Unsupervised Hashing Retrieval via Efficient Correlation Distillation

Xi Zhang, Xiumei Wang, and Peitao Cheng

***Abstract*— Deep hashing has been widely used in multimedia retrieval systems due to its storage and computation efficiency. Unsupervised hashing has received a lot of attention in recent years because it does not rely on label information. However, existing deep unsupervised hashing methods usually use rough pairwise relations to constrain the similarity between hash codes locally, which is insufficient and inefficient to reconstruct accurate correlations across samples. To address this issue, we propose a generic distillation framework for the preservation of the similarity relationship. Specifically, we design a distillation loss to reconstruct the batchwise similarity distribution between feature space and hash code space, allowing us to capture the global correlation knowledge contained in features and propagate it into hash codes efficiently. This framework can apply to both intra-modal and inter-modal scenarios. Furthermore, we design a new quantization method that quantizes the continuous values to a clipping value instead of ±1 to reduce the inconsistency between continuous features and hash codes. This method can also avoid the vanishing gradient problem during training. Finally, extensive experiments for image hashing retrieval and cross-modal hashing retrieval on public datasets demonstrate that the proposed method can yield compact hash codes and outperforms the state-of-the-art baselines.**

***Index Terms*— Hashing retrieval, unsupervised hashing, correlation distillation.**

I. Introduction

With the explosion of multimedia data including texts, images, and videos from social media, hashing-based searching technologies have gradually become popular due to low storage costs and efficient Hamming distance calculation [1]. Hashing methods convert high-dimensional data into low-dimensional binary codes while maintaining the semantic similarity between data points. As a result, hashing retrieval produces similar binary codes for similar items.

Thanks to the advantages of deep learning, deep hashing methods have made significant gains in retrieval performance. Unlike the shallow hashing methods relying on hand-crafted features, deep hashing methods obtain binary representations within an end-to-end process that learns high-level semantic and hash codes simultaneously. These methods can be categorized into supervised methods [2], [3], [18], [19], [20], [22], [34], [51] and unsupervised methods [4], [5], [6], [7], [8], [9], [10], [11], [12], [14], [16], [17], [25], [26], [27], [28], [29], [30], [31], [32], [33], [49], depending on whether semantic labels are utilized. The supervised hashing methods show better performance than unsupervised ones by leveraging semantic information from annotations. However, it is costly to gather large-scale annotated data, so unsupervised hashing methods are more practical in a real-world scenario. In this paper, we mainly focus on unsupervised deep hashing for multimodal retrieval, including image hashing and cross-modal hashing. Image hashing transforms images into compact binary codes, and cross-modal hashing maps heterogeneous multimedia content, especially in vision and language, to compact binary codes in common space.

For the unsupervised deep hashing methods, it is critical to capture the semantic relationship in the feature space and preserve it well in the hash code space. The semantic relationship, i.e., the guiding information, comes from similarity information among image features in unsupervised image hashing, while in unsupervised cross-modal hashing, it comes from correlations between data points from different modalities. Most existing works in unsupervised deep hashing focus on constructing and reconstructing the similarity matrix to capture and preserve the semantic relations. Semantic Structure based unsupervised Deep Hashing (SSDH) [4], for example, builds the similarity matrix based on the distance histogram between image features. DistillHash [7] gets a more confident similarity matrix from the noisy semantic relevance. Binary Generative Adversarial Networks (BGAN) [5] creates the similarity matrix relying on the nearest neighbor relations. Analogously, the similarity matrix is also widely used in unsupervised cross-modal hashing. Deep Joint-Semantics Reconstructing Hashing (DJSRH) [25] and Deep Semantic-Alignment Hashing (DSAH) [28] both employ a joint-semantic affinity matrix to capture the semantic relations of different modalities. Joint-modal Distributionbased Similarity Hashing (JDSH) [29] builds a weighted similarity matrix based on the distance distribution. Knowledge Distillation Cross-Modal Hashing (KDCMH) [33] follows JDSH to construct the similarity matrix. Deep Graph-neighbor Coherence Preserving Network (DGCPN) [32] proposes graph-neighbor coherence to improve the accuracy of the similarity matrix. After obtaining the similarity matrix, conventional methods to

Manuscript received 2 September 2022; revised 10 December 2022; accepted 25 December 2022. Date of publication 4 January 2023; date of current version 3 July 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 61972305, Grant 61871308, and Grant 62050175; in part by the Natural Science Basic Research Program of Shaanxi Program under Grant 2023-JC-ZD-39; and in part by the Key Research and Development Program of Shaanxi under Grant 2021ZDLGY02-03. This article was recommended by Associate Editor H. Zhang. *(Corresponding author: Peitao Cheng.)*

Xi Zhang and Xiumei Wang are with the School of Electronic Engineering, Xidian University, Xi'an 710071, China (e-mail: zhangxi19982017@gmail.com; wangxm@xidian.edu.cn).

Peitao Cheng is with the School of Electro-Mechanical Engineering, Xidian University, Xi'an 710071, China (e-mail: chengpeitao@163.com).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSVT.2023.3234037.

Digital Object Identifier 10.1109/TCSVT.2023.3234037

TABLE I
A SUMMARY OF MANNERS TO PRESERVE SIMILARITY IN DEEP UNSUPERVISED IMAGE HASHING METHODS AND DEEP UNSUPERVISED CROSS-MODAL HASHING METHODS

Approach	Manners	Similarity Preserving
PUDH [6]	-	Classification Loss
UTH [11]	Triplet-wise	Triplet Loss
BGAN [5]	Pairwise	$\left(s_{ij}^f - s_{ij}^h\right)^2$
DistillHash [7]		$\log\left(1+\exp\left(s_{ij}^h\right)\right) - s_{ij}^f s_{ij}^h$
GreedyHash [16]		$\left(s_{ij}^f - s_{ij}^h\right)^2$
Bi-Half [17]		$\left(s_{ij}^f - s_{ij}^h\right)^2$
MLS^{3}RDUH [8]		$\log\left(\cosh\left(s_{ij}^f - s_{ij}^h\right)\right)$
DJSRH [25]		$\left(s_{ij}^f - s_{ij}^h\right)^2$
DSAH [28]		$\left(s_{ij}^f - s_{ij}^h\right)^2$
JDSH [29]		$\left(s_{ij}^f - s_{ij}^h\right)^2$
UKD [27]		$\left(s_{ij}^f - s_{ij}^h\right)^2$
DGCPN [32]		$\left(s_{ij}^f - s_{ij}^h\right)^2$
AGCH [26]		$\left(s_{ij}^f - s_{ij}^h\right)^2$
Ours	Batchwise	Correlation Distillation Loss

We denote the similarity between item i and j in the feature space and hash code space as s_{ij}^f and s_{ij}^h, respectively. s_{ij}^f is generally obtained by the distance between samples or refined based on correlations across samples. s_{ij}^h is calculated by inner product or cosine similarity between hash codes.

preserve the similarity information in hash code space is to reconstruct the similarity relationship by regression [5], [16], [25], [28] or maximum likelihood [7]. However, as shown in Table I, most of these methods perform similarity reconstruction locally in the manner of pairwise similarity preserving, which lacks coverage of data distribution. This manner may not fully exploit the abundant similarity structure among all data points. Moreover, these processed similarity matrices are not accurate enough and may discard some semantic relevance information. It is desirable to find a unified way to explore the intrinsic semantic correlations globally in intra-modal and inter-modal views, as well as, to preserve them in hash code space efficiently.

Thus, in this paper, we develop a new distillation mechanism to capture and preserve the semantic relationship for both unsupervised image hashing and unsupervised cross-modal hashing retrieval. Specifically, different from knowledge distillation used to transfer knowledge from a large model to a smaller one, we distill the global semantic correlations in continuous feature space into binary code space by establishing connections between the two similarity distributions implied in the features and hash codes.

In addition to similarity reconstruction, quantization is another crucial procedure that converts continuous representations to binary codes for hashing retrieval. Most existing methods use tanh as the activation function to get relaxed binary codes instead of sgn because the standard back-propagation is infeasible for sgn function. Then, approximate binary values closed to ± 1 can be obtained by employing quantization loss [5], [6], [11], [16] or continuation skill [2], [5], [25], [28], [49]. Yet, quantizing the activated values to ± 1 may increase the gap between continuous values and hash codes and make the quantization harder. Moreover, as the tanh function value gradually approaches ± 1, its gradient is almost zero, which makes the model difficult to optimize and slow to converge.

To handle the quantization issue, this paper proposes a new quantization scheme, dubbed *clip-quan*, which clips and quantizes the activation value to an upper bound instead of ± 1. In this way, the hash layer can achieve lower quantization error and easier optimization.

The major contributions of our work can be summarized as follows:

- We describe a unified and novel distillation way for image and cross-modal hashing in an unsupervised fashion, called **U**nsupervised **H**ashing via **C**orrelation **D**istillation (CDUH). This way can efficiently transfer the intrinsic semantic structure across data samples into hash codes in intra-modal or inter-modal view.
- A novel *clip-quan* quantization strategy is adopted to obtain nearly binary hash codes, which reduces the gap caused by quantization and alleviates the vanishing gradient problem in tanh.
- Numerous experiments on image hashing and cross-modal hashing datasets demonstrate that our proposed method shows significant improvement over the state-of-the-art unsupervised hashing methods.

II. RELATED WORK

We will briefly review some related works in this section from three aspects: deep image hashing retrieval, deep cross-modal hashing retrieval, and knowledge distillation.

A. Deep Image Hashing

Deep image hashing can be roughly categorized into supervised image hashing and unsupervised image hashing. Supervised image hashing methods take advantage of the label information to get the semantic relationship. Deep Supervised Hashing (DSH) [51] uses the pairwise supervision from annotations to learn the hash function and directly constrains the generated hash codes without activation functions. HashNet [2] preserves the similarity relationship with a weighted pairwise cross-entropy loss and obtains exactly binary codes by the continuation technique. Max-Margin Hamming Hashing (MMHH) [3] enhances the robustness of the loss function to noisy data by refining the probability function of the similarity between hash codes.

Regarding unsupervised image hashing, a mass of methods has been proposed in recent years. Most of these methods are based on the reconstruction of semantic similarity. SSDH [4] creates a similarity matrix based on the distribution of distances between features. BGAN [5] constructs a similarity matrix according to the k-nearest neighbor graph and utilizes an adversarial Auto-encoder network to reconstruct the input images. DistillHash [7] distills a more confident similarity

relationship from the original similarity relationship. In these methods, the similarity relationship is artificially allocated as -1 or $+1$, which is not accurate enough and may discard similarity information. Deep Unsupervised Hashing via Manifold based Local Semantic Similarity Structure Reconstructing (MLS^3RDUH) [8] obtains a similarity matrix by assuming that the data has a manifold structure, which reduces the noise in semantic relations. Twin-Bottleneck Hashing (TBH) [9] constructs a code-driven similarity graph to explore the semantic structure and employs an Auto-encoder to reconstruct the image features from continuous hash codes. Contrast-based Unsupervised Hashing Learning with Multi-hashcode (CUHM) [49] captures and preserves the semantic similarity between data based on contrastive learning. These three methods achieve better performance by virtue of more accurate similarity relations.

Artificial supervision is also used in some works to learn hash functions. Pseudo label based Unsupervised deep Discriminative Hashing (PUDH) [6] trains a classification network based on the labels obtained by K-means and gets hash codes from the intermediate features. Unsupervised Deep K-means Hashing (UDKH) [53] also generates discriminative hash codes under the guidance of K-means cluster labels. DeepBit [10] learns the hash function by minimizing the distance between the rotated image and the original image. But this method can not guarantee that the hash codes for different images are distinguished. Therefore, Unsupervised Triplet Hashing (UTH) [11] adds a randomly selected image to form a triplet training set to learn more discriminative binary representations. It is notable that some recent works introduce contrastive learning to obtain image representations and generate hash codes via quantization. Self-supervised Product Quantization (SPQ) [12] introduces a cross contrastive learning strategy to learn latent representations inspired by SimCLR [15]. Analogously, Contrastive Quantization with Code Memory (MeCoQ) [14] enhances contrastive learning with a quantization code memory and a debiased technique. Although these methods do not depend on the quality of the features extracted from the pre-trained backbones, the optimization of the network requires a lot of computational and storage resources due to the characteristics of contrastive learning.

Quantization, as an important process of hashing, has also attracted attention in some works. GreedyHash [16] optimizes hash codes directly with the straight-through estimator. Bihalf [17] applies the same optimization method and manages to maximize the information capacity of every hash bit. However, the estimated gradient used in these methods may be inaccurate. SPQ [12] and MeCoQ [14] deploy Product Quantization [13] to generate the binary codes that contain richer representations than conventional hash codes. Nevertheless, the optimization of the codebook and the indexing of binary codes are time-consuming in the training and inference stages, respectively.

B. Deep Cross-Modal Hashing

Existing deep cross-modal hashing methods can also be grouped into supervised methods and unsupervised methods. The supervised methods exploit semantic information in labels to learn discriminative binary representations in hash space for different modalities. Deep Cross-Modal Hashing (DCMH) [18] proposes an end-to-end framework that performs feature learning and hash function learning simultaneously. Self-Supervised Adversarial Hashing (SSAH) [19] introduces two adversarial networks to maximize consistency between different modalities. Zhang et al. [34] employ an attention module to focus on the discriminative contents in features. Cross-Modal Mutual Quantization (CMMQ) [23] proposes a proxy-based contrastive loss to mitigate the gap between different modalities and trains networks with small loss samples to combat noisy labels. Dual Encoding for Video Retrieval by Text (Dual Encoding) [24] proposes a non-hash method focusing on video retrieval by text with a dual deep encoding network. In addition to hash retrieval, some works like ALign the image and text representations BEfore Fusing (ALBEF) [36] and Contrastive Language-Image Pre-training (CLIP) [35] in the field of large-scale visual language representation learning have also made great progress in cross-modal retrieval. Other notable methods include Deep Adversarial Discrete Hashing (DADH) [20], Deep Multiscale Fusion Hashing (DMFH) [21], and Mask Deep Cross-modal Hashing (MDCH) [22].

Unsupervised methods only use correlations from representations of different modalities, which alleviates dependence on annotations. Most unsupervised cross-modal retrieval methods, like unsupervised image retrieval methods, focus on similarity reconstruction. DJSRH [25] constructs a joint-semantic affinity matrix to capture the semantic relations among the inputs. DSAH [28] improves this work with a scheme of Auto-encoder. JDSH [29] weights the joint-modal similarity matrix according to its statistics. Unsupervised Knowledge Distillation (UKD) [27] learns a similarity matrix from an unsupervised model and uses it to guide a supervised model. KDCMH [33] employs the triplet loss to generate more discriminative hash codes. DGCPN [32] explores data neighbors to obtain more accurate similarity between data. Recently, several works introduce Generative Adversarial Network (GAN) to improve unsupervised cross-modal hashing. Unsupervised Generative Adversarial Cross-modal Hashing (UGACH) [30] utilizes GAN to explore the underlying manifold structure of cross-modal data. Unsupervised coupled Cycle generative adversarial Hashing (UCH) [31] employs coupled cycle GANs to learn common representations and hash codes with adversarial learning and feature reconstruction. Besides, Aggregation-based Graph Convolutional Hashing (AGCH) [26] uses Graph Convolutional Networks (GCNs) to explore the semantic structure of data and preserve the intra-modal and inter-modal relationship in hash embeddings.

C. Knowledge Distillation

Knowledge distillation [37], [38] refers to the process of transferring knowledge from a large teacher model to a smaller student model without loss of validity. The student model can be deployed on systems with little computing power while maintaining about the same performance as the teacher model. As a consequence, it has attracted increasing attention in recent

years. The vanilla knowledge distillation [38] transfers the "dark knowledge" to the student model by mimicking the logit from the classifier layer in the teacher model. A temperature factor is introduced to increase the information in negative logits. Furthermore, the activations and feature maps of intermediate layers can also be used as the knowledge to guide the student model [37], improving the performance of the student model. Moreover, the relationship between data samples contains rich knowledge that can be transferred to the student model [37]. Tung and Mori [39] propose similarity-preserving knowledge distillation and preserve the pairwise similarity in the student model, which complements the traditional knowledge distillation.

Knowledge distillation has been employed for hashing retrieval in some works. Deep Transfer Hashing (DTH) [52] transfers the similarity knowledge from the teacher model to the hash model for better retrieval performance. UKD [27] uses the similarity information generated by the unsupervised teacher model to guide a supervised student model, with pairwise correlations preserved. Similarly, Semi-supervised Knowledge Distillation for Cross-Modal Hashing (SKDCH) [42] exploits the relevance knowledge from a semi-supervised teacher modal. Miech et al. [40] propose a non-hash method for fast text-to-image retrieval, which distills the inter-modal correlations from a heavy transformer-based model to a light dual-encoder model. However, the intra-modal similarity may not be preserved well only with the cross-modal relations distilled. Simultaneous Similarity-based Self-Distillation (S2SD) [41] solves the standard DML objective simultaneously in some high-dimensional embedding spaces and a low-dimensional space while applying multiscale knowledge distillation between these high-dimensional spaces and the low-dimensional space. Although knowledge distillation is used in S2SD to improve the generalization performance of existing Deep Metric Learning (DML) objectives in low-dimensional space, our work is quite different from it in terms of the task, motivation, and framework.

III. The Proposed Method

We elaborate the proposed method in this section. We first present the notations and problems for image hashing and cross-model hashing. Then, we formulate the knowledge distillation for intra-modal and inter-modal relevance in detail. Finally, the quantization strategy is introduced. The framework of our method, CDUH, is illustrated in Fig. 1.

A. Notation and Problem Definition

We consider the case that a training mini-batch contains N image-text paired data, i.e., $\boldsymbol{\mathcal{D}} = \{\boldsymbol{\mathcal{I}}_n, \boldsymbol{\mathcal{T}}_n\}_{n=1}^{N}$, where $\boldsymbol{\mathcal{I}}_n$ is an image and $\boldsymbol{\mathcal{T}}_n$ is a text. Here, we use the superscript I and T to denote 'image' and 'text'. The cross-modal hashing aims to learn compact binary codes $\boldsymbol{B}^{\mathrm{I}} = \{\boldsymbol{b}_n^I\}_{n=1}^{N}$ and $\boldsymbol{B}^{\mathrm{T}} = \{\boldsymbol{b}_n^T\}_{n=1}^{N}$. The $\boldsymbol{b}_n^{\mathrm{I}}$ and $\boldsymbol{b}_n^{\mathrm{T}}$ falling within $\{-1, +1\}^L$ represent the hash codes for an image and a text, where L is the length of codes. The Hamming distance (based on binary XOR operation) between hash codes for the similar image-text pair is near enough. And for image hashing, it only learns the binary codes $\boldsymbol{B}^{\mathrm{I}}$ for images in which the distance between hash codes for similar pair of images is close enough.

Assuming two pre-trained encoders for vision and language are denoted by $f^{\mathrm{I}}(\cdot)$ and $f^{\mathrm{T}}(\cdot)$ respectively, the high-dimensional embeddings of an image and a text can be obtained by

$$\begin{aligned} \boldsymbol{v}_n^{\mathrm{I}} &= f^{\mathrm{I}}(\boldsymbol{\mathcal{I}}_n) \in \mathbb{R}^D, \\ \boldsymbol{v}_n^{\mathrm{T}} &= f^{\mathrm{T}}(\boldsymbol{\mathcal{T}}_n) \in \mathbb{R}^D, \end{aligned} \tag{1}$$

where D, e.g., 1024 is the dimension of the embedding space.

Taking the features of images and texts, we focus on generating corresponding hash codes for different modalities. We employ two hash layers to project raw features into embeddings with a certain length. In detail, the hash layer is a Fully Connected (FC) layer without the active function. We formulate this process as follows:

$$\begin{aligned} f_{\boldsymbol{\theta}^*}(\cdot, L) &= FC(\cdot, L), \\ \boldsymbol{t}_n^* &= f_{\boldsymbol{\theta}^*}\left(\boldsymbol{v}_n^*, L\right) \in \mathbb{R}^L, * \in \{\mathrm{I}, \mathrm{T}\}, \end{aligned} \tag{2}$$

where the number L in FC operator is the output dimension of FC layer, and $\boldsymbol{\theta}^*$ are parameters of hash layers. Then, the binary codes, $\boldsymbol{b}_n^{\mathrm{I}}$ and $\boldsymbol{b}_n^{\mathrm{T}}$, can be obtained by conducting signum function on $\boldsymbol{t}_n^{\mathrm{I}}$ and $\boldsymbol{t}_n^{\mathrm{T}}$:

$$\boldsymbol{b}_n^* = \mathrm{sgn}\left(\boldsymbol{t}_n^*\right) \in \{-1, +1\}^L, * \in \{\mathrm{I}, \mathrm{T}\}. \tag{3}$$

However, backpropagation is infeasible for binary values at the training phase, so we use $\mathrm{quan}(\cdot, \gamma)$ to get the continuous approximation of binary codes:

$$\boldsymbol{h}_n^* = \mathrm{quan}\left(\boldsymbol{t}_n^*, \gamma\right) \in \left[-\gamma, +\gamma\right]^L, * \in \{\mathrm{I}, \mathrm{T}\}, \tag{4}$$

where $\mathrm{quan}(\cdot, \gamma)$ is an activation function used in our *clip-quan* strategy which will be introduced in the following subsection. During the inference phase, sgn is used to yield discrete binary codes.

In addition, the cosine similarity $s(\boldsymbol{u}, \boldsymbol{v})$ is used to represent the relevance between two vectors, $\boldsymbol{u}$ and $\boldsymbol{v}$, which can be formulated as

$$s(\boldsymbol{u}, \boldsymbol{v}) = \frac{\boldsymbol{u}\boldsymbol{v}^{\mathsf{T}}}{\|\boldsymbol{u}\|_2 \|\boldsymbol{v}\|_2} \in [-1, +1]. \tag{5}$$

B. Learning to Hash via Distillation

With the high-dimensional features rich in semantics, we manage to learn hash functions preserving the relevance across features in hash codes. Instead of pairwise constraint which is commonly used in most unsupervised deep hashing methods, we try to distill the knowledge, the similarity structure among samples, to hash codes.

As opposed to the vanilla knowledge distillation for classification tasks, no logits indicate the probability of each class. As a result, the original objective in knowledge distillation [38] cannot be employed directly. To break through this limitation, we develop a new way to distill the relationship among high-dimensional features to hash codes.

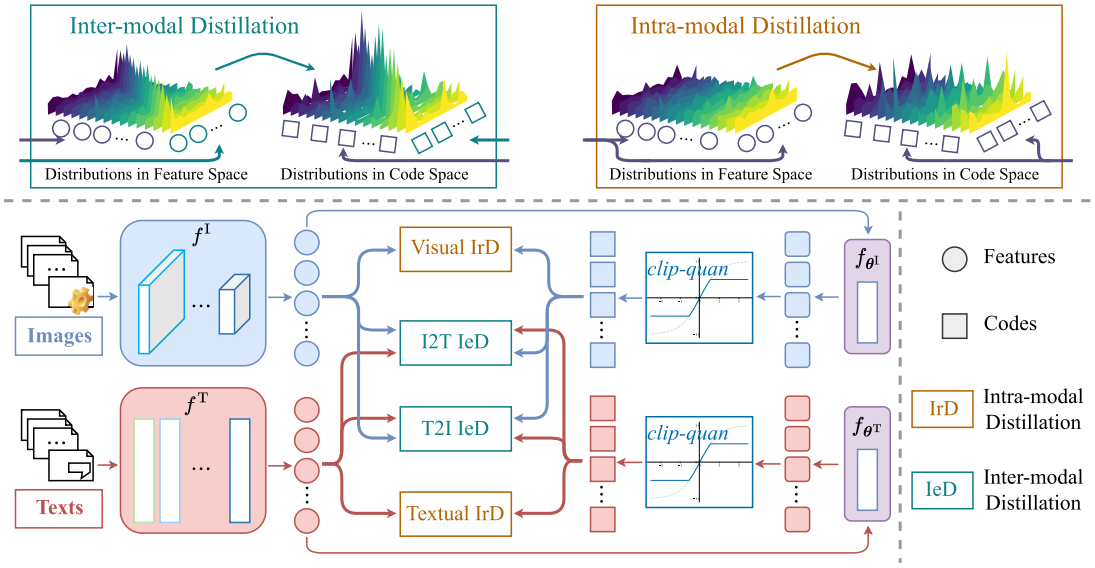


Fig. 1. The framework of our proposed methods which consists of intra-modal distillation, inter-modal distillation, and quantization. I2T/T2I is short for image-to-text/text-to-image. Image hashing retrieval only involves visual IrD.

1) Intra-Modal Distillation: We first consider to learn the hash function from the intra-modal view, which is important for unimodal hashing retrieval (involving only one modality).

Taking the visual modality as an example, we select the feature $\boldsymbol{v}_i^{\mathrm{I}}$ as an anchor and try to distill the global relationship between the anchor and the rest samples in the same batch. In concrete terms, we use discrete probability distribution $\boldsymbol{p}_i^{\mathrm{I}}$ to represent the similarity between the embedding of i-th image and those of other images in the mini-batch, and the formula is

$$\boldsymbol{p}_i^{\mathrm{I}}(j) = \frac{\exp\left(s\left(\boldsymbol{v}_i^{\mathrm{I}}, \boldsymbol{v}_j^{\mathrm{I}}\right)/\sigma\right)}{\sum_{k=1,k\neq i}^{N} \exp\left(s\left(\boldsymbol{v}_i^{\mathrm{I}}, \boldsymbol{v}_k^{\mathrm{I}}\right)/\sigma\right)}, \quad j \in \{1, 2, \ldots, N\}, j \neq i. \tag{6}$$

Then, the similarity distribution $\boldsymbol{q}_i^{\mathrm{I}}$ of hash codes in visual modality can be obtained by

$$\boldsymbol{q}_i^{\mathrm{I}}(j) = \frac{\exp\left(s\left(\boldsymbol{h}_i^{\mathrm{I}}, \boldsymbol{h}_j^{\mathrm{I}}\right)\right)}{\sum_{k=1,k\neq i}^{N} \exp\left(s\left(\boldsymbol{h}_i^{\mathrm{I}}, \boldsymbol{h}_k^{\mathrm{I}}\right)\right)}, \quad j \in \{1, 2, \ldots, N\}, j \neq i. \tag{7}$$

Given two distributions $\boldsymbol{p}_i^{\mathrm{I}}$ and $\boldsymbol{q}_i^{\mathrm{I}}$, we use the cross-entropy to measure the distance between them, which establishes the knowledge transfer from embedded feature space to compact binary code space:

$$\mathcal{H}\left(\boldsymbol{p}_i^{\mathrm{I}}, \boldsymbol{q}_i^{\mathrm{I}}\right) = -\mathbb{E}_{\boldsymbol{p}_i^{\mathrm{I}}} \log\left[\boldsymbol{q}_i^{\mathrm{I}}\right]. \tag{8}$$

And for all images in the mini-batch, we can obtain the distillation loss for intra-modal view:

$$\mathcal{L}_{\mathrm{intra}^{\mathrm{I}}} = \frac{1}{N}\sum_{i=1}^{N} \mathcal{H}\left(\boldsymbol{p}_i^{\mathrm{I}}, \boldsymbol{q}_i^{\mathrm{I}}\right). \tag{9}$$

It is noted that the temperature in vanilla knowledge distillation is generally set to a value greater than 1, which aims to produce a soft probability distribution over classes. However, in our case, the semantically similar images in a mini-batch may be very sparse, so the distribution $\boldsymbol{p}_i^{\mathrm{I}}$ is too soft to guide the hash layer to generate discriminative hash codes. Therefore, a parameter $\sigma \in (0, 1)$ in (6) is used to reduce the smoothness of the teacher distribution, which makes intra-modal distillation from $\boldsymbol{p}_i^{\mathrm{I}}$ to $\boldsymbol{q}_i^{\mathrm{I}}$ easier. And the temperature in the student distribution $\boldsymbol{q}_i^{\mathrm{I}}$ is set to 1.

The intra-modal loss for textual modality, i.e., $\mathcal{L}_{\mathrm{intra}^{\mathrm{T}}}$, can be obtained in the same way. Finally, we get the objective for intra-modal distillation as

$$\mathcal{L}_{\mathrm{intra}} = \frac{1}{2}\left(\mathcal{L}_{\mathrm{intra}^{\mathrm{I}}} + \mathcal{L}_{\mathrm{intra}^{\mathrm{T}}}\right). \tag{10}$$

2) Inter-Modal Distillation: In addition to retaining semantic structure within one modality, preserving correlations between different modalities is more crucial for cross-modal retrieval.

Similar to the distillation of intra-modal view, we use the image feature $\boldsymbol{v}_i^{\mathrm{I}}$ as an anchor, and transfer the correlations between the anchor and all text features in the same batch for image-to-text hashing retrieval. Specifically, the distribution $\boldsymbol{cp}_i^{\mathrm{I2T}}$, that indicates the relevance score between the i-th image and all texts in feature space, can be computed by

$$\boldsymbol{cp}_i^{\mathrm{I2T}}(j) = \frac{\exp\left(s\left(\boldsymbol{v}_i^{\mathrm{I}}, \boldsymbol{v}_j^{\mathrm{T}}\right)/\tau\right)}{\sum_{k=1}^{N} \exp\left(s\left(\boldsymbol{v}_i^{\mathrm{I}}, \boldsymbol{v}_k^{\mathrm{T}}\right)/\tau\right)}, \quad j \in \{1, 2, \ldots, N\}. \tag{11}$$

And the corresponding probability distribution in hash space is

$$\boldsymbol{cq}_i^{\mathrm{I2T}}(j) = \frac{\exp\left(s\left(\boldsymbol{h}_i^{\mathrm{I}}, \boldsymbol{h}_j^{\mathrm{T}}\right)\right)}{\sum_{k=1}^{N} \exp\left(s\left(\boldsymbol{h}_i^{\mathrm{I}}, \boldsymbol{h}_k^{\mathrm{T}}\right)\right)}, \quad j \in \{1, 2, \ldots, N\}. \tag{12}$$

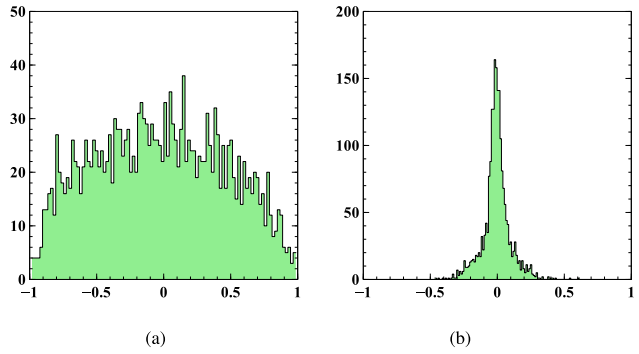


Fig. 2. The histograms of continuous hash codes in the initial state.

The cross-entropy between them can be expressed as

$$\mathcal{H}\left(\boldsymbol{cp}_i^{\mathrm{I2T}}, \boldsymbol{cq}_i^{\mathrm{I2T}}\right) = -\mathbb{E}_{\boldsymbol{cp}_i^{\mathrm{I2T}}} \log\left[\boldsymbol{cq}_i^{\mathrm{I2T}}\right]. \tag{13}$$

The parameter τ in (11) has the same effect as σ in (6) and it reduces the difficulty of distillation in the inter-modal aspect.

The distillation loss for image-to-text in the mini-batch is defined as

$$\mathcal{L}_{\mathrm{inter}^{\mathrm{I2T}}} = \frac{1}{N}\sum_{i=1}^{N} \mathcal{H}\left(\boldsymbol{cp}_i^{\mathrm{I2T}}, \boldsymbol{cq}_i^{\mathrm{I2T}}\right). \tag{14}$$

And the text-to-image distillation loss $\mathcal{L}_{\mathrm{inter}^{\mathrm{T2I}}}$ can be computed in the same way. Then, the overall loss for inter-modal view is

$$\mathcal{L}_{\mathrm{inter}} = \frac{1}{2}\left(\mathcal{L}_{\mathrm{inter}^{\mathrm{I2T}}} + \mathcal{L}_{\mathrm{inter}^{\mathrm{T2I}}}\right). \tag{15}$$

By minimizing the batchwise distillation loss $\mathcal{L}_{\mathrm{intra}}$ and $\mathcal{L}_{\mathrm{inter}}$ between high-dimensional representations and hash codes, the relevance relationship of intra-modality and inter-modality is both maximally reconstructed in relaxed hash codes respectively.

C. Quantization

In order to obtain exactly binary codes, we analyze the statistics of relaxed values activated by tanh. Specifically, we randomly sample images in a mini-batch from MIRFlickr25K and get the continuous codes with code length of 32, $\tanh\left(\boldsymbol{t}_n^{\mathrm{I}}\right)$. As shown in Fig. 2a, we get a histogram of values on all bits of these hash codes. The same histogram of hash codes, $\tanh\left(\boldsymbol{t}_n^{\mathrm{I}}\right)$ and $\tanh\left(\boldsymbol{t}_n^{\mathrm{T}}\right)$, in cross-modal hashing is shown in Fig. 2b. It can be seen that most of these values are around zero and the value range still has a margin to ± 1. Therefore, quantizing the activated values to ± 1 may increase the inconsistency between continuous and discrete hash codes. Moreover, the gradient of tanh is almost zero when its value is close to ± 1, which makes the optimization gradually difficult as the quantization progresses.

To mitigate the above problems, we develop a new quantization strategy named *clip-quan*. It is easy to know that, given a set of relaxed codes, the metric relationship is consistent in Hamming space for discrete codes quantized to different values, e.g. ± 1 and ± 0.5. Therefore, we attempt to quantize the relaxed codes to an upper bound instead of ± 1. We define a new activation function $\mathrm{quan}(\cdot, \gamma)$,

$$\mathrm{quan}(x, \gamma) = \begin{cases} -\gamma, & \tanh(x) \le -\gamma, \\ \tanh(x), & -\gamma < \tanh(x) < +\gamma, \\ +\gamma, & \tanh(x) \ge +\gamma, \end{cases} \tag{16}$$

which clamps the output of tanh into the range $\left[-\gamma, +\gamma\right]$ and $\gamma \in (0, 1)$.

Then, a quantization loss is adopted to minimize the quantization error and retain the discreteness of hash codes. The quantization loss for hash codes is

$$\mathcal{L}_{\mathrm{quan}*} = \frac{1}{NL}\sum_{n=1}^{N} \left\| \left|\boldsymbol{h}_n^*\right| - \gamma \mathbf{1} \right\|_2^2, * \in \{\mathrm{I}, \mathrm{T}\}, \tag{17}$$

and overall quantization loss for both modalities can be written as

$$\mathcal{L}_{\mathrm{quan}} = \frac{1}{2}\left(\mathcal{L}_{\mathrm{quan}^{\mathrm{I}}} + \mathcal{L}_{\mathrm{quan}^{\mathrm{T}}}\right). \tag{18}$$

Compared with quantizing values to ± 1, tanh has larger gradient when its value is around $\pm\gamma$, so the vanishing gradient problem is alleviated. Additionally, through the quantization strategy described above, the discrepancy between continuous codes and binary codes is reduced as much as possible.

The distributions of activated values with different backbones in the initial state are not the same, as seen in Fig. 2a and Fig. 2b. As a result, setting γ that is too small may discard most values, and setting γ too large increases the inconsistency between the hash codes and continuous values and leads to the vanishing gradient problem. So, γ is set to 0.5 in our experiment empirically. Besides, when γ is set to 1, this quantization strategy degenerates into the conventional quantization method.

D. Objective Function

For the scenario of image retrieval, only the similarity structure within the image modality needs to be distilled and transferred into hash codes. So the optimization objective for image hashing is $\min_{\boldsymbol{\theta}^{\mathrm{I}}} \mathcal{L}_{\mathrm{I}}$ and

$$\mathcal{L}_{\mathrm{I}} = \mathcal{L}_{\mathrm{intra}^{\mathrm{I}}} + \alpha \mathcal{L}_{\mathrm{quan}^{\mathrm{I}}}. \tag{19}$$

In the case of cross-modal retrieval, both intra-modal and inter-modal correlations need to be captured and preserved in hash codes. Therefore, the optimization objective for cross-modal hashing is $\min_{\boldsymbol{\theta}^{\mathrm{I}}, \boldsymbol{\theta}^{\mathrm{T}}} \mathcal{L}_{\mathrm{C}}$ and

$$\mathcal{L}_{\mathrm{C}} = \mathcal{L}_{\mathrm{intra}} + \mathcal{L}_{\mathrm{inter}} + \beta \mathcal{L}_{\mathrm{quan}}. \tag{20}$$

The trade-off parameters α and β weight the quantization loss terms for image hashing and cross-modal hashing, respectively.

IV. Experiments

In this section, we conduct extensive experiments for image hashing and cross-modal hashing on various public datasets. The results and analyses can validate the effectiveness of our proposed method.

TABLE II

THE PERFORMANCE COMPARISON OF IMAGE RETRIEVAL ON MIRFLICKR25K, MSCOCO, AND NUSWIDE WITH DIFFERENT CODE LENGTHS IN TERMS OF MAP AND NDCG

	MAP												NDCG											
	MIRFlickr25K				MSCOCO				NUSWIDE				MIRFlickr25K				MSCOCO				NUSWIDE			
	16	32	64	128	16	32	64	128	16	32	64	128	16	32	64	128	16	32	64	128	16	32	64	128
PUDH	69.4	74.3	76.7	78.0	69.9	72.7	74.0	75.9	66.2	70.9	73.5	74.7	41.6	46.1	49.3	50.2	36.8	39.8	41.3	42.6	36.5	40.6	43.3	44.1
SSDH	66.4	68.4	70.8	69.6	61.8	67.0	72.7	74.7	50.2	52.0	58.7	61.9	38.6	42.2	44.5	44.4	27.8	33.9	37.6	40.7	24.0	29.9	35.8	39.0
GreedyHash	66.9	66.8	69.3	71.9	67.6	69.5	71.1	73.1	57.8	60.0	63.5	67.8	38.4	39.1	41.5	44.2	33.6	36.1	37.8	39.7	32.0	34.2	38.5	41.3
TBH	72.1	74.5	74.2	74.9	69.6	72.4	73.2	73.8	68.4	70.1	71.2	71.1	43.2	47.3	45.3	46.7	38.4	40.9	42.2	42.1	36.1	37.6	38.7	39.2
Bi-half	71.7	74.0	76.4	77.7	71.6	73.9	75.2	76.2	68.2	70.8	73.8	74.6	43.8	46.3	49.4	50.4	36.9	40.1	40.5	42.4	38.7	41.3	43.9	44.8
CUHM	75.0	76.7	77.4	77.8	**73.9**	75.0	76.0	76.7	69.9	72.3	74.0	75.1	48.4	50.4	51.3	51.8	**39.6**	41.1	41.9	42.7	38.6	41.3	43.1	44.2
MLS^{3}RUDH	74.4	75.8	76.6	77.5	72.8	74.5	75.2	76.1	**71.6**	73.3	74.2	73.9	47.8	49.0	49.5	51.2	38.4	39.8	40.8	42.0	**39.8**	41.4	42.0	43.4
Ours	**76.2**	**77.7**	**78.5**	**79.1**	72.3	**75.7**	**77.0**	**78.1**	70.3	**73.5**	**75.5**	**76.4**	**49.4**	**50.9**	**51.7**	**52.4**	37.3	**42.3**	**44.1**	**45.0**	39.5	**42.6**	**45.0**	**46.2**

Best results are in bold. The setting in other tables is the same.

TABLE III

THE PERFORMANCE COMPARISON OF CROSS-MODAL RETRIEVAL ON MSCOCO WITH DIFFERENT CODE LENGTHS IN TERMS OF MAP AND NDCG

	MAP								NDCG							
	Image-to-Text				Text-to-Image				Image-to-Text				Text-to-Image			
	16	32	64	128	16	32	64	128	16	32	64	128	16	32	64	128
UGACH	78.49	79.34	79.47	79.89	78.64	79.80	79.68	80.00	42.80	43.74	44.28	44.55	42.60	43.97	43.73	44.10
UKD	76.83	76.64	77.13	76.95	75.02	76.30	76.61	76.51	40.49	40.31	41.17	41.47	39.13	40.87	41.05	41.16
DJSRH	70.73	75.88	79.15	81.21	71.26	76.34	79.31	81.26	34.54	38.55	41.20	44.14	35.30	38.35	41.29	43.75
DSAH	78.44	82.01	83.38	84.29	78.28	81.45	83.06	84.29	41.77	44.14	46.29	47.07	41.33	43.33	45.92	46.90
JDSH	78.90	80.27	82.34	83.22	79.07	79.93	82.29	83.10	41.47	42.47	44.70	46.39	41.11	42.09	44.52	45.88
KDCMH	79.00	81.36	82.63	83.57	78.88	81.43	82.47	83.61	42.93	45.84	46.53	**47.60**	42.39	44.89	45.90	46.47
DGCPN	80.60	81.59	82.67	83.58	81.02	82.02	82.65	83.51	**43.51**	44.53	46.36	47.10	**43.39**	44.60	46.31	46.67
Ours	**81.35**	**83.46**	**84.23**	**84.47**	**81.50**	**83.59**	**84.36**	**84.53**	42.82	**45.90**	**46.95**	47.21	42.69	**45.65**	**46.36**	**46.97**

TABLE IV

THE PERFORMANCE COMPARISON OF CROSS-MODAL RETRIEVAL ON IAPR TC-12 WITH DIFFERENT CODE LENGTHS IN TERMS OF MAP AND NDCG

	MAP								NDCG							
	Image-to-Text				Text-to-Image				Image-to-Text				Text-to-Image			
	16	32	64	128	16	32	64	128	16	32	64	128	16	32	64	128
UGACH	50.85	52.22	52.55	53.07	50.57	52.17	52.61	52.87	47.33	49.04	49.38	49.65	47.00	48.85	49.43	49.37
UKD	48.12	49.20	49.41	50.27	48.86	50.17	51.45	51.72	46.38	47.32	47.59	48.07	45.96	47.21	48.60	49.10
DJSRH	43.13	46.42	48.56	50.85	42.71	45.81	48.68	50.79	39.88	42.33	43.94	46.03	39.54	41.99	43.98	45.81
DSAH	48.91	50.12	52.64	54.32	48.41	49.96	52.37	54.17	44.09	44.88	47.30	48.99	43.31	44.64	46.87	48.84
JDSH	47.86	49.27	51.55	53.39	47.74	49.15	51.38	53.32	42.94	44.25	46.42	48.36	42.71	44.22	46.28	48.27
KDCMH	50.81	52.96	54.27	55.00	50.34	52.35	53.63	54.52	46.93	48.88	50.27	50.76	46.64	48.47	49.73	50.22
DGCPN	51.50	53.09	54.69	55.18	51.22	52.86	54.67	55.13	47.14	48.67	50.32	50.96	46.84	48.40	50.14	50.80
Ours	**52.52**	**54.92**	**55.85**	**56.49**	**52.31**	**54.64**	**55.57**	**56.18**	**47.97**	**50.38**	**51.68**	**52.35**	**47.66**	**50.08**	**51.20**	**52.07**

A. Models, Datasets, and Implementation Details

1) Models: For unsupervised image hashing, we adopt the pre-trained AlexNet [50] as the backbone, parameters of which are frozen in the training process. The hash layer is appended to `fc7` to generate hash codes. For unsupervised cross-modal hashing, most existing cross-modal hashing methods obtain the textual features via Multilayer Perceptron (MLP). Compared with the heavy encoder for images, such a shallow model for texts cannot fully capture the high-level semantics in texts, which may diminish the role of textual modality and hinder the model learn common embedded space with rich semantics. Therefore, we use a transformer-based encoder to extract high-level features from raw texts to obtain rich semantic relationship. Specifically, for cross-modal hashing retrieval, we employ the image encoder and text encoder in CLIP, a pre-trained vision-language model which is well known for its excellent transferability, to obtain the visual and textual representations respectively. As described in CLIP, the image encoder is the modified ResNet-50 [43] in which the global average pooling layer is replaced with an attention pooling mechanism [35], and the text encoder is a Transformer [44] with 12 layers, 512D hidden size, and 8 attention heads. The last token in the top layer is treated as the feature embedding of the text. The parameters of the image and text encoder are frozen during the training procedure.

TABLE V
THE PERFORMANCE COMPARISON OF IMAGE RETRIEVAL ON MSCOCO WITH CROSS-MODAL HASHING METHODS

	MAP				NDCG			
	16	**32**	**64**	**128**	**16**	**32**	**64**	**128**
UGACH	78.51	79.28	79.46	80.03	43.93	44.93	44.98	45.91
UKD	76.71	77.80	78.58	79.25	41.00	42.00	43.91	44.91
DJSRH	76.17	78.41	80.54	82.18	39.98	42.91	43.95	45.98
DSAH	79.61	82.39	83.71	84.61	43.93	45.91	47.69	**48.93**
JDSH	79.19	80.48	82.62	83.32	42.20	43.51	46.91	47.96
KDCMH	79.61	81.57	83.02	83.85	44.03	46.73	47.54	48.42
DGCPN	80.99	82.06	82.91	83.77	44.16	45.40	47.49	48.22
Ours	**81.51**	**83.52**	**84.41**	**84.70**	**45.08**	**47.17**	**48.31**	48.62

TABLE VI
THE PERFORMANCE COMPARISON OF IMAGE RETRIEVAL ON IAPR TC-12 WITH CROSS-MODAL HASHING METHODS

	MAP				NDCG			
	16	**32**	**64**	**128**	**16**	**32**	**64**	**128**
UGACH	51.40	52.86	53.29	53.76	**48.91**	49.99	50.95	50.96
UKD	50.58	52.79	53.29	54.12	47.90	50.00	50.92	51.91
DJSRH	46.85	49.85	50.77	52.92	43.93	49.55	46.91	47.99
DSAH	50.22	51.06	53.83	55.61	45.92	46.00	48.96	50.96
JDSH	48.94	50.35	52.93	54.66	43.99	45.95	47.99	49.99
KDCMH	52.12	54.26	55.56	56.28	48.54	50.65	52.06	52.53
DGCPN	52.42	54.14	55.74	56.36	48.21	49.87	51.59	52.34
Ours	**53.57**	**56.02**	**56.85**	**57.36**	48.76	**52.02**	**52.89**	**53.71**

2) Datasets: We carry out experiments for image hashing on three datasets: MIRFlickr25K [45], MSCOCO [47], and NUSWIDE [46]. MIRFlickr25K is a multi-label dataset with 24581 images and each image is annotated with at least one of 24 unique labels. 2000 images are randomly selected as the query set with the rest images as the database set. And 5000 images in the database are randomly chosen as the training set.

MSCOCO is a large-scale dataset for object detection, segmentation, and image caption. We use the 2017 version dataset including 118287 images and each image is labeled by some of 80 categories. We get a subset containing 92306 images with the top 20 frequent concepts. The numbers of images for query, database, and training are the same as those in MIRFlickr25K.

NUSWIDE is a multi-label dataset like MIRFlickr25K which provides 269648 images and each image is tagged with some classes in 81 categories. We select 195834 images whose annotations include the most 21 frequent categories. We sample 5000 images as query images from this subset and the rest images are left as the database in which 10000 images are randomly selected as training images.

For cross-modal hashing retrieval, two public datasets are used in our experiments. One is MSCOCO used in image hashing, another is IAPR TC-12 [48] datasets. We only use the first caption of each image in the subset of MSCOCO and obtain 92306 image-text pairs. We sample 2000 pairs as query samples and the rest pairs are used as the database. In the database set, 10000 image-text pairs are randomly chosen as training samples.

IAPR TC-12 consists of 20000 natural images and each image is associated with some text captions and some of 275 categories. A subset of images with 25 the most frequent categories are selected. And then we only use the first sentence in captions of each image, resulting in a total of 18938 image-text pairs. How images are selected for different subsets is the same as MSCOCO.

It is noted that we don't carry out cross-modal hashing retrieval experiments on MIRFlickr25K and NUSWIDE datasets because the textual encoder employed in our experiments takes the raw text as input, but these two datasets just have several tags in textual modality.

3) Implementation Details: We conduct all experiments on a workstation with a single NVIDIA GeForce RTX 2080 Ti GPU. All images are resized into 224×224 as inputs. Adam is used to update parameters in hash layers. The learning rate is fixed to 3×10^{-5}. The batch size is set to 32 in all experiments. The feature dimension D are 4096 and 1024 in AlexNet and CLIP, respectively. For experiments of cross-modal hashing, we further finetune the pre-trained CLIP model on MSCOCO and IAPR TC-12 via contrastive learning described in CLIP.

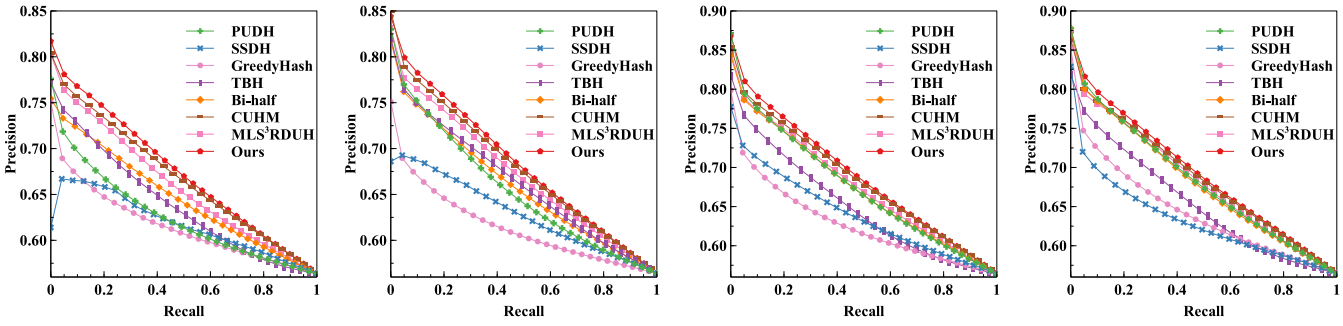
B. Baselines and Evaluation Metrics

We compare our method with some unsupervised image hashing retrieval methods, including PUDH [6], SSDH [4], GreedyHash [16], TBH [9], Bi-half [17], CUHM [49], and MLS^{3}RDUH [8]. Moreover, for unsupervised cross-modal hashing retrieval, we compare our method with UGACH [30], UKD [27], DJSRH [25], DSAH [28], JDSH [29], DGCPN [32], and KDCMH [33]. For a fair comparison, we retrain and evaluate all these methods with identical settings. Additionally, the deep features of images and texts used in different methods are extracted through the same backbones, i,e, AlexNet or encoders in CLIP.

To evaluate the performance of different methods on the image retrieval and cross-modal retrieval tasks, we adopt three standard metrics: Mean Average Precision (MAP), precision-recall (PR) curve, and Normalized Discounted Cumulative Gain (NDCG). MAP and PR curves are widely used in hashing retrieval, and NDCG is more convincing to evaluate the performance on multi-label datasets than MAP [49]. The number of returned points is set to 5000 for both MAP and NDCG.

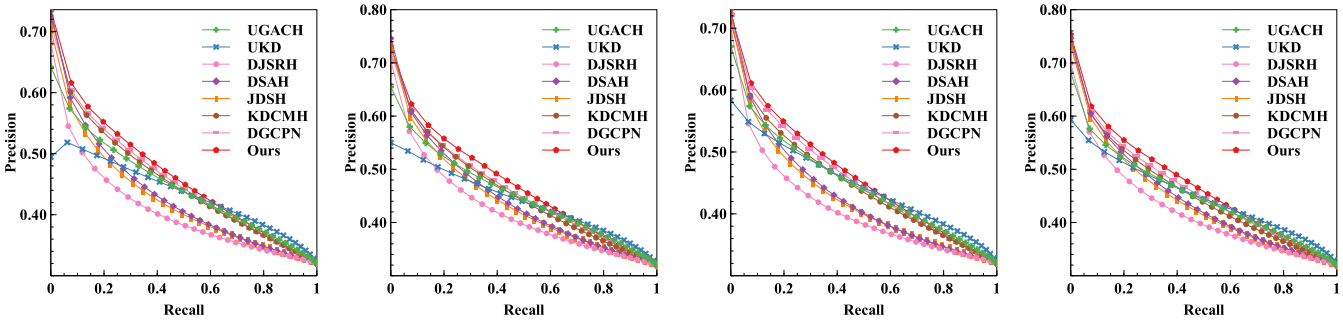
C. Results and Discussions

1) Image Retrieval: The performances in terms of MAP and NDCG on different datasets with various code lengths are reported in Table II. As shown in this table, our method outperforms the other deep unsupervised hashing methods except for few cases. Specifically, although our performance is lower than the current SOTA method MLS^{3}RDUH in few cases, our method is more stable across different datasets than it. Compared with MLS^{3}RDUH, our method achieves an average improvement of 1.80%, 1.10%, and 0.70% in MAP



(a) PR curve @ 16-bits (b) PR curve @ 32-bits (c) PR curve @ 64-bits (d) PR curve @ 128-bits

Fig. 3. The PR curves for image retrieval on MIRFlickr25K.



(a) I2T PR curve @ 64-bits (b) I2T PR curve @ 128-bits (c) T2I PR curve @ 64-bits (d) T2I PR curve @ 128-bits

Fig. 4. The PR curves for cross-modal retrieval on IAPR TC-12.

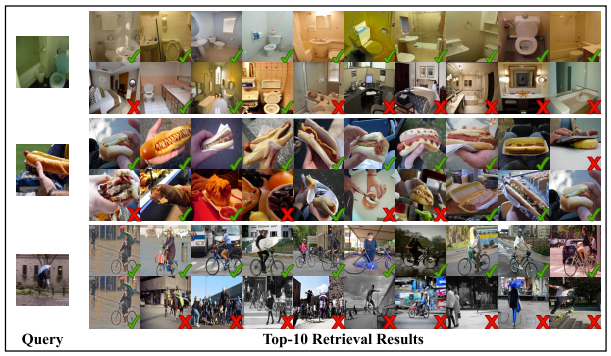


Fig. 5. Examples of top-10 items for image retrieval on MSCOCO. For each query image, the first row is our result and the second row is the result of MLS^{3}RDUH.

Fig. 6. Examples of top-10 items for text-to-image retrieval on MSCOCO. For each query text, the first row is our result and the second row is the result of DGCPN.

and 1.70%, 1.90%, and 1.70% in NDCG on MIRFlickr25K, MSCOCO, and NUSWIDE. The PR curves of all methods on MIRFlickr25K datasets with different code lengths are shown in Fig. 3. It can be seen that our method can achieve superior performance. Additionally, we visualize some results of our method and MLS^{3}RDUH for image retrieval with 128-bits codes on MSCOCO in Fig. 5. These results demonstrate the effectiveness of our method for unimodal hashing retrieval.

2) Cross-Modal Retrieval: For all unsupervised deep cross-modal hashing retrieval methods, we conduct experiments on the Image-to-Text (I2T), Text-to-Image (T2I), and Image-to-Image (I2I) retrieval tasks and the performances in MAP and NDCG with different code lengths on different datasets are reported in Table III, IV, V and VI. In general, our proposed method achieves the best performance in most cases regardless of the criteria and tasks. On MSCOCO, compared to the DGCPN, our method yields 1.27%, 1.20%, and 1.10% higher average MAP and 0.35%, 0.18%, and 0.98% higher average NDCG for I2T, T2I, and I2I. On IAPR TC-12, our method outperforms DGCPN by 1.68%, 1.97%, and 1.29% in average MAP and 1.39%, 1.49%, and 1.34% in average NDCG for I2T, T2I, and I2I. Moreover, we plot the PR curves on IAPR TC-12 for I2T and T2I of all methods in Fig. 4. It can be observed that the curves of our methods are higher than those of the others.

TABLE VII

THE PERFORMANCE COMPARISON OF IMAGE RETRIEVAL ON MIRFLICKR25K WITH DIFFERENT OBJECTIVES

	MAP@5000				NDCG@5000			
	16	32	64	128	16	32	64	128
Pairwise loss	70.49	71.26	72.50	73.84	41.76	42.88	45.03	46.25
Distillation loss	76.18	77.70	78.52	79.06	49.36	50.91	51.72	52.37

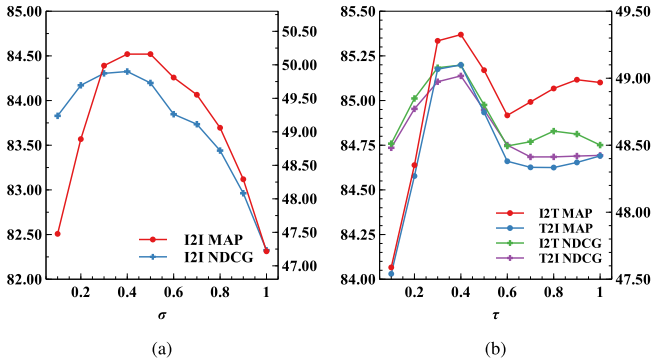
Additionally, in Fig. 6, we illustrate several results for text-to-image retrieval with 128-bits codes on MSCOCO, which shows the superiority of our method.

3) Effect of Correlation Distillations: To further illustrate the advantages of the correlation distillation over pairwise similarity constraint, we replace the intra-modal distillation loss in the case of image hashing retrieval with the pairwise similarity loss, i.e., $1/N^2 \sum_{i=1}^{N} \sum_{j=1}^{N} \left(s_{ij}^f - s_{ij}^h\right)^2$, where s_{ij}^f and s_{ij}^h are cosine similarity calculated in feature space and hash space, respectively. The performance of different code lengths with different objectives on MIRFlickr25K is shown in Table VII. It can be seen that the performance of distillation loss is much better than that of pairwise loss, indicating that our proposed correlation distillation can exploit the global relationship among features and preserve it in the hash code space efficiently.

4) Component Analysis: Our method of learning the hash function mainly consists of three components: intra-modal distillation, inter-modal distillation, and quantization. As analyzed in Section III, $\mathcal{L}_{\text{intra}}$ helps to learn the hash function via intra-modal similarity reconstruction, and $\mathcal{L}_{\text{inter}}$ is the further help by distilling the inter-modal correlations. $\mathcal{L}_{\text{quan}}$ assists in learning the hash function from the perspective of keeping hash codes discrete. We study the contribution of each component on the cross-modal retrieval task.

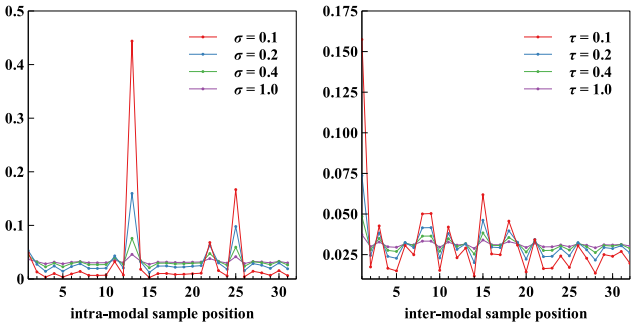
The performances on MSCOCO with different settings are reported in Table VIII. From the results, we can obtain the following observations. There is little decline in the retrieval performance without intra-modal distillation, showing that our method can get good results on I2T, T2I, and I2I tasks only by maintaining the inter-modal similarity relationship. Intuitively, the inter-modal distillation can implicitly guide the learning of the intra-modal relationship. For example, the distribution $\boldsymbol{cp}_i^{\text{I2T}}$ in (11) also implies the similarity among different texts. In the absence of inter-modal distillation, the performance of cross-modal retrieval suffers a lot, but the performance of intra-modal retrieval is almost not degraded. This demonstrates that, while intra-modal distillation cannot guarantee the establishment of inter-modal relations, it plays an important role in mining and preserving the intra-modal similarity relationship. Additionally, without the constraint of quantization loss, the codes generated by the model lack discreteness, leading to performance degradation in hash retrieval.

5) Effect of Temperature Parameters: To investigate the impact of temperature parameters of σ in (6) and τ in (11) on the intra-modal distillation and inter-modal distillation, respectively, we conduct comprehensive experiments on MSCOCO.



(a) (b)

Fig. 7. The effects of the temperature σ and τ. For a better view, the left ordinate denotes MAP and the right one denotes NDCG.



(a) (b)

Fig. 8. The intra-modal distribution $\boldsymbol{p}_0^{\text{I}}$ and inter-modal distribution $\boldsymbol{cp}_0^{\text{I2T}}$ with different temperatures in terms of the first sample in a mini-batch.

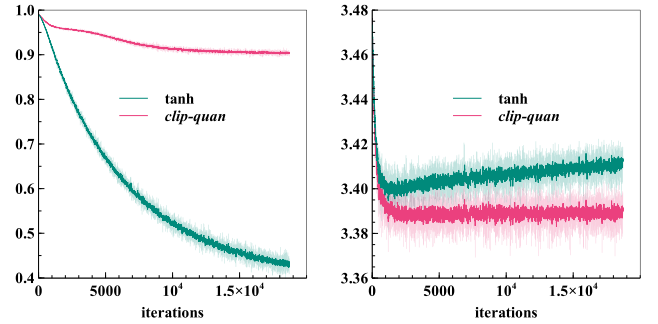
Specifically, $\mathcal{L}_{\text{intra}}$ and $\mathcal{L}_{\text{inter}}$ are involved independently in the loss function when exploring the effects of σ and τ, where $\sigma, \tau \in \{0.1, 0.2, 0.3, \cdots, 0.9, 1.0\}$. The MAP and NDCG results with the case of 32 bits are reported in Fig. 7 and they are both computed using continuous hash codes without clipping to exclude the influence of quantization.

As shown in Fig. 7a and 7b, proper σ and τ bring a certain improvement for the unimodal and cross-modal retrieval. Specifically, the temperature σ and τ adjust the smoothness of the intra-modal similarity distribution and the inter-modal similarity distribution of high-dimensional representations, respectively. Temperature parameters that are too small make the similarity distribution of features sharp, which brings errors to the distribution of the teacher and reduces the knowledge it carries. Conversely, a larger temperature will make the distribution smoother, which increases the difficulty for the student to distill the similarity information. Moreover, with rising temperatures, the performance degradation for unimodal retrieval is more pronounced than that for cross-modal retrieval. This is because, in a mini-batch, there are few similar samples within one modality, so the intra-modal distribution maybe is too uniform when σ is too large. But this problem does not exist for inter-model distribution, because there are at least one pair of similar samples from different modalities, e.g., the text and image from the same data point. Examples of intra-modal and inter-modal distributions

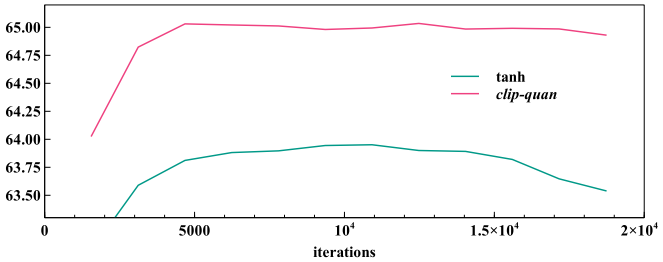
TABLE VIII
RETRIEVAL PERFORMANCE ON MSCOCO FOR DIFFERENT SETTINGS IN TERMS OF MAP AND DNCG

	MAP						NDCG					
	Image-to-Text		Text-to-Image		Image-to-Image		Image-to-Text		Text-to-Image		Image-to-Image	
	32	64	32	64	32	64	32	64	32	64	32	64
Ours	83.46	84.23	83.59	84.36	83.52	84.41	45.90	46.95	45.65	46.36	47.17	48.31
w/o $\mathcal{L}_{\text{inter}}$	$70.32_{\downarrow 13.1}$	$67.94_{\downarrow 16.3}$	$52.34_{\downarrow 31.3}$	$62.31_{\downarrow 22.1}$	$81.00_{\downarrow 2.53}$	$82.48_{\downarrow 1.93}$	$28.15_{\downarrow 17.8}$	$31.27_{\downarrow 15.7}$	$23.30_{\downarrow 22.4}$	$28.00_{\downarrow 18.4}$	$44.93_{\downarrow 2.23}$	$46.10_{\downarrow 2.21}$
w/o $\mathcal{L}_{\text{intra}}$	$82.78_{\downarrow 0.67}$	$84.40_{\uparrow 0.17}$	$83.55_{\downarrow 0.04}$	$84.13_{\downarrow 0.24}$	$83.22_{\downarrow 0.30}$	$84.50_{\uparrow 0.09}$	$46.24_{\uparrow 0.33}$	$47.23_{\uparrow 0.28}$	$45.56_{\downarrow 0.09}$	$46.63_{\uparrow 0.27}$	$47.01_{\downarrow 0.16}$	$48.13_{\downarrow 0.17}$
w/o $\mathcal{L}_{\text{quan}}$	$82.48_{\downarrow 0.98}$	$83.72_{\downarrow 0.50}$	$82.40_{\downarrow 1.19}$	$83.66_{\downarrow 0.70}$	$82.73_{\downarrow 0.79}$	$83.94_{\downarrow 0.47}$	$44.52_{\downarrow 1.38}$	$46.48_{\downarrow 0.47}$	$43.98_{\downarrow 1.67}$	$46.29_{\downarrow 0.07}$	$45.78_{\downarrow 1.39}$	$47.71_{\downarrow 0.59}$
w/o clip	$82.41_{\downarrow 1.05}$	$83.12_{\downarrow 1.10}$	$81.53_{\downarrow 2.07}$	$83.26_{\downarrow 1.10}$	$82.78_{\downarrow 0.74}$	$83.62_{\downarrow 0.79}$	$45.24_{\downarrow 0.67}$	$46.72_{\downarrow 0.23}$	$44.48_{\downarrow 1.17}$	$46.41_{\uparrow 0.05}$	$46.29_{\downarrow 0.87}$	$47.78_{\downarrow 0.53}$

The subscript means the drop of performance compared with our method in default settings.

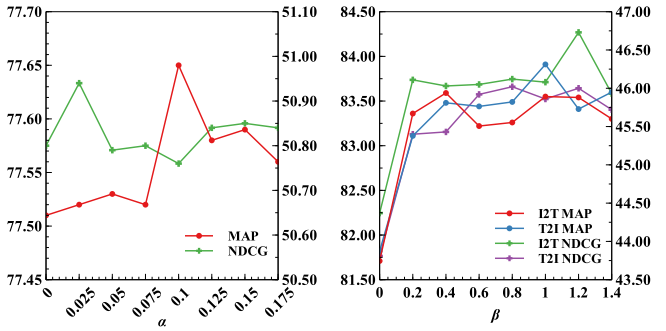


(a) The average derivative of tanh (b) The value of inter-modal distillation loss



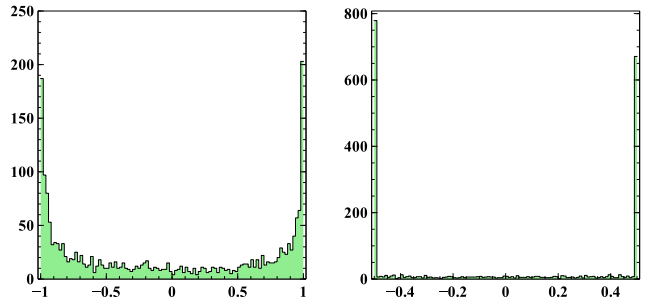
(c) The average of evaluation metrics

Fig. 9. Values of tanh and *clip-quan* through training process.



(a) (b)

Fig. 10. The effects of the temperature α and β. For a better view, the left ordinate denotes MAP and the right one denotes NDCG.



(a) tanh ($\gamma = 1$) (b) *clip-quan* ($\gamma = 0.5$)

Fig. 11. The histograms of continuous hash codes in the image hashing experiment.

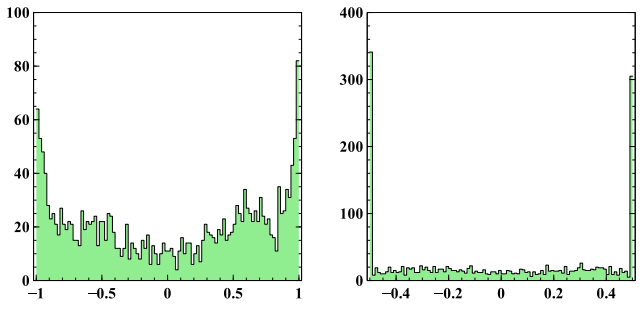
at different temperatures shown in Fig. 8 can help illustrate the above conclusions.

In experiments of image hashing retrieval, $\sigma = 0.1$ for MIRFlickr25K, $\sigma = 0.4$ for MSCOCO, and $\sigma = 0.2$ for NUSWIDE. In experiments of cross-modal hashing retrieval, $\sigma = \tau = 0.4$ and $\sigma = \tau = 0.2$ for MSCOCO and IAPR TC-12, respectively.

6) Quantization: As described previously, the quantization approach *clip-quan* can assist the model to generate exact binary codes while distilling the similarity in hash code space. In Table VIII, we show the retrieval performance of hash codes obtained by tanh (w/o clip, i.e., quantizing value to ± 1 without clipping by setting γ to 1). The proposed *clip-quan* outperforms the conventional quantization way by margins of 1.14% in average MAP on MSCOCO.

In addition to quantitative analysis, we compare the change of some values of *clip-quan* and tanh through the training process on MSCOCO with code length of 32. The average derivative of tanh, inter-modal distillation loss, and the average of evaluation metrics (MAP and NDCG) are shown in Fig. 9. The *clip-quan* can obtain more stable derivative values, alleviating the problem of vanishing gradient in tanh. Besides, it enables smaller training loss than tanh and makes the optimization easier, resulting in better performance during training.

What's more, we plot the histograms of continuous hash codes to emphasize the effectiveness of our quantization strategy intuitively. Specifically, Fig. 11a and 11b show the histograms of hash codes obtained with the tanh and the *clip-quan* in the experiment of image hashing, respectively. The



(a) $\tanh$ ($\gamma = 1$) (b) *clip-quan* ($\gamma = 0.5$)

Fig. 12. The histograms of continuous hash codes in the cross-modal hashing experiment.

same histograms of the hash codes in cross-modal hashing are shown in Fig. 12. It can be found that *clip-quan* can help to obtain more compact hash codes. In addition, the distribution of the continuous hash codes in the initial state is different because the two experiments employ different backbone networks. However, it has no impact on obtaining compact hash codes, which illustrates the stability of this approach for different initial distributions.

Besides, we explore the sensitivity of hyper-parameters α and β which balance the weight of the quantization loss items for image hashing and cross-modal hashing respectively. The performance with the case of 32 bits on MIRFlickr25K and MSCOCO are shown in Fig. 10a and 10b, and it can be found that our method is not sensitive to weight for the quantization loss. We set α to 0.1 and β to 1.2 in our experiments.

V. Conclusion

In this paper, we propose a novel method for deep unsupervised image hashing retrieval and unsupervised cross-modal hashing retrieval. The proposed method makes use of a batch-wise distillation loss to transfer the similarity distribution of features into hash codes, which exploits the global intra-modal and inter-modal relationship and preserves it in the hash code space efficiently. Furthermore, with the proposed *clip-quan* quantization strategy, the model can generate more compact binary codes and improve the performance of hashing retrieval. Experiments for image hashing retrieval and cross-modal hashing retrieval on public benchmarks show the superiority of our method.

References

[1] J. Wang, T. Zhang, J. Song, N. Sebe, and H. T. Shen, "A survey on learning to hash," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 769–790, Apr. 2017.

[2] Z. Cao, M. Long, J. Wang, and P. S. Yu, "HashNet: Deep learning to hash by continuation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Oct. 2017, pp. 5608–5617.

[3] R. Kang, Y. Cao, M. Long, J. Wang, and P. S. Yu, "Maximum-margin Hamming hashing," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8252–8261.

[4] E. Yang, C. Deng, T. Liu, W. Liu, and D. Tao, "Semantic structure-based unsupervised deep hashing," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 1064–1070.

[5] J. Song, T. He, L. Gao, X. Xu, A. Hanjalic, and H. T. Shen, "Binary generative adversarial networks for image retrieval," in *Proc. AAAI Conf. Artif. Intell.*, vol. 32, no. 1, 2018, pp. 394–401.

[6] Q. Hu, J. Wu, J. Cheng, L. Wu, and H. Lu, "Pseudo label based unsupervised deep discriminative hashing for image retrieval," in *Proc. 25th ACM Int. Conf. Multimedia*, Oct. 2017, pp. 1584–1590.

[7] E. Yang, T. Liu, C. Deng, W. Liu, and D. Tao, "DistillHash: Unsupervised deep hashing by distilling data pairs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2946–2955.

[8] 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.

[9] Y. Shen et al., "Auto-encoding twin-bottleneck hashing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 2815–2824.

[10] K. Lin, J. Lu, C.-S. Chen, and J. Zhou, "Learning compact binary descriptors with unsupervised deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1183–1192.

[11] S. Huang, Y. Xiong, Y. Zhang, and J. Wang, "Unsupervised triplet hashing for fast image retrieval," in *Proc. Thematic Workshops ACM Multimedia Thematic Workshops*, Oct. 2017, pp. 84–92.

[12] Y. K. Jang and N. I. Cho, "Self-supervised product quantization for deep unsupervised image retrieval," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 12065–12074.

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

[14] J. Wang, Z. Zeng, B. Chen, T. Dai, and S.-T. Xia, "Contrastive quantization with code memory for unsupervised image retrieval," in *Proc. AAAI Conf. Artif. Intell.*, vol. 36, no. 3, Jun. 2022, pp. 2468–2476.

[15] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, Nov. 2020, pp. 1597–1607.

[16] S. Su, C. Zhang, K. Han, and Y. Tian, "Greedy hash: Towards fast optimization for accurate hash coding in CNN," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 806–815.

[17] Y. Li and J. van Gemert, "Deep unsupervised image hashing by maximizing bit entropy," in *Proc. AAAI Conf. Artif. Intell.*, vol. 35, no. 3, 2021, pp. 2002–2010.

[18] Q.-Y. Jiang and W.-J. Li, "Deep cross-modal hashing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3270–3278.

[19] 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.

[20] C. Bai, C. Zeng, Q. Ma, J. Zhang, and S. Chen, "Deep adversarial discrete hashing for cross-modal retrieval," in *Proc. Int. Conf. Multimedia Retr.*, Jun. 2020, pp. 525–531.

[21] X. Nie et al., "Deep multiscale fusion hashing for cross-modal retrieval," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 1, pp. 401–410, Jan. 2021.

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

[23] E. Yang, D. Yao, T. Liu, and C. Deng, "Mutual quantization for cross-modal search with noisy labels," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 7541–7550.

[24] J. Dong et al., "Dual encoding for video retrieval by text," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 8, pp. 4065–4080, Aug. 2022.

[25] 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.

[26] 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, 2022.

[27] 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. 3120–3129.

[28] 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.

[29] 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.
[30] J. Zhang, Y. Peng, and M. Yuan, "Unsupervised generative adversarial cross-modal hashing," in *Proc. AAAI Conf. Artif. Intell.*, vol. 32, no. 1, 2018, pp. 539–546.
[31] C. Li, C. Deng, L. Wang, D. Xie, and X. Liu, "Coupled cyclegan: Unsupervised hashing network for cross-modal retrieval," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, no. 1, 2019, pp. 176–183.
[32] 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.*, vol. 35, no. 5, 2021, pp. 4626–4634.
[33] M. Li and H. Wang, "Unsupervised deep cross-modal hashing by knowledge distillation for large-scale cross-modal retrieval," in *Proc. Int. Conf. Multimedia Retr.*, Aug. 2021, pp. 183–191.
[34] X. Zhang, H. Lai, and J. Feng, "Attention-aware deep adversarial hashing for cross-modal retrieval," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 591–606.
[35] A. Radford et al., "Learning transferable visual models from natural language supervision," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 8748–8763.
[36] J. Li, R. Selvaraju, A. Gotmare, S. Joty, C. Xiong, and S. C. H. Hoi, "Align before fuse: Vision and language representation learning with momentum distillation," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 34, 2021, pp. 9694–9705.
[37] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *Int. J. Comput. Vis.*, vol. 129, pp. 1789–1819, Mar. 2021.
[38] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015, *arXiv:1503.02531*.
[39] F. Tung and G. Mori, "Similarity-preserving knowledge distillation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1365–1374.
[40] A. Miech, J.-B. Alayrac, I. Laptev, J. Sivic, and A. Zisserman, "Thinking fast and slow: Efficient text-to-visual retrieval with transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 9821–9831.
[41] K. Roth, T. Milbich, B. Ommer, J. P. Cohen, and M. Ghassemi, "Simultaneous similarity-based self-distillation for deep metric learning," in *Proc. Int. Conf. Mach. Learn.*, vol. 139, Jul. 2021, pp. 9095–9106.
[42] M. Su, G. Gu, X. Ren, H. Fu, and Y. Zhao, "Semi-supervised knowledge distillation for cross-modal hashing," *IEEE Trans. Multimedia*, early access, Nov. 22, 2021, doi: 10.1109/TMM.2021.3129623.
[43] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[44] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.
[45] M. J. Huiskes and M. S. Lew, "The MIR Flickr retrieval evaluation," in *Proc. 1st ACM Int. Conf. Multimedia Inf. Retr. (MIR)*, 2008, pp. 39–43.
[46] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. Zheng, "NUS-WIDE: A real-world web image database from national university of Singapore," in *Proc. ACM Int. Conf. Image Video Retr. (CIVR)*, 2009, pp. 1–9.
[47] T.-Y. Lin et al., "Microsoft coco: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2014, pp. 740–755.
[48] 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.
[49] X. Zhang, X. Wang, and P. Cheng, "Contrast-based unsupervised hashing learning with multi-hashcode," *IEEE Signal Process. Lett.*, vol. 29, pp. 219–223, 2022.
[50] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1–9.
[51] H. Liu, R. Wang, S. Shan, and X. Chen, "Deep supervised hashing for fast image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2064–2072.
[52] 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.
[53] X. Dong, L. Liu, L. Zhu, Z. Cheng, and H. Zhang, "Unsupervised deep K-means hashing for efficient image retrieval and clustering," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 8, pp. 3266–3277, Aug. 2021.

Xi Zhang received the bachelor's degree in communication engineering from the Nanjing University of Posts and Telecommunications, Jiangsu, China, in 2020. He is currently pursuing the Ph.D. degree with the School of Electric Engineering, Xidian University. His research interests include multimodal retrieval, multimodal machine learning, and computer vision.

Xiumei Wang received the M.Sc. and Ph.D. degrees from Xidian University in 2005 and 2010, respectively. She joined the School of Electric Engineering, Xidian University, as a Lecturer, in 2010, and is currently a Professor in signal and information processing. Her research interests include machine learning and image processing. In these areas, she has published around 30 papers, including IJCAI, IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON COMPUTERS, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B: CYBERNETICS, *Pattern Recognition*, and *Neurocomputing*.

Peitao Cheng received the M.Sc. and Ph.D. degrees in mechanical manufacturing and automation from Xidian University, China, in 2005 and 2017. respectively. Since 2005, he has been with the School of Mechano-Electronic Engineering, Xidian University. He is currently an Associate Professor in control science and control engineering. His research interests include image processing, machine learning, and artificial intelligence, and he has published around 20 papers in these areas.