "Learning transferable visual models from natural language supervision," in *Proc. Int. Conf. Mach. Learn.*, PMLR, 2021, pp. 8748–8763.
[92] M. Cherti et al., "Reproducible scaling laws for contrastive language-image learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 2818–2829.

Zhikai Hu received the BSc degree in computer science from China Jiliang University, Hangzhou, China, in 2015, and the MSc degree in computer science from Huaqiao University, Xiamen, China, in 2019. He is currently working toward the PhD degree with the Department of Computer Science, Hong Kong Baptist University, Hong Kong SAR, China, under the supervision of Prof. Yiu-Ming Cheung. His present research interests include multimedia information retrieval and data mining.

Yiu-Ming Cheung (Fellow, IEEE) received the PhD degree from the Department of Computer Science and Engineering, The Chinese University of Hong Kong in Hong Kong. He is a fellow of AAAS, IET, BCS, and AAIA. He is currently a chair professor (Artificial Intelligence) with the Department of Computer Science, Hong Kong Baptist University, Hong Kong SAR, China. His research interests include machine learning, visual computing, data science, pattern recognition, multi-objective optimization, and information security. He is currently the editor-in-chief of *IEEE Transactions on Emerging Topics in Computational Intelligence*. Also, he serves as an associate editor for *IEEE Transactions on Cybernetics*, *IEEE Transactions on Cognitive and Developmental Systems*, *IEEE Transactions on Neural Networks and Learning Systems* (2014–2020), *Pattern Recognition*, *Pattern Recognition Letters*, and *Neurocomputing*, to name a few.

Mengke Li received the BEng degree in communication engineering from Southwest University, Chongqing, China, in 2015, the MSc degree in electronic engineering from Xidian University, Xi'an, China, in 2018, and the PhD degree from the Department of Computer Science, Hong Kong Baptist University, Hong Kong SAR, China, under the supervision of Prof. Yiu-Ming Cheung, in 2022. She is currently an associate researcher with the Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ) and Visual Computing Research Center, Shenzhen University, Guangdong, China. Her current research interests include image restoration and imbalanced data learning.

Weichao Lan received the BEng degree in electronics and information engineering from Sichuan University, Chengdu, China, in 2019. She is currently working toward the PhD degree with the Department of Computer Science, Hong Kong Baptist University, Hong Kong SAR, China, under the supervision of Prof. Yiu-Ming Cheung. Her present research interests include network compression and acceleration, lightweight models.