International Journal of Multimedia Information Retrieval (2019) 8:47–59
https://doi.org/10.1007/s13735-018-0164-0

REGULAR PAPER

Multi-view collective tensor decomposition for cross-modal hashing

Limeng Cui[1] · Jiawei Zhang[2] · Lifang He[3] · Philip S. Yu[4]

Received: 1 September 2018 / Revised: 6 December 2018 / Accepted: 12 December 2018 / Published online: 1 January 2019

Abstract
With the development of social media, data often come from a variety of sources in different modalities. These data contain complementary information that can be used to produce better learning algorithms. Such data exhibit dual heterogeneity: On the one hand, data obtained from multiple modalities are intrinsically different; on the other hand, features obtained from different disciplines are usually heterogeneous. Existing methods often consider the first facet while ignoring the second. Thus, in this paper, we propose a novel multi-view cross-modal hashing method named Multi-view Collective Tensor Decomposition (MCTD) to mitigate the dual heterogeneity at the same time, which can fully exploit the multimodal multi-view feature while simultaneously discovering multiple separated subspaces by leveraging the data categories as supervision information. We propose a novel cross-modal retrieval framework which consists of three components: (1) two tensors which model the multi-view features from different modalities in order to get better representation of the complementary features and a latent representation space; (2) a block-diagonal loss which is used to explicitly enforce a more discriminative latent space by leveraging supervision information; and (3) two feature projection matrices which characterize the data and generate the latent representation for incoming new queries. We use an iterative updating optimization algorithm to solve the objective function designed for MCTD. Extensive experiments prove the effectiveness of MCTD compared with state-of-the-art methods.

Keywords Cross-modal hashing · Tensor factorization · Metric learning · Multi-view learning

1 Introduction

The development of social media has greatly enriched the multi-source data. For example, a news coverage often contains texts, images and videos. It is intuitive to use one modality to retrieve the other semantically similar modality, such as using texts to search for video clips, or searching for the story behind an oil painting. Such procedure is cross-modal retrieval, which has drawn much attention in recent years. In the past few years, this has become a fundamental problem in several emerging applications including visual search [2], image annotation [8,31] and object detection/recognition [6]. Hashing method has been widely used in cross-modal retrieval, which embeds multimodal data into a common latent representation space and generates similar binary codes for similar objects [34]. However, this is a challenging problem due to the existed dual heterogeneity: For different modalities, the distribution of data is intrinsically different; for features obtained in multi-view, simple concatenation cannot take full advantage of the information for each feature. Thus, in this paper, we focus on how to mitigate the dual heterogeneity effectively in order to facilitate the cross-modal retrieval procedure.

Most existing cross-modal retrieval methods only try to overcome the heterogeneity among modalities, while failing to fully exploit the useful multi-view features. However, the features extracted through different views in each modality can provide complementary information. In this paper, we refer to the feature representations extracted from different views as "multi-view features," which are shown in Fig. 1. For example, handcrafted features and deep-learned features characterize the different aspects of image data [1,13,32]. Similarly, in textual data feature extraction, explicit features and latent features play different roles. In order to fully utilize

✉ Limeng Cui
lzc334@psu.edu

1 College of Information Science and Technology, Pennsylvania State University, State College, PA, USA

2 IFM Lab, Department of Computer Science, Florida State University, Tallahassee, FL, USA

3 Weill Cornell Medicine, Cornell University, New York, NY, USA

4 Department of Computer Science, University of Illinois at Chicago, Chicago, IL, USA

the multi-view features, an intuitive solution is to concatenate all the features together. But as the features are highly nonlinear, such concatenation may let dense views dominate the feature space and override the effects of the sparse ones. Hence, we focus on effective fusion strategy which explores feature interactions across different views.

In this paper, we also consider the supervision information as a complementary information, like news article categories as shown in Fig. 1. Supervised cross-modal retrieval methods take advantage of this feature and achieve better results than unsupervised methods. Most existing methods enforce the same-class samples lie as close as possible in the representation space. However, separating different subspaces that correspond to different classes is widely ignored. As a result, these methods lose the interclass discriminant ability. Motivated by this gap, we propose to embed the supervision information into our framework to learn a more discriminative representation space.

In this paper, we proposed to fuse the multimodal and multi-view data, along with the supervision information, in order to facilitate the cross-modal retrieval procedure. It is a challenging problem due to the following problems:

- As the distributions of multimodal multi-view features are different, is there a way to fuse these data properly and explore the potential correlations to facilitate the cross-modal retrieval task?
- How to embed the supervision information properly in order to maintain a discriminative latent space?
- How to map the incoming new queries into the latent representation space and obtain the hash code?

We propose a novel multi-view cross-modality hashing method, called Multi-view Collective Tensor Decomposition (MCTD) in this paper, which uses a collective tensor decomposition framework in order to fuse the multimodal multi-view features. To our best knowledge, our work is the first to introduce tensor decomposition into cross-modal retrieval task. In order to embed the multi-view features properly, we propose a fusion strategy which uses tensor to model the multi-view feature from different modalities and collectively learns a latent tensor space by using tensor decomposition. Next, we enforce the supervision information into the learning procedure by maintaining the block-diagonal structure of the obtained latent tensor space. A block-diagonal structure loss term is proposed upon the above idea. Finally, for incoming new queries, two groups of mapping matrices are proposed to map the features into the latent space and generate the hash codes. Experimental results demonstrate the effectiveness of the proposed method MCTD and our fusion strategy.

This paper is organized as follows: The introduction briefly introduces the background of multimodal data and explains the perspective of multi-view feature interaction and the necessity of learning. In the related work, the relevant theoretical basis and algorithm of cross-modal retrieval problem are summarized. Next, we discuss the preliminaries of our method. Then, the proposed algorithm is introduced in detail, including three parts: latent semantic space learning, structural learning and extension to new query data. We also propose the overall architecture of the model and the optimization process. After that, extensive experiments demonstrate the effectiveness of the algorithm. Finally, we talk about the conclusion and future work on this topic.

2 Related work

Numerous papers have been published on cross-modal retrieval over the past decades [2,5,12,16,18,19,21,31,33,40,44,45]. Interested readers are referred to [35] for a comprehensive survey of various cross-modal retrieval methods. We now discuss related work in rank-based, deep learning-based and subspace learning-based methods.

Rank-based methods In [38], authors propose a cross-modal retrieval method based on Local Regression and Global Alignment. In [20], authors use cross-modal retrieval as a sorting problem and propose a rank-based cross-modal algorithm to optimize ranking loss.

Deep learning methods In [5], authors propose Deep Visual-Semantic Hash (DVSH) to characterize the association between visual data and natural language. In [4], authors use quantitative methods to learn deep images and text representations. In [36], authors use Convolutional Neural Network (CNN) feature of images to perform cross-modal retrieval between images and text.

Subspace learning methods In [46], authors proposed to perform cross-modal similarity search by employing Sparse Coding and Matrix Factorization (SCMF) to bridge the semantic gap and capture high-level latent semantic information. In [19], the Non-negative Matrix Factorization (NMF) was applied across the different modalities to tackle the multimodal problem. In [39], authors proposed Semantic Consistency Hashing (SCH) method by learning a shared semantic space. A cross-modality hashing method based on matrix factorization (SMFH) [33] was proposed to consider the label consistency across different modalities. In [17], authors proposed a ranking-based method which constructs a common Hamming space where the cross-modal similarity can be measured by using Hamming distance.

Our collective tensor decomposition differs from these methods. On the one hand, we explore the correlations of the input multi-view features. Different from traditional multi-view methods [30], MCTD considers multi-view features. On

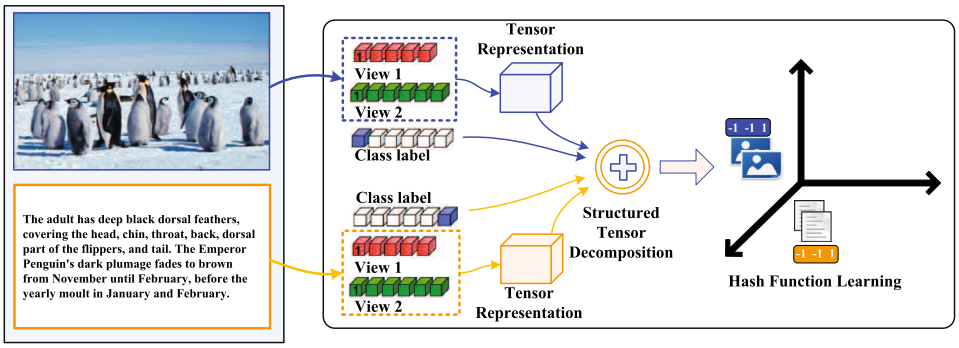

Fig. 1 Multi-view Tensor Decomposition Hashing (MCTD) for cross-modal retrieval of images and text sentences

the other hand, inspired by the idea of collective matrix factorization, we propose to use tensor decomposition to learn the latent space which captures a broader view of features.

The prior cross-modal hashing methods can be roughly divided into three categories including unsupervised, semi-supervised [41] and supervised cross-modal hashing.

Unsupervised cross-modal hashing [9] introduced a procrustean approach called dubbed iterative quantization (ITQ) to learn the binary codes for large-scale image retrieval. [7] proposed to learn the unified hash codes by collective matrix factorization with a latent factor model from different modalities of one instance. Based on the assumption that the hash codes of different modalities of one instance are identical, [46] presented a novel Latent Semantic Sparse Hashing (LSSH) to perform cross-modal similarity search by using sparse coding and matrix factorization to learn semantic features for images and text, respectively. [21] employed a three-step approach called Regularized Cross-Modal Hashing (RCMH), which can project annotation and visual feature descriptors into a common Hamming space.

Supervised cross-modal hashing In [40], Semantic Correlation Maximization (SCM) was proposed to seamlessly integrate semantic labels into the hashing learning procedure for large-scale data modeling. Semantics-Preserving Hashing (SePH) was proposed in [18], which can transform the given semantic affinities of training data into a probability distribution and approximate it with the hash codes in Hamming space. Semi-paired Discrete Hashing (SPDH) [29] jointly learns the latent features and hash codes with a factorization-based coding scheme. Discrete Cross-modal Hashing (DCH) was proposed in [37], which directly learns discriminative binary codes while retaining the discrete constraints. As for deep learning methods, [12] presented a deep hashing model to capture the cross-modal correspondences between visual data and natural language. [43] used GAN for unsupervised representation learning to exploit the underlying manifold structure of cross-modal data. [24] proposed a hierarchical network with multi-grained fusion for cross-modal correlation learning. Cross-Media multiple Deep Network (CMDN) was proposed in [23] to exploit the cross-modal correlation by hierarchical learning. Cross-modal Hybrid Transfer Network (CHTN), proposed in [10], uses two subnetworks for transferring knowledge to both two modalities simultaneously. [25] introduced reinforcement learning into cross-modal retrieval task. [42] used fine-grained ranking for different queries by weighted Hamming distance.

As supervised methods can fully use the supervision information, we only discuss the supervised cross-modal hashing method.

In contrast to previous work, we maintain the intraclass similarity and the interclass dissimilarity at the same time. This allows us to learn more subtle variations in the data structure and leads to a more accurate and efficient algorithm.

Table 1 List of basic symbols

Symbol	Definition and description
x	Each lowercase letter represents a scale
$\mathbf{x}$	Each boldface lowercase letter represents a vector
$\mathbf{X}$	Each boldface capital letter represents a matrix
$\mathcal{X}$	Each calligraphic letter represents a tensor, or space
$[1:K]$	A set of integers in the range of 1 to K inclusively
$[\mathbf{a};\mathbf{b}]$	Denotes two vectors are concatenated by column
$[\mathbf{a},\mathbf{b}]$	Denotes two vectors are concatenated by row
$\circ$	Denotes outer product
$\langle\cdot\rangle$	Denotes inner product
$\otimes$	Denotes the Kronecker product of matrices
$*$	Denotes Hadamard product

3 Preliminary

In this section, we first introduce some related concepts and notations about tensor. Then, we describe the problem of cross-model retrieval with multimodal data. Table 1 lists some basic symbols that will be used throughout the paper.

3.1 Basic concepts and notations

A tensor is a multidimensional array. More formally, a Kth-order tensor is an element of the tensor product of K vector spaces, each of which has its own coordinate system. For example, a vector is a first-order tensor and matrix is a second-order tensor. According to [14], a Kth-order tensor is denoted by $\mathcal{X} \in \mathbb{R}^{J_1 \times \cdots \times J_K}$ and its element is denoted by $x_{j_1,\ldots,j_K}$. Some notations and operations about tensor are given as follows which will be used in this paper.

Definition 1 (*Outer product*) The *outer product* of K vectors $\mathbf{x}^{(k)} \in \mathbb{R}^{J_k}$ for $k \in [1:K]$ is an Kth-order tensor and defined elementwise by

$$\left(\mathbf{x}^{(1)} \circ \cdots \circ \mathbf{x}^{(K)}\right)_{j_1,\ldots,j_K} = x^{(1)}_{j_1} \cdots x^{(K)}_{j_K},$$

for all values of the indices.

Definition 2 (*Matricization*) *Matricization* is the process of reordering the elements of a K-way array into a matrix. Specifically, the *mode-n matricization* of a tensor $\mathcal{X} \in \mathbb{R}^{J_1 \times \cdots \times J_K}$ is denoted by $\mathbf{X}_{(n)}$ and arranges the model-n fibers to be the columns of the resulting matrix, which means that the tensor element $x_{j_1,j_2,\ldots,j_K}$ maps to matrix element $x_{j_n,l}$ where

$$l = 1 + \sum_{\substack{k=1 \\ k \neq n}}^{K} (j_k - 1) L_k \quad with \;\; L_k = \prod_{\substack{m=1 \\ m \neq n}}^{k-1} J_m$$

Definition 3 (*n-Mode product*) The *n-mode product* of a tensor $\mathcal{X} \in \mathbb{R}^{J_1 \times \cdots \times J_K}$ with a matrix $\mathbf{U} \in \mathbb{R}^{R \times J_n}$ is denoted by $\mathcal{X} \times_n \mathbf{U}$ and is of size $J_1 \times \cdots \times J_{n-1} \times R \times J_{n+1} \times \cdots \times J_K$. Elementwise, we have

$$(\mathcal{X} \times_n \mathbf{U})_{j_1 \cdots j_{n-1} r j_{n+1} \cdots j_K} = \sum_{j_n=1}^{J_n} x_{j_1 \cdots j_K} u_{r j_n}.$$

This idea can also be expressed in terms of unfolded tensors, where the mode-n fiber $\mathbf{X}_{(n)}$ is multiplied by the matrix $\mathbf{U}$:

$$\mathcal{X} \times_n \mathbf{U} \Leftrightarrow \mathbf{U}\mathbf{X}_{(n)}$$

Definition 4 (*Tucker decomposition*) The *Tucker Decomposition* can decompose a tensor into a core tensor multiplied (or transformed) by a matrix along each mode. Thus, for a tensor $\mathcal{X} \in \mathbb{R}^{J_1 \times \cdots \times J_K}$, we have

$$\mathcal{X} = \mathcal{G} \times_1 \mathbf{A}^{(1)} \times_2 \mathbf{A}^{(2)} \cdots \times_K \mathbf{A}^{(K)}$$

where $\mathcal{G} \in \mathbb{R}^{I_1 \times \cdots \times I_K}$ is the core tensor and $\mathbf{A}^{(k)} \in \mathbb{R}^{J_k \times I_k}$ are the factor matrices for $k \in [1:K]$.

In fact, the *Tucker decomposition* can be transformed into the matricized forms by

$$\mathbf{X}_{(n)} = \mathbf{A}^{(n)} \mathbf{G}_{(n)} \left(\mathbf{A}^{(K)} \otimes \cdots \otimes \mathbf{A}^{(n+1)} \otimes \mathbf{A}^{(n-1)} \otimes \cdots \otimes \mathbf{A}^{(1)}\right)^T \tag{1}$$

where $\mathbf{X}_{(n)}$ and $\mathbf{G}_{(n)}$ are the mode-n matricization of tensor $\mathcal{X}$ and $\mathcal{G}$, respectively, $n \in [1:K]$ and $\otimes$ denotes the *Kronecker product* of matrices.

4 Multi-view collective tensor decomposition

Suppose that we have training data with n instances drawn from two modalities $\mathcal{I}$ and $\mathcal{T}$, where the data from each modality are composed with $V \in \mathbb{R}$ views. Specifically, $\mathbf{X}_I = [\mathbf{X}_I^{(1)}; \mathbf{X}_I^{(2)}; \cdots; \mathbf{X}_I^{(V)}] \in \mathbb{R}^{(m_I^1 + \cdots + m_I^V) \times n}$ and $\mathbf{X}_T = [\mathbf{X}_T^{(1)}; \mathbf{X}_T^{(2)}; \cdots; \mathbf{X}_T^{(V)}] \in \mathbb{R}^{(m_T^1 + \cdots + m_T^V) \times n}$ are the training data matrices drawn from modality $\mathcal{I}$ and modality $\mathcal{T}$, respectively, where $\mathbf{X}_I^{(v)} \in \mathbb{R}^{m_I^v \times n}$ and $\mathbf{X}_T^{(v)} \in \mathbb{R}^{m_T^v \times n}$ are the data matrix for the vth view, m_I^v and m_T^v are the corresponding dimensions of view $v \in [1:V]$. The goal of MCTD is to learn two groups of hash functions for the data from each modality that are able to generate unified hash codes. i.e., $f(\mathcal{K}_I) : \mathbb{R}^{d_I^1 \times \cdots \times d_I^V} \rightarrow \{-1, +1\}^{R^V}$ and $g(\mathcal{K}_T) : \mathbb{R}^{d_T^1 \times \cdots \times d_T^V} \rightarrow \{-1, +1\}^{R^V}$, where $\mathcal{K}_I$ and $\mathcal{K}_T$ are the instances drawn from each modal, respectively, and R^V is the length of binary codes.

Multi-view Collective Tensor Decomposition (MCTD) is a unified framework with three main components for supervised learning to hash, as shown in Fig. 2. The framework accepts input in an image–text pairwise form and processes them through latent representation learning: (1) collective tensor decomposition to generate a common latent representation space between two modalities represented in full-order tensor form; (2) a block-diagonal loss for exploiting supervision information; and (3) two groups of linear projections for mapping the new queries into the latent space.

4.1 Collective tensor decomposition

Most cross-modal hashing methods are built upon a reasonable assumption that heterogeneous data with the same semantic label share a common subspace [19,33,47], called latent representation space. In the latent space, the semantic representations of relevant data from different modalities are close to each other. We follow this idea and pursue a more general framework. In this part, we explore the correlations on multi-view across different modalities and propose a novel latent representation space learning method by using collective tensor decomposition.

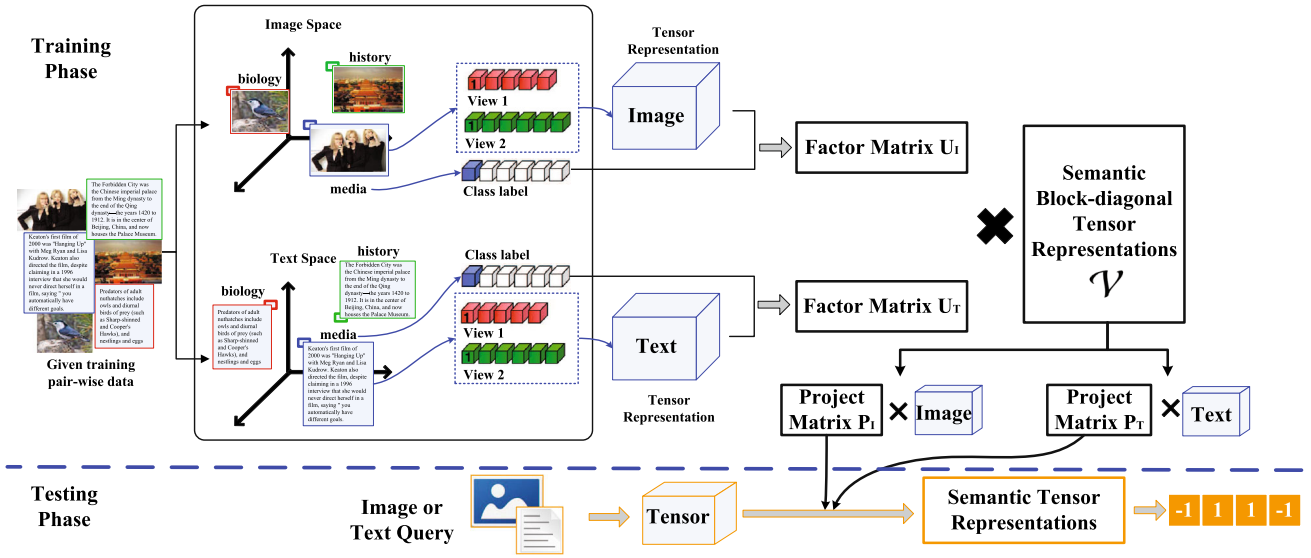

Fig. 2 MCTD constitutes: (1) collective tensor decomposition to generate a common latent representation space between two modalities represented in full-order tensor form; (2) a block-diagonal loss for exploiting supervision information; and (3) two groups of linear projections for mapping the new queries into the latent space

Modeling correlations on multi-view In order to capture interactions among the features across multi-view on two modalities, here we propose a fusion strategy by exploring the concept of Factorization Machines [27] to capture the second-order interactions as well as the concept of Multi-view Machines [3] to capture higher order interactions.

Hence, to fully utilize the complementary information provided by multi-view, we use the full-order interactions among all the V views to represent each of the data instead of the direct concatenating. Specifically, for each instance $\mathbf{x}_I = [\mathbf{x}_I^{(1)}; \cdots; \mathbf{x}_I^{(V)}]$ from modality $\mathcal{I}$, we can compose the full-order interactions among different views through the outer product of the feature vectors from different views as follows:

$$\begin{aligned} &1^{st}\ order : \mathbf{x}_I^{(v)} \quad \forall v \in [1:V] \\ &2^{nd}\ order : \mathbf{x}_I^{(v_1)} \circ \mathbf{x}_I^{(v_2)} \quad \forall v_1, v_2 \in [1:V], v_1 \neq v_2 \\ &\cdots \\ &V^{th}\ order : \mathbf{x}_I^{(1)} \circ \cdots \circ \mathbf{x}_I^{(V)} \end{aligned} \tag{2}$$

It is easy to integrate all the interactions into a unified tensor representation by adding a constant value "1" to each feature vector $\mathbf{x}_I^{(v)}$, $v \in [1:V]$. Let $\mathbf{k}_I^{(v)} = [1; \mathbf{x}_I^{(v)}]$, we have the tensor representation for each instance as $\mathcal{K}_I = \mathbf{k}_I^{(1)} \circ \cdots \circ \mathbf{k}_I^{(V)} \in \mathbb{R}^{d_I^1 \times \cdots \times d_I^V}$, where $d_I^v = m_I^v + 1$ for all $v \in [1:V]$. Different from directly modeling the interactions of feature $\mathbf{x}_I^{(v)}$, now we can get feature interactions with different orders which reflect complementary insights.

Then, the data matrix from modality $\mathcal{I}$ can be transformed into the data tensor $\mathcal{X}_I = [\mathcal{K}_{I1}, \mathcal{K}_{I2}, \ldots, \mathcal{K}_{In}] \in \mathbb{R}^{d_I^1 \times \cdots \times d_I^V \times n}$. Similarly, we can get the tensor representation for the data matrix of modality $\mathcal{T}$: $\mathcal{X}_T \in \mathbb{R}^{d_T^1 \times \cdots \times d_T^V \times n}$, where $d_T^v = m_T^v + 1$ for all $v \in [1:V]$.

Learning latent representation space In cross-modal hashing, heterogeneous data are mapped into a unified latent representation space so that the similarity can be directly compared. Learning such latent space is of great importance. In this section, we propose a method called Collective Tensor Decomposition (CTD) to obtain the common representation. We apply Tucker tensor decomposition, which can be considered as a higher order generalization of Principal Component Analysis (PCA). It decomposes a tensor into a core tensor multiplied by a matrix along each mode [14].

Suppose that we are given two heterogeneous data tensors $\mathcal{X}_I \in \mathbb{R}^{d_I^1 \times \cdots \times d_I^V \times n}$ and $\mathcal{X}_T \in \mathbb{R}^{d_T^1 \times \cdots \times d_T^V \times n}$. According to [22], the results of CTD on $\mathcal{X}_I$ and $\mathcal{X}_T$ can be expressed by

$$\begin{cases} \mathcal{X}_I \approx \mathcal{V} \times_1 \mathbf{U}_I^{(1)} \times_2 \mathbf{U}_I^{(2)} \cdots \times_V \mathbf{U}_I^{(V)} \\ \mathcal{X}_T \approx \mathcal{V} \times_1 \mathbf{U}_T^{(1)} \times_2 \mathbf{U}_T^{(2)} \cdots \times_V \mathbf{U}_T^{(V)} \end{cases} \tag{3}$$

where $\{\mathbf{U}_I^{(v)} \in \mathbb{R}^{d_I^v \times R}\}_{v=1}^V$, $\{\mathbf{U}_T^{(v)} \in \mathbb{R}^{d_T^v \times R}\}_{v=1}^V$ are the factor matrices (which are usually orthogonal) and can be thought of as the *principal components in each view*, $\mathcal{V} \in \mathbb{R}^{R \times \cdots \times R \times n}$ is the *core tensor* and its entries show the level of interaction between the different components. The $(V+1)$th-order tensor $\mathcal{V}$ is the common latent representation of $\mathcal{X}_I$ and $\mathcal{X}_T$.

The average decomposition loss for CTD is defined as

$$\mathcal{L}_{ctd} = \alpha||\mathcal{X}_I - \mathcal{V} \times_1 \mathbf{U}_I^{(1)} \cdots \times_V \mathbf{U}_I^{(V)}||^2 + (1-\alpha)||\mathcal{X}_T - \mathcal{V} \times_1 \mathbf{U}_T^{(1)} \cdots \times_V \mathbf{U}_T^{(V)}||^2 \quad (4)$$

where α is a trade-off parameter.

4.2 Block-diagonal structure loss

It is natural to assume that the intrinsic representations of data points from the same class are embedded in the same subspace and that these subspaces are separated. Therefore, it is straightforward to explicitly pursue the block-diagonal structure of the latent tensor representation by exploring the labeled data. A novel loss named block-diagonal structure loss is proposed in this part.

Assume that these n data points are sampled from C classes and each instance is labeled with one class label. To better illustrate the block-diagonal structure, the labeled data instances are arranged according to their labels. For the tensor instances that belong to c class $\mathcal{X}_{Ic}$ and $\mathcal{X}_{Tc}$, their ideal common representation is denoted by $\mathcal{V}_c^* \in \mathbb{R}^{r\times\cdots\times r\times n_c}$, where r is the dimensionality of each subspace, n_c is the instance number of class c and $c \in [1 : C]$. Since then, the ideal block-diagonal structured tensor representation $\mathcal{V}^*$ of data tensors $\mathcal{X}_I$ and $\mathcal{X}_T$ is shown as follows:

$$\mathcal{V}^* = diag(\mathcal{V}_1^*, \mathcal{V}_2^*, \ldots, \mathcal{V}_C^*) \quad (5)$$

However, the dimension of $\mathcal{V}$ is defined by hash code length, not by r. So we introduce a group of auxiliary matrices $\mathbf{Z}^{(v)}$ to change the mode of $\mathcal{V}$ into $\mathcal{V}^*$ with arbitrary dimension:

$$\mathcal{V}^* = \mathcal{V} \times_1 \mathbf{Z}^{(1)} \cdots \times_V \mathbf{Z}^{(V)} \quad (6)$$

where $\mathbf{Z}^{(v)} \in \mathbb{R}^{rC\times R}$ and $v \in [1 : V]$. To enforce the block-diagonal structure of $\mathcal{V}$, we propose a loss function. In detail, let $\mathcal{E}_0 \in \mathbb{R}^{rC\times\cdots\times rC\times n}$ and $\mathcal{E}_c^* \in \mathbb{R}^{r\times\cdots\times r\times n_c}$ ($c \in [1 : C]$) be the tensors with all elements equal "1". We first define an indicator tensor as

$$\mathcal{E} = \mathcal{E}_0 - diag(\mathcal{E}_1^*, \mathcal{E}_2^*, \ldots, \mathcal{E}_C^*) \quad (7)$$

Then, we have the loss of block-diagonal structure (BDS) as follows:

$$\mathcal{L}_{bds} = \frac{1}{2}||\mathcal{E} * (\mathcal{V} \times_1 \mathbf{Z}^{(1)} \cdots \times_V \mathbf{Z}^{(V)})||^2 \quad (8)$$

in which $*$ is the *Hadamard product*, which denotes the elementwise multiplication operator.

In fact, the block-diagonal structure loss can be seen as a global form of structural regularization that can influence the representations of all the classes. In this step, pursuing block-diagonal representations of the latent space guarantees that the representations of data points from the same class will be embedded in the same subspace and that different subspaces can be easily separated.

4.3 New query projection

For new queries, we can map the original feature interactions into the latent representation space by two groups of linear projections, respectively:

$$\begin{cases} \mathcal{V}_I = \mathcal{X}_I \times_1 \mathbf{P}_I^{(1)} \times_2 \mathbf{P}_I^{(2)} \cdots \times_V \mathbf{P}_I^{(V)} \\ \mathcal{V}_T = \mathcal{X}_T \times_1 \mathbf{P}_T^{(1)} \times_2 \mathbf{P}_T^{(2)} \cdots \times_V \mathbf{P}_T^{(V)} \end{cases} \quad (9)$$

where $\mathbf{P}_I^{(v)} \in \mathbb{R}^{R\times d_I^v}$ and $\mathbf{P}_T^{(v)} \in \mathbb{R}^{R\times d_T^v}$ are the projecting matrix groups for all $v \in [1 : V]$.

Since the tensors from different modalities that describe the same objects have the same semantic representations, we can present the loss for linear projections as

$$\begin{aligned} \mathcal{L}_{lp} &= ||\mathcal{V} - \mathcal{V}_I||^2 + ||\mathcal{V} - \mathcal{V}_T||^2 \\ &= ||\mathcal{V} - \mathcal{X}_I \times_1 \mathbf{P}_I^{(1)} \cdots \times_V \mathbf{P}_I^{(V)}||^2 \\ &\quad + ||\mathcal{V} - \mathcal{X}_T \times_1 \mathbf{P}_T^{(1)} \cdots \times_V \mathbf{P}_T^{(V)}||^2 \end{aligned} \quad (10)$$

4.4 Overall objective function

The overall objective function, consisting of the collective tensor decomposition term $\mathcal{L}_{ctd}$ in Eq. (4), the block-diagonal structure term $\mathcal{L}_{bds}$ in Eq. (8), the linear projection term $\mathcal{L}_{lp}$ in Eq. (10) and a regularization term, is given as follows:

$$\begin{aligned} \min \mathcal{L} &= \mathcal{L}_{ctd} + \mu\mathcal{L}_{bds} + \beta\mathcal{L}_{lp} \\ &\quad + \lambda\Psi(\{\mathbf{U}_I^{(v)}\}, \{\mathbf{U}_T^{(v)}\}, \{\mathbf{P}_I^{(v)}\}, \{\mathbf{P}_T^{(v)}\}, \{\mathbf{Z}^{(v)}\}, \mathcal{V}) \\ &= \alpha||\mathcal{X}_I - \mathcal{V} \times_1 \mathbf{U}_I^{(1)} \cdots \times_V \mathbf{U}_I^{(V)}||^2 \\ &\quad + (1-\alpha)||\mathcal{X}_T - \mathcal{V} \times_1 \mathbf{U}_T^{(1)} \cdots \times_V \mathbf{U}_T^{(V)}||^2 \\ &\quad + \frac{\mu}{2}||\mathcal{E} * (\mathcal{V} \times_1 \mathbf{Z}^{(1)} \cdots \times_V \mathbf{Z}^{(V)})||^2 \\ &\quad + \beta(||\mathcal{V} - \mathcal{X}_I \times_1 \mathbf{P}_I^{(1)} \cdots \times_V \mathbf{P}_I^{(V)}||^2 \\ &\quad + ||\mathcal{V} - \mathcal{X}_T \times_1 \mathbf{P}_T^{(1)} \cdots \times_V \mathbf{P}_T^{(V)}||^2) \\ &\quad + \lambda\Psi(\{\mathbf{U}_I^{(v)}\}, \{\mathbf{U}_T^{(v)}\}, \{\mathbf{P}_I^{(v)}\}, \{\mathbf{P}_T^{(v)}\}, \{\mathbf{Z}^{(v)}\}, \mathcal{V}) \end{aligned} \quad (11)$$

where μ, β and λ are the trade-off parameters of the corresponding terms, and the regularization term $\Psi(\cdot)$ is used to prevent overfitting.

The proposed formulation in (11) is hard to be directly solved since it is not convex or smooth with matrices $\{\mathbf{U}_I^{(v)}\}_{v=1}^V$, $\{\mathbf{U}_T^{(v)}\}_{v=1}^V$, $\{\mathbf{P}_I^{(v)}\}_{v=1}^V$, $\{\mathbf{P}_T^{(v)}\}_{v=1}^V$, $\{\mathbf{Z}^{(v)}\}_{v=1}^V$ and tensor $\mathcal{V}$. Therefore, we adopt an iterative multiplicative strategy. Specifically, the optimization procedure can be divided into the following steps:

Step 1 With $\{\mathbf{P}_I^{(v)}\}_{v=1}^V$, $\{\mathbf{P}_T^{(v)}\}_{v=1}^V$, $\{\mathbf{Z}^{(v)}\}_{v=1}^V$ and $\mathcal{V}$ fixed, the minimization over $\{\mathbf{U}_I^{(v)}\}$ and $\{\mathbf{U}_T^{(v)}\}$ is given by

$$\frac{\partial \mathcal{L}}{\partial \mathbf{U}_I^{(v)}} = -2\alpha \left(\mathbf{X}_{I(v)} - \mathbf{U}_I^{(v)} \mathbf{V}_{(v)} \left(\otimes \prod_{\substack{v'=V+1 \\ v' \neq v}}^{1} \mathbf{U}_I^{(v')} \right)^T \right) \cdot \left(\left(\otimes \prod_{\substack{v'=V+1 \\ v' \neq v}}^{1} \mathbf{U}_I^{(v')} \right) \mathbf{V}_{(v)}^T \right) + \lambda \frac{\partial \Psi \left(\mathbf{U}_I^{(v)} \right)}{\partial \mathbf{U}_I^{(v)}} \tag{12}$$

and

$$\frac{\partial \mathcal{L}}{\partial \mathbf{U}_T^{(v)}} = -2(1-\alpha) \left(\mathbf{X}_{T(v)} - \mathbf{U}_T^{(v)} \mathbf{V}_{(v)} \left(\otimes \prod_{\substack{v'=V+1 \\ v' \neq v}}^{1} \mathbf{U}_T^{(v')} \right)^T \right) \cdot \left(\left(\otimes \prod_{\substack{v'=V+1 \\ v' \neq v}}^{1} \mathbf{U}_T^{(v')} \right) \mathbf{V}_{(v)}^T \right) + \lambda \frac{\partial \Psi \left(\mathbf{U}_T^{(v)} \right)}{\partial \mathbf{U}_T^{(v)}} \tag{13}$$

where $\mathbf{X}_{I(v)}$, $\mathbf{X}_{T(v)}$ and $\mathbf{V}_{(v)}$ are the mode-v matricization of tensors $\mathcal{X}_I$, $\mathcal{X}_T$ and $\mathcal{V}$, respectively, $\otimes$ is the *Kronecker product* of matrices, and $\mathbf{U}_I^{(V+1)} = \mathbf{U}_T^{(V+1)} = \mathbf{E} \in \mathbb{R}^{n \times n}$ is the identity matrix.

Step 2 With $\{\mathbf{U}_I^{(v)}\}_{v=1}^V$, $\{\mathbf{U}_T^{(v)}\}_{v=1}^V$, $\{\mathbf{P}_I^{(v)}\}_{v=1}^V$, $\{\mathbf{P}_T^{(v)}\}_{v=1}^V$ and $\{\mathbf{Z}^{(v)}\}_{v=1}^V$ fixed, we have

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \mathcal{V}} = & -2\alpha \big((\mathcal{X}_I \times_1 \mathbf{U}_I^{(1)T} \cdots \times_V \mathbf{U}_I^{(V)T} - \mathcal{V}) \\ & - 2(1-\alpha) \big((\mathcal{X}_T \times_1 \mathbf{U}_T^{(1)T} \cdots \times_V \mathbf{U}_T^{(V)T} - \mathcal{V}) \\ & + 2\beta \big((\mathcal{V} - \mathcal{X}_I \times_1 \mathbf{P}_I^{(1)} \cdots \times_V \mathbf{P}_I^{(V)}) \\ & + (\mathcal{V} - \mathcal{X}_T \times_1 \mathbf{P}_T^{(1)} \cdots \times_V \mathbf{P}_T^{(V)}) \big) \\ & + \mu (\mathcal{E} \times_1 \mathbf{Z}^{(1)T} \cdots \times_V \mathbf{Z}^{(V)T}) * \mathcal{V} + \lambda \frac{\partial \Psi (\mathcal{V})}{\partial \mathcal{V}} \end{aligned} \tag{14}$$

Step 3 With $\{\mathbf{U}_I^{(v)}\}_{v=1}^V$, $\{\mathbf{U}_T^{(v)}\}_{v=1}^V$, $\{\mathbf{P}_I^{(v)}\}_{v=1}^V$, $\{\mathbf{P}_T^{(v)}\}_{v=1}^V$ and $\mathcal{V}$ fixed, the gradient w.r.t. $\{\mathbf{Z}^{(v)}\}$ is shown as

$$\frac{\partial \mathcal{L}}{\partial \mathbf{Z}^{(v)}} = \mu \left(\mathbf{E}_{(v)} * \mathbf{Z}^{(v)} \mathbf{V}_{(v)} \left(\otimes \prod_{\substack{v'=V+1 \\ v' \neq v}}^{1} \mathbf{Z}^{(v')} \right)^T \right)) \cdot \left(\left(\otimes \prod_{\substack{v'=V+1 \\ v' \neq v}}^{1} \mathbf{Z}^{(v')} \right) \mathbf{V}_{(v)}^T \right) + \lambda \frac{\partial \Psi \left(\mathbf{Z}^{(v)} \right)}{\partial \mathbf{Z}^{(v)}} \tag{15}$$

in which $\mathbf{E}_{(v)}$ is the mode-v matricization of tensor $\mathcal{E}$ and $\mathbf{Z}^{(V+1)} = \mathbf{E} \in \mathbb{R}^{n \times n}$ is the identity matrix.

Step 4 Similarly, with all the $\{\mathbf{U}_I^{(v)}\}_{v=1}^V$, $\{\mathbf{U}_T^{(v)}\}_{v=1}^V$, $\{\mathbf{Z}^{(v)}\}_{v=1}^V$ and $\mathcal{V}$ fixed, we can obtain

$$\frac{\partial \mathcal{L}}{\partial \mathbf{P}_I^{(v)}} = -2\beta \left(\mathbf{V}_{(v)} - \mathbf{P}_I^{(v)} \mathbf{X}_{I(v)} \left(\otimes \prod_{\substack{v'=V+1 \\ v' \neq v}}^{1} \mathbf{P}_I^{(v')} \right)^T \right) \cdot \left(\left(\otimes \prod_{\substack{v'=V+1 \\ v' \neq v}}^{1} \mathbf{P}_I^{(v')} \right) \mathbf{X}_{I(v)}^T \right) + \lambda \frac{\partial \Psi \left(\mathbf{P}_I^{(v)} \right)}{\partial \mathbf{P}_I^{(v)}} \tag{16}$$

and

$$\frac{\partial \mathcal{L}}{\partial \mathbf{P}_T^{(v)}} = -2\beta \left(\mathbf{V}_{(v)} - \mathbf{P}_T^{(v)} \mathbf{X}_{T(v)} \left(\otimes \prod_{\substack{v'=V+1 \\ v' \neq v}}^{1} \mathbf{P}_T^{(v')} \right)^T \right) \cdot \left(\left(\otimes \prod_{\substack{v'=V+1 \\ v' \neq v}}^{1} \mathbf{P}_T^{(v')} \right) \mathbf{X}_{T(v)}^T \right) + \lambda \frac{\partial \Psi \left(\mathbf{P}_T^{(v)} \right)}{\partial \mathbf{P}_T^{(v)}} \tag{17}$$

in which $\mathbf{P}_I^{(V+1)} = \mathbf{P}_T^{(V+1)} = \mathbf{E} \in \mathbb{R}^{n \times n}$ is the identity matrix.

The optimization procedure of MCTD is summarized in Algorithm 1.

Overall, for any new instance $\mathbf{x}_I = [\mathbf{x}_I^{(1)}; \cdots; \mathbf{x}_I^{(V)}]$ and $\mathbf{x}_T = [\mathbf{x}_T^{(1)}; \cdots; \mathbf{x}_T^{(V)}]$ drawn from each modality, we first transfer them into the full-order interactions presented in the form of tensor representations $\mathcal{K}_I$ and $\mathcal{K}_T$ according to Eq. (2). Then, the MCTD is to learn two groups of hash functions for the data from each modality that are able to generate unified hash codes, i.e., $f(\mathcal{K}_I) = sign(\mathcal{K}_I \times_1 \mathbf{P}_I^{(1)} \cdots \times_V \mathbf{P}_I^{(V)}) : \mathbb{R}^{d_I^1 \times \cdots \times d_I^V} \rightarrow \{-1, +1\}^{R^V}$ and $g(\mathcal{K}_T) = sign(\mathcal{K}_T \times_1 \mathbf{P}_T^{(1)} \cdots \times_V \mathbf{P}_T^{(V)}) : \mathbb{R}^{d_T^1 \times \cdots \times d_T^V} \rightarrow \{-1, +1\}^{R^V}$, where d_I^v

Algorithm 1 MCTD

Require: Image feature matrix $\mathbf{X}_I$ and text feature matrix $\mathbf{X}_T$ both in V views, the length of hash codes R, the category C, and the model parameters α, β, μ and λ.
Ensure: Unified hash codes $\mathbf{H}$, and the projection matrix groups $\{\mathbf{P}_I^{(v)}\}_{v=1}^V$ and $\{\mathbf{P}_T^{(v)}\}_{v=1}^V$.
1: Transforming the data matrix $\mathbf{X}_I$ and $\mathbf{X}_T$ into the tensor representations $\mathcal{X}_I$ and $\mathcal{X}_T$.
2: Randomly initializing $\{\mathbf{U}_I^{(v)}\}$, $\{\mathbf{U}_T^{(v)}\}$, $\{\mathbf{P}_I^{(v)}\}$, $\{\mathbf{P}_T^{(v)}\}$, $\{\mathbf{Z}^{(v)}\}$ and $\mathcal{V}$, respectively.
3: **while** not converged **do**
4: **for** $v := 1$ to V **do**
5: Fixing $\{\mathbf{P}_I^{(v)}\}$, $\{\mathbf{P}_T^{(v)}\}$, $\{\mathbf{Z}^{(v)}\}$ and $\mathcal{V}$, update $\mathbf{U}_I^{(v)}$ and $\mathbf{U}_T^{(v)}$.
6: **end for**
7: Fixing $\{\mathbf{U}_I^{(v)}\}$, $\{\mathbf{U}_T^{(v)}\}$, $\{\mathbf{P}_I^{(v)}\}$, $\{\mathbf{P}_T^{(v)}\}$ and $\{\mathbf{Z}^{(v)}\}$, update $\mathcal{V}$.
8: **for** $v := 1$ to V **do**
9: Fixing $\{\mathbf{U}_I^{(v)}\}$, $\{\mathbf{U}_T^{(v)}\}$, $\{\mathbf{P}_I^{(v)}\}$, $\{\mathbf{P}_T^{(v)}\}$ and $\mathcal{V}$, update $\mathbf{Z}^{(v)}$.
10: **end for**
11: **for** $v := 1$ to V **do**
12: Fixing $\{\mathbf{U}_I^{(v)}\}$, $\{\mathbf{U}_T^{(v)}\}$, $\{\mathbf{Z}^{(v)}\}$ and $\mathcal{V}$, update $\mathbf{P}_I^{(v)}$ and $\mathbf{P}_T^{(v)}$.
13: **end for**
14: **end while**
15: Generating the hash codes by $\mathbf{H} = sign(\mathbf{V}_{(V+1)})$.

and d_T^v are the dimensions of mode-v fiber of tensors $\mathcal{K}_I$ and $\mathcal{K}_T$, and R^V is the length of binary codes.

4.5 Complexity analysis

In the application, MCTD firstly generates the latent representation for a new query based on the achieved projection matrix groups $\{\mathbf{P}_I^{(v)}\}$ and $\{\mathbf{P}_T^{(v)}\}$, and then the hash codes can be obtained. The main time consumption of the proposed MCTD is the tensor decomposition, and its complexity is $O(\Pi_{v=1}^V(d_I^v + d_T^v)R^{V-1}n^2)$. The parameters in Algorithm 1 are updated simultaneously, which indicates that the computation procedure can be paralleled. Therefore, the complexity caused by the interaction across V views is ameliorated. The convergence criterion used in our experiments is that the number of iterations is greater than a threshold (e.g., 200) or the decrease of the objective function value is smaller than a threshold.

5 Experiments

In this section, we conduct extensive experiments to evaluate the effectiveness of the proposed method MCTD comparing with several state-of-the-art hashing methods on two public cross-modal datasets.

5.1 Datasets

Experiments are conducted to validate the advantages of the proposed cross-modality hashing method on two real-world datasets.

Wiki[1] Wiki dataset is collected from Wikipedia consisting of 2173/693 (training/testing) multimedia documents. Each document contains a single image and at least 70 words. Totally ten categories are considered in this dataset, and each image–text pair is labeled by one of them. Documents are considered to be similar if they belong to the same category.

Pascal VOC[2] The dataset [11] consists of 5011/4952 (training/testing) image–tag pairs, which can be categorized into 20 different classes. Since some images are multi-labeled, researchers usually select images with only one object as the way in [28], resulting in 1865 training and 1905 testing data. The image features include histograms of bag of visual words, GIST and color. The text features are 399-D tag occurrence features.

5.2 Compared methods

We compare the performance of our method with several state-of-the-art hashing-based cross-modal retrieval methods including **CMFH**[3] [7], **LSSH**[4] [46], **SCM** [40], **SePH**[5] [18], **SMFH**[6] [33] and **DCMH**[7] [12], which can be organized into three categories:

- **Unsupervised hashing** **LSSH** is an unsupervised method, which learns a joint abstraction space for image and text by using sparse coding and matrix factorization.
- **Supervised hashing (with shallow architecture) SCM** is a representative supervised method for cross-modal hashing, which is proposed to seamlessly integrate semantic labels into the hashing learning procedure. **CMFH** and **SMFH** are two methods based on matrix factorization, which both learn a common latent space for image and text. **SePH** uses the semantic affinities of training instances into a probability distribution and aims to approximate it in Hamming space. In the experiments, we use RBF kernel and take 500 as sampling size as advised in [18].
- **Supervised hashing (with deep architecture) DCMH** is the most recent work on deep cross-modal hashing, which integrates feature learning and hash-code learning into the same framework.

As existing cross-modal hashing methods cannot deal with the multi-view, we concatenate the features to fit the model.

[1] http://www.svcl.ucsd.edu/projects/crossmodal/.
[2] http://www.cs.utexas.edu/~grauman/research/datasets.html.
[3] http://ise.thss.tsinghua.edu.cn/MIG/code_data_cm.zip.
[4] http://ise.thss.tsinghua.edu.cn/MIG/LSSH_code.rar.
[5] https://bitbucket.org/linzijia72/.
[6] We thank the authors for kindly providing the codes.
[7] https://github.com/jiangqy/DCMH-CVPR2017.

Table 2 Mean Average Precision (MAP) for cross-modal retrieval tasks on two datasets

Task	Method	Wiki				Pascal VOC			
		16 bits	32 bits	64 bits	128 bits	16 bits	32 bits	64 bits	128 bits
I → T	CMFH	0.2115	0.2230	0.2238	0.2351	0.1575	0.1508	0.1490	0.1429
	LSSH	0.1541	0.1546	0.1544	0.1521	0.2988	0.3083	0.3194	0.3166
	SCM_orth	0.1527	0.1331	0.1216	0.1172	0.4063	0.4040	0.4067	0.4144
	SCM_seq	0.2257	0.2459	0.2461	0.2510	0.3842	0.4868	0.3972	0.4115
	SePH	0.2562	0.2654	0.2793	0.2823	0.4356	0.4424	0.4242	0.4245
	SMFH	0.2507	0.2646	0.2715	0.2787	0.2291	0.2477	0.2586	0.2500
	DCMH	0.2798	0.2809	0.2910	0.2993	0.4564	0.4613	0.4793	0.4801
	MCTD	**0.2919**	**0.3048**	**0.3068**	**0.3138**	**0.4921**	**0.4927**	**0.5194**	**0.5072**
T → I	CMFH	0.5351	0.5445	0.5586	0.5616	0.1576	0.1550	0.1523	0.1463
	LSSH	0.2641	0.2723	0.2795	0.2803	0.6145	0.6177	0.6042	0.5906
	SCM_orth	0.1532	0.1393	0.1297	0.1273	0.4791	0.4526	0.4962	0.4721
	SCM_seq	0.2341	0.2410	0.2445	0.2554	0.4816	0.5455	0.4526	0.4866
	SePH	0.6276	0.6324	0.6513	0.6514	0.6476	0.6524	0.6153	0.6571
	SMFH	0.4481	0.4827	0.4920	0.5038	0.4189	0.4942	0.6035	0.7388
	DCMH	0.6292	0.6524	0.6674	0.6720	0.6513	0.6504	0.6638	0.6708
	MCTD	**0.6482**	**0.6832**	**0.6898**	**0.6972**	**0.6567**	**0.6553**	**0.7074**	**0.7464**

Items in bold indicate the best performance

5.3 Evaluation protocols

For Wiki dataset, each image is represented by a 128-D SIFT histogram and a 128-D CNN feature. We use the output of layer *fc*8 in the AlexNet [15], which is pre-trained on ImageNet. Each text is represented by a 200-D bag-of-words feature and a 10-D topics' vector generated by Latent Dirichlet Allocation (LDA) model [26]. For Pascal VOC dataset, each image is also represented by both handcrafted features and deep features. The ground-truth neighbors are defined as those image–text pairs which share category label.

We perform two cross-modal retrieval tasks: using image queries to search relevant text ($I \rightarrow T$) and text query on image databases ($T \rightarrow I$). Following [18,33,40], we evaluate the retrieval performance based on two metrics: Mean Average Precision (MAP) and precision–recall curves. In our experiments, we repeat ten times for each group of parameters and report the mean MAP score. The results of numerical experiments are summarized in Table 2.

For our method, based on the rule of thumb, we set the parameters $r = 2$, $\alpha = 0.5$ and $\lambda = 0.05$ throughout the paper. The grid searching is applied to identify optimal values for the parameters from $\mu \in [0.001, 10]$ and $\beta \in [1, 200]$.

5.4 Quantitative results

We evaluate all methods with different lengths of hash codes, i.e., 16, 32, 64 and 128 bits, and report their MAP results in Table 2, where the best results are presented in bold figures.

From the experimental results, we can see that the effectiveness of the proposed MCTD method is proved through that it substantially surpasses all the compared methods for cross-modal retrieval tasks. Specifically, compared to the best results of CMFH, LSSH, SCM_orth, SCM_seq, SePH and SMFH, MCTD achieves absolute increases of 1.66%/2.44% and 3.36%/3.24% in average MAP score for two cross-modal tasks $I \rightarrow T$ and $T \rightarrow I$ on Wiki and Pascal VOC datasets. This can show that our feature fusion strategy is of great importance and very practical.

We find our method performs better than the deep method DCMH. We speculate that the reason behind is that our model incorporates both the handcrafted features and deep-learned features and exploit the correlations of the features. To confirm the above two assumptions, we further test the effect of correlations on multi-view in Sect. 5.5.

The precision–recall curves with 32 bits for the two cross-modal tasks $I \rightarrow T$ and $T \rightarrow I$ on these two datasets are presented in Fig. 3, respectively. We can observe from the results that MCTD is highly competitive compared with alternative methods.

We then validate the convergence of MCTD with 32 bits on these two datasets. In order to show the result clearer, we adjust the objective value to the log value, which is shown in Fig. 4. We can see that our method can converge in less than 20 iterations on both datasets in the optimization procedure, which is a satisfactory convergence rate.

Finally, the training time of all these methods is tested. The experiments are conducted on the Wiki dataset and run on a PC with 2.5 GHz Intel Core i7 CPU and 16 GB RAM. As

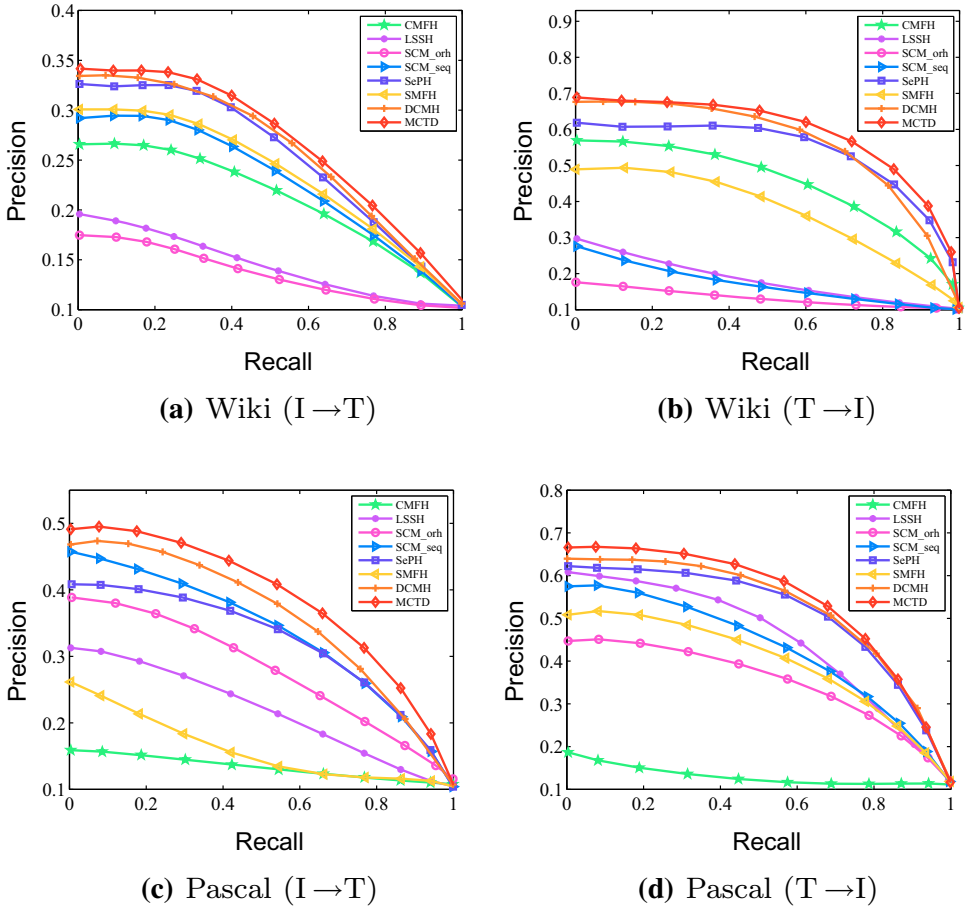

Fig. 3 Precision–recall curves of cross-modal retrieval on Wiki and Pascal

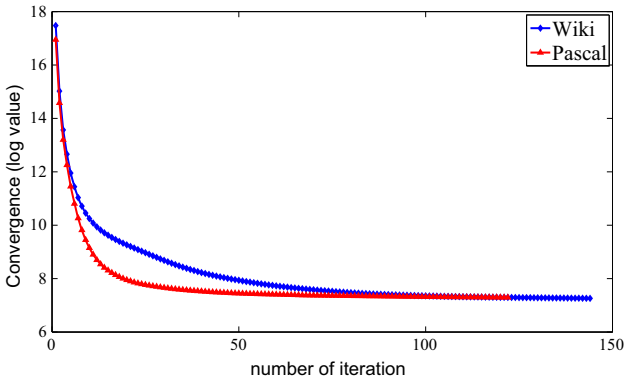

Fig. 4 Curves of convergence validation: objective function value @32 bits on both datasets

for the deep method DCMH, we use a server with NVIDIA GeForce GTX 1080 Ti GPU. We did not record the training time of DCMH. Here, we only evaluate the case that the code length is 32 bits. The results are reported in Table 3. We can observe that the time consumption of MCTD is of the same order of magnitude as that of CMFH and SMFH, both of which involve the computation of matrix inversion. The time cost is acceptable in comparison with that of LSSH and SePH. As MCTD needs to compute the tensor structure of multi-view, it spends more time than others in the training phase. And the space cost will grow as the number of views grows.

Table 3 Training time (s) of each method on Wiki

Method	Dataset	
	Wiki	Pascal VOC
CMFH	14.02	16.38
LSSH	432.22	361.86
SCM_orth	2.90	29.72
SCM_seq	2.11	8.76
SePH	189.85	154.40
SMFH	39.02	60.78
MCTD	45.33	73.24

5.5 Effect of correlations on multi-view

In order to validate our assumption that the correlations on multi-view features can boost the performance of the proposed method, we present two variants of MCTD for

Table 4 Effect of correlations on multi-view

Task	Method	Wiki			
		16 bits	32 bits	64 bits	128 bits
I → T	MCTD_c	0.2594	0.2783	0.2817	0.2909
	MCTD_h	0.2708	0.2865	0.2902	0.2944
	MCTD	**0.2919**	**0.3048**	**0.3068**	**0.3138**
T → I	MCTD_c	0.5536	0.6054	0.6231	0.6504
	MCTD_h	0.6024	0.6365	0.6592	0.6467
	MCTD	**0.6482**	**0.6832**	**0.6898**	**0.6972**

Items in bold indicate the best performance

comparison. The first one uses concatenated features and the second one only uses highest order correlations among all the views, where these two methods are denoted as MCTD_c and MCTD_h. We use MAP scores to evaluate the performance on the dataset Wiki with various hash code lengths. The results are summarized in Table 4. We can see that the correlations on multi-view produce positive results and the full-order correlations can lead to better performance by providing more comprehensive information, which verifies the effectiveness of the proposed fusion strategy.

5.6 Parameter sensitivity

We analyze the influence of two important parameters β and μ by computing the MAP score on 32 bits on both cross-modal retrieval tasks. We can see that the parameters in our method are not sensitive and MCTD can achieve satisfactory results in a wide range of parameter settings. The results are shown in Fig. 5.

6 Conclusion

Fusing multimodal multi-view features is a new and challenging job in cross-modal retrieval. In this paper, we propose a novel cross-modal hashing method called MCTD, which is a first attempt to use collective tensor decomposition to model the multi-view features and learn the latent space. In addition, our method can embed the supervised information into the learning procedure and enforce multiple separated subspaces. Our contributions are shown as follows: Firstly, we use two tensors to model the multi-view features and collective tensor decomposition to learn a latent tensor representation. Secondly, a block-diagonal structure loss is introduced to exploit the supervision information and maintain the global structure of the subspace. Thirdly, two groups of mapping matrices are proposed to project the incoming new queries to the latent space and generate corresponding hash codes. We also propose an optimization algorithm to solve the proposed objective function, which can effectively update multiple parameters simultaneously. We have conducted extensive experiments to validate the effectiveness of our method and the proposed feature fusion strategy.

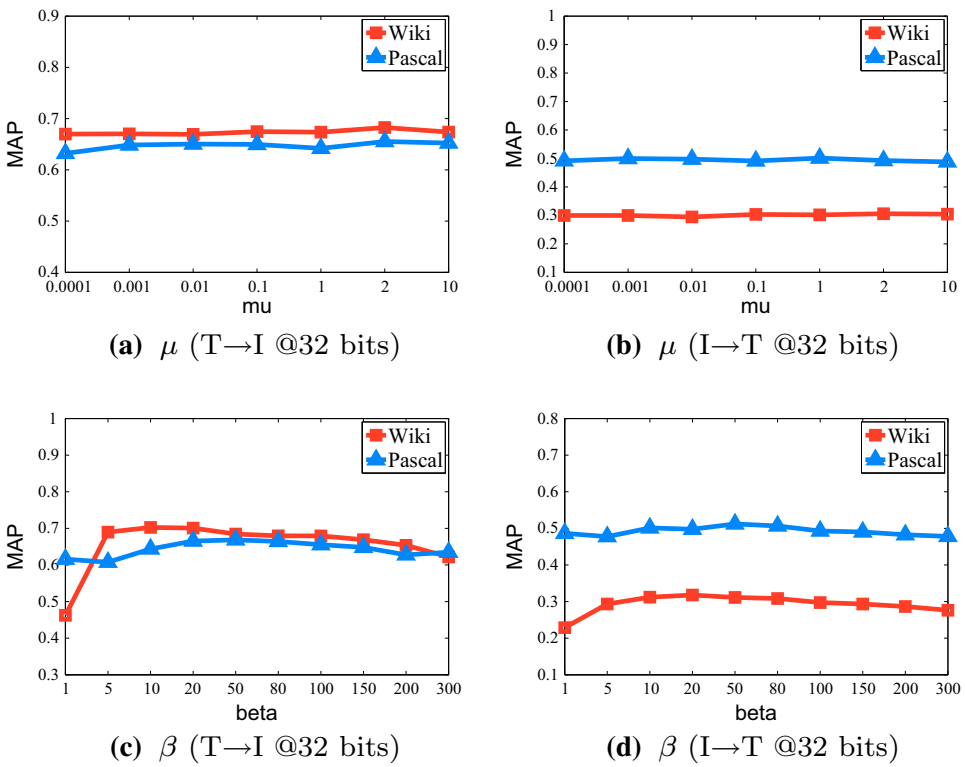

Fig. 5 Parameter sensitive analysis: the MAP score @32 bits on both the cross-modal retrieval tasks

Source code is available online: https://github.com/cuilimeng/MCTD.

7 Limitations and future work

Multimodal data provide a broad platform for machine learning. With the popularity of smart phones and the enrichment of online image resources, it is easier to collect related data. The proposed method is presented and evaluated based on the visual and textual data, but they can also be applied to a wider range of data. Based on this paper, we can continue to study and explore the following aspects:

- In this problem setting, each image has only one specific category label. However, in the real world, each image often contains multiple objects, which is reflected in the multi-label problem. This problem poses new challenges to potential semantic subspace learning. We can combine feature learning to model the association between tags to provide guidance for multi-label cross-modal retrieval problems.
- In addition, we only conduct experiments on the two data modalities in this paper and we can expand the model to other data modalities such as audio and video.

Acknowledgements The work is supported by the National Natural Science Foundation of China under Grant No.: 61672313 and 61503253, the National Science Foundation under Grant Nos.: IIS-1526499, IIS-1763365 and CNS-1626432 and Natural Science Foundation of Guangdong Province under Grant No.: 2017A030313339.

References

1. Antipov G, Berrani SA, Ruchaud N, Dugelay JL (2015) Learned versus hand-crafted features for pedestrian gender recognition. In: Proceedings of the 23rd ACM international conference on Multimedia. ACM, pp 1263–1266
2. Bronstein MM, Bronstein AM, Michel F, Paragios N (2010) Data fusion through cross-modality metric learning using similarity-sensitive hashing. In: Computer vision and pattern recognition (CVPR), 2010 IEEE conference on. IEEE, pp 3594–3601
3. Cao B, Zhou H, Li G, Yu PS (2016) Multi-view machines. In: Proceedings of the ninth ACM international conference on web search and data mining. ACM, pp 427–436
4. Cao Y, Long M, Wang J, Liu S (2017) Collective deep quantization for efficient cross-modal retrieval. In: AAAI, pp 3974–3980
5. Cao Y, Long M, Wang J, Yang Q, Yu PS (2016) Deep visual-semantic hashing for cross-modal retrieval. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. ACM, pp 1445–1454
6. Ding C, Tao D (2015) Robust face recognition via multimodal deep face representation. IEEE Trans Multimedia 17(11):2049–2058
7. Ding G, Guo Y, Zhou J (2014) Collective matrix factorization hashing for multimodal data. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2075–2082
8. Gong Y, Ke Q, Isard M, Lazebnik S (2014) A multi-view embedding space for modeling internet images, tags, and their semantics. Int J Comput Vis 106(2):210–233
9. Gong Y, Lazebnik S, Gordo A, Perronnin F (2013) Iterative quantization: a procrustean approach to learning binary codes for large-scale image retrieval. IEEE Trans Pattern Anal Mach Intell 35(12):2916–2929
10. Huang X, Peng Y, Yuan M (2017) Cross-modal common representation learning by hybrid transfer network. In: Proceedings of the 26th international joint conference on artificial intelligence. AAAI Press, pp 1893–1900
11. Hwang SJ, Grauman K (2012) Reading between the lines: object localization using implicit cues from image tags. IEEE Trans Pattern Anal Mach Intell 34(6):1145–1158
12. Jiang QY, Li WJ (2017) Deep cross-modal hashing. In: Computer vision and pattern recognition (CVPR), 2017 IEEE conference on. IEEE, pp 3270–3278
13. Jin L, Gao S, Li Z, Tang J (2014) Hand-crafted features or machine learnt features? Together they improve rgb-d object recognition. In: Multimedia (ISM), 2014 IEEE international symposium on. IEEE, pp 311–319
14. Kolda TG, Bader BW (2009) Tensor decompositions and applications. SIAM Rev 51(3):455–500
15. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105
16. Kumar S, Udupa R (2011) Learning hash functions for cross-view similarity search. In: IJCAI proceedings-international joint conference on artificial intelligence, vol 22, p 1360
17. Li K, Qi GJ, Ye J, Hua KA (2017) Linear subspace ranking hashing for cross-modal retrieval. IEEE Trans Pattern Anal Mach Intell 39(9):1825–1838
18. Lin Z, Ding G, Han J, Wang J (2017) Cross-view retrieval via probability-based semantics-preserving hashing. IEEE Trans Cybernet 47(12):4342–4355
19. Liu H, Ji R, Wu Y, Hua G (2016) Supervised matrix factorization for cross-modality hashing. In: Proceedings of the twenty-fifth international joint conference on artificial intelligence. AAAI Press, pp 1767–1773
20. Lu X, Wu F, Tang S, Zhang Z, He X, Zhuang Y (2013) A low rank structural large margin method for cross-modal ranking. In: Proceedings of the 36th international ACM SIGIR conference on research and development in information retrieval. ACM, pp 433–442
21. Moran S, Lavrenko V (2015) Regularised cross-modal hashing. In: Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval. ACM, pp 907–910
22. Mørup M, Hansen LK, Arnfred SM (2008) Algorithms for sparse nonnegative Tucker decompositions. Neural Computation 20(8):2112–2131
23. Peng Y, Huang X, Qi J (2016) Cross-media shared representation by hierarchical learning with multiple deep networks. In: IJCAI, pp 3846–3853
24. Peng Y, Qi J, Huang X, Yuan Y (2018) Ccl: cross-modal correlation learning with multigrained fusion by hierarchical network. IEEE Trans Multimedia 20(2):405–420
25. Qi J, Peng Y (2018) Cross-modal bidirectional translation via reinforcement learning. In: IJCAI, pp 2630–2636
26. Rasiwasia N, Costa Pereira J, Coviello E, Doyle G, Lanckriet GR, Levy R, Vasconcelos N (2010) A new approach to cross-modal multimedia retrieval. In: Proceedings of the 18th ACM international conference on Multimedia. ACM, pp 251–260
27. Rendle S (2010) Factorization machines. In: Data mining (ICDM), 2010 IEEE 10th international conference on. IEEE, pp 995–1000

28. Sharma A, Kumar A, Daume H, Jacobs DW (2012) Generalized multiview analysis: a discriminative latent space. In: Computer vision and pattern recognition (CVPR), 2012 IEEE conference on. IEEE, pp 2160–2167
29. Shen X, Shen F, Sun QS, Yang Y, Yuan YH, Shen HT (2017) Semi-paired discrete hashing: learning latent hash codes for semi-paired cross-view retrieval. IEEE Trans Cybern 47(12):4275–4288
30. Shen X, Shen F, Sun QS, Yuan YH (2015) Multi-view latent hashing for efficient multimedia search. In: Proceedings of the 23rd ACM international conference on Multimedia. ACM, pp 831–834
31. Song J, Yang Y, Yang Y, Huang Z, Shen HT (2013) Inter-media hashing for large-scale retrieval from heterogeneous data sources. In: Proceedings of the 2013 ACM SIGMOD international conference on management of data. ACM, pp 785–796
32. Tang J, Jin L, Li Z, Gao S (2015) Rgb-d object recognition via incorporating latent data structure and prior knowledge. IEEE Trans Multimedia 17(11):1899–1908
33. Tang J, Wang K, Shao L (2016) Supervised matrix factorization hashing for cross-modal retrieval. IEEE Trans Image Process 25(7):3157–3166
34. Wang J, Shen HT, Song J, Ji J (2014) Hashing for similarity search: a survey. arXiv preprint arXiv:1408.2927
35. Wang K, Yin Q, Wang W, Wu S, Wang L (2016) A comprehensive survey on cross-modal retrieval. arXiv preprint arXiv:1607.06215
36. Wei Y, Zhao Y, Lu C, Wei S, Liu L, Zhu Z, Yan S (2017) Cross-modal retrieval with cnn visual features: a new baseline. IEEE Trans Cybern 47(2):449–460
37. Xu X, Shen F, Yang Y, Shen HT, Li X (2017) Learning discriminative binary codes for large-scale cross-modal retrieval. IEEE Trans Image Process 26(5):2494–2507
38. Yang Y, Xu D, Nie F, Luo J, Zhuang Y (2009) Ranking with local regression and global alignment for cross media retrieval. In: Proceedings of the 17th ACM international conference on multimedia. ACM, pp 175–184
39. Yao T, Kong X, Fu H, Tian Q (2016) Semantic consistency hashing for cross-modal retrieval. Neurocomputing 193:250–259
40. Zhang D, Li WJ (2014) Large-scale supervised multimodal hashing with semantic correlation maximization. AAAI 1:7
41. Zhang J, Peng Y (2017) Ssdh: semi-supervised deep hashing for large scale image retrieval. IEEE Trans Circuits Syst Video Technol
42. Zhang J, Peng Y (2018) Query-adaptive image retrieval by deep weighted hashing. IEEE Trans Multimedia
43. Zhang J, Peng Y, Yuan M (2018) Unsupervised generative adversarial cross-modal hashing
44. Zhen Y, Yeung DY (2012) Co-regularized hashing for multimodal data. In: Advances in neural information processing systems, pp 1376–1384
45. Zhen Y, Yeung DY (2012) A probabilistic model for multimodal hash function learning. In: Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, pp 940–948
46. Zhou J, Ding G, Guo Y (2014) Latent semantic sparse hashing for cross-modal similarity search. In: Proceedings of the 37th international ACM SIGIR conference on research & development in information retrieval. ACM, pp 415–424
47. Zhu X, Huang Z, Shen HT, Zhao X (2013) Linear cross-modal hashing for efficient multimedia search. In: Proceedings of the 21st ACM international conference on multimedia. ACM, pp 143–152