IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 24 November 2022, accepted 9 February 2023, date of publication 14 February 2023, date of current version 22 February 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3245074

RESEARCH ARTICLE

Non-Co-Occurrence Enhanced Multi-Label Cross-Modal Hashing Retrieval Based on Graph Convolutional Network

MINGYONG LI, (Member, IEEE), JIABAO FAN, AND ZIYONG LIN
School of Computer Technology and Information Science, Chongqing Normal University, Chongqing 401331, China

Corresponding author: Mingyong Li (limingyong@cqnu.edu.cn)

This work was supported in part by the Chongqing Natural Science Foundation of China under Grant CSTB2022NSCQ-MSX1417, in part by the Science and Technology Research Program of Chongqing Municipal Education Commission under Grant KJZD-K202200513, and in part by the Chongqing Normal University Fund under Grant 22XLB003.

ABSTRACT Supervised cross-modal retrieval has significant advantages in retrieval efficiency and storage cost. In the field of hashing retrieval, existing supervised methods are divided into single-label and multi-label methods. For the single-label method, simply using a single label to measure the semantic relevance between instances will cause an error in supervision information. However, the existing multi-label hashing methods also have some problems. For example, only considering the co-occurrence of multiple labels among instances may not accurately reflect their similarity. At the same time, in the previous methods, the text modality processing did not reach the fine level of image modality, resulting in insufficient use of text information. To address these issues, we proposed Non-co-occurrence enhanced Multi-label cross-modal hashing retrieval based on Graph Convolutional Network (MHGCN). Firstly, we introduced a multi-label non-co-occurrence similarity measurement method, which adds multi-label non-co-occurrence information among instances in the multi-label similarity measurement to measure the differences between instances; Secondly, we used Graph Convolutional Networks (GCNS) to process the information on text modality; Thirdly, we introduced the memory mechanism to restrict the difference of hash code learning. Many experiments show that the proposed method has excellent performance. In three widely used datasets (NUS-WIDE, MIRFlickr-25k, IAPR TC-12), MAP performance in image-text and text-image tasks was significanlty improved by about 8%, 9%, and 7%, respectlively.

INDEX TERMS Non-co-occurrence enhanced hashing retrieval, graph convolutional network, multi-label method.

I. INTRODUCTION

Since entering the network era, especially the era of big data, various fields have intersected with the Internet. So multi-modal data (e.g., videos, texts, images, audios, etc.) has shown explosive growth. Cross-modal retrieval [13], [14], [15], [16] aims to start from one modality of data to find information about other relevant modalities (e.g., retrieving videos by querying texts). Because the multi-modal data of an instance describes the instance from different dimensions, there is a semantic gap. Therefore, filling in the semantic gap and getting the same semantic description is a great challenge. To this end, scholars have developed hashing retrieval technology [8], [9], [14], [47], hoping to obtain a close hash representation by mapping different modalities of instances to Hamming space, one of the most effective and popular methods.

Hash codes are widely used in various fields of computers. Mapping original data to Hamming space form it through the hash function, which is not only fast but also has low computational cost and storage consumption. Early cross-modal hashing methods [7], [10], [11], [12], [17], [19], [20], [21], [22], [23], [24] are based on hand-crafted features, with simple architecture that cannot extract deep semantic features well. Therefore, the accuracy of retrieval results cannot be further improved. The outstanding performance of neural

The associate editor coordinating the review of this manuscript and approving it for publication was Yassine Maleh.

networks stems from their ability to extract high-level features from original sensory data and easily capture the effective representation of instances. So far, various methods of applying deep neural networks to cross-modal hashing retrieval have been proposed. In the field of cross-modal hashing retrieval, much research focuses on supervised and unsupervised retrieval. The difference between the two research directions is whether to use pre-annotated labels. In the unsupervised methods, the instance features extracted by the network are used to build the affinity matrix as the guidance for network training [4], [5], [6], [7]. In the supervised methods, we directly use the label information as the strong supervision in the training process [29], [30], [31], [32], [33], [34], [35], [36], [37], [38].

Due to the strong representation ability of graphs, graph-based hashing has been widely studied by scholars. Traditionally, affinity graphs are used as a guide in the learning process. However, in the process of model training, we need to use the global similarity measurement, so the time cost is very large. Because of this, much research has been done on graphs recently, and researchers hope to add it to the feature learning process to extract more semantic information. such as Graph Convolutional Hashing (GCH) [47] and Aggregation-based Graph Convolutional Hashing for Unsupervised Cross-modal Retrieval (AGCH) [1]. Specifically, GCH adds a Graph Convolutional Network (GCN) to the learning framework and uses it to explore the inherent similarity structure between data points, which will help to generate differentiated hash codes. In AGCH, the intrinsic information embedded in each modal is effectively combined through graph convolution to aggregate the complementary semantic information in different modalities.

In real life, everything is multifaceted. It is only possible to effectively distinguish similarities using more than one label to describe instances, which may lead to suboptimal retrieval results. In fact, most instances share multiple labels, and we can use multiple shared labels between paired instances as supervised information, which can more accurately describe the semantic similarity between instances. According to the number of co-occurrence labels between paired instances, we can measure the similarity between instance pairs: the greater the number of co-occurrence labels between instances, the more similar; otherwise, the less similar (Figure 1).

FIGURE 1. From the perspective of a single label, the similarity between instances a, b and a, c is the same, which is unreasonable.

However, even if the number of co-occurrence labels between two pairs of instances is equal, their similarity should be different. Inspired by MDMCH [2], we add multi-label non-co-occurrence information between instances to multi-label similarity measurement. If instances a and b have

FIGURE 2. From the perspective of multiple labels, even if a, b, and a, c share the same number of labels, the similarity should also be different.

the same number of shared labels as instances a and c, but the number of non-co-occurrence labels between instances a and c is less than the instances a and b, then we have reason to think that the similarity of the latter is greater than the former (Figure 2).

To this end, we propose Non-co-occurrence enhanced Multi-label cross-modal hashing retrieval based on Graph Convolutional Network(MHGCN) for cross-modal multi-label hashing retrieval. We use non-co-occurrence information between instances to enhance our similarity matrix. In order to make text modality processing reach the fine level of image modality, we use Graph Convolutional Network [78] to mine semantic features and retain the semantic information between instances in the original space as much as possible.

The contributions of our MHGCN are the following:

1: We introduced a multi-label non-co-occurrence similarity measurement method, which adds multi-label non-co-occurrence information among instances in the multi-label similarity measurement to enhance the similarity matrix. Therefore, we can judge more accurately the similarity between instances.

2: Because graph networks have strong representation ability, we introduce Graph Convolutional Networks [78] into our proposed model (MHGCN). Therefore, our model can fully mine the semantic information in the text, which helps our model learn the hash codes.

3: In addition, we introduced a memory bank [70] to retain the hash code generated in our learning process effectively. Therefore, we can constrain the hash representation in the whole training process, not only on the mini-batch.

4: Our model performs better on the three benchmark datasets in most cases than the most recent excellent work. This indicates that our model can better extract the semantic features in the instance and generate hash codes with richer semantic information, which will be conducive to downstream tasks.

The remaining chapters are summarized as follows. We review the related works in section II. In Section III, we introduce our method (MHGCN) and give the symbol definition. Section IV gives a detailed description of the optimization algorithm of our framework. We describe the experimental analysis and results in Section V. We give our conclusions in Section VI.

II. RELATED WORK

The rapid development of the Internet connects the whole world, and many multimodal data are released daily. As a hot research field, cross-modal hashing retrieval has been widely studied by scholars, and a large number of efficient methods have been proposed. According to whether the pre-annotated

labels are used, we can divide the cross-modal hashing method into supervised method and unsupervised method. Unsupervised methods usually use affinity matrices to constrain the generation of consistent hash codes. Some excellent unsupervised cross-modal hashing methods include Deep Joint-Semantics Reconstructing Hashing for Large-Scale Unsupervised Cross-Modal Retrieval (DJSRH) [4], Semantic Topic Multimodal Hashing for Cross-Media Retrieval (STMH) [7], Unsupervised Contrastive Cross-modal Hashing (UCCH) [5], Unsupervised Deep Cross-modal Hashing with Virtual Label Regression (UDCH-VLR) [6] and so on.

Unlike unsupervised methods, supervised methods usually use pre-annotated labels to construct the similarity matrix, which serves as the guidance of the training process. A number of excellent methods include but are not limited to Cross-modality Metric Learning using Similarity-Sensitive Hashing (CMSSH) [19], which by means of embedding incommensurable data into a common metric space. Semantics-Preserving Hashing for Cross-View Retrieval (SePH) [20], which standardizes all Hamming distances by transforming each into a probability that depends on all others. Thus, combining the correlation between hamming distances. Seamless integration of semantic labels into the hash learning process for large-scale data modeling (SCM) [21]. Generalized semantic preserving hashing for cross-modal retrieval (GSPH) [22] using kernel logistic regression. Although the above methods are very effective, they are all based on hand-crafted features. They cannot extract deeper semantic features in the instance, which will cause inaccuracy in the training process and lead to suboptimal experimental results.

A. SINGLE-LABEL METHOD

With the improvement of hardware performance, deep learning has spread to many other fields. Deep Cross-Modal Hashing (DCMH) [29] Introduces Deep Neural Network into Cross-modal hashing. In DCMH [29], image network and text network are used to extract features for cross-modal data, respectively, and then used negative log likelihood function to optimize loss. Adversary Guided Asymmetric Hashing for Cross-Modal Retrieval (AGAH) [30] introduces the thought of Adversary Guided into end-to-end hashing learning and obtains consistent hash codes through the adversarial between text and image output. Pair-wise relationship guided deep hashing for cross-modal retrieval (PRDH) [31] integrates different types of pairwise constraints to encourage the similarities of the hash codes from an intra-modal view and an inter-modal view, respectively. Cross-modal Hamming hashing (CMHH) [32] achieves efficient retrieval by punishing the instance pairs whose hamming distance is greater than the threshold. Correlation hashing network for efficient cross-modal retrieval (CHN) [33] optimize the maximum margin loss on similar pairs.

B. MULTI-LABEL METHOD

Due to the powerful learning ability of the deep neural network, the above methods have excellent performance. However, almost all of the above methods are based on a single label to calculate the similarity between instances, which will cause much delicate semantic information to be ignored. Using multiple labels method can enrich the semantic features extracted from network. Improved Deep Hashing with Soft Pairwise Similarity for Multi-label Image Retrieval (IDHN) [34] uses soft and hard similarity to distinguish semantic similarity between instances. Deep Multi-Level Semantic Hashing for Cross-Modal Retrieval (DMSH) [35] uses multi-level semantic similarity to construct similarity matrix. Self-supervised adversarial hashing networks for cross-modal retrieval (SSAH) [36] uses multiple labels as supervisory information. Multi-label semantics preserving based deep cross-modal hashing (MLSPH) [3] define a new similarity calculation method to utilize multiple labels information. However, even between instance pairs with the same similarity, they should be different, not identical. Multiple deep neural networks with multiple labels for cross-modal hashing retrieval (MDMCH) [2] to measure the difference between instances by calculating semantic factors.

C. GRAPH CONVOLUTIONAL NETWORK

Graph neural network regards data as a node and uses an adjacency matrix to measure the relationship between data. Since GNN [13] was first proposed, it has attracted extensive research interest in classification, association prediction and other fields. In GNN, each iteration uses the features of neighbours to update itself. Finally, the information of neighbours can be aggregated. Thus, the relationship between data can be captured. Nevertheless, it is easy to cause the features of the node itself in the iteration process to be ignored. Therefore, GCN [78] is proposed to solve the problem. GCN strengthens its own features while weakening the information of neighbours in the aggregation process so that the features of data can be extracted well. Some other representative work includes GraphSAGE [14], Graph Generative Networks (DGMG) [16] and MolGAN [79], Graph Attention Networks (GATs) [15]. Specifically, GATs is a space-based graph convolution networks, which use attention in the aggregation process and can amplify the impact of the most important part of data. GraphSAGE leverages node feature information (e.g., text attributes) to efficiently generate node embeddings for previously unseen data. DGMG will generate a node in the graph during each iteration and will make decision and judgment after each node is added. If the judgment is true, it will extract nodes from existing nodes and add edges. When finished, DGMG will update the representation of the graph. MolGAN [79] improves the authenticity of the generated object through the competition between the discriminators and the generators. In this paper, in order to make the text modality processing reach the fine level of image modality, we select GCN to improve the ability of processing text information.

Inspired by MDMCH [2], we use multi-label non-co-occurrence information to enhance the similarity matrix, which can make our similarity matrix more delicate.

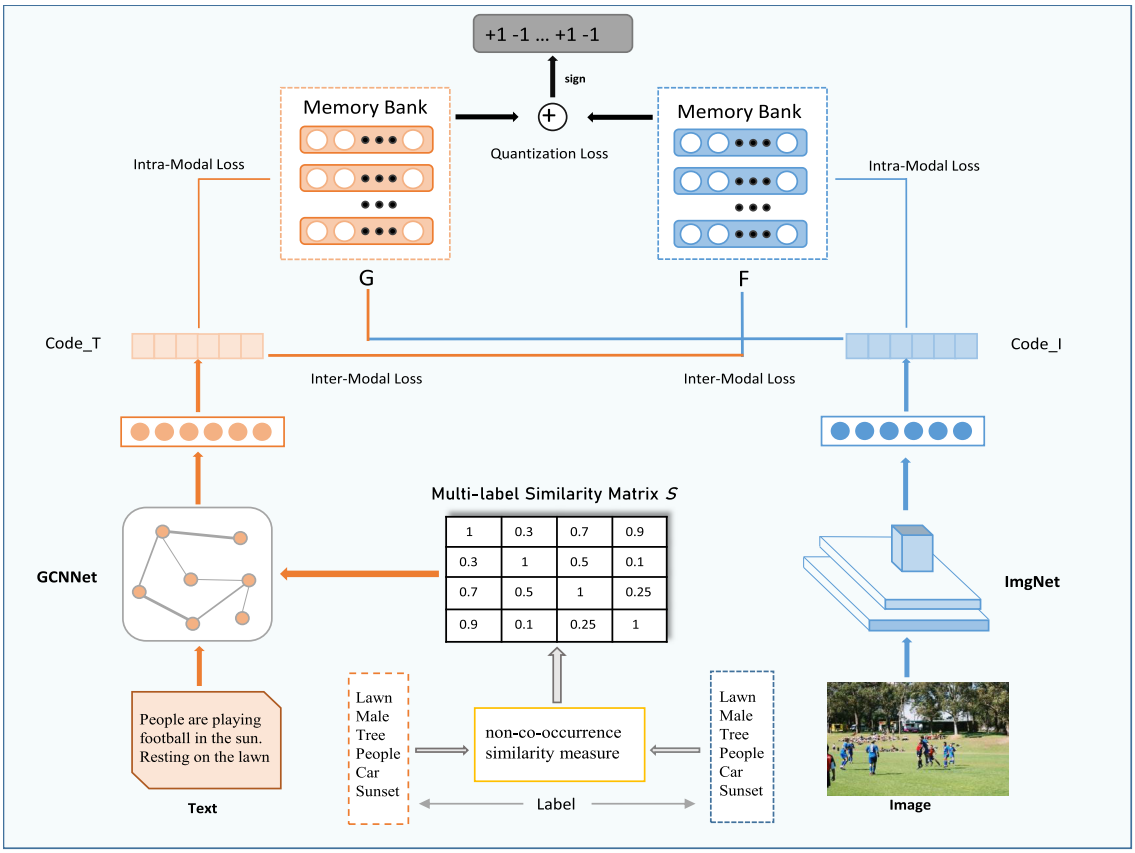


FIGURE 3. The framework of MHGCN mainly includes: (1) Using non-co-occurrence similarity measures to build a similarity matrix. (2) Learn the features extracted from GCN to generate hash codes.

Moreover, we introduce Graph Convolutional Network to improve our representation of text features. To ensure the generation of hash codes with correct semantic information, we reduce the error between semantic feature similarity and label similarity by using mean square error loss.

III. METHOD

A. PROBLEM DEFINITION AND NOTATION

We give the definition of symbols used throughout the paper. We use the uppercase letter, e.g., $\mathbf{V}$ to represent the matrix, and the lowercase letter, e.g., $\mathbf{v}$ to represent the vector. Row i and column j of matrix $\mathbf{V}$ are respectively expressed as $\mathbf{V}_{i*}$ and $\mathbf{V}_{*j}$. We use $\mathbf{O} = \{o_1, o_2, \ldots, o_N\}$ to represent multi-modal dataset, where N represents the number of instances. Each $o_i = (o_i^v, o_i^t)$ is a instance in the multi-modal dataset. We use $\mathbf{L} = \{l_i\}_{i=1}^N \in R^{N\times c}$ denotes the label matrix, where c represents the number of label categories of the instance. $l_{ik} = 1$ represents the instance o_i belongs to semantic category k, otherwise $l_{ik} = 0$. $sign(\cdot)$ is a sign function defined as:

$$\text{sign}(x) = \begin{cases} 1 & x \geq 0 \\ -1 & x < 0 \end{cases} \tag{1}$$

B. SIMILARITY CONSTRUCTION

Traditionally, $s_{ij} = 1$ represents that instances of different modalities share at least one identical semantic class, otherwise $s_{ij} = 0$. Obviously, this method is not helpful to distinguish the degree of similarity between instances. In IDHN [34] and ISDH [37], the pairwise similarity is divided into soft similarity and hard similarity. The similarity between instances can be obtained by calculating the distance between their labels with the cosine function. Similarly, the hamming distance between instances can be calculated with the cosine function.

$$s_{ij} = \frac{\langle l_i, l_j \rangle}{\|l_i\| \, \|l_j\|} \tag{2}$$

DMSH [35] use 'multi-level semantic similarity' to construct the similarity matrix to preserve the semantic information.

$$S^{(l)}(m,n) = \frac{\sum_i^{|t_m|} \sum_j^{|t_n|} s(t_m(i), t_n(j))}{|t_m| \times |t_n|} \tag{3}$$

In MLSPH [3], The similarity matrix is constructed by using method similar to Intersection of Union.

$$S_{ij}^{vt} = \frac{2\left\|l_i^v \cap l_j^t\right\|}{\left\|l_i^v\right\| + \left\|l_j^t\right\| - \left\|l_i^v \cap l_j^t\right\|} - 1 \tag{4}$$

Through the above method, we can see that the similarity between instances is no longer a simple 0 and 1, but a number

between 0 and 1, which helps us to distinguish the degree of similarity between different instances.

For us, inspired by MDMCH [2], we use semantic factors as multi-label non-co-occurrence information to enhance the similarity matrix, which can make our similarity matrix more delicate.

$$c_{ij}=\frac{2\left\|l_i-l_j\right\|_F^1}{|l_i|+\left|l_j\right|-\left\|l_i-l_j\right\|_F^1} \tag{5}$$

where l_i is the label vector of instance o_i, $|l_i|$ represents the length of label vector l_i. $\left\|l_i-l_j\right\|_F^1$ is the number of non-co-occurrence class labels between instance o_i and o_j.

Through the above way, we can use non-co-occurrence information between instances. e.g., we have two label vectors of instances o_1 and o_2, which is $l_1 = \{0, 1, 1, 0, 1\}$ and $l_2 = \{1, 0, 0, 0, 1\}$. Both instance o_1 and o_2 do not belong to the fourth semantic class. If defined as before, this non-co-occurrence information will not be regarded as the similarity of instance one and two. However, we believe that if both instances do not belong to the same semantic class, this can also be regarded as a kind of similarity information.

Therefore, we can use non-co-occurrence information and traditional similarity to construct our similarity matrix S^{new} as follows:

$$s_{ij}^{new}=\begin{cases}-1, & s_{ij}=0\\ 2s_{ij}-E_{ij}-c_{ij}, & s_{ij}=1\end{cases} \tag{6}$$

To reduce complexity, in the following, we use S instead of S^{new}.

C. FRAMEWORK

Fig. 1 shows the framework of our model (MHGCN), which mainly includes two networks for feature learning and multi-label non-co-occurrence information enhancement methods for building a similarity matrix. The image network is responsible for extracting image information to generate the hash representation of images. Similarly, a text network is responsible for extracting text information for generating the hash representation of texts. We consider the non-co-occurrence label information between instances to optimize the similarity matrix and fully use the label information to guide network learning. At the same time, the similarity matrix is input into the text network as the adjacency matrix.

For the text network, inspired by the AGCH [1], we use Graph Convolution Network (GCN) to mine text semantic features, and input the similarity matrix formed by labels into GCN as the adjacency matrix. Because of the powerful representation ability of Graph Convolutional Network, we can get more robust text hash code. For the image network, we use ResNet34 [41], because its internal residual blocks use shortcut connections, which alleviates the problem of gradient disappearance caused by increasing depth in the deep neural network, and it has excellent performance in image classification.

We use cur_f to represent the image features of instance $o^i=\left\{o^v, o^t\right\}$ extracted from the image network $f(o^v;\theta_v)$. $F \in R^{k\times N}$ denotes the hash representation of the image stored in the memory bank. Similarly, cur_g represents the text features extracted from the GCN network $g(o^t;\theta_t)$. $G \in R^{k\times N}$ denotes the hash representation of the text stored in the memory bank. N and k represent the number of instances and the length of the final hash code, respectively. The Graph Convolutional Network (GCN) propagate process is written as follows:

$$H^{(l+1)}=\sigma(l)\left(\widetilde{D}^{-\frac{1}{2}}\widetilde{A}\widetilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right) \tag{7}$$

where $\tilde{D}_{ii}=\sum_j\tilde{A}_{ij}$ and $W^{(l)}$ is a layer-specific weight matrix. $\sigma(l)$ denotes an activation function. $H^l \in R^{d\times m}$ is the matrix of activations in the Lth layer. So, the binary code B_I and B_T generation are expressed as follows:

$$\begin{aligned} B\_I &= sign(f(o^v;\theta_v)),\\ B\_T &= sign(g(o^t;\theta_t)),\end{aligned} \tag{8}$$

where θ_v and θ_t represent parameters in the networks. In order to avoid the back-propagate gradient problem caused by $sign(\cdot)$ function in the training process, we use $tanh(\cdot)$ to replace it.

From Eq. (7) and (8), we can see that in the graph convolutional network, an instance is regarded as a node, and we can update the features of the node through its neighbours. Through the weighted summation cascade of the neighbours of the node, the features of its neighbours are allocated to the node, which indicates that the features of adjacent nodes in the feature space will be closer. Therefore, the generated hash code can well reflect the relationship of instances in the feature space.

The construction of adjacency matrix is crucial for node learning in the graph convolutional, which are elaborated before.

D. HASH LEARNING

The quality of hash codes determines the accuracy of retrieval to a certain extent. Therefore, it is a challenge to obtain hash codes that not only contain rich semantic information but also can distinguish semantically similar instances. We constrain the network learning process through three aspects of losses. In order to bridge the semantic gap, we use an inter-modal loss to reduce the semantic difference between different modalities of information of the same instance. For similar instances, we use intra-modal loss and quantitative loss to generate hash codes with discrimination. We calculate the cosine similarity between hash representations to represent the semantic similarity between instances learned by the model. By continuously reducing the loss of mean square error, we can obtain hash codes that retain more and more semantic information. Cosine similarity is defined as follows:

$$\cos(f_i, g_j)=\frac{\langle f_i, g_j\rangle}{\|f_i\|_{L_2}\left\|g_j\right\|_{L_2}} \tag{9}$$

The range of cosine similarity $\cos(f_i, g_j)$ is $[-1, 1]$, $\|\cdot\|_{L2}$ is the L_2 norm. $\langle.\rangle$ represents the inner product. Obviously, if the

similarity of two instances is lower, the cosine similarity of their hash representation will be lower.

The intra-modal pair-wise loss consists of two parts, image-to-image pairwise loss and text-to-text pairwise loss, which are defined as follows:

$$L^{v}{}_{MSE} = \frac{1}{N^2}\sum_{i=1,j=1}^{N}(\cos(f_i, f_j) - S_{ij})^2 \tag{10}$$

$$L^{t}{}_{MSE} = \frac{1}{N^2}\sum_{i=1,j=1}^{N}(\cos(g_i, g_j) - S_{ij})^2 \tag{11}$$

where N represents the training instances. The S in Eq.(10) and (11), which represents the similarity between instances. For example, S_{ij} represents the similarity between instance i and instance j. Through the intra-modal loss, we can preserve the similarity of the same modality of different instances.

Meanwhile, the inter-modal loss define as follows:

$$L^{vt}{}_{MSE} = \frac{1}{N^2}\sum_{i=1,j=1}^{N}(\cos(f_i, g_j) - S_{ij})^2 \tag{12}$$

We introduce quantization loss to smooth the difference between hash codes and hash representations, while reducing the distance between them.

$$L_{QL\_V} = \frac{\sum_{i=1}^{N}\sum_{j=1}^{K}(b_{ij} - f_{ij})}{NK} \tag{13}$$

$$L_{QL\_T} = \frac{\sum_{i=1}^{N}\sum_{j=1}^{K}(b_{ij} - g_{ij})}{NK} \tag{14}$$

where f_i represents the hash representation extracted from the image network, and f_{ij} represents the jth value. Similarly, g_i represents the hash representation extracted from the text network, and g_{ij} represents the jth value. We use b_{ij} to represent the jth bit of the unified hash code b_i. From Eq.(10),(11),(12),(13) and (14), we can obtain the final objective function:

$$\begin{aligned}\min_{B,\theta_v,\theta_t} L = {} & \alpha L^{v}{}_{MSE} + \alpha L^{t}{}_{MSE} + \beta L^{vt}{}_{MSE} \\ & + \gamma L_{QL\_V} + \gamma L_{QL\_T}\end{aligned} \tag{15}$$

α, β and γ are the hyper-parameters used for loss calculation.

In addition, we introduce a memory bank [70] to memorize the hash representations in each training batch. In each training batch, we use the up-to-date hash representations and label constraints between instances for loss calculation to retain the semantic information in instances and the semantic relevance between instances.

IV. OPTIMIZATION

For all parameters in the network (θ_v, θ_t, B), we adopt the alternating strategy to optimize. Specifically, we adopt the strategy of alternating updating parameters, updating one parameter and fixing other parameters in each iteration. The optimization algorithm of our model(MHGCN) is summarized in Algorithm 1.

Algorithm 1 MHGCN

Input: N training instances of dataset O, where $o_i = \{o^v, o^t, o^l\}, i = \{1, 2, 3, \ldots, n\}$
Similarity matrix S.

Output: Parameters θ_v, θ_t of the deep neural networks, and binary code matrix B.

Initialize the deep neural network Parameters θ_v, θ_t and each modality hash representations stored in memory bank: F, G. Set mini-batch size $n_v = n_t = 128$, and iteration number $num_x = \lceil N/n_v \rceil$, $num_y = \lceil N/n_t \rceil$;
Calculate multi-label semantic similarity matrix S using Eq. (5), (6);
repeat
  for iter = 1 to num_x **do**
    Randomly sample n_v instances from O to construct a mini-batch;
    For each sampled instcnce o_i in the mini-batch, calculate f_i by forward propagation;
    Calculate the derivative using Eq. (16);
    Update parameters θ_v by using back propagation;
  end for
  for iter = 1 to num_y **do**
    Randomly sample n_t instances from O to construct a mini-batch;
    For each sampled instance o_i in the mini-batch, calculate g_i by forward propagation;
    Calculate the derivative using Formula. (17);
    Update parameters θ_t by using back propagation;
  end for
  Update B using Eq. (18);
until convergence.

A. UPDATING θ_v

By fixing θ_t and B, we can learn the parameters θ_v of the image network through the stochastic gradient descent (SGD) with back-propagation (BP). Each time we randomly select a mini-batch from the training set for loss calculation. We calculate the gradient as follows and update the parameters in the network through backpropagation.

$$\begin{aligned}\frac{\partial L}{\partial f_i} = {} & \frac{2\beta}{N^2}\sum_{j=1}^{N}(cos(f_i, g_j) - S_{ij})sin(f_i, g_j) \\ & + \frac{2\alpha}{N^2}\sum_{j=1}^{N}(cos(f_i, f_j) - S_{ij})sin(f_i, f_j) \\ & - \frac{\gamma\sum_{j=1}^{K}(b_{ij} - f_{ij})}{2NK}\end{aligned} \tag{16}$$

Then, it can use the chain rule to calculate $\frac{\partial L}{\partial \theta_v}$ with $\frac{\partial L}{\partial f_*}$. Finally, parameter θ_v can be updated based on BP (Back Propagation Algorithm).

TABLE 1. Comparison table of experimental results.

Task	Name		NUS-WIDE			MIRFlickr-25k			IAPR TC-12		
			16bits	32bits	64bits	16bits	32bits	64bits	16bits	32bits	64bits
I-T	Hand-crated methods	SePH [20]	0.4797	0.4859	0.4906	0.6740	0.6813	0.6803	0.4186	0.4298	0.4315
		CMSSH [19]	0.3092	0.3099	0.3396	0.5600	0.5709	0.5836	0.3049	0.3074	0.3010
		GSPH [22]	0.4015	0.4151	0.4214	0.6068	0.6191	0.6230	0.3716	0.3921	0.4015
		SCM [21]	0.4626	0.4792	0.4886	0.6354	0.6407	0.6556	0.3887	0.3945	0.4068
	Deep Methods	PRDH [31]	0.5919	0.6059	0.6116	0.6952	0.7072	0.7108	0.4761	0.4883	0.4925
		DCMH [29]	0.5445	0.5597	0.5803	0.7316	0.7343	0.7446	0.4536	0.4727	0.4919
		CMHH [32]	0.5530	0.5698	0.5559	0.7334	0.7281	0.7444	0.4903	0.5074	0.5152
		CHN [33]	0.5754	0.5966	0.6015	0.7504	0.7495	0.7461	0.4962	0.5070	0.5241
		SSAH [36]	0.6163	0.6278	0.6140	0.7745	0.7882	0.7990	0.5348	0.5619	0.5781
		SCAHN [38]	0.6429	0.6510	0.6635	0.8168	0.8311	0.8328	0.5427	0.5764	0.5908
		MLSPH [3]	0.6405	0.6604	0.6734	0.8076	0.8235	0.8337	0.5342	0.5721	0.5994
		MDMCH [2]	0.6579	0.6583	0.6521	0.7891	0.7952	0.8112	xxxx	xxxx	xxxx
		OURS	**0.7783**	**0.7831**	**0.7878**	**0.8911**	**0.9018**	**0.9019**	**0.6047**	**0.6230**	**0.6335**
T-I	Hand-crated methods	SePH [20]	0.6072	0.6280	0.6291	0.7139	0.7258	0.7294	0.4667	0.4857	0.4936
		CMSSH [19]	0.3167	0.3171	0.3179	0.5726	0.5776	0.5753	0.3189	0.3282	0.3229
		GSPH [22]	0.4995	0.5233	0.5351	0.6282	0.6458	0.6503	0.4177	0.4452	0.4641
		SCM [21]	0.4261	0.4372	0.4478	0.6340	0.6458	0.6541	0.3824	0.3897	0.4002
	Deep Methods	PRDH [31]	0.6155	0.6286	0.6349	0.7626	0.7718	0.7755	0.5112	0.5283	0.5403
		DCMH [29]	0.5793	0.5922	0.6014	0.7607	0.7737	0.7805	0.4851	0.4976	0.5171
		CMHH [32]	0.5739	0.5786	0.5639	0.7320	0.7183	0.7279	0.4790	0.4951	0.4963
		CHN [33]	0.5816	0.5967	0.5992	0.7776	0.7775	0.7798	0.4994	0.5370	0.5397
		SSAH [36]	0.6204	0.6251	0.6215	0.7860	0.7974	0.7910	0.5265	0.5594	0.5726
		SCAHN [38]	0.6501	0.6575	0.6685	0.8034	0.8105	0.8193	0.5297	0.5558	0.5693
		MLSPH [3]	0.6433	0.6633	0.6724	0.7852	0.8041	0.8146	0.5252	0.5624	0.5938
		MDMCH [2]	0.6892	0.6955	0.7035	0.8132	0.8183	0.8211	xxxx	xxxx	xxxx
		OURS	**0.7709**	**0.7803**	**0.7819**	**0.9077**	**0.9098**	**0.9162**	**0.6593**	**0.6874**	**0.6884**

B. UPDATING θ_t

Similar to update θ_v, by fixing θ_v and B, we can learn the parameters θ_t of the text network through the stochastic gradient descent (SGD) with back-propagation (BP).

$$\begin{aligned}\frac{\partial L}{\partial g_i} = &\frac{2\beta}{N^2}\sum_{j=1}^{N}(cos(f_j, g_i) - S_{ij})sin(f_j, g_i)\\ &+\frac{2\alpha}{N^2}\sum_{j=1}^{N}(cos(g_i, g_j) - S_{ij})sin(g_i, g_j)\\ &-\frac{\gamma\sum_{j=1}^{K}(b_{ij} - g_{ij})}{2NK}\end{aligned} \tag{17}$$

Then, it can use the chain rule to calculate $\frac{\partial L}{\partial \theta_t}$ with $\frac{\partial L}{\partial g_*}$. Finally, parameter θ_t can be updated based on BP (Back Propagation Algorithm).

C. UPDATING B

When θ_v and θ_t are fixed, we can update B as follows:

$$B = sign(F + G) \tag{18}$$

D. OUT-OF-SAMPLE EXTENSION

For instances that are not in the training set, we can obtain the hash code of the instance through the well-trained model easily. Specifically, for the imaging modality of the query instance, we can obtain the hash code through forward propagation as follows:

$$b_q^v = sign(f(o^v; \theta_v)) \tag{19}$$

Similarly, we can also obtain the hash code for the text modality of query instance as follows:

$$b_q^t = sign(g(o^t; \theta_t)) \tag{20}$$

Then, we can obtain the corresponding retrieval instance by calculating the distance of the hash codes.

V. EXPERIMENTS

In this section, we will detail the model evaluation indicators and the datasets used in the experiments. We compare our method (MHGCN) with the excellent methods and discuss its performance of it.

A. DATASETS

The IAPRTC-12 dataset is a commonly used dataset in the field of cross-modal retrieval, which contains 20,000 instances. Since one of the instances is not labeled, we use the remaining 19,999 instances as our experimental data. For each instance, the image is resized to 224 * 224 * 3 and the text is converted to a 1,251-dimensional bag-of-words vector. For all instances in the dataset, we use 10,000 of them to build training set, 2,000 of them to build query set, and the rest as database. All selected datasets are mutually exclusive.

The NUS-WIDE dataset is a commonly used cross-modal retrieval dataset, which contains 269,468 instances and 81 semantic class labels. After removing some instances with incorrect labels, we selected 190,421 of them. Their labels are frequent 21 semantic categories. The text of all instances is converted to 1,000-dimensional bag-of-words vector. For all instances in the dataset, we use 10,500 of them to build

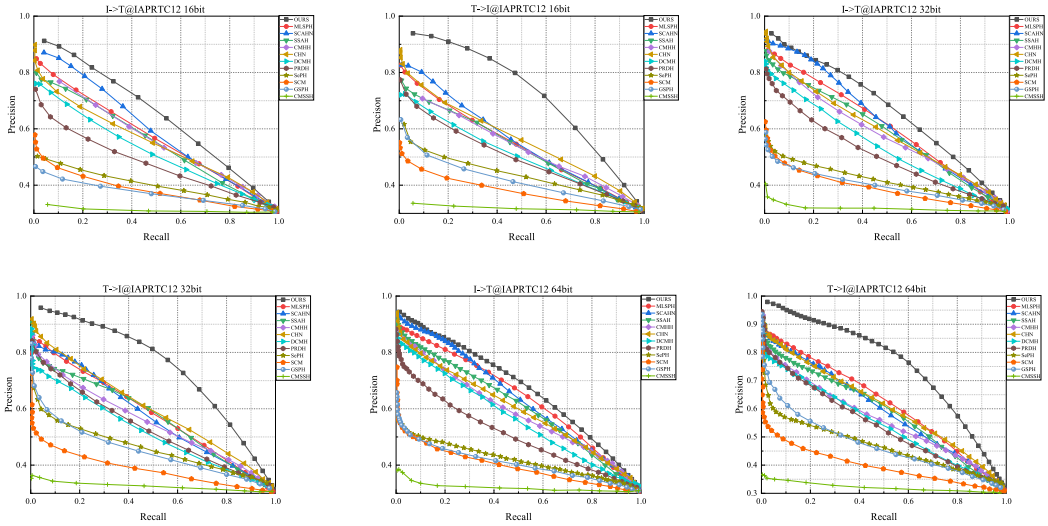


FIGURE 4. Precision–recall curves on IAPRTC12.

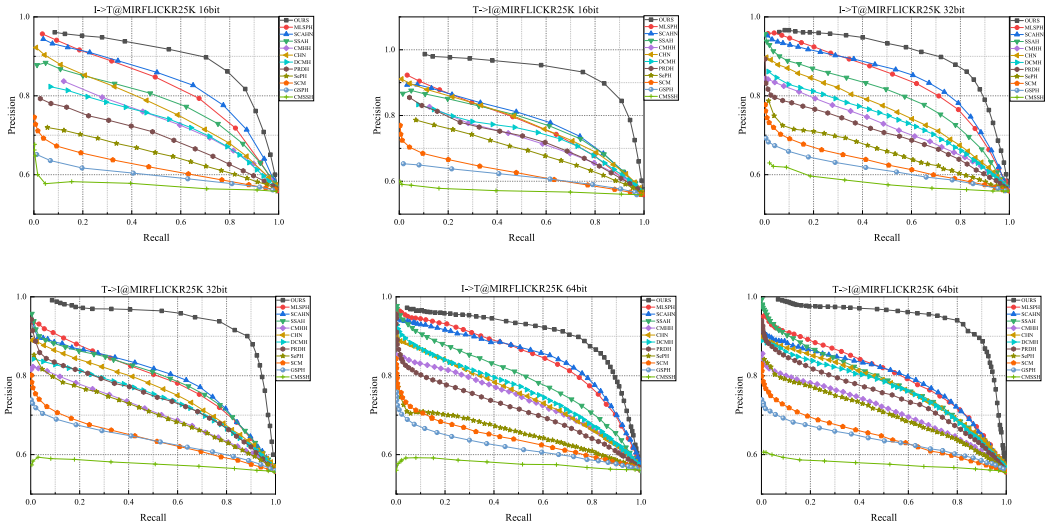


FIGURE 5. Precision–recall curves on MIRFLICKR25K.

training set, 2,100 of them to build query set, and the rest as database. All selected datasets are mutually exclusive.

We selected 20,015 instances in MIRFLICKR-25K dataset containing 25,000 instances as our experimental dataset. The text of all instances is converted to 1,386-dimensional bag-of-words vector. For all instances in the dataset, we use 10,000 of them to build training set, 2,000 of them to build query set, and the rest as database. All selected datasets are mutually exclusive.

B. EVALUATION PROTOCOL

In cross-modal retrieval, we aim to return the information of another modality through the information of one modality of an instance. For example, given a sentence describing an instance, the described picture is returned. In order to measure the retrieval accuracy of the model, we use the mean average precision (MAP) as the reference parameters. The retrieval accuracy of the model is proportional to the MAP value. Similarly, the top-N curve reflects the retrieval accuracy of the model in different recall quantities. The area enclosed by precision-recall curve (PR curve) determines the performance of the model. The average precision (AP) is defined as follows:

$$AP(i) = \frac{1}{N}\sum_{i=1}^{n_r} p_i t(i) \tag{21}$$

where i represents the instance used for query. n_r indicates the number of instances waiting to be retrieved in the database. N represents the returned query results.

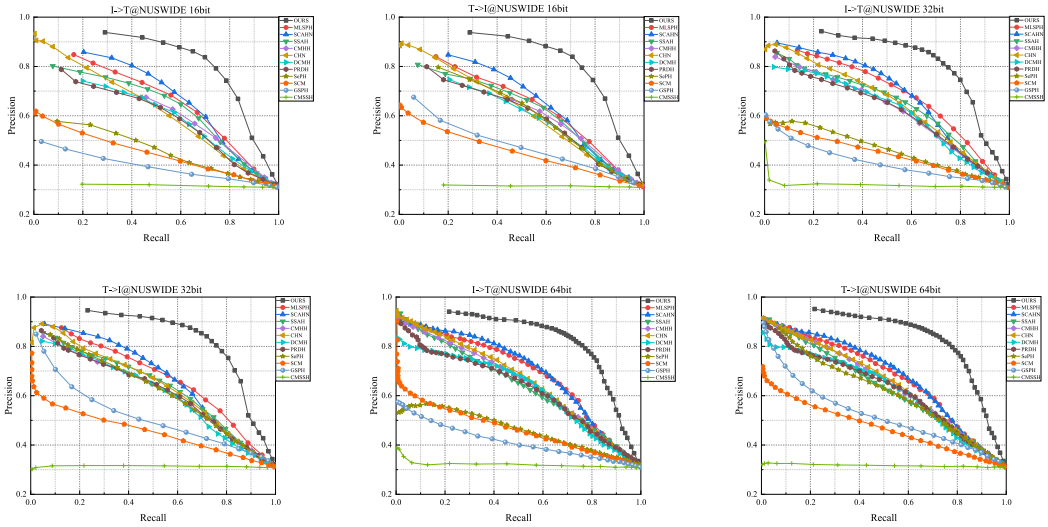


FIGURE 6. Precision–recall curves on NUSWIDE.

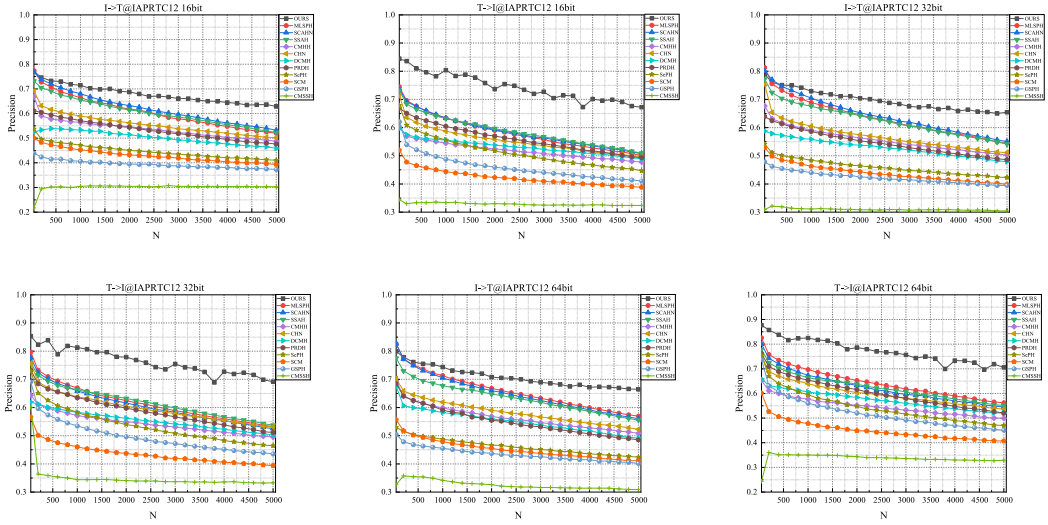


FIGURE 7. topN-precision curves on IAPRTC12.

p_i represents the probability that the query instance is similar to the top i retrieval results. $t(i) = 1$ or 0 to indicate similarity or dissimilarity. From the average precision (AP), we can get the definition of MAP as follows:

$$MAP = \frac{1}{n_{query}} \sum_{i=1}^{n_{query}} AP(q_i) \tag{22}$$

The MAP reflects the precision of model retrieval. The retrieval accuracy of the model in different recall quantities constitutes the top-N precision curve. The precision recall curve is used to measure the accuracy of the hash lookup protocol.

C. MODEL EVALUATION

We have compared our methods with the state-of-the-art existing methods, including CMSSH [19], SePH [20], SCM [21], GSPH [22], DCMH [29], PRDH [31], CMHH [32], CHN [33], SSAH [36], SCAHN [38], MLSPH [3], MDMCH [2] (Because we cannot obtain the complete experimental code of MDMCH, we cannot conduct a complete experiment on IAPRTC-12 dataset). From left to right, there are four hand-crafted methods and eight methods based on deep features. The results of all the above methods are from the recurrence experiments or the original paper settings.

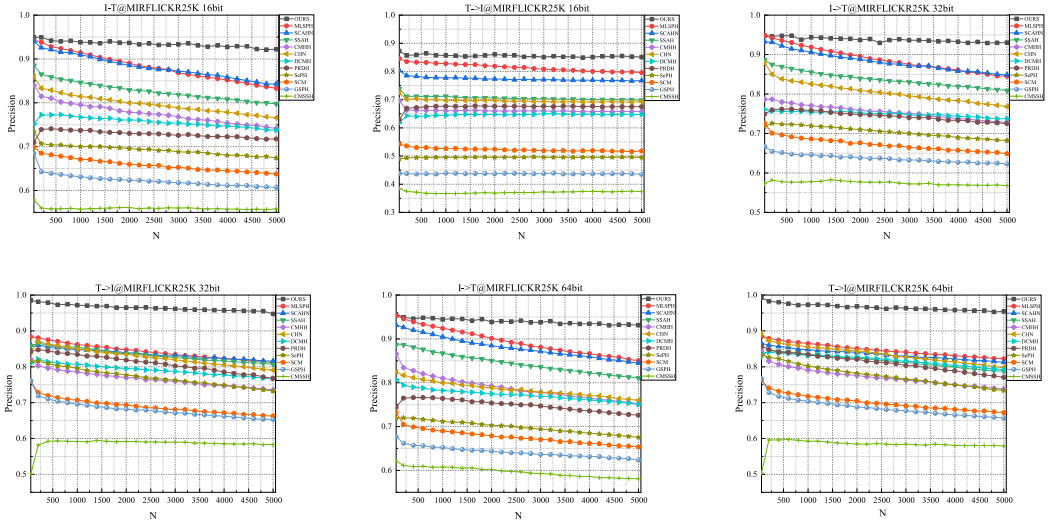


FIGURE 8. topN-precision curves on MIRFLICKR25K.

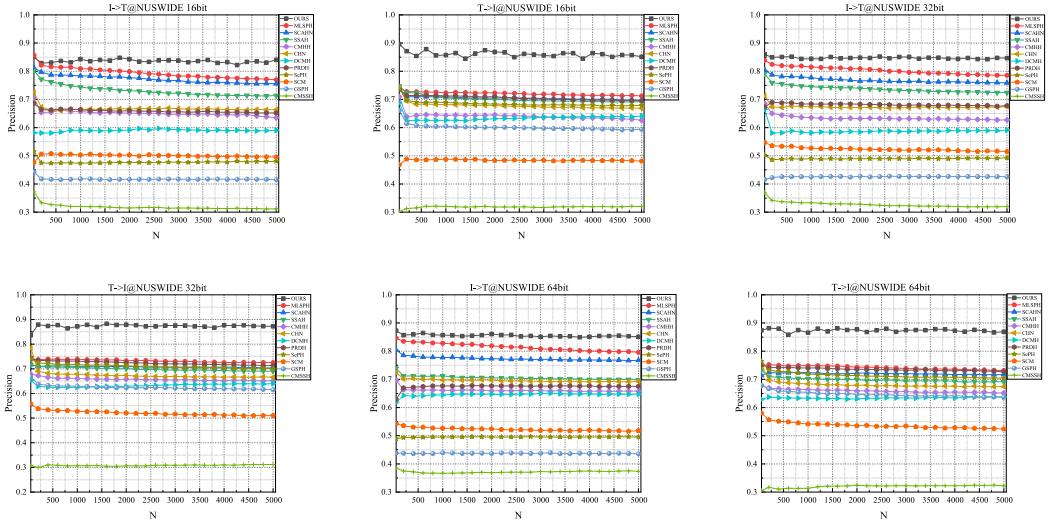


FIGURE 9. topN-precision curves on NUSWIDE.

The performance of our model with different code lengths on MIRFLICKR-25K, NUS-WIDE, and IAPRTC-12 datasets is shown in Table 1.

Because our model adopts a more accurate similarity matrix construction method, we can get more distinctive semantic features in the training process. We can effectively distinguish instances with an equal number of labels, and our model can better learn deeper semantic features in the learning process to generate more discriminating hash codes. Graph neural network regards data as a node and uses an adjacency matrix to measure the relationship between data. Because of its powerful presentation capability, our method achieves a higher MAP value than the baseline methods in most cases.

MLSPH, SePH and MDMCH consider the semantic relevance of multiple labels when generating hash codes, so MLSPH, SePH and MDMCH are outstanding in the baseline method. This shows that compared with the single-label method, the multi-label method is easier to guide the network to mine the semantic information hidden in the instance.

We change the Hamming radius from 0 to k to draw the precision-recall curves. We can see from Figs 4, 5 and 6 that our MHGCN method is superior to other methods in different code lengths (16, 32, 64) of the three datasets.

From 1 to 5000, we take a value every 200 as our recall number and draw the results of different hash lengths (16,32,64) on three datasets in Figures 7, 8 and 9. From the

figures, we can see that our topN accuracy curves are better than the other baseline methods.

D. IMPLEMENTATION DETAILS AND ABLATION STUDY

We set the hyper-parameters α, β and γ in the experiment to 0.9, 1.2 and 0.1 respectively, and the mini-batch to 128. We set the learning rate from initial $10^{-1.5}$ to 10^{-6} in 200 iterations. All experiments are conducted under the above settings. Our experimental platform is the open-source environment Pytorch and a NVIDIA 3080Ti GPU.

To compare the effects of different similarity matrix construction methods on the experiment, we made statistics on the experimental results of 64-bit code length on MIRFlickr-25K dataset. We can see the impact of the three methods on the model performance in Table 2.

TABLE 2. Comparison table of similarity methods.

Construction method of similarity matrix	I-T	T-I
Hard-soft similarity	0.8875	0.9023
multi-label semantic relevance(MLSPH)	0.8954	0.9108
non-co-occurrence similarity measure	0.9041	0.9207

To verify the improvement of the Graph Convolutional Network on our model performance, We use two commonly used model architectures to replace it. We make statistics on the experiments of 64-bit code length on the MIRFlickr-25K dataset. The results are given in Table 3. We can see that the Graph Convolutional Network can fully exploit the features in the text information due to its strong representation ability, which brings huge performance improvement to the model.

TABLE 3. Experimental results of using different text networks on MIRFlickr-25K dataset.

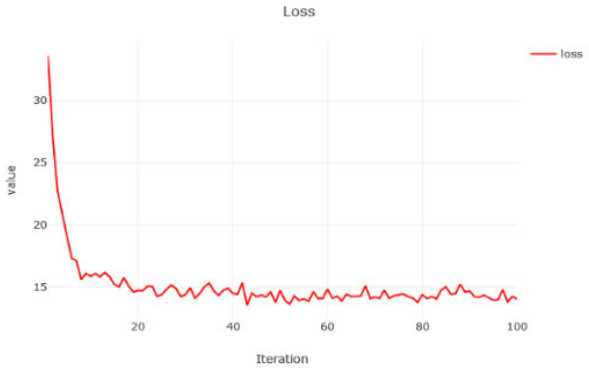
Text network structure	I-T	T-I
Fully Connected(DCMH)	0.7976	0.7947
Multi-scale fusion (SSAH)	0.8148	0.8197
Graph Convolutional Network	0.9019	0.9089

E. CONVERGENCE ANALYSIS

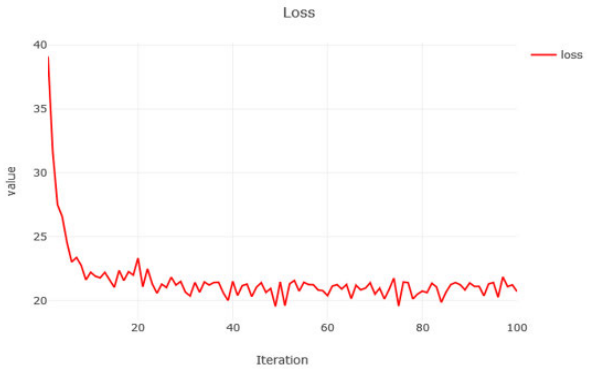
To verify the convergence of our model optimization algorithm. We conducted experiments on three datasets and plotted the convergence curves of the three datasets in the case of 64-bit code length in Figure 10. We can see in the figure that our objective loss decreases rapidly and tends to be stable.

F. FUTURE RESEARCH

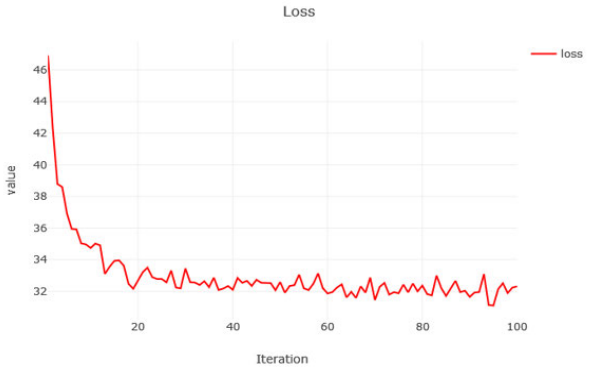
When building the similarity matrix, we refine the similarity between instances by adding non-co-occurrence label information between instances. However, we assign the same weight to instances for co-occurrence label information (1-1) and non-co-occurrence label information (0-0). Obviously, the information implied in the former is more important, and we can reduce the contribution of the latter to the construction of similar matrix, which is worth studying.



MIRFlickr-25K



NUSWIDE



IAPR TC-12

FIGURE 10. The convergence curves.

VI. CONCLUSION

In this paper, we propose an effective cross-modal hashing retrieval method called Non-co-occurrence enhanced Multi-label cross-modal hashing retrieval based on Graph convolutional Network (MHGCN). We use a novel multi-label non-co-occurrence similarity measure to construct our similarity matrix, which makes our similarity matrix more refined and makes it easier to distinguish similar instances. Compared with the single-label method, our similarity matrix is more delicate; Compared with the multi-label method, our similarity matrix can better distinguish instances with consistent co-occurrence information. In addition, we use a Graph Convolutional Network with a strong representation ability to

extract features, which enables the hash codes generated by the model to retain more semantic information. By analyzing the experimental performance of our MHGCN method on three benchmark datasets. Our model has an excellent performance in cross-modal hashing retrieval tasks.

REFERENCES

[1] 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.

[2] Y. Xie, X. Zeng, T. Wang, L. Xu, and D. Wang, "Multiple deep neural networks with multiple labels for cross-modal hashing retrieval," *Eng. Appl. Artif. Intell.*, vol. 114, Sep. 2022, Art. no. 105090.

[3] X. Zou, X. Wang, E. M. Bakker, and S. Wu, "Multi-label semantics preserving based deep cross-modal hashing," *Signal Process., Image Commun.*, vol. 93, Apr. 2021, Art. no. 116131.

[4] 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.

[5] 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.

[6] T. Wang, L. Zhu, Z. Cheng, J. Li, and Z. Gao, "Unsupervised deep cross-modal hashing with virtual label regression," *Neurocomputing*, vol. 386, pp. 84–96, Apr. 2020.

[7] 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.

[8] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[9] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," 2014, *arXiv:1405.3531*.

[10] 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.

[11] 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.

[12] 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.

[13] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Trans. Neural Netw.*, vol. 20, no. 1, pp. 61–80, Dec. 2009.

[14] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–15.

[15] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," 2017, *arXiv:1710.10903*.

[16] H. Wang, "Learning graph representation with generative adversarial nets," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 8, pp. 3090–3103, Aug. 2021.

[17] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: A review," *Neurocomputing*, vol. 187, pp. 27–48, Apr. 2016.

[18] X. Liu, G. Yu, C. Domeniconi, J. Wang, Y. Ren, and M. Guo, "Ranking-based deep cross-modal hashing," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 4400–4407.

[19] M. M. Bronstein, A. M. Bronstein, F. Michel, and N. Paragios, "Data fusion through cross-modality metric learning using similarity-sensitive hashing," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 3594–3601.

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

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

[22] D. Mandal, K. N. Chaudhury, and S. Biswas, "Generalized semantic preserving hashing for cross-modal retrieval," *IEEE Trans. Image Process.*, vol. 28, no. 1, pp. 102–112, Jan. 2019.

[23] C. Deng, Z. Chen, X. Liu, X. Gao, and D. Tao, "Triplet-based deep hashing network for cross-modal retrieval," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 3893–3903, Aug. 2018.

[24] H. Zhu, M. Long, J. Wang, and Y. Cao, "Deep hashing network for efficient similarity retrieval," in *Proc. AAAI Conf. Artif. Intell.*, vol. 30, 2016, pp. 1–7.

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

[26] L. Wu, Y. Wang, and L. Shao, "Cycle-consistent deep generative hashing for cross-modal retrieval," *IEEE Trans. Image Process.*, vol. 28, no. 4, pp. 1602–1612, Apr. 2019.

[27] 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.

[28] Z.-D. Chen, W.-J. Yu, C.-X. Li, L. Nie, and X.-S. Xu, "Dual deep neural networks cross-modal hashing," in *Proc. AAAI Conf. Artif. Intell.*, vol. 32, 2018, pp. 1–8.

[29] 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.

[30] W. Gu, X. Gu, J. Gu, B. Li, Z. Xiong, and W. Wang, "Adversary guided asymmetric hashing for cross-modal retrieval," in *Proc. Int. Conf. Multimedia Retr.*, Jun. 2019, pp. 159–167.

[31] E. Yang, C. Deng, W. Liu, X. Liu, D. Tao, and X. Gao, "Pairwise relationship guided deep hashing for cross-modal retrieval," in *Proc. AAAI Conf. Artif. Intell.*, vol. 31, 2017, pp. 1–8.

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

[33] Y. Cao, M. Long, J. Wang, and P. S. Yu, "Correlation hashing network for efficient cross-modal retrieval," 2016, *arXiv:1602.06697*.

[34] Z. Zhang, Q. Zou, Y. Lin, L. Chen, and S. Wang, "Improved deep hashing with soft pairwise similarity for multi-label image retrieval," *IEEE Trans. Multimedia*, vol. 22, no. 2, pp. 540–553, Feb. 2020.

[35] Z. Ji, W. Yao, W. Wei, H. Song, and H. Pi, "Deep multi-level semantic hashing for cross-modal retrieval," *IEEE Access*, vol. 7, pp. 23667–23674, 2019.

[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] Z. Zhang, Q. Zou, Y. Lin, L. Chen, and S. Wang, "Improved deep hashing with soft pairwise similarity for multi-label image retrieval," 2018, *arXiv:1803.02987*.

[38] X. Wang, X. Zou, E. M. Bakker, and S. Wu, "Self-constraining and attention-based hashing network for bit-scalable cross-modal retrieval," *Neurocomputing*, vol. 400, pp. 255–271, 2020.

[39] Y. Peng, X. Huang, and Y. Zhao, "An overview of cross-media retrieval: Concepts, methodologies, benchmarks, and challenges," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 9, pp. 2372–2385, Sep. 2018.

[40] X. Lu, H. Zhang, J. Sun, Z. Wang, P. Guo, and W. Wan, "Discriminative correlation hashing for supervised cross-modal retrieval," *Signal Process., Image Commun.*, vol. 65, pp. 221–230, Jul. 2018.

[41] H. Chen, G. Ding, Z. Lin, S. Zhao, and J. Han, "Cross-modal image-text retrieval with semantic consistency," in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 1749–1757.

[42] 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.

[43] X. Zhao, G. Ding, Y. Guo, J. Han, and Y. Gao, "TUCH: Turning cross-view hashing into single-view hashing via generative adversarial nets," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Aug. 2017, pp. 1–7.

[44] X. Zhang, H. Lai, and J. Feng, "Attention-aware deep adversarial hashing for cross-modal retrieval," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 591–606.

[45] S. Xiang, H. Deng, L. Zhu, J. Wu, and L. Yu, "Exemplar-based depth inpainting with arbitrary-shape patches and cross-modal matching," *Signal Process., Image Commun.*, vol. 71, pp. 56–65, Feb. 2019.

[46] Z. Wang, X. Liu, H. Li, L. Sheng, J. Yan, X. Wang, and J. Shao, "CAMP: Cross-modal adaptive message passing for text-image retrieval," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 5764–5773.

[47] R. Xu, C. Li, J. Yan, C. Deng, and X. Liu, "Graph convolutional network hashing for cross-modal retrieval," in *Proc. IJCAI*, 2019, pp. 982–988.
[48] P. Hu, X. Wang, L. Zhen, and D. Peng, "Separated variational hashing networks for cross-modal retrieval," in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 1721–1729.
[49] Y. Cao, M. Long, J. Wang, and S. Liu, "Collective deep quantization for efficient cross-modal retrieval," in *Proc. AAAI Conf. Artif. Intell.*, vol. 31, 2017, pp. 1–7.
[50] M. J. Huiskes and M. S. Lew, "The MIR Flickr retrieval evaluation," in *Proc. 1st ACM Int. Conf. Multimedia Inf. Retr.*, Oct. 2008, pp. 39–43.
[51] 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.*, Jul. 2009, pp. 1–9.
[52] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 740–755.
[53] H. J. Escalante, C. A. Hernández, J. A. Gonzalez, A. López-López, M. Montes, E. F. Morales, L. E. Sucar, L. Villaseñor, and M. Grubinger, "The segmented and annotated IAPR TC-12 benchmark," *Comput. Vis. Image Understand.*, vol. 114, no. 4, pp. 419–428, 2010.
[54] M.-L. Zhang and Z.-H. Zhou, "A review on multi-label learning algorithms," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 8, pp. 1819–1837, Jun. 2014.
[55] M. S. Sorower, "A literature survey on algorithms for multi-label learning," *Oregon State Univ., Corvallis*, vol. 18, no. 1, p. 25, 2010.
[56] S. Wu, A. Oerlemans, E. M. Bakker, and M. S. Lew, "A comprehensive evaluation of local detectors and descriptors," *Signal Process., Image Commun.*, vol. 59, pp. 150–167, Nov. 2017.
[57] A. Andoni and P. Indyk, "Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions," *Commun. ACM*, vol. 51, no. 1, pp. 117–122, Jan. 2008.
[58] T. Liu, A. Moore, K. Yang, and A. Gray, "An investigation of practical approximate nearest neighbor algorithms," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 17, 2004, pp. 1–13.
[59] J. Liu, C. Xu, and H. Lu, "Cross-media retrieval: State-of-the-art and open issues," *Int. J. Multimedia Intell. Secur.*, vol. 1, no. 1, pp. 33–52, 2010.
[60] K. Wang, Q. Yin, W. Wang, S. Wu, and L. Wang, "A comprehensive survey on cross-modal retrieval," 2016, *arXiv:1607.06215*.
[61] W. Kong and W.-J. Li, "Isotropic hashing," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1–15.
[62] S. Wu, A. Oerlemans, E. M. Bakker, and M. S. Lew, "Deep binary codes for large scale image retrieval," *Neurocomputing*, vol. 257, pp. 5–15, Sep. 2017.
[63] 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.*, 2019, vol. 33, no. 1, pp. 176–183.
[64] J. Zhang, Y. Peng, and M. Yuan, "Unsupervised generative adversarial cross-modal hashing," in *Proc. AAAI Conf. Artif. Intell.*, vol. 32, 2018, pp. 1–8.
[65] G. Wu, Z. Lin, J. Han, L. Liu, G. Ding, B. Zhang, and J. Shen, "Unsupervised deep hashing via binary latent factor models for large-scale cross-modal retrieval," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 2854–2860.
[66] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 2, pp. 84–90, Jun. 2012.
[67] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.
[68] 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.
[69] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proc. COMPSTAT*. Cham, Switzerland: Springer, 2010, pp. 177–186.
[70] Z. Zhong, L. Zheng, Z. Luo, S. Li, and Y. Yang, "Invariance matters: Exemplar memory for domain adaptive person re-identification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 598–607.
[71] C.-Y. Chiu and S. Markchit, "Effective and efficient indexing in cross-modal hashing-based datasets," *Signal Process., Image Commun.*, vol. 80, Feb. 2020, Art. no. 115650.
[72] 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.
[73] Z. Zhou and W. Li, "Learning to hash for big data: Current status and future trends," *Chin. Sci. Bull.*, vol. 60, nos. 5–6, pp. 485–490, Feb. 2015.
[74] M. Hu, Y. Yang, F. Shen, N. Xie, R. Hong, and H. T. Shen, "Collective reconstructive embeddings for cross-modal hashing," *IEEE Trans. Image Process.*, vol. 28, no. 6, pp. 2770–2784, Jun. 2019.
[75] 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.
[76] 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.
[77] 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.
[78] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*.
[79] N. De Cao and T. Kipf, "An implicit generative model for small molecular graphs," 2018, *arXiv:1805.11973*.

MINGYONG LI (Member, IEEE) received the B.S. degree from Central China Normal University, in 2003, and the Ph.D. degree from the Department of Computer Science and Technology, Donghua University, in 2021. He is currently an Associate Professor with the School of Computer and Information Science, Chongqing Normal University. His current research interests include cross-modal big data processing, large-scale data retrieval, and deep learning.

JIABAO FAN is currently pursuing the master's degree with the College of Computer and Information Science, Chongqing Normal University. His current research interests include cross-modal retrieval and deep learning.

ZIYONG LIN is currently pursuing the master's degree with the College of Computer and Information Science, Chongqing Normal University. His current research interests include cross-modal retrieval and deep learning.