Unsupervised Cross-Modal Hashing With Modality-Interaction

Rong-Cheng Tu, Jie Jiang, Qinghong Lin, Chengfei Cai, Shangxuan Tian, Hongfa Wang, and Wei Liu, *Fellow, IEEE*

***Abstract*— Recently, numerous unsupervised cross-modal hashing methods have been proposed to deal the image-text retrieval tasks for the unlabeled cross-modal data. However, when these methods learn to generate hash codes, almost all of them lack modality-interaction in the following two aspects: 1) The instance similarity matrix used to guide the hashing networks training is constructed without image-text interaction, which fails to capture the fine-grained cross-modal cues to elaborately characterize the intrinsic semantic similarity among the datapoints. 2) The binary codes used for quantization loss are inferior because they are generated by directly quantizing a simple combination of continuous hash codes from different modalities without the interaction among these continuous hash codes. Such problems will cause the generated hash codes to be of poor quality and degrade the retrieval performance. Hence, in this paper, we propose a novel Unsupervised Cross-modal Hashing with Modality-interaction, termed UCHM. Specifically, by optimizing a novel hash-similarity-friendly loss, a modality-interaction-enabled (MIE) similarity generator is first trained to generate a superior MIE similarity matrix for the training set. Then, the generated MIE similarity matrix is utilized as guiding information to train the deep hashing networks. Furthermore, during the process of training the hashing networks, a novel bit-selection module is proposed to generate high-quality unified binary codes for the quantization loss with the interaction among continuous codes from different modalities, thereby further enhancing the retrieval performance. Extensive experiments on two widely used datasets show that the proposed UCHM outperforms state-of-the-art techniques on cross-modal retrieval tasks.**

***Index Terms*— Cross-modal retrieval, hashing, modality-interaction, bit-selection.**

I. Introduction

RECENTLY, with tremendous amounts of multimedia data such as images and texts being generated, cross-modal search has become a fundamental task, which is widely used in real world applications, such as retrieving images with text description in the Internet, and product search. Cross-modal hashing [1], [2], [3], [4], [5], [6], [7] is one of the cross-modal search techniques, and its core idea is to map high-dimensional cross-modal datapoints into low-dimensional hash codes. Then in the retrieval phase, we can sort the data points by the Hamming distances between their corresponding hash codes with simple bit-wise XOR operations, which is high retrieval efficiency and low storage cost. Thus, cross-modal hashing has arisen increasing research attention.

Existing cross-modal hashing methods can be roughly divided into two categories: supervised and unsupervised methods. The supervised hashing methods [8], [9], [10], [11], [12], [13] usually use the manually-annotated labels to supervise the learning of hash models. Benefiting from the label information, these methods can achieve state-of-the-art retrieval performance. Nevertheless, in real-world scenarios, it is hard or expensive to acquire the manually annotated labels for datapoints. Hence, unsupervised cross-modal hashing methods [1], [14], [15], [16], [17], which learn hashing models with unlabeled data, are appreciated in such scenarios, and then in recent years, much more research interest has arisen in unsupervised cross-modal hashing.

Although plenty of deep unsupervised cross-modal hashing methods [1], [14], [15], [18], [19], [20], [21] have been proposed, most of them lack modality-interaction in generating hash codes for two aspects: 1) Lack modality-interaction when constructing the instance similarity matrix. Most previous cross-modal unsupervised hashing methods [15], [19], [20] usually construct an intra-modal similarity matrix for each modality, and then simply combine these modality-specific similarity matrices based on the image-text matching information to get the final instance similarity matrix without image-text interaction. However, such a way fails to capture the fine-grained cross-modal cues between images and texts, which leads to the poor quality of the generated instance similarity matrix. 2) Inferior binary codes generated for the quantization loss. Most previous deep cross-modal hashing methods usually add a quantization loss $\mathcal{L}_q(\boldsymbol{H}^e, \boldsymbol{C}^e)$ to control the quantization error, where $\boldsymbol{H}^e$ denotes the continuous hash code matrix outputted by a deep hashing network of the datapoints from the e^{th} modality, and $\boldsymbol{C}^e$ denotes the corresponding binary code matrix which $\boldsymbol{H}^e$ should be optimized to. Usually, $\boldsymbol{C}^e$ is cheaply defined as $\boldsymbol{C}^e = sgn(\sum_e \boldsymbol{H}^e)$ [1] or $\boldsymbol{C}^e = sgn(\boldsymbol{H}^{\bar{e}})$ [15], where $\bar{e}$ denotes another modality other than e, and $sgn(\cdot)$ is an element-wise sign function which

Manuscript received 20 October 2022; revised 17 January 2023 and 13 February 2023; accepted 21 February 2023. Date of publication 2 March 2023; date of current version 6 September 2023. This article was recommended by Associate Editor H. Zeng. *(Rong-Cheng Tu and Jie Jiang contributed equally to this work.) (Corresponding author: Wei Liu.)*

Rong-Cheng Tu was with Tencent, Shenzhen 518100, China. He is now with the Department of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China (e-mail: turongcheng@gmail.com).

Jie Jiang, Chengfei Cai, Shangxuan Tian, Hongfa Wang, and Wei Liu are with Tencent Data Platform, Shenzhen, Guangdong 518051, China (e-mail: wl2223@columbia.edu).

Qinghong Lin is with the Electrical and Computer Engineering, National University of Singapore, Singapore 138600.

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSVT.2023.3251395.

Digital Object Identifier 10.1109/TCSVT.2023.3251395

returns 1 if the element is positive and -1 otherwise. It can be found that such ways do not consider the quality of each bit in the continuous hash code $\boldsymbol{H}^e$ of different modalities, which will make the generated $\boldsymbol{C}^e$ inferior, thereby hindering the further improvement of retrieval performance.

Thus, to tackle the aforementioned issues, we propose a novel **U**nsupervised **C**ross-modal **H**ashing with **M**odality-interaction, termed UCHM, which consists of four ingredients, including an MIE similarity generator, a bit-selection module, and two hashing networks. Specifically, by optimizing a novel hash-similarity-friendly loss, the MIE similarity generator is first trained to generate a high-quality MIE similarity matrix for the training set. Then, the generated MIE similarity matrix is utilized to guide the training of the two hashing networks which are used to map datapoints into continuous hash codes. Furthermore, during the training of hashing networks, the bit-selection module is used to generate superior binary codes for the quantization loss by taking the interaction between different-modal continuous hash codes into consideration.

Furthermore, please note, there are some matrix factorization based methods [9], [22], [23], which mainly try to learn semantic information preserved unified codes for both modalities through matrix factorization. In the optimizing process, the data from different modalities may transmit some information to each other, which can be treated as a kind of interaction. But such an interaction is weak. Different from these methods, there are two kinds of modality-interaction in our proposed method. The first one is used to align the visual regions of images and the words of texts to improve the quality of the semantic similarity matrix. The second one can evaluate the quality of every hash bit from different modalities, and then it is exploited to generate unified binary codes for quantization loss, which is different from the matrix factorization based methods.

In a nutshell, the main contributions of UCHM are summarized as follows:

- To the best of our knowledge, the proposed UCHM is the first work in deep unsupervised cross-modal hashing, which introduces modality-interaction to generate superior MIE similarity to train hashing networks.
- Different from the objective function of existing image-text matching methods, we propose a novel hash-similarity-friendly loss to make the generated MIE similarity more suitable for guiding the training of hashing networks.
- To further improve the retrieval performance, a novel bit-selection module is proposed, which exploits the interaction between different-modal continuous hash codes to generate high-quality binary codes for the quantization loss.
- Extensive experiments conducted on two benchmark datasets demonstrate that our proposed UCHM significantly outperforms state-of-the-art unsupervised cross-modal hashing methods.

II. Related Work

Generally, existing cross-modal hashing methods can be roughly divided into two categories: supervised and unsupervised cross-modal hashing methods.

Supervised cross-modal hashing methods [11], [24], [25], [26], [27], [28], [29], [30], [31], [32] mainly use the manually-annotated label information of datapoints to learn hash functions. Benefiting from the label information, these methods usually can achieve impressive retrieval performance. Discrete Cross-modal Hashing (DCH) [33] learns a set of modality-dependence hash projections as well as discriminative binary codes to keep the classification consistent with the label for multi-modal data. Discrete Latent Factor hashing (DLFH) [10] utilizes the discrete latent factor model to efficiently optimizes hash codes without continuous relaxation. Matrix Tri-Factorization Hashing (MTFH) [34] can learn the modality-specific hash codes with different code lengths, while synchronously learning two semantic correlation matrices to semantically correlate the different hash representations for heterogeneous data comparable. Deep cross-modal hashing (DCMH) [35] generates cross-modal similarity-preserving hash codes by minimizing a negative log-likelihood loss. SSAH [36] generates binary hash codes by utilizing two adversarial networks to jointly model different modalities and capture their semantic relevance under the supervision of the learned semantic feature. EGDH [30] first maps the labels into hash codes and then uses them to guide the learning of hash codes for the datapoints through a classification loss. DCPH [8] proposes a margin-dynamic-softmax loss to use the proxy hash codes learned from the class vectors to guide the training of hashing models. DRCH [32] combines both label-level and feature-level information to preserve semantic similarity and proposes a ranking alignment loss function to narrow the inherent gap between modalities.

Unsupervised cross-modal hashing methods [16], [37], [38], [1], [5], [18], [33] learn hash functions from unlabeled data, which allows them to be used in almost all scenarios. Among these existing unsupervised cross-modal hashing methods, some works [22] learn hash functions based on matrix factorization. For instance, Unsupervised Deep Multimodal Hashing (UDCMH) [22] constructs the Laplacian constraint for each modality to learn binary codes with the neighborhood structure of original data preserved. Self-supervised Deep Multimodal Hashing (SSDMH) [23] proposes regularized binary latent model to learn discrete unified binary codes without relaxation and with the weights of different modalities optimized dynamically. Moreover, some works adopt contrastive learning to learn hashing model. For example, Unsupervised Contrastive Cross-modal Hashing (UCCH) [39] proposes a contrastive learning based cross-modal ranking learning loss to exploit the discrimination from all instead of the hardest negative pairs and a novel momentum based binarization optimizer to learn binary hash codes. Unsupervised Cross-modal Contrastive Hashing in Remote Sensing (DUCH) [40] adopts a contrastive objective function that considers both inter- and intra-modal similarities and an adversarial objective function that assists in generating modality-invariant representations to learn hash codes.

Furthermore, these methods mainly focus on designing a more high-quality instance similarity matrix to guide the training of hashing networks. For example, Semantics Preserving Hashing (SePH) [41] first transforms the pre-defined

semantic similarity matrix into a probability distribution and approximates it with the probability distribution derived from learned hash codes by minimizing their Kullback–Leibler divergence. DJSRH [18] fuses the semantic similarities of different modalities into a unified affinity matrix and uses it as the guiding information to train the hashing networks. DGCPN [15] mines the semantic relationships between data and their neighbors through graph-neighbor coherence to construct a similarity matrix and uses it to guide the learning process of hashing networks. However, almost all the instance similarity matrices of these methods are constructed lack of image-text interaction, which may fail to capture the fine-grained cross-modal cues to characterize the semantic similarity among datapoints. Recently, CLIP4Hashing [42] is proposed for video-text retrieval, which adopts the pre-trained CLIP [43] to extract data feature to construct a similarity matrix with dynamic weighting to guide the training of the hashing model. The similarity matrix of this method is based on cosine similarity between the global features of video and text, which still lacks fine-grained interactions between visual region features and word features.

Hence, in this paper, we propose a new hashing method UCHM. UCHM trains a fine-grained image-text interaction network with a novel hashing-similarity-friendly loss to generate a high-quality instance similarity matrix, and it proposes a bit-selection module to generate binary codes for quantization loss through modality-interactions.

III. Proposed Approach

Suppose we have a training set consisting of n instances with image-text pairs, denoted by $\boldsymbol{O} = \{\boldsymbol{o}_i\}_{i=1}^{n}$. For each instance $\boldsymbol{o}_i = (\boldsymbol{x}_i, \boldsymbol{y}_i)$, $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$ denote its image and text datapoints, both of which describe the same content. The goal of deep cross-modal hashing is to learn an image-modality hashing network $\mathcal{F}^v$ and a text-modality hashing network $\mathcal{F}^t$. The image-modality one is used to map each image $\boldsymbol{x}_i$ into continuous hash code $\boldsymbol{h}_i^v = \mathcal{F}^v(\boldsymbol{x}_i; \boldsymbol{W}^v)$, where $\boldsymbol{W}^v$ is the set of parameters in the image-modality hashing network, and then the element-wise sign function $sgn(\cdot)$ is used to get the binary hash code $\boldsymbol{b}_i^v = sgn(\boldsymbol{h}_i^v)$. Similarly, the text-modality hashing network is used to project each text $\boldsymbol{y}_i$ into continuous hash code $\boldsymbol{h}_i^t = \mathcal{F}^t(\boldsymbol{y}_i; \boldsymbol{W}^t)$, where $\boldsymbol{W}^t$ is the set of parameters in the text-modality hashing network. Then the corresponding binary hash code of the text $\boldsymbol{y}_i$ is $\boldsymbol{b}_i^t = sgn(\boldsymbol{h}_i^t)$.

A. Architecture

In this paper, we propose a novel Unsupervised Cross-modal Hashing with Modality-interaction (UCHM), whose architecture is shown in Figure 1. The UCHM mainly consists of four parts, including a modality-interaction-enabled (MIE) similarity generator, a bit-selection module, and two modality-specific hashing networks.

Specifically, the core of MIE similarity generator is an image-text interaction network whose architecture is the same as that of image-text matching network BFAN [44]. The goal of image-text interaction network is used to capture the fine-grained cross-modal cues well to elaborately characterize the intrinsic cross-modal similarities between images and texts, and then use the cross-modal similarities to construct the high-quality MIE similarities among the datapoints.

The core of bit-selection module is a binary code generating algorithm, which takes the interaction between continuous hash codes of different modality into consideration to generate superior binary codes. The generated binary codes are used to construct quantization loss to further improve the performance of cross-modal retrieval tasks.

The image-modality hashing network is the VGG19 [45] with the last layer replaced by a k-dimensional fully-connection layer, where k denotes the length of hash codes. The goal of this network is to map image $\boldsymbol{x}_i$ into continuous hash code $\boldsymbol{h}_i^v = \mathcal{F}^v(\boldsymbol{x}_i; \boldsymbol{W}^v)$, where $\boldsymbol{W}^v$ is the set of parameters in the image modality-specific hashing network, and then an element-wise sign function $sgn(\cdot)$, which returns 1 if the element is positive and returns -1 otherwise, is used to get the binary hash $\boldsymbol{b}_i^v = sgn(\boldsymbol{h}_i^v)$.

The text-modality hashing network is a two-layer Multilayer Perceptron (MLP). The first fc layer in MLP has 4096 units with the ReLU [46] as activation function, and the second fc layer has k units, where k denotes the length of hash codes. The target of the text-modality hashing network is to project text $\boldsymbol{y}_i$ into continuous hash code $\boldsymbol{h}_i^t = \mathcal{F}^t(\boldsymbol{y}_i; \boldsymbol{W}^t)$, where $\boldsymbol{W}^t$ is the set of parameters in the text-modality hashing network, and then followed by a sign function $sgn(\cdot)$, the corresponding binary hash code is $\boldsymbol{b}_i^t = sgn(\boldsymbol{h}_i^t)$.

In the following, we first introduce the MIE similarity generator in detail in subsection III-B. Then, in subsection III-C, we show how to use the generated similarity to train hashing networks and how the bit-selection module generates high-quality binary codes for the quantization loss.

B. MIE Similarity Generator

The core of the MIE similarity generator is an image-text interaction network, which is used to generate cross-modal similarities of image-text pairs by capturing the fine-grained cross-modal cues between images and texts. Moreover, in the image-text matching area [44], [47], [48], [49], [50], and [51], how to capture the fine-grained cross-modal cues has been widely studied in cross-modal attention based image-text matching (CAIM) networks. For example, BFAN [44] proposed a focal attention mechanism to eliminate irrelevant fragments from shared semantics. CASC [49] proposes a joint framework that performs cross-modal attention for local alignment and multi-label prediction for global semantic consistence. URL [50] designs a weight-sharing network for learning the implicit interaction between image and text.

Furthermore, because the goal of our method is to define a high-quality similarity matrix as guiding information through modality-interactions but not to focus on designing a novel modality-interaction mechanism. Thus, in our paper, we directly chose BFAN [44] as our image-text interaction network. Nevertheless, it is not appropriate to train our image-text interaction network directly with the original objective function of image-text matching methods, which will be analyzed in the following subsection. To this end, we propose

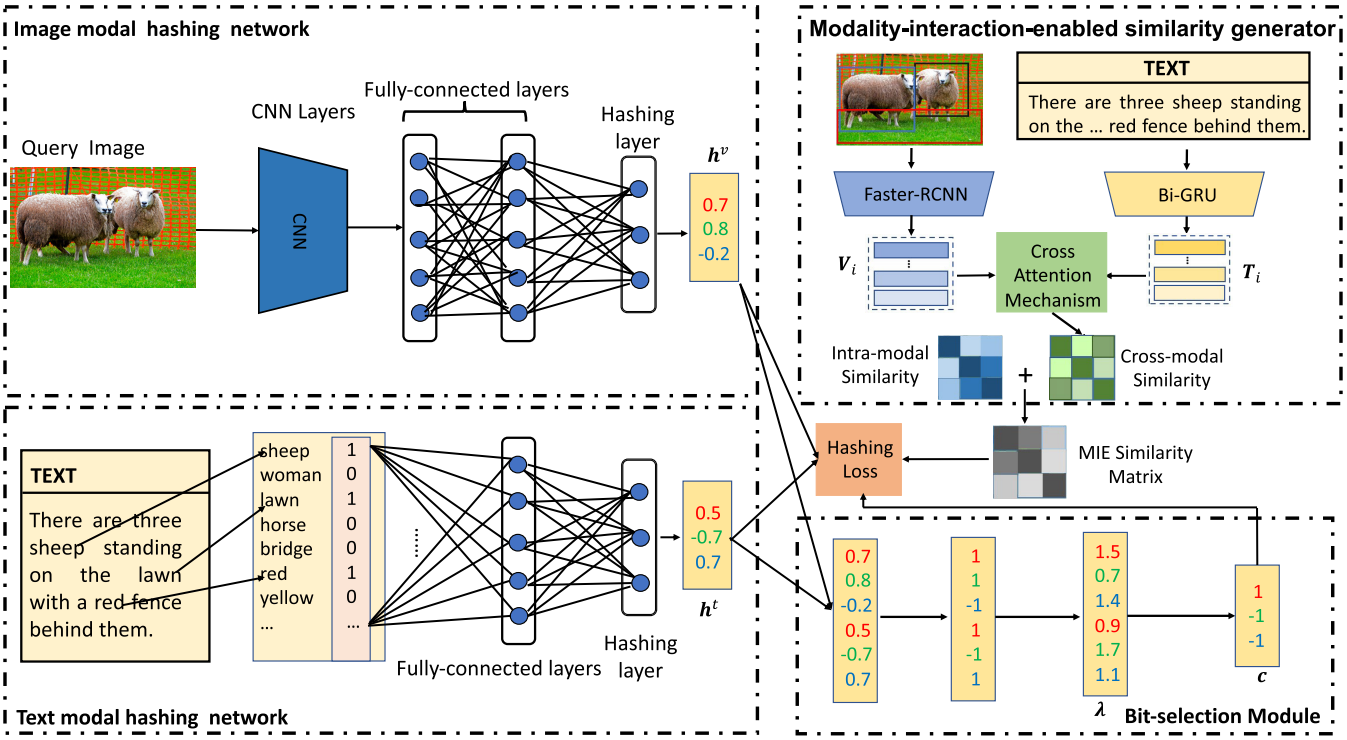


Fig. 1. The architecture of our proposed UCHM.

a novel hash-similarity-friendly loss to optimize our image-text interaction network.

Next, we show the general pipeline of CAIM networks, and then introduce the hash-similarity-friendly loss. Finally, we show how to generate the MIE similarity matrix.

1) General Pipeline of CAIM Networks: First, the CAIM networks process the images and texts into the region-level visual features and the word-level textual features, respectively. Specifically, to obtain the region-level visual features $\boldsymbol{V}_i = [\boldsymbol{v}_1^i, \cdots, \boldsymbol{v}_m^i]$ of image $\boldsymbol{x}_i$, the Faster R-CNN [52] model, which is pre-trained on the Visual Genomes dataset [53] by [54], is used to extract the top m region proposals of $\boldsymbol{x}_i$. Then, by average-pooling the spatial feature map, a feature vector $\boldsymbol{v}_j^{i\prime} \in \mathcal{R}^{2048}$ for the j^{th} region proposal is calculated. Finally, through a linear projection layer, d-dimentional region features are obtained:

$$\boldsymbol{v}_j^i = \boldsymbol{W}_v \boldsymbol{v}_j^{i\prime} + \boldsymbol{b}_v, \tag{1}$$

where $\boldsymbol{W}_v$ and $\boldsymbol{b}_v$ are to-be-learned parameters, and $\boldsymbol{v}_j^i$ is the visual feature for the j^{th} region of the image $\boldsymbol{x}_i$.

To obtain the word-level textual features $\boldsymbol{T}_h = [\boldsymbol{t}_1^j, \cdots, \boldsymbol{t}_n^j]$ of the text $\boldsymbol{y}_j$ with n words, the i^{th} word $\boldsymbol{w}_i^j$ of the text $\boldsymbol{y}_j$ is first embed into a 300-dimensional vector $\boldsymbol{t}_i^{j\prime}$ by a linear layer. Then, to enhance the word-level features with sufficient context information, a bi-directional GRU [55] with d-dimensional hidden units is used to extract information from both forward and backward directions:

$$\overrightarrow{\boldsymbol{e}_i^j} = \overrightarrow{GRU}(\overrightarrow{\boldsymbol{e}_{i-1}^j}, \boldsymbol{t}_i^{j\prime}), \quad \overleftarrow{\boldsymbol{e}_i^j} = \overleftarrow{GRU}(\overleftarrow{\boldsymbol{e}_{i-1}^j}, \boldsymbol{t}_i^{j\prime}), \tag{2}$$

where $\overrightarrow{\boldsymbol{e}_i^j}$ and $\overleftarrow{\boldsymbol{e}_i^j}$ denote hidden states from forward GRU and backward GRU, respectively. Then, the textual feature of the i^{th} word $\boldsymbol{w}_i^j$ in the text $\boldsymbol{y}_j$ is defined as $\boldsymbol{t}_i^j = \frac{\overrightarrow{\boldsymbol{e}_i^j} + \overleftarrow{\boldsymbol{e}_i^j}}{2}$.

Then to calculate the similarity between $\boldsymbol{x}_i$ and $\boldsymbol{y}_j$, variant cross-attention mechanisms are proposed to associate shared semantics between the region-level features $\boldsymbol{V}_i$ and the word-level features $\boldsymbol{T}_j$, which are uniformly formulated as:

$$\boldsymbol{s}_{ij} = \mathcal{F}_{att}(\boldsymbol{V}_i, \boldsymbol{T}_j; \boldsymbol{W}), \tag{3}$$

where $\boldsymbol{s}_{ij}$ denotes semantic similarity between the image $\boldsymbol{x}_i$ and the text $\boldsymbol{y}_j$; $\mathcal{F}_{att}(\cdot; \boldsymbol{W})$ denotes the cross-modal attention mechanism and $\boldsymbol{W}$ is the set of learnable parameters. For example, in our settings, $\mathcal{F}_{att}(\cdot; \boldsymbol{W})$ denotes the focal attention mechanism proposed by BFAN [44].

Finally, for a query, to make the semantic similarity score between it and its corresponding matching datapoint larger than the scores between it and its mismatching datapoints, a hinge-based triplet ranking loss with emphasis on the hard negatives [56] is used as the loss function. Specifically, given a matching image-text pair $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$, $\bar{i}$ denotes the index of the hardest negative datapoint when using the text (image) to match images (texts), then the ranking loss is as follows:

$$\mathcal{L}_m = \sum_i ([mg - \boldsymbol{s}_{ii} + \boldsymbol{s}_{i\bar{i}}]_+ + [mg - \boldsymbol{s}_{ii} + \boldsymbol{s}_{\bar{i}i}]_+), \tag{4}$$

where $[a]_+ = max(0, a)$; mg is a pre-defined margin which is usually defined as 0.2 [44], [51]. Although Eq.(4) does not require the calculated similarity of dissimilar (similar) image-text pairs to be as small (large) as possible, it can make the similarity $\boldsymbol{s}_{ii}$ of an image-text pair from the same instance, i.e., the matching pair, larger than the mismatching pairs. Because

the goal of image-text matching task is to find the matching image (text) for a query text (image). So, Eq.(4) is suitable as the objective function for the image-text matching task.

However, as the definition of similarity between datapoints in the image-text matching area is dissimilar to that of cross-modal hashing, directly using Eq.(4) to train our image-text interaction network is not good enough. In the image-text matching area, an image $\boldsymbol{x}_i$ (text $\boldsymbol{y}_i$) is only similar to the text $\boldsymbol{y}_i$ (image $\boldsymbol{x}_i$), i.e., they both belong to the same instance $\boldsymbol{o}_i$, while in the cross-modal hashing area, an image $\boldsymbol{x}_i$ (a text $\boldsymbol{y}_i$) is similar to all the texts (images) that share at least one common label with $\boldsymbol{x}_i$ ($\boldsymbol{y}_i$), and the similarities of similar (dissimilar) pairs should be large (small). Then, to meet the learning objective of the hashing networks, the similarities calculated by our image-text interaction network should be as large (small) as possible for similar (dissimilar) pairs, which means that it is not good enough to train our image-text interaction network by directly using Eq.(4) as the objective function. Therefore, we propose a hash-similarity-friendly loss to train our image-text interaction network.

2) Hash-Similarity-Friendly Loss: To make the similarity generated by our image-text interaction network more suitable for hashing, we propose a novel hash-similarity-friendly loss, which is formulated as:

$$\mathcal{L}_s = \mathcal{L}_m + \alpha \sum_i |s_{ii} - 1| + \beta \sum_{\{i,j\}\in\Psi} (1 - \boldsymbol{s}_{ij}) \cdot max(0, \boldsymbol{s}_{ij}), \tag{5}$$

where $|\cdot|$ denotes l_1 norm; Ψ denotes a set of $\eta\%$ image-text pairs randomly sampled from the current mini-batch.

The first item is the ranking loss Eq.(4) which optimizes the network to capture the fine-grained cross-modal semantic similarity among the datapoints. Moreover, as the similarity of similar image-text pair is defined as 1 and those of dissimilar pairs are defined as 0 in the hashing area, minimizing the second item is to make the similarities of image-text pairs from the same instances as large as possible.

Furthermore, the third item is adopted to make the similarity of dissimilar image-text pairs as small as possible for the following reason. As there is no label information for the datapoints, it is unknown whether the image-text pairs from different instances are similar or not. However, for an image (text), the number of similar texts (images) is much smaller than that of dissimilar texts (images) [36]. It means that if we randomly select an image and a text, then this data pair may be dissimilar with a high probability. Based on that, we randomly select $\eta\%$ image-text pairs, denoted as a set Ψ, from each mini-batch as "dissimilar" pairs, then we minimize $max(0, \boldsymbol{s}_{ij})$ to make the similarity of each selected image-text pair as small as possible. In addition, as some data pairs in the set Ψ may be similar data pair, then in the third item of Eq.(5), we further multiply $max(0, \boldsymbol{s}_{ij})$ by a weight $1 - \boldsymbol{s}_{ij}$ to make the loss function more robust. When $\boldsymbol{s}_{ij}$ is large, the weight will be small, i.e., the third item will slightly penalize on such "dissimilar" data pair as it maybe a similar pair. Otherwise, when $\boldsymbol{s}_{ij}$ is small, $1-\boldsymbol{s}_{ij}$ will be large, then the third item will penalize on such "dissimilar" data pair to make $\boldsymbol{s}_{ij}$ as small as possible.

Thus, by minimizing Eq. (5), it will make the similarities generated by the image-text interaction network more suitable for the training of our hashing network, then we called the Eq. 5 as hash-similarity-friendly loss.

3) MIE Similarity Generating: After optimizing our image-text interaction network by taking Eq.(5) as its objective function, for an image $\boldsymbol{x}_i$ and a text $\boldsymbol{y}_j$, their corresponding cross-modal similarity can be calculated as $\boldsymbol{s}_{ij} = \mathcal{F}_{att}(\boldsymbol{V}_i, \boldsymbol{T}_j, \boldsymbol{W})$. Then, we can get the cross-modal similarity matrix $\boldsymbol{S}$ of the training set, where $\boldsymbol{s}_{ij}$ is the i^{th} row j^{th} column element of $\boldsymbol{S}$. Moreover, to ensure that the elements on the diagonal of $\boldsymbol{S}$ are 1 and each element of $\boldsymbol{S}$ is in [0, 1], the matrix $\boldsymbol{S}$ needs to be further transformed into $\boldsymbol{Q}$ through the following way:

$$\boldsymbol{q}_{ij} = min(1, max(0, \boldsymbol{s}_{ij}/\boldsymbol{s}_{ii})). \tag{6}$$

Now, we can directly use the similarity matrix $\boldsymbol{Q}$ as guiding information to train the hashing networks. However, as each $\boldsymbol{q}_{ij}$ is a cross-modal similarity mainly containing the inter-modal information, the intra-modal similarity, which may contain some useful intra-modal information, is missed. Considering that almost all the instance similarity matrices in existing deep cross-modal unsupervised hashing methods are constructed by combining the intra-modal similarity matrices, we can choose one of them as the intra-modal similarity and incorporate it into cross-modal similarity matrix $\boldsymbol{Q}$ to generate the final MIE similarity matrix. Specifically, in our method, we chose the instance similarity matrix $\boldsymbol{G}$ calculated by DGCPN [15], and then our MIE similarity matrix $\boldsymbol{A}$ is formulated as follows:

$$\boldsymbol{A} = \gamma \boldsymbol{Q} + (1 - \gamma)\boldsymbol{G}, \tag{7}$$

where $\gamma \in [0, 1]$ is a hyper-parameter. Furthermore, as the similarities of hash codes are in the range from -1 to 1, then we further let $\boldsymbol{A} = 2\boldsymbol{A} - 1$ to make each element $\boldsymbol{a}_{ij}$ of $\boldsymbol{A}$ in the range of $[-1, 1]$. Moreover, the closer value of $\boldsymbol{a}_{ij}$ is to 1, the larger similarity between $\boldsymbol{x}_i$ ($\boldsymbol{y}_j$) and $\boldsymbol{y}_i$ ($\boldsymbol{x}_j$) is.

C. Learning to Hash

Now, with the MIE similarity matrix $\boldsymbol{A}$ as guiding information, the image- and text-modality hashing networks are trained to project the images $\boldsymbol{X} = \{\boldsymbol{x}_i\}_{i=1}^n$ and texts $\boldsymbol{Y} = \{\boldsymbol{y}_i\}_{i=1}^n$ into continuous hash codes $\boldsymbol{H}^v = \{\boldsymbol{h}_i^v\} \in [-1, 1]^{k\times n}$ and $\boldsymbol{H}^t = \{\boldsymbol{h}_i^t\}_{i=1}^n \in [-1, 1]^{k\times n}$, respectively. k is the length of hash codes. Then, followed by a $sgn(\cdot)$ function, we can get their corresponding binary hash codes $\boldsymbol{B}^v = sgn(\boldsymbol{H}^v) = \{\boldsymbol{b}_i^v\}_{i=1}^n \in \{-1, 1\}^{k\times n}$ and $\boldsymbol{B}^t = sgn(\boldsymbol{H}^t) = \{\boldsymbol{b}_i^t\}_{i=1}^n \in \{-1, 1\}^{k\times n}$.

Similar to [19] and [15], the defined objective function should consider three aspects: (1) The co-occurrence information in data should be used well. For an image $\boldsymbol{x}_i$ and its corresponding text $\boldsymbol{y}_i$, as they are from the same instance $\boldsymbol{o}_i$, the semantic information contained in them is the same, then their corresponding hash codes $\boldsymbol{b}_i^v$ and $\boldsymbol{b}_i^t$ should be the same, i.e., the similarity between $\boldsymbol{b}_i^v$ and $\boldsymbol{b}_i^t$ should be 1. (2) The generated hash codes need to preserve the MIE similarity $\boldsymbol{A}$. (3) The intra- and inter-modality consistency should be

preserved well by the generated hash codes. In other words, for two instances $\boldsymbol{o}_i = \{\boldsymbol{x}_i, \boldsymbol{y}_i\}$ and $\boldsymbol{o}_j = \{\boldsymbol{x}_j, \boldsymbol{y}_j\}$, as the semantic similarities between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, $\boldsymbol{x}_i$ and $\boldsymbol{y}_j$, and $\boldsymbol{y}_i$ and $\boldsymbol{y}_j$ are consistent, then the similarities between $\boldsymbol{b}_i^v$ and $\boldsymbol{b}_j^v$, $\boldsymbol{b}_i^v$ and $\boldsymbol{b}_j^t$, and $\boldsymbol{b}_i^t$ and $\boldsymbol{b}_j^t$ should be consistent.

Hence, the objective function consists of three losses: 1) co-occurrence similarity preserving loss $\mathcal{L}_c(\boldsymbol{B}^v, \boldsymbol{B}^t)$; 2) MIE similarity preserving loss $\mathcal{L}_a(\boldsymbol{B}^v, \boldsymbol{B}^t)$; 3) intra- and inter-modality consistency preserving loss $\mathcal{L}_i(\boldsymbol{B}^v, \boldsymbol{B}^t)$, which are defined as follows:

$$\mathcal{L}_c(\boldsymbol{B}^v, \boldsymbol{B}^t) = \left\| \Gamma(\boldsymbol{\Phi}(\boldsymbol{B}^v, \boldsymbol{B}^t)) - 1.5\boldsymbol{I} \right\|_F, \tag{8}$$

$$\mathcal{L}_a(\boldsymbol{B}^v, \boldsymbol{B}^t) = \sum_{e,u} \left\| \boldsymbol{\Phi}(\boldsymbol{B}^e, \boldsymbol{B}^u) - \boldsymbol{A} \right\|_F, \tag{9}$$

$$\mathcal{L}_i(\boldsymbol{B}^v, \boldsymbol{B}^t) = \sum_{e,u,e_1,u_1} \left\| \boldsymbol{\Phi}(\boldsymbol{B}^e, \boldsymbol{B}^u) - \boldsymbol{\Phi}(\boldsymbol{B}^{e_1}, \boldsymbol{B}^{u_1}) \right\|_F, \tag{10}$$

where $e, u, e_1, u_1 \in \{v, t\}$; $\Gamma(\cdot)$ denotes the vector constructed by the diagonal elements of the matrix; $\boldsymbol{I}$ is a vector whose elements are 1, and followed [19], we use 1.5 as the optimization goal of $\mathcal{L}_c(\boldsymbol{B}^v, \boldsymbol{B}^t)$ that slightly improves performance; $\boldsymbol{\Phi}(\boldsymbol{B}^e, \boldsymbol{B}^u)$ denotes the cosine similarity matrix, which is calculated based on the hash codes $\boldsymbol{B}^e$ and $\boldsymbol{B}^u$, and each element in the similarity matrix is defined as:

$$\boldsymbol{\Phi}(\boldsymbol{B}^e, \boldsymbol{B}^u)_{ij} = \frac{\boldsymbol{b}_i^{e\top} \boldsymbol{b}_j^u}{\left\| \boldsymbol{b}_i^e \right\|_2 \cdot \left\| \boldsymbol{b}_j^u \right\|_2}, \tag{11}$$

where $\boldsymbol{\Phi}(\boldsymbol{B}^e, \boldsymbol{B}^u)_{ij}$ denotes the i^{th} row and j^{th} column element in the matrix $\boldsymbol{\Phi}(\boldsymbol{B}^e, \boldsymbol{B}^u)$; $\boldsymbol{b}_i^e$, which is the i^{th} column of $\boldsymbol{B}^e$, denotes the hash code of the e modality datapoint in the i^{th} instance $\boldsymbol{o}_i$.

Then the objective function of hashing networks is:

$$\mathcal{L}_1 = \mathcal{L}_c(\boldsymbol{B}^v, \boldsymbol{B}^t) + \mathcal{L}_a(\boldsymbol{B}^v, \boldsymbol{B}^t) + \mathcal{L}_i(\boldsymbol{B}^v, \boldsymbol{B}^t). \tag{12}$$

However, as $\boldsymbol{B}^v = sgn(\boldsymbol{H}^v)$ and $\boldsymbol{B}^t = sgn(\boldsymbol{H}^t)$, where the $sgn(\cdot)$ function is non-differentiable at zero and its derivation will be zero for a non-zero input, then the parameters of hashing networks cannot be updated with the back-propagation algorithm when minimizing the loss function $\mathcal{L}_1$. To tackle this problem, a common way is to directly discard the $sgn(\cdot)$ function and add a quantization loss to make each element of the output of hashing networks close to "+1" or "−1", which is formulated as follows:

$$\begin{aligned}\mathcal{L} = {} & \mathcal{L}_c(\boldsymbol{H}^v, \boldsymbol{H}^t) + \mathcal{L}_a(\boldsymbol{H}^v, \boldsymbol{H}^t) \\ & + \mathcal{L}_i(\boldsymbol{H}^v, \boldsymbol{H}^t) + \sum_{e\in\{v,t\}} \mathcal{L}_q(\boldsymbol{H}^e, \boldsymbol{C}^e),\end{aligned} \tag{13}$$

where $\mathcal{L}_q$ is a quantization loss and $\boldsymbol{C}^e$ denotes the binary codes which $\boldsymbol{H}^e$ should be optimized to. For the quantization loss $\mathcal{L}_q$, there are some variants. For example, in some previous works [1], [35], the quantization loss is defined as $\mathcal{L}_q = \left\| \boldsymbol{H}^e - \boldsymbol{C}^e \right\|_F^2$. Moreover, Yu et al. [15] proposed a half-real and half-binary optimization strategy whose quantization loss is $\mathcal{L}_q = \sum_{e\in\{v,t\}} (\mathcal{L}_c(\boldsymbol{H}^e, \boldsymbol{C}^e) + \mathcal{L}_a(\boldsymbol{H}^e, \boldsymbol{C}^e) + \mathcal{L}_i(\boldsymbol{H}^e, \boldsymbol{C}^e))$, and in our method, we chose such loss as our quantization loss.

Although many kinds of quantization loss functions have been proposed, the binary codes $\boldsymbol{C}^v$ and $\boldsymbol{C}^t$ are simply generated by binarizing the continuous hash codes $\boldsymbol{H}^v$ and $\boldsymbol{H}^t$, such as $\boldsymbol{C}^v = \boldsymbol{C}^t = sgn(\boldsymbol{H}^v + \boldsymbol{H}^t)$ [1], or $\boldsymbol{C}^t = sgn(\boldsymbol{H}^v)$ and $\boldsymbol{C}^v = sgn(\boldsymbol{H}^t)$ [15]. However, such ways do not consider the quality of each bit in the continuous hash codes $\boldsymbol{H}^v$ and $\boldsymbol{H}^t$ through modality-interaction, which will lead to the poor quality of the generated binary codes $\boldsymbol{C}^v$ and $\boldsymbol{C}^t$, thus hindering the improvement of retrieval performance. Take the first way as an example, and suppose there is an instance $\boldsymbol{o}_i$. The continuous hash code of its image $\boldsymbol{x}_i$ is $\boldsymbol{h}_i^v = (0.7, 0.8, -0.2)$, and the continuous hash code of its text $\boldsymbol{y}_i$ is $\boldsymbol{h}_i^t = (0.5, -0.7, 0.7)$. Then the binary code $\boldsymbol{c}_i$ used for quantization is $(1, 1, 1)$. While if the quality of the second bit of text hash codes is better than that of image hash codes, and the quality of the third bit of image hash codes is better than that of text hash codes, then the binary code $\boldsymbol{c}_i$ will be better to become $(1, -1, -1)$.

To tackle the above problem, we propose a novel bit-selection module, which takes the quality of each bit in continuous hash codes $\boldsymbol{H}^v$ and $\boldsymbol{H}^t$ into consideration through modality-interaction, to generate superior unified binary codes $\boldsymbol{C}^v = \boldsymbol{C}^t = \boldsymbol{C}$ for the quantization loss.

Bit-Selection Module: In the following, we first introduce how to distinguish the quality of hash codes, and then introduce how to distinguish the quality of each bit in the hash codes.

Given the hash codes $\boldsymbol{B} = \{\boldsymbol{b}_i\}_{i=1}^n \in \{-1, 1\}^{k\times n}$ and the similarity matrix $\boldsymbol{S} \in [-1, 1]^{n\times n}$ that the hash codes $\boldsymbol{B}$ should preserve, then the smaller value of $\mathcal{J}$ means that the hash codes $\boldsymbol{B}$ preserve the similarity defined in $\boldsymbol{S}$ better, i.e., the quality of hash codes $\boldsymbol{B}$ are higher. The $\mathcal{J}$ is defined as follows:

$$\mathcal{J} = \left\| \boldsymbol{\Phi}(\boldsymbol{B}, \boldsymbol{B}) - \boldsymbol{S} \right\|_F^2 = \sum_{ij} (\frac{1}{k} \boldsymbol{b}_i^\top \boldsymbol{b}_j - \boldsymbol{S}_{ij})^2 \tag{14}$$

where S_{ij} denotes the ith row and jth column of $\boldsymbol{S}$, i.e., the similarity between the ith and the jth datapoint.

In the Formula 14, when calculating the similarity between hash codes, i.e., the value of $\frac{1}{k}\boldsymbol{b}_i^\top \boldsymbol{b}_j$, it assigns the same weight to each bit. However, as the quality of each bit is different, if we pay a larger weight to the high-quality bits and a smaller weight to the low-quality bits to calculate the similarity of hash codes, then the calculated value of $\mathcal{J}$ will be smaller. Hence, based on the above insight, we add a weight λ_z to the z^{th} of hash codes when calculating the similarity between hash codes, i.e., $\frac{1}{k}\sum_{z=1}^{k} \lambda_z \boldsymbol{b}_{iz}^\top \boldsymbol{b}_{jz}$, and then get a novel optimization function, which can be formulated as follows:

$$\begin{aligned}\min_{\Lambda} \mathcal{J} &= \sum_{ij} (\frac{1}{k} \sum_{z=1}^{k} \lambda_z \boldsymbol{b}_{iz}^\top \boldsymbol{b}_{jz} - \boldsymbol{S}_{ij})^2 \\ &= \left\| \frac{1}{k} \boldsymbol{B}^\top \Lambda \boldsymbol{B} - \boldsymbol{S} \right\|_F^2\end{aligned} \tag{15}$$

where $\Lambda = diag(\boldsymbol{\lambda})$ is a diagonal matrix whose diagonal elements are $\lambda_1, \lambda_2, \ldots, \lambda_k$; the λ_z denotes the z^{th} element of $\boldsymbol{\lambda}$. After minimizing the Formula (15) with $\boldsymbol{B}$ and $\boldsymbol{S}$ fixed, the larger the λ_z is, the more attention is paid in the z^{th} bit of

Algorithm 1 Learning Algorithm for UCHM

Require: Instances $\boldsymbol{O}$, the length of hash codes k.

Ensure: Parameters of image-text interaction network $\boldsymbol{W}$, parameters of image and text hashing networks $\boldsymbol{W}^v$ and $\boldsymbol{W}^t$, image and text hash codes $\boldsymbol{B}^v$ and $\boldsymbol{B}^t$.

Initialize parameters: $\boldsymbol{W}$, $\boldsymbol{W}^v$, $\boldsymbol{W}^t$, η, α, β γ. Learning rate: lr, iteration number: T, mini-batch size t.
repeat
  for $j = 1 : \frac{n}{t}$ **do**
    Randomly sample t image-text pairs from database as a mini-batch.
    Calculate the semantic similarity $\boldsymbol{s}_{ij}$ between image $\boldsymbol{x}_i$ and text $\boldsymbol{y}_i$ by the image-text interaction network.
    Update parameters $\boldsymbol{W}$ of the image-text interaction network by minimizing Formula (5).
  end for
until Convergence
Generate the MIE similarity matrix $\boldsymbol{A}$ through Formula (7) with the trained well image-text interaction network.
repeat
  for $j = 1 : \frac{n}{t}$ **do**
    Randomly sample t image-text pairs from database as a mini-batch.
    Generate continuous hash code $\boldsymbol{h}_i^v$ with image $\boldsymbol{x}_i$ as input by image hash network.
    Generate continuous hash code $\boldsymbol{h}_i^t$ with text $\boldsymbol{y}_i$ as input by text hash network.
    Generate binary hash code $\boldsymbol{c}_i$ with based on the continuous hash codes $\boldsymbol{h}_i^v$ and $\boldsymbol{h}_i^t$ through the bit-selection module.
    Update parameters of hash networks $\boldsymbol{W}^v$ and $\boldsymbol{W}^t$ by minimizing Formula (13).
  end for
until Convergence
Generate image hash codes $\boldsymbol{B}^v$ and $\boldsymbol{B}^t$.

hash codes, i.e., the higher quality the z^{th} bit of hash codes is. Hence, the λ_z can be used to denote the quality of z^{th} bit of hash codes. Moreover, the optimal solution of Eq.(15) is:

$$\boldsymbol{\lambda} = k[(\boldsymbol{B}\boldsymbol{B}^\top) \circ (\boldsymbol{B}\boldsymbol{B}^\top)]^{-1}\boldsymbol{l}, \tag{16}$$

where $\circ$ denotes element-wise multiplication, and $\boldsymbol{l}$ is a k-dimensional vector constructed by the diagonal elements of the matrix $\boldsymbol{B}\boldsymbol{S}\boldsymbol{B}^\top$.

Hence, inspired by the above insight, the bit-selection module generates the unified binary codes $\boldsymbol{C}$ for the continuous hash codes $\boldsymbol{H}^v$ and $\boldsymbol{H}^t$ through modality-interaction with the following steps:

(1) We concatenate the continuous hash codes $\boldsymbol{H}^v$ and $\boldsymbol{H}^t$ followed a $sgn(\cdot)$ function to get binary hash code $\boldsymbol{D} = sgn([\boldsymbol{H}^{v\top}; \boldsymbol{H}^{t\top}]^\top) \in \{-1, 1\}^{2k\times n}$;
(2) We take the $\boldsymbol{D}$ and MIE similarity $\boldsymbol{A}$ as $\boldsymbol{B}$ and $\boldsymbol{S}$ in Eq.(15), and obtain the vector $\boldsymbol{\lambda} \in \mathcal{R}^{2k}$ through Eq.(16), where the larger λ_z, the higher quality of the z^{th} bit of $\boldsymbol{D}$ is;
(3) A bit-selection vector $\boldsymbol{r} \in \{0, 1\}^k$ can be generated. Specifically, when $i \in \{1, \cdots, k\}, \lambda_i \leq \lambda_{i+k}$, it means that the quality of the i^{th} bit in $\boldsymbol{H}^t$ is better than that in $\boldsymbol{H}^v$. Then we set $\boldsymbol{r}_i = 0$ denoting that we generate the i^{th} bit of $\boldsymbol{C}$ according to that of $\boldsymbol{H}^v$, otherwise we set $\boldsymbol{r}_i = 1$;
(4) Finally, we obtain the unified binary codes $\boldsymbol{C} = (\boldsymbol{r}\boldsymbol{1}^\top) \circ sgn(\boldsymbol{H}^v) + (1 - \boldsymbol{r}\boldsymbol{1}^\top) \circ sgn(\boldsymbol{H}^t)$, where $\boldsymbol{1}$ is an n-dimensional vector whose elements are 1.

Through this way, the superior unified binary codes $\boldsymbol{C}^v = \boldsymbol{C}^t = \boldsymbol{C}$ can be generated to construct the quantization loss to further improve the retrieval performance of our hash models. Moreover, the details of the algorithm are shown in Algorithm 1.

IV. Experiments

A. Datasets and Evaluation

1) Datasets: We conduct experiments on the ***MS COCO*** [58] and ***IAPR TC12*** [59] datasets. Specifically, for the MS COCO dataset, There are 122,218 image-text pairs in it, and all these data pairs belong to 80 categories. Each text datapoint is represented by a 1,000-dimensional bag of words (BoW) vector. We randomly selected 5,000 image-text pairs as the test set, with the rest as the retrieval set, and we randomly select 10,000 image-text pairs from the retrieval set as the training set. For the IAPR TC12 dataset, it contains 19, 999 image-text pairs annotated by 255 categories, and the text datapoint in each data pair of IAPR TC12 is represented as a 2000-dimensional BoW vector. Moreover, 2,000 image-text pairs are randomly sampled as the test set, and then the rest data pairs are treated as the retrieval set, and 5,000 image-text pairs are randomly selected from the retrieval set as the training set. Furthermore, for the two datasets, each two datapoints will be defined as a similar pair if they share at least one common label. Otherwise, they will be defined as a dissimilar pair.

2) Evaluation: To evaluate the retrieval quality of our proposed UCHM, similar to previous hashing works [60], [61], [62] we use three evaluation metrics: Mean Average Precision (**MAP**), Precision curves with respect to the number of top 100 returned results (**P@100**) and Precision-Recall curves (**PR**). MAP, and P@100 are used to measure the accuracy of Hamming ranking protocol. PR curve is used to evaluate the accuracy of hash lookup protocol.

Specifically, given a query datapoint, we can calculate the Average Precision (AP) score of the top M returned datapoints for the query:

$$AP = \sum_{i=1}^{M} \frac{I(i)}{m} \sum_{j=1}^{i} \frac{I(j)}{i} \tag{17}$$

where $I(i)$ equals 1 when the i^{th} returned datapoint is similar to the query; otherwise $I(i)$ is 0. m represents the total number of datapoints similar to the query in the returned top M datapoints. Then, the Mean Average Precision (MAP) is the mean of all the APs for queries. Moreover, for all the two datasets, we set M as 5000.

TABLE I
MAP OF HAMMING RANKING FOR ALL THE COMPARED METHODS WITH DIFFERENT NUMBERS OF BITS ON THE TWO DATASETS

Task	Method	MS COCO					IAPR TC12				
		16bits	32bits	64bits	96bits	128bits	16bits	32bits	64bits	96bits	128bits
T2I	UGACH [57]	0.523	0.513	0.569	0.613	0.587	0.397	0.456	0.481	0.454	0.472
	DJSRH [18]	0.502	0.570	0.601	0.612	0.608	0.425	0.435	0.445	0.456	0.460
	UKD-SS [1]	0.497	0.562	0.544	0.622	0.630	0.431	0.447	0.475	0.471	0.477
	DSAH [19]	0.583	0.611	0.625	0.628	0.633	0.464	0.478	0.485	0.491	0.489
	JDSH [20]	0.586	0.607	0.617	0.621	0.627	0.424	0.442	0.458	0.465	0.467
	DGCPN [15]	0.590	0.622	0.631	0.632	0.636	0.468	0.480	0.484	0.485	0.487
	UCHM	**0.621**	**0.657**	**0.673**	**0.675**	**0.676**	**0.479**	**0.493**	**0.501**	**0.504**	**0.503**
I2T	UGACH [57]	0.519	0.525	0.570	0.617	0.584	0.397	0.457	0.478	0.465	0.474
	DJSRH [18]	0.487	0.553	0.582	0.584	0.581	0.425	0.433	0.440	0.451	0.457
	UKD-SS [1]	0.495	0.562	0.547	0.632	0.638	0.431	0.449	0.477	0.473	0.482
	DSAH [19]	0.584	0.609	0.626	0.626	0.632	0.468	0.482	0.486	0.494	0.492
	JDSH [20]	0.583	0.603	0.615	0.624	0.627	0.428	0.451	0.467	0.474	0.478
	DGCPN [15]	0.592	0.623	0.630	0.631	0.634	0.470	0.480	0.484	0.486	0.486
	UCHM	**0.622**	**0.657**	**0.673**	**0.674**	**0.675**	**0.481**	**0.494**	**0.502**	**0.504**	**0.504**

TABLE II
PRECISION@100 FOR ALL THE COMPARED METHODS WITH DIFFERENT NUMBERS OF BITS ON THE TWO DATASETS

Task	Method	MS COCO					IAPR TC12				
		16bits	32bits	64bits	96bits	128bits	16bits	32bits	64bits	96bits	128bits
T2I	UGACH [57]	0.661	0.717	0.812	0.854	0.850	0.491	0.592	0.637	0.623	0.642
	DJSRH [18]	0.660	0.839	0.878	0.889	0.874	0.576	0.611	0.630	0.638	0.640
	UKD-SS [1]	0.693	0.784	0.802	0.854	0.868	0.540	0.597	0.624	0.629	0.644
	DSAH [19]	0.837	0.869	0.890	0.897	0.898	0.603	0.644	0.649	0.649	0.655
	JDSH [20]	0.838	0.876	0.901	0.902	0.907	0.593	0.617	0.632	0.640	0.638
	DGCPN [15]	0.839	0.883	0.893	0.892	0.896	0.593	0.622	0.637	0.637	0.635
	UCHM	**0.856**	**0.899**	**0.911**	**0.914**	**0.918**	**0.628**	**0.657**	**0.667**	**0.672**	**0.674**
I2T	UGACH [57]	0.667	0.754	0.823	0.852	0.847	0.490	0.588	0.628	0.631	0.642
	DJSRH [18]	0.611	0.796	0.840	0.845	0.841	0.590	0.609	0.626	0.634	0.635
	UKD-SS [1]	0.696	0.799	0.817	0.854	0.865	0.536	0.600	0.631	0.635	0.650
	DSAH [19]	0.824	0.848	0.868	0.868	0.871	0.618	0.638	0.655	0.663	0.667
	JDSH [20]	0.815	0.851	0.875	0.880	0.882	0.609	0.637	0.650	0.655	0.657
	DGCPN [15]	0.828	0.869	0.872	0.876	0.876	0.596	0.625	0.639	0.638	0.643
	UCHM	**0.843**	**0.883**	**0.892**	**0.895**	**0.896**	**0.639**	**0.664**	**0.672**	**0.675**	**0.677**

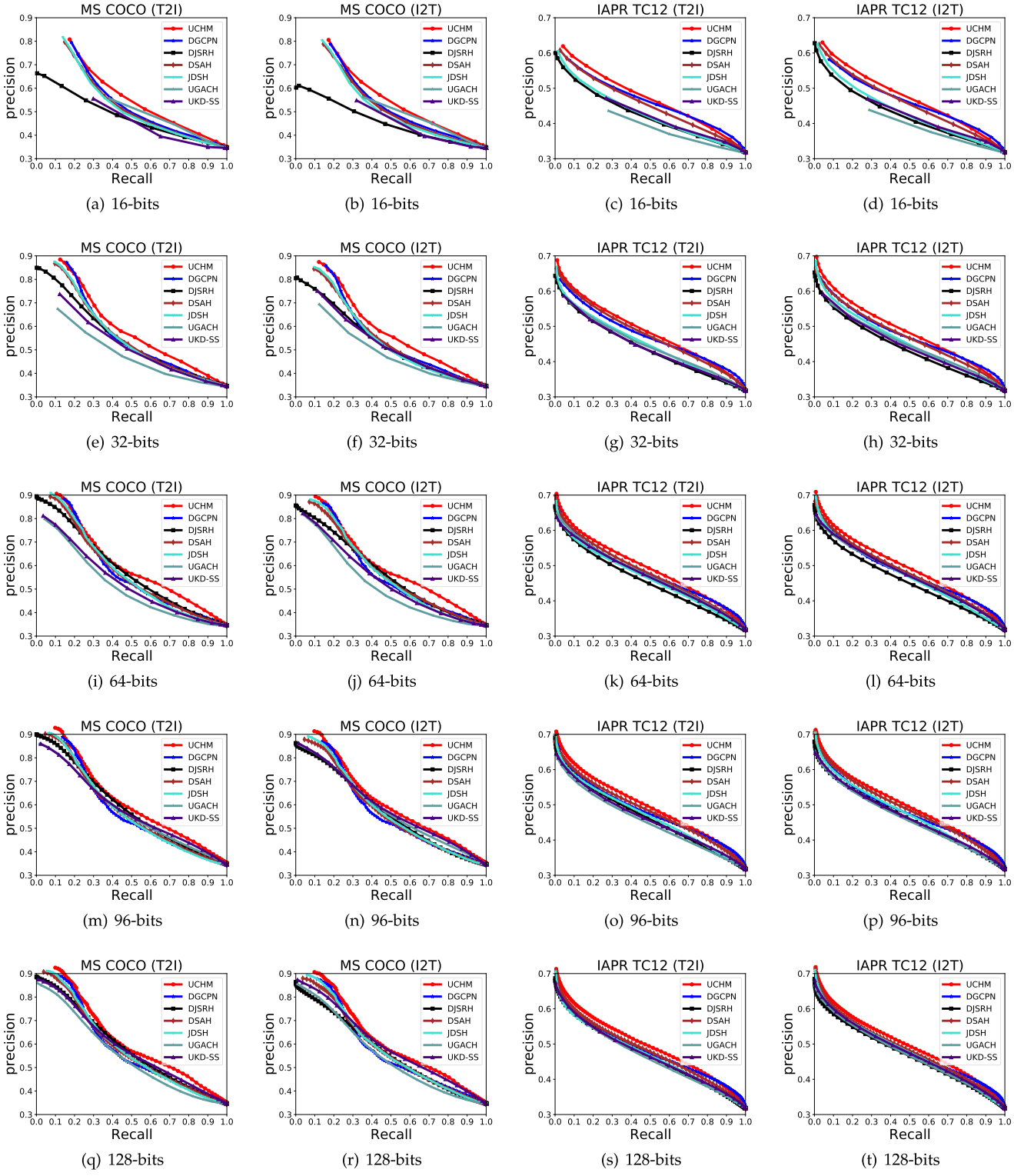
B. Baselines and Implementation Details

We compare our proposed method with six state-of-the-art deep unsupervised hashing methods, including **UGACH** [57], **DJSRH** [18], **UKD-SS** [1], **DSAH** [19], **JDSH** [20], and **DGCPN** [15].

For our proposed method, there are three sub-networks: the image-text interaction network, the image- and text-modality hashing networks. Specifically, the image-text interaction network is the same as the one used in BFAN [44], and for the input of this network, we pre-process the images and texts in the way of BFAN [44]. Moreover, similar to BFAN [44], we adopt Adam [63] with a mini-batch size of 128 and a learning rate 10^{-4} as our optimization algorithm. As for the image-modality hashing network, similar to [15] and [1], we utilize the VGG19 [45], which has been pre-trained on the ImageNet datasets [64], with the last layer replaced by a randomly initialized k-dimensional fully-connection layer as the image-modality hashing network. k denotes the length of hash codes. And for text-modality hashing network, similar to [15] and [19], it is a two-layer Multilayer Perceptron (MLP). The first fc layer in MLP has 4096 units with the ReLU [46] as activation function, and the second fc layer has k units to produce k bits continuos hash codes. Moreover, we use the 224×224 raw pixels of images and the BoW vectors of texts as inputs of the image- and text-modality hashing networks, respectively. We adopt SGD algorithm [65] with a mini-batch size of 128 and a learning rate within 10^{-3} to 10^{-2} to optimize the two hashing networks. And the settings of hyper-parameters are discussed in subsection IV-E.1 in detail.

C. Experimental Results

1) Hamming Ranking Protocol: Table I and Table II show the MAP and P@100 results of all baselines and our method on the two datasets, respectively. "I2T" ("T2I") denotes retrieving texts (images) with image (text) queries. From the two tables, it can be observed that our proposed UCMH outperforms all state-of-the-art baselines on the two evaluation metrics. For instance, compared with the best baseline DGCPN, the MAP results of our method for "I2T" task have average increases of 3.8% and 1.6% on datasets MS COCO and IAPR TC12, respectively. Moreover, as shown in Table II, among the baselines, JDSH achieves the best performance on MS COCO dataset for the P@100 evaluation metric, while compared with it, our proposed UCHM still achieves average increases of 1.5% and 2.1% for "IT2I" and "I2T" tasks. Over the IAPR TC12 dataset, for the P@100 evaluation metric, compared with the best competitor DSAH, our proposed UCHM achieves average increases of 2.0% and



(a) 16-bits (b) 16-bits (c) 16-bits (d) 16-bits

(e) 32-bits (f) 32-bits (g) 32-bits (h) 32-bits

(i) 64-bits (j) 64-bits (k) 64-bits (l) 64-bits

(m) 96-bits (n) 96-bits (o) 96-bits (p) 96-bits

(q) 128-bits (r) 128-bits (s) 128-bits (t) 128-bits

Fig. 2. Precision-recall curves of all the methods over the two datasets.

1.7% for "IT2I" and "I2T" tasks. These results indicate that the hash codes generated by our proposed method can preserve more original semantic similarity of datapoints than state-of-the-art baselines to achieve better cross-modal retrieval performance.

2) Hash Lookup Protocol: To evaluate our proposed UCHM through the lookup protocol, we calculate the PR curves for the returned points by varying the Hamming radius from 0 to k with a step-size of 1. The PR curves of our proposed UCHM and all the baselines with different hash code lengths over the two datasets are shown in Figure 2. It can be found that our proposed method outperforms state-of-the-art baselines over the two datasets. For example, in Figure 2, all the PR curves of our method are higher than all the ones

TABLE III
MAP OF UCHM AND ITS VARIANTS WITH DIFFERENT NUMBER OF BITS ON THE TWO DATASETS

Task	Method	MS COCO					IAPR TC12				
		16bits	32bits	64bits	96bits	128bits	16bits	32bits	64bits	96bits	128bits
T2I	$UCHM_1$	0.609	0.640	0.649	0.654	0.655	0.477	0.489	0.495	0.497	0.498
	$UCHM_2$	0.611	0.651	0.664	0.667	0.670	0.477	0.489	0.497	0.499	0.498
	$UCHM_3$	0.616	0.656	0.673	0.674	0.676	0.472	0.486	0.495	0.497	0.499
	$UCHM_4$	0.617	0.652	0.660	0.665	0.666	0.476	0.487	0.496	0.497	0.497
	$UCHM_5$	0.612	0.650	0.662	0.665	0.663	0.475	0.489	0.496	0.498	0.497
	$UCHM_6$	0.617	0.651	0.664	0.666	0.666	0.477	0.489	0.493	0.494	0.497
	UCHM	**0.621**	**0.657**	**0.673**	**0.675**	**0.676**	**0.479**	**0.493**	**0.501**	**0.504**	**0.503**
I2T	$UCHM_1$	0.608	0.639	0.650	0.655	0.655	0.480	0.491	0.491	0.499	0.500
	$UCHM_2$	0.612	0.650	0.664	0.667	0.670	0.476	0.491	0.499	0.500	0.501
	$UCHM_3$	0.617	0.653	0.671	0.673	0.675	0.474	0.489	0.498	0.500	0.501
	$UCHM_4$	0.617	0.651	0.659	0.664	0.666	0.478	0.487	0.497	0.497	0.498
	$UCHM_5$	0.613	0.649	0.661	0.664	0.662	0.477	0.490	0.496	0.498	0.497
	$UCHM_6$	0.618	0.652	0.662	0.665	0.665	0.477	0.493	0.495	0.496	0.498
	UCHM	**0.622**	**0.657**	**0.673**	**0.674**	**0.675**	**0.481**	**0.494**	**0.502**	**0.504**	**0.504**

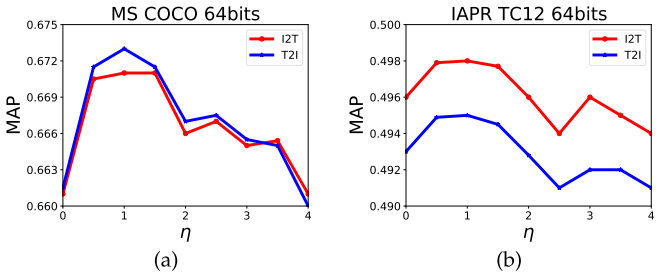


Fig. 3. A sensitivity analysis of the hyper-parameter η.

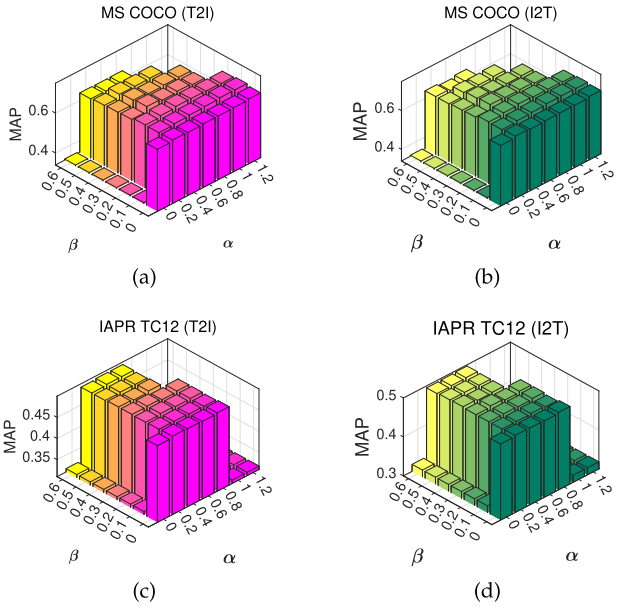


Fig. 4. A sensitivity analysis of the hyper-parameters α and β.

of baselines on the whole. These results demonstrate that compared with the baselines, our proposed UCHM is able to generate hash codes for similar datapoints in a smaller Hamming radius.

D. Discussion

1) Ablation Study: We totally propose five variants of UCHM. Specifically, to investigate the MIE similarity, we proposed three variants: **(i)** UCHM-1 trains the image-text interaction network with the original ranking loss Eq.(4) but not our proposed hashing-similarity-friendly loss Eq.(5); **(ii)** UCHM-2 directly chooses the corresponding most dissimilar text (image) for each image (text) in a mini-batch to construct the Ψ in Eq.(5). **(iii)** UCHM-3 directly uses the cross-modal similarity matrix $\boldsymbol{Q}$ as the guiding information to train hashing networks, i.e., $\gamma = 1$ in Eq.(7). Furthermore, to investigate the bit-selection module, we propose three variants: **(i)** UCHM-4 directly uses $\boldsymbol{C}^t = sgn(\boldsymbol{H}^v)$ and $\boldsymbol{C}^v = sgn(\boldsymbol{H}^t)$ but not our proposed bit-selection module to generate binary codes to construct the quantization loss; **(ii)** Similarly, UCHM-5 directly uses $\boldsymbol{C}^v = \boldsymbol{C}^t = sgn(\boldsymbol{H}^v + \boldsymbol{H}^t)$ to replace our bit-selection module. **(iii)** UCHM-6 directly uses the modality-specific binary hash codes to construct the quantization loss, i.e., $\boldsymbol{C}^t = sgn(\boldsymbol{H}^t)$ and $\boldsymbol{C}^v = sgn(\boldsymbol{H}^v)$.

The MAP results of the above six variants on 64 and 96 bits over the two datasets are shown in Table III. Based on these results, the following observations are obtained: (1) Compared UCHM with UCHM-1, the MAP of UCHM for "T2I" task has average increases of 2.2% and 0.7% on datasets MS COCO and IAPR TC12, respectively. These results demonstrate that our proposed hash-similarity-friendly loss is more suitable than the original ranking loss as the objective function, which will optimize the image-text interaction network to generate better cross-modal similarity for hashing. (2) Compared UCHM with UCHM-2, the MAP of UCHM has average increases of 0.8% and 0.9% on datasets MS COCO for "I2T" and "T2I" tasks. Maybe it is because the 'dissimilar' pairs constructed by selecting the most dissimilar point are too simple to contain useful information, and compared with such a way, by randomly selecting η% data pairs from the current mini-batch as 'dissimilar' pairs, it will select some hard pairs containing enough useful information to train image-text interaction network well. (3) Compared UCHM with UCHM-3, the MAP results of UCHM can achieve better performance. It reveals that the intra-modal similarity also contains some useful information to guide the hashing network to generate better hash codes. (4) The MAP results of UCHM for "I2T"

TABLE IV
THE INFERENCE TIME (IN MILLISECOND) OF OUR METHOD ON THE TWO DATASETS. FE MEANS EXTRACTING FEATURE FOR THE QUERY. HASHING MEANS MAPPING THE FEATURE INTO HASH CODE. RETRIEVING DENOTES RETRIEVING DATA WITH THE GENERATED HASH CODE. TOTAL REPRESENTS THE WHOLE TIME

Dataset	Task	FE	Hashing	Retrieving	Total
MS COCO	$T2I$	0.13	0.36	20.09	20.58
	$I2T$	4.56	0.35	20.32	25.23
IAPR TC12	$T2I$	0.14	0.32	3.32	3.78
	$I2T$	4.12	0.32	3.32	7.76

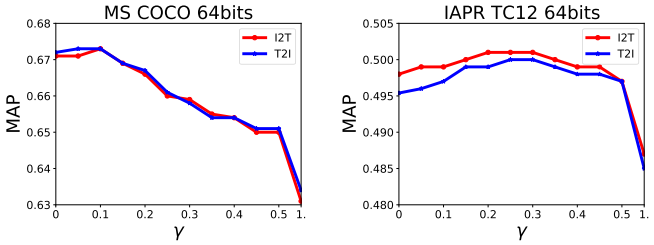


Fig. 5. A sensitivity analysis of the hyper-parameter γ.

task have average increases of 0.9%, 1.3% and 0.8% on MS COCO than UCHM-4, UCHM-5 and UCHM-6, respectively. These results show that the binary codes generated by our proposed bit-selection module for the quantization loss are more high-quality to further improve the retrieval performance than those generated by the original ways.

E. Inference Efficiency

We further conduct experiments to study the inference efficiency of cross-modal hashing. After training the hashing networks, we can first generate hash codes for all the datapoints in the database, and then store them in the memory. When coming a new query, the inference phase of our method and those of DGCPN, JDSH, DASH and DJSRH are the same, which consists of three steps: extracting feature for the query, mapping the feature into hash code, retrieving data with the generated hash code. Hence, we directly show the time of each step testing on our method in Table IV. It can be found that given a query image, the whole retrieval time is only 7.76ms on the IAPR TC12. Moreover, given a query text, the whole retrieval time is 20.58ms on the MS COCO dataset. Hence, such a speed is fast which can meet the daily needs of users.

1) Sensitivity to Hyper-Parameters: Here, we study how the hyper-parameters η, α, β, and γ affect our proposed UCHM when setting the hash code length as 64-bits.

First, to investigate the influence of hyper-parameter η on our similarity matrix $\boldsymbol{Q}$ generated by Eq.(6), we set $\alpha = 1, \beta = 0.1$ for MS COCO ($\alpha = 0.4, \beta = 0.3$ for IAPR TC12), $\gamma = 0$ for the two datasets. As the results are shown in Figure 3, when $\eta = 1$, i.e., we randomly selected 1% image-text pairs from the current mini-batch as Ψ to train the image-text interaction network, the calculated similarity matrix is the best quality that helps the hashing networks achieve the best retrieval performance.

Moreover, to study the influence versus the variations of α and β, we set $\eta = 1, \gamma = 0$ for the two datasets. As the results are shown in Figure 4, it can be found that when $\alpha = 1$, $\beta = 0.1$ for MS COCO ($\alpha = 0.4, \beta = 0.3$ for IAPR TC12), our method can achieve the best performance.

Furthermore, to investigate the hyper-parameter γ, we set $\alpha = 1, \beta = 0.1$ for MS COCO ($\alpha = 0.4, \beta = 0.3$ for IAPR TC12) and $\eta = 1$ for the two datasets. The results are shown in Figure 5. It can be seen that when $\gamma = 0.1$ for MS COCO ($\gamma = 0.25$ for IAPR TC12), our method can achieve the best performance.

V. CONCLUSION

In this paper, we proposed a novel Unsupervised Cross-modal Hashing with Modality-interaction. It leverages a novel hash-similarity-friendly loss to train an image-text interaction network to generate a high-quality MIE similarity matrix for the training set. Then, the generated MIE similarity matrix is used to guide the training of hashing networks. Furthermore, during the training process of the hashing networks, a novel bit-selection module is proposed to generate superior binary codes for the quantization loss to further improve the retrieval performance. Extensive experiments on two benchmark datasets demonstrate the effectiveness of our proposed UCHM.

REFERENCES

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

[2] Q. Lin, W. Cao, Z. He, and Z. He, "Mask cross-modal hashing networks," *IEEE Trans. Multimedia*, vol. 23, pp. 550–558, 2021.

[3] Y. Cao, M. Long, J. Wang, and H. Zhu, "Correlation autoencoder hashing for supervised cross-modal search," in *Proc. ACM Int. Conf. Multimedia Retr.*, Jun. 2016, pp. 197–204.

[4] X. Nie, B. Wang, J. Li, F. Hao, M. Jian, and Y. Yin, "Deep multiscale fusion hashing for cross-modal retrieval," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 1, pp. 401–410, Jan. 2021.

[5] D. Wang, Q. Wang, and X. Gao, "Robust and flexible discrete hashing for cross-modal similarity search," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 10, pp. 2703–2715, Oct. 2018.

[6] 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.

[7] R.-C. Tu et al., "Deep cross-modal hashing with hashing functions and unified hash codes jointly learning," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 2, pp. 560–572, Feb. 2022.

[8] R.-C. Tu et al., "Deep cross-modal proxy hashing," *IEEE Trans. Knowl. Data Eng.*, early access, Jun. 28, 2022, doi: 10.1109/TKDE.2022.3187023.

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

[10] Q.-Y. Jiang and W.-J. Li, "Discrete latent factor model for cross-modal hashing," *IEEE Trans. Image Process.*, vol. 28, no. 7, pp. 3490–3501, Jul. 2019.

[11] Y. Cao, B. Liu, M. Long, and J. Wang, "Cross-modal Hamming hashing," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 202–218.

[12] H. Zhai, S. Lai, H. Jin, X. Qian, and T. Mei, "Deep transfer hashing for image retrieval," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 2, pp. 742–753, Feb. 2021.

[13] X. Li, J. Yu, Y. Wang, J.-Y. Chen, P.-X. Chang, and Z. Li, "DAHP: Deep attention-guided hashing with pairwise labels," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 3, pp. 933–946, Mar. 2022.

[14] 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. (CVPR)*, Jul. 2017, pp. 7380–7388.

[15] 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, vol. 35, no. 5, pp. 4626–4634.

[16] 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*, Jun. 2013, pp. 785–796.

[17] Y. Shi et al., "Deep adaptively-enhanced hashing with discriminative similarity guidance for unsupervised cross-modal retrieval," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 10, pp. 7255–7268, Oct. 2022.

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

[19] D. Yang, D. Wu, W. Zhang, H. Zhang, B. Li, and W. Wang, "Deep semantic-alignment hashing for unsupervised cross-modal retrieval," in *Proc. Int. Conf. Multimedia Retr.*, Jun. 2020, pp. 44–52.

[20] 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. 43rd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2020, pp. 1379–1388.

[21] Y. Zhao, Y. Zhu, S. Liao, Q. Ye, and H. Zhang, "Class concentration with twin variational autoencoders for unsupervised cross-modal hashing," in *Proc. Asian Conf. Comput. Vis.*, 2022, pp. 349–365.

[22] G. Wu et al., "Unsupervised deep hashing via binary latent factor models for large-scale cross-modal retrieval," in *Proc. IJCAI*, 2018, vol. 1, no. 3, p. 5.

[23] G. Wu, J. Han, Z. Lin, G. Ding, B. Zhang, and N. Qiang, "Joint image-text hashing for fast large-scale cross-media retrieval using self-supervised deep learning," *IEEE Trans. Ind. Electron.*, vol. 66, no. 12, pp. 9868–9877, Dec. 2019.

[24] D. Wang, X. Gao, X. Wang, and L. He, "Semantic topic multimodal hashing for cross-media retrieval," in *Proc. 24th Int. Joint Conf. Artif. Intell.*, 2015, pp. 3890–3896.

[25] D. Zhang and W.-J. Li, "Large-scale supervised multimodal hashing with semantic correlation maximization," in *Proc. 28th AAAI Conf. Artif. Intell.*, 2014, pp. 1–7.

[26] V. E. Liong, J. Lu, Y.-P. Tan, and J. Zhou, "Cross-modal deep variational hashing," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4077–4085.

[27] Q. Lin, W. Cao, Z. He, and Z. He, "Semantic deep cross-modal hashing," *Neurocomputing*, vol. 396, pp. 113–122, Jul. 2020.

[28] R. Xu, C. Li, J. Yan, C. Deng, and X. Liu, "Graph convolutional network hashing for cross-modal retrieval," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 982–988.

[29] X. Luo, X.-Y. Yin, L. Nie, X. Song, Y. Wang, and X.-S. Xu, "SDMCH: Supervised discrete manifold-embedded cross-modal hashing," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 2518–2524.

[30] Y. Shi, X. You, F. Zheng, S. Wang, and Q. Peng, "Equally-guided discriminative hashing for cross-modal retrieval," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 4767–4773.

[31] B. Wang, Y. Yang, X. Xu, A. Hanjalic, and H. T. Shen, "Adversarial cross-modal retrieval," in *Proc. 25th ACM Int. Conf. Multimedia*, Oct. 2017, pp. 154–162.

[32] X. Liu, H. Zeng, Y. Shi, J. Zhu, and K.-K. Ma, "Deep rank cross-modal hashing with semantic consistent for image-text retrieval," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2022, pp. 4828–4832.

[33] 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.

[34] 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.

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

[36] 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/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4242–4251.

[37] S. Kumar and R. Udupa, "Learning hash functions for cross-view similarity search," in *Proc. 22nd Int. Joint Conf. Artif. Intell.*, 2011, pp. 1–6.

[38] J. Zhou, G. Ding, and Y. Guo, "Latent semantic sparse hashing for cross-modal similarity search," in *Proc. 37th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2014, pp. 415–424.

[39] 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.

[40] G. Mikriukov, M. Ravanbakhsh, and B. Demir, "Unsupervised contrastive hashing for cross-modal retrieval in remote sensing," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2022, pp. 4463–4467.

[41] Z. Lin, G. Ding, J. Han, and J. Wang, "Cross-view retrieval via probability-based semantics-preserving hashing," *IEEE Trans. Cybern.*, vol. 47, no. 12, pp. 4342–4355, Dec. 2016.

[42] Y. Zhuo, Y. Li, J. Hsiao, C. Ho, and B. Li, "CLIP4Hashing: Unsupervised deep hashing for cross-modal video-text retrieval," in *Proc. Int. Conf. Multimedia Retr.*, Jun. 2022, pp. 158–166.

[43] A. Radford et al., "Learning transferable visual models from natural language supervision," in *Proc. 38th Int. Conf. Mach. Learn.*, 2021, pp. 8748–8763.

[44] C. Liu, Z. Mao, A.-A. Liu, T. Zhang, B. Wang, and Y. Zhang, "Focus your attention: A bidirectional focal attention network for image-text matching," in *Proc. 27th ACM Int. Conf. Multimedia*, 2019, pp. 3–11.

[45] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[46] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, 2011, pp. 315–323.

[47] J. Wei, Y. Yang, X. Xu, X. Zhu, and H. T. Shen, "Universal weighting metric learning for cross-modal retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 10, pp. 6534–6545, Oct. 2022.

[48] R.-C. Tu et al., "Hashing based efficient inference for image-text matching," in *Proc. Findings Assoc. Comput. Linguistics (ACL-IJCNLP)*, 2021, pp. 743–752.

[49] X. Xu, T. Wang, Y. Yang, L. Zuo, F. Shen, and H. T. Shen, "Cross-modal attention with semantic consistence for image–text matching," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 12, pp. 5412–5425, Feb. 2020.

[50] Q. Cheng, Z. Tan, K. Wen, C. Chen, and X. Gu, "Semantic pre-alignment and ranking learning with unified framework for cross-modal retrieval," *IEEE Trans. Circuits Syst. Video Technol.*, early access, Jun. 13, 2022, doi: 10.1109/TCSVT.2022.3182549.

[51] H. Chen, G. Ding, X. Liu, Z. Lin, J. Liu, and J. Han, "IMRAM: Iterative matching with recurrent attention memory for cross-modal image-text retrieval," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 12655–12663.

[52] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[53] R. Krishna et al., "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *Int. J. Comput. Vis.*, vol. 123, no. 1, pp. 32–73, 2017.

[54] P. Anderson et al., "Bottom-up and top-down attention for image captioning and visual question answering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6077–6086.

[55] K. Cho et al., "Learning phrase representations using RNN encoder–decoder for statistical machine translation," 2014, *arXiv:1406.1078*.

[56] F. Faghri, D. J. Fleet, J. Ryan Kiros, and S. Fidler, "VSE++: Improving visual-semantic embeddings with hard negatives," 2017, *arXiv:1707.05612*.

[57] J. Zhang, Y. Peng, and M. Yuan, "Unsupervised generative adversarial cross-modal hashing," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 1–8.

[58] T.-Y. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 740–755.

[59] H. J. Escalante et al., "The segmented and annotated IAPR TC-12 benchmark," *Comput. Vis. Image Understand.*, vol. 114, no. 4, pp. 419–428, 2010.

[60] R.-C. Tu, X.-L. Mao, and W. Wei, "MLS3RDUH: Deep unsupervised hashing via manifold based local semantic similarity structure reconstructing," in *Proc. 29th Int. Joint Conf. Artif. Intell.*, Jul. 2020, pp. 3466–3472.

[61] R.-C. Tu, X.-L. Mao, J.-N. Guo, W. Wei, and H. Huang, "Partial-softmax loss based deep hashing," in *Proc. Web Conf.*, Apr. 2021, pp. 2869–2878.

[62] R.-C. Tu et al., "Unsupervised hashing with semantic concept mining," 2022, *arXiv:2209.11475*.

[63] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[64] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
[65] H. Robbins and S. Monro, "A stochastic approximation method," *Ann. Math. Statist.*, vol. 22, no. 3, pp. 400–407, 1951.

Rong-Cheng Tu received the bachelor's degree from the Beijing Institute of Technology, China, in 2018, where he is currently pursuing the Ph.D. degree with the Department of Computer Science and Technology. His research interests include deep learning, information retrieval, and learning to hash.

Jie Jiang received the Ph.D. degree from Peking University. He has long been devoted to big data and distributed computing. He is well-known in China for his expertise in data science. He is also a member of CCF Task Force on Big Data. He has been invited to give keynote speeches at SACC and Hadoop in China for many times.

Qinghong Lin is currently pursuing the master's degree with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China. His current research interests include deep learning and computer vision.

Chengfei Cai received the bachelor's and master's degrees from Zhejiang University, China. He is currently a Senior Researcher of Ads Multimedia AI with Tencent Data Platform. His research interests include information retrieval, machine learning, deep learning, data mining, and computer vision.

Shangxuan Tian received the B.S. degree from the School of Computer Science and Technology, Northwestern Polytechnical University, and the Ph.D. degree from the School of Computing, National University of Singapore. His current research interests include document image analysis (OCR) and multimedia retrieval.

Hongfa Wang received the B.S. degree from the Department of Mathematics, Southeast University, Nanjing, China, in 2005, and the master's degree in operation science and control theory from the Chinese Academy of Sciences, Beijing, China, in 2008. He is currently an Expert Researcher of Ads Multimedia AI with Tencent Data Platform. His research interests include computer vision, machine learning, and pattern recognition.

Wei Liu (Fellow, IEEE) received the Ph.D. degree in EECS from Columbia University, USA. He is currently a Distinguished Scientist with Tencent and the Director of Ads Multimedia AI with Tencent Data Platform. He was a Research Scientist with the IBM T. J. Watson Research Center, USA. He has long been devoted to fundamental research and technology development in core fields of AI. His research works win a number of awards and honors, such as the 2013 Jury Award for Best Thesis of Columbia University, the 2016 and 2017 SIGIR Best Paper Award Honorable Mentions, and the 2018 "AI's 10 To Watch" Honor. He is a fellow of International Association for Pattern Recognition (IAPR), Asia-Pacific Artificial Intelligence Association (AAIA), and the Institute of Mathematics and its Applications (IMA), and an Elected Member of the International Statistical Institute (ISI). He currently serves on the editorial boards of IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, and IEEE OPEN JOURNAL OF CONTROL SYSTEMS. He is the Area Chair of NeurIPS, ICML, CVPR, ICCV, IJCAI, and AAAI.