Multimedia Tools and Applications (2021) 80:17299–17314
https://doi.org/10.1007/s11042-020-09869-4

Semantics-preserving hashing based on multi-scale fusion for cross-modal retrieval

Hong Zhang [1,2] · Min Pan [1,2]

Received: 13 March 2020 / Revised: 18 August 2020 / Accepted: 11 September 2020 /
Published online: 2 November 2020

Abstract
Research on hash-based cross-modal retrieval has been a hotspot in the field of content-based multimedia retrieval research. Most deep cross-modal hashing methods only consider inter-modal loss that can remain local information of training data, and ignore the loss within data samples of the same modality that can remain the global information of dataset. In addition, they also ignore the factor that different scales of single modal data contain different semantic information, which affects the representation of data features. In this paper, we propose a semantics-preserving hashing method based on multi-scale fusion. More concretely, a multi-scale fusion pooling model is proposed for both image feature training network and text feature training network. Therefore, we can extract the multi-scale features of image dataset and solve the sparsity problem of text BOW vectors. When constructing the loss function, we consider intra-modal loss while considering inter-modal loss. Therefore, the output hash code retains both global and local underlying semantic correlation when image and text feature training network are trained. Experiment results on NUS-WIDE and MIRFlickr-25 K prove that against other existing methods, our algorithm improves cross-modal retrieval accuracy.

Keywords Cross-modal retrieval · Multi-scale fusion · Hash learning · Semantics preserving · Deep learning

1 Introduction

Development in information technology has led to explosive growth of multimedia data. At the same time, people's demand for information search to obtain diverse results is increasing.

✉ Hong Zhang
zhanghong_wust@163.com

[1] College of Computer Science & Technology, Wuhan University of Science & Technology, Wuhan 430081, China

[2] Hubei Province Key Laboratory of Intelligent Information Processing and Real-time Industrial System, Wuhan, China

Therefore, there are more and more researches [18, 23, 33, 19, 31, 20, 32, 15] on multimedia data analysis and cross-modal retrieval technology. Cross-modal retrieval is point to all relevant data of other modalities are accurately and quickly retrieved through the data of one modal.

Hash learning is widely used in cross-modal retrieval models [27, 21, 29, 1], because of its good low storage and efficient retrieval. In the past few decades of research, there are many hash methods for single-modal retrieval [25, 22, 16, 14, 8, 13, 35]. However, these methods are not suitable for cross-modal hash retrieval, because of the semantic gap between data in different modalities. Most existing cross-modal retrieval hashing methods [34] solve semantic gaps by mining the correlations of different modal data. The main cross-modal hashing methods can be divided into two categories: deep cross-modal hashing methods [5, 30, 7, 17, 24] and shallow cross-modal hashing methods [28, 12, 3, 11, 4]. Shallow cross-modal hashing methods mainly map each sample into a binary code based on hand-crafted features, so as to learn the hash function. However, this hash function cannot express the underlying features of samples and the retrieval efficiency is not ideal. Deep cross-modal hashing methods, in contrast to shallow cross-modal hashing methods, using the feature extraction capability of deep learning to learning effective representations of different modalities, which solve the problem of limited hand-crafted features expression ability. In addition, this method can also integrate feature learning into the process of hash code learning, ensuring the accuracy of hash code to obtain better retrieval efficiency.

Up to now, there has been a lot of research on deep cross-modal hashing retrieval, but they both ignore two attributes of cross-modal data. These two attributes contribute to retrieval accuracy, which is also the motivation of this paper. First, different scales of single modal data contain different semantic information; second, when judging the similarity between cross-modal data, most deep cross-modal hashing methods treat two cases in the same way, where only one tag is similar between different modal data and more than one tag is similar. They both ignore the fact that the similarity between different modal data is related to the number of labels they share.

Taken together, this paper proposes a Semantic-Preserving Hashing based on Multi-scale Fusion (SPHMF). The framework of SPHMF is shown in Fig. 1. First, Image Pooling Model for Multi-scale Fusion (IPMSF) and Text Pooling Model for Multi-scale Fusion (TPMSF) are used to extract multi-scale feature information of different modal data. Secondly, image-text pairs label information is used to train self-supervise network [17], so as to better mine the relevance between image and text. Finally, when we construct the loss function, we use multi-level similarity information of image-text pairs to construct the intra-modal loss. In addition, the loss function also includes pairwise loss and inter-modal loss.

The remaining paper is structured in the following manner. We summarize work related to cross-modal retrieval (section 2), present the deep learning architecture proposed (section 3), describe the construction of the loss function (section 4), discuss the results and experiments (section 5), and draw a conclusion (section 6).

2 Related works

As mentioned above, our proposed method is SPHMF, which is a cross-modal hashing retrieval method based on deep learning. Therefore, we conducted a series of researches on the two types of shallow and deep cross-modal hashing methods.

Most existing shallow cross-modal hashing methods are independent of feature learning and hash code learning, which leads to unsatisfactory retrieval results. This kind of typical

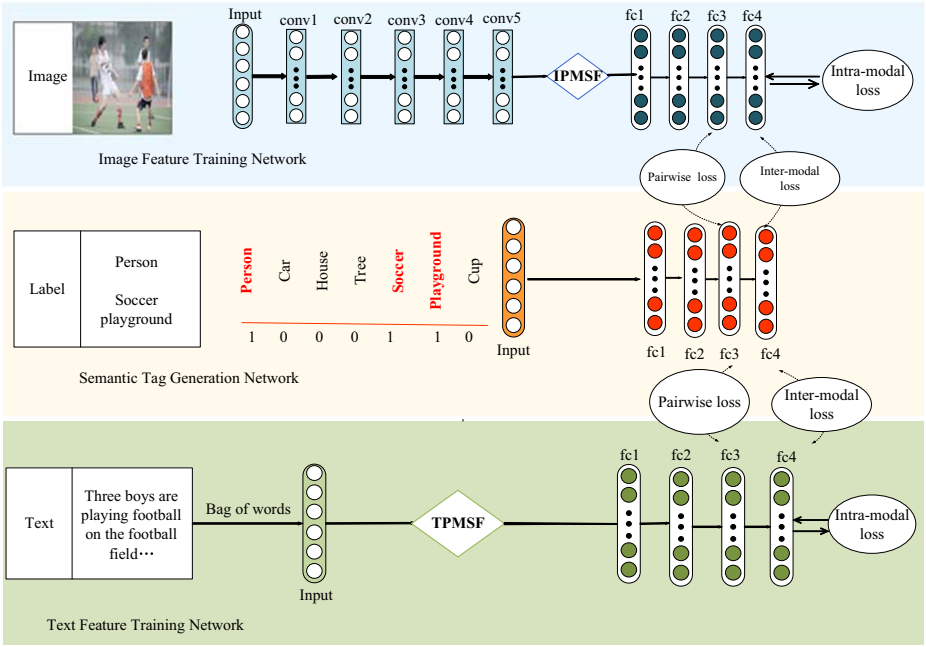

Fig. 1 The framework of SPHMF

methods include CVH [28], CCQ [12], CMSSH [3], SCM [11], and SePh [4]. CVH considers the similarity of intra-view and inter-view. CCQ jointly learns related maximal maps and composite quantizes. It converts multimedia data into binary code through an isomorphic potential space. CMSSH is a supervised cross-modal hash, which models hash learning through an enhanced classification paradigm. SCM uses tag information to build semantically similar matrices to learn a hash function. SePh changes the semantic matrix to a probability distribution, which is combined with the minimization of hamming spatial distribution to learn hamming space.

Deep cross-modal hashing methods use deep learning framework to study hash function. It can effectively catch the non-linear correlation between cross-modal instances. This kind of typical methods include CMNNH [5], DCMH [30], PRDH [29], A CMR [7], SSAH [17], and MCSCH [24]. CMNNH learns the hash function in deep learning framework by maintaining the relationship of intra-modal and the pairwise correspondence of inter-modal. DCMH performs feature learning and hash function learning simultaneously. PRDH guides the learning of hash codes by constructing different pairwise losses. ACMR differentiates the data of different modes and learns binary hash codes by adversarial learning methods and classification methods. SSAH uses a self-supervised network to generate semantic information of tags, as well as uses these information to guide the feature learning process of different modal data. MCSCH proposes to use multi-scale features to guide sequence hash learning, enhance the diversity of hash codes and promote the studying of hash functions.

Above related works achieved good results. Compared with unsupervised learning methods, supervised learning has achieved better results. For supervised learning methods, the key is to use limited training data and supervised information to learn semantic information with similar neighbor relations of the original data. The self-supervised network in the SSAH can build semantic association between multimedia data and find more information in labels. Therefore,

our method also uses this label self-supervised network. Our approach of SPHMF is different from the above methods in the following two ways: it extracts semantic information of different scales from single modal data and integrates it into the feature learning process; we construct intra-modal loss with multilevel semantic affinity matrix, inter-modal loss and pairwise loss.

3 Deep learning framework based on multi-scale fusion

As shown in Fig. 1, the overall network structure contains three parts, namely Image Feature Training Network (IMFN), Text Feature Training Network (TEFN), and Semantic Tag Generation Network (STGN). We use the output of STGN to guide training of IMFN and TEFN. After IMFN and TEFN are trained, the respective hash functions of image dataset and text dataset can be obtained. The hash function can be used to obtain the hash code of each modal data, then by counting and sorting hamming distance to complete cross-modal retrieval.

3.1 Image pooling model for multi-scale fusion

The overall structure of IMFN proposed in this paper is shown in the upper part of Fig. 1. Considering that features at different scales in image dataset represent different semantics, we propose an Image Pooling Model for Multi-scale Fusion (IPMSF) for IMFN. The setting of IPMSF is based on the idea of Spatial Pyramid Pooling [6] (SPP). We take the output of conv5 as the input of each pooling layer in IPMSF. Different pooling layers perform maximum pooling operation according to different scale regions, and then the output vectors of each pooling layer are concatenated as the input of fc1 to complete the training of IMFN. Therefore, the problems of limitation on size of input image in traditional CNN network and unreliable feature learning due to some information loss are solved. The settings for IPMSF are shown in Table 1. Unlike SPP, SPP directly connects the features of different scales, while IPMSF first merges the features of the same scale, and then connects the features of different scales in series, thereby reducing the parameters of the network model. In section 5.4.5, it can be seen that the IPMSF model reduces the computational overhead of the training process while maintaining the retrieval accuracy.

3.2 Text pooling model for multi-scale fusion

For text dataset, it is usually represented by a bag of words vector, which can easily lead to sparsity. To solve this problem, we design a Text Pooling Model for Multi-scale Fusion (TPMSF). First, the multi-scale features of text samples are extracted by pooling layer, and then multiple features are fused through convolutional layer. This process can capture the relevance among various words in text modal construction, and that is very useful for semantic relevance. The overall setting of TPMSF is shown in Table 2, and c is the number of class labels.

Table 1 Parameter settings for IPMSF model

Layers	Input	Pool	Kernel size	Stride
Pool1	The output of conv5	Max pooling	1 × 13	1 × 13
Pool2	The output of conv5	Max pooling	1 × 7	1 × 5
Pool3	The output of conv5	Max pooling	1 × 5	1 × 4

Table 2 Parameter settings for TPMSF model

Layers	Input	Pool	Kernel size	Stride
Pool1	BoW vector	Average pooling	1 × 50	1 × 50
Pool2	BoW vector	Average pooling	1 × 30	1 × 50
Pool3	BoW vector	Average pooling	1 × 15	1 × 50
Pool4	BoW vector	Average pooling	1 × 10	1 × 50
Pool5	BoW vector	Average pooling	1 × 5	1 × 50
Conv1	BoW vector	None	1 × 1 × 512 × c	1 × 1

The output of the TPMSF model is used as the input of TEFN. The model architecture for TEFN is shown in bottom part of Fig. 1.

3.3 Semantic tag generation network

In this paper, STGN is used to extract label semantic information of text-image pairs, and guide training of IMFN and TEFN. The overall setting of STGN is shown in the middle part of the Fig. 1.

The STGN is trained through the class label information and the neighbor relationship matrix **S**. After the training of STGN is completed, the label semantic hash code $H^{(s)}$ and label semantic features $F^{(s)}$ can be obtained through this network to guide training of IMFN and TEFN. In the training process, the inner product between the vectors is used to represent the correlation between any two output features or two hash codes. And in the meanwhile, the likelihood function is used to represent the inner product value between outputs under S supervision, as shown in formula (1):

$$p(S|H) = \begin{Bmatrix} sig\ (\theta_{ij}), S_{ij} = 1 \\ 1{-}sig\ (\theta_{ij}), S_{ij} = 0 \end{Bmatrix} \tag{1}$$

where sig () represents the sigmoid function, $\theta_{ij} = 1/2 < H_i, H_j >$, H_i, H_j represents the hash code of a set of samples output by the hash layer, and $S_{ij} = 1$ indicates that the two sample vectors are similar, $S_{ij} = 0$ means dissimilar.

Maximizing the likelihood function by minimizing the form of the negative log-likelihood function yields:

$$\min R = -\log p(S|H) = -\sum S_{ij}\langle H_i, H_j\rangle - \log\left(1 + e^{\langle H_i, H_j\rangle}\right) \tag{2}$$

If all parameters in the STGN are set to θ, then formula (2) can be used to represent all samples in $F^{(s)}$, $H^{(s)}$:

$$\min_{\theta} Js = -\sum_{i,j=1}^{n}\left(S_{ij}\langle F_i, F_j\rangle - log\left(1 + e^{\langle F_i, F_j\rangle}\right)\right) - \sum_{i,j=1}^{n}\left(S_{ij}\langle H_i, H_j\rangle - log\left(1 + e^{\langle H_i, H_j\rangle}\right)\right) \tag{3}$$

where F_i, F_j, H_i, H_j represent the features of the ith and jth groups and the hash codes of the ith and jth groups, respectively.

Since the hash code is lost from output to quantization into a binary hash code, a quantization error is added to the function, as follows:

$$\alpha\|H_i - sign(H_i)\|_F^2 \tag{4}$$

So, the final objective function is:

$$\min_{\theta} Js = -\sum_{i,j=1}^{n}\left(S_{ij}\langle F_i, F_j\rangle - log\left(1+e^{\langle F_i,F_j\rangle}\right)\right) - \sum_{i,j=1}^{n}\left(S_{ij}\langle H_i, H_j\rangle - log\left(1+e^{\langle H_i,H_j\rangle}\right)\right) + \alpha\|H_i - sign(H_i)\|_F^2 \tag{5}$$

In this paper, the parameter θ of STGN is studied by stochastic gradient descent and back propagation.

4 Learning cross-modal hash functions

Suppose there are n sets of training data points, each set of training data is composed of pairs of text-image, $X = \{x_i\}^n{}_{i=1}$ represents high-dimensional original image dataset, $Y = \{y_j\}^n{}_{j=1}$ represents the text dataset descripting image. Each pair of training data has a class label vector $l_i = \{l_{i1}, l_{i2}, \ldots, l_{ic}\}$, c is the number of dataset categories. In the label semantic information learning part, the class label information $\mathbf{L}^{n*c}$ $(l_1, l_2, \ldots, l_n)$ can be obtained from the text-image pairs (l_i represents a c-dimensional binary vector) to construct a similarity matrix **S**, $S_{ij} = 1$ means x_i is similar to y_j, $S_{ij} = 0$ means that they are not similar. In the hash function part, we specify the length of the output hash code as m.

Given the image data, text data and the similarity matrix **S**, the target of SPHMF is to study a hash code **B** that retains similarity. When constructing the entire objective function, the features of image and text and the semantic feature $F^{(s)}$ are used to construct pairwise loss to convey the neighborhood relationship in the label. Hash code of modal data construct inter-modal and intra-modal losses to preserve global and local semantic structure. Therefore, the overall objective function is:

$$J = J_{se} + \eta J_{inter} + \beta J_{intra} \tag{6}$$

where J_{se} is the pairwise loss, J_{inter} is the inter-modal loss, and J_{intra} is the intra-modal loss. ŋ and ß are used for balance the impacts of each term.

4.1 Pairwise loss

For the feature of image and text modal data, the inner product method is used to represent the similarity relationship, and the pairwise loss is used to transfer the nearest neighbor relationship of $F^{(s)}$, as shown in formula (7), (8):

$$\min_{\theta x} J_{se}^{x} = -\sum_{i,j=1}^{n}\left(S_{ij} < F_i^{(s)}, Z_j^{x} > - log\left(1+e^{< F_i^{(s)}, Z_j^{x} >}\right)\right) \tag{7}$$

$$\min_{\theta_y} J_{se}^{y} = -\sum_{i,j=1}^{n}\left(S_{ij} < F_i^{(s)}, Z_j^{y} > -log\left(1+e^{< F_i^{(s)}, Z_j^{y} >}\right)\right) \tag{8}$$

where J_{se}^{x} represent the pairwise loss for image samples, J_{se}^{y} represent the pairwise loss for text samples, $<F_i^{(s)}, Z_j^{y}>$, $<F_i^{(s)}, Z_j^{x}>$ represent the inner product of two vectors, respectively. They are used to weigh the similarity between text, image features and semantic retention features. Z_j^{x}, Z_j^{y} represent the feature representations of the jth group of image samples and text samples, respectively. θ_x, θ_y represent the parameters of image net and text net.

4.2 Inter-modal loss

The hash codes of the image and text are respectively constructed with the hash codes of label to construct the cross-entropy loss, that is, the inter-modal loss, to study the hash function, so that the label information can be inset in the modal data and make hash codes closer to the ideal hash codes.

$$\begin{aligned}\min_{\theta_x,\theta_y} J_{inter} = &-\frac{1}{n}\sum_{i,j=1}^{n}\left(H^{(s)}log\left(\sigma\left(H^{(x)}\right)\right)+\left(1-H^{(s)}\right)log\left(1-\sigma\left(H^{(x)}\right)\right)\right)\\ &-\frac{1}{n}\sum_{i,j=1}^{n}\left(H^{(s)}log\left(\sigma\left(H^{(y)}\right)\right)+\left(1-H^{(s)}\right)log\left(1-\sigma\left(H^{(y)}\right)\right)\right)\end{aligned} \tag{9}$$

where σ () represents the sigmoid function, n represents the number of training samples, $H^{(s)}$ represent the label semantic hash code and $H^{(x)}$ and $H^{(y)}$ represent the hash codes output by IMFN and TEFN. Because IMFN and TEFN are separate training, so you need to add a cross-modal adaptive constraint, as shown in the following formula:

$$\min_{\theta_x,\theta_y,B}\left(\left\|B^{(x)}-H^{(x)}\right\|_F^2+\left\|B^{(y)}-H^{(y)}\right\|_F^2\right) \tag{10}$$

Among them, $B^{(x)}$ and $B^{(y)}$ are the binary hash codes from images and text, so that information loss caused by the quantization of the hash code can be reduced. In addition, in order to obtain better performance, we set $\mathbf{B} = \mathbf{B}^{(x)} = \mathbf{B}^{(y)}$, so:

$$\min_{\theta_x,\theta_y,B}\left(\left\|B-H^{(x)}\right\|_F^2+\left\|B-H^{(y)}\right\|_F^2\right) \tag{11}$$

Therefore, the inter-modal loss function is formula (12), where γ is used for balance the impacts of the term of cross-modal adaptive constraint.

$$\begin{aligned}\min_{\theta_x,\theta_y,B} J_{inter} = &-\frac{1}{n}\sum_{i,j=1}^{n}\left(H^{(s)}\log\left(\sigma\left(H^{(x)}\right)\right)+\left(1-H^{(s)}\right)\log\left(1-\sigma\left(H^{(x)}\right)\right)\right)\\ &-\frac{1}{n}\sum_{i,j=1}^{n}\left(H^{(s)}\log\left(\sigma\left(H^{(y)}\right)\right)+\left(1-H^{(s)}\right)\log\left(1-\sigma\left(H^{(y)}\right)\right)\right)\\ &+\gamma\left(\left\|B-H^{(x)}\right\|\frac{2}{F}+\left\|\frac{2}{F}+\left\|B-H^{(x)}\right\|\frac{2}{F}\right)\end{aligned} \tag{12}$$

4.3 Intra-modal loss

The global semantic similarity matrix **A** of the training instance is used as supervision information to study each mode of the global semantic retention hash code, that is, the intra-modal loss. The element A_{ij} is defined as:

$$A_{\mathrm{ij}} = l_{\mathrm{i}}^{T} l_j \tag{13}$$

According to formula (13), we can obtain the complete matrix **A**, thereby obtaining the joint probability distribution P. The element P_{ij} is:

$$p_{ij} = \frac{A_{ij}}{\sum_{i=1}^{n}\sum_{j=1,j\neq i}^{n} A_{ij}} \tag{14}$$

The hamming distance between the hashing codes $B^{(x)}$ ($B^{(y)}$) of the network output is extracted by the image (text) feature to calculate the probability distribution Q^x (Q^y) in the image (text) mode. Use q_{ij}^x represent the similarity between two image instances, and q_{ij}^y represent the similarity between two text instances:

$$q_{ij}^{x} = \frac{e^{-d_H\left(b_i^x, b_j^x\right)}}{\sum_{k=1}^{n}\sum_{m=1,m\neq k}^{n} e^{-d_H\left(b_k^x, b_m^x\right)}} \tag{15}$$

$$q_{ij}^{y} = \frac{e^{-d_H\left(b_i^y, b_j^y\right)}}{\sum_{k=1}^{n}\sum_{m=1,m\neq k}^{n} e^{-d_H\left(b_k^y, b_m^y\right)}} \tag{16}$$

where $d_H()$ represent the hamming distance of two instances, b_i^x and b_j^x indicate binary code for two image data, b_i^y and b_j^y indicate binary code for two text data.

Here, we use the KL divergence to measure the similarity of the two probability distributions, P and Q^x (Q^y), to represent the intra-modal loss for image (text). As shown in the following formula:

$$\min_{\theta_x,B} J_{\mathrm{intra}}^{x} = KL\left(P \| Q^x\right) = \sum_{i=1}^{n}\sum_{j=1,j\neq i}^{n} p_{ij}\log\frac{p_{ij}}{q_{ij}^x} \tag{17}$$

$$\min_{\theta_y,B} J_{\mathrm{intra}}^{y} = KL\left(P \| Q^y\right) = \sum_{i=1}^{n}\sum_{j=1,j\neq i}^{n} p_{ij}\log\frac{p_{ij}}{q_{ij}^y} \tag{18}$$

$$\min_{\theta_x,\theta_y,B} J_{\mathrm{intra}} = \min_{\theta_x,\theta_y,B}\left(J_{\mathrm{intra}}{}^{x} + J_{\mathrm{intra}}^{y}\right) \tag{19}$$

We use the same method as in STGN network to optimize the object function to train image and text networks. The algorithm process as shown in Algorithm 1.

Algorithm 1. Learning algorithm for SPHMF

Require: Database: X, Y, similarity matrix **S**, multilevel semantic affinity matrix **A**, semantic feature $F^{(s)}$, semantic hash code $H^{(s)}$, the parametersβ,η,γ.

Ensure: Database instances codes **B**, θ_x and θ_y for image and text networks.

Train:

Initialization:

Initialize image and text network parameters θ_x, θ_y, batch size N_x=N_y=128, Number of iterations, t_x, t_y.

1: **repeat**

2: **for** iteration=1, 2, …, t_x **do**

3: Randomly select N_x image samples to form batch data;

4: For each sample x_i, output the feature Z_i^x and the output hash $H_i^{(x)}$ by forward propagation;

5: Construct objective function (6), according to formulas (7), (11), (17);

6: Update image network parameters θ_x using objective function (6) by using back propagation;

7: **end for**

8: **for** iteration=1, 2, …, t_y **do**

9: Randomly select 128 text samples to form batch data;

10: For each sample y_i, output the feature Z_i^y and the output hash $H_i^{(y)}$ by forward propagation;

11: Construct objective function (6), according to formulas (8), (11), (18);

12: Update text network parameters θ_y using objective function (6) by using back propagation;

13: **end for**

14: update **B** by: $B = sign((H^{(x)} + H^{(y)}))$

15: **Until** reach the maximum number of iterations

Retrieval:

1. Obtain the hash code output of the image and text using the trained image and text modal respectively;
2. Select the image(text) instance as the query term, the text(image) instance as the search term, and use their hash codes to count the hamming distance;
3. Sort the Hamming distance. Hamming distance is proportional to similarity.

5 Experiments

5.1 Datasets

In this paper, two standard cross-modal data sets are selected to complete cross-modal retrieval between text and image data, namely NUS-WIDE [2] data set and MIRFlickr-25 K [10] dataset.

NUS-WIDE contains 269,648 samples that is associated with text markup web images. We choose 10 frequent concepts, including 186,577 image-text pairs, and randomly select 105,000 examples for training set and 81,577 examples for test set.

MIRFlickr-25 K contains 25,000 image-text pairs. In our experiments, we select those samples that they are tagged by 20 text at least, and finally we have 20,015 image-text pairs. We randomly extract 15,000 pairs of image-text pairs for training set, and 5015 pairs for test set.

5.2 Implementation details

We choose five shallow cross-modal hashing methods CCQ [12], CVH [28], SCM_seq [11], CMSSH [3], SePh [4] and DCMH [30] of deep cross-modal hashing method to evaluate the performance of SPHMF. For the text network, we convert text samples into a 1000-dimensional word bag vector as input of the text network. As for the image network, we use deep features extracted from pretrained VGG-Net [9] on the ImageNet [26] as image input in all shallow cross-modal methods, and we use images of the same size as input of all deep cross-modal methods.

For the proposed method SPHMF, we use the pretrained VGG-Net model to initialize the first five convolutional networks in the image feature part. The hyper-parameters γ, , ß in SPHMF are empirically set as 1,1,10^{-4}, respectively, and they will be discussed in part 5.4.1 For the selection of learning rate of network, choose from 10^{-4} to 10^{-8} for network training. Set the network's batch training size to 128.

5.3 Evaluation protocol

We use mean accuracy precision (MAP) and precision-recall (PR) curves to evaluate the performance of all algorithm.

MAP: MAP represents the average value of the accuracy rate (AP) of each query. The value of MAP is positively correlated with the performance of the algorithm. AP is calculated as follows:

$$AP = \frac{1}{R}\sum_{k=1}^{n}\frac{\mathrm{k}}{n} * rel_k \tag{20}$$

where n represents the total number returned by the query, R is the number of related items in the retrieval set, R_k is the first k targets in the returned related targets, and rel_k indicates whether the k-th sample is a related sample. $rel_k = 1$ means it is a relevant sample, and $rel_k = 0$ means it is not relevant.

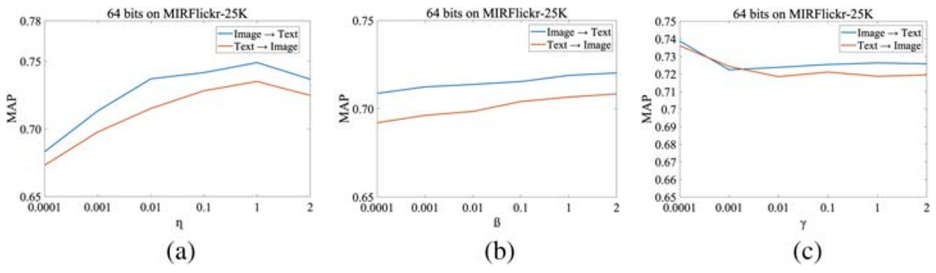

Fig. 2 MAP values with different hyper-parameters

Table 3 The MAP values of the label semantic hash code in different bits

Dataset	16bits	32bits	64bits
NUS-WIDE	0.9028	0.9558	0.9917
MIRFlickr-25 K	0.9718	0.9878	0.9951

Precision-Recall: precision reflects the retrieval accuracy. Recall reflects the comprehensiveness of the search. PR curves are often used in information retrieval to evaluate search efficiency.

5.4 Cross-modal retrieval results

5.4.1 Parameters sensitivity evaluation results

We study the effect of hyper-parameter ŋ, ß and γ on retrieval results on MIRFlickr-25 K with the hash length being 64-bits. Figure 2 (a) show the effect of the hyper-parameter ŋ with the value between 0.0001 and 2. Figure 2 (b) show the effect of the hyper-parameter ß with the value between 0.0001 and 2. Figure 2 (c) show the influence of hyper-parameter γ with the value between 0.0001 and 2. It can be found that SPHMF is sensitive to the hyper-parameter ŋ. Furthermore, SPHMF can get the best retrieval result when the hyper-parameter $ŋ = 1$, $ß = 1$, $\gamma = 10^{-4}$.

5.4.2 Hash retrieval task

In the experiment, the first step is to train STGN, and then we can obtain the semantic retention features $F^{(s)}$ and semantic retention hash codes $H^{(s)}$ of training data. We evaluate $H^{(s)}$ in NUS-

Table 4 Comparison of MAP values of I → T and T → I on MIRFlickr-25 K dataset

Task	Method	The length of hash code		
		16 bits	32 bits	64 bits
I → T	SPHMF	**0.7352**	**0.7450**	**0.7501**
	CCQ	0.6492	0.6513	0.6513
	CVH	0.6135	0.5983	0.5891
	SCM_seq	0.6482	0.6523	0.6533
	CMSSH	0.6227	0.6135	0.6105
	SePh	0.7012	0.7120	0.7191
	DCMH	0.7248	0.7330	0.7364
T → I	SPHMF	**0.7764**	**0.7835**	**0.7845**
	CCQ	0.6357	0.6357	0.6296
	CVH	0.6145	0.5983	0.588
	SCM_seq	0.6691	0.6721	0.6762
	CMSSH	0.6195	0.6114	0.6105
	SePh	0.6421	0.6525	0.5658
	DCMH	0.7634	0.7685	0.7787

Table 5 Comparison of MAP values of I → T and T → I on NUS-WIDE dataset

Task	Method	The length of hash code		
		16bits	32bits	64bits
I → T	SPHMF	**0.6218**	**0.6276**	**0.6391**
	CCQ	0.5137	0.5147	0.5157
	CVH	0.3886	0.3822	0.3769
	SCM_seq	0.5270	0.5238	0.5279
	CMSSH	0.49397	0.4765	0.4609
	SePh	0.6175	0.6246	0.6297
	DCMH	0.5230	0.5230	0.5323
T → I	SPHMF	**0.5991**	**0.5887**	**0.6152**
	CCQ	0.5051	0.5021	0.4980
	CVH	0.3640	0.3628	0.3601
	SCM_seq	0.5243	0.5162	0.5233
	CMSSH	0.4209	0.3803	0.3728
	SePh	0.5676	0.5737	0.5967
	DCMH	0.5887	0.5964	0.6089

WIDE and MIRFlickr-25 K datasets, and calculate the MAP values under different bits, as shown in Table 3. By comparison, 64bits hash code can be considered ideal Hash code.

We use $F^{(s)}$ and $H^{(s)}$ to guide the training of IMFN and TEFN. After completing the training of IMFN and TEFN, we calculate the MAP values and PR curves for two retrieval tasks: image retrieval text (I → T) and text retrieval image (T → I) on two datasets. As shown in Tables 4 and 5, SPHMF increases the MAP value from 0.7364 to 0.7501 and 0.6297 to 0.6391 in I → T task based on 64bits hash code. In T → I task, the MAP value increased from 0.7787 to 0.7845, and from 0.6089 to 0.6152 based on 64bits hash code.

The corresponding precision-recall curves on two datasets are plotted in Figs. 3 and 4. We can see from the Figs. 3 and 4, the algorithm proposed in this paper achieves higher accuracy at most recall levels than comparison methods.

It can be seen from the above analysis that SPHMF has significant advantages. Compared with the unsupervised methods, we use label information for supervised training, and these label information can provide the original relationship of data for hash code learning. Compared with the supervised methods, we consider the complementary information and correlation among multi-scale features, make the most of the multi-scale information of image and work out the sparsity of text BOW vectors. In addition, the construction of inter-modal loss in the loss function can provide more accurate judgment of similarity or dissimilarity for two data

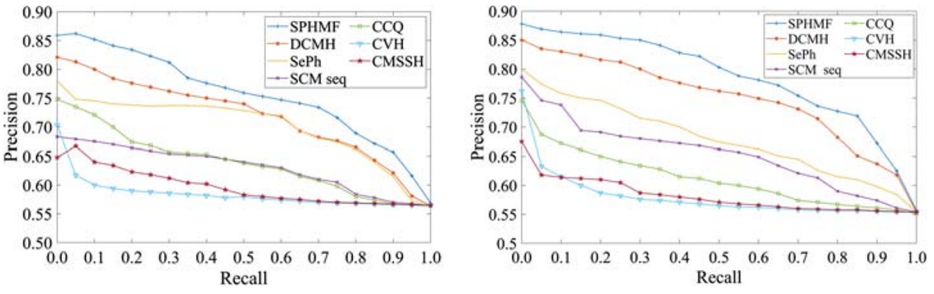

Fig. 3 Precision-Recall curves (MIRFlickr-25 K dataset 64bits hash)

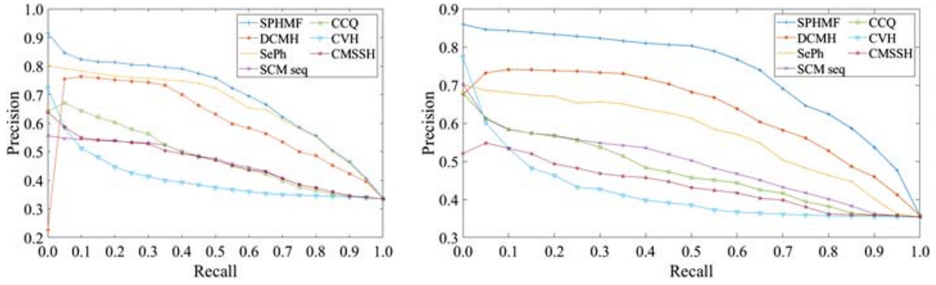

Fig. 4 Precision-Recall curves (NUS-WIDE dataset 64bits hash code)

of different modes, and the construction of intra-modal loss can make the hash code of model output have global potential semantic correlation. Thus, the accuracy of retrieval is improved to some extent.

5.4.3 Comparison of training time

We conduct a comparative experiment between DCMH and SPHMF on MIRFlickr-25 K dataset to assess the training efficiency of SPHMF. We can observe that SPHMF trains faster than DCMH, and the value of the MAP on the retrieval task is better from Fig. 5.

5.4.4 Impact analysis of each loss function

The objective function is composed of three parts, namely intra-modal loss, inter-modal loss and pairwise loss. In order to find out the influence of each loss function on the final search results, we conduct experiments. Therefore, we divide the experiment into the following three situations: the objective function includes intra-modal loss and inter-modal loss (SPHMF-1); the total function includes intra-modal loss and pairwise loss (SPHMF-2); the objective function includes inter-modal loss and pairwise loss (SPHMF-3).

The experimental results of different methods on two datasets are shown in Table 6. We can find out that the order of the MAP values of the experimental results from large to small is:

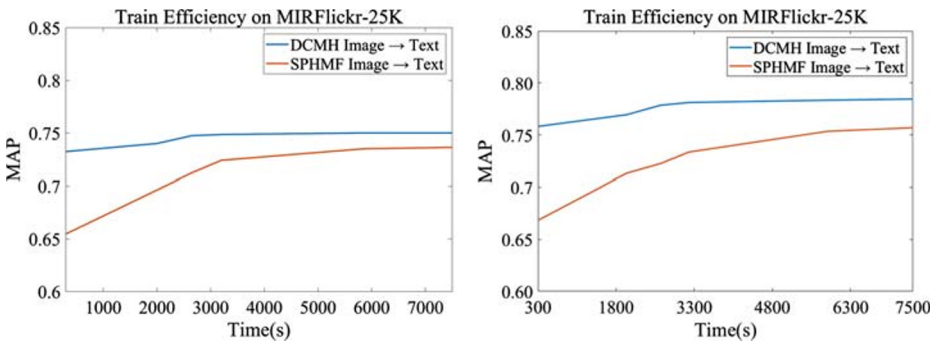

Fig. 5 Training efficiency of SPHMF and DCMH

Table 6 The impact of different loss on MAP values (MIRFlickr-25 K dataset and NUS-WIDE dataset 64 bits)

Task	Method	Datasets	
		MIRFlickr-25 K	NUS-WIDE
I → T	SPHMF	**0.7501**	**0.6391**
	SPHMF-1	0.6994	0.5771
	SPHMF-2	0.6891	0.5714
	SPHMF-3	0.7482	0.6273
T → I	SPHMF	**0.7845**	**0.6152**
	SPHMF-1	0.7687	0.5634
	SPHMF-2	0.7434	0.5154
	SPHMF-3	0.7837	0.5932

SPHMF, SPHMF-3, SPHMF-1, and SPHMF-2. Therefore, we know that inter-modal loss has the greatest influence on the final retrieval results, followed by pairwise loss, and finally intra-modal loss. However, the retrieval effect of combining these three loss functions is the best.

5.4.5 Compare IPMSF pooling and SPP pooling

In order to prove the effectiveness of IPMSF, the IPMSF model in the network structure shown in Fig. 1 is replaced with the SPP model, and other network settings are the same, and comparative experiments are performed. Table 7 is the result of experimental comparison.

Table 7 shows that the using the IPMSF model can slightly improve retrieval performance compared to the SPP model, but it is not much different. However, from Table 8 can be seen that the IPMSF-based network model uses less space than the SPP-based network model in space utilization. This is because the IPMSF model reduces the parameters of the network model compared with SPP model.

6 Conclusions

This paper proposes a semantics-preserving hashing method based on multi-scale fusion for cross-modal retrieval, called SPHMF. SPHMF supervises both image feature training network and text feature training network by using cross-modal label information. For image feature training network and text feature training network, multi-scale fusion pooling model is proposed to extract multi-scale information of data; We construct intra-modal loss with multilevel semantic affinity matrix, inter-modal loss and pairwise loss. Therefore, the hash

Table 7 Comparison of MAP values of different pooling methods (MIRFlickr-25 K dataset and NUS-WIDE dataset 64 bits)

Task	Method	Datasets	
		MIRFlickr-25 K	NUS-WIDE
I → T	IPMSF	**0.7501**	**0.6391**
	SPP	0.7467	0.6361
T → I	IPMSF	**0.7845**	**0.6152**
	SPP	0.7768	0.6127

Table 8 Comparison of the space between the trained IPMSF and SPP models

Method	Model size with different lengths of hash codes (MB)		
	16 bits	32 bits	64 bits
IPMSF	592.1	592.1	592.2
SPP	989.5	989.6	989.6

code learned by SPHMF can better retain the original information of modal data. The NUS - WIDE and MIRFlickr-25 K datasets verify the validity of SPHMF. But this article only explores retrieval method between image and text. In the later work, we will further improve our retrieval algorithm and apply it to multimedia data of more modalities, including image, text, audio and video.

References

1. Baeza-Yates R, Ribeiro-Neto B (1999) Modern information retrieval[M]. ACM press, New York
2. Bronstein M M, Bronstein A M, Michel F, et al. (2010) Data fusion through cross-modality metric learning using similarity-sensitive hashing[C]//2010 IEEE computer society conference on computer vision and pattern recognition. IEEE, 3594-3601
3. Chua T S, Tang J, Hong R, et al. (2009) NUS-WIDE: a real-world web image database from National University of Singapore[C]//Proceedings of the ACM international conference on image and video retrieval. 1-9
4. Han Y, Wu F, Tian Q, Zhuang Y (2012) Graph-Guided Sparse Reconstruction for Region Tagging. IEEE Conference on Computer Vision and Pattern Recognition
5. He K, Zhang X, Ren S et al (2015) Spatial pyramid pooling in deep convolutional networks for visual recognition[J]. IEEE transactions on pattern analysis and machine intelligence 37(9):1904–1916
6. He X, Peng Y, Xie L (2019) A new benchmark and approach for fine-grained cross-media retrieval[C]// Proceedings of the 27th ACM International Conference on Multimedia. 1740-1748
7. Huiskes MJ, Lew MS (2008) The MIR flickr retrieval evaluation[C]//Proceedings of the 1st ACM international conference on Multimedia information retrieval. 39-43
8. J Zhang J, Peng Y (2018) Query-adaptive image retrieval by deep-weighted hashing[J]. IEEE Transactions on Multimedia 20(9):2400–2414
9. Jiang QY, Li WJ (2017) Deep cross-modal hashing[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 3232-3240
10. Kumar S, Udupa R (2011) Learning hash functions for cross-view similarity search[C]//Twenty-Second International Joint Conference on Artificial Intelligence
11. Li C, Deng C, Li N et al. (2018) Self-supervised adversarial hashing networks for cross-modal retrieval[C]// Proce-edings of the IEEE conference on computer vision and pattern recognition. 4242-4251
12. Lin Y, Zheng Z, Zhang H, et al. Bayesian query expansion for multi-camera person re-identification[J]. Pattern Recognition Letters, 2018.
13. Lin Z, Ding G, Hu M, et al. Semantics-preserving hashing for cross-view retrieval[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 3864-3872.
14. Long M, Cao Y, Wang J, et al. Composite correlation quantization for efficient multimodal retrieval[C]//// Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval. 2016: 579-588.
15. Lu X, Chen Y, Li X (2017) Hierarchical recurrent neural hashing for image retrieval with hierarchical convolutional features[J]. IEEE Transactions on Image Processing 27(1):106–120
16. Mu N, Xu X, Zhang X et al (2018) Salient object detection using a covariance-based CNN model in low-contrast images[J]. Neural Computing and Applications 29(8):181–192
17. Peng Y, Huang X, Zhao Y (2017) An overview of cross-media retrieval: Concepts, methodologies, benchmarks, and challenges[J]. IEEE Transactions on circuits and systems for video technology 28(9): 2372–2385

18. Peng Y, Zhang J, Ye Z. Deep reinforcement learning for image hashing[J]. IEEE Transactions on Multimedia, 2019.
19. Russakovsky O, Deng J, Su H et al (2015) Imagenet large scale visual recognition challenge[J]. International journal of computer vision 115(3):211–252
20. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.
21. Wang B, Yang Y, Xu X, et al. Adversarial cross-modal retrieval[C]//Proceedings of the 25th ACM international conference on Multimedia. 2017: 154-162.
22. Wu F, Han Y, Liu X et al (2012) The heterogeneous feature selection with structural sparsity for multimedia annotation and hashing: a survey[J]. International Journal of Multimedia Information Retrieval 1(1):3–15
23. Xu Y, Han Y, Hong R et al (2018) Sequential video VLAD: Training the aggregation locally and temporally[J]. IEEE Transactions On Image Processing 27(10):4933–4944
24. Yang E, Deng C, Liu W, et al. Pairwise relationship guided deep hashing for cross-modal retrieval[C]// Thirty-first AAAI conference on artificial intelligence. 2017.
25. Yang Y, Ma Z, Hauptmann AG et al (2012) Feature selection for multimedia analysis by sharing information among multiple tasks[J]. IEEE Transactions on Multimedia 15(3):661–669
26. Ye Z, Peng Y. Multi-scale correlation for sequential cross-modal hashing learning[C]//Proceedings of the 26th ACM international conference on Multimedia. 2018: 852-860.
27. Zhaoda Ye and Yuxin Peng. 2019. Sequential Cross-Modal Hashing Learning via Multi-scale Correlation Mining. ACM Trans. Multimedia Comput. Commun. Appl. 15, 4, Article 105 (December 2019), 20 pages.
28. Yuan M, Peng Y. Text-to-image synthesis via symmetrical distillation networks[C]//Proceedings of the 26th ACM international conference on Multimedia. 2018: 1407-1415.
29. Yuwono B, Lee DL. Server ranking for distributed text retrieval systems on the internet[M]//Database Systems For Advanced Applications' 97. 1997: 41-49.
30. Zhang D, Li W J. Large-scale supervised multimodal hashing with semantic correlation maximization[C]// Twenty-Eighth AAAI Conference on Artificial Intelligence. 2014.
31. Zhang H, Wang T, Dai G (2020) Semi-supervised cross-modal common representation learning with vector-valued manifold regularization[J]. Pattern Recognition Letters 130:335–344
32. Zhang J, Han Y, Jiang J (2017) Semi-supervised tensor learning for image classification[J]. Multimedia Systems 23(1):63–73
33. Zhang J, Peng Y (2017) SSDH: semi-supervised deep hashing for large scale image retrieval[J]. IEEE Transactions on Circuits and Systems for Video Technology 29(1):212–225
34. Zhang J, Peng Y (2019) Multi-pathway generative adversarial hashing for unsupervised cross-modal retrieval[J]. IEEE Transactions on Multimedia 22(1):174–187
35. Zhuang Y, Yu Z, Wang W, et al. Cross-media hashing with neural networks[C]//Proceedings of the 22nd ACM international conference on Multimedia. 2014: 901-904.