

# Text-Enhanced Graph Attention Hashing for Cross-Modal Retrieval

Qiang Zou , Shuli Cheng \*, Anyu Du  and Jiayi Chen

College of Computer Science and Technology, Xinjiang University, Urumqi 830046, China; zouq@stu.xju.edu.cn (Q.Z.); anydxju@xju.edu.cn (A.D.); 107552203982@stu.xju.edu.cn (J.C.)

\* Correspondence: cslxju@xju.edu.cn

**Abstract:** Deep hashing technology, known for its low-cost storage and rapid retrieval, has become a focal point in cross-modal retrieval research as multimodal data continue to grow. However, existing supervised methods often overlook noisy labels and multiscale features in different modal datasets, leading to higher information entropy in the generated hash codes and features, which reduces retrieval performance. The variation in text annotation information across datasets further increases the information entropy during text feature extraction, resulting in suboptimal outcomes. Consequently, reducing the information entropy in text feature extraction, supplementing text feature information, and enhancing the retrieval efficiency of large-scale media data are critical challenges in cross-modal retrieval research. To tackle these, this paper introduces the Text-Enhanced Graph Attention Hashing for Cross-Modal Retrieval (TEGAH) framework. TEGAH incorporates a deep text feature extraction network and a multiscale label region fusion network to minimize information entropy and optimize feature extraction. Additionally, a Graph-Attention-based modal feature fusion network is designed to efficiently integrate multimodal information, enhance the affinity of the network for different modes, and retain more semantic information. Extensive experiments on three multilabel datasets demonstrate that the TEGAH framework significantly outperforms state-of-the-art cross-modal hashing methods.

**Keywords:** cross-modal hashing; graph attention; feature fusion; vision transformer; information entropy



**Citation:** Zou, Q.; Cheng, S.; Du, A.; Chen, J. Text-Enhanced Graph Attention Hashing for Cross-Modal Retrieval. *Entropy* **2024**, *26*, 911. <https://doi.org/10.3390/e26110911>

Academic Editor: Geert Verdoolaege

Received: 20 August 2024

Revised: 19 October 2024

Accepted: 25 October 2024

Published: 27 October 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

In the era of information explosion, data take on various forms, including text, images, and videos, across different modalities. These diverse modalities of data have accumulated massive information resources in areas such as the internet, multimedia retrieval, and social networks. However, effectively organizing, retrieving, and understanding data from different modalities pose a significant challenge in the field of information retrieval [1]. Cross-modal retrieval aims to address this issue by establishing semantic links between data of different modalities, enabling users to search for one modality of data (such as text) with another (such as images) [2]. To achieve efficient cross-modal retrieval, hashing methods are widely applied in the indexing and retrieval processes of cross-modal data [3,4]. Cross-modal hash retrieval learns a common hash function to map data from different modalities into the same hash space, ensuring that even data from different modalities can be mapped to similar hash codes as long as they share semantic content. This method effectively bridges the semantic gap between different modalities, realizing fast and accurate cross-modal data retrieval [5–8].

The development of Convolutional Neural Networks (CNN) in recent years has significantly enhanced the performance of cross-modal hash retrieval. CNNs, with their powerful nonlinearity and translation invariance, enable the extraction of higher-quality features from different modalities [9,10]. However, the semantic gap between different modalities is an inherent issue. Images contain more semantic features than text and usually offer richer high-level features. Despite this, most CNN-based cross-modal hashing works [11–15]

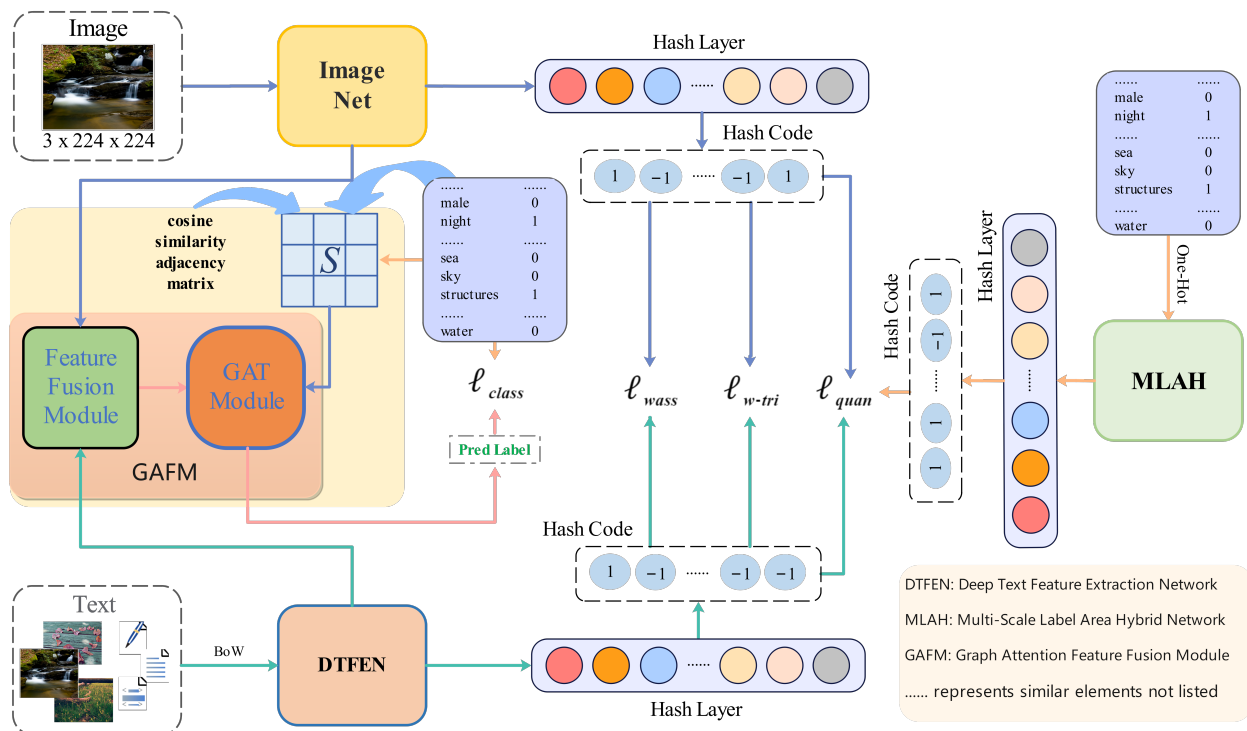
employ the same loss function for feature learning across modalities. Such approaches, however, overlook the differences in information between modalities and do not adequately address the issue of semantic discrepancies [16–22]. Therefore, reducing this semantic gap to enhance the representation of weaker modal features becomes particularly crucial.

Current research is delving deeper into methods for cross-modal hash learning, yet traditional approaches in this field still face numerous challenges in both theory and application. Methods cited in references [23–25] aim to exploit the high-order semantic associations among data's multilabels and utilizing label information to extract implicit semantic content. However, these methods overlook the prior knowledge contained in label information, specifically the weight information of labels. This oversight results in an inability to effectively enhance retrieval performance. Methods referenced in [26–29] employ the fusion of multiscale features from different modalities for semantic optimization, make the more compact between the hash codes. Additionally, they align similar semantic features across modalities using loss functions that incorporate semantic optimization, which improves retrieval performance. However, these approaches do not optimize the cross-modal retrieval process from a textual standpoint, offering limited consideration for text features and thereby failing to effectively integrate features from different modalities. Approaches cited in [30–33] utilize unsupervised clustering to optimize the extraction of intrinsic similarity structures between modalities, enhancing their integration through semantic alignment. However, these methods ignore the structural discrepancies between textual and visual semantics, struggling to effectively align and merge semantic features from different modalities.

With the widespread application of Transformers [34] in the visual domain, an increasing number of cross-modal hash retrieval methods have started to incorporate Transformer models, such as Swin-Transformer [35]. The continuous development of these Transformer-based variants has made Transformer models more suitable for image tasks. Consequently, leveraging the powerful performance of Transformers for retrieval tasks has significantly enhanced cross-modal retrieval performance [36–39]. However, these cross-modal hash retrieval methods, based on both convolutional neural networks and Transformers, have not addressed the issue of textual semantic information scarcity and the severe imbalance between textual data and visual data. Merely optimizing from the image perspective struggles to rectify the inherent issues within text.

Addressing the aforementioned issues, this paper introduces a Text-Enhanced Graph Attention Hashing for Cross-Modal Retrieval (TEGAH) framework. This framework employs Graph Attention (GAT) [40] combined with a multiscale approach for modal feature fusion, effectively mitigating semantic loss and discrepancies caused by the fusion of different modal information. Additionally, it leverages label information to supplement and enhance textual information. As far as we know, GAT is not used in cross-modal hash retrieval to realize the relevant application of feature fusion, but GAT or GCN is more used for feature extraction or classification [41,42]. This paper is the first to introduce the GAT into cross-modal hash retrieval for cross-modal feature fusion, effectively generating pseudo-labels that incorporate features from different modalities, thereby achieving efficient cross-modal hash retrieval. As illustrated in Figure 1, the overall architecture of this paper comprises four parts: an image network, a text network, a label network, and a feature fusion network, collectively referred to as the cross-modal feature fusion learning network. The image network utilizes a Transformer architecture image encoder to model long-distance visual dependencies of images and capture their global information. In the text network, two deep feature extraction modules and an autoencoder are designed for text feature learning, with each text being transformed into a Bag-of-Words (BoW) vector. To alleviate the scarcity of textual information, a novel Multiscale Label Area Hybrid Network (MLAH) is proposed, focusing on supplementing text with multilabel information. This network builds attention graphs for sparse labels and performs multiscale feature extraction to capture their global dependencies, enriching text feature information and optimizing the semantic feature extraction effect of the text network. Finally, the

graph attention feature fusion module (GAFM) proposed in this paper deeply merges and aligns the acquired image semantic features and text semantic features, which can dynamically adjust the similar structure between different modal nodes and generate better prediction labels to supplement the whole network. This framework effectively reduces the semantic discrepancies between different modalities, to some extent enhancing the feature representation capability of different modalities for better semantic alignment and fusion, resulting in higher quality binary hash codes.



**Figure 1.** The overall framework of TEGAH can be divided into five parts: (1) Image-Net: employing the Swin Transformer-Small (SwinT-S) model to extract semantic features from images and map these features into the feature space; (2) Graph Attention Feature Fusion Module (GAFM): a feature fusion and alignment network that weights and merges image and text features to address semantic discrepancies between different modalities; (3) Multiscale Label Area Hybrid Network (MLAH): utilizing multiscale features across four layers and incorporating multiscale attention to mitigate issues related to insufficient textual information; (4) Deep Text Feature Extraction Network (DTFFEN): improving upon traditional methods by capturing high-quality textual feature information; (5) Hash Learning Module: transforming features into hash codes through nonlinear changes, with training assisted by a combination of cosine-weighted triplet loss, label distillation loss, Wasserstein loss, and quantization loss, each component specifically designed to enhance the extraction, fusion, and representation of multimodal features, thereby improving the accuracy and efficiency of cross-modal hash retrieval.

The main contributions of our work are as follows:

- The Text-Enhanced Graph Attention Hashing for Cross-Modal Retrieval (TEGAH) framework proposed in this paper marks the first instance of integrating graph attention to achieve cross-modal feature fusion, and a Graph Attention Feature Fusion Module (GAFM) was designed for deep fusion between different modal features to solve the semantic divergence problem in cross modal hash retrieval. It facilitates better semantic alignment and feature fusion between different modalities and, to some extent, compensates for semantic losses incurred during the fusion process, thereby enhancing the performance of the network.

- The paper introduces a Multiscale Label Area Hybrid Network (MLAH), which mitigates the issue of sparse label distribution by drawing closer the sparsely distributed label information and fully exploring their interconnections. This approach reduces the model's misinterpretation of the semantic relevance of different labels caused by sparse label distribution. Additionally, by fully extracting multilabel features at different scales, MLAH adds multigranularity feature representations to textual features, thereby enhancing the expressive capability of textual features.
- The paper proposes a Deep Text Feature Extraction Network (DTFEN) that modifies the text network by incorporating deep feature extraction modules and an auto-encoder, as opposed to the conventional use of fully connected layers for extracting text information. By employing a deep feature extraction module prior to the hash function, it more effectively integrates deep textual features, thereby improving the utilization rate of text features.

The remainder of this paper is organized as follows. Section 2 provides an overview of work related to cross-modal hashing methods. Section 3 details our Text-Enhanced Graph Attention Transformer for Hash-based Cross-Modal Retrieval (TEGAH) method. Section 4 presents our experimental results and analysis. Finally, Section 6 offers our conclusions.

## 2. Related Work

Hashing methods have garnered widespread attention in retrieval scenarios. Cross-modal hashing methods learn hash functions to map high-dimensional information from diverse modalities into a unified common space, minimizing the semantic gaps between modalities. This ensures that the Hamming distance between semantically related data is smaller than that between semantically unrelated data. In this section, we briefly review supervised and unsupervised cross-modal hashing methods, as well as the work involving Transformer and GAT in cross-modal hash retrieval.

### 2.1. Supervised Cross-Modal Hashing

Supervised cross-modal hash methods enable efficient and accurate retrieval by utilizing label information or inter-modal pairing information to map data from different modalities into a shared hash space. The essence of these methods lies in effectively leveraging available supervisory information to guide the hash encoding process, ensuring the semantic consistency across modalities is maintained. DCMH [5] is a prime example, utilizing AlexNet [9] as the foundation for feature extraction and combining it with fully connected layers for learning features and hash codes, facilitating effective retrieval between images and texts. DVSH [1] seeks to enhance semantic matching across modalities by integrating the textual semantics of images. PRDH [28] introduces inter-modal and decorrelation losses to optimize the similar structure across modalities. CMHH [6] employs a Bayesian method for joint optimization, finely tuning the losses in the quantization process. HSCH [29] explores fine-grained data information to avoid semantic conflicts and preserve important similarity features. DJSAH [26] ensures high-level discriminative semantics are preserved in the hash codes through semantic alignment and latent representations in a shared latent space. SSCH [30] learns hash representations for various data through an alignment-free pseudo-label process and label enhancement strategy. MAFH [24] adopts a collective matrix decomposition method to map kernelized features of different modalities to a shared latent space, optimizing hash code length through semantic labels for bit scalability. DAPH [18], GCDH [42], and MIAN [19] propose their optimization strategies, such as novel hash loss, GCN, and a probabilistic modality alignment framework, refining features and optimizing cross-modal hash retrieval performance from different perspectives. SCCGDH [20] and MESDCH [25] further enhance the robustness and discriminability of hash encoding through category center hash functions and multilabel modality-enhanced attention modules.



## 2.2. Unsupervised Cross-Modal Hashing

Unsupervised cross-modal hashing methods do not utilize real labels but instead learn hash functions by discovering the inherent similarities within the data. For example, UCMH [39] enhances retrieval performance by optimizing a novel hash-similarity friendly loss. It initially trains a Modality Interaction Enabled (MIE) similarity generator to produce a superior MIE similarity matrix for the training set. Then, it uses the generated MIE similarity matrix to guide the training of a deep hash network, introducing a novel bit selection module that interacts between continuous codes of different modalities to generate high-quality unified binary codes for the quantization loss, thereby further improving retrieval performance. DAEH [32] designs an Adaptive Teacher-Guided Enhancement (ATGE) optimization strategy, utilizing information theory to identify weaker hash functions. UKD [33] introduces a new cross-modal hash distillation method, allowing supervised methods to be guided by the outputs produced by unsupervised methods. UCCH [22] incorporates contrastive learning into cross-modal hash retrieval, introducing a novel momentum optimizer that enables the binary hash function to learn, thus bridging the gap between contrastive learning and hashing algorithms. To overcome the False Negative Pair (FNP) challenge, UCCH proposes a Cross-Modal Ranking Learning Loss (CRL), leveraging all pairs instead of hard negative pairs for better performance and robustness.

## 2.3. Transformer-Based Cross-Modal Hashing

With the widespread application of Transformers in both vision and text, Transformer-based cross-modal hashing employs Transformers to learn the intrinsic similarities between images and texts for hash function learning. For example, DCHMT [36] introduces a selection mechanism to generate hash codes, transforming the discrete space into a continuous one. Hash codes are encoded as a series of 2D vectors. UCMFH [37] is the first to explore the effectiveness of the CLIP [43] model in cross-modal hash retrieval, proposing a simple yet powerful baseline model. It utilizes the CLIP model to extract textual and visual features, then generates hash codes through contrastive learning and multimodal fusion. However, it employs a simple weighted averaging method, not fully considering the semantic alignment and complementarity between text and images. DSPH [23] proposes a novel semantic-aware proxy loss for training a MIE similarity generator, creating a superior MIE similarity matrix for the training set. It then uses this matrix as guidance to train a deep hash network, with two Transformer encoders serving as feature extractors for images and texts.

## 2.4. Graphical Attention Network

The Graph Attention (GAT) Network is a graph neural network based on the self-attention mechanism. MS2GAH [41] builds graph features using the adjacency of nodes and allocates varying weights to neighboring edges to bolster the model's resilience. It further employs multilabel annotations to connect the semantic relevance across modalities with greater detail. Introducing the GAT network into cross-modal hash retrieval enables the effective learning of representations for graph-structured data. GAT leverages the structural information of heterogeneous graphs to build image and text data in a unified space, thus capturing the high-level semantic relationships between data more effectively. Through multilayer graph attention networks aggregating neighbor features, the expressive power of each node is enhanced, and different weights can be adaptively allocated to different neighbors. GAT can train the model with adversarial loss and triplet loss, achieving personalized cross-modal retrieval, and enhancing the accuracy and efficiency of retrieval.

## 3. Methodology

In this section, we will look at TEGAH in detail. See Section 3.1 for definitions. The details of the Graph Attention Feature Fusion Module (GAFM) are described in Section 3.2. Multiscale Label Area Hybrid networks (MLFW) and deep text feature extraction networks are described in more detail in Sections 3.3 and 3.4. Section 3.5 shows the details of the

TEGAH model loss function, and in Section 3.6, we detail the various state-of-the-art (SOTA) methods we compared and the parameter settings of the datasets.

### 3.1. Formula Definition

Similar to most other approaches, in this paper, the cross-modal hash retrieval framework utilizes image-text-label triples as input, assuming there are  $N$  pairs of image-text data and their labels. Typically, these triples can be represented as  $\{(X_n, Y_n, Z_n)\}_{n=1}^N$ , where  $X$  denotes images,  $Y$  denotes texts, and  $Z$  represents labels. This paper uses sets to represent these three types of information, i.e., the image set  $X = \{x_1, x_2, \dots, x_N\}$ , the text set  $Y = \{y_1, y_2, \dots, y_N\}$ , and the label set  $Z = \{z_1, z_2, \dots, z_N\}$ , where  $Z_n \in \{0, 1\}^C$  represents a One-Hot encoding,  $C$  stands for the number of label categories, and  $z_n = 1$  if the image and text samples belong to this class; otherwise,  $z_n = 0$ . The aforementioned information can be defined as a set of triples  $S = \{(x_j, y_j, z_j) | x_j \in X, y_j \in Y, z_j \in Z\}$ , which contains all triples of images, texts, and labels. The features extracted through the image network, text network, label network, and cross-modal feature fusion network are denoted as  $F_i^k \in \mathbb{R}^k, i \in \{x, y, z\}$ , mapping different modal features to the same  $k$ -dimensional feature space through the mentioned networks. The overall architecture of our proposed TEGA framework is shown in Figure 1. The hash codes are represented as  $b_x \in \{-1, 1\}^M, b_y \in \{-1, 1\}^M, \text{ and } b_z \in \{-1, 1\}^M$ , where  $M$  denotes the length of the hash code, using  $H(\cdot)$  to denote the hash function, and  $Tanh$  function to map different modal features to the corresponding hash code length. The Tanh function can be represented as follows:

$$\text{Tanh}(t) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (1)$$

The sign function is used to generate the corresponding binary hash codes, typically denoted as  $\text{sign}$ . Its definition is as follows:

$$\text{sign}(t) = \begin{cases} 1, & \text{if } t \geq 0 \\ -1, & \text{if } t < 0 \end{cases} \quad (2)$$

### 3.2. Graph Attention Feature Fusion Module (GAFM)

The Graph Attention Network (GAT) is a graph neural network based on the self-attention mechanism, enabling the utilization of structural information from heterogeneous graphs. By constructing image and text data within a unified space, GAT can effectively capture high-level semantic relationships between data. Through the aggregation of neighbor features by multilayer graph attention networks, the expressive capability of each node is enhanced, allowing for the adaptive allocation of different weights to different neighbors. Moreover, GAT leverages the self-attention mechanism to adaptively weight different modal data, generating more accurate pseudo-labels. This assists in addressing the issue of insufficient cross-modal data labeling, thereby improving the model's generalization ability. In summary, GAT is an effective method for integrating image and textual features, enhancing the performance of cross-modal hash retrieval.

Some existing methods utilize GCN as a feature extractor to extract features from different modalities, which can degrade retrieval performance. In contrast, by weighting the features from different modalities through fusion, the generated pseudo-labels can, to some extent, compensate for the lack of richness in textual data, serving as a supplement to textual information.

In the method proposed in this paper, we have designed a Graph Attention Feature Fusion Module (GAFM). Unlike the original GAT network, which requires the additional generation of co-occurrence matrix information, our approach repurposes the adjacency matrix as the co-occurrence matrix through cosine quantization and weighting operations. To a certain extent, we can consider the label information as a type of weight matrix. By reusing the label matrix and adjacency matrix, feature fusion can optimize the retrieval process beyond just the training phase, enhancing retrieval effectiveness and efficiency. The

structure of our Graph Attention Feature Fusion Module is detailed in Figure 2. The feature representation extracted by the Image Encoder (ImgEncoder) from the image network is denoted as  $\text{ImgEncode}$ , and the hash code obtained from the image network is represented as  $H_x^k$ , where  $M$  denotes the  $M$ -dimensional hamming space, as shown in Equation (1). Finally, we use  $H(\cdot)$  to map the multiscale label fusion features obtained above to the hash code length we need, taking image features as an example:

$$H_x^M = H\left(F_x^k\right) \quad (3)$$

where  $H_x^M$  represents the image hash code,  $M$  represents the length of the hash code, and  $H(\cdot)$  represents the hash function. Assume that the features extracted by the image feature extractor and text feature extractor are  $F_x^k$  and  $F_y^k$ , respectively, and the features extracted by the label network are  $F_z^k$ , the adjacency matrix is defined as  $A \in \mathbb{R}^{C \times C}$ , where  $C$  represents the category of the label number. We use two layers of GAT, and use Concat to combine different modal features. We perform deep feature fusion and optimization through the network as a whole, that is,  $F_c = \text{Concat}\left(F_x^k, F_y^k\right)$ , and  $F_{\text{fusion}}$  denotes the fused features and the overall formula of the modal feature fusion method, which is expressed as follows:

$$F_{\text{fusion}} = \text{LFF}\left(F_c + \text{LRP}_2\left(\text{Attention}_g\right)\right) \quad (4)$$

Among them, the  $\text{LRP}_i$  component is defined as follows:

$$\begin{aligned} \text{LRP}_i &= \text{PReLU}\left(\text{RMSNorm}\left(\text{Linear}(\cdot)\right)\right) \\ & \text{s.t. } i \in \{1, 2, 3, 4\} \end{aligned} \quad (5)$$

We employ the PReLU activation function, Root Mean Square Layer Normalization (RMSNorm), and a Linear mapping function to maintain normalization and dimensional consistency of the fused features, ensuring the stability and generalization capability of the network.

To better fuse coarse-grained features and enhance the representational effect of the fused features, we propose a Local Feature Fusion Module (LFF) composed of Local Kernel Alignment (LKA), convolution, the activation function (ReLU), and Global Max Pooling (GMP). LKA provides multigranularity local fusion features for predicting labels, as illustrated in Figure 2. Additionally, we derive different weighted scores  $\lambda$  and  $h$  from features  $F_c''$  and  $F_c'$  of varying depths for the purpose of weighted fusion, specifically expressed as follows:

$$\begin{aligned} \text{Attention}_g &= \lambda * F_c'' + \hbar * F_c' \\ & \text{s.t. } \lambda = \sigma(F_c''), \hbar = \sigma(F_c') \end{aligned} \quad (6)$$

to better integrate feature representations from different modalities, we employ the Gate Recurrent Unit (GRU) [44] for a deeper level of cross-modal fusion. The specific operations are as follows:

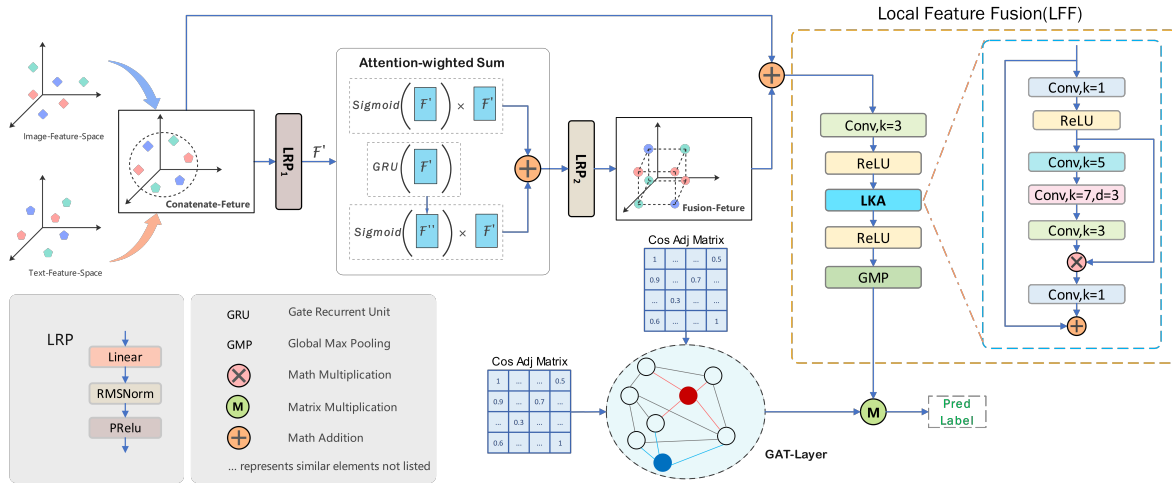
$$\begin{aligned} F_c' &= \text{LRP}_1(F_c) \\ F_c'' &= \text{GRU}\left(\text{LRP}_1(F_c)\right) \end{aligned} \quad (7)$$

the asterisk (\*) represents the multiplication operation and  $\sigma(t)$  denotes the sigmoid function, which is expressed as:

$$\sigma(t) = \frac{1}{1 + e^{-t}} \quad (8)$$

where  $F_c'$  represents the feature vector obtained after the first feature depth fusion extraction layer  $\text{LRP}_1$ , and  $F_c''$  denotes the features extracted after passing through both  $\text{LRP}_1$  and the  $\text{GRU}$ . The  $\text{GRU}$ , widely utilized in Natural Language Processing (NLP), employs a gating mechanism to fuse information from different modalities, while also filtering out dissimilar features and retaining those with semantic relevance. It is evident that, throughout our

feature fusion process, we achieve multilevel and multigranularity deep feature fusion and alignment, taking into account the global and local relevance within the fused features.



**Figure 2.** Graph Attention Feature Fusion Module (GAFM) architecture integrates and aligns image and text features through the interaction of Layerwise Propagation Rule (LRP) and Gated Recurrent Unit (GRU), employing Local Linear Fusion (LLF) to mine multiscale information internally. The features are ultimately fed into the GAT to generate predicted pseudo-labels.

Given the GAT’s capacity for thorough exploration of label information, we utilize it as a classifier to merge with the fused features obtained from the above process to generate pseudo-labels. The weighted sum  $h'_i$  of node  $i$  and its adjacent node features  $j$  is determined by the normalized attention weight coefficients  $e_{ij}$ :

$$h'_i = \sum_{j=1}^N \frac{e_{ij}}{\sum_{k=1}^N e_{ik}} (Wh_j) \tag{9}$$

$$e_{ij} = \text{LeakyReLU}(a^T [Wh_i || Wh_j])$$

where  $h_i$  and  $h_j$  represents the feature vector for node  $i$  and  $j$ ,  $W$  denotes the weight matrix,  $a$  signifies the attention weight vector, which is a parameter that needs to be learned, LeakyReLU refers to a nonlinear activation function, and the concatenation operation of vectors is indicated by  $||$  and  $N$  represents the number of nodes. In GAT, each node  $i$  has a corresponding attentional weight  $e_{ij}$  with every other node  $j$  to adjust the propagation of information. These attention weights are obtained through the linear combination of  $a$  and the node features, followed by normalization to ensure their sum equals 1.

As depicted in Figure 2, the weighted sum and weight parameters of  $G^{l+1}$  are defined as  $\Theta_g$ . The hierarchical propagation rule of GAT can be defined as follows:

$$G^{l+1} = \varphi(e_{ij} A_c G^l W^l) \tag{10}$$

$$\tilde{\Omega} = F_{fusion} * (F_G)^T$$

where  $G$  represents the pseudo-labels obtained through the GAT module and fusion module,  $\varphi$  represents the nonlinear operation,  $\tilde{\Omega}$  signifies the predicted labels,  $A_c$  stands for the adjacency matrix optimized through cosine similarity,  $Z^T$  represents the transpose of label  $Z$ , and  $F_G$  represents the weight set obtained from the adjacency matrix weighting calculation, which is calculated as follows:

$$F_G = G^{l+1}(A_c | \Theta_g)$$

$$A_c = \cos(Z^T \cdot Z, Z \cdot Z^T) = \frac{(Z^T \cdot Z \cdot Z^T \cdot Z^T)}{\|(Z^T \cdot Z)\| \times \|Z^T \cdot Z\|} \tag{11}$$

We calculate the similarity probability score with the final value obtained through the GAFM and the real label. By comparing the similarity, we can balance the differences of different modal features, generate a better hash feature representation, and improve the retrieval effect. Label classification losses are calculated as follows:

$$\mathcal{L}_{cls} = \sum_{i=1}^N \sum_{i=1}^N Z_i \log(\sigma(\tilde{\Omega})) + (1 - Z_i) \log(1 - \sigma(\tilde{\Omega})) \quad (12)$$

Due to the effective integration of label information into the fusion embeddings by GAT, the generated pseudo-labels encompass information from both modalities while preserving the semantic relevance of the original modalities. This ensures that the subsequently generated hash codes are more discriminative.

### 3.3. Multiscale Label Area Hybrid Network (MLAH)

Label information, akin to text, encompasses a wealth of feature information. In cross-modal hash retrieval tasks, labels can serve as a complement to text, compensating for the lack of rich semantic features in text. Unlike the aforementioned method, GAT primarily utilizes labels as a form of supervisory information to guide the transformation of fused features into pseudo-labels, whereas the label network considers labels as a new modality of information. In cross-modal hash retrieval tasks, a single image sample corresponds to multiple different labels, naturally leading us to consider the multiscale information within labels. Moreover, due to the sparsity and diversity of label information distribution, the truly useful information is often nonadjacent. To address these issues, we have designed a multiscale area hybrid module, as shown in Figure 3, to establish connections between nonadjacent areas of label features while incorporating a self-attention mechanism to deepen internal semantic relevance. The overall algorithm is as follows:

$$\begin{aligned} F_z^k &= \text{AutoEncoder}(\text{SoftMax}(\frac{QK^T}{\sqrt{d_k}})V) \\ \text{s.t. } Q &= \text{Concat}(AEMM_1(Z_i), AEMM_2(Z_i)) \\ K &= AEMM_3(Z_i) \\ V &= AEMM_4(Z_i) \end{aligned} \quad (13)$$

where  $H_z^k$  represents the final label features obtained,  $d_k$  denotes the modulation factor, *AutoEncoder* refers the automatic codec, and LKA stands for Local Kernel Alignment, as illustrated in Figure 2. Specifically, to reduce the presence of irrelevant information within tokens and make useful information more compact, we employ the Attention-Enhanced Multiscale Module (AEMM) to aggregate sparse information from the labels and perform multiscale regional blending. The specific algorithm for *AEMM* is as follows:

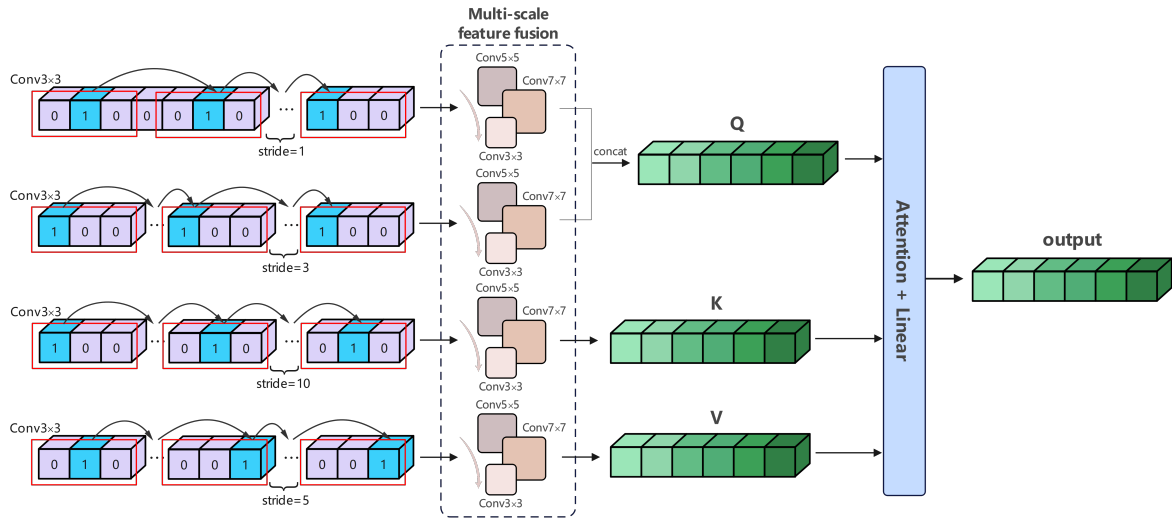
$$\begin{aligned} AEMM_j &= \text{GMP}(\text{ReLU}(\text{LKA}(\text{ReLU}(\text{Conv1D}(Z_i)))))) \\ \text{s.t. } j &\in \{1, 2, 3, 4\} \end{aligned} \quad (14)$$

where  $j$  indicates the use of different strides in one-dimensional convolution,  $Z_i$  represents the use of different label samples, and *GMP* stands for Global Max Pooling.

We assign two smaller strides to  $Q$  through  $AEMM_1$  and  $AEMM_2$ , and two larger strides to  $K$  and  $V$  through  $AEMM_3$  and  $AEMM_4$ , respectively. This results in the acquisition of a query matrix containing compact information, as well as key and value matrices with significant information. The global dependencies of label information are obtained through the self-attention mechanism. Finally, we use  $H(\cdot)$  to map the obtained multiscale label fusion features to the required hash code length:

$$H_z^M = H(F_z^k) \quad (15)$$





**Figure 3.** Multiscale Label Area Hybrid Network (MLAH) consists of a feature extraction module followed by four hierarchical multiscale attention modules, which are ultimately integrated through weighted fusion.

3.4. Deep Text Feature Extraction Network (DTFEN)

Most cross-modal hash retrieval methods convert text into Bag-of-Words (BoW) vectors and then use a multilayer perceptron (MLP) for feature extraction. This approach leads to sparse information characteristics in feature embeddings, which are not conducive to generating compact text hash codes. Therefore, this paper adopts a fine-grained text feature extraction method to replace the traditional MLP or Transformer used in previous methods. Compared to the former, this approach can better learn text features by aggregating more sparse text features. Relative to the latter, our method reduces computational resources, accelerates computation speed, and does not significantly increase the number of parameters compared to the previous MLP. The network structure is shown in Figure 4, and the proposed text network is described as follows:

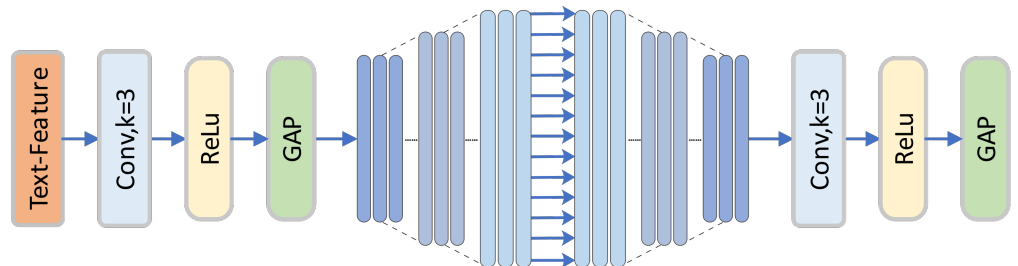
$$F_y^k = Stage_2(AutoEncoder(Stage_1(Y_i))) \tag{16}$$

$$Stage_i = GAP(ReLU(Conv1D_j(\cdot))) \tag{17}$$

*s.t.*  $i, j \in \{1, 2\}$

where  $Conv1D_j$  represents a one-dimensional convolution with a kernel size of  $3 \times 3$ , stride of 1, and padding of 1.  $Stage_i$  indicates different feature extraction stages, and GAP stands for Global Average Pooling. To better blend fine-grained text representations, we utilize a deep module fusion with an autoencoder to obtain the text’s deep mixed features  $F_y^k$ . Finally, we use  $H(\cdot)$  to transform text features into the required hash code length:

$$H_y^M = H(F_y^k) \tag{18}$$



**Figure 4.** Deep Text Feature Extraction Network (DTFEN) comprises two deep extraction modules and an autoencoder.

### 3.5. Hash Learning

The algorithm of TEGAH model is summarized in Algorithm 1. The following four loss functions are used to optimize the backpropagation process of TEGAH model. The four loss functions are described in detail below.

---

#### Algorithm 1 Hash Learning algorithm of TEGAH

---

**Input:**

Training set  $\{X_i, Y_i, Z_i\}_{i=1}^N$ , Binary code length  $M$ , Hyper-parameters  $\partial$ , Query sets  $Query_i$ , Parameters for TEGAH.

**Output:**

Binary code  $B_{i=x,y,z}^M$ , Parameters  $\Theta_X, \Theta_Y$  and  $\Theta_Z$ .

**Initialization:**

Initialize the parameters  $\Theta_X, \Theta_Y$  and  $\Theta_Z$ , maximum iteration number  $epoch$ , mini-batch size 80.

- 1: **while**  $iter < epoch$  **do**
- 2:     Compute  $F_x^k, F_y^k, F_z^k$  features using Equations (3), (15) and (18) for the training set.
- 3:     Compute GAT fusion features using Equation (11).
- 4:     Calculate losses  $\mathcal{L}_{tri}, \mathcal{L}_{class}, \mathcal{L}_{quan}, \mathcal{L}_{wass}$  and  $\mathcal{L}_{kl}$  using Equations (12), (19), (22), (26) and (27).
- 5:     Calculate approximate binary hash codes using query sets data.
- 6:     Input to the trained TEGAH model.
- 7:     Calculate binary hash codes using the function.
- 8: **end while**

**return** the TEGAH model after training.

---

#### 3.5.1. Cosine Weighted Triplet Loss

To maintain similarity among hash codes from different modalities, this paper introduces a cosine-weighted triplet loss mechanism. This approach maps features from various modalities to a binary Hamming space that reflects similar semantic meanings, thereby facilitating efficient similarity measurement and retrieval. The hash function is trained using triplet samples  $\{\tilde{b}_i^x, \tilde{b}_k^{y-}, \tilde{b}_j^{y+}\}$ , each consisting of an anchor and two positive samples derived from both identical and distinct modalities. A sample is defined as positive if it shares at least one label with the anchor; otherwise, it is considered negative. The aim of the cosine-weighted triplet loss during training is to decrease the cosine distance between hash codes of the same modality while increasing the distance between those of different modalities. By adjusting the weights of the weighted terms, the model learns a mapping function that preserves semantic similarity across modalities within the hash space. Additionally, the introduction of normalized weighting factors optimizes structural similarity within the multilabel semantic space. The aforementioned analysis yields the following definition:

$$\mathcal{L}_{tri} = \mathcal{L}_{tri}^{i->t} + \mathcal{L}_{tri}^{t->i} \tag{19}$$

where  $\mathcal{L}_{tri}^{i->t}$  represents the cosine-weighted triplet loss from images to text and  $\mathcal{L}_{tri}^{t->i}$  represents the cosine-weighted triplet loss from text to images, which is similar to  $\mathcal{L}_{tri}^{i->t}$  and will not be elaborated further here. Among them,  $\mathcal{L}_{tri}^{i->t}$  can be defined as follows:

$$\mathcal{L}_{tri}^{i->t} = \sum_{i,j,k} \tau_{jk} \max((\lambda^x_{i,k} - \lambda^x_{i,j} + m, 0) \tag{20}$$

where  $m$  represents the margin coefficient, which adjusts the threshold of similarity for the triplet loss,  $\eta$  denotes the regularization coefficient,  $\tau_{jk}$  represents the weight factor,  $v_j$  and  $v_k$  indicate the similarity between the labels of the positive and negative samples from different modalities with the anchor, computed through cosine similarity, and  $\lambda^x_{i,k}$  and

$\lambda_{i,j}^x$  represent the similarity matrices obtained through cosine similarity calculations. The definition is as follows:

$$\begin{aligned}\tau_{jk} &= \frac{2^{v_j} - 2^{v_k}}{\eta} \\ \lambda_{i,k}^x &= \cos(\tilde{b}_i^x, \tilde{b}_k^{y-}) \\ \lambda_{i,j}^x &= \cos(\tilde{b}_i^x, \tilde{b}_j^{y+})\end{aligned}\quad (21)$$

where  $\tilde{b}_i^x$  and  $\tilde{b}_i^y$  represent the hash codes that have been processed by the tanh activation function but have not yet been binarized. These continuous-valued representations serve as intermediate outputs. In contrast, the hash codes without the tilde symbol (e.g.,  $b_i^x$ ) denote the final binarized codes, which are obtained after applying the sign function.

### 3.5.2. Label Distillation Loss

To optimize the structural semantics of multilabel data, relying solely on cosine-weighted triplet loss is insufficient. It is crucial to maintain consistency in the semantic space between labels and hash codes. Therefore, we employ label distillation loss to preserve the semantic relevance between hash codes and labels. The definition of label distillation loss is as follows:

$$KL_{loss} = \frac{1}{2}(KL_{L,H} + KL_{H,L}) + KL_S \quad (22)$$

where  $KL_{H,L}$  denotes the distillation loss from hash codes to labels,  $KL_{L,H}$  represents the distillation loss from labels to hash codes,  $KL_S$  indicates the mean squared error of similarity. The definitions are as follows:

$$KL_{L,H} = \frac{1}{BN} \sum_{i=1}^B \sum_{j=1}^N \max(0, S_{L,H}) \quad (23)$$

$$KL_{H,L} = \frac{1}{BN} \sum_{i=1}^B \sum_{j=1}^N \max(0, S_{H,L}) \quad (24)$$

$$\begin{aligned}KL_S &= \frac{1}{BN} \sum_{i=1}^B \sum_{j=1}^N (\text{sim}(H_i, L_j) - \text{sim}(L_i, H_j))^2 \\ \text{s.t. } S_{L,H} &= \text{sim}(L_i, H_j) - \text{sim}(H_i, L_j) \\ S_{H,L} &= \text{sim}(H_i, L_j) - \text{sim}(L_i, H_j)\end{aligned}\quad (25)$$

where  $S_{L,H}$  and  $S_{H,L}$ , respectively, represent the similarity matrices from hash codes to labels and from labels to hash codes,  $B$  represents the batch size, and  $N$  denotes the number of samples used for training.

### 3.5.3. Quantization Loss

Quantization loss, through learning a hash function, maps real-valued features to binary hash codes, aiming to preserve data similarity as much as possible.  $b_{*ij}^M$  represents a hash code of length  $M$ ,  $N$  denotes the number of samples to be learned in each batch, and  $x$ ,  $y$ , and  $z$  represent images, text, and labels, respectively. We define  $b_{*ij}^M, i \in B, j \in N, * \in \{H_X^M, H_Y^M, H_Z^M\}$ . We employ the squared L2 norm loss to measure the distance between discrete hash codes and continuous values, training the model by minimizing the distance or discrepancy between real-valued features and their corresponding binary hash codes. By calculating the Hamming distance between binary hash codes, semantically similar

cross-modal data can be found and their similar structure can be maintained. The following definition can be obtained:

$$\begin{aligned}\mathcal{L}_{quan} &= \frac{1}{BN} \sum_{i=1}^B \sum_{j=1}^N (b_{x_{i,j}}^M + b_{y_{i,j}}^M + b_{z_{i,j}}^M) \\ b_x^M &= \|\text{sign}(H_x^M) - H_x^M\|_2^2 \\ b_y^M &= \|\text{sign}(H_y^M) - H_y^M\|_2^2 \\ b_z^M &= \|\text{sign}(H_z^M) - H_z^M\|_2^2\end{aligned}\quad (26)$$

### 3.5.4. Wasserstein Loss

**Wasserstein Loss:** The Wasserstein distance [45], in mathematics, refers to a distance function between probability distributions on a given metric space  $M$ . By incorporating it into the TEGAH framework, it is utilized to balance differences between various modalities, aiming to achieve effective optimization for cross-modal hash retrieval. The definition of Wasserstein loss is as follows:

$$\begin{aligned}\mathcal{L}_{wass} &= EMD(P_i, P_j) \\ &= \inf_{\gamma(x,y) \in \Pi} \sum_{x,y} \|H_x^M - H_y^M\| \gamma(H_x^M, H_y^M) \\ &= \inf_{\gamma(x,y) \in \Pi} \mathbb{E}_{(x,y) \sim \gamma} \|H_x^M - H_y^M\|\end{aligned}\quad (27)$$

where  $P_i$  and  $P_j$  are two probability distributions,  $X$  and  $Y$  are random variables in  $P_i$  and  $P_j$ ,  $\|H_x^M - H_y^M\|$  denotes the distance between the image modality and the text modality hash code, which is measured here using the Euclidean distance, and  $\gamma(\cdot)$  denotes the minimum of all distances. We introduce the Wasserstein distance into cross-modal hash retrieval to better compensate for the differences between modalities. Finally, our proposed TEGAH method uses cosine-weighted ternary loss, label distillation loss, quantization loss, Wasserstein distance loss, and the total Loss can be computed by the following equation:

$$\mathcal{L}_{total} = \alpha(\mathcal{L}_{tri}^{i>t} + \mathcal{L}_{tri}^{t>i}) + \mathcal{L}_{class} + \mathcal{L}_{quan} + \mathcal{L}_{wass} + \mathcal{L}_{kl} \quad (28)$$

where  $\alpha$  is the hyperparameter to balance the cosine-weighted triad loss with other losses, and in our experiments  $\alpha$  is taken to be 10.

### 3.6. Baseline Setting

In our experiments, we selected 14 state-of-the-art cross-modal hash retrieval methods for comparison, including DCMH [5], CMHH [6], AGAH [7], CPAH [8], DADH [14], SC-AHN [17], DCHUC [21], MESDCH [25], SCCGDH [20], MIAN [19], GCDH [42], DAPH [18], MAFH [24], and DSPH [23]. For all methods, we utilized the same experimental setup and maintained consistency in the division of datasets, retrieval sets, and query sets across all approaches, aligning them with our experimental configurations.

## 4. Experiments

To validate the effectiveness of our proposed Text-Enhanced Graph Attention Transformer for Hash-based Cross-Modal Retrieval (TEGAH) method, we carried out comprehensive experiments on three public multimodal retrieval datasets: MIRFLICKR-25K, NUS-WIDE, and MS-COCO. In the following sections, we elaborate on the experimental results of several state-of-the-art algorithms compared to our approach. Furthermore, we provide detailed descriptions of the three datasets used for experimental training, explain the experimental details of TEGAH, evaluate TEGAH's performance metrics, and describe the experimental setup.

#### 4.1. Datasets

In the experiments of this paper, we employ the same sampling strategy across three large-scale multilabel datasets: MIRFLICKR-25K (<https://press.liacs.nl/mirflickr> (accessed on 17 October 2024)) [46], NUS-WIDE (<https://www.kaggle.com/datasets/xinleili/nuswide> (accessed on 17 October 2024)) [47], and MS-COCO (<https://cocodataset.org/> (accessed on 17 October 2024)) [48]. Each dataset is divided into training sets, test sets, and retrieval sets. For different datasets, images and texts are processed in the same manner, with the input network's image resolution set to  $224 \times 224$ . Text is represented using Bag-of-Words (BoW) encoding. Specific details about the division of datasets and the dimensions of text feature encodings are presented in Table 1.

**Table 1.** Characterization statistics for the three benchmark datasets.

Dataset Details	MIRFLICKR-25K	NUS-WIDE	MS-COCO
Dataset Size	20,015	186,577	122,218
Training Size	10,000	10,500	10,000
Retrieval Size	18,015	184,477	117,218
Query Size	2000	2100	5000
Number of Categories	24	10	80
Dim of Text Features	1386	1000	2026

#### 4.2. Evaluation Criteria

In our work, we employ Mean Average Precision (mAP) and the Precision-Recall curve (PR curve) as evaluation metrics for our experiments. These metrics are detailed as follows.

##### 4.2.1. Mean Average Precision (mAP)

mAP is a method used to assess the performance of retrieval systems, measuring the average level of accuracy within the retrieval results. The mAP value represents the average precision, assessing whether the modality retrieved matches the query modality category, commonly used to evaluate the performance of cross-modal retrieval algorithms. Given a set of query data  $Q$  and  $N$  retrieval results, the mean average precision can be expressed as:

$$mAP = \frac{1}{QR} \sum_{q=1}^Q \sum_{i=1}^N P(i) \delta(i) \quad (29)$$

where  $P(i)$  denotes the precision of the top  $i$  retrieval results, and  $\delta(i) = 1$  equals 1 if the retrieval result is relevant to the query, and 0 otherwise, i.e.,  $\delta(i) = 0$ ,  $Q$  represents the number of queries initiated, and  $R$  represents the size of the entire search set.

The mAP serves as a metric to assess the performance of retrieval systems, aiding in the evaluation of the accuracy of retrieval outcomes and the effectiveness of the retrieval system. In our work, we utilize the mAP@all evaluation metric, where "all" refers to the size of the entire retrieval set.

##### 4.2.2. Precision-Recall (PR) Curve

The PR curve represents the precision of the retrieved ranked list at different recall levels.

#### 4.3. Experimental Details

In this paper, we employ a model pre-trained on ImageNet-1K as the backbone network for image processing, extract textual features using a deep text network, and utilize a GAT to optimize cross-modal feature fusion. Additionally, a label network supplements textual semantic features. The input to our framework's GAT consists of two adjacency matrices constructed from label information optimized through cosine similarity. Our



TEGAH framework is implemented in PyTorch version 2.1.0, with Python version 3.10 and CUDA version 12.1. All experiments were conducted on a computer equipped with an NVIDIA RTX-3090 Ti GPU and 128 GB RAM. In our experiments, the learning rate for the image network was set between  $10 \times 10^{-5}$  and  $10 \times 10^{-6}$ , while for the GAT network, text network, and label network, it ranged from  $10 \times 10^{-4}$  to  $10 \times 10^{-5}$ . The batch size was set at 80, and the number of epochs at 300. We optimized the image and GAT networks using AdamW optimizer and the text and label networks using the Adam optimizer.

#### 4.4. Analysis of Experimental Results

##### 4.4.1. Comparison with the Baselines

To assess the effectiveness and advancement of our proposed TEGAH method, we conducted a comparative analysis with 14 state-of-the-art (SOTA) cross-modal hash retrieval methods in terms of mAP values and PR curves. This comparison encompasses two evaluation tasks: using images to retrieve text, denoted as “Image-to-Text” (I2T), and using text to retrieve images, denoted as “Text-to-Image” (T2I). Table 2 present the mAP comparison results for each method across three different datasets with 16, 32, and 64-bit hash codes. Compared to the second-best method GCDH, our TEGAH method shows a maximum performance increase of 1.7% and an average increase of 1.35% on the MIRFLICKR-25K dataset, a maximum increase of 2.3% and an average increase of 0.75% on the NUS-WIDE dataset, and a notable maximum increase of 4.4% and an average increase of 3.8% on the MS-COCO dataset. The significant improvement on the MS-COCO dataset may be attributed to its larger number of labels compared to the other two datasets (MIRFLICKR-25K has 24 category labels, while NUS-WIDE has only 10), as a limited number of category labels can affect the multiscale feature information extracted by the Multiscale Label Area Hybrid Network (MLAH) and lead to sparser features when fusing modal features for the Graph Attention Module, ultimately impacting retrieval performance. Although TEGAH’s performance on the other two datasets (MIRFLICKR-25K and NUS-WIDE) did not reach the level achieved on the MS-COCO dataset, the results indicate that our TEGAH method can still effectively learn multiscale label features and optimize text feature extraction in scenarios of sparse text feature information and limited label category information. By employing the Multiscale Label Area Hybrid Network and the Deep Text Feature Extraction Network, TEGAH can compensate for the scarcity of textual information and category labels. Furthermore, the Graph Attention Feature Fusion Module enables the alignment and fusion of different modal information, utilizing learned implicit information to bridge the information gap between modalities, thereby optimizing the generation of final hash codes and enhancing retrieval performance. Figures 5–7 showcase the Precision-Recall (PR) curves for hash code lengths of 32 and 64 bits. It is observable that, in most instances, the PR curve trends of our proposed TEGAH method outperform those of other methods across the three datasets. On the NUS-WIDE dataset, our method surpasses the second-best method GCDH in T2I performance, but slightly lags behind GCDH in I2T performance. This discrepancy can be attributed to the lesser number of category labels used in the NUS-WIDE dataset, which results in an insufficient number of features for the adjacency matrix required by the MLAH and the GAFM. Consequently, the GAFM cannot fully utilize the feature information from different modalities for alignment and fusion, thereby affecting the generation of hash codes. This highlights the importance of adequate category labels and adjacency matrix features in enhancing the effectiveness of cross-modal feature fusion and alignment, which are critical for generating distinctive and accurate hash codes.

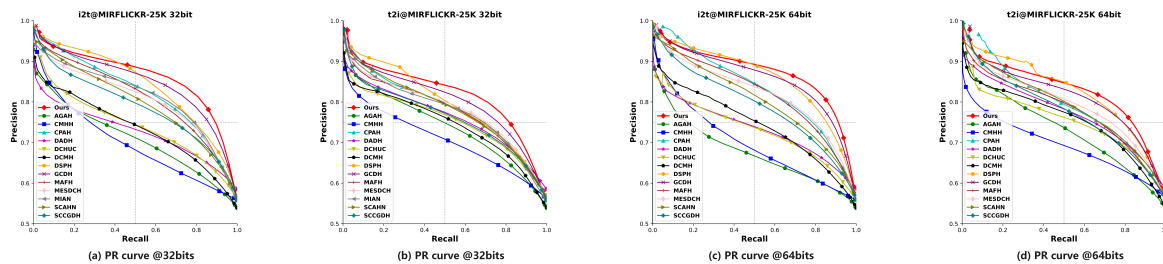


Figure 5. Results of PR curves of 32 bits and 64 bits on MIRFLICKR-25K dataset.

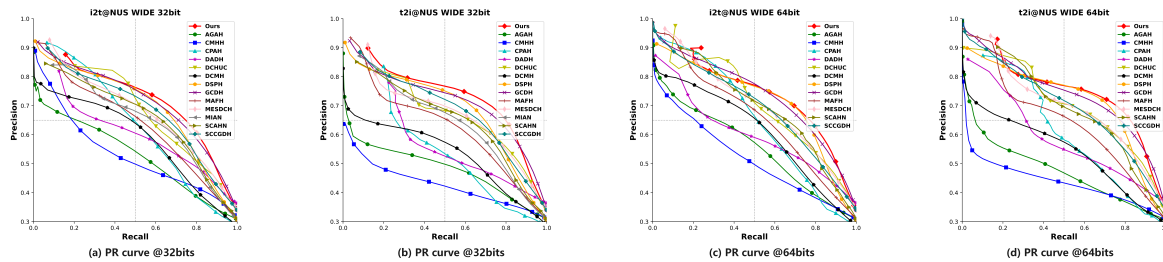


Figure 6. Results of PR curves of 32 bits and 64 bits on NUS-WIDE dataset.

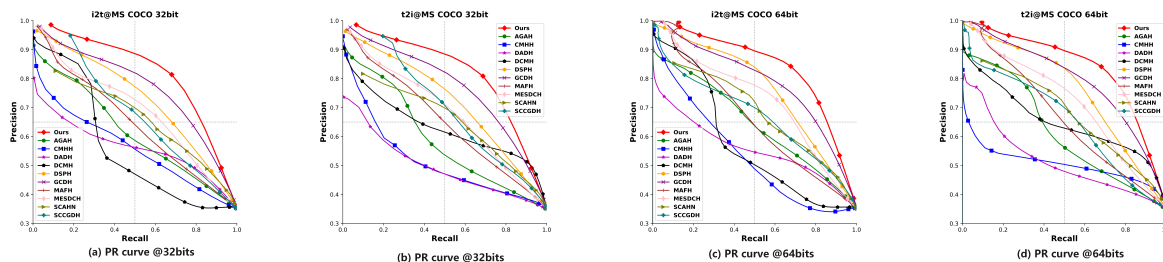


Figure 7. Results of PR curves of 32 bits and 64 bits on MS-COCO dataset.

#### 4.4.2. Ablation Experiments

For cross-modal retrieval tasks, our proposed method performs two evaluation tasks: “I2T” for retrieving text using images and “T2I” for retrieving images using text. In order to validate the effectiveness of our proposed TEGAH method, we conducted extensive experiments on three public datasets with the following details:

Table 3 outlines the design of our ablation experiments, featuring eight variants: (a) ‘baseline’ refers to the base model, where the GAFM, MLAH, and DTFEN modules are removed from the final network, while all other parameter settings are retained. (b) ‘TEGAH’ represents the complete model, incorporating the GAFM, MLAH, and DTFEN modules. (c) ‘TEGAH-V1’ adds only the MLAH module to the baseline model. (d) ‘TEGAH-V2’ introduces only the GAFM module into the baseline model. (e) ‘TEGAH-V3’ includes only the DTFEN module in the baseline model. (f) ‘TEGAH-V4’ integrates both the GAFM and MLAH modules into the baseline model. (g) ‘TEGAH-V5’ integrates both the DTFEN and MLAH modules into the baseline model. (h) ‘TEGAH-V6’ incorporates both the DTFEN and GAFM modules into the baseline model. Table 4 presents the results of ablation studies. The outcomes from experiments TEGAH-V3, TEGAH-V5, and TEGAH-V6 indicate that the incorporation of the DTFEN notably enhances the text feature extraction, particularly yielding better results for hash codes of lower bit lengths. This improvement in text feature extraction concurrently elevates the performance of the image feature extraction network to a certain extent. Furthermore, the results from TEGAH-V1, TEGAH-V4, and TEGAH-V5 demonstrate significant improvements in “Text-to-Image” (T2I) retrieval following the integration of the MLAH. This suggests that treating label information as a modality for multiscale weighted fusion can effectively compensate for the scarcity of textual feature information. Additionally, the introduction of the GAFM, as evidenced by the results

from TEGAH-V2, TEGAH-V4, and TEGAH-V6, leads to enhanced retrieval performance compared to the baseline. This enhancement indicates that GAFM can effectively integrate and align features from different modalities, reinforcing their representation and mitigating information loss in hash codes. Finally, the comprehensive performance improvement observed in the results from TEGAH-V0, where all three modules were utilized, validates the efficacy and rationale of our TEGAH framework.

**Table 2.** The MAP protocols on MIRFLICKR25K, NUS-WIDE, and MS-COCO (MAP@ALL). Results are indicated in bold. ‘/’ denotes unavailable results, and ‘\*’ indicates results cited from the original paper.

Task	Method	MIRFLICKR-25K			NUS-WIDE			MS-COCO		
		16 bits	32 bits	64 bits	16 bits	32 bits	64 bits	16 bits	32 bits	64 bits
I2T	DCMH [5]	0.7323	0.7432	0.7502	0.5248	0.6000	0.6197	0.5179	0.5314	0.5472
	CMHH [6]	0.6863	0.6901	0.6887	0.5233	0.5171	0.5236	0.5530	0.5461	0.4714
	AGAH [7]	0.7006	0.7241	0.6912	0.3945	0.4107	0.4258	0.5501	0.5515	0.5518
	CPAH [8]	0.8063	0.8237	0.8305	0.5686	0.6207	0.6342	0.5949	0.6426	0.6448
	DADH [14]	0.7333	0.7449	0.7496	0.5953	0.6084	0.6030	0.5750	0.5788	0.5755
	SCAHN [17]	0.7828	0.7942	0.8021	0.6550	0.6580	0.6744	0.6479	0.6426	0.6431
	DCHUC [21]	0.7358	0.7464	0.7427	0.6159	0.6460	0.6755	0.5282	0.5489	0.5338
	MESDCH [25]	0.7898	0.8032	0.8153	0.6607	0.6832	0.6968	0.6590	0.6960	0.7212
	SCCGDH [20]	0.7748	0.7949	0.7933	0.6770	0.6931	0.6977	0.6044	0.6351	0.6647
	MIAN [19]	0.8044	0.8178	0.8183	0.6303	0.6433	0.6374	0.5856	0.6121	0.6131
	GCDH [42]	0.8373	0.8545	0.8630	<b>0.7136</b>	<b>0.7263</b>	<b>0.7424</b>	0.7268	0.7630	0.7826
	DAPH* [18]	/	/	/	/	0.6840	0.6930	0.6870	0.7180	/
	MAFH [24]	0.7981	0.8168	0.8263	0.6367	0.6422	0.6582	0.6044	0.6689	0.6871
DSPH [23]	0.8016	0.8301	0.8446	0.6847	0.7015	0.7125	0.6864	0.7493	0.7704	
Ours	<b>0.8484</b>	<b>0.8665</b>	<b>0.8740</b>	0.7052	0.7236	0.7356	<b>0.7542</b>	<b>0.8021</b>	<b>0.8219</b>	
T2I	DCMH [5]	0.7554	0.7716	0.7788	0.5545	0.5903	0.5957	0.5508	0.5883	0.6049
	CMHH [6]	0.6809	0.7134	0.7012	0.4795	0.4541	0.4668	0.4847	0.4980	0.5053
	AGAH [7]	0.6873	0.7496	0.7478	0.4344	0.3980	0.4382	0.5012	0.5146	0.5191
	CPAH [8]	0.7947	0.8064	0.8082	0.5605	0.5686	0.6053	0.5891	0.6384	0.6413
	DADH [14]	0.7641	0.7748	0.7813	0.5631	0.5609	0.5711	0.4767	0.4819	0.4921
	SCAHN [17]	0.7845	0.7956	0.7997	0.6692	0.6715	0.6795	0.6470	0.6430	0.6396
	DCHUC [21]	0.7522	0.7712	0.7708	0.6356	0.6795	0.7019	0.5220	0.5269	0.5185
	MESDCH [25]	0.7741	0.7898	0.7991	0.6662	0.6840	0.6977	0.6345	0.6737	0.7019
	SCCGDH [20]	0.7622	0.7785	0.7903	0.6759	0.7072	0.7115	0.5949	0.6427	0.6475
	MIAN [19]	0.7947	0.8013	0.8082	0.6486	0.6685	0.6586	0.5459	0.5997	0.5940
	GCDH [42]	0.8103	0.8230	0.8319	0.7195	0.7348	0.7474	0.7219	0.7597	0.7845
	DAPH* [18]	/	/	/	/	0.6770	0.6890	0.7030	0.7300	/
	MAFH [24]	0.7841	0.7982	0.8006	0.6357	0.6480	0.6542	0.5963	0.6733	0.6912
DSPH [23]	0.7972	0.8133	0.8351	0.7025	0.7177	0.7315	0.6921	0.7520	0.7714	
Ours	<b>0.8238</b>	<b>0.8406</b>	<b>0.8481</b>	<b>0.7403</b>	<b>0.7578</b>	<b>0.7665</b>	<b>0.7593</b>	<b>0.8044</b>	<b>0.8263</b>	

**Table 3.** Ablation experiment settings for each module. ✓ indicates that the module is used, ✗ indicates that the module is not used.

	MLAH	GAFM	DTFEN
TEGAH	✓	✓	✓
TEGAH-V1	✓	✗	✗
TEGAH-V2	✗	✓	✗
TEGAH-V3	✗	✗	✓
TEGAH-V4	✓	✓	✗
TEGAH-V5	✓	✗	✓
TEGAH-V6	✗	✓	✓

Table 4. Ablation experiment results.

Task	Method	MIRFLICKR-25K			NUS-WIDE			MS-COCO		
		16 bits	32 bits	64 bits	16 bits	32 bits	64 bits	16 bits	32 bits	64 bits
I2T	baseline	0.8219	0.8500	0.8620	0.6868	0.7174	0.7312	0.6816	0.7526	0.7859
	TEGAH-V1	0.8225	0.8506	0.8620	0.6822	0.7176	0.7436	0.6987	0.7502	0.7882
	TEGAH-V2	0.8261	0.8537	0.8696	0.6980	0.7187	<b>0.7442</b>	0.6845	0.7523	0.7913
	TEGAH-V3	0.8441	0.8571	0.8644	<b>0.7053</b>	0.7155	0.7283	0.7287	0.7822	0.8054
	TEGAH-V4	0.8260	0.8513	0.8632	0.6960	0.7219	0.7415	0.6874	0.7536	0.7900
	TEGAH-V5	0.8480	0.8592	0.8643	0.7044	0.7119	0.7268	0.7322	0.7841	0.8063
	TEGAH-V6	0.8476	0.8575	0.8651	0.7036	0.7107	0.7290	0.7346	0.7817	0.8086
	TEGAH	<b>0.8484</b>	<b>0.8665</b>	<b>0.8740</b>	0.7052	<b>0.7236</b>	0.7356	<b>0.7542</b>	<b>0.8021</b>	<b>0.8219</b>
T2I	baseline	0.7794	0.8095	0.8292	0.6844	0.7189	0.7303	0.6804	0.7448	0.7801
	TEGAH-V1	0.7981	0.8156	0.8339	0.6911	0.7237	0.7430	0.6970	0.7471	0.7842
	TEGAH-V2	0.7836	0.8089	0.8262	0.7092	0.7202	0.7418	0.6882	0.7439	0.7860
	TEGAH-V3	0.8108	0.8267	0.8368	0.7254	0.7401	0.7498	0.7227	0.7812	0.8069
	TEGAH-V4	0.7956	0.8110	0.8318	0.7004	0.7288	0.7464	0.6822	0.7475	0.7850
	TEGAH-V5	0.8176	0.8313	0.8410	0.7234	0.7367	0.7519	0.7268	0.7828	0.8094
	TEGAH-V6	0.8158	0.8295	0.8397	0.7278	0.7377	0.7466	0.7312	0.7812	0.8126
	TEGAH	<b>0.8238</b>	<b>0.8406</b>	<b>0.8481</b>	<b>0.7403</b>	<b>0.7578</b>	<b>0.7665</b>	<b>0.7593</b>	<b>0.8044</b>	<b>0.8263</b>

#### 4.4.3. Top-5 Retrieval Outcomes

To showcase the effective retrieval capability of our introduced TEGAH approach, we employed the MS-COCO dataset for Hamming ranking, as illustrated in Figure 8. The retrieval instances obtained through our TEGAH approach are all pertinent. This suggests that the TEGAH approach can markedly improve the performance of the text feature extraction network. Moreover, the Multiscale Label Area Hybrid Network can, to some extent, compensate for the scarcity of textual information. Consequently, through the Graph Attention Feature Fusion Module, it is possible to better integrate the semantic information of multilabels, generating more distinctive hash codes. This ability to accurately retrieve relevant results underscores TEGAH's effectiveness in addressing the challenges of cross-modal hash retrieval, particularly in bridging the semantic gap between different modalities and improving the richness of textual features for more accurate and efficient search outcomes.

#### 4.4.4. Visualization Results

To further validate the capability of the image feature extraction network in capturing global information, Figure 9 presents several examples of feature visualization using our proposed TEGAH method. Across three datasets, we selected 10 images each and visualized their feature maps using the Grad-CAM method, specifically visualizing the outputs before the LayerNorm layer of the last encoder block of the image feature extractor. GradCAM visualizations highlight the regions of the image feature extraction network that may influence the classification decision, which often contain key descriptive elements. For instance, on the MS-COCO and NUS-WIDE datasets, certain images encompass two identical objectives, and the image feature extraction network within the TEGAH method can precisely capture both objectives. This demonstrates TEGAH's effectiveness not only in feature extraction but also in ensuring that the extracted features are meaningful and relevant to the image content, thereby enhancing the accuracy of subsequent retrieval tasks.






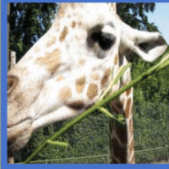
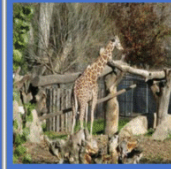
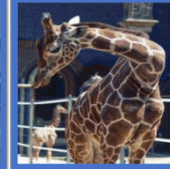

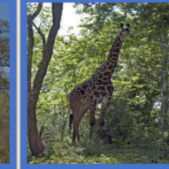








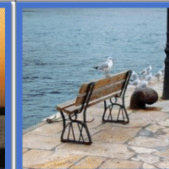

Task	Query Samples	Retrieved Samples				
I2T		A rusty fire hydrant is on a city sidewalk. A rusty, yellow fire hydrant on a sidewalk. A fire hydrant out side a large building outside. A rusted yellow fire hydrant is show at the edge of a sidewalk. The fire hydrant is on the side of the street by the building.	An old fire hydrant on a snowy street. A red and yellow fire hydrant beside a concrete wall next to a street. A fire hydrant is standing on the side of a sandy road as a truck passes. A yellow and red fire hydrant near a wall. A red and yellow fire hydrant sitting on a sidewalk.	An ambulance parked next to a fire hydrant with the driver relaxing. An ambulance that has parked in front of a fire hydrant. An ambulance responds to an emergency and blocks a fire hydrant. An ambulance with an open door in front of a fire hydrant. An ambulance parked next to a yellow fire hydrant.	A fire hydrant in the park is rusted and old. A water hydrant on a field with grass. A fire hydrant sits in the middle of the grass. A red and white fire hydrant on a field of grass. A red and white fire hydrgn in a grassy field.	A brown goat standing next to an orange fire hydrant. A deer standing next to a fire hydrant on a sidewalk. A deer like animal standing on a sidewalk next to a fire hydrant. Small goat standing next to fire hydrant in zoo like area. A small deer like animal standing next to a fire hydrant.
		Three bears swimming in a rushing river surrounded by greenery. Three bears are in a fast moving river. Some animals that are in the water together. Three bears swimming in the water near a waterfall. Three bears in the rapids of a river.	close up of a bear surrounded by different types of bushes. A bear walking in the bushes and plants in the wild. A bear is walking through some plants and brush. A Black bear semi concealed by smaller plants and bushes. A black bear stares out while in the woods.	A bear walks through tall grass near several trees. A black bear is walking on an open field next to some trees. there is a black bear that can be seen walking own a hill. A black bear cub walking down a small hillside. A black bear walks down a hill away from a tree.	A large brown bear wading through a pool of water. a close up of a black bear standing in the water. A picture of a bear drinking water from a lake. A big bear in a lake on a sunny day. A large bear taking a swim in a river.	That looks like a fully grown bear walking in the grass. I am unable to see the image above. a bear in a field behind a fence. A bear walking on all fours near a fence. A black bear in grassy area next to a fence.
		A clock on a wall with four different time zones listed. Four clocks in a picture hung up on the wall on a building. A guest house clock has four clocks with different time zones. A clock hanging on a wall shows the time in different countries. A clock presents the time for four countries.	a clock the a weather vein on the top of it. The clock has a weather vane mounted on it. a close up of a clock on a pole with a wind tool. A rustic roman numeral clock with a weather vane on top. the clock is pointed towards the south and west.	An elaborate clock tower is made of stone. The bell tower of a building with clocks on the faces. A domed tower with gold finial, a pillars, and a clock. Two clocks viewable at the top of a building. a clock on a white tower in front of a clear sky.	A brick tower with a big clock in the middle of it. A large building with a metal clock on the front. A clock is shown on the side of a building. A church steeple clock with roman numerals set at 1:00. A clock set onto the face of an old stone building.	A tall clock tower is beside a shorter building. A white and red brick clock tower next to a building. A tall building has keyhole shaped windows and clock. A tall thin tower with a clock near a building. Old time clock tower next to modern office building.
T2I	Two tall giraffe standing next to each other. two giraffes and some green trees and yellow flowers. Two giraffes are peering up above some bushes. Two giraffes are towering over the plants under them. Two top half of two giraffes walking in trees.					
	An air plane wing flying over a very large mountain. the view outside of a plane with the view of a mountain. The view from an airplane, with the airplane wing extending out and white snow capped mountains below. A view from a plane window near a large snowy peak. The view of the beautiful snow capped mountaintop was partially obscured by the plane's wing.					
	A bird is sitting on top of a bench. A brown and orange bird is sitting on the back of a bench. A bird sitting on top of a wooden bench. a small bird is standing on a bench. A tiny bird is perched on the back of a chair.					

Figure 8. Utilizing our TEGAH framework, original samples are encoded and subjected to retrieval within the MS-COCO dataset, employing 64-bit hash codes to ascertain the top 5 results. Samples returned and denoted with a blue marker signify relevance to the query sample.

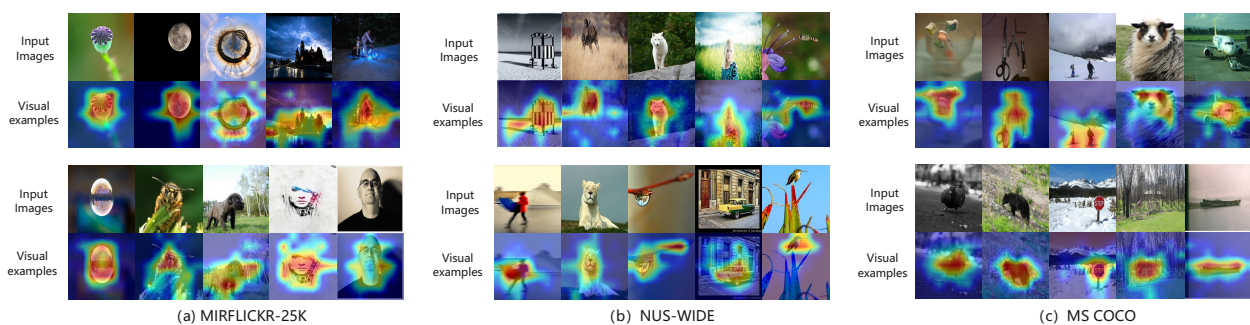


Figure 9. The results of visualization of 10 images randomly selected in three datasets using the Grad-CAM method.

### 5. Discussion

Although TEGAH has demonstrated strong performance, there remains room for improvement in the image modality. At present, the potential of image feature extraction and modality alignment has not been fully realized in certain complex scenarios, which may negatively impact the overall retrieval performance.



Furthermore, with the advancement of large language models (LLMs), these models have evolved into systems capable of processing multimodal information. This development presents new opportunities for enhancing both image and text modalities. Moving forward, we will shift our focus towards improving existing cross-modal retrieval methods. We will explore novel approaches for optimizing the extraction and integration of image features without relying on labeled information. Additionally, we will leverage the capabilities of LLMs to enrich text modality features, with the goal of generating more distinctive and robust hash codes.

While the current approach relies on labeled data, our future work will aim to reduce this dependency. We plan to investigate weakly-supervised, semi-supervised, and unsupervised learning methods to mitigate the reliance on high-quality labeled data, thereby making the retrieval methods more adaptable and applicable to a broader range of real-world scenarios. Moreover, we will validate the performance of these improved methods on large-scale unsupervised datasets to ensure their generalizability and scalability across different data environments.

## 6. Conclusions

In this paper, we propose a new Text-Enhanced Graph Attention Hashing for the Cross-Modal Retrieval (TEGAH) framework. First, we use the deep text feature extraction network to extract deep features of text information so that we can directly improve the extracted text features without changing the text features and improve the retrieval effect. Secondly, we regard label information as a mode and propose a multiscale label region hybrid network, which can supplement the modal features of text and alleviate the information gap when text information is scarce. Finally, in order to integrate the features of different modes, TEGAH uses GAT to learn a set of interdependent modal features, and optimizes the learned features for modal alignment and fusion, preserving the common features of different modes, bridging the information gap between different modes, and generating more distinctive hash codes. A large number of experiments on MIRFLICKR-25K, NUS-WIDE, and MS-COCO datasets prove that TEGAH method has good retrieval performance. A large number of experiments on the MIRFLICKR-25K, NUS-WIDE, and MS COCO datasets demonstrate that the TEGAH method achieves outstanding retrieval performance and significantly outperforms existing cross-modal hashing methods.

**Author Contributions:** Conceptualization, S.C.; methodology, Q.Z.; software, A.D.; validation, J.C.; formal analysis, S.C.; investigation, Q.Z.; resources, A.D.; data curation, J.C.; writing—original draft preparation, Q.Z.; writing—review and editing, J.C.; visualization, J.C.; supervision, A.D.; project administration, S.C.; funding acquisition, S.C. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Scientific and Technological Innovation 2030 Major Project under Grant 2022ZD0115800, the Basic Research Funds for Colleges and Universities in Xinjiang Uygur Autonomous Region under Grant XEDU2023P008, the Key Laboratory Open Projects in Xinjiang Uygur Autonomous Region under Grant 2023D04028, and the Graduate Research and Innovation Project of Xinjiang Uygur Autonomous Region under Grant XJ2024G086.

**Institutional Review Board Statement:** Not applicable.

**Data Availability Statement:** The MIRFLICKR-25K Dataset in this study is openly and freely available at <https://press.liacs.nl/mirflickr> (accessed on 21 October 2024). The NUS-WIDE dataset in this study are openly and freely available at <https://www.kaggle.com/datasets/xinleili/nuswide> (accessed on 21 October 2024). The MS-COCO 2014 dataset in this study are openly and freely available at <https://cocodataset.org> (accessed on 21 October 2024). Our code is available at <https://github.com/ShiShuMo/TEGAH> (accessed on 21 October 2024).

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Shi, L.; Luo, J.; Zhu, C.; Kou, F.; Cheng, G.; Liu, X. A survey on cross-media search based on user intention understanding in social networks. *Inf. Fusion* **2023**, *91*, 566–581. [\[CrossRef\]](#)
2. Zhu, L.; Zheng, C.; Guan, W.; Li, J.; Yang, Y.; Shen, H.T. Multi-Modal Hashing for Efficient Multimedia Retrieval: A Survey. *IEEE Trans. Knowl. Data Eng.* **2024**, *36*, 239–260. [\[CrossRef\]](#)
3. Zhang, Z.; Cheng, S.; Wang, L. Combined query image retrieval based on hybrid coding of CNN and Mix-Transformer. *Expert Syst. Appl.* **2023**, *234*, 121060. [\[CrossRef\]](#)
4. Chao, Z.; Cheng, S.; Li, Y. Deep internally connected transformer hashing for image retrieval. *Knowl. Based Syst.* **2023**, *279*, 110953. [\[CrossRef\]](#)
5. Jiang, Q.; Li, W. Deep Cross-Modal Hashing. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, Honolulu, HI, USA, 21–26 July 2017; pp. 3270–3278.
6. Cao, Y.; Liu, B.; Long, M.; Wang, J. Cross-Modal Hamming Hashing. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; Volume 11205, pp. 207–223.
7. Gu, W.; Gu, X.; Gu, J.; Li, B.; Xiong, Z.; Wang, W. Adversary Guided Asymmetric Hashing for Cross-Modal Retrieval. In Proceedings of the International Conference on Multimedia Retrieval, ICMR, Ottawa, ON, Canada, 10–13 June 2019; pp. 159–167.
8. Xie, D.; Deng, C.; Li, C.; Liu, X.; Tao, D. Multi-Task Consistency-Preserving Adversarial Hashing for Cross-Modal Retrieval. *IEEE Trans. Image Process.* **2020**, *29*, 3626–3637. [\[CrossRef\]](#)
9. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1106–1114. [\[CrossRef\]](#)
10. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
11. Jin, S.; Zhou, S.; Liu, Y.; Chen, C.; Sun, X.; Yao, H.; Hua, X. SSAH: Semi-Supervised Adversarial Deep Hashing with Self-Paced Hard Sample Generation. *Proc. Aaai Conf. Artif. Intell.* **2020**, *34*, 11157–11164. [\[CrossRef\]](#)
12. Liu, X.; Hu, Z.; Ling, H.; Cheung, Y. MTFH: A Matrix Tri-Factorization Hashing Framework for Efficient Cross-Modal Retrieval. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 964–981. [\[CrossRef\]](#)
13. Ye, Z.; Peng, Y. Sequential Cross-Modal Hashing Learning via Multi-scale Correlation Mining. *ACM Trans. Multim. Comput. Commun. Appl.* **2020**, *15*, 1–20. [\[CrossRef\]](#)
14. Bai, C.; Zeng, C.; Ma, Q.; Zhang, J.; Chen, S. Deep Adversarial Discrete Hashing for Cross-Modal Retrieval. In Proceedings of the International Conference on Multimedia Retrieval, ICMR, Dublin, Ireland, 8–11 June 2020; pp. 525–531.
15. Meng, M.; Sun, J.; Liu, J.; Yu, J.; Wu, J. Semantic Disentanglement Adversarial Hashing for Cross-Modal Retrieval. *IEEE Trans. Circuits Syst. Video Technol.* **2024**, *34*, 1914–1926. [\[CrossRef\]](#)
16. Yao, D.; Li, Z.; Li, B.; Zhang, C.; Ma, H. Similarity Graph-correlation Reconstruction Network for unsupervised cross-modal hashing. *Expert Syst. Appl.* **2024**, *237*, 121516. [\[CrossRef\]](#)
17. Wang, X.; Zou, X.; Bakker, E.M.; Wu, S. Self-constraining and attention-based hashing network for bit-scalable cross-modal retrieval. *Neurocomputing* **2020**, *400*, 255–271. [\[CrossRef\]](#)
18. Tu, R.; Mao, X.; Ji, W.; Wei, W.; Huang, H. Data-Aware Proxy Hashing for Cross-modal Retrieval. In Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR, Taipei, China, 23–27 July 2023; pp. 686–696.
19. Zhang, Z.; Luo, H.; Zhu, L.; Lu, G.; Shen, H.T. Modality-Invariant Asymmetric Networks for Cross-Modal Hashing. *IEEE Trans. Knowl. Data Eng.* **2023**, *35*, 5091–5104. [\[CrossRef\]](#)
20. Shu, Z.; Bai, Y.; Zhang, D.; Yu, J.; Yu, Z.; Wu, X. Specific class center guided deep hashing for cross-modal retrieval. *Inf. Sci.* **2022**, *609*, 304–318. [\[CrossRef\]](#)
21. Tu, R.; Mao, X.; Ma, B.; Hu, Y.; Yan, T.; Wei, W.; Huang, H. Deep Cross-Modal Hashing with Hashing Functions and Unified Hash Codes Jointly Learning. *IEEE Trans. Knowl. Data Eng.* **2022**, *34*, 560–572. [\[CrossRef\]](#)
22. Hu, P.; Zhu, H.; Lin, J.; Peng, D.; Zhao, Y.; Peng, X. Unsupervised Contrastive Cross-Modal Hashing. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 3877–3889. [\[CrossRef\]](#)
23. Huo, Y.; Qin, Q.; Dai, J.; Wang, L.; Zhang, W.; Huang, L.; Wang, C. Deep Semantic-Aware Proxy Hashing for Multi-Label Cross-Modal Retrieval. *IEEE Trans. Circuits Syst. Video Technol.* **2024**, *34*, 576–589. [\[CrossRef\]](#)
24. Li, X.; Yu, J.; Lu, H.; Jiang, S.; Li, Z.; Yao, P. MAFH: Multilabel aware framework for bit-scalable cross-modal hashing. *Knowl. Based Syst.* **2023**, *279*, 110922. [\[CrossRef\]](#)
25. Zou, X.; Wu, S.; Bakker, E.M.; Wang, X. Multi-label enhancement based self-supervised deep cross-modal hashing. *Neurocomputing* **2022**, *467*, 138–162. [\[CrossRef\]](#)
26. Wang, S.; Zhao, H.; Li, K. Discrete Joint Semantic Alignment Hashing for Cross-Modal Image-Text Search. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *32*, 8022–8036. [\[CrossRef\]](#)
27. Cao, Y.; Long, M.; Wang, J.; Yang, Q.; Yu, P.S. Deep Visual-Semantic Hashing for Cross-Modal Retrieval. In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, SIGKDD, San Francisco, CA, USA, 13–17 August 2016; pp. 1445–1454.
28. Yang, E.; Deng, C.; Liu, W.; Liu, X.; Tao, D.; Gao, X. Pairwise Relationship Guided Deep Hashing for Cross-Modal Retrieval. In Proceedings of the AAAI Conference on Artificial Intelligence, AAAI, California, CA, USA, 4–9 February 2017; pp. 1618–1625.

29. Wang, Y.; Chen, Z.; Luo, X.; Xu, X. A High-Dimensional Sparse Hashing Framework for Cross-Modal Retrieval. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *32*, 8822–8836. [[CrossRef](#)]
30. Zhang, X.; Liu, X.; Nie, X.; Kang, X.; Yin, Y. Semi-supervised semi-paired cross-modal hashing. *IEEE Trans. Circuits Syst. Video Technol.* **2023**, *34*, 6517–6529. [[CrossRef](#)]
31. Tu, R.; Jiang, J.; Lin, Q.; Cai, C.; Tian, S.; Wang, H.; Liu, W. Unsupervised Cross-Modal Hashing with Modality-Interaction. *IEEE Trans. Circuits Syst. Video Technol.* **2023**, *33*, 5296–5308. [[CrossRef](#)]
32. Shi, Y.; Zhao, Y.; Liu, X.; Zheng, F.; Ou, W.; You, X.; Peng, Q. Deep Adaptively-Enhanced Hashing with Discriminative Similarity Guidance for Unsupervised Cross-Modal Retrieval. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *32*, 7255–7268. [[CrossRef](#)]
33. Hu, H.; Xie, L.; Hong, R.; Tian, Q. Creating Something From Nothing: Unsupervised Knowledge Distillation for Cross-Modal Hashing. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, Seattle, WA, USA, 14–19 June 2020; pp. 3120–3129.
34. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is All you Need. In Proceedings of the Advances in Neural Information Processing Systems, NIPS, Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
35. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, ICCV, Montreal, BC, Canada, 11–17 October 2021; pp. 9992–10002.
36. Tu, J.; Liu, X.; Lin, Z.; Hong, R.; Wang, M. Differentiable Cross-modal Hashing via Multimodal Transformers. In Proceedings of the ACM International Conference on Multimedia, ACM MM, Lisboa, Portugal, 10–14 October 2022; pp. 453–461.
37. Xia, X.; Dong, G.; Li, F.; Zhu, L.; Ying, X. When CLIP meets cross-modal hashing retrieval: A new strong baseline. *Inf. Fusion.* **2023**, *100*, 101968. [[CrossRef](#)]
38. Liu, Y.; Wu, Q.; Zhang, Z.; Zhang, J.; Lu, G. Multi-Granularity Interactive Transformer Hashing for Cross-modal Retrieval. In Proceedings of the ACM International Conference on Multimedia, ACM MM, Ottawa, ON, Canada, 29 October–3 November 2023; pp. 893–902.
39. Wang, J.; Zeng, Z.; Chen, B.; Wang, Y.; Liao, D.; Li, G.; Wang, Y.; Xia, S. Hugs Are Better Than Handshakes: Unsupervised Cross-Modal Transformer Hashing with Multi-granularity Alignment. In Proceedings of the British Machine Vision Conference, BMVC, London, UK, 21–24 November 2022; p. 1035.
40. Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; Bengio, Y. Graph Attention Networks. In Proceedings of the International Conference on Learning Representations, ICLR, Vancouver, BC, Canada, 30 April–3 May 2018.
41. Duan, Y.; Chen, N.; Zhang, P.; Kumar, N.; Chang, L.; Wen, W. MS2GAH: Multi-label semantic supervised graph attention hashing for robust cross-modal retrieval. *Pattern Recognit.* **2022**, *128*, 108676. [[CrossRef](#)]
42. Bai, C.; Zeng, C.; Ma, Q.; Zhang, J. Graph convolutional network discrete hashing for cross-modal retrieval. *IEEE Trans Neural Networks Learn. Syst.* **2022**, *35*, 4756–4767. [[CrossRef](#)]
43. Radford, A.; Kim, J.W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. Learning Transferable Visual Models From Natural Language Supervision. In Proceedings of the International Conference on Machine Learning, ICML, Virtual Event, 18–24 July 2021; Volume 139, pp. 8748–8763.
44. Chung, J.; Gulcehre, C.; Cho, K.; Bengio, Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. In Proceedings of the Workshop on Deep Learning, NIPS, Montreal, QC, Canada, 13 December 2014.
45. Rubner, Y.; Tomasi, C.; Guibas, L.J. The Earth Mover’s Distance as a Metric for Image Retrieval. *Int. J. Comput. Vis.* **2000**, *40*, 99–121. [[CrossRef](#)]
46. Huiskes, M.J.; Lew, M.S. The mir flickr retrieval evaluation. In Proceedings of the 1st ACM International Conference on Multimedia Information Retrieval, Vancouver, BC, Canada, 30–31 October 2008; pp. 39–43.
47. Chua, T.S.; Tang, J.; Hong, R.; Li, H.; Luo, Z.; Zheng, Y. Nus-wide: A real-world web image database from national university of singapore. In Proceedings of the ACM international conference on image and video retrieval, Santorini Island, Greece, 8–10 July 2009; pp. 1–9.
48. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In Proceedings of the Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland, 6–12 September 2014, Proceedings, Part V 13; pp. 740–755.

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.