IET Journals
The Institution of Engineering and Technology

Self-Taught Cross-Modal Hashing with Minimal Semantic Loss

ISSN 1751-8644
doi: 0000000000
www.ietdl.org

Jianing Du[1], *Zhikui Chen*[1,2*], *Fangming Zhong*[1], *Xiru Qiu*[1]

[1] *School of Software Technology, Dalian University of Technology, Dalian, China*
[2] *Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, Dalian, China*
* *E-mail: zkchen@dlut.edu.cn*

Abstract: Cross-modal hashing has received widespread attention due to the high retrieval efficiency, which plays an extremely important role in cross-modal retrieval. Recently, many cross-modal hashing methods have been proposed to establish the semantic connection of different modalities. However, most of these methods only use a simple quantization strategy, resulting in large quantization error and inferior hash codes. To address this issue, in this paper, we propose a novel Self-Taught Cross-Modal Hashing (STCMH) to minimize the semantic encoding loss. In particular, the common semantic representations across different modalities are firstly learned based on collective matrix factorization. Then the quantization procedure based on orthogonal transformation is integrated to encode the semantic representations into discriminative binary codes. Moreover, similarity preservation is imposed to further boost the discriminative power. Finally, hashing functions learning is formulated as a binary classification problem by self-taught scheme. Experimental results on three public datasets demonstrate that STCMH significantly outperforms most state-of-the-art cross-modal hashing methods.

1 Introduction

During the past decades, multimedia data have been growing dramatically on Internet and social websites. The considerable volume, high dimensionality, and various modalities of these data make it challenging to perform multimedia retrieval, especially the cross-modal retrieval that mainly concerns the query of using samples from one modality such as image to search relevant instances from another modality such as text. A number of researchers have devoted to cross-modal retrieval [1–3], which have witnessed great success. However, the searching and storage costs of most cross-modal retrieval methods are prohibitively high in the case of large-scale and high-dimensional datasets [4–6]. Fortunately, hashing is exploited as an effective solution for low storage and fast search [7], which has been successfully applied to computer vision [8] and image retrieval [9, 10] tasks.

Inspired by this, some researchers integrate hashing with cross-modal retrieval, named as cross-modal hashing, to transform high-dimensional data of different modalities into compact binary codes while preserving the manifold structure of original data. However, due to inconsistent feature dimensions and semantic gaps between different modalities, the design of cross-modal hashing methods is still a significant challenge. Most previous cross-modal hashing approaches [11–13] concentrate on finding a common Hamming space, where the cross-correlation among different modalities can be directly measured using linear or nonlinear projection. The cross-modal retrieval can be performed effectively in the Hamming space by bit XOR operation, which significantly reduces the computational complexity [17]. In addition, the storage cost of binary codes has also been highly compressed compared with the original high-dimensional data. Therefore, the large-scale and high-dimensional data can be handled effectively by cross-modal hashing.

Recently, more cross-modal hashing approaches have been proposed to bridge the semantic gap across different modalities, and impressive success has been achieved [14–19]. According to the utilization of label information, they can be classified into unsupervised methods and supervised methods. Unsupervised cross-modal hashing methods [14–16] only utilize co-occurrence information of training data to mine the latent semantic concept of different modalities. For example, Latent Semantic Sparse Hashing (LSSH) [14] uses matrix decomposition and sparse coding to learn latent semantic representations and then integrates them into a joint abstraction space. Collective Matrix Factorization Hashing (CMFH) has been proposed in [15] which learns a unified hash code for different modalities of the same object by matrix factorization. Different from unsupervised methods, the supervised cross-modal hashing methods [17–19] exploit label information to preserve semantic similarity. Thus they generally can effectively solve the semantic gap and get better results. For instance, Supervised Matrix Factorization Hashing (SMFH) [17], which is an extension of CMFH, considers preserving similarity information by taking advantage of available labels. The results of SMFH are significantly superior to that of CMFH.

However, a common limitation shared by most of the existing unsupervised and supervised methods is that they generate the binary codes using only a simple thresholding strategy. It will lead to a large quantization error and decrease the discriminative ability of the learned binary codes [6, 20]. Because the optimization of learning hash codes with binary constraints is NP-hard [11, 21], most of the previous methods disregard the binary constraints to learn a continuous representation. Then, a simple thresholding strategy is adopted to the continuous representation for generating binary codes. Such scheme results in a large quantization loss of the continuous representation and decreases the discriminative power of binary codes. In [22, 23], the authors employed the *sigmoid* or *tanh* relaxation instead of the *sign* function to avoid the large quantization loss. And the results in [22, 23] show that reducing quantization error can exactly improve the quality of hash codes in spite of the high computational cost of them for large-scale data [6].

In this paper, we put forward an effective Self-Taught Cross-Modal Hashing (STCMH) method. The motivation of our work is to reduce the quantization error and learn more discriminative binary codes to further improve the cross-modal retrieval performance. We mainly focus on minimizing the encoding loss of common semantic representations. In addition, we consider the semantic consistence of samples from different modalities. The samples that are from different modalities and describe the same object should be close to each other in the expected space, and vice versa. The whole framework of the proposed STCMH consists of two phases, namely offline and online process, as illustrated in Fig. 1. The offline process, which aims at generating binary hash codes for the database and learning hashing functions for out-of-sample data, includes the following

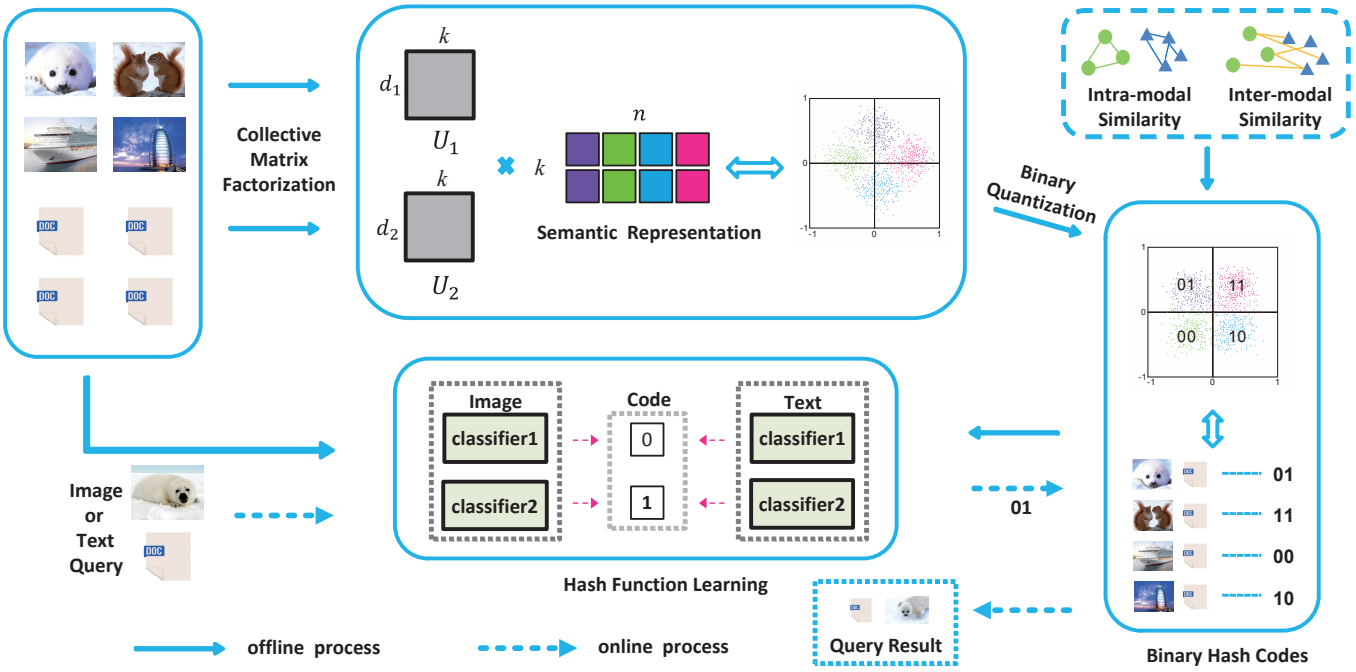


Fig. 1: The whole framework of the proposed STCMH.

three steps. Firstly, to extract common semantic information, the features from different modalities are jointly projected into latent semantic space using collective matrix factorization. Secondly, the binary quantization loss is minimized by orthogonal transformation, thus samples of the same class could be further converted to similar binary codes. Moreover, the similarity preservations including intra-modal and inter-modal similarities are taken into consideration by leveraging local geometric structure and label information, respectively. Thirdly, motivated by Self-Taught Hashing (STH) [24], hashing functions learning is formulated as a binary classification problem. A set of classifiers are trained based on training data and the learned binary codes. In the online process that performs query encoding and searching, the binary codes of out-of-sample data can be generated directly by the hashing functions. Therefore, cross-modal retrieval can be easily conducted based on the binary codes by Hamming distance.

The major contributions of this paper can be summarized as follows:

- We put forward a novel self-taught cross-modal hashing method that combines the semantic feature learning and the binary quantization process to project the original data from different modalities into the common low-dimensional Hamming space with minimal semantic loss.
- The binary codes learning for out-of-sample data is formulated as a binary classification problem. Different from existing methods that learn linear or nonlinear projections, our method takes advantage of SVM classifier to generate more discriminative binary codes and further reduce the quantization error.
- Extensive experiments are conducted on three datasets to evaluate the effectiveness of the proposed method. Experimental results show that our method outperforms several state-of-the-art methods.

The rest of this paper is organized as follows. Section 2 gives a brief introduction to the related work. The proposed method is presented in details in Section 3. Section 4 reports the experimental results on three datasets. Finally the conclusions are given in Section 5.

2 Related work

Many cross-modal hashing approaches have been proposed in recent years. They can be separated into unsupervised, pairwise-based, and supervised approaches.

The first class is unsupervised cross-modal hashing [11, 14, 15, 25–27], which is widely applied without any prior knowledge. Typically, it only makes use of features from various modalities to learn a common space where their correlations are maximized. The extension of Spectral Hashing (SH) [21], called Cross-View Hashing (CVH) [11], learns hashing functions with a generalized eigenvalue formulation. Song et al. put forward Inter-Media Hashing (IMH) [25] to learn hashing functions by linear regression while taking into account both intra-media and inter-media consistency. To improve training efficiency for large-scale multi-modal data, a new model named Linear Cross-Modal Hashing (LCMH) [26] was proposed to shorten the training process to linear time. LCMH also takes inter-similarity and intra-similarity into consideration, which makes generated hash codes more effective and achieves better retrieval performance in large-scale datasets. The aforementioned methods generate two hash codes that correspond to instances from two different modalities, respectively, which may cause ambiguity. To alleviate this problem, several methods that learn unified codes for different modalities, such as CMFH [15], LSSH [14], and C-JMFH [27] have been proposed. With the assumption of identical binary codes generated by different modalities, Collective Matrix Factorization Hashing (CMFH) [15] introduces collective matrix factorization into cross-modal domain for the first time and improves the retrieval accuracy effectively. Similarly, Cluster-based Joint Matrix Factorization Hashing (C-JMFH) [27] integrates the cross-modal cluster representation into the joint matrix factorization process with the constraint of generating unified hash codes. It simultaneously calculates intra-modal, inter-modal, and cluster-based similarities. Besides, Latent Semantic Sparse Hashing (LSSH) [14] captures the latent representations from text and image by matrix factorization and sparse coding, respectively. Then, the latent representations are combined to learn a unified binary hash codes.

The second category is pairwise-based cross-modal hashing [12, 28–31], which can take advantage of similar or dissimilar pairs in training data and then better common space can be learned. One of the earliest proposed methods in the field of cross-modal hashing is Cross-Modal Similarity Sensitive Hashing (CMSSH) [28] which

learns two hashing functions for different modalities by standard AdaBoost algorithm. Nevertheless, it only uses the similar pairs and dissimilar pairs between various modalities. Thus, the intra-modal similarity is not considered. To address this issue, an approach [29] based on coupled siamese networks is presented to jointly integrate intra-modal and inter-modal similarity, called MM-NN. Zhen et al. presented Co-Regularization Hashing (CRH)[12] that studies hashing functions learning based on a boosted co-regularization framework. They also introduced a probabilistic model [30] to generate hash codes while preserving intra-modal and inter-modal similarities. In [31], Quantized Correlation Hashing (QCH) integrates both hashing functions learning and binary codes generation into a single objective function for multi-modality data, where only similar pairs are concerned, which belongs to the weakly supervised case.

The last category involves supervised cross-modal hashing approaches [4, 17, 32, 33] that exploit additional label information to yield better retrieval performance. Semantic Correlation Maximization (SCM) [32] seamlessly integrates semantic labels into the hashing functions learning with linear-time complexity. In [17], Supervised Matrix Factorization Hashing (SMFH) extends CMFH to a supervised learning framework, which preserves intra-modal similarity with local structure of data and inter-modal similarity by label information and has achieved promising results. Based on multi-modal dictionary learning, Wu et al. proposed a hashing method named sparse multi-modal hashing (SM2H) [4] which adopts the hypergraph to build the correlations across multi-modal data, and then imposes it on dictionary learning, thus the sparse codesets generated by a hashing scheme can maintain inter-similarity across different modalities and intra-similarity in each modality simultaneously. Semantic Boosting Cross-Modal Hashing (SBCMH) [18] focuses on reducing the semantic gap and transforms original data into the semantic representations by multi-class logistic regression. Then weak classifiers and strong classifiers are used to learn hashing functions and binary hash codes, which further enhances semantic consistency.

Motivated by the excellent results of [14, 15, 17], we make further efforts to explore a novel hashing method with minimal semantic loss. Compared with them, our method can not only capture latent semantic information, but also make encoding quantization loss minimized.

3 Proposed method

The details of the proposed STCMH are described in this section. For simplifying the presentation, we restrict the discussion of STCMH to the most common two modalities, i.e., image and text. Note that our STCMH can be further extended to the multi-modal scenario.

3.1 Problem formulation

Suppose that the training set consists of n objects with two modalities, denoted by $O = \{o_i\}_{i=1}^{n}$. For the i-th object $o_i = \{x_i^{(1)}, x_i^{(2)}\}$, $x_i^{(1)}$ represents the d_1-dimensional image feature, and $x_i^{(2)}$ represents the d_2-dimensional text feature (usually, $d_1 \neq d_2$). Moreover, $\overline{\mathbf{L}} \in \{0,1\}^{c\times n}$ is the available class label for supervised case, where c is the number of categories. Here, we denote the original data from image modality as $\mathbf{X}^{(1)} = \{x_1^{(1)}, ..., x_n^{(1)}\} \in \mathbb{R}^{d_1\times n}$, and text modality as $\mathbf{X}^{(2)} = \{x_1^{(2)}, ..., x_n^{(2)}\} \in \mathbb{R}^{d_2\times n}$, respectively, where both $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$ are zero-centered.

The purpose of STCMH is to learn unified k-bit hash codes $\mathbf{b}_i \in \{-1,1\}^k$ with minimal semantic loss for different modalities of each training object $o_i (i = 1, 2, ..., n)$. In addition, two groups of modality-specific hashing functions $f_1(x^{(1)}) : \mathbb{R}^{d_1} \rightarrow \{-1,1\}^k$ and $f_2(x^{(2)}) : \mathbb{R}^{d_2} \rightarrow \{-1,1\}^k$ for image and text modalities are learned by self-taught scheme, respectively.

3.2 Learning latent semantic representation

Matrix factorization, one of the excellent method for discovering latent concept and dimensionality reduction [13], has been widely used in a large number of research fields. Following [15, 33], the common semantic features from heterogeneous data are learned by collective matrix factorization. Specifically, given the image data $\mathbf{X}^{(1)}$ and the text data $\mathbf{X}^{(2)}$, we decompose them jointly as follows:

$$\begin{aligned} J_{mf}(\mathbf{U}_1, \mathbf{U}_2, \mathbf{V}) &= \alpha||\mathbf{X}^{(1)} - \mathbf{U}_1\mathbf{V}^T||_F^2 \\ &\quad + (1-\alpha)||\mathbf{X}^{(2)} - \mathbf{U}_2\mathbf{V}^T||_F^2 \end{aligned} \tag{1}$$

where $\mathbf{U}_1 \in \mathbb{R}^{d_1\times k}$, $\mathbf{U}_2 \in \mathbb{R}^{d_2\times k}$, $\mathbf{V} \in \mathbb{R}^{n\times k}$, and k is the number of latent factors, which also equals to the binary code length. Each column $\mathbf{v}_i^T$ of the matrix $\mathbf{V}^T$ can be treated as the common semantic representations of the i-th object with two modalities. The parameter α is used for balancing the importance of image and text modalities.

3.3 Binary quantization

Different from the simple and direct thresholding strategy used in previous works which leads to large quantization error, we enforce an orthogonal transformation on the learned common semantics to obtain binary codes with minimal quantization loss.

In fact, orthogonal transformation balances the variance of the different dimensional data in $\mathbf{V}$ and satisfies the maximum variance condition. From the geometric point of view, it finds k directions again by some geometric transformations, such as rotation, so that the projection variance of the data in these k directions is maximum, that is, the k directions contain the most original information. Therefore, performing quantization in these k directions can increase the discrimination of binary codes, further reducing the quantization error. In addition, In addition, the orthogonal transformation eliminates the correlation among k bits data on original common semantic space $\mathbf{V}$ and makes each bit of generated binary code independent. It is theoretically guaranteed that the generated binary codes are more distinguishable, and the quantization loss can be further reduced.

Thus, given the latent common semantic representation $\mathbf{V}$, the quantization loss is minimized by optimizing the following formula:

$$\begin{aligned} &\min_{\mathbf{B},\mathbf{T}} J_{bq}(\mathbf{B}, \mathbf{T}) = ||\mathbf{B} - \mathbf{V}\mathbf{T}||_F^2 \\ &s.t.\ \mathbf{T}\mathbf{T}^T = \mathbf{I} \end{aligned} \tag{2}$$

where $\mathbf{B} \in \mathbb{R}^{n\times k}$ is the binary codes, and $\mathbf{T} \in \mathbb{R}^{k\times k}$ is the transformation matrix with an orthogonal constraint. Therefore, the data of same class but with uncorrelated spatial feature are encoded into similar binary codes, and semantic quantization loss is minimized correspondingly.

3.4 Mixed graph regularization term

To boost the discriminative power of binary codes, extra graph regularization constraints on the binary space $\mathbf{B}$ are added to preserve intra-modal and inter-modal similarities.

3.4.1 Intra-modal similarity preservation: We first capture neighbourhood relationship using local geometric structure for each modality and then construct k-nn graph model. We denote adjacency matrix as $\mathbf{S}^{(m)}(m = 1, 2)$ whose elements $s_{ij}^{(m)}$ between $x_i^{(m)}$ and $x_j^{(m)}$ are defined as follows.

$$s_{ij}^{(m)} = \begin{cases} 1,\ if\ x_i^{(m)} \in \mathbf{N}_k(x_j^{(m)})\ or\ x_j^{(m)} \in\ \mathbf{N}_k(x_i^{(m)}) \\ 0,\ otherwise \end{cases} \tag{3}$$

where $\mathbf{N}_k(\cdot)$ indicates the set of k-nearest neighbours.

Algorithm 1 STCMH

Input: Feature matrices $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$, class label $\overline{\mathbf{L}}$, code length k, parameters α, β, γ, and λ.
Output: Binary code matrix $\mathbf{B} \in \mathbb{R}^{n\times k}$.
1: Centralize $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$ by means.
2: Initialize $\mathbf{V}$, $\mathbf{T}$ by random matrices respectively.
3: Construct the graph Laplacian matrix $\mathbf{L}$ via (3) and (4).
4: **repeat**
5: Fix $\mathbf{U}_1$, $\mathbf{U}_2$, $\mathbf{V}$, $\mathbf{T}$, update $\mathbf{B}$ by (8).
6: Fix $\mathbf{V}$, $\mathbf{T}$, $\mathbf{B}$, update $\mathbf{U}_1$, $\mathbf{U}_2$ by (10).
7: Fix $\mathbf{U}_1$, $\mathbf{U}_2$, $\mathbf{T}$, $\mathbf{B}$, update $\mathbf{V}$ by (11).
8: Fix $\mathbf{U}_1$, $\mathbf{U}_2$, $\mathbf{V}$, $\mathbf{B}$, update $\mathbf{T}$ by (12).
9: **until** convergence
10: Compute $\mathbf{B}$ by (9).

3.4.2 Inter-modal similarity preservation: Since different modalities of the same object share closely similar semantics, the semantic correlations of different modalities are further built with label information, denoted by similarity matrix $\mathbf{S}^{(12)}$, which can be computed as:

$$s_{ij}^{(12)} = \begin{cases} 1, \ if \ x_i^{(1)} \ and \ x_j^{(2)} \ belong \ to \ the \ same \ category \\ 0, \ otherwise \end{cases} \tag{4}$$

In terms of the above definition of similarities, the mixed graph regularization term is represented as:

$$\begin{aligned} J_{mgr}(\mathbf{B}) &= \frac{1}{2}\sum_{i,j=1}^{n}(s_{ij}^{(1)} + s_{ij}^{(2)} + s_{ij}^{(12)})||\mathbf{b}_i - \mathbf{b}_j||^2 \\ &= \frac{1}{2}\sum_{i,j=1}^{n} w_{ij}||\mathbf{b}_i - \mathbf{b}_j||^2 \end{aligned} \tag{5}$$

where $w_{ij} = s_{ij}^{(1)} + s_{ij}^{(2)} + s_{ij}^{(12)}$. Equation (5) can be further rewritten as the following matrix form:

$$J_{mgr}(\mathbf{B}) = tr(\mathbf{B}^T\mathbf{L}\mathbf{B}) \tag{6}$$

where $\mathbf{L} = \mathbf{D} - \mathbf{W}$ is the Laplacian matrix. $\mathbf{W} \in \mathbb{R}^{n\times n}$ is a matrix composed of w_{ij} and $\mathbf{D} \in \mathbb{R}^{n\times n}$ is a diagonal matrix with $d_{ii} = \sum_j w_{ij}$.

3.5 Overall objective function

Combining the latent semantic representations term J_{mf} in (1), the binary quantization term J_{bq} in (2), the mixed graph regularization term J_{mgr} in (6), and a regularization term, the overall objective function of STCHM is formulated as follows:

$$\begin{aligned} &\min_{\mathbf{U}_1,\mathbf{U}_2,\mathbf{V},\mathbf{T},\mathbf{B}} J(\mathbf{U}_1,\mathbf{U}_2,\mathbf{V},\mathbf{T},\mathbf{B}) \\ &= J_{mf} + J_{bq} + J_{mgr} + \lambda R(\mathbf{U}_1,\mathbf{U}_2,\mathbf{V},\mathbf{B}) \\ &= \alpha||\mathbf{X}^{(1)} - \mathbf{U}_1\mathbf{V}^T||_F^2 + (1-\alpha)||\mathbf{X}^{(2)} - \mathbf{U}_2\mathbf{V}^T||_F^2 \\ &\quad +\beta||\mathbf{B} - \mathbf{V}\mathbf{T}||_F^2 + \gamma tr(\mathbf{B}^T\mathbf{L}\mathbf{B}) + \lambda R(\mathbf{U}_1,\mathbf{U}_2,\mathbf{V},\mathbf{B}) \\ &s.t.\ \mathbf{T}\mathbf{T}^T = \mathbf{I} \end{aligned} \tag{7}$$

where $\alpha, \beta, \gamma, \lambda$ are the tradeoff parameters and the $R(\cdot) = ||\cdot||_F^2$ is the regularization term to avoid overfitting. Specifically, the last term in (7) can be written as $R(\mathbf{U}_1,\mathbf{U}_2,\mathbf{V},\mathbf{B}) = ||\mathbf{U}_1||_F^2 + ||\mathbf{U}_2||_F^2 + ||\mathbf{V}||_F^2 + ||\mathbf{B}||_F^2$.

It is intractable to directly optimize the overall objective function in (7) due to its non-convex with respect to $\mathbf{U}_1$, $\mathbf{U}_2$, $\mathbf{V}$, $\mathbf{T}$ and $\mathbf{B}$ jointly. Hence, we adopt an iterative strategy to solve the above problem, i.e., updating each variable respectively while fixing the others, and the detailed steps are listed as follows.

(i) Update $\mathbf{B}$ by fixing $\mathbf{U}_1$, $\mathbf{U}_2$, $\mathbf{V}$, $\mathbf{T}$. Let $\frac{\partial J}{\partial \mathbf{B}} = 0$, we have:

$$\mathbf{B} = 2\beta(2(\beta+\lambda)\mathbf{I} + \gamma(\mathbf{L} + \mathbf{L}^T))^{-1}\mathbf{V}\mathbf{T} \tag{8}$$

where $\mathbf{I}$ is the identity matrix. Note that once the algorithm converges, we obtain the final binary matrix $\mathbf{B}$ as the following form:

$$\mathbf{B} = \mathrm{sgn}(2\beta(2(\beta+\lambda)\mathbf{I} + \gamma(\mathbf{L} + \mathbf{L}^T))^{-1}\mathbf{V}\mathbf{T}) \tag{9}$$

(ii) Update $\mathbf{U}_1$, $\mathbf{U}_2$ by fixing $\mathbf{V}$, $\mathbf{T}$, $\mathbf{B}$. Specifically, let $\frac{\partial J}{\partial \mathbf{U}_1} = 0$, $\frac{\partial J}{\partial \mathbf{U}_2} = 0$, then we can obtain:

$$\begin{aligned} \mathbf{U}_1 &= \mathbf{X}^{(1)}\mathbf{V}(\mathbf{V}^T\mathbf{V} + \frac{\lambda}{\alpha}\mathbf{I})^{-1} \\ \mathbf{U}_2 &= \mathbf{X}^{(2)}\mathbf{V}(\mathbf{V}^T\mathbf{V} + \frac{\lambda}{1-\alpha}\mathbf{I})^{-1} \end{aligned} \tag{10}$$

(iii) Update $\mathbf{V}$ by fixing $\mathbf{U}_1$, $\mathbf{U}_2$, $\mathbf{T}$, $\mathbf{B}$. Let $\frac{\partial J}{\partial \mathbf{V}} = 0$, we have:

$$\begin{aligned} \mathbf{V} = (\alpha{\mathbf{X}^{(1)}}^T\mathbf{U}_1 + (1-\alpha){\mathbf{X}^{(2)}}^T\mathbf{U}_2 + \beta\mathbf{B}\mathbf{T}^T) \\ (\alpha{\mathbf{U}_1}^T\mathbf{U}_1 + (1-\alpha){\mathbf{U}_2}^T\mathbf{U}_2 + (\beta+\lambda)\mathbf{I})^{-1} \end{aligned} \tag{11}$$

(iv) Update $\mathbf{T}$ by fixing $\mathbf{U}_1$, $\mathbf{U}_2$, $\mathbf{V}$, $\mathbf{B}$. It essentially is a classic Orthogonal Procrustes problem, which can be solved by singular value decomposition (SVD). The matrix $\mathbf{B}^T\mathbf{V}$ is decomposed into $\mathbf{W}_1\mathbf{\Sigma}{\mathbf{W}_2}^T$ using SVD, and then the matrix $\mathbf{T}$ can be computed by the following formula:

$$\mathbf{T} = \mathbf{W}_2\mathbf{W}_1^T \tag{12}$$

More detailed derivation of Eq. (12) are given in the Appendix.

The whole procedure of our proposed STCMH is summarized in Algorithm 1.

Algorithm 2 Out-of-Sample Extension

Input: Feature matrices $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$, the new query $x_q^{(m)}$.
Output: The hashing function $f_m(x^{(m)})$ and binary codes $\mathbf{b}_q$ for the new query.
1: Generate binary code matrix $\mathbf{B} \in \mathbb{R}^{n\times k}$ by Algorithm 1.
2: **for** $m = 1$ to 2 **do**
3: **for** $l = 1$ to k **do**
4: Obtain l-th bit linear SVM model $f_l^{(m)}$ by self-taught scheme.
5: **end for**
6: **end for**
7: Integrate the above learned model $f_l^{(m)}$ into the modality-specific hash function $f_{(m)}(x^{(m)})$ by (13), $m = 1, 2$.
8: Generate binary codes $\mathbf{b}_q$ for the query $x_q^{(m)}$ according to the hash function.

3.6 Out-of-sample extension

Motivated by self-taught scheme [24], we adopt a direct way to learn k-bit binary codes for out-of-sample data, which differs from traditional methods. Specifically, the hashing functions learning is considered as a binary classification problem.

The k binary classifiers for each modality are trained by the linear Support Vector Machine (SVM), which can be easily extended to non-linear case. In particular, the original feature $\mathbf{X}^{(m)}(m = 1, 2)$

and each bit of the binary code $\mathbf{b}_l \in \{-1, 1\}^n$ are treated as input and class label, respectively, where the vector $\mathbf{b}_l (l = 1, ..., k)$ represents each column of the matrix $\mathbf{B}$. The corresponding l-th bit linear SVM model $f_l^{(m)}$ can be trained by finding the hyperplane of maximum margin.

Integrating each of the above learned SVM model, the hashing functions for out-of-sample data can be obtained. Therefore, the modality-specific hashing functions $f_m(x^{(m)})(m = 1, 2)$ can be represented as follows.

$$f_m(x^{(m)}) = \{f_1^{(m)}(x^{(m)}), f_2^{(m)}(x^{(m)}), ..., f_k^{(m)}(x^{(m)})\} \quad (13)$$

For a new query $x_q^{(m)}$, its k bits binary codes can be generated according to (13), i.e. $\mathbf{b}_q = f_m(x_q^{(m)})$.

In summary, Algorithm 2 gives the self-taught hashing functions and the generation procedure of binary codes for out-of-sample data.

3.7 Complexity Analysis

The computational cost of training STCMH is mainly related to binary code generation in Algorithm 1 and hash function learning in Algorithm 2. In Algorithm 1, constructing the graph Laplacian matrix firstly takes $O(dn^2)$, where $d = \max\{d_1, d_2\}$. Secondly, in each of the subsequent iteration, the time cost of solving (8) is $O(n^3 + kn^2)$; solving (10) is $O((dk + k^2)n + k^3)$; $O((kd + k^2)n + dk^2 + k^3)$ and $O(k^3)$ for solving (11) and (12), respectively. In Algorithm 2, each linear SVM model is trained in $O(dn)$ time or less, so $O(dkn)$ is required for k SVM models. Therefore, the overall training complexity is $O(dn^2 + t(n^3 + kn^2 + (dk + k^2)n + dk^2 + k^3) + kdn)$, where t represents the number of iterations. In the search phase, encoding a query requires only k dot-product operation, thus the query time cost is very low.

Compared with the training time complexity of unsupervised baseline methods, such as CVH and CMFH, which are at least $O(n^2)$, our method involves the computation of matrix inversion due to mixed graph regularization term, resulting in approximately $O(n^3)$. In addition, the time complexity for training process of unsupervised method LSSH is $O(n)$, which is available for large-scale data. However, its overall performance is poorer than most of the supervised methods because it ignores the label information. For supervised baseline methods, SCM_Orth and SCM_Seq reconstruct the similarity matrix to avoid high complexity and obtain a linear time complexity $O(n)$, but their performances are inferior and even lower than some unsupervised methods on Wiki and Pascal VOC 2007 datasets. In terms of the supervised method SMFH, the time complexity scales $O(n^2)$.

Although the training time complexity of our method is higher than others, considering its great advantages in performance, our STCMH can be competitive with the compared methods.

3.8 Extension to multi-modalities

As mentioned above, the proposed STCMH method can be easily extended to the multi-modal scenario. The main idea of multi-modal scenario is similar to that of two modalities, which maps more than two modalities to a common space. Thus, multiple modalities can be retrieved from each other. Therefore, suppose that the training data consists of n objects with M modalities, which is denoted as $\mathbf{X}^{(m)}(m = 1, 2, ..., M)$, the objective function of STCMH for multi-modalities can be formulated as:

$$\begin{aligned}\min_{\mathbf{U}_m\mathbf{V},\mathbf{T},\mathbf{B}} & \sum_{m=1}^{M} \alpha_m ||\mathbf{X}^{(m)} - \mathbf{U}_m\mathbf{V}^T||_F^2 + \beta||\mathbf{B} - \mathbf{V}\mathbf{T}||_F^2 \\ & + \gamma tr(\mathbf{B}^T\mathbf{L}\mathbf{B}) + \lambda R(\sum_{m=1}^{M} \mathbf{U}_m, \mathbf{V}, \mathbf{B}) \\ s.t.\ & \mathbf{T}\mathbf{T}^T = \mathbf{I}\end{aligned}$$

where $\{\mathbf{U}_m\}_{m=1}^M$ is the decomposition factor for each modality, and $\sum_{m=1}^{M} \alpha_m = 1$. The third term of the objective function introduces more similarity measurement for multiple modalities, which also mixes intra-modal similarity $\mathbf{S}^{(m)}$ and inter-modal similarity $\mathbf{S}^{(mj)}$. The graph Laplacian matrix $\mathbf{L}$ is computed based on $\mathbf{W} = \sum_{m=1}^{M} \mathbf{S}^{(m)} + \sum_{m=1}^{M} \sum_{j>m}^{M} \mathbf{S}^{(mj)}$.

In order to learn the hashing functions for out-of-sample multimodal data, we just need to train several SVM models for each modality with the same approach in Algorithm 2.

4 Experiments

In this section, the experiments and results are presented to evaluate the effectiveness of our method in cross-modal retrieval. We compare our STCMH with six state-of-the-art methods on three benchmark datasets. The experiment results show that STCMH can significantly outperform several baseline methods.

4.1 Experiment setting

4.1.1 Datasets: To evaluate the performance of the proposed STCMH, our experiment chooses three popular datasets, including Wiki [34], Pascal VOC 2007 [35], and NUS-WIDE [36]. The details of each dataset are listed as follows.

Wiki consists of 2866 documents which were gathered from Wikipedia, each with 128-dimensional SIFT feature for image and 10-dimensional topic vector for text. It is grouped into 10 categories and each instance is annotated with one of them. We randomly extract 2173 samples as the training set and the rest 693 as the testing set.

Pascal VOC 2007 contains 5011 training and 4952 testing image-tag pairs, which were downloaded from Flickr. The images with only one label are used in our experiments. Hence, a new dataset contains 2808 training items and 2841 testing items in 20 different categories are generated. The image is described by 512-dimensional GIST features, and the text is depicted with 399-dimensional word frequency vectors.

NUS-WIDE is also a real data set crawled from Flickr, and it contains a total of 269648 instances with 81 categories. Following [14, 37, 38], the experimental data are chosen from the largest 10 categories, including 186577 images. Each image is represented by a 500-dimensional Bag-of-Visual-Words SIFT feature, and each text is represented by a 1000-dimensional feature. Here we randomly select 5000 image-text pairs for training, and use 1866 image-text pairs selected from the remaining samples for testing.

4.1.2 Baseline methods: Two types of cross-modal retrieval tasks were evaluated in the experiment, termed 'Txt to Img' and 'Img to Txt', respectively. In both tasks, we compare the proposed STCMH with six state-of-the-art cross-modal hashing methods, including CVH [11], SCM_orth [32], SCM_Seq [32], CMFH [15], LSSH [14], and SMFH [17]. Generally, they can be divided into two groups. CVH, CMFH, LSSH are unsupervised methods, and the rest are supervised ones. Source codes of most approaches are available publicly.

4.1.3 Evaluation metric: We study three types of evaluation metric on all datasets, i.e., mean average precision (mAP), the precision-recall curve and the topN-precision curve.

The mAP is the mean of the average precision, which widely used for evaluating the performance for retrieval tasks. Given N query samples, the mAP is computed by:

$$mAP = \frac{1}{N} \sum_{i=1}^{N} AP(q_i)$$

Table 1 mAP results on three datasets. The best result is shown in boldface.

Task	Method	Wiki				Pascal VOC 2007				NUS-WIDE			
		16bits	32bits	64bits	128bits	16bits	32bits	64bits	128bits	16bits	32bits	64bits	128bits
Img to Txt	CVH	0.1855	0.1586	0.1648	0.1616	0.1521	0.1636	0.1637	0.1559	0.4188	0.4059	0.3943	0.3817
	SCM_Orth	0.1532	0.1393	0.1297	0.1273	0.1565	0.1362	0.1287	0.1232	0.4051	0.3844	0.3713	0.3611
	SCM_Seq	0.2341	0.2410	0.2445	0.2554	0.2554	0.3253	0.2451	0.3388	**0.5246**	**0.5390**	0.5453	0.5444
	CMFH	0.2133	0.2307	0.2366	0.2426	0.2114	0.2269	0.2373	0.2433	0.3912	0.3962	0.3962	0.3910
	LSSH	0.2167	0.2180	0.2265	0.2211	0.2541	0.2671	0.2774	0.2787	0.4653	0.4729	0.4784	0.4927
	SMFH	0.2723	0.2839	0.2913	0.2989	0.2240	0.2436	0.2620	0.2728	0.4534	0.4553	0.4600	0.4549
	STCMH	**0.3147**	**0.3305**	**0.3394**	**0.3450**	**0.3408**	**0.3886**	**0.4069**	**0.4150**	0.4927	0.5316	**0.5676**	**0.5896**
Txt to Img	CVH	0.2228	0.1847	0.1572	0.1881	0.1815	0.2090	0.2346	0.2531	0.4145	0.4046	0.4092	0.4004
	SCM_Orth	0.1527	0.1331	0.1216	0.1172	0.1982	0.1484	0.1197	0.1006	0.4131	0.3902	0.3754	0.3675
	SCM_Seq	0.2257	0.2459	0.2461	0.2510	0.2989	0.4108	0.2652	0.4531	0.5333	0.5540	0.5678	0.5690
	CMFH	0.4909	0.5198	0.5337	0.5441	0.6073	0.6923	0.6790	0.6617	0.3909	0.3952	0.3973	0.3959
	LSSH	0.4974	0.5204	0.5318	0.5387	0.5416	0.6030	0.6177	0.6307	0.5600	0.5771	0.6047	0.6108
	SMFH	0.6164	0.6281	0.6385	0.6412	0.6307	0.7493	0.7791	0.7765	0.4663	0.4678	0.4754	0.4684
	STCMH	**0.7148**	**0.7272**	**0.7375**	**0.7434**	**0.8329**	**0.9176**	**0.9296**	**0.9326**	**0.6656**	**0.6963**	**0.7156**	**0.7304**

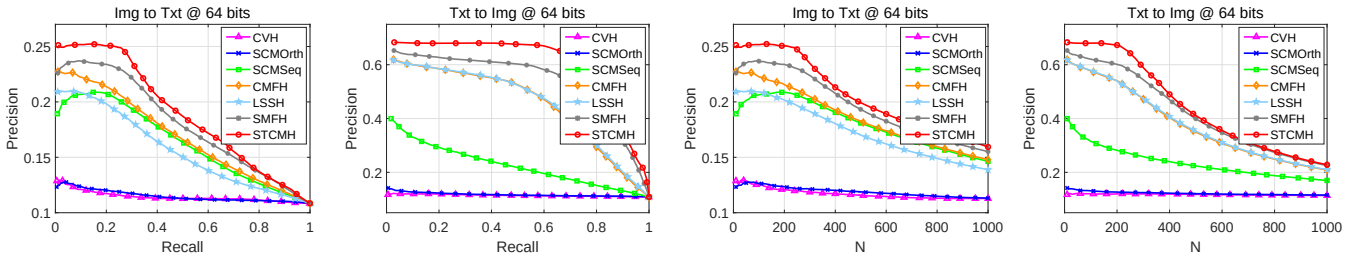


Fig. 2: precision-recall curves and topN-precision curves with 64 bits code length on both tasks of Wiki.

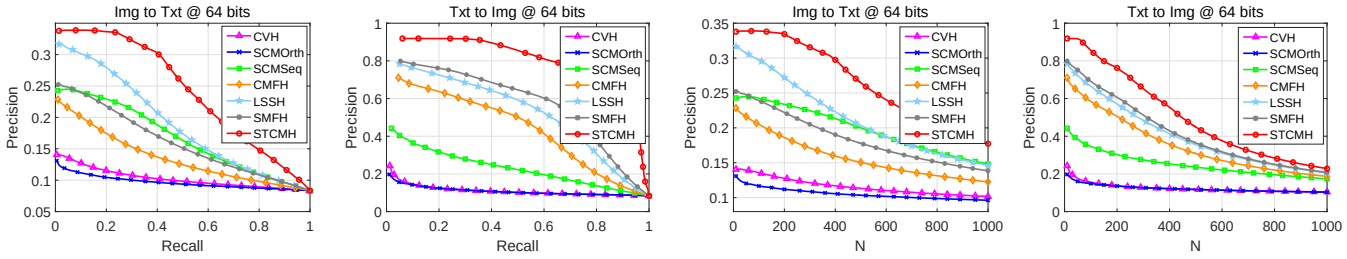


Fig. 3: precision-recall curves and topN-precision curves with 64 bits code length on both tasks of Pascal VOC 2007.

where q_i represents a query. $AP(\cdot)$ is the average retrieved precision, which is defined as:

$$AP(q_i) = \frac{1}{G}\sum_{r=1}^{R} P_{q_i}(r)\delta(r)$$

where G denotes the number of instances related to i-th query q_i in top R retrieved set, and $P_{q_i}(r)$ represents the precision of top r retrieved instances. The value of indicator function $\delta(r)$ is 1 if the query q_i is related to r-th retrieval instances, 0 otherwise.

Additionally, the precision-recall and topN-precision curves are studied to show the overall trend of variation. The precision-recall curve reflects the variation of precision as the recall increases. The topN-precision curve reflects the precision at different numbers of retrieved instances, which plays a significant role in large-scale searching for focusing on a small number of results. To avoid random effects, the results for all methods are averaged over ten runs.

4.2 *Experiment results*

4.2.1 Results on Wiki: In terms of the mAP results reported in Table 1, it is easy to observe that STCMH outperforms all baselines with different hash bits, which demonstrates the superiority of our method. In addition, with the code length increasing, the performance of some approaches degrades to some extent, such as CVH and SCM_Orth. However, our method still yields better mAP results with longer codes.

In Fig. 2, we also plot the precision-recall and topN-precision curves on both tasks when code length is 64 bits. We can find that STCMH outperforms other methods consistently. Note that on the last case, the precision of SMFH is close to our method at the final stage. However, our method performs much better at the beginning stage, which is of great importance to retrieval tasks. Furthermore, some methods such as CVH and SCM_Orth consistently behave inferior.

4.2.2 Results on Pascal VOC 2007: The mAP values of our STCMH and compared approaches on both tasks are shown in Table 1. Similar to the performance on Wiki, STCMH significantly outperforms all baseline methods and achieves remarkable mAP results. Specifically, compared with all other methods, the mAP value of our method has been improved by more than 6% and 15% for 'Img to Txt' and 'Txt to Img' tasks, respectively, which demonstrates the importance of minimizing semantic coding loss. Moreover, the mAP value of 'Txt to Img' task is much higher than that of 'Img to Txt' task at all hash code lengths, indicating that the text is better than the image to describe the semantic information.

Fig. 3 presents the precision-recall curves and the topN-precision curves on Pascal VOC 2007 when code length is 64 bits, which shows the advantage of our STCMH. It can be noticed that the unsupervised approach LSSH performs comparably to or even better than

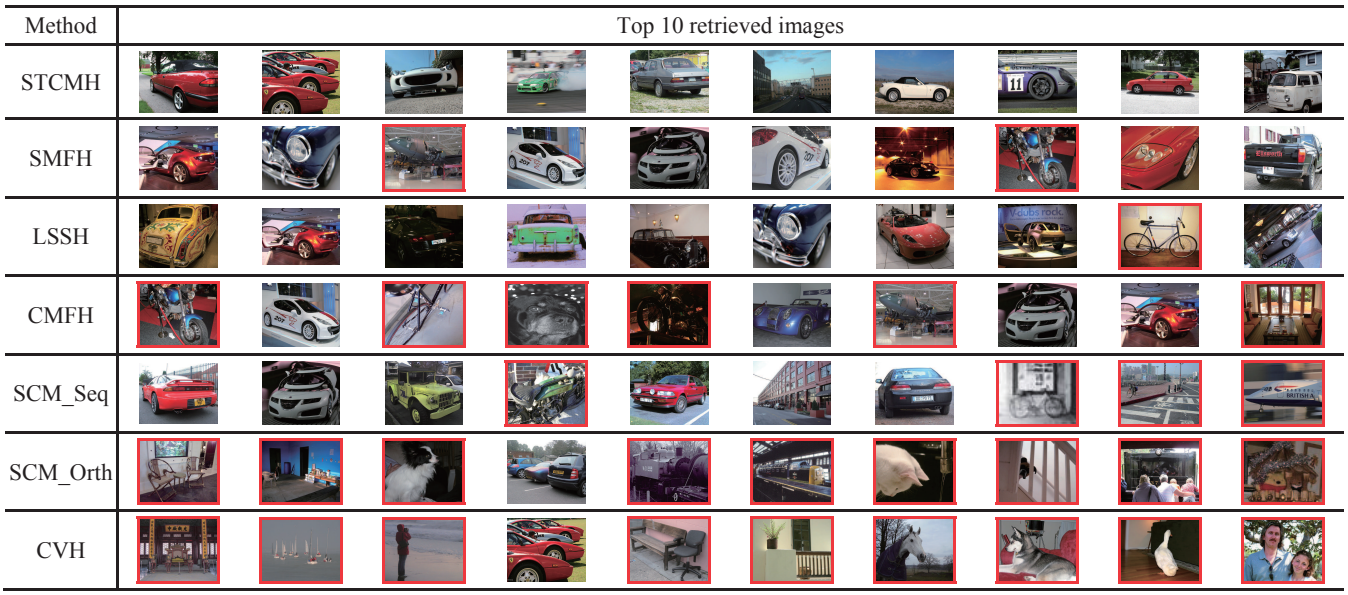


Fig. 4: An example of 'Txt to Img' retrieval task by querying the text 'car' on the Pascal VOC 2007, The top 10 retrieved results of different hashing methods are illustrated in the second column. The incorrect retrievals are marked with red rectangle.

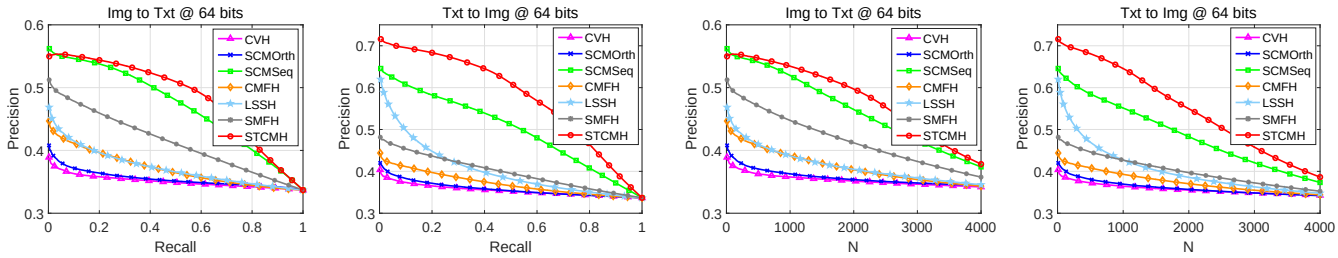


Fig. 5: precision-recall curves and topN-precision curves with 64 bits code length on both tasks of NUS-WIDE.

the supervised method SMFH. However, the proposed approach still achieves outstanding performance.

Fig. 4 illustrates an example of the top 10 retrieved images by different hashing methods for text queries of 'car'. Obviously, the proposed method achieves more reasonable results for text query than other methods.

4.2.3 Results on NUS-WIDE: The mAP results of different approaches on NUS-WIDE dataset are also listed in Table 1. Slightly different from the results on the above two datasets, STCMH achieves superior performance to most of the baseline methods except SCM_seq. On the task of 'Img to Txt', SCM_Seq obtains better results than STCMH with relatively short code length, such as 16 and 32. The reason may be that the unified and short code length constraints for different modalities are too strict for NUS-WIDE dataset, which limits the performance improvement. With the code length increasing from 32 to 128 bits, the mAP values of STCMH improve gradually and even better than SCM_Seq when hash code length is 64 and 128 bits. In terms of the 'Txt to Img' task, STCMH achieves the highest score among all methods obviously.

Correspondingly, Fig. 5 also shows the precision-recall and topN-precision curves on NUS-WIDE when code length is 64 bits. It can be observed that our STCMH outperforms significantly other approaches on 'Txt to Img' task, while it is slightly lower than SCM_Seq at the beginning of the 'Img to Txt' task. However, SCM_Seq performs poorly on wiki and Pascal VOC 2007 datasets. Hence, it cannot achieve stable performance on different datasets, while our method is relatively more stable.

Taking into account the great advantage of the proposed STCMH on the three datasets, we can conclude that STCMH is effective for cross-modal retrieval.

4.3 Effect of orthogonal transformation and graph regularization

In our method, the discriminative binary codes are generated using quantization procedure based on orthogonal transformation, while incorporating the mixed graph regularization term simultaneously. To evaluate the advantages of both, we present two variations of STCMH approach. One performs the binary quantization process without orthogonal constraint, namely STCMH_noT. The other does not build the graph regularization, namely STCMH_noM. We conduct the experiments for them respectively. Fig. 6 reports the mAP comparison results of LSSH, STCMH_noM, STCMH_noT and STCMH for both 'Img to Txt' and 'Txt to Img' tasks on Wiki dataset at the code length of 16, 32 and 64 bits. Comparing STCMH_noM with STCMH_noT, we can see that the latter produces better results in both tasks, which implies that the supervised method STCMH_noT with mixed regular terms has a greater impact. Both of them are superior to the unsupervised baseline method LSSH, indicating that both orthogonal transform and mixed graph regularization term help improve the performance of cross-modal retrieval. Notably, combining the advantages of both, our method STCMH obtains the highest mAP score.

4.4 Effect of classifiers

To evaluate the effect of different classifiers on retrieval performance, we conduct experiments using the proposed method with five classical classifiers on Wiki dataset. Specifically, the classifiers included in this experiment are Linear SVM, logistic regression (LR), random forest (RF), k-Nearest Neighbours (KNN), and

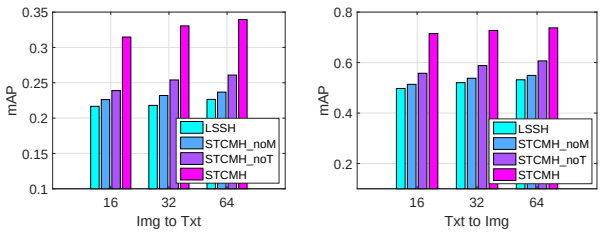


Fig. 6: mAP comparison results of four methods for both 'Img to Txt' and 'Txt to Img' tasks on Wiki dataset.

Adaboost (Ada). Our STCMH using various classifiers are denoted as the default STCMH (SVM), STCMH-LR, STCMH-RF, STCMH-KNN, and STCMH-Ada for convenience. The mAP values of different methods on Wiki dataset are illustrated in Table 2.

From Table 2, we can observe that the mAP results of our method with different classifiers are relatively close in most cases. The results also demonstrate that the selection of classifier for our method slightly affect the performance but not much. Because the kernel contribution of our method is improving the discriminative power of the unified binary codes, the selection of classifier is not the focus of this work. We also note that the proposed STCMH with different classifiers can still outperform the baseline methods, which shows the proposed STCMH is able to incorporate different classifiers and can achieve superior performance. It also validates the kernel contribution of our work.

4.5 Parameter sensitivity

STCMH has four essential parameters in the overall objective function, i.e., α , β, γ, and λ. In our previous experiments, we empirically set $\alpha = 0.5$, $\beta = 0.01$, $\gamma = 1$, $\lambda = 0.001$. Here we study the effect of different parameter settings on algorithm performance and conduct experiments for both tasks on Wiki dataset. The results on Pascal VOC 2007 and NUS-WIDE are similar to that on Wiki dataset. Specifically, we vary one parameter while keeping others unchanged in the case of 64 bits. The mAP score variations of STCMH with the different values of four parameters on Wiki dataset are plotted in Fig. 7. It can be observed that our STCMH is not sensitive to all the parameters, which validates that STCMH can achieve satisfactory score over the wide range of parameter values.

4.6 Convergence study

Since an iterative manner is adopted to solve the proposed STCMH in Algorithm 1, we study the convergence of the iterative algorithm with 64 bits in this subsection. Fig. 8 gives the convergence curves on three datasets. As can be seen, our STCMH converges fast on all datasets. More specifically, our STCMH converges within 10 iterations on Pascal VOC 2007 and NUS-WIDE. The trend of other hash bits is similar to that of 64 bits. Therefore, our STCMH can obtain excellent retrieval performance with efficient training time.

Table 2 mAP values of STCMH using different classifiers on Wiki.

classifier	Txt to Img		Img to Txt	
	16bits	32bits	16bits	32bits
STCMH	0.7148	0.7272	0.3147	0.3305
STCMH-LR	0.6945	0.6950	0.3140	0.3162
STCMH-RF	0.7373	0.7473	0.3220	0.3436
STCMH-KNN	0.7244	0.7305	0.2849	0.2916
STCMH-Ada	0.7131	0.7312	0.3057	0.3281

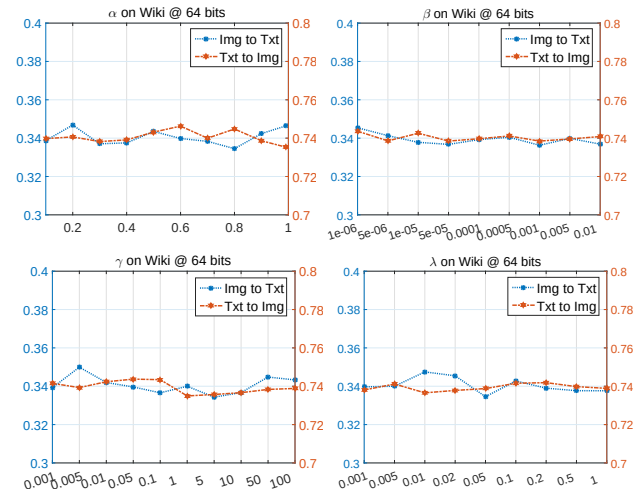
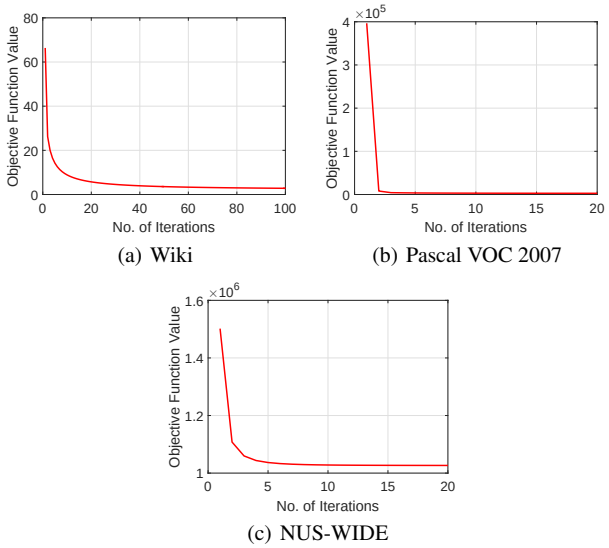


Fig. 7: Parameter Sensitivity Analysis on Wiki.



(a) Wiki (b) Pascal VOC 2007

(c) NUS-WIDE

Fig. 8: Convergence curves of the objective function on three datasets.

5 Conclusion

In this paper, we have proposed a Self-Taught Cross-Modal Hashing approach for cross-modal retrieval, aiming at minimizing the semantic loss of binary codes. Specifically, we adopt the collective matrix factorization to learn the latent common semantic feature, while minimizing the quantization loss by rotating learned semantic space simultaneously. To further reduce the quantization error, we consider the binary codes learning for query samples as a binary classification problem. As a result, STCMH can directly generate the binary hash codes for unseen data with minimal semantic loss. Experimental results on three datasets have demonstrated the excellent performance of STCMH over six baseline approaches for cross-modal retrieval.

6 Acknowledgments

This work is supported by the National Key Research and Development Program of China under Grant 2018YFC0831305.

7 References

1 Rasiwasia, N., Pereira, J.C., Coviello, E., et al.: 'A new approach to cross-modal multimedia retrieval'. Proc. Int. Conf. Multimedia, Firenze, Italy, October 2010, pp. 251–260

2 Xu, X., Yang, Y., Shimada, A., Taniguchi, R., He, L.: 'Semi-supervised Coupled Dictionary Learning for Cross-modal Retrieval in Internet Images and Texts'. Proc. Annu. ACM Conf. Multimedia, Brisbane, Australia, October 2015, pp. 847–850

3 Feng, F., Wang, X., Li, R.: 'Cross-modal Retrieval with Correspondence Autoencoder'. Proc. ACM Int. Conf. Multimedia, Orlando, FL, USA, November 2014, pp. 7–16

4 Wu, F., Yu, Z., Yang, Y., Tang, S., Zhang, Y., Zhuang, Y.: 'Sparse Multi-Modal Hashing', *IEEE Trans. Multimedia*, 2014, **16**, pp. 427–439

5 Wang, J., Zhang, T., song, J., Sebe, N., Shen, H.T.: 'A Survey on Learning to Hash', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2018, **40**, pp. 769–790

6 Liu, L., Lin, Z., Shao, L., Shen, F., Ding, G., Han, J.: 'Sequential Discrete Hashing for Scalable Cross-Modality Similarity Retrieval', *IEEE Trans. Image Processing*, 2017, **26**, pp. 107–118

7 Grauman, K., Fergus, R.: 'Learning Binary Hash Codes for Large-Scale Image Search', *Machine Learning for Computer Vision*, 2013, pp. 49–87

8 Han, J., Zhang, D., Wen, S., Guo, L., Liu, T., Li, X.: 'Two-Stage Learning to Predict Human Eye Fixations via SDAEs', *IEEE Trans. Cybernetics*, 2016, **46**, pp. 487–498

9 Tang, J., Li, Z., Zhu, X.: 'Supervised deep hashing for scalable face image retrieval', *Pattern Recogn.*, 2018, **75**, pp. 25–32

10 Song, J., Gao, L., Liu, L., Zhu, X., Sebe, N.: 'Quantization-based hashing: a general framework for scalable image and video retrieval', *Pattern Recogn.*, 2018, **75**, pp. 175–187

11 Kumar, S., Udupa, R.: 'Learning hashing functions for cross-view similarity search'. Proc. Int. Joint Conf. Artif. Intell., Barcelona, Catalonia, Spain, July 2011, pp. 1360–1365

12 Zhen, Y., Yeung, D.-Y.: 'Co-Regularized Hashing for Multimodal Data'. Proc. Annu. Conf. Neural Inf. Process. Syst., Lake Tahoe, Nevada, United States, December 2012, pp. 1385–1393

13 Rastegari, M., Choi, J., Fakhraei, S., III, H.D., Davis, L.S.: 'Predictable Dual-View Hashing'. Proc. Int. Conf. Machine Learning, Atlanta, GA, USA, June 2013, pp. 1328–1336

14 Zhou, J., Ding, G., Guo, Y.: 'Latent semantic sparse hashing for cross-modal similarity search'. Proc. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr., Gold Coast , QLD, Australia, July 2014, pp. 415–424

15 Ding, G., Guo, Y., Zhou, J.: 'Collective Matrix Factorization Hashing for Multimodal Data'. Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Columbus, OH, USA, June 2014, pp. 2083–2090

16 Xie, L., Zhu, L., Chen, G.: 'Unsupervised multi-graph cross-modal hashing for large-scale multimedia retrieval', *Multimedia Tools Appl.*, 2016, **75**, pp. 9185–9204

17 Tang, J., Wang, K., Shao, L.: 'Supervised Matrix Factorization Hashing for Cross-Modal Retrieval', *IEEE Trans. Image Processing*, 2016, **25**, pp. 3157–3166

18 Wang, K., Tang, J., Wang, N., Shao, L.: 'Semantic Boosting Cross-Modal Hashing for efficient multimedia retrieval', *Inf. Sci.*, 2016, **330**, pp. 199–210

19 Yao, T., Kong, X., Fu, H., Tian, Q.: 'Semantic consistency hashing for cross-modal retrieval', *Neurocomputing*, 2016, **193**, pp. 250–259

20 Shen, F., Shen, C., Liu, W., Shen, H.T.: 'Supervised Discrete Hashing'. Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Boston, MA, USA, June 2015, pp. 37–45

21 Weiss, Y., Torralba, A., Fergus, R.: 'Spectral Hashing'. Proc. Adv. Neural Inf. Process. Syst., Vancouver, British Columbia, Canada, December 2008, pp. 1753–1760

22 Song, D., Liu, W., Ji, R., Meyer, D.A., Smith, J.R.: 'Top Rank Supervised Binary Coding for Visual Search'. Proc. IEEE Int. Conf. Computer Vision, Santiago, Chile, December 2015, pp. 1922–1930

23 Wang, J., Wang, J., Yu, N., Li, S.: 'Order preserving hashing for approximate nearest neighbor search'. Proc. ACM Int. Conf. Multimedia, Barcelona, Spain, October 2013, pp. 133–142

24 Zhang, D., Wang, J., Cai, D., Lu, J.: 'Self-taught hashing for fast similarity search'. Proc. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr., Geneva, Switzerland, July 2010, pp. 18–25

25 Song, J., Yang, Y., Yang, Y., Huang, Z., Shen, H.T.: 'Inter-media hashing for large-scale retrieval from heterogeneous data sources'. Proc. ACM SIGMOD Int. Conf. Manage. Data, New York, NY, USA, June 2013, pp. 785–796

26 Zhu, X., Huang, Z., Shen, H.T., Zhao, X.: 'Linear cross-modal hashing for efficient multimedia search'. Proc. ACM Int. Conf. Multimedia, Barcelona, Spain, October 2013, pp. 143–152

27 Rafailidis, D., Crestani, F.: 'Cluster-based Joint Matrix Factorization Hashing for Cross-Modal Retrieval'. Proc. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr., Pisa, Italy, July 2016, pp. 781–784

28 Bronstein, M.M., Bronstein, A.M., Michel, F., Paragios, N.: 'Data fusion through cross-modality metric learning using similarity-sensitive hashing'. Proc. IEEE Conf. Comput. Vis. Pattern Recognit., San Francisco, CA, USA, June 2010, pp. 3594–3601

29 Masci, J., Bronstein, M.M., Bronstein, A.M., Schmidhuber, J.: 'Multimodal Similarity-Preserving Hashing', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2014, **36**, pp. 824–830

30 Zhen, Y., Yeung, D.-Y.: 'A probabilistic model for multimodal hash function learning'. Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, Beijing, China, August 2012, pp. 940–948

31 Wu, B., Yang, Q., Zheng, W.-S., Wang, Y., Wang, J.: 'Quantized Correlation Hashing for Fast Cross-Modal Search'. Proc. Int. Joint Conf. Artif. Intell., Buenos Aires, Argentina, July 2015, pp. 3946–3952

32 Zhang, D., Li, W.-J.: 'Large-Scale Supervised Multimodal Hashing with Semantic Correlation Maximization'. Proc. AAAI Conf. Artif. Intell., July 2014, Québec City, Québec, Canada, pp. 2177–2183

33 Bouchard, G., Yin, D., Guo, S.: 'Convex Collective Matrix Factorization'. Proc. Int. Conf. Artif. Intell. Statist., Scottsdale, AZ, USA, April-May, 2013, pp. 144–152

34 Pereira, J.C., Coviello, E., Doyle, G., et al.: 'On the Role of Correlation and Abstraction in Cross-Modal Multimedia Retrieval', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2014, **36**, pp. 521–535

35 Everingham, M., Gool, L.J.V., Williams, C.K.I., Winn, J.M., Zisserman, A.: 'The Pascal Visual Object Classes (VOC) Challenge', *Int. J. Comput. Vision*, 2010, **88**, pp. 303–338

36 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'. Proc. ACM Int. Conf. Image Video Retrieval, Santorini Island, Greece, July 2009, pp. 48

37 Lin, Z., Ding, G., Hu, M., Wang, J.: 'Semantics-preserving hashing for cross-view retrieval'. Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Boston, MA, USA, June 2015, pp. 3864–3872

38 Yu, Z., Wu, F., Yang, Y., Tian, Q., Luo, J., Zhuang, Y.: 'Discriminative coupled dictionary hashing for fast cross-media retrieval'. Proc. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr., Gold Coast , QLD, Australia, July 2014, pp. 395–404

8 APPENDIX

8.1 Derivation of Eq. (12)

In step (iv) of Section 3.5, the matrix $\mathbf{T}$ is calculated by (12). Here, we give the derivation process of Eq. (12).

Firstly, since the matrices $\mathbf{U}_1$, $\mathbf{U}_2$, $\mathbf{V}$, $\mathbf{B}$ are fixed, the overall function to solve can be simplified as:

$$\begin{aligned} &\min_{\mathbf{T}} \beta||\mathbf{B}-\mathbf{V}\mathbf{T}||_F^2 \\ &s.t.\ \mathbf{T}\mathbf{T}^T=\mathbf{I} \end{aligned} \tag{14}$$

The Eq. (14), which is the form of classic Orthogonal Procrustes problem, can be further derived to obtain the following formula:

$$\begin{aligned} ||\mathbf{B}-\mathbf{V}\mathbf{T}||_F^2 &= tr((\mathbf{B}-\mathbf{V}\mathbf{T})(\mathbf{B}-\mathbf{V}\mathbf{T})^T) \\ &= tr((\mathbf{B}-\mathbf{V}\mathbf{T})(\mathbf{B}^T-\mathbf{T}^T\mathbf{V}^T)) \\ &= tr(\mathbf{B}\mathbf{B}^T)-2tr(\mathbf{B}^T\mathbf{V}\mathbf{T})+tr(\mathbf{V}\mathbf{V}^T) \end{aligned} \tag{15}$$

Since both $\mathbf{B}$ and $\mathbf{V}$ are fixed, Eq. (14) is equivalent to maximizing $tr(\mathbf{B}^T\mathbf{V}\mathbf{T})$.

Secondly, $\mathbf{B}^T\mathbf{V}$ is decomposed into $\mathbf{W}_1\mathbf{\Sigma}\mathbf{W}_2^T$ in terms of SVD, where both $\mathbf{W}_1 \in \mathbb{R}^{k\times k}$ and $\mathbf{W}_2 \in \mathbb{R}^{k\times k}$ are orthogonal matrices, and $\mathbf{\Sigma} = diag(\sigma_1, ..., \sigma_q)$. Thus, we have:

$$\begin{aligned} \mathrm{tr}(\mathbf{B}^T\mathbf{V}\mathbf{T}) &= tr(\mathbf{W}_1\mathbf{\Sigma}\mathbf{W}_2^T\mathbf{T}) \\ &= tr(\mathbf{W}_2^T\mathbf{T}\mathbf{W}_1\mathbf{\Sigma}) = tr(\mathbf{P}\mathbf{\Sigma}) \\ &= \sum_{i=1}^{q} p_{ii}\sigma_i \le \sum_{i=1}^{q}\sigma_i = tr(\mathbf{\Sigma}) \end{aligned} \tag{16}$$

where $\mathbf{P} = \mathbf{W}_2^T\mathbf{T}\mathbf{W}$ is a orthogonal matrix. To maximize $tr(\mathbf{B}^T\mathbf{V}\mathbf{T})$, let $\mathbf{P} = \mathbf{I}$, therefore we have:

$$\mathbf{W}_2^T\mathbf{T}\mathbf{W}_1 = \mathbf{I} \tag{17}$$

Next, since it has been previously obtained that both $\mathbf{W}_1$ and $\mathbf{W}_2$ are orthogonal matrices, it implies the following formulas.

$$\begin{aligned} \mathbf{W}_1\mathbf{W}_1^T &= \mathbf{I} \\ \mathbf{W}_2\mathbf{W}_2^T &= \mathbf{I} \end{aligned} \tag{18}$$

Finally, combining Eq. (17) and Eq. (18), we can further derive to get $\mathbf{T} = \mathbf{W}_2\mathbf{W}_1^T$, which is also the Eq. (12) mentioned earlier.