Aggregation-based Graph Convolutional Hashing for Unsupervised Cross-modal Retrieval

Peng-Fei Zhang, Yang Li, Zi Huang, Xin-Shun Xu

***Abstract*—Cross-modal hashing has sparked much attention in large-scale information retrieval for its storage and query efficiency. Despite the great success achieved by supervised approaches, existing unsupervised hashing methods still suffer from the lack of reliable learning guidance and cross-modal discrepancy. In this paper, we propose Aggregation-based Graph Convolutional Hashing (AGCH) to tackle these obstacles. First, considering that a single similarity metric can hardly represent data relationships comprehensively, we develop an efficient aggregation strategy that utilises multiple metrics to construct a more precise affinity matrix for learning. Specifically, we apply various similarity measures to exploit the structural information of multiple modalities from different perspectives and then aggregate the obtained information to produce a joint similarity matrix. Furthermore, a novel deep model is designed to learn unified binary codes across different modalities, where the key components include modality-specific encoders, Graph Convolutional Networks (GCNs) and a fusion module. The modality-specific encoders are tasked to learn feature embeddings for each individual modality. On this basis, we leverage GCNs to further excavate the semantic structure of data, along with a fusion module to correlate different modalities. Extensive experiments on three real-world datasets demonstrate that the proposed method significantly outperforms the state-of-the-art competitors.**

***Index Terms*—Multimodal, Unsupervised Hashing, Cross-Modal Search, Graph Convolutional Networks.**

I. Introduction

In the light of an unprecedented amount of multimedia data nowadays, there exists an urgent demand for effective and efficient search techniques. To fulfill this request, proposed hashing based methods aim to map high-dimensional data into compact binary codes with the original semantic relations preserved [1]–[8]. Due to the attractive properties of fast query speed and low storage consumption, hashing technology has aroused broad interests [9]–[15].

Most conventional hashing methods are mainly focused on unimodal retrieval [2], [16]–[21], where the query and the database are homogeneous, for example, both the query items and the database items are images. However, in many real-world applications, data often comes in various types, such as text, image, audio or video, which implies that it is necessary to perform searches across different modalities. For instance, given a query item of text, images or videos, those are semantically relevant to the query will be returned. In this case, the unimodal approaches are no longer applicable. Consequently, Cross-Modal Hashing (CMH) [22]–[27] has been vigorously studied, including supervised and unsupervised methods.

Supervised CMH methods [28]–[33] directly leverage pre-annotated labels or pre-obtained similarity relationships as the unified guidance to learn desired hash codes. With such strong supervision, supervised CMH can achieve promising results. Recently, prevalent deep neural networks have further advanced the development of supervised methods as high-level nonlinear features with abundant semantic information able to be extracted [34]–[37]. However, in the real-world, it is a tough job to label a large-scale dataset, which is highly time consuming and always requires expertise. In contrast, unsupervised CMH [30], [38]–[41] generates binary codes by exploiting the underlying geometric structure of the training data in the absence of semantic tags, thus obtaining a less competitive performance. Nevertheless, it is more practical than supervised methods in practice because manual annotation is not required.

Graph-based hashing has been a long-standing research interest, which traditionally takes affinity graphs as the guidance for learning [10], [42], [43]. For example, Spectral Hashing (SH) [16] learns binary codes by solving the graph Laplacian eigenvectors. However, as the method requires a global similarity measure, solving the optimization problem is quite time-consuming. To overcome this shortcoming, [17] constructs a low-rank similarity preserving algorithm to solve in linear time. [44] designs a bit-wise sequential learning strategy to effectively approximate the global affinity via feature transformation. Furthermore, considering the binary constraints, [45] transforms the original optimization problem into two subproblems so as to derive the optimal solution. Recently, there have also been a number of work studies attempting to incorporate graphs in the process of feature extraction so as to learn more semantics, such as Graph Convolutional Network Hashing (GCNH) [13] and Graph Convolutional Hashing (GCH) [46]. More specifically, GCNH has been proposed to deal with semi-supervised retrieval via a direct convolution operation on the input data and the anchor graph. GCH utilizes both Convolutional Neural Networks (CNNs) and a Graph Convolutional Network (GCN) in the learning framework to learn more discriminative hash codes.

Although there are many methods with promising results presented, deep hashing for unsupervised search is rarely considered. In this paper, we focus on the retrieval task under the unsupervised cross-modal setting. To pursue high-quality hash codes and functions, two problems need to be considered.

This work partially is supported by Australian Research Council Discovery Project (ARC DP190102353), and China Scholarship Council.

Peng-Fei Zhang, Yang Li and Zi Huang (corresponding author) are with the School of Information Technology & Electrical Engineering, the University of Queensland, email: mima.zpf@gmail.com, yang.li@uq.edu.au and huang@itee.uq.edu.au.

Xin-Shun Xu is with the School of Software, Shandong University, Jinan, China, email: xuxinshun@sdu.edu.cn.

The first is how to obtain reliable supervisory guidance for learning. Second, since the data of different media types generally reside in different feature spaces and thus have specific characteristics, the heterogeneity problem needs to be solved.

To handle the above issues, in this paper, we propose a novel deep cross-modal hashing model, called Aggregation-based Graph Convolutional Hashing (AGCH), for large-scale unsupervised heterogeneous data retrieval. On the one hand, to gain semantics comprehensively, a novel aggregation-based similarity matrix construction approach is developed. More concretely, as multi-modal data describes objects from different views and are characterized by different statistical properties, they usually contain complementary and diverse information. To take full advantage of this, we adaptively incorporate intrinsic information embedded in every individual modality to achieve a comprehensive understanding of the data item relations. Furthermore, in contrast to previous methods that exploit data relationships from only one perspective, i.e., according to a single distance-based metric, we oversee it through various metrics to exploit rich structural information contained in multi-modal data. In light of this, we obtain a reliable supervised signal for learning with less bias. On the other hand, to handle the data heterogeneity, a novel deep model is built, consisting of multiple sub-networks, each of which is for one modality to learn modality-specific hashing functions. In particular, each sub-network takes a deep network as the data encoder and further employs a Graph Convolutional Network (GCN) [47] to deeply explore the underlying neighborhood structure, benefiting discriminative binary code learning. A fusion module is proposed to connect each modality to learn unified binary representations. An effective objective function is designed to preserve both the intra- and the inter-modal consistencies. Without loss of generality, the model can be easily extended to the cases with more modalities by adding extra sub-networks, each of which corresponds to one modality. The overview of our proposed method is diagrammed in Figure 1 and the contributions are summarized as follows.

- To the best of our knowledge, this paper describes the first attempt to construct an affinity graph by using various distance-based similarity metrics. In addition, we merge the diverse and complementary information from different modalities, through which the intrinsic semantic structures are well captured, with a better representation of the similarity relationship among multi-modal data.
- This is also the first work that applies GCNs in unsupervised hashing learning. Our model enables the intra- and inter-modal consistency preservation and the interaction between different modals. More importantly, it fully considers the neighbourhood relevance in the learning procedure. As a consequence, binary representations of high quality are obtained.
- The promising results obtained from extensive experiments on three benchmark datasets demonstrate the effectiveness of the proposed AGCH algorithm.

In the following sections, we will in turn introduce the related work (Section II), the proposed algorithm (Section III) and the validating experiments (Section IV). Finally, the conclusion will be given in Section V.

II. Related work

In this section, we briefly review the related cross-modal hashing work, including Hashing and Graph Convolutional Networks.

A. Hashing

Existing Hashing methods can be divided into non-deep (shallow) and deep methods according to whether deep neural networks are utilized to perform non-linear feature learning or not.

1) Non-deep Hashing: Non-deep methodologies take hand-crafted features as the input, which can be further divided into supervised methods and unsupervised methods, according to whether supervised information is used.

One the one hand, supervised approaches directly utilize label information or similarity matrix to supervise the learning procedure. Representative techniques include Cross View Hashing (CVH) [48], Semantics Preserving Hashing (SePH) [49], Semantic Correlation Maximization Hashing (SCM) [50], Supervised Discrete Manifold-embedded Cross-modal Hashing (SDMCH) [32], scalable supervised asymmetric hashing (SSAH) [51] and Subspace Relation Learning for Cross-modal Hashing (SRLCH) [52]. CVH extends spectral hashing from single-modal scenarios into multi-modal settings, and learns binary embedding functions by minimizing the similarity-weighted distances between heterogeneous data. SePH proposes a probability distribution-based model which learns binary codes by minimizing the KL-divergence. SCM designs a scalable algorithm that applies spectral relaxation and imposes orthogonality constraints to learn balanced binary codes. SDMCH integrates manifold learning into the learning framework and fully leverages semantics to improve efficiency. SSAH learns dual-stream asymmetric hashing functions to fully explore the semantic information, resulting in high-quality discriminative hashing codes. SRLCH transforms class labels into a subspace so as to exploit semantic relation information to learn more discriminative hash codes.

One the other hand, without semantic labels available, unsupervised methods can only exploit the intra- and inter-modal correlations among original data to learn hashing codes and mapping functions. Inter-Media Hashing (IMH) [39] is one of the representative unsupervised methods that learns linear functions to generate binary codes under the intra- and inter-view consistency constraints. Collective Matrix Factorization Hashing (CMFH) [40] adopts the matrix factorization (MF) technique to collectively learn cross-view binary codes. The MF technology is also used in Latent Semantic Sparse Hashing (LSSH) [38] to learn text concepts, with the sparse coding technique to deal with image modality. Robust and Flexible Discrete Hashing (RFDH) [53] utilizes the discrete matrix decomposition to directly learn hash codes, at the same time, it employs $\ell_{2,1}$-norm to boost the robustness of the algorithm. In spite of the impressive progress achieved on unsupervised

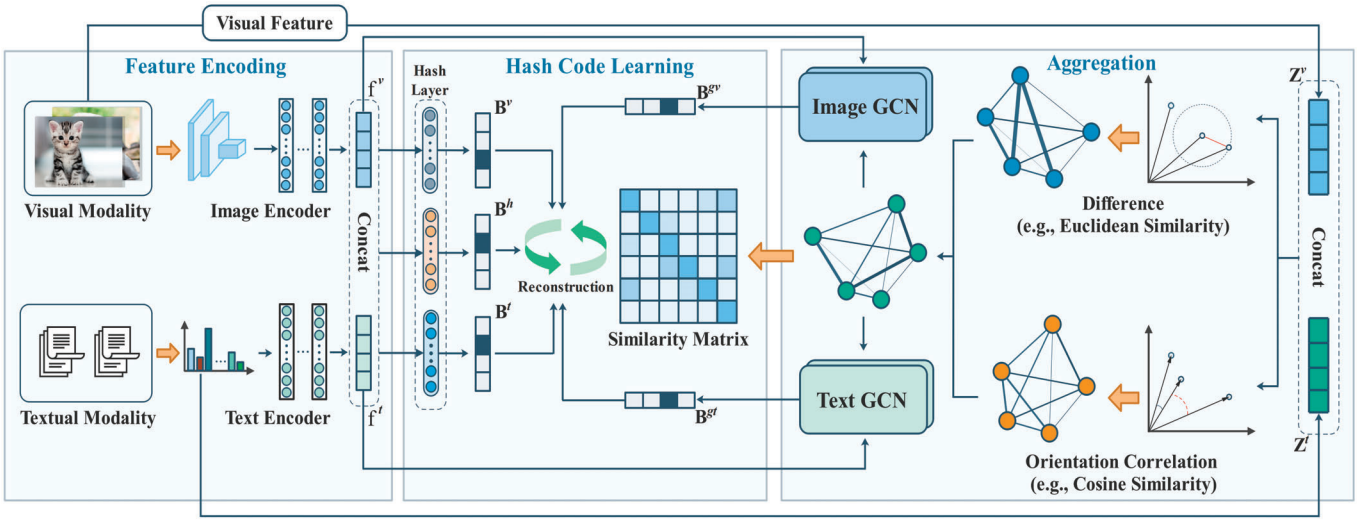


Fig. 1: Illustration of the proposed AGCH framework, which consists of three main parts: aggregation-based similarity matrix construction, feature encoding and hash codes learning.

hashing methods, their ability to capture non-linear relations across different modalities is still limited due to the insufficient semantics of hand-crafted features.

2) Deep Hashing: Revolutionary deep learning has shown its superiority in many research fields, e.g., computer vision, data mining and information retrieval. Deep hashing methods have also been proposed, which can simultaneously extract non-linear features and perform projecting function learning. Existing efforts are mainly made to build hashing models to cope with supervised scenarios, such as Deep Visual-Semantic Hashing (DVSH) [54], Deep Cross-Modal Hashing (DCMH) [55], Pairwise Relation Guided Deep Hashing (PRDH) [56], and Scalable Deep Multimodal Learning (SDML) [57]. More detailed, DVSH is a hybrid deep hashing network that aims to capture spatial dependency and temporal dynamics for images and texts, respectively, in order to facilitate cross-modal embedding learning. DCMH performs simultaneous hashing code and function learning with an end-to-end network. PRDH imposes various kinds of pairwise constraints to pursue intrinsic semantics, such that high-quality binary codes are learnt. SDML builds a scalable learning model that contains a set of modality-specific networks to deal with different modalities.

In comparison, unsupervised deep CMH receives less attention, with relatively fewer approaches available. Specifically, Deep Binary Reconstruction (DBRC) [58] presents a binary reconstruction model to enable the multi-modal relation modeling and hashing learning in one step. Unsupervised Deep Cross-Modal Hashing (UDCMH) [59] integrates the matrix factorization into deep neural networks to learn discrete hashing codes in a self-taught manner. Deep Joint-Semantics Reconstructing Hashing (DJSRH) [60] constructs a high-order similarity matrix by treating the original similarity matrix as features to uncover the latent semantic structure to give reliable supervisory signals for the learning procedure.

B. Graph Convolutional Network

Graph Neural Networks (GNNs) are effective for graph representation learning and have raised a surge of research interest in classification, prediction and so on. GNNs are first proposed in [61], which leverages a recursive neighbourhood aggregation strategy to calculate the feature for each node. By aggregating the features of the neighbours in each iteration, the k-hop neighbourhood structure information can be captured, which significantly improves performance. Some other representative work includes Semi-supervised Graph Convolutional Network [47], GraphSAGE [62], Graph Attention Networks (GATs) [63] and JK-Nets [64]. More specifically, the Semi-supervised Graph Convolutional Network performs a first-order approximation of spectral graph convolutions, which is scalable to a large number of graph edges. Instead of the transductive learning, GraphSAGE designs an inductive framework which leverages the local structure information to encode features for unseen data. GATs proposes a self-attention strategy which can deal in parallel with nodes in a graph. JK-Nets explores the node-specific neighborhood ranges to enable an adaptive structure-aware representation learning. In this paper, we take advantage of the strong representation power of Graph Convolutional Networks to boost the performance of hashing learning on unsupervised cross-model hashing.

III. Method

In this section, we elaborate our work, including the notation and problem definition, the details of the proposed method, the optimization scheme and its extensions. For ease of representation, we restrict our discussion to a bi-modal case, i.e., image and text, although it can be easily extended to more modalities via a slight adjustment.

A. Notation and Problem Definition

In this paper, boldface uppercase letters, e.g., $\mathbf{Q}$, are used to represent matrices and boldface lowercase letters, e.g., $\mathbf{q}$,

denote vectors. $\mathbf{Q}_{i*}$ and $\mathbf{Q}_{*j}$ represent the i-th row and j-th column of $\mathbf{Q}$, respectively. $\mathbf{Q}_{ij}$ is the element at the position (i, j) of matrix $\mathbf{Q}$. The transpose of $\mathbf{Q}$ is indicated as $\mathbf{Q}^T$ and $\mathbf{Q}^{-1}$ denotes the inverse of the matrix $\mathbf{Q}$. In addition, $\|\cdot\|_F$ denotes the Frobenius of a vector or matrix, $sign(\cdot)$ is an element-wise sign function which is defined as follows:

$$sign(\mathbf{x}) = \begin{cases} 1 & \mathbf{x} > 0 \\ -1 & \mathbf{x} \leq 0. \end{cases} \tag{1}$$

We also define the Cosine distance between two vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ as follows:

$$\cos(\mathbf{x}_i, \mathbf{x}_j) = \frac{\mathbf{x}_i^T \mathbf{x}_j}{\|\mathbf{x}_i\|_2 \|\mathbf{x}_j\|_2}. \tag{2}$$

where $\|\cdot\|_2$ denotes ℓ_2 norm.

The symbol $\odot$ signifies the Hadamard matrix product (i.e., element-wise product), which is defined as follows:

$$\begin{aligned} &\mathbf{X} \odot \mathbf{Y} \\ &= \begin{bmatrix} \mathbf{x}_{11} & \mathbf{x}_{12} & \dots & \mathbf{x}_{1n} \\ \mathbf{x}_{21} & \mathbf{x}_{22} & \dots & \mathbf{x}_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{x}_{n1} & \mathbf{x}_{n2} & \dots & \mathbf{x}_{nn} \end{bmatrix} \odot \begin{bmatrix} \mathbf{y}_{11} & \mathbf{y}_{12} & \dots & \mathbf{y}_{1n} \\ \mathbf{y}_{21} & \mathbf{y}_{22} & \dots & \mathbf{y}_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{y}_{n1} & \mathbf{y}_{n2} & \dots & \mathbf{y}_{nn} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{x}_{11} \cdot \mathbf{y}_{11} & \mathbf{x}_{12} \cdot \mathbf{y}_{12} & \dots & \mathbf{x}_{1n} \cdot \mathbf{y}_{1n} \\ \mathbf{x}_{21} \cdot \mathbf{y}_{21} & \mathbf{x}_{22} \cdot \mathbf{y}_{22} & \dots & \mathbf{x}_{2n} \cdot \mathbf{y}_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{x}_{n1} \cdot \mathbf{y}_{n1} & \mathbf{x}_{n2} \cdot \mathbf{y}_{n2} & \dots & \mathbf{x}_{nn} \cdot \mathbf{y}_{nn} \end{bmatrix} \end{aligned} \tag{3}$$

where $\mathbf{X} = \{x_i\}_{i=1}^n$ and $\mathbf{Y} = \{y_i\}_{i=1}^n$.

Suppose that $\boldsymbol{O} = \{\mathbf{o}_i\}_{i=1}^n$ is a multi-modal dataset with n coupled instances, $\mathbf{o}_i = (\mathbf{o}_i^v, \mathbf{o}_i^t)$, where $\mathbf{o}_i^v \in \mathbb{R}^{d_1}$ and $\mathbf{o}_i^t \in \mathbb{R}^{d_2}$ denote the d_1-dimensional image feature vector and the d_2-dimensional text feature vector for the i-th data point pair. In experiments, we make an assumption that samples from both modalities in the training set are all observed.

Given the training data, the ultimate purpose of our method is to learn modality-specific projecting functions $f(\mathbf{o}^v; \theta_\mathbf{v})$ and $g(\mathbf{o}^t; \theta_\mathbf{t})$ for images and texts, where $\theta_\mathbf{v}$ and $\theta_\mathbf{t}$ are network parameters, and gain the unified binary representations $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_n] \in \{-1, 1\}^{c \times n}$, where $\mathbf{b}_i$ is the binary code of the i-th instance and c is the code length. In principle, $\mathbf{b}_i$ and $\mathbf{b}_j$ learnt by embedding functions are expected to accurately preserve the similarity in the original multi-modal spaces.

B. Framework

As shown in Figure 1, our model is an end-to-end learning framework that contains four major components, i.e., the Image Encoder and Image GCN, Text Encoder and Text GCN, to deal with images and texts, separately, and a fusion module to yield modal-invariant binary embeddings.

In each epoch, along the hierarchical processing of the Image Encoder, the images $\mathbf{o}^v$ from the training set are first transformed into the nonlinear features $\mathbf{f}^v \in \mathbb{R}^{d_{e1} \times m}$, which are further fed into the hashing layer to generate modality-specific binary representations $\mathbf{B}^v \in \{1, -1\}^{c \times m}$, where c and m denote the code length and batch size, respectively. The Text Encoder is tasked to extract high-level features $\mathbf{f}^t \in \mathbb{R}^{d_{e2} \times m}$ and accordingly generates the hash codes $\mathbf{B}^t \in \{1, -1\}^{c \times m}$ of the input texts $\mathbf{o}^t$. This binary generation process is expressed as follows:

$$\begin{aligned} \mathbf{B}^v &= sign\left(f\left(\mathbf{o}^v; \theta_v\right)\right), \\ \mathbf{B}^t &= sign\left(f\left(\mathbf{o}^t; \theta_t\right)\right), \end{aligned} \tag{4}$$

where θ_v and θ_t are weight parameters of their corresponding networks. This process is also employed to generate the binary representations for queries in the testing stage. In addition, we notice that in the training stage, the $sign(\cdot)$ function would cause the intractable back-propagate gradient problem. To avoid this, we replace it with the $tanh(\cdot)$ function.

In the following stage, the high-level nonlinear features $\mathbf{f}^v$ and $\mathbf{f}^t$ from both modalities will be aggregated and then embedded in the fusion branch to gain the unified binary codes $\mathbf{B}^h \in \{1, -1\}^{c \times m}$ for different modalities as follows:

$$\begin{aligned} \mathbf{o}^h &= \mathbf{f}^v \oplus \mathbf{f}^t, \\ \mathbf{B}^h &= tanh(f(\mathbf{o}^h; \theta_h)), \end{aligned} \tag{5}$$

where $\oplus$ is the concatenation operation, θ_h denotes parameters of the sub-network. By the non-linear operation of the fusion branch, in which the meaningful information from multiple sources is adaptively selected and the irrelevant information is ignored or suppressed, the proposed method can learn more effective and representative binary representations.

Synchronously, $\mathbf{f}^v$ and $\mathbf{f}^t$ with the adjacency matrix $\widetilde{\mathbf{A}}$, defined in the following subsection, are fed into the corresponding GCN modules to obtain more structural semantics. The graph convolution process for each layer is written as follows:

$$\mathbf{H}_{(l)}^k = \sigma_{(l)}\left(\widetilde{\mathbf{D}}^{-1/2} \widetilde{\mathbf{A}} \widetilde{\mathbf{D}}^{-1/2} \mathbf{H}_{(l-1)}^k \mathbf{W}_{(l)}^k\right), \quad s.t. \quad k \in \{v, t\}, \tag{6}$$

where $\widetilde{\mathbf{D}}_{ii} = \sum_j \widetilde{\mathbf{A}}_{ij}$ and $\mathbf{W}_{(l)}^k$ serve as the convolutional filters for the k-modality at the l-th layer. $\sigma_{(l)}$ is the activation function for the l-th GCN layer. $\mathbf{H}_{(l-1)}^k \in \mathbb{R}^{d_{l-1} \times m}$ and $\mathbf{H}_{(l)}^k \in \mathbb{R}^{d_l \times m}$ represent the corresponding input and output respectively. In the first layer of each GCN (i.e., $l = 1$), $\mathbf{H}_{(l-1)}^k$ represents the features extracted from the penultimate layer of the corresponding encoder, i.e., $\mathbf{f}^v$ and $\mathbf{f}^t$. With GCN modules, we can obtain the binary codes as:

$$\begin{aligned} \mathbf{B}^{gv} &= sign\left(f\left(\mathbf{f}^v; \theta_{gv}\right)\right), \\ \mathbf{B}^{gt} &= sign\left(f\left(\mathbf{f}^t; \theta_{gt}\right)\right), \end{aligned} \tag{7}$$

where θ_{gv} and θ_{gt} are weight parameters of Image GCN and Text GCN, respectively.

From Eq. (6) and (7), we can see that for a node, through the graph convolutional layer in which the neighbors of such a node are concatenated by a weighted summation, the new features will be assigned to this node, which means the features of neighboring nodes are encouraged to be closer. As a result, the obtained binary codes reflect the intrinsic spatial relationship in feature space.

In addition, an effective affinity matrix construction strategy and corresponding similarity constraints are imposed to optimize the learning procedure, which are elaborated below.

C. Similarity Construction

In unsupervised cross-modal learning, it is rather critical to faithfully capture the underlying neighbourhood structure of the training data as a reliable relationship indicator can guide the learning towards a desired result. To this end, existing methods generally utilize pre-trained deep neural networks to extract nonlinear features to construct an affinity matrix and learn binary codes by nearing it, which has been proved feasible [59], [60], [65]. Inspired by these work, we propose an aggregation-based affinity matrix construction scheme. For the image of the i-th pair in each batch, we feed it into the Alexnet pre-trained on ImageNet, and extract the d_v-dimensional representation $\mathbf{Z}_{i*}^v = [\mathbf{Z}_{i1}^v, \mathbf{Z}_{i2}^v, ..., \mathbf{Z}_{id_v}^v] \in \mathbb{R}^{d_v \times m}$ from the penultimate (fc-7) layer as its original high-level representation. For its corresponding text in the pair, we directly use its d_t-dimensional BoW feature or the original LDA topic vector as the descriptor, denoted as $\mathbf{Z}_{i*}^t = [\mathbf{Z}_{i1}^t, \mathbf{Z}_{i2}^t, ..., \mathbf{Z}_{id_t}^t] \in \mathbb{R}^{d_t \times m}$. We normalize these features and integrate them together:

$$\widetilde{\mathbf{Z}}_{i*} = [\gamma_v[\widetilde{\mathbf{Z}}_{i1}^v, \widetilde{\mathbf{Z}}_{i2}^v, ..., \widetilde{\mathbf{Z}}_{id_v}^v], \gamma_t[\widetilde{\mathbf{Z}}_{i1}^t, \widetilde{\mathbf{Z}}_{i2}^t, ..., \widetilde{\mathbf{Z}}_{id_t}^t]], \tag{8}$$

where $\widetilde{\mathbf{Z}}_{i*}^v$ and $\widetilde{\mathbf{Z}}_{i*}^t$ are the normalized $\mathbf{Z}_{i*}^v$ and $\mathbf{Z}_{i*}^t$, respectively. γ_v and γ_t are weight parameters to regulate the importance of two modalities. It is necessary as the image and text describe data from different views and have different statistical properties, different weights are adopted for two modalities so as to adaptively fuse information from different modalities.

After obtaining the original high-level nonlinear features, we leverage these features to construct the affinity matrix as the supervisory signals. Specifically, we first calculate the inner product of these samples to quantify the similarity between them, which is defined as follows:

$$\mathbf{C}_{ij} = (\widetilde{\mathbf{Z}}_{i*})^T \widetilde{\mathbf{Z}}_{j*}, \tag{9}$$

which is equivalent to the linear combination of the Cosine similarity between original features:

$$\mathbf{C}_{ij} = \sum_{k \in G} \gamma_k \cos\left(\mathbf{Z}_{i*}^k, \mathbf{Z}_{j*}^k\right), \quad s.t. \quad G = \{v, t\}. \tag{10}$$

Generally, the larger $\mathbf{C}_{ij}$ is, the more similar $\mathbf{Z}_{i*}$ and $\mathbf{Z}_{j*}$ are. By leveraging fused information from different modalities to construct the similarity matrix, we obtain more faithful information assuming complimentary or supplementary information carried by various sources (i.e., multi-modalities) can enhance and make up for each other. Nonetheless, there is still room for improvement. First, a single Cosine measure is not enough to excavate sufficient structural semantics. For instance, the inner product of two normalized vectors which have large differences in some dimensions (e.g., $a = [0.5, 0.5, 0.5, 0.5, 0]$, $b = [0.3, 0.5, 0.7, 0.4, 0.1]$, $a \cdot b = 0.95$) is quite similar to that of other pairs with small variances (e.g., $a = [0.5, 0.5, 0.5, 0.5, 0]$, $c = [0.5, 0.5, 0.5, 0.5, 0]$, $a \cdot b = 1$). In this case, the similarity between data instances can not be well differentiated. Besides, this approach only considers the common non-zero dimensions of two attended vectors, which potentially omit much information that might be useful. As a simple example, if a vector has a zero value in one dimension, no matter what a value the other vector has in this dimension, the product between them is always zero.

Considering these, we introduce an auxiliary matrix based on the dimension-wise difference between data items to make up for the shortcomings of the Cosine similarity measure. We denote the auxiliary matrix as $\mathbf{D}$ and aggregate it with the Cosine similarity, which is defined as follows:

$$\mathbf{S} = \mathbf{C} \odot \mathbf{D} \tag{11}$$

In practice, many distance metrics that calculate the difference between data can be adopted to construct $\mathbf{D}$, such as the Euclidean distance, the Mahalanobis distance, and so on. In this work, we employ an Euclidean distance-based similarity, then we have:

$$\begin{aligned} \mathbf{S}_{ij} &= \mathbf{C}_{ij} \cdot \mathbf{D}_{ij} \\ &= \left((\widetilde{\mathbf{Z}}_{i*})^T \widetilde{\mathbf{Z}}_{j*}\right) \cdot \exp(-\sqrt{\|\widetilde{\mathbf{Z}}_{i*} - \widetilde{\mathbf{Z}}_{j*}\|_2}/\rho), \end{aligned} \tag{12}$$

where ρ is a scaling parameter. Intuitively, combining similarity information by evaluating entities from different perspectives can benefit capturing the underlying similarity relationships capture. Besides, this similarity construction approach effectively remedies the shortcoming of the Cosine similarity, thereby further differentiating the similarity between sample points and consequently providing strong supervisory signals for the learning tasks. For example, according to this scheme, the similarity for a and b is $0.95 \times 0.92 = 0.874$ (we set $\rho = 4$ in the experiments), and for a and c, the value is 1. In result, a and c are distinguished. In addition, this matrix also serves as the adjacent matrix $\widetilde{\mathbf{A}}$ in Eq. (6) for the graph convolution operation. In the following quantization procedure, we further regulate $\mathbf{S} = 2\mathbf{S} - 1$ to give a flexible quantization area.

Some existing work [66]–[68] state that the Cosine similarity is optimal in terms of measuring similarity for text data while the Euclidean distance-based similarity metric is better for image data. Our method does not conflict with this scheme but reinforces it as we engage both metrics in similarity measure towards a faithful evaluation of the relationships of sample points. It is further validated in the experiments.

D. Objective Function

In order to ensure that the to-be-learnt binary codes preserve the similarity relationships strictly, the following objective function is defined to keep the intra-modal affinity via reconstructing the similarity matrix:

$$\begin{aligned} &\min_{\theta_k} \mathcal{L}_1 = \sum_{i,j=1}^{m} \sum_{k \in G} \|\mathbf{S}_{ij} - (\mathbf{B}_{i*}^k)^T \mathbf{B}_{j*}^k\|_F^2, \\ &s.t. \quad G = \{v, t, h\}, \end{aligned} \tag{13}$$

where $\mathbf{B}^k = tanh(f(\mathbf{o}^k; \theta_k)), k \in \{v, t, h\}$.

The GCN modules is applied to equip the features from Image and Text Encoders with strong structural relationships,

which favors the precise hash code and embedding function learning. The objective is updated as follows:

$$\begin{aligned}\min_{\theta_k}\mathcal{L}_2 &= \sum_{i,j=1}^{m}\sum_{k\in G'}\|\mathbf{S}_{ij}-(\mathbf{B}_{i*}^{g_k})^T\mathbf{B}_{j*}^{g_k}\|_F^2 \\ &+\sum_{i=1}^{m}\sum_{k\in G'}\|\mathbf{B}_{i*}^{g_k}-\mathbf{B}_{i*}^{k}\|_F^2, \\ s.t.\ \ G' &= \{v,t\},\end{aligned} \tag{14}$$

where $\mathbf{B}^{g_k}=tanh(f(\mathbf{o}^{g_k};\theta_{g_k})), k\in\{v,t\}$ are the outputs of the corresponding GCN modules with the network parameters θ_{g_k}.

It is common knowledge that in multi-modal settings, data instances are from different modalities represented in various feature spaces. To achieve effective cross-modal retrieval, it is essential to correlate heterogeneous data with same semantics. The inter-modal similarity is considered as the similarity between instances of different modalities. To preserve inter-modal consistency, the following loss function is derived as:

$$\begin{aligned}\min_{\theta_k}\mathcal{L}_3 &= \beta\sum_{i,j=1}^{m}\sum_{k\in G'}\|\mathbf{S}_{ij}-(\mathbf{B}_{i*}^{k})^T\mathbf{B}_{j*}^{h}\|_F^2 \\ &+\lambda\sum_{i=1}^{m}\sum_{k\in G'}\|\mathbf{B}_{i*}^{k}-\mathbf{B}_{i*}^{h}\|_F^2, \\ s.t.\ \ G' &= \{v,t\},\end{aligned} \tag{15}$$

where $\mathbf{B}^{k}=tanh(f(\mathbf{o}^{k};\theta_k)), k\in\{v,t\}$.

The final objective function is an ensemble of the beforementioned three individual loss functions as:

$$\begin{aligned}\min_{\theta_k}\mathcal{L} &= \alpha\mathcal{L}_1+\delta\mathcal{L}_2+\mathcal{L}_3 \\ &= \alpha\sum_{i,j=1}^{m}\sum_{k\in G}\|\mathbf{S}_{ij}-(\mathbf{B}_{i*}^{k})^T\mathbf{B}_{j*}^{k}\|_F^2 \\ &+\delta\sum_{i,j=1}^{m}\sum_{k\in G'}\|\mathbf{S}_{ij}-(\mathbf{B}_{i*}^{g_k})^T\mathbf{B}_{j*}^{g_k}\|_F^2 \\ &+\beta\sum_{i,j=1}^{m}\sum_{k\in G'}\|\mathbf{S}_{ij}-(\mathbf{B}_{i*}^{k})^T\mathbf{B}_{j*}^{h}\|_F^2 \\ &+\delta\sum_{i=1}^{m}\sum_{k\in G'}\|\mathbf{B}_{i*}^{g_k}-\mathbf{B}_{i*}^{k}\|_F^2 \\ &+\lambda\sum_{i=1}^{m}\sum_{k\in G'}\|\mathbf{B}_{i*}^{k}-\mathbf{B}_{i*}^{h}\|_F^2, \\ s.t.\ \ G &= \{v,t,h\},\quad G'=\{v,t\}.\end{aligned} \tag{16}$$

In a nut shell, the modality-specific representations $\mathbf{B}^v,\mathbf{B}^t$ from different modalities are aligned through a unified representation $\mathbf{B}^h$, and GCN modules further endow these embeddings with the sufficient neighbourhood information. Furthermore, the learning of these binary descriptors and the update of their corresponding embedding functions are in a collaborative way. In other words, they are interactive with each other during learning. In light of these, the modality gap can be effectively bridged and the quality of the to-be-learnt binary codes and hash functions can be well ensured. In addition, the binary representations $\mathbf{B}^k, k\in\{v,t,h,g_v,g_t\}$ are all normalized. For brevity, we omit this process in the above loss functions.

E. Optimization Algorithm

We solve the problem of Eq. (15) by alternately updating the network parameters $\theta_k,(k\in\{v,t,h,g_v,g_t\})$. More specifically, with four of five parameters (e.g., $\theta_k,(k\in\{t,h,g_v,g_t\})$ fixed, we can easily update the other parameter (e.g., θ_v) by back-propagation algorithm.

The whole optimization scheme is summarized in **Algorithm 1**.

F. Extensions

1) Out-of-sample: Once the model is well-trained, we can use it to generate the hash codes for any new query sample easily. Specifically, given a query $\mathbf{o}_i^k\in\mathbb{R}^{d_k\times 1}, k\in\{v,t\}$, we get its hash code by:

$$\begin{aligned}\mathbf{b}_{\mathbf{o}_i^k} &= sign\left(f\left(\mathbf{o}_i^k;\theta_k\right)\right), \\ s.t.\ \ k &\in\{v,t\}.\end{aligned} \tag{17}$$

2) More Modalities: The proposed method can also be easily extended to the cases with more modalities by adding a new subnetwork for each new modality and slightly modifying the objective function Eq. (16) as

$$\begin{aligned}\min_{\theta_k}\mathcal{L} &= \alpha\mathcal{L}_1+\delta\mathcal{L}_2+\mathcal{L}_3 \\ &= \alpha\sum_{i,j=1}^{m}\sum_{k\in G}\|\mathbf{S}_{ij}-(\mathbf{B}_{i*}^{k})^T\mathbf{B}_{j*}^{k}\|_F^2 \\ &+\delta\sum_{i,j=1}^{m}\sum_{k\in G'}\|\mathbf{S}_{ij}-(\mathbf{B}_{i*}^{g_k})^T\mathbf{B}_{j*}^{g_k}\|_F^2 \\ &+\beta\sum_{i,j=1}^{m}\sum_{k\in G'}\|\mathbf{S}_{ij}-(\mathbf{B}_{i*}^{k})^T\mathbf{B}_{j*}^{h}\|_F^2 \\ &+\delta\sum_{i=1}^{m}\sum_{k\in G'}\|\mathbf{B}_{i*}^{g_k}-\mathbf{B}_{i*}^{k}\|_F^2 \\ &+\lambda\sum_{i=1}^{m}\sum_{k\in G'}\|\mathbf{B}_{i*}^{k}-\mathbf{B}_{i*}^{h}\|_F^2, \\ s.t.\ \ G &= \{h,1,2,...,p\},\quad G'=\{1,2,...,p\}.\end{aligned} \tag{18}$$

where p represents the p-th modality, the high-order similarity matrix $\mathbf{S}$ can be obtained by slightly adjusting our high-order similarity construction defined in the Eq. (12) as:

$$\begin{aligned}\mathbf{S}_{ij} &= \mathbf{C}_{ij}\cdot\mathbf{D}_{ij} \\ &= \left((\widetilde{\mathbf{Z}}_{i*})^T\widetilde{\mathbf{Z}}_{j*}\right)\cdot\exp(-\sqrt{\|\widetilde{\mathbf{Z}}_{i*}-\widetilde{\mathbf{Z}}_{j*}\|_2}/\rho),\end{aligned} \tag{19}$$

where $\widetilde{\mathbf{Z}}_{i*}$ is calculated by :

$$\widetilde{\mathbf{Z}}_{i*}=[\gamma_1[\widetilde{\mathbf{Z}}_{i*}^1],\gamma_2[\widetilde{\mathbf{Z}}_{i*}^2],...,\gamma_p[\widetilde{\mathbf{Z}}_{i*}^p]]. \tag{20}$$

We can easily solve the optimization problem of Eq. (18) by adjusting **Algorithm 1**.

Algorithm 1 Aggregation-based Graph Convolutional Hashing

Input: The training data: $\boldsymbol{O} = \{\mathbf{o}\}_{i=1}^{n}$, mini-batch size: m, hash code length: c, iteration times: t, balance parameters: $\alpha, \beta, \lambda, \gamma_v, \gamma_t, \rho, \delta$.

Output: Parameters of the network: $\theta_k, (k \in \{v, t, h, g_v, g_t\})$.

Procedure:

Randomly initialize the network parameters θ_k;

Repeat:

1. Randomly select m image-text pairs from the dataset to construct a mini-batch;

2. Extract image and text features to the construct similarity matrix;

3. Compute $\mathbf{f}^v = f\left(\mathbf{o}^v; \theta_v\right), \mathbf{f}^t = f\left(\mathbf{o}^t; \theta_t\right), \mathbf{o}^h = \mathbf{f}^v \oplus \mathbf{f}^t$ for samples by forward-propagation.

4. Generate binary codes $\mathbf{B}^v = tanh\left(\mathbf{f}^v\right), \mathbf{B}^t = tanh\left(\mathbf{f}^t\right), \mathbf{B}^h = tanh(f(\mathbf{o}^h; \theta_h)), \mathbf{B}^{gv} = tanh\left(f\left(\mathbf{f}^v; \theta_{gv}\right)\right), \mathbf{B}^{gt} = tanh\left(f\left(\mathbf{f}^t; \theta_{gt}\right)\right)$;

5. Calculate the loss for the whole network with the Equation 16;

6. Update the parameter θ_k by using backpropagation;

Until convergent.

Return: $\theta_k, (k \in \{v, t, h, g_v, g_t\})$.

G. Computational complexity analysis

In this section, we analyse the computational complexity of the proposed AGCN. In each iteration of the training stage, we construct the unified matrix $\widetilde{\mathbf{S}}$, which costs $O((n+m)(d_v+d_t))$. Totally, the computational cost of our iterative process is $O(((n+m)(d_v+d_t))t)$ $(m, c, d_v, d_t \ll n)$ which is linear to the size of datasets. The competitive computational efficiency shows the practicality and effectiveness of our proposed algorithm when dealing with large-scale real-world cross-modal searching tasks.

IV. EXPERIMENTS

To validate the effectiveness of the proposed Aggregation-based Graph Convolutional Hashing (AGCH) method, we conducted extensive experiments and compare it with nine state-of-the-art cross-modal hashing methods on three widely-used benchmark datasets for cross-modal retrieval, i.e., Wiki [69], MIRFlickr-25K [70], and NUS-WIDE [71].

A. Datasets

Wiki [69]: It is a single label dataset with 2,866 coupled image-text instances. All these pairs are labeled with one of ten semantic classes. The dataset has been divided into 2173 training pairs and 693 testing pairs, all of which are used in our experiments.

MIRFlickr-25K [70]: The multi-label MIRFlickr-25K dataset currently consists of 25,000 images with corresponding textual tags from 24 unique categories, which are crawled from Flickr website. The dataset also provides a 1,386-dimensional feature vector derived from PCA on the binary tagging vector to represent the corresponding textual content. In our experiments, for fair comparison, following previous work [59], [60], we randomly choose 5,000 samples and 2,000 samples for training and testing, respectively.

TABLE I: Statistics of Three Benchmark Datasets

Dataset	Wiki	MIRFlickr-25K	NUS-WIDE
Database	2,866	25,000	186,577
Training	2173	5,000	5,000
Testing	693	2,000	2,000
Labels	10	24	10

NUS-WIDE [71]: There are 269,648 images and it corresponding textual tags collected from Flickr in the NUS-WIDE dataset. 10 most commonly used concepts along with corresponding 186,577 images-tags instance pairs are selected as the final experimental dataset. Each visual-text pair is annotated by at least 1 of 10 concepts, for the text of each image, we choose an index vector of 1,000 annotated tags with the highest frequency. In experiments, the numbers of training and testing instances is the same as that on MIRFlickr-25K.

We summarize the statistics of two dataset in Table I.

B. Baselines and Evaluation Metric

In experiments, both state-of-the-art non-deep and deep unsupervised approaches are adopted as baselines for comparison, including: CVH [48], IMH [39], LCMH [22], CMFH [40], LSSH [38], DBRC [58], RFDH [53], UDCMH [59], DJSRH [60]. In particular, CVH, IMH, LCMH, CMFH, LSSH, RFDH are shallow models, and the others are deep learning based methods.

We focus on two retrieval tasks: (1) "Image-to-Text", which searches relevant texts given any image query. (2) "Text-to-Image", which uses texts to retrieve similar images in the database.

To evaluate the effectiveness of the proposed AGCH and all compared methods, we choose the widely-used Mean Average Precision (mAP) to evaluate the performance of all compared methods, which can well reflect both ranking information and precision. More specifically, for a set of queries $\mathbf{Q} = [\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_p]$, mAP is defined as follows:

$$mAP = \frac{1}{p}\sum_{i=1}^{p} AP, \tag{21}$$

where p is the size of the query set $\mathbf{Q}$, and AP (Average Precision) is defined as:

$$AP = \frac{1}{N_g}\sum_{r=1}^{n} P_i(r)\zeta_i(r), \tag{22}$$

where N_g is the number of ground-truth neighbors of the query $\mathbf{q}_i$ in the database, n is the number of entities in the database, $P_i(r)$ denotes the precision of the top r retrieved entities, and $\zeta_i(r) = 1$ if the r-th retrieved entity is a ground-truth neighbour and $\zeta_i(r) = 0$, otherwise. The ground-truth neighbors are defined as those sharing at least one semantic label.

Besides, the precision-recall (PR) and the top-N precision curves are also adopted as important evaluation criteria in terms of the effectiveness of hashing algorithms.

TABLE II: The mAP results of all methods on Wiki ($I \rightarrow T$ means the search task of Image-to-Text, and vice versa). The best results are shown in boldface.

Task	Method	16 bits	32 bits	64 bits	128 bits
$I \rightarrow T$	CVH	0.179	0.162	0.153	0.149
	IMH	0.201	0.203	0.204	0.195
	LCMH	0.115	0.124	0.134	0.149
	CMFH	0.251	0.253	0.259	0.263
	LSSH	0.197	0.208	0.199	0.195
	DBRC	0.253	0.265	0.269	0.288
	RDFH	0.242	0.246	0.244	0.243
	UDCMH	0.309	0.318	0.329	0.346
	DJSRH	0.388	0.403	0.412	0.434
	AGCH	**0.408**	**0.425**	**0.433**	**0.450**
$T \rightarrow I$	CVH	0.252	0.235	0.171	0.154
	IMH	0.467	0.478	0.453	0.456
	LCMH	0.132	0.142	0.154	0.157
	CMFH	0.595	0.601	0.616	0.622
	LSSH	0.569	0.593	0.593	0.595
	DBRC	0.574	0.588	0.598	0.599
	RDFH	0.590	0.596	0.603	0.610
	UDCMH	0.622	0.633	0.645	**0.658**
	DJSRH	0.611	0.635	0.646	**0.658**
	AGCH	**0.627**	**0.640**	**0.648**	**0.658**

TABLE III: The mAP results of all methods on MIRFlickr-25K ($I \rightarrow T$ means the search task of Image-to-Text, and vice versa). The best results are shown in boldface.

Task	Method	16 bits	32 bits	64 bits	128 bits
$I \rightarrow T$	CVH	0.606	0.599	0.596	0.598
	IMH	0.612	0.601	0.592	0.579
	LCMH	0.559	0.569	0.585	0.593
	CMFH	0.621	0.624	0.625	0.627
	LSSH	0.584	0.599	0.602	0.614
	DBRC	0.617	0.619	0.620	0.621
	RDFH	0.632	0.636	0.641	0.652
	UDCMH	0.689	0.698	0.714	0.717
	DJSRH	0.810	0.843	0.862	0.876
	AGCH	**0.865**	**0.887**	**0.892**	**0.912**
$T \rightarrow I$	CVH	0.591	0.583	0.576	0.576
	IMH	0.603	0.595	0.589	0.580
	LCMH	0.561	0.569	0.582	0.582
	CMFH	0.642	0.662	0.676	0.685
	LSSH	0.637	0.659	0.659	0.672
	DBRC	0.618	0.626	0.626	0.628
	RDFH	0.681	0.693	0.698	0.702
	UDCMH	0.692	0.704	0.718	0.733
	DJSRH	0.786	0.822	0.835	0.847
	AGCH	**0.829**	**0.849**	**0.852**	**0.880**

C. Implementation Details

The Image Encoder is a CNN-based network, where any deep CNN network can be utilized to be the backbone of it. In our work, for a fair comparison with other methods, like UDMCH [59], DJSRH [60], we adopt the Alexnet [72] (pre-trained on Imagenet dataset [73]) that consists of five convolution layers and three fully connected layers, to build the Image Encoder. We replace the fc-8 layer (the last layer) of the original Alexnet with a fully connected layer with c hidden units. The Text Encoder is a three-layer Multi-layer Perception (MLP) with the c hidden units ($K \rightarrow 4096 \rightarrow c$, where K is the length of the BoW feature of the text). Each GCN module in our architecture is composed of two graph convolutional layers and one fully connected layer ($4096 \rightarrow 2048 \rightarrow c$). The fusion module is one fully connected layer structure. Except for the last layer of these modules which adopt the $tanh(\cdot)$ activation function, we use $ReLU(\cdot)$ activation functions for all the rest of the layers.

The proposed AGCH method is implemented with Pytorch on a workstation (with Intel XEON E5-2650 v3 @ 2.60GHz CPU, NVIDIA 1080Ti GPU). With regard to the settings for parameters, we fix the batch size to 32, weight decay to 0.0005, and set momentum as 0.9. The learning rates for the Image Encoder, Text Encoder, GCN modules and fusion module are 0.0001, 0.01, 0.001 and 0.01, respectively. The balance parameters in Eq. (8) and Eq. (15) are selected by a validation procedure. Concretely, for three datasets, ρ , δ are set as 4 and $1e^{-2}$. And we set $\alpha = 0.4, \beta = 0.3, \lambda = 5, \gamma_v = 1, \gamma_t = 0.8$ For Wiki. In experiments on MIRFlickr-25K, $\alpha = \beta = 1, \lambda = 10, \gamma_v = 2, \gamma_t = 0.3$, and on NUS-WIDE, $\alpha = \beta = 1, \lambda = 5, \gamma_v = 2, \gamma_t = 0.3$. The iteration time is fixed as 150, 40 and 80 for Wiki, MIRFlickr-25K and NUS-WIDE, separately. With respect to the mAP evaluation, the number of retrieved points is set to 50.

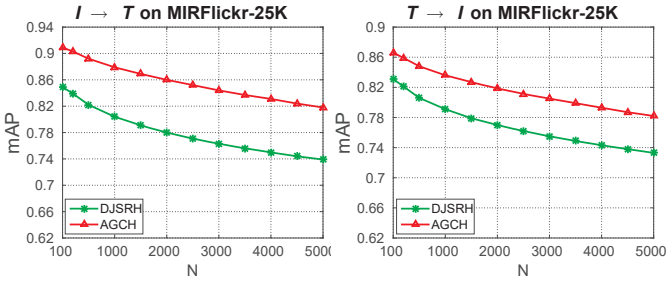


Fig. 2: The mAP curves of DJSRH and AGCH with different number of top returned points on MIRFlickr-25K. The code length is 128.

D. Results and Discussions

1) mAP Results: We compare the proposed AGCH with all baselines in the "Image-to-Text" and "Text-to-Image" search tasks with code length varying from 16 bits to 128 bits and summarize the mAP values in Table II, III and IV. From the results, we can have the following observations.

- AGCH achieves the best performance over all compared shallow and deep methods with various hash code lengths, verifying its effectiveness. In particular, our method achieves up to 4.7 % increase on MIRFlickr-25K. While on NUS-WIDE, it surpasses the second best competitor, i.e., DJSRH, by 5.4 % - 11.7 % in the "Image-to-Text" task and gains an average improvement of 4.2 % in

TABLE IV: The mAP results of all methods on NUS-WIDE ($I \rightarrow T$ means the search task of Image-to-Text, and vice versa). The best results are shown in boldface.

Task	Method	16 bits	32 bits	64 bits	128 bits
$I \rightarrow T$	CVH	0.372	0.362	0.406	0.390
	IMH	0.470	0.473	0.476	0.459
	LCMH	0.354	0.361	0.389	0.383
	CMFH	0.455	0.459	0.465	0.467
	LSSH	0.481	0.489	0.507	0.507
	DBRC	0.424	0.459	0.447	0.447
	RDFH	0.488	0.492	0.494	0.508
	UDCMH	0.511	0.519	0.524	0.558
	DJSRH	0.724	0.773	0.798	0.817
	AGCH	**0.809**	**0.830**	**0.831**	**0.852**
$T \rightarrow I$	CVH	0.401	0.384	0.442	0.432
	IMH	0.478	0.483	0.472	0.462
	LCMH	0.376	0.387	0.408	0.419
	CMFH	0.529	0.577	0.614	0.645
	LSSH	0.577	0.617	0.642	0.663
	DBRC	0.455	0.459	0.468	0.473
	RDFH	0.612	0.641	0.658	0.680
	UDCMH	0.637	0.653	0.695	0.716
	DJSRH	0.712	0.744	0.771	0.789
	AGCH	**0.769**	**0.780**	**0.798**	**0.802**

the "Text-to-Image" task. Furthermore, in the "Image-to-Text" retrieval task on both datasets, the performance of AGCH at 16 bits is even better than those of other methods at 64 bits. The reason is that the proposed method combines more useful information, reliably captures structural properties, and realizes the interplay between different modalities in the training procedure. It is also noticed that our improvements on Wiki are not as significant as those on others, where a potential reason is that fewer training samples are available for Wiki dataset. Nevertheless, our method still performs the best compared to the baseline methods.

- Compared to shallow models, deep methods usually achieve better performance, which reflects that an end-to-end structure successfully facilitate the feature learning and hashing code learning.
- As code length increases, the performances of all methods increase, which implies that longer binary codes can better maintain the semantic information.

We further compare the proposed AGCH with the DJSRH method, which has the second best overall performance in terms of mAP in the foregoing experiments. The method DJSRH learns binary codes by preserving the joint high-order similarity which is constructed by using the original similarity as features. The top-N mAP curves and PR curves are plotted in Figure 2 and 3, respectively. It is observed that our method overpasses DJSRH in all cases, which further verifies the superiority of the proposed hashing learning strategy.

2) Top-N Precision Curves: The top-N precision curves of the cases with 128 bits on three datasets are plotted in Figure 4. The results exhibit that the proposed method outperforms other

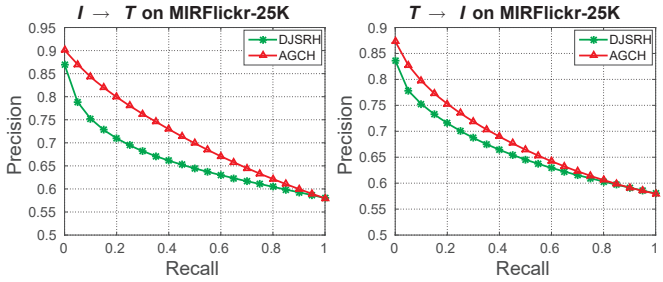
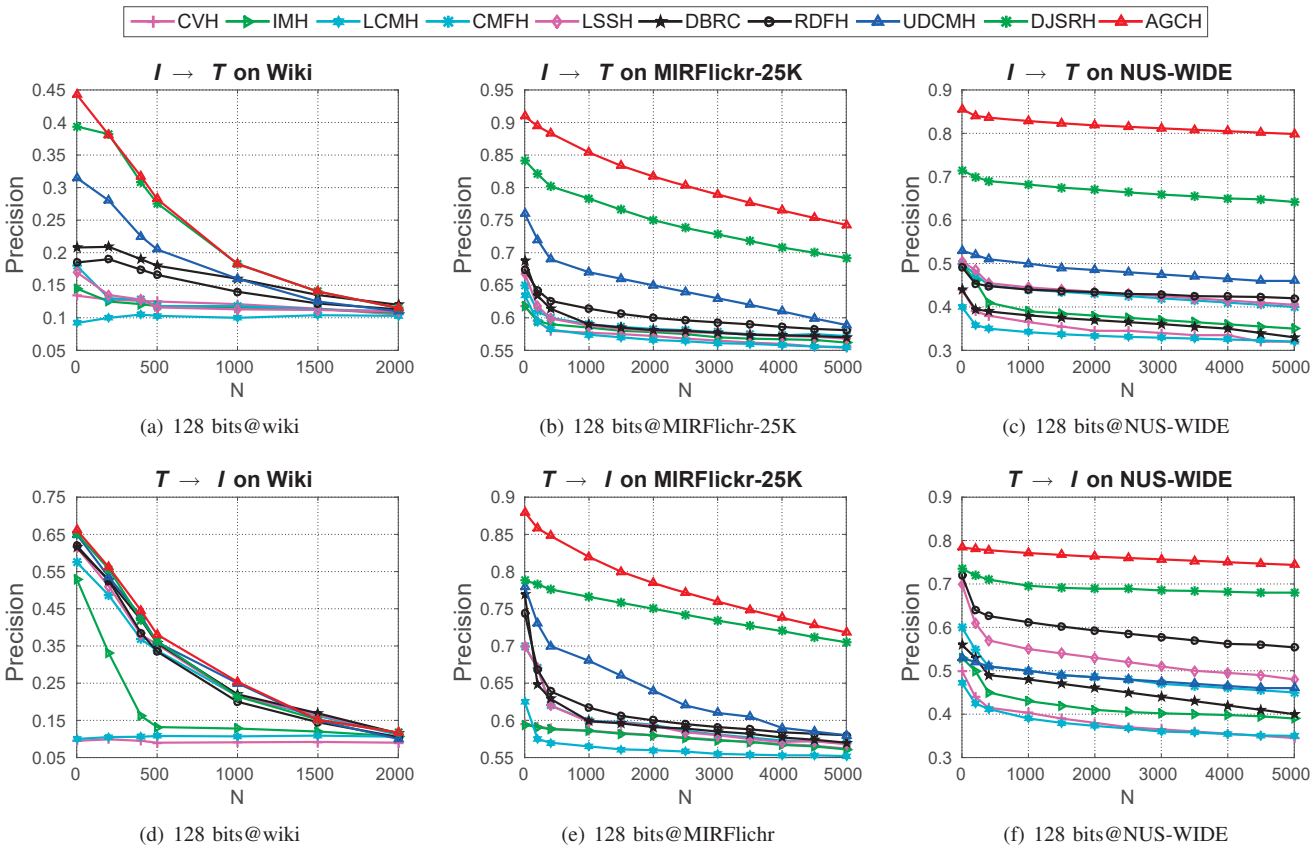


Fig. 3: The Precision-recall curves of DJSRH and AGCH with 128-bit on MIRFlickr-25K.

methodologies by a large margin over a range of retrieved points on MIRFlichr-25K and NUS-WIDE, and on Wiki, it also shows strong competitiveness. It is worth noting that our method can gain higher precision with numerous retrieved points (i.e., 5000) than other approaches with few retrieved points (e.g., 1000) on NUS-WIDE. All these more holistically demonstrate the high representational power of the proposed model on unsupervised cross-modal retrieval tasks.

3) Parameter Sensitivity Analysis: There are several parameters, i.e., α, β, λ, γ_1 and γ_2, may have impacts on the performance of the proposed AGCH. To investigate it, we conduct comprehensive experiments on MIRFlickr-25K by using different values of these parameters. In detail, α, β are tuned in $\{1e^{-4}, 1e^{-3}, 1e^{-2}, 1e^{-1}, 1, 2, 5, 10\}$, λ is tuned in $\{1e^{-3}, 1e^{-2}, 1e^{-1}, 1, 10, 20, 30\}$, these parameters balance the importance of each similarity preserving loss item. γ_1, γ_2 are used to control the weight of each modality in terms of information fusion, tuned in $\{0.1, 0.3, 0.5, 0.7, 1, 1.5, 2\}$. Limited by space, we only report the mAP results with the case of 128 bits which are shown in Figure 5, where the first row is the performance in the "Image-to-Text" task while the second row shows the results in the "Text-to-Image" task. From Figure. 5(a), We can observe that with other parameters fixed, our method is not sensitive to α and β when α do not sit in $\{1e^{-4}, 1e^{-3}, 1e^{-2}, 1e^{-1}\}$ and β is not equal to $1e^{-4}$. Given $\beta = 1$, $\gamma_1 = 2$ and $\gamma_2 = 0.3$, it is also not sensitive to α and λ in the "Image-to-Text" task, while its performance is unstable in the "Image-to-Text" task. Particularly, when $\alpha = 1$ and $\lambda = 10$, it has the best overall performance. With β, γ_1 and γ_2 fixed, the proposed method can achieve higher scores when α and λ are relatively small (see Figure. 5(b)). And while α, γ_1 and γ_2 are fixed at 1, 2 and 0.3, separately, the smaller β and the larger λ are, the better results achieved by AGCH, plotted in Figure. 5(c). With regard to γ_1 and γ_2, we can observe that when the other parameters fixed (i.e., $\alpha = 1$, $\beta = 1$ and $\lambda = 10$), the image modality can provide more important information, as when the value of γ_1 is larger, our method performs better (shown in Figure. 5(d)). Empirically, the optimal hyperparameters would not vary greatly across different datasets for the same task, so that we usually set a range around the optimal parameters for the previous datasets to choose appropriate parameters for new datasets.

4) Ablation Study: Here, we implement comprehensive experiments to analyse the contribution of each component of our



(a) 128 bits@wiki (b) 128 bits@MIRFlichr-25K (c) 128 bits@NUS-WIDE

(d) 128 bits@wiki (e) 128 bits@MIRFlichr (f) 128 bits@NUS-WIDE

Fig. 4: The top-N precision curves of various models with 128-bit on three datasets.

TABLE V: The ablation comparison among different variants of AGCH on MIRFlickr-25K. ($I \rightarrow T$ means the search task of Image-to-Text, and vice versa). The best mAPs for each category are shown in boldface.

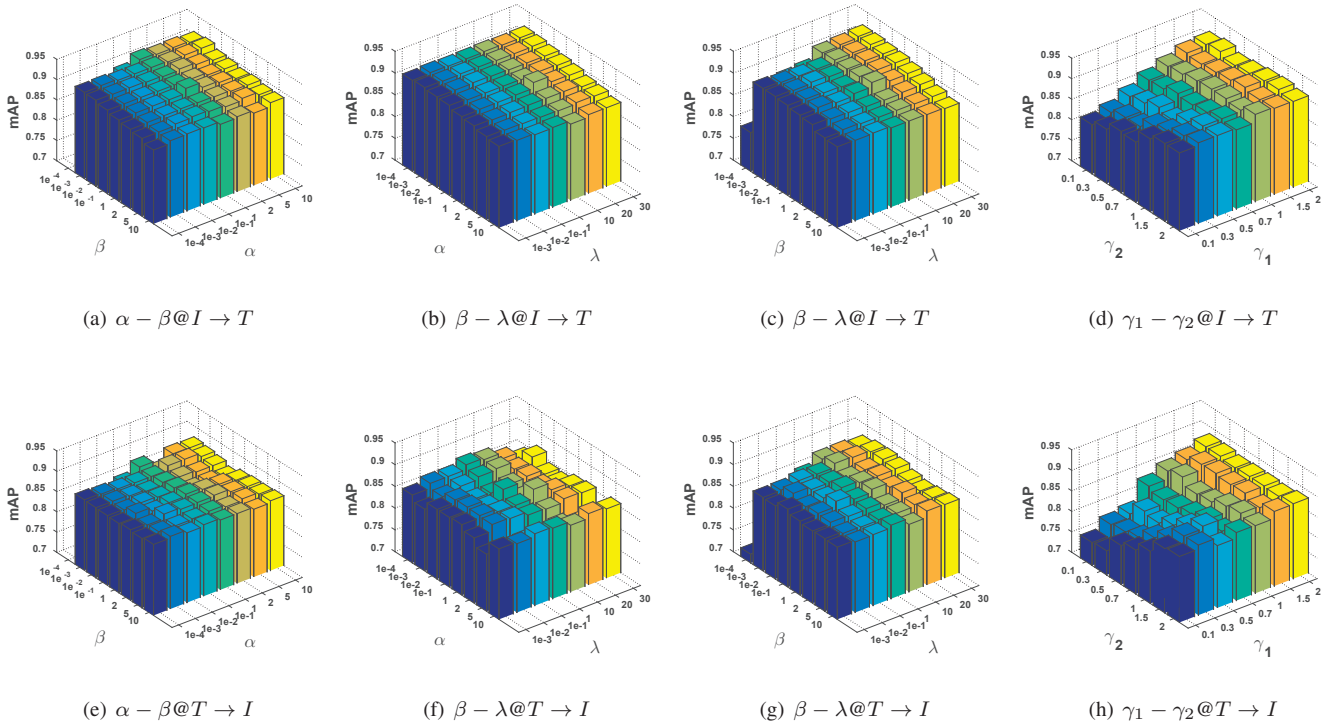
Task	Method	16 bits	32 bits	64 bits	128 bits
$I \rightarrow T$	AGCH	**0.865**	**0.887**	**0.892**	**0.912**
	AGCH-1	0.833	0.869	0.867	0.881
	AGCH-2	0.832	0.840	0.867	0.867
	AGCH-3	0.841	0.872	0.875	0.899
	AGCH-4	0.848	0.868	0.881	0.906
	AGCH-5	0.855	0.877	0.872	0.883
$T \rightarrow I$	AGCH	**0.829**	**0.849**	**0.852**	**0.880**
	AGCH-1	0.812	0.835	0.825	0.853
	AGCH-2	0.826	0.817	0.829	0.826
	AGCH-3	0.785	0.826	0.844	0.859
	AGCH-4	0.815	0.838	0.850	0.852
	AGCH-5	0.819	0.829	0.842	0.875

approach. AGCH has three essential parts, i.e., the similarity capture scheme, the fusion branch and GCN modules. We introduce three variants of the proposed method :

- AGCH-1: We design the first variant 'AGCH-1', which only measures the Cosine similarity for both modalities as the supervision, i.e., $\mathbf{C}$ in Eq. (9), as thefinal similarity matrix.
- AGCH-2: It refers to the variant that only computes the Euclidean distance-based similarity for both modalities as the supervision , i.e., $\mathbf{D}$ in Eq. (12), as the final similarity matrix.
- AGCH-3: As many previous methods [66]–[68] state that the Cosine distance- and Euclidean distance-based metrics are typically used as the similarity measure for texts and images, respectively, we design the first variant 'AGCH-3', which constructs a Cosine similarity matrix $\mathbf{C}$ based on text features and an Euclidean similarity matrix $\mathbf{D}$ based on visual features, and then merges them together, i.e., $\mathbf{S} = \gamma\mathbf{C} + (1-\gamma)\mathbf{D}$, as the final similarity matrix.
- AGCH-4: The fusion branch is removed from AGCH to generate the second variant 'AGCH-4', in order to test the effectiveness of selectively merging information from different modalities.
- AGCH-5: We remove the GCN modules in our framework to test if the aggregation of neighbourhood information can facilitate the more powerful binary representation learning.

We report the mAP results on MIRFlickr-25K with various bit lengths in Table V. From the results, we can have the following observations.

- All three main components collectively make significant contribution to the proposed method AGCH. The overall performance decreases when any of them is removed. Furthermore, it is worth noting that each variant also



(a) $\alpha - \beta @ I \rightarrow T$ (b) $\beta - \lambda @ I \rightarrow T$ (c) $\beta - \lambda @ I \rightarrow T$ (d) $\gamma_1 - \gamma_2 @ I \rightarrow T$

(e) $\alpha - \beta @ T \rightarrow I$ (f) $\beta - \lambda @ T \rightarrow I$ (g) $\beta - \lambda @ T \rightarrow I$ (h) $\gamma_1 - \gamma_2 @ T \rightarrow I$

Fig. 5: The effects of the parameters with 128-bit on MIRFlickr-25K.

performs better than other compared methods in the previous experiments, which indicates the efficacy of these proposed strategies.

- The leverage of different similarity measures indeed boosts the performance of AGCH in both "Image-to-Text" and "Text-to-Image" retrieval tasks. Especially in short length code, without the aggregation-based similarity matrix, the performance decreases dramatically. All these imply that the aggregation-based similarity matrix can carry out precise structural semantics.
- The fusion branch has comparatively less influence in the "Image-to-Text" task while for the "Image-to-Text" task, the lack of fusion operation will limit the upper bound of the performance as it does not have an obvious increase in performance when the bit length increases to 128 bits.
- The GCN modules also have an impact on the performance of the proposed AGCH. Specifically, from the results of 'AGCH-3', we can see that the performance would not increase largely with the code length increases, in most cases, of both tasks.

5) Convergence Analysis: We perform experiments on the three datasets to verify the convergence of the optimization algorithm. The convergence curves of the case with 16 bits code length are plotted in Figure 6. For ease of representation, the objective values are all normalized by dividing the initial loss on each dataset, i.e., when the number of iteration is 1 in the training stage. From this figure, we can see that on MIRFlicker-25K and NUS-WIDE datasets, the objective values of AGCH decrease quickly and reach stable states after forty iterations, while on Wiki, our method converges after about fifty iterations. These demonstrate the efficiency of the proposed optimization scheme.

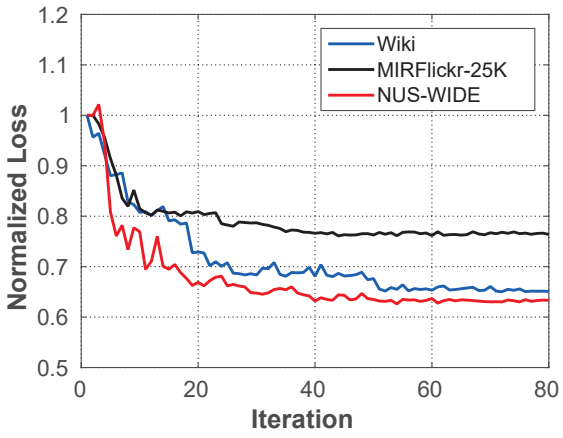


Fig. 6: The convergence curves of AGCH in the case with 16-bit code length on the three datasets.

6) Time Cost Analysis: The comparative training time cost of our proposed method and some deep models is reported on MIRFlickr-25k in Table VI. It can be seen that the training and query time cost of AGCH is acceptable. Especially, even with the code length increasing, the training time of the proposed AGCH does not increase significantly. In terms of both the training time cost and the retrieval accuracy, we draw the conclusion that AGCH is practical for cross-modal searching

TABLE VI: Time cost (in seconds) of various models on MIRFlickr-25k.

Method	Training				Query			
	16 *bits*	32 *bits*	64 *bits*	128 *bits*	16 *bits*	32 *bits*	64 *bits*	128 *bits*
UDCMH	1101.55	1129.36	1157.24	1149.12	58.22	56.48	59.13	59.28
DJSRH	995.58	1029.27	1032.88	1025.11	57.51	59.30	59.96	58.47
AGCH	1122.18	1124.43	1130.68	1139.86	59.46	58.88	59.29	59.58

tasks.

7) Visualization: Some retrieval results are visualized in Fig 7. The query data is listed in the first column and their binary codes generated from different deep models are displayed in the second column. The top-5 retrieved data is shown in the third column with their binary codes. Correct retrievals are marked in green, incorrect in red. It is observed that the proposed method can return plausible searching results.

V. CONCLUSION

In this paper, we propose the Aggregation-based Graph Convolutional Hashing (AGCH) model to deal with unsupervised cross-modal retrieval tasks. AGCH is an end-to-end deep hashing learning framework, based on the Graph Convolutional Networks, which takes full advantage of the complementary and diverse specificities of multimedia data from various modalities. Furthermore, we take a deep look into the correlation between multi-modal data by simultaneously considering their directional relations and the difference in dimensions, thus constructing an affinity matrix that well distinguishes the similarity between paired data. A new GCN-based deep structure is built to enable the intra- and inter- modal consistency preserving, the interaction between different modals and the neighbourhood structure consideration, thus learning more faithful binary embeddings. Extensive experiments on three benchmark datasets showcase that our AGCH method can lead to substantial improvements in unsupervised retrieval tasks.

REFERENCES

[1] B. Kulis and K. Grauman, "Kernelized locality-sensitive hashing for scalable image search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 2130–2137.

[2] C. Strecha, A. Bronstein, M. Bronstein, and P. Fua, "Ldahash: improved matching with smaller descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 1, pp. 66–78, 2011.

[3] L. Nie, S. Yan, M. Wang, R. Hong, and T.-S. Chua, "Harvesting visual concepts for image search with complex queries," in *Proc. ACM Conf. Multimedia Conf.*, 2012, pp. 59–68.

[4] W. W. Ng, X. Tian, W. Pedrycz, X. Wang, and D. S. Yeung, "Incremental hash-bit learning for semantic image retrieval in nonstationary environments," *IEEE Trans. Cybern.*, vol. 49, no. 11, pp. 3844–3858, 2018.

[5] Y. Luo, Y. Yang, F. Shen, Z. Huang, P. Zhou, and H. T. Shen, "Robust discrete code modeling for supervised hashing," *Pattern Recognit.*, vol. 75, pp. 128–135, 2018.

[6] J. Song, Y. Yang, X. Li, Z. Huang, and Y. Yang, "Robust hashing with local models for approximate similarity search," *IEEE Trans. Cybern.*, vol. 44, no. 7, pp. 1225–1236, 2014.

[7] H. T. Shen, S. Jiang, K.-L. Tan, Z. Huang, and X. Zhou, "Speed up interactive image retrieval," *VLDB J.*, vol. 18, no. 1, pp. 329–343, 2009.

[8] Z. Wang, Z. Zhang, Y. Luo, and Z. Huang, "Deep collaborative discrete hashing with semantic-invariant structure."

[9] J. Wang, S. Kumar, and S.-F. Chang, "Semi-supervised hashing for scalable image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 3424–3431.

[10] ——, "Semi-supervised hashing for large-scale search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 12, pp. 2393–2406, 2012.

[11] Y. Yang, Y. Luo, W. Chen, F. Shen, J. Shao, and H. T. Shen, "Zero-shot hashing via transferring supervised knowledge," in *Proc. ACM Conf. Multimedia Conf.*, 2016, pp. 1286–1295.

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

[13] X. Zhou, F. Shen, L. Liu, W. Liu, L. Nie, Y. Yang, and H. T. Shen, "Graph convolutional network hashing," *IEEE Trans. Cybern.*, pp. 1–13, 2018.

[14] M. Lin, R. Ji, S. Chen, X. Sun, and C.-W. Lin, "Similarity-preserving linkage hashing for online image retrieval," *IEEE Trans. Image Process.*, vol. 29, pp. 5289–5300, 2020.

[15] J. Song, Y. Yang, Z. Huang, H. T. Shen, and J. Luo, "Effective multiple feature hashing for large-scale near-duplicate video retrieval," *IEEE Trans. Multim.*, vol. 15, no. 8, pp. 1997–2008, 2013.

[16] Y. Weiss, A. Torralba, and R. Fergus, "Spectral hashing," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, pp. 1753–1760.

[17] W. Liu, J. Wang, S. Kumar, and S.-F. Chang, "Hashing with graphs," in *Proc. Int. Conf. Mach. Learn.*, 2011, pp. 1–8.

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

[19] Y. Mu and Z. Liu, "Deep hashing: a joint approach for image signature learning," in *Proc. AAAI Conf. Artif. Intell.*, 2017, pp. 2380–2386.

[20] H.-F. Yang, K. Lin, and C.-S. Chen, "Supervised learning of semantics-preserving hash via deep convolutional neural networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 2, pp. 437–451, 2017.

[21] Z. Zhang, G.-s. Xie, Y. Li, S. Li, and Z. Huang, "Sadih: semantic-aware discrete hashing," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 5853–5860.

[22] X. Zhu, Z. Huang, H. T. Shen, and X. Zhao, "Linear cross-modal hashing for efficient multimedia search," in *Proc. ACM Conf. Multimedia Conf.*, 2013, pp. 143–152.

[23] N. Srivastava and R. R. Salakhutdinov, "Multimodal learning with deep boltzmann machines," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 2222–2230.

[24] Y. Zhen and D.-Y. Yeung, "Co-regularized hashing for multimodal data," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1376–1384.

[25] D. Wang, P. Cui, M. Ou, and W. Zhu, "Deep multimodal hashing with orthogonal regularization," in *Proc. Int. Joint Conf. Artif. Intell.*, 2015, pp. 2291–2297.

[26] L. Zhu, Z. Huang, X. Liu, X. He, J. Sun, and X. Zhou, "Discrete multimodal hashing with canonical views for robust mobile landmark search," *IEEE Trans. Multim.*, vol. 19, no. 9, pp. 2066–2079, 2017.

[27] X. Zhu, Z. Huang, H. Cheng, J. Cui, and H. T. Shen, "Sparse hashing for fast multimedia search."

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

[29] 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, 2017.

[30] B. Wu, Q. Yang, W.-S. Zheng, Y. Wang, and J. Wang, "Quantized correlation hashing for fast cross-modal search," in *Proc. Int. Joint Conf. Artif. Intell.*, 2015, pp. 3946–3952.

[31] C.-X. Li, Z.-D. Chen, P.-F. Zhang, X. Luo, L. Nie, W. Zhang, and X.-S. Xu, "Scratch: a scalable discrete matrix factorization hashing for cross-modal retrieval," in *Proc. ACM Conf. Multimedia Conf.*, 2018, pp. 1–9.

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

[33] Z. Wang, Z. Zhang, Y. Luo, Z. Huang, and H. T. Shen, "Deep collaborative discrete hashing with semantic-invariant structure construction," *IEEE Trans. Multim.*, 2020.

[34] R. Xia, Y. Pan, H. Lai, C. Liu, and S. Yan, "Supervised hashing for image retrieval via image representation learning," in *Proc. AAAI Conf. Artif. Intell.*, 2014.

[35] V. Erin Liong, J. Lu, G. Wang, P. Moulin, and J. Zhou, "Deep hashing for compact binary codes learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 2475–2483.

[36] W.-J. Li, S. Wang, and W.-C. Kang, "Feature learning based deep supervised hashing with pairwise labels," in *Proc. Int. Joint Conf. Artif. Intell.*, 2016, pp. 1711–1717.

[37] H. Liu, R. Wang, S. Shan, and X. Chen, "Deep supervised hashing for fast image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2064–2072.

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

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

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

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

[42] F. Shen, C. Shen, Q. Shi, A. Van Den Hengel, and Z. Tang, "Inductive hashing on manifolds," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 1562–1569.

[43] X. Wang, Z. Li, and D. Tao, "Subspaces indexing model on grassmann manifold for image search," *IEEE Trans. Image Process.*, vol. 20, no. 9, pp. 2627–2635, 2011.

[44] Q.-Y. Jiang and W.-J. Li, "Scalable graph hashing with feature transformation." in *Proc. Int. Joint Conf. Artif. Intell.*, 2015, pp. 2248–2254.

[45] W. Liu, C. Mu, S. Kumar, and S.-F. Chang, "Discrete graph hashing," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 3419–3427.

[46] R. Xu, C. Li, J. Yan, C. Deng, and X. Liu, "Graph convolutional network hashing for cross-modal retrieval," in *Proc. Int. Joint Conf. Artif. Intell.*, 2019, pp. 10–16.

[47] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907.*, 2016.

[48] S. Kumar and R. Udupa, "Learning hash functions for cross-view similarity search," in *Proc. Int. Joint Conf. Artif. Intell.*, 2011, pp. 1360–1365.

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

[50] D. Zhang and W.-J. Li, "Large-scale supervised multimodal hashing with semantic correlation maximization," in *Proc. AAAI Conf. Artif. Intell.*, 2014, pp. 7–13.

[51] Z. Zhang, Z. Lai, Z. Huang, W. K. Wong, G.-S. Xie, L. Liu, and L. Shao, "Scalable supervised asymmetric hashing with semantic and latent factor embedding," *IEEE Trans. Image Process.*, vol. 28, no. 10, pp. 4803–4818, 2019.

[52] H. T. Shen, L. Liu, Y. Yang, X. Xu, Z. Huang, F. Shen, and R. Hong, "Exploiting subspace relation in semantic labels for cross-modal hashing," *IEEE Trans. Knowl.Data Eng.*, 2020.

[53] D. Wang, Q. Wang, and X. Gao, "Robust and flexible discrete hashing for cross-modal similarity search," *IEEE Trans. Circuits Syst. Video Techn.*, vol. 28, no. 10, pp. 2703–2715, 2017.

[54] Y. Cao, M. Long, J. Wang, Q. Yang, and P. S. Yu, "Deep visual-semantic hashing for cross-modal retrieval," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining.*, 2016, pp. 1445–1454.

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

[56] 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.*, 2017, pp. 1618–1625.

[57] P. Hu, L. Zhen, D. Peng, and P. Liu, "Scalable deep multimodal learning for cross-modal retrieval," in *Proc. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2019, pp. 635–644.

[58] D. Hu, F. Nie, and X. Li, "Deep binary reconstruction for cross-modal hashing," *IEEE Trans. Multimedia.*, vol. 21, no. 4, pp. 973–985, 2018.

[59] 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. Int. Joint Conf. Artif. Intell.*, 2018, pp. 2854–2860.

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

[61] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Trans. Neural Networks.*, vol. 20, no. 1, pp. 61–80, 2008.

[62] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 1024–1034.

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

[64] K. Xu, C. Li, Y. Tian, T. Sonobe, K.-i. Kawarabayashi, and S. Jegelka, "Representation learning on graphs with jumping knowledge networks," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 5453–5462.

[65] F. Shen, Y. Xu, L. Liu, Y. Yang, Z. Huang, and H. T. Shen, "Unsupervised deep hashing with similarity-adaptive and discrete optimization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 12, pp. 3034–3044, 2018.

[66] D. Zhang, J. Wang, D. Cai, and J. Lu, "Self-taught hashing for fast similarity search," in *Proc. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2010, pp. 18–25.

[67] Y. Gong, S. Kumar, V. Verma, and S. Lazebnik, "Angular quantization-based binary codes for fast similarity search," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1196–1204.

[68] 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, 2018.

[69] N. Rasiwasia, J. Costa Pereira, E. Coviello, G. Doyle, G. R. Lanckriet, R. Levy, and N. Vasconcelos, "A new approach to cross-modal multimedia retrieval," in *Proc. ACM Conf. Multimedia Conf.*, 2010, pp. 251–260.

[70] M. J. Huiskes and M. S. Lew, "The mir flickr retrieval evaluation," in *Proc. ACM Int. Conf. Multimedia Retr.*, 2008, pp. 39–43.

[71] 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. Multimedia Retr.*, 2009, p. 48.

[72] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[73] 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.*, 2009, pp. 248–255.

Peng-Fei Zhang received his B.Sc. and M.S. degrees from Shandong University, China, in 2015 and 2018, respectively. He is currently a PhD candidate at the School of Information Technology and Electrical Engineering, University of Queensland.. His research interests include machine learning, information retrieval, privacy protection and multimedia analysis and search.

Yang Li received his B.Sc. degree from Zhejiang Sci-Tech University in 2016 and his M.S. degree of Computer Science from the University of Queensland, Australia in 2018. He is currently a PhD candidate at the School of Information Technology and Electrical Engineering, University of Queensland. His research interests include graph neural network, representation learning, sequential recommendation and social network mining.

Zi Huang received the B.Sc. degree from Tsinghua University, Beijing, China, and the Ph.D. degree in computer science from the University of Queensland, Brisbane, QLD, Australia. She is currently an ARC Future Fellow with the School of Information Technology and Electrical Engineering, University of Queensland. Most of her publications have been published in leading conferences and journals, including ACM Multimedia, ACM SIGMOD, IEEE ICDE, the IEEE TRANSACTIONS ON MULTIMEDIA, the IEEE TRANSACTIONS ON KNOWLEDGE AND DATA, the ACM Transactions on Information Systems, and ACM Computing Surveys. Her current research interests include multimedia search, social media analysis, database, and information retrieval.

Xin-Shun Xu is currently a professor with the School of Software, Shandong University. He received his M.S. and Ph.D. degrees in computer science from Shandong University, China, in 2002, and Toyama University, Japan, in 2005, respectively. He joined the School of Computer Science and Technology at Shandong University as an associate professor in 2005, and joined the LAMDA group of Nanjing University, China, as a postdoctoral fellow in 2009. From 2010 to 2017, he was a professor at the School of Computer Science and Technology, Shandong University. He is the founder and the leader of MIMA (Machine Intelligence and Media Analysis) group of Shandong University. His research interests include machine learning, information retrieval, data mining and image/video analysis and retrieval. He has published in TIP, TKDE, TMM, TCSVT, AAAI, CIKM, IJCAI, MM, SIGIR, WWW and other venues. He also serves as a program committee member or a reviewer for various international conferences and journals, e.g., AAAI, CIKM, CVPR, IJCAI, MM, TCSVT, TIP, TKDE, and TMM.

Fig. 7: Examples of retrieval results on MIRFlickr-25k. First column: query data, second column: binary codes, third column: top-5 retrieved data (first row) and their binary codes (second row).