Neural Computing and Applications (2023) 35:6665–6684
https://doi.org/10.1007/s00521-022-08006-6

ORIGINAL ARTICLE

Robust supervised matrix factorization hashing with application to cross-modal retrieval

Zhenqiu Shu[1] · Kailing Yong[1] · Donglin Zhang[2] · Jun Yu[3] · Zhengtao Yu[1] · Xiao-Jun Wu[2]

Received: 12 May 2022 / Accepted: 26 October 2022 / Published online: 27 November 2022

Abstract

In recent years, hashing methods have received extensive attention in multimedia search due to their high computational and storage efficiency. However, most of them explore the common representation of multi-modality data and then use it to generate the hash codes but ignore the specific properties of each modality. To mitigate this problem, we propose a novel hashing method, called Robust Supervised Matrix Factorization Hashing (RSMFH), which keeps both the shared and the specific properties of multimodality data by decomposing each modality into a common representation and an inconsistent representation. Moreover, we impose sparse constraints on the inconsistent part of each modality and minimize the production of the consistent parts, simultaneously. In addition, the supervised label information among the data is embedded into the learned hash codes enhancing the discriminative ability of RSMFH. We employ an efficient discrete optimization strategy to solve the proposed model. Massive experiments on four benchmark databases show that our approach achieves promising results in cross-modal retrieval tasks.

Keywords Hashing · Cross-modal retrieval · Matrix factorization · Supervised · Label information · Representation · Sparse · Iterative

1 Introduction

In the past few years, the focus of retrieval technology is transferred from single-modal to cross-modal owing to the massive growth of multimedia data. Actually, the multimodal data are often not independent, but have essential connections. Therefore, it has become a hot topic to learn the correlation information between multimodality, which is referred to as the heterogeneity gap. Recently, researchers have made many efforts to bridge the heterogeneity gap between multimodality and achieved promising results in many real applications [1–6].

Hashing retrieval has been extensively investigated owing to its powerful performance in large-scale search tasks [7–9]. It aims to project original samples into compact binary codes, which preserves their similarity in the Hamming space. Therefore, it is an effective way to represent and search for massive data using binary codes due to its high storage and computational efficiency. As a popular data-independent method, Local Sensitive Hashing (LSH) [10] adopts the random projection as the hash function. Spectral hashing (SH) [11] is a well-known data-dependent method and learns the hash codes by thresholding the Laplace Beltrami eigenfunction of the manifold. Sparse hashing [12] adopts a non-negative sparse coding method to transform the original sample into a low-dimensional representation, and then it is embedded into a binary code.

In cross-modal retrieval applications, the hashing technology can be divided into supervised and unsupervised learning methods. Unsupervised methods explore correlations from heterogeneous data by using binary codes. They

✉ Zhenqiu Shu
shuzhenqiu@kust.edu.cn

[1] Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China

[2] Jiangsu Provincial Engineering Laboratory of Pattern Recognition and Computational Intelligence, Jiangnan University, Wuxi 231001, China

[3] The College of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou, China

include graph-based methods and matrix factorization-based methods. The former maintains the correlations of hash codes by constructing similar graphs, but it needs to take expensive computational costs [12–14]. The latter seeks the latent semantic correlations of multi-modalities [15–19]. Therefore, they can avoid the construction of a large-scale affinity matrix and thus greatly reduce the computational complexity. To improve the search performance by considering the label information, some supervised hashing methods based on cross-modal retrieval have been proposed in the past few years. Although the supervised hashing models have improved considerably over the unsupervised hashing models, it can be found that the supervised hashing models still exist the following challenges. Firstly, the hash codes are learned from the latent common representation and the specific properties of each modality are neglected during matrix decomposition. Therefore, it leads to achieving unsatisfactory retrieval performance. Secondly, to improve the retrieval performance of hashing algorithm, the supervised label information of multimodality data should be embedded in the hash codes in real applications. Thirdly, most of them ignore the discreteness of the hash codes in the optimization procedure. The relaxation strategy is used to obtain a continuous solution, and then quantify the continuous solution to generate a hash code. However, this strategy leads to large quantization errors and thus reduces the performance in real applications.

In this work, a novel cross-modal retrieval method, called Robust Supervised Matrix Factorization Hashing (RSMFH), is proposed to address the aforementioned challenges. It maintains both the shared and the specific attributes of multi-modalities by decomposing each modality into a consistent representation and an inconsistent representation. Specifically, the inconsistency may be caused by the noise and diversity of different modes in the training data points. To reduce the impact of this inconsistency on cross-modal retrieval, we impose sparse constraints on the inconsistency of each modality. Therefore, the robustness of our proposed model can be improved. In addition, the proposed RSMFH method learns hash codes from the shared latent semantic representations and considers the supervised label information, simultaneously. We develop an effective discrete scheme to optimize the proposed model. Figure 1 shows the framework of our RSMFH method in cross-modal retrieval. The

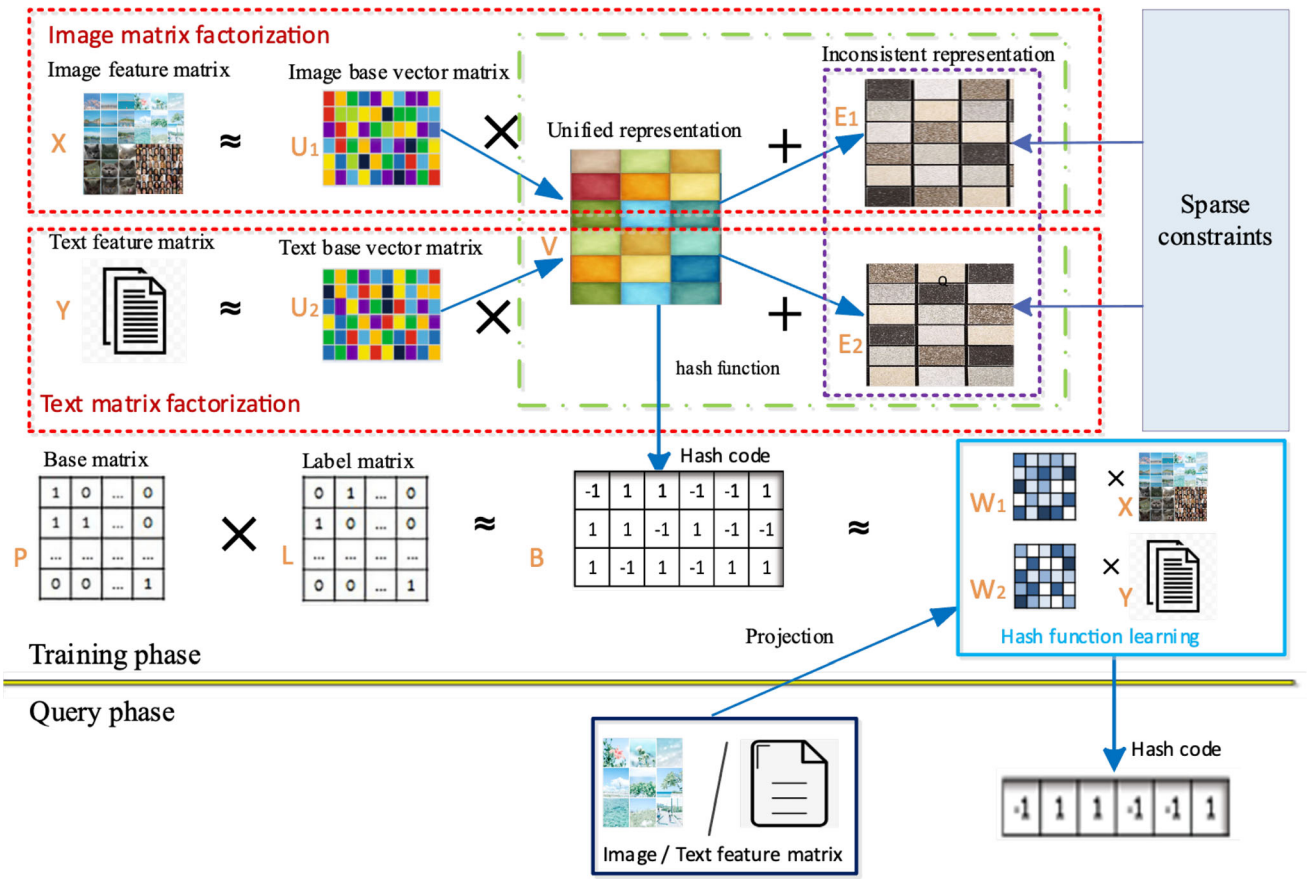

Fig. 1 The framework of our RSMFH approach

experimental results show that our RSMFH approach can achieve superior retrieval performance.

The contributions of this work can be highlighted as follows:

(1) We present a unified matrix factorization framework for learning hash codes. It decomposes each modality into a consistent representation and an inconsistent representation. Then hash codes are generated from the consistent representation of the multimodal data. In addition, we impose the sparse constraints on the inconsistent parts and minimize their inner product, simultaneously. Therefore, the proposed RSMFH approach further considers the specific attributes of different modalities compared with traditional methods.
(2) To consider the supervised label information, our approach improves the accuracy by embedding the supervised information into the hash codes. Therefore, more discriminative hash codes are generated from our proposed model.
(3) We present a discrete optimization scheme to solve our proposed model and then give its complexity analysis. Comprehensive experimental results on four benchmark datasets have shown the superiority of our RSMFH method.

The remainder of the paper is organized as follows: Sect. 2 introduces previous work of cross-modal retrieval. Section 3 details our approach. Section 4 gives the experimental result and the analysis. Section 5 draws a conclusion of this work.

2 Related work

This section provides a preliminary introduction to the related work of cross-modal hashing.

2.1 Cross-modal unsupervised hashing

In the real world, most of the multimodal data are unlabeled and it would take a lot of labor and time to label. Therefore, unsupervised hashing approaches have attracted extensive attention in cross-modal retrieval. Linear cross-modal hashing (LCMH) [7] is a typical graph-based method and utilizes the anchor graph to preserve the similarity of both intra-model and inter-model in the Hamming space. However, its disadvantage is that it needs expensive time cost to construct the similarity graph. Matrix factorization-based cross-modal retrieval methods aim to seek the latent correlations semantic hidden in multimodal data. Ding et al [17] proposed to learn unified hash codes generated from the common representation, which is obtained by using the collect matrix factorization (CMF). Latent semantic sparse hashing (LSSH) [20] was proposed to learn hash codes by integrating matrix factorization and sparse coding. Semantic topic multimodal hashing (STMH) [18] adopts the robust matrix factorization to generate hash codes. Wang et al [21] proposed to learn hash codes using discrete matrix factorization. These methods, such as CMFH, STMH and RFDH, learn the hash codes by finding the common representation of multimodality. LSSH learns separate hash codes that tend to keep the particular space of every modality. Joint and individual matrix factorization hashing (JIMFH) [19] and discrete robust matrix factorization hashing (DRMFH) [22] explore both the shared and the modal-specific properties of multimodal data. The difference is that the former decomposes the multimodal data twice and neglects the discreteness of the hash codes. The latter not only decomposes the multimodal data only once, but also obtains the discrete hash codes. However, the aforementioned methods are completely unsupervised learning hashing retrieval ones, and cannot use the supervised information to further improve the retrieval performances.

2.2 Cross-modal supervised hashing

Different from the above-mentioned methods, supervised hashing methods attempt to obtain more semantic relevance from supervised label information to improve retrieval accuracy. As a typical supervised hashing method, cross-modal similarity-sensitive hashing (CMSSH) [23] was proposed to use a binary classification approach to learn hash codes. Kumar et al [24] extend spectral hashing to multiple modalities and aim to maintain both intra-modal and inter-modal correlations. Semantic correlation maximization (SCM) [25] maximizes the correlations between different modalities to generate the hash function. Semantic preserving hashing algorithm (SePH) [26] learns hash codes by minimizing the Kullback–Leibler (KL) divergence of a probability distribution. Wang et al [27] proposed to consider both the local geometric structure of each modality and the label information across different modalities. Generalized semantic preservation hashing (GSePH) [28] learns hash codes by capturing the semantic similarity of different modalities. Label consistent matrix factorization hashing (LCMFH) [29] imposes the label information to constrain matrix decomposition. Label category supervised matrix factorization hashing (LCSMFH) [30] not only maintains both the inter-modal and the intra-modal similarities of original samples, but also utilizes label information to enhance the discriminative ability. Label consistent flexible matrix factorization hashing (LFMH) [31] can jointly learn modality-specific latent semantic

spaces with similar semantics through flexible matrix factorization. Three supervised methods, LCMFH, LFMH and LCSMFH learn a unified representation that tends to preserve the shared properties of multimodal data. However, they neglect the specific properties of multi-modality data in hash code learning. In this paper, a new supervised multimodal hashing method, named robust supervised matrix factorization hashing (RSMFH), is proposed to preserve the shared properties and specific properties of multimodal data. In addition, compared with the aforementioned three methods, our proposed method directly learns hash codes and optimizes our proposed model with an efficient discrete optimization scheme. Thereby, it can keep the discreteness of the hash codes and effectively reduces the quantization loss.

3 Robust supervised matrix factorization hashing (RSMFH)

This section introduces the proposed model (RSMFH) in detail. Figure 1 plots the framework of the proposed RSMFH approach. Specifically, it is divided into two steps: training and retrieval. In the training step, each modality is decomposed into a shared latent semantic representation and an inconsistent specific representation. Then the proposed RSMFH approach directly learns hash codes from both the shared latent semantic representation and the supervised label information, and gets the mapping matrix of each modality. In the second step, we use the mapping matrix learned in the training step to project the query sample to generate a hash code, and then the retrieval task is performed based on this hash code.

4 Notations

In this paper, we take image modality and text modality as an example. Given a set of multimodal data is $O = \{o_i\}_{i=1}^{n}$, where $o_i = (x_i, y_j)$ is a multimodal data point, and x_i and y_i are the feature vectors of the $i-$ th instance of two modalities, respectively. $X = \{x_1, x_2, \ldots, x_n\} \in R^{d_x \times n}$ and $Y = \{y_1, y_2, \ldots, y_n\} \in R^{d_y \times n}$ are the feature matrices of two modalities, where n denotes the number of samples, d_x and d_y are the dimensionality of image samples and text samples, respectively, and $d_x \neq d_y$. In general, we set the centers of samples to zero, i.e., $\sum_{i=1}^{n} x_i = 0$, $\sum_{i=1}^{n} y_i = 0$.

Besides, we use $L = \{l_1, l_2, \ldots, l_n\} \in R^{c \times n}$ to represent the label matrix, where c is the number of categories. If the i-th sample belongs to class j, then$l_{ij} = 1$; otherwise,$l_{ij} = 0$. Assuming that the length and the matrix of hash code are k and$B = \{b, b_2, \ldots, b_n\} \in R^{k \times n}$, respectively. $\|\cdot\|$ is the Frobenius norm and $sgn(\cdot)$ denotes the sign function.

4.1 The proposed method

(a) Matrix factorization

Several studies have demonstrated that matrix factorization approaches can effectively explore the common representation of multimodality. Ding et al [17] firstly applied CMF method for cross-modal retrieval. It learns the hash codes from the shared properties of multimodal data. However, the specific properties of each modality are ignored. To solve this issue, the proposed method decomposes each multimodality into a consistent representation and an inconsistent representation. Given the feature matrices X and Y of two modalities, they can be decomposed into the product of $U_1 = [u_{11}, u_{12}, \ldots, u_{1r}]^{d_x \times r}$ and $V = [v_1, v_2, \ldots, u_n]^{r \times n}$, and the product of $U_2 = [u_{21}, u_{22}, \ldots, u_{2r}]^{d_y \times r}$ and $V = [v_1, v_2, \ldots, u_n]^{r \times n}$, respectively. Therefore, the model of CMF is given as follows:

$$\begin{cases} X \approx U_1(V + E_1) \\ Y \approx U_2(V + E_2) \end{cases}, \tag{1}$$

where U_1 and U_2 denote the basis matrices of the two modalities, respectively. V stands for the unified representation matrix. $E_1 \in R^{k \times n}$ and $E_2 \in R^{k \times n}$ are the inconsistent parts of the two modalities, respectively.

Using the Euclidean distance as the metric, Eq. (1) can be rewritten as the following minimization problem:

$$F_1 = \lambda X - U_1(V + E_1)_F^2 + (1 - \lambda)Y - U_2(V + E_2)_F^2. \tag{2}$$

An ideal hypothesis is that the inconsistent parts of multi-modalities should have a gap as large as possible. To achieve this goal, the inner product of both E_1 and E_2 is minimized and then added into Eq. (2). Thus, we can further rewrite Eq. (2) as follows:

$$\begin{aligned} F_1 = {} & \lambda X - U_1(V + E_1)_F^2 + (1 - \lambda)Y \\ & - U_2(V + E_2)_F^2 + \alpha Tr\left(E_1 E_2^T\right), \end{aligned} \tag{3}$$

where $Tr(.)$ denotes the matrix trace, and α stands for the nonnegative parameter.

(b) Sparse constraint

Many studies on sparse representation theory have been shown that $l_{2,1}$-norm constraint is effective in reducing the effect of noise and outliers [32–34]. The inconsistency of each modality can be considered as a special kind of noise, and thus we impose the $l_{2,1}$-norm-based sparse constraint on both E_1 and E_2. Therefore, the loss function F_2 is given as follows:

$$F_2 = E_{12,1} + E_{22,1} \tag{4}$$

(c) Hash codes scheme

Here, we consider learning the hash codes from the common representation of the multimodality. By introducing an auxiliary matrix, the latent unified representation of different modalities is mapped to hash codes. Therefore, the loss function $\mathcal{F}_3$ is expressed as follows:

$$F_3 = B - RV_F^2 \; s.t. \; RR^T = I, B \in \{-1, 1\}^{k \times n}. \tag{5}$$

where $R \in R^{k \times k}$ is an projection matrix.

To generate more discriminative hash codes, the label information of the data is fully considered. Specifically, the learned hash codes B in Hamming space can be reconstructed using a certain basis set. Therefore, the relationship of both the label information L and the hash codes B can be represented as follows:

$$F_4 = \beta B - PL_F^2 \; s.t. \; B \in \{-1, 1\}^{k \times n}. \tag{6}$$

where $P \in R^{k \times c}$ denotes a basis matrix, and β denotes a nonnegative parameter.

(d) Overall objective function.

Kernel trick aims to map the original samples into a high-dimensional feature space, and is suitable for dealing with the linear non-separable problem [35, 36]. In particular, $\phi(X) = [\phi(x_1), \phi(x_2), \ldots, \phi(x_n)]$, $\phi(Y) = [\phi(y_1), \phi(y_2), \ldots, \phi(y_n)]$ are the kernel feature matrices of two modalities, respectively. Here, $\phi(\cdot)$ denotes the RBF kernel function. Therefore, the kernel features $\phi(x_i)$ and $\phi(y_i)$ are given as

$$\begin{aligned} \phi(x_i) &= \left[\exp\left(-\frac{x_i - \alpha_1^{(1)2}}{2\sigma_{(1)}^2}\right), \ldots, \exp\left(-\frac{x_i - \alpha_m^{(1)2}}{2\sigma_{(1)}^2}\right)\right], \\ \phi(y_i) &= \left[\exp\left(-\frac{y_i - \alpha_1^{(2)2}}{2\sigma_{(2)}^2}\right), \ldots, \exp\left(-\frac{y_i - \alpha_m^{(2)2}}{2\sigma_{(2)}^2}\right)\right], \end{aligned} \tag{7}$$

where $\left\{\alpha_j^{(t)}\right\}_{j=1}^m (t = 1,2)$ denotes m anchor points. $\sigma_{(1)} = \frac{1}{nm}\sum_{i=1}^n \sum_{j=1}^m x_i - \alpha_j^{(1)}$ and $\sigma_{(2)} = \frac{1}{nm}\sum_{i=1}^n \sum_{j=1}^m y_i - \alpha_j^{(2)}$ denotes the kernel widths of two motilities, respectively. For convenience, we replace $\phi(X)$ and $\phi(Y)$ by using X and Y, respectively. Thus, we get

$$X = \phi(X), \; Y = \phi(Y). \tag{8}$$

By integrating Eqs. (3), (4), (5) and (6), the overall objective function of the proposed RSMFH approach is given as follows:

$$\begin{aligned} &\min_{U_1, U_2, E_1, E_2, P, R, V, B} FU_1, U_2, E_1, E_2, P, R, V, B \\ &= F_1 + F_2 + F_3 + F_4 \\ &= \lambda X - U_1(V + E_1)_F^2 + (1 - \lambda)Y - U_2(V + E_2)_F^2 \\ &\quad + \alpha Tr\left(E_1 E_2^T\right) + \mu B - RV_F^2 + \beta B - PL_F^2 \\ &\quad + \gamma E_{12,1} + \gamma E_{22,1} + \gamma R(U_1, U_2, P, V) \\ &s.t. RR^T = I, B \in \{-1, 1\}^{k \times n}. \end{aligned} \tag{9}$$

where $R(.) = \cdot_F^2$ aims to avoid overfitting, and γ is the regularization parameter.

4.2 Algorithm optimization

Obviously, Eq. (9) is nonconvex and it is impossible to get its global optimization solution. We can update one variable while fixing other variables in this paper. Therefore, Eq. (9) is solved by the following steps:

Step-1: update U_1 by fixing $U_2, E_1, E_2, P, R, V, B$. Equation (9) with respect to U_1 can be simplified as follows:

$$\min_{U_1} \lambda X - U_1(V + E_1)_F^2 + \gamma R(U_1). \tag{10}$$

By setting the partial derivative w.r.t. U_1 to zero, we can derive the closed solution of U_1 as follows:

$$U_1 = X(V + E_1)^T \left((V + E_1)(V + E_1)^T + \frac{\gamma}{\lambda} I\right)^{-1}. \tag{11}$$

Step-2: Update U_2 by fixing $U_1, E_1, E_2, P, R, V, B$. Equation (9) becomes the following form:

$$\min_{U_2} (1 - \lambda)Y - U_2(V + E_2)_F^2 + \gamma R(U_2) \tag{12}$$

Similarly, we can get the closed solution of U_2 as follows:

$$U_2 = Y(\mathrm{V} + E_2)^T \left((V + E_2)(\mathrm{V} + E_2)^T + \frac{\gamma}{1 - \lambda} I\right)^{-1}. \tag{13}$$

Step-3: Update E_1 by fixing $U_1, U_2, E_2, P, R, V, B$. Equation (9) can be simplified as follows:

$$\min_{E_1} \lambda X - U_1(V + E_1)_F^2 + \gamma E_{12,1} + \alpha Tr\left(E_1 E_2^T\right). \tag{14}$$

To solve the $l_{2,1}$-norm-based optimization problem, we first introduce the following weighting matrix:

$$D_{ii}^{(1)} = \frac{1}{2E_{12}^i}, \tag{15}$$

where $D_{ii}^{(1)}$ is the i-th diagonal element of D_1 and E_1^i is the i-th row of matrix E_1. Then, Eq. (14) can be restated as

$$\min_{E_1} \lambda X - U_1(V + E_1)_F^2 + \gamma Tr\left(E_1^T D_1 E_1\right) + \alpha Tr\left(E_1 E_2^T\right). \tag{16}$$

By calculating the partial derivative of Eq. (16) for E_1 to zero, the closed solution of E_1 is derived as follows:

$$E_1 = \left(U_1^T U_1 + \frac{\gamma}{\lambda} D_1\right)^{-1} \left(U_1^T (X - U_1 V) - \frac{\alpha}{\lambda} E_2\right). \tag{17}$$

Step-4: Update E_2 by fixing $U_1, U_2, E_1, P, R, V, B$. We rewrite Eq. (9) as follows:

$$\min_{E_2} (1 - \lambda) Y - U_2(V + E_2)_F^2 + \gamma E_{22,1} + \alpha Tr\left(E_1 E_2^T\right). \tag{18}$$

Similarly, we first introduce another weighting matrix as follows:

$$D_{ii}^{(2)} = \frac{1}{2E_{22}^i}, \tag{19}$$

where $D_{ii}^{(2)}$ denotes the $i-$ th diagonal element of D_2 and E_2^i stands for the $i-$ th row of matrix E_2. Then Eq. (18) can be re-expressed as follows:

$$\min_{E_2} (1 - \lambda) Y - U_2(V + E_2)_F^2 + \gamma Tr\left(E_2^T D_2 E_2\right) + \alpha Tr\left(E_1 E_2^T\right). \tag{20}$$

By setting the derivative of Eq. (20) w.r.t. E_2 to zero, the closed solution of E_2 is given as

$$E_2 = \left(U_2^T U_2 + \frac{\gamma}{1 - \lambda} D_2\right)^{-1} \left(U_2^T (Y - U_2 V) - \frac{\alpha}{\lambda} E_1\right). \tag{21}$$

Step-5: Update P b y fixing $U_2, U_1, E_1, E_2, R, V, B$. Equation (9) can be simplified as:

$$\min_P \beta B - P L_F^2 + \gamma R(P). \tag{22}$$

We take the partial derivative of Eq. (22) with respect to P and set it to zero. Therefore, the closed solution of P is derived as follows:

$$P = \left(\beta B L^T\right)\left(\beta L L^T + \gamma I\right)^{-1}. \tag{23}$$

Step-6: Updating V b y fixing $U_2, U_1, E_1, E_2, R, P, B$. Equation (9) can be simplified as:

$$\min_V \lambda X - U_1(V + E_1)_F^2 + (1 - \lambda) Y - U_2(V + E_2)_F^2 + \mu B - R V_F^2 + \gamma R(V) \tag{24}$$

By adopting a similar solution scheme, the closed solution of V is given as follows:

$$V = \left(\lambda U_1^T U_1 + (1 - \lambda) U_2^T U_2 + R^T R + \gamma I\right)^{-1} \times \left(\lambda U_1^T (X - U_1 E_1) + (1 - \lambda) U_2^T (Y - U_2 E_2) + R^T B\right). \tag{25}$$

Step-7: Update B b y fixing $U_2, U_1, E_1, E_2, R, P, V$. Equation (9) can be simplified as

$$\min_B \mu B - R V_F^2 + \beta B - P L_F^2 \quad s.t. B \in \{-1, 1\}^{k \times n}. \tag{26}$$

Equation (26) can be written equivalently as follows:

$$\min_B \mu\left(Tr\left(B^T B\right) - 2Tr\left(B^T R V\right) + Tr\left(V^T R^T R V\right)\right) + \beta\left(Tr\left(B^T B\right) - 2Tr\left(B^T P L\right) + Tr\left(L^T P^T P L\right)\right) \quad s.t. B \in \{-1, 1\}^{k \times n}. \tag{27}$$

By removing the irrelevant parts with the variable B, Eq. (27) can be written as follows:

$$\arg\min_B \mu Tr\left(B^T B\right) - 2\mu Tr\left(B^T R V\right)) + \beta Tr\left(B^T B\right) - 2\beta Tr\left(B^T P L\right) \; s.t. \; B \in \{-1, 1\}^{k \times n}. \tag{28}$$

Since $Tr(B^T B)$ is a constant, the closed solution of B is given as follows:

$$B = \text{sgn}(\mu R V + \beta P L). \tag{29}$$

Step-8: Update R by fixing $U_2, U_1, E_1, E_2, B, P, V$. We can simplify Eq. (9) as

$$\min_R \mu B - R V_F^2 \; s.t. \; R R^T = I \tag{30}$$

It is clear to see that Eq. (30) is a classic orthogonal Procrustes problem and thus can be optimized through solving SVD. We have $B V^T = W \Omega \overline{W}^T$ and the solution can be obtained as follows:

$$R = W \overline{W}^T. \tag{31}$$

In summary, Algorithm 1 describes the solution steps of our RSMFH approach in detail.

Algorithm 1 RSMFH

Training stage

Input: Hash code length k, label matrix L, feature matrices X and Y of two modalities, and parameters λ, α, β, γ, μ.

Output: hash codes B, mapping matrices W_1 and W_2 .

Procedure:

1. Calculate $\phi(X)$ and $\phi(Y)$.

2. Initialize V, E_1, E_2, B, D_1, D_2, P and R.

Repeat

(1) Update U_1 by Eq. (11).

(2) Update U_2 by Eq. (13).

(3) Update D_1 by Eq. (15).

(4) Update E_1 by Eq. (17).

(5) Update D_2 by Eq. (19).

(6) Update E_2 by Eq. (21).

(7) Update P by Eq. (23).

(8) Update V by Eq. (25).

(9) Update B by Eq. (29).

(10) Update R by Eq. (31).

Until reaching the maximum iteration or convergence.

4. Learn mapping matrices W_1 and W_2 by Eq. (34) and Eq. (35), respectively.

Retrieval Stage

Input: The feature matrix X_{quey} and Y_{quey} of the retrieved data, mapping matrices W_1 and W_2 .

Output: B_x and B_y.

Procedure:

1. Calculate $\phi(X_{quey})$ and $\phi(Y_{quey})$.

2. For X_{quey}: calculate hash code by $B_x = sgn(W_1 X_{quey})$.

For Y_{quey}: calculate hash code by $B_y = sgn(W_2 Y_{quey})$.

4.3 Hash function

As mentioned previously, our RSMFH approach includes hash function learning and hash coding learning separately. Specifically, we use Algorithm 1 to get the optimal hash codes B, and then learn modality-specific hash functions to deal with the out-of-sample problem. Generally, the hash functions are learned by minimizing the following least-squares regression problems:

$$\min_{W_1} B - W_1 X_F^2 + \gamma R(W_1), \tag{32}$$

$$\min_{W_2} B - W_2 Y_F^2 + \gamma R(W_2). \tag{33}$$

By setting the partial derivatives of Eqs. (34) and (35) w.r.t. W_1 and W_2 to zero, respectively, the closed solutions of W_1 and W_2 are given as follows:

$$W_1 = \left(BX^T\right)\left(XX^T + \gamma I\right)^{-1}, \tag{34}$$

$$W_2 = \left(BY^T\right)\left(YY^T + \gamma I\right)^{-1}. \tag{35}$$

Given a query data X_{quey} or Y_{quey}, their hash codes are obtained according to the following formula:

$$B_x = \text{sgn}\left(W_1 X_{\text{quey}}\right), \tag{36}$$

$$B_y = \text{sgn}\left(W_2 Y_{\text{quey}}\right). \tag{37}$$

4.4 Complexity analysis

In this subsection, we present the complexity analysis of our RSMFH approach. As we know from Sect. 3.3, the overall complexity of the proposed optimization scheme consists of updating $U_2, U_2, E_1, E_2, P, R, V, B$. Specifically, the computational complexity of updating U_2, U_2, E_1, E_2 is $O(((mk + k^2)n + k^3 + mk^2)t)$. We need $O((kcn + c^2n + c^3 + kc^2)t)$ and $O((k^2n + k^3)t)$ to update P and R, respectively. The cost of updating V and B is $O((2k^2m + 2k^3 + 2k^2n + 4kmn)t)$ and $O((k^2n + kcn)t)$, respectively. Here, c is the number of categories and m is the number of anchor points. k is the length of the hash code. t is the number of update iterations, and $c, m, k \ll n$. Therefore, the overall complexity of our RSMFH method is linear with n (the size of the training dataset).

5 Experiments

5.1 Datasets

LabelMe [37]: The dataset consists of 2688 outside scenes from eight categories. We use a 245-dimensional phrase frequency and a 512-dimensional GIST feature to describe each sample of text and image modalities, respectively. In our experiments, 2016 image-text pairs are randomly selected as the training dataset, and the rest 672 image-text pairs are as the testing dataset.

UCI Handwritten Digit (UCI) [38]: The dataset includes handwritten numerals (0–9). Each sample of image modality and text modality is described by a 76-dimensional vector and a 64-dimensional vector, respectively. In the experiments, we randomly sampled 1500 image-text pairs for training and the remaining 500 image-text pairs were used as a testing dataset.

Pascal sentences [39]: This dataset consists of 1000 image-text pairs divided with 20 categories from VOC 2008. We utilize 5 separate sentences to describe each image. For a fair comparison, we randomly selected 800 image-text pairs as the training dataset (40 pairs per class), and 100 image-text pairs as the testing dataset (5 pairs per class).

Wiki [40]: It contains 2866 image-text pairs collected from Wikipedia. The AlexNet and the Latent Dirichlet Allocation (LDA) model are used to extract the features of all image and the text, respectively. Then each image and each text can be represented by a 128-dimensional vector and a 10-dimensional topic vector, respectively. The training and testing datasets contain 2173 samples and 693 samples, respectively. Table 1 shows the statistics of the four data sets.

5.2 Baselines and implementation details

To evaluate the effectiveness of RSMFH, we select several state-of-the-art cross-modal hashing methods as the comparison algorithms.

- *CCA* [41]: This method learns hash codes by finding the linear relationship of multimodal data.
- *SCM_orth and SCM_seq* [25]: The goal of SCM is to make the distance of hash codes equal to the similarity of label vectors. SCM_orth uses orthogonal projection to learn hash codes. SCM_seq represents a sequential learning method for generating hash codes.

Table 1 Statistics of the four data sets

Statistics	LabelMe	UCI	Pascal sentences	Wiki
Total size	2688	2000	1000	2866
Training dataset size	2016	1500	800	2173
Query dataset size	672	500	100	693
Category	8	10	20	10
Image feature	512-D	76-D	4096-D	128-D
Text feature	245-D	64-D	300-D	10-D

- *CMFH* [17]: It learns the unified hash codes by collective matrix factorization.
- *STMH* [18]: This method learns the modal-specific hash codes by matrix factorization and topic model for image and text, respectively.
- *JIMFH* [19]: It embeds the shared and specific attributes of multimodal data into the learned hash codes, thus improving the retrieval performance.
- *DRMFH* [22]: This method formulates the consistency and inconsistency across different modalities into a matrix factorization model, improving the retrieval performance.
- *OCMFH* [42]: It learns discriminative hash codes for streaming data by CMF in an online optimization scheme.
- *DCH* [43]: It directly learns discriminative binary codes while retaining the discrete constraints using label information to guide hash code learning.
- *SMFH* [27]: It considers the local geometric structure of each modal and the label information among multi-modality data.
- *LCMFH* [29]: It imposes the label information to constrain matrix decomposition.
- *LCSMFH* [30]: This method not only maintains both the inter-modal and the intra-modal similarities of original samples, but also utilizes label information to enhance the discriminative ability.
- *LFMH* [31]: It can jointly learn modality-specific latent semantic spaces with similar semantics through flexible matrix factorization.

Among them, CCA, CMFH, STMH, DRMFH, OCMFH and JIMFH are unsupervised learning methods, and SCM_seq, SCM_orth, DCH, SMFH, LCMFH, LCSMFH, LFMH and our proposed RSMFH fully consider the supervised information. In this experiment, the codes of DRMFH and LCSMFH are reproduced by ourselves, and the source codes of the rest are publicly available. To comprehensively validate the effectiveness of our proposed model, we perform two common tasks: (1) Txt2Img: search for the images using text; (2) Img2Txt: search for the text using images.

5.3 Evaluation metrics

The first well-known evaluation metric is the mean of Average Precision (mAP). Given a query and the retrieved results, the definition of average precision (AP) is given as follows:

$$AP = \frac{1}{N}\sum_{r=1}^{R} P(r)\delta(r), \tag{38}$$

where N denotes the number of relevant instances in the query set, and $P(r)$ is the precision of the top r-th retrieval instance. $\delta(r) = 1$ if the r-th retrieval instance is a neighbor of the query, otherwise, $\delta(r) = 0$. In our experiments, R is empirically set to 50, and thus it is noted as mAP@50. Due to the randomness of the initialization of the variables, we run the algorithms five times and reported their average values as the final results.

Another well-known metric is the Precision-Recall (PR) curve, whose goal is to show the tradeoff between recall and precision. In general, the greater the Precision-Recall curve, the higher the retrieval performance. Its detailed description can be referred to in [44].

5.4 Experimental results

Four lengths of binary codes are used in the experiments to verify the retrieval performances. Specifically, the lengths of hash codes are empirically set with difference values, such as 16 bits, 32 bits, 64 bits and 128 bits. The results of all approaches on four multi-modal datasets are presented in this subsection.

5.4.1 Results on LabelMe database

The first experiment was carried out on the LabelMe database. The mAP values of both RSMFH and other competitors on the LabelMe dataset are reported in Table 2, and their PR curves are shown in Fig. 2. In this experiment, we empirically varied the length of hash codes from 16 to 128 bits. From Table 2, it can be seen that our proposed RSMFH performs better than other competitors in both Txt2Img and Img2Txt tasks on the LabelMe dataset. Compared with the best performances among competitors, our RSMFH approach is improved by 4.31%, 2.98%, 2.98%, 2.46% in the Img2Txt task, respectively, and 3.29%, 1.89%, 1.17%, and 1% in the Txt2Img task, respectively. In addition, it is worth noting that the performance of DRMFH is higher than that of JIMFH. This is because the discrete optimization of the DRMFH model plays an important role during the optimization process. Moreover, we can find that our proposed RSMFH method outperforms the DRMFH method, which fully demonstrates the effectiveness of embedding the label information into hash codes. In addition, it can be seen from Fig. 2 that our RSMFH method outperforms other state-of-the-art approaches on different tasks with different hash code lengths. Therefore, it also verifies the effectiveness of our RSMFH approach from another aspect. In our paper, the bold in the tables indicate the best performances.

Table 2 mAP@50 Results on LabelMe

Methods	Txt2Img				Img2Txt			
	16bits	32bits	64bits	128bits	16bits	32bits	64bits	128bits
CCA	0.5127	0.4612	0.3994	0.3743	0.6289	0.5741	0.5553	0.5540
SCM_seq	0.6773	0.7382	0.7421	0.7494	0.8587	0.8760	0.8756	0.8774
SCM_orth	0.4740	0.3628	0.2873	0.2364	0.6443	0.4900	0.3962	0.2364
CMFH	0.6081	0.6262	0.6568	0.6880	0.7524	0.7773	0.7963	0.8028
STMH	0.6236	0.7050	0.7331	0.7485	0.7670	0.8109	0.8225	0.8237
JIMFH	0.6243	0.6762	0.6944	0.7096	0.7688	0.8164	0.8273	0.8320
DRMFH	0.7681	0.7888	0.7937	0.7951	0.8449	0.8711	0.8827	0.8778
OCMFH	0.6181	0.6362	0.6868	0.6910	0.7511	0.7806	0.8055	0.8128
DCH	0.6845	0.7512	0.7515	0.7757	0.8857	0.8887	0.8926	0.8949
SMFH	0.7214	0.7618	0.7895	0.8048	0.8561	0.8675	0.8795	0.8851
LCMFH	0.7282	0.7502	0.7703	0.7813	0.8537	0.8864	0.8849	0.8929
LFMH	0.7729	0.8020	0.8125	0.8221	0.8652	0.8780	0.8798	0.8849
LCSMFH	0.7346	0.7613	0.7889	0.8128	0.8689	0.8857	0.8948	0.9012
RSMFH	**0.8160**	**0.8318**	**0.8424**	**0.8467**	**0.9018**	**0.9046**	**0.9065**	**0.9112**

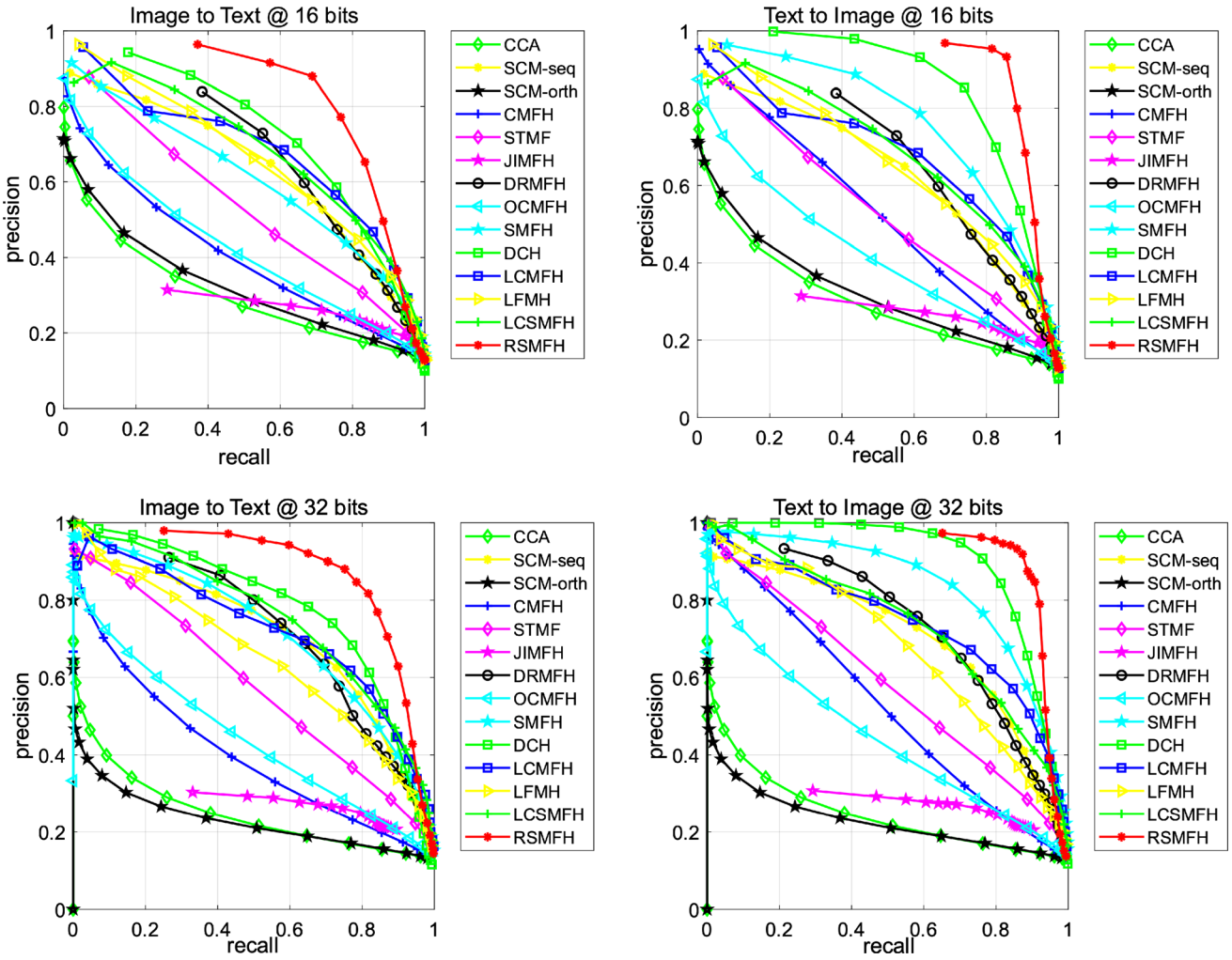

Fig. 2 PR curves varied code length on LabelMe

Table 3 mAP@50 Results on UCI

Methods	Txt2Img			Img2Txt		
	16bits	32bits	64bits	16bits	32bits	64bits
CCA	0.5407	0.4586	0.3652	0.5540	0.5459	0.4685
SCM_seq	0.7196	0.7284	0.7437	0.8774	0.7200	0.7421
SCM_orth	0.5363	0.3942	0.3255	0.2364	0.6075	0.4440
CMFH	0.6632	0.7477	0.7597	0.8028	0.7917	0.8466
STMH	0.6242	0.6513	0.6782	0.8237	0.8014	0.8544
JIMFH	0.5129	0.5249	0.5337	0.8320	0.7456	0.8072
DRMFH	0.7203	0.7838	0.7776	0.8778	0.8134	0.8634
OCMFH	0.6645	0.7377	0.7397	0.8128	0.7727	0.8267
DCH	0.7140	0.7376	0.7547	0.8028	0.8692	0.8902
SMFH	0.6617	0.7057	0.7467	0.8851	0.9090	0.9160
LCMFH	0.7303	0.7480	0.7640	0.8813	0.8929	0.9080
LFMH	0.7563	0.7787	0.7927	0.9007	0.9199	0.9271
LCSMFH	0.7460	0.7520	0.7620	0.9012	0.9032	0.9067
RSMFH	**0.7907**	**0.8058**	**0.8087**	**0.9412**	**0.9487**	**0.9536**

5.4.2 Results on UCI database

The second experiment was carried out on the UCI database. The mAP values of our RSMFH approach and its competitors are shown in Table 3, and the PR curves are plotted in Fig. 3. In particular, we only set the length of hash codes to 16 bits, 32 bits, and 64 bits because the features of the two modalities on the UCI dataset are 76 and 64 dimensions, respectively. It is easy to know from Table 3 that the mAP value of LCSMFH is higher than that of LCMFH. This is because LCSMFH maintains inter-modal and intra-modal similarity by adding graph structure constraint. In addition, it is clear to see that our RSMFH approach outperforms other comparison methods regardless of the length setting of hash codes in both Txt2Img and Img2Txt tasks on the UCI dataset. The main reason is that our RSMFH approach considers more knowledge hidden in multimodal data than other competitors. The PR curves of all approaches are plotted in Fig. 3. Noting that the

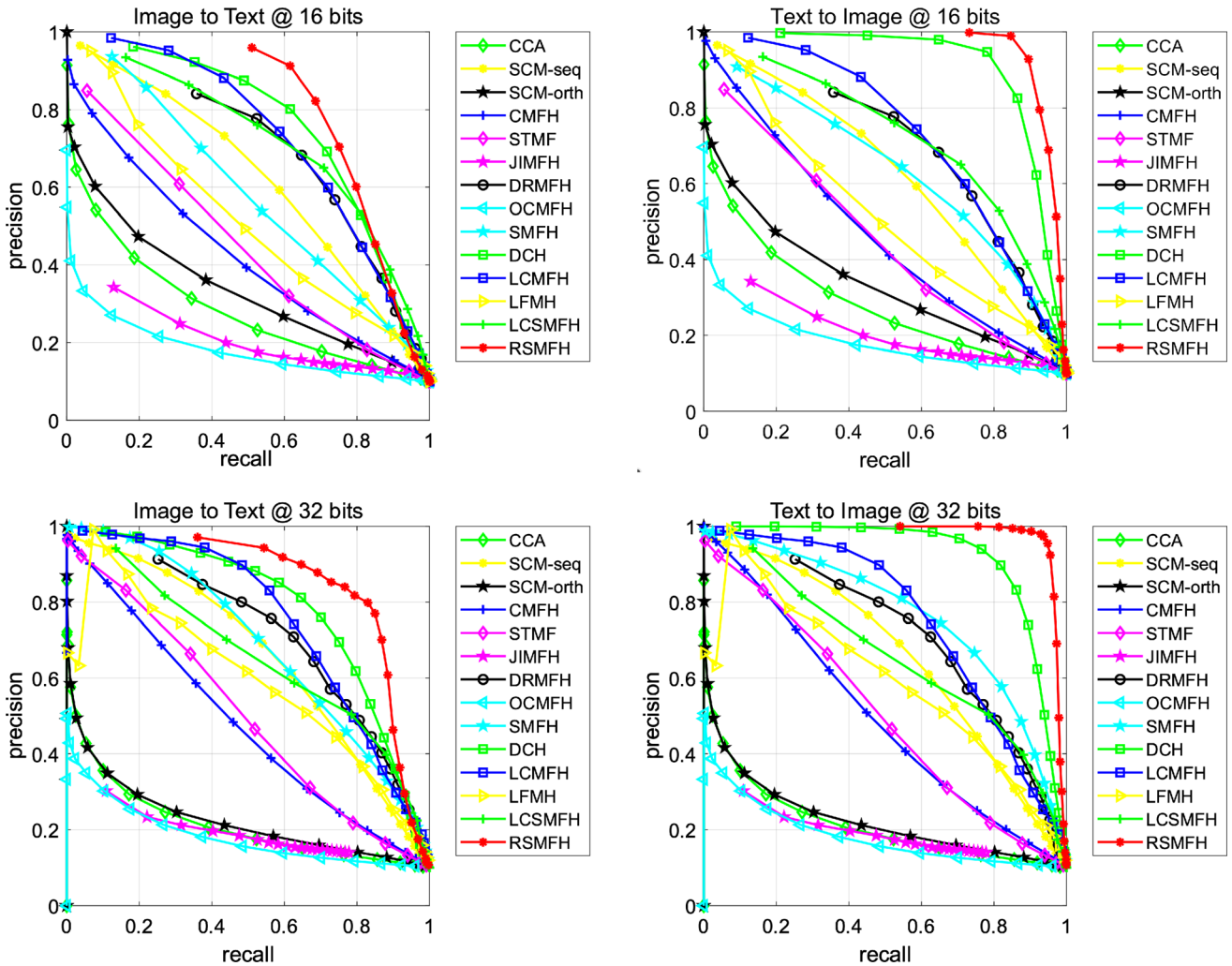

Fig. 3 PR curves varied code length on UCI

Table 4 mAP@50 Results on pascal

Methods	Txt2Img				Img2Txt			
	16bits	32bits	64bits	128bits	16bits	32bits	64bits	128bits
CCA	0.3855	0.3864	0.4263	0.4227	0.4213	0.4703	0.4940	0.4586
SCM_seq	0.5986	0.6453	0.6555	0.6639	0.7264	0.7679	0.7882	0.8029
SCM_orth	0.5542	0.4384	0.3536	0.2524	0.6977	0.5558	0.4140	0.2863
CMFH	0.5402	0.5614	0.5725	0.5972	0.6113	0.6537	0.6825	0.6950
STMH	0.4516	0.4667	0.4972	0.5140	0.5714	0.6023	0.6376	0.6465
JIMFH	0.4331	0.5136	0.5248	0.5282	0.4386	0.4807	0.4838	0.4877
DRMFH	0.6272	0.6491	0.6543	0.6771	0.7235	0.7307	0.7450	0.7525
OCMFH	0.4654	0.4800	0.4890	0.5551	0.5544	0.5909	0.6264	0.6719
DCH	0.6550	0.6590	0.6716	0.7164	0.7937	0.8144	0.8161	0.8194
SMFH	0.5563	0.5790	0.6049	0.6214	0.6438	0.7132	0.7369	0.7381
LCMFH	0.5747	0.5920	0.6320	0.6335	0.7294	0.7560	0.7879	0.7898
LFMH	0.4872	0.6215	0.6438	0.6833	0.7084	0.7842	0.8054	0.8053
LCSMFH	0.5860	0.5972	0.6156	0.6568	0.7334	0.7557	0.7694	0.7821
RSMFH	**0.6724**	**0.7182**	**0.7353**	**0.7406**	**0.8042**	**0.8184**	**0.8266**	**0.8370**

proposed RSMFH method shows better retrieval performance than other state-of-the-art methods in the Img2Txt and Txt2Img tasks. Overall, our RSMFH approach achieves satisfactory performances with different evaluation metrics and different lengths of hash codes on the UCI dataset.

5.4.3 Results on Pascal sentences database

In this experiment, all across-modal retrieval methods were run on the Pascal sentences dataset. Table 4 shows the mAP values of all methods. From Table 4, it should be noted that the most effective unsupervised method and supervised methods are DRMFH and DCH, respectively. Moreover, we can see that our RSMFH approach outperforms both DRMFH and DCH in both Txt2Img and Img2Txt tasks on the Pascal dataset. This is because the proposed RSMFH model not only effectively considers the shared and specific properties of multimodality data, but also embeds the supervised label information in the hash codes. From Fig. 4, it can also be seen that the PR curve of our RSMFH approach is located at the top. Therefore, it indicates that its retrieval performance is higher than other methods using the PR curve metric.

5.4.4 Results on Wiki database

In this subsection, we carried out all cross-modal methods on the Wiki dataset. Their mAP values in the Txt2Img and Img2Txt tasks on the Wiki dataset are summarized in Table 5. From Table 5, it can be observed that our RSMFH approach achieves the best mAP value in two query tasks. One possible reason is that RSMFH can capture more semantic information in text modalities. In addition, the performances of most methods in the Img2Txt task are inferior to that in the Txt2Img task. This is because the image modal loses less information when the multi-modal data are mapped to the hash codes. Finally, we can see that the performances of our RSMFH approach are also improving with the increase of the hash code length. It indicates that the longer the hash code can retain more semantic information. However, some baseline methods, such as CCA, are still inconsistent with the above observations. This phenomenon has been discussed and explained in reference [29]. In addition, the PR curve in Fig. 5 shows the effectiveness of our RSMFH approach in retrieval tasks.

5.4.5 Discussion

From the results on four benchmark datasets, we can draw the following observations:

1) By setting different lengths of hash codes, the proposed RSMFH approach achieves the best mAP values in two query tasks on four datasets (i.e., LabelMe, UCI, Pascal sentences, Wiki). It demonstrates the effectiveness of the proposed RSMFH model for dealing with multimedia data.
2) It can be found that the mAP values of all methods have a relatively slight improvement on the Wiki. One possible explanation is that the semantic differences between the two modalities on the Wiki database are greater than in other databases.
3) We can see that most of the retrieval approaches achieve higher mAP values in Txt2Img than Img2Txt on four datasets. The main reason is that the original

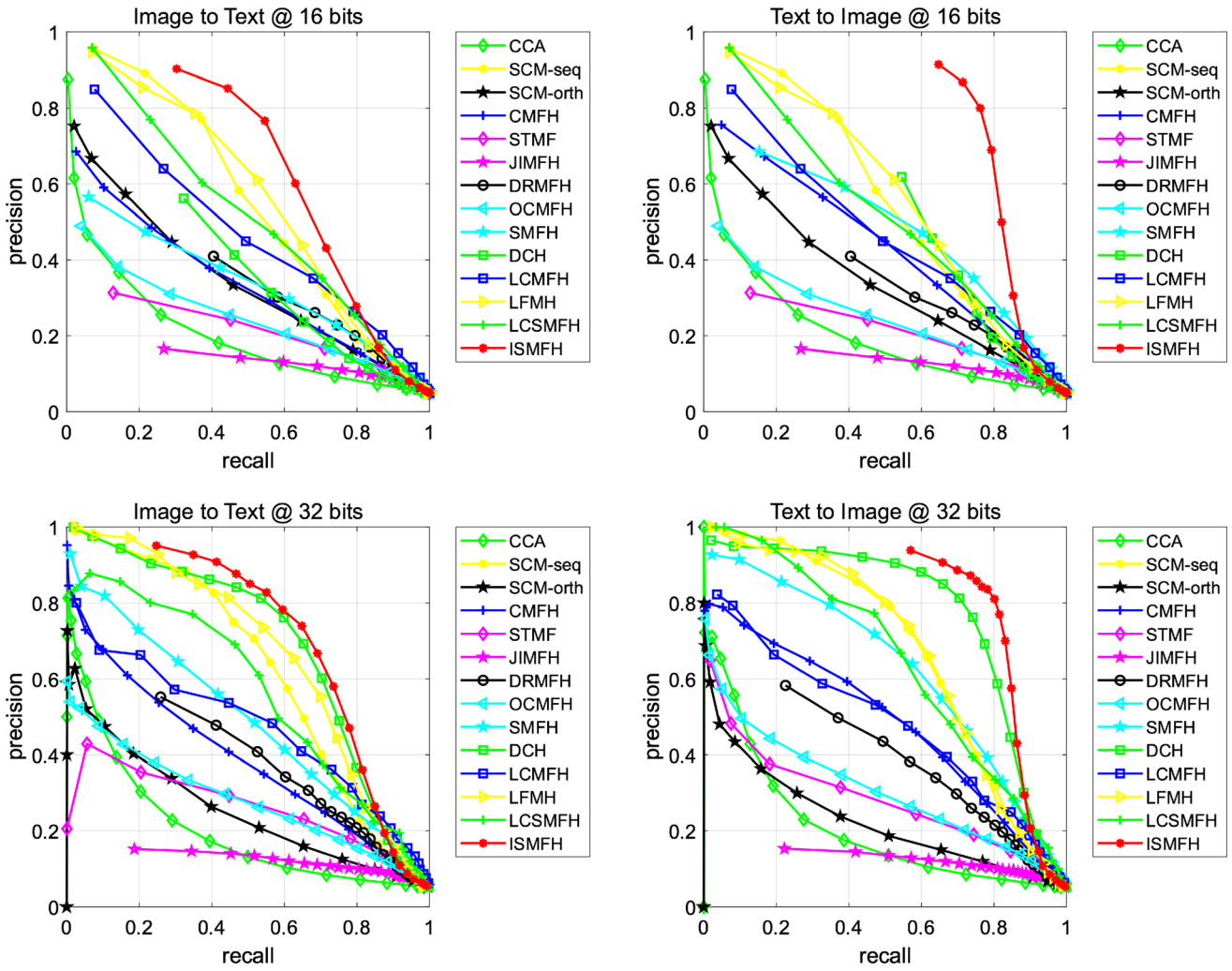

Fig. 4 PR curves varied code length on Pascal

Table 5 mAP@50 Results on Wiki

Methods	Txt2Img				Img2Txt			
	16bits	32bits	64bits	128bits	16bits	32bits	64bits	128bits
CCA	0.2265	0.1814	0.1554	0.1590	0.2902	0.2707	0.2299	0.2020
SCM_seq	0.2474	0.2367	0.2340	0.2634	0.3819	0.4479	0.4253	0.4325
SCM_orth	0.2134	0.1985	0.1941	0.1946	0.2973	0.2540	0.2179	0.2067
CMFH	0.2455	0.2564	0.2593	0.2641	0.6151	0.6303	0.6384	0.6474
STMH	0.2186	0.2368	0.2541	0.2646	0.6155	0.6342	0.6453	0.6536
JIMFH	0.2383	0.2402	0.2531	0.2576	0.6133	0.6272	0.6371	0.6464
DRMFH	0.2476	0.2636	0.2638	0.2687	0.5428	0.6110	0.6227	0.6255
OCMFH	0.2124	0.2289	0.2356	0.2221	0.6069	0.6193	0.6276	0.5998
DCH	0.2360	0.2547	0.2681	0.2714	0.6642	0.6791	0.6824	0.6931
SMFH	0.2285	0.2432	0.2570	0.2688	0.6326	0.6400	0.6628	0.6700
LCMFH	0.2285	0.2385	0.2489	0.2558	0.6442	0.6578	0.6612	0.6659
LFMH	0.2270	0.2441	0.2544	0.2614	0.6228	0.6342	0.6432	0.6421
LCSMFH	0.2315	0.2346	0.2457	0.2552	0.6508	0.6602	0.6643	0.6723
RSMFH	**0.2598**	**0.2674**	**0.2742**	**0.2843**	**0.6745**	**0.6954**	**0.6989**	**0.7012**

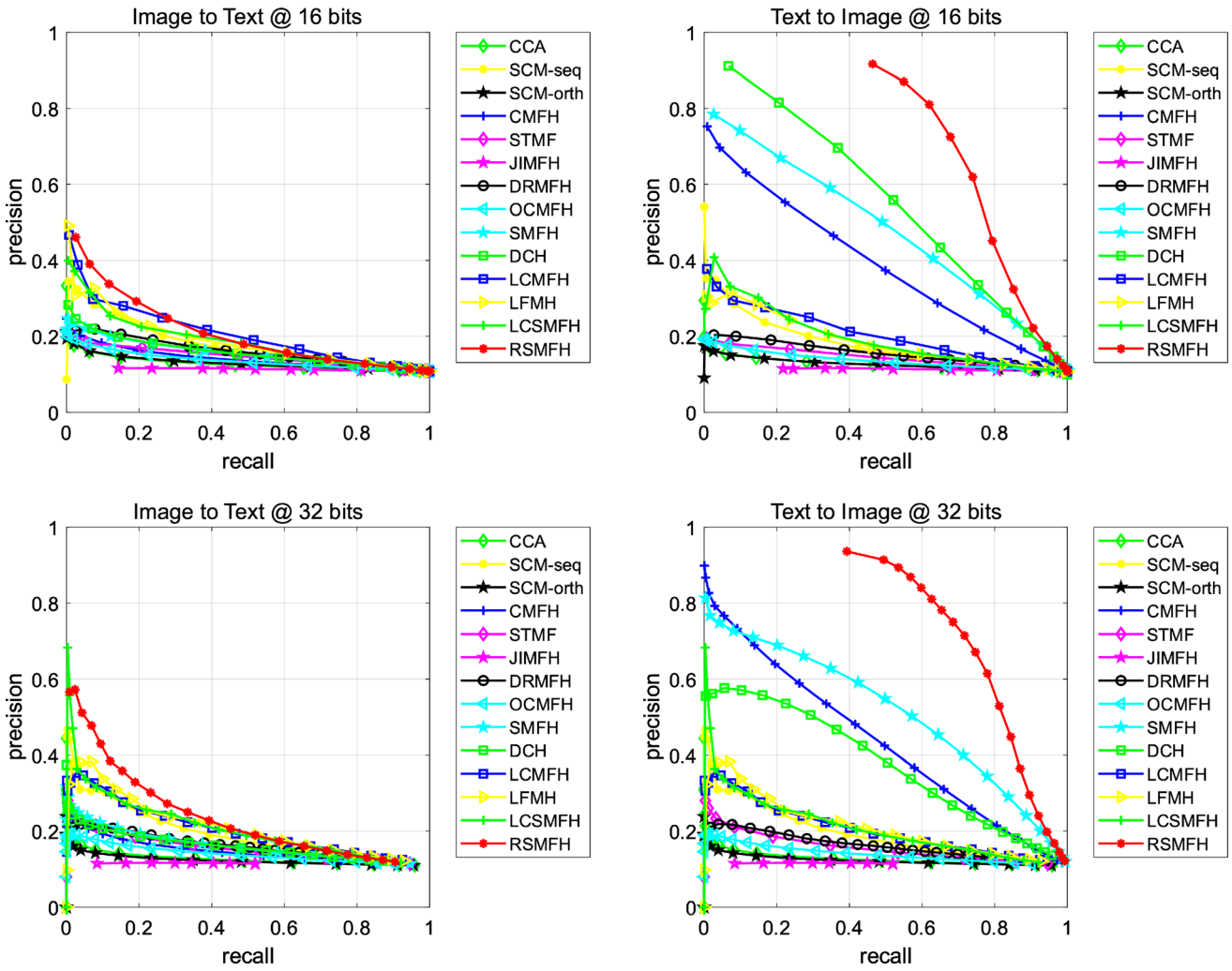

Fig. 5 PR curves varied code length on Wiki

high-dimensional feature matrix of image modality retains less semantic information than text modality in the hash codes. From the semantic point of view, textual information can effectively characterize semantic information than visual features.

4) It is worth noting that our RSMFH approach outperforms other approaches on four datasets. This is because our proposed model can fully explore the potential semantic information by utilizing the label information and the specific attributes of each modality. Thus, the hash codes learned from the proposed RSMFH method are embedded with more semantic information.

5.5 Convergence analysis

Since the proposed model is optimized by using the iterative updating strategy, the convergence rate of the algorithm is very essential on the retrieval performance. Figure 6 indicates the convergence curves of RSMFH on four datasets, where we set the hash code length to 32 bits. In Fig. 6, the x-axis is the iteration times, and the y-axis is the values of the modal. From Fig. 6, it can be seen that the proposed RSMFH method converges within 10 iterations on all datasets, and this phenomenon demonstrates the efficacy of the optimization scheme in practice.

5.6 Parameter sensitivity analysis

In this subsection, we set the length of the hash code to 32 bits, and analyze the parameters sensitivity in the proposed RSMFH model. Specifically, the values of one parameter are varied while others are fixed. λ denotes the penalty parameter controlling the weights of two modalities and we set its values to 0.5. β stands for the weight parameter controlling the label embedding, whose values are set from 0.0001 to 3000. α is the weight parameter controlling the inconsistency of E_1 and E_2, and in this paper we vary it

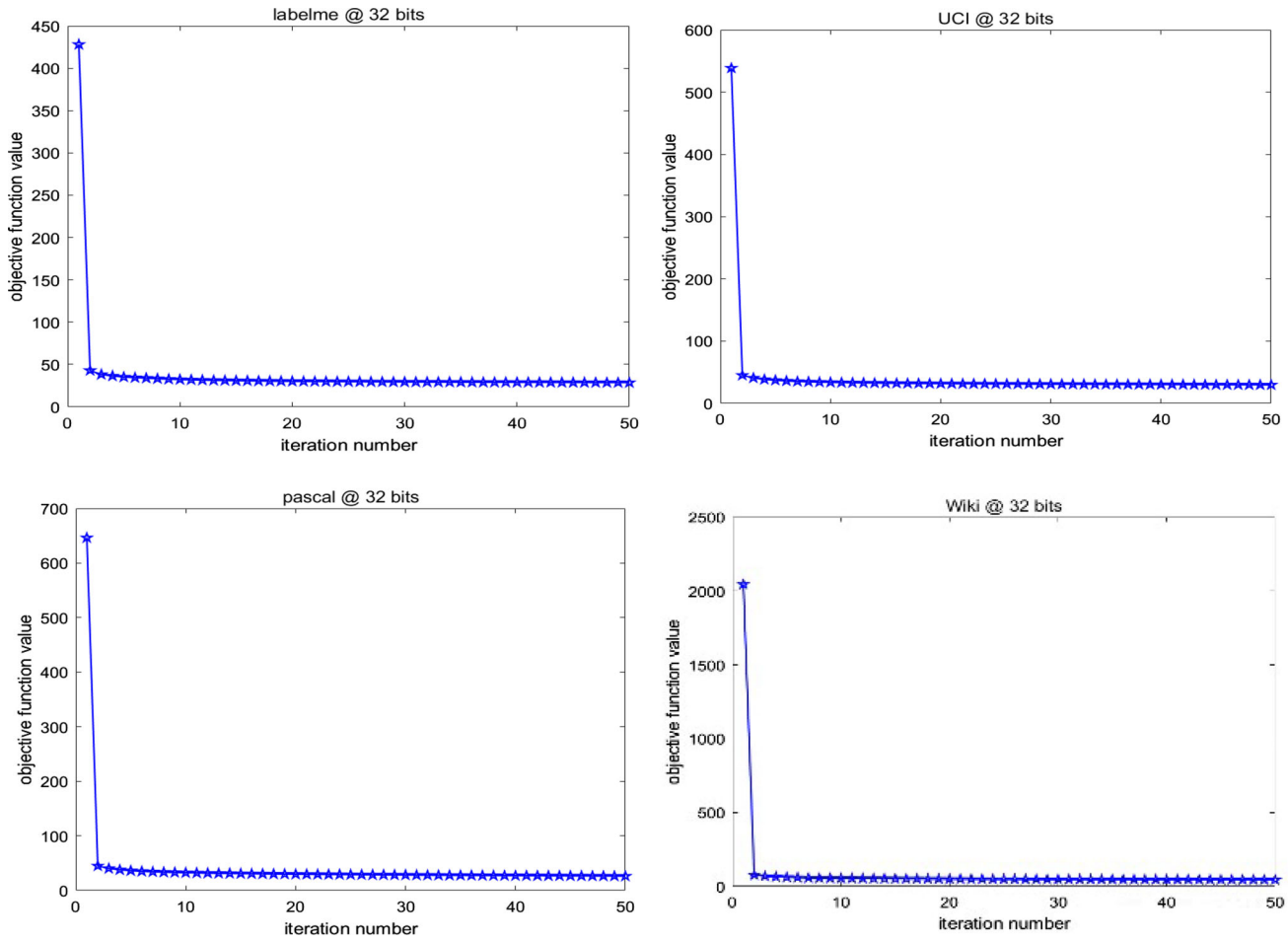

Fig. 6 Convergence curves on four datasets

from $1e-8$ to $1e-3$. μ is the weight parameter that controls the mapping from the unified representation to the hash code, whose values are changed from $1e-6$ to $1e+2$. γ denotes the weight parameter and its range is set to $[0.1\ 1000]$. Figures 7, 8, 9, and 10 show the retrieval performances of the proposed RSMFH approach with different values of four parameters β, α, μ, and γ. From these figures, we can see that the performances of our RSMFH approach can keep a relatively stable state in a large range of parameters.

5.7 Ablation study

In this subsection, we conduct ablation experiments to verify the effectiveness of several components of our proposed model. Therefore, four variants of RSMFH, i.e., RSMFH -L, RSMFH-E, RSMFH-E1 and RSMFH-E2, are constructed for comparison. Specifically, RSMFH-L discards the label information of multimodal data in comparison to the original model. RSMFH-E learns the unified representation for each modality, but ignores the inconsistent representation of each modality. RSMFH-E1 removes the inner product term of both E_1 and E_2. RSMFH-E2 is constructed by removing the sparse constraint items. We compare the performances of RSMFH with its four variants in two scenarios on the four datasets: unseen class retrieval and seen class retrieval. Table 6 reports the mAP results with different hash code lengths. From the experimental results, we can obtain the following observations:

(1) As can be seen from Table 6, the mAP values of RSMFH on four datasets is better than its variants. This is because our proposed RSMFH method not only retains shared attributes, but also considers the specific attributes. At the same time, the sparse constraints are used for the inconsistent representation, which effectively deals with the noise. In addition, we also use label information to guide hash code learning. This demonstrates the effectiveness of our proposed RSMFH method.

(2) Table 6 shows that RSMFH-L has the worst performance among all the four data, which indicates that

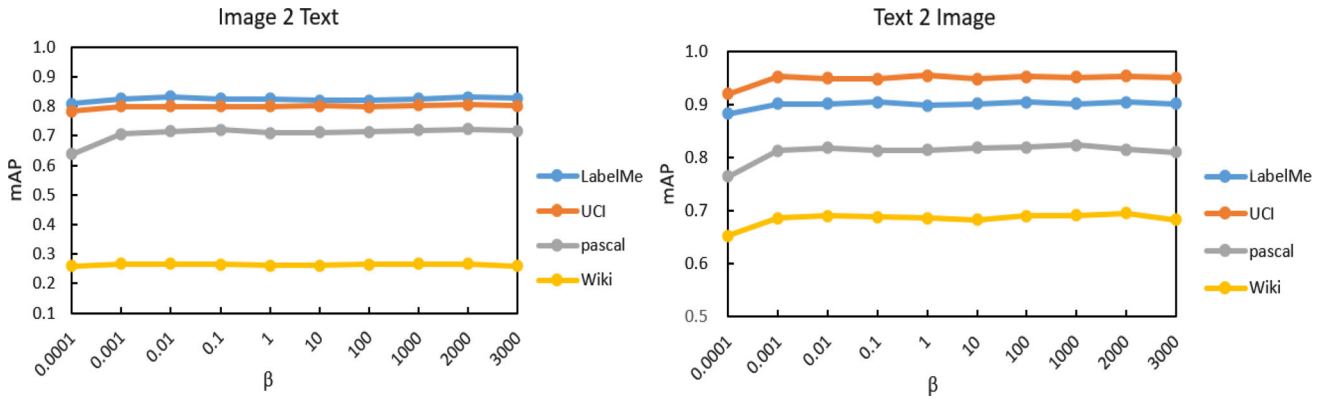

Fig. 7 mAP versus parameter β

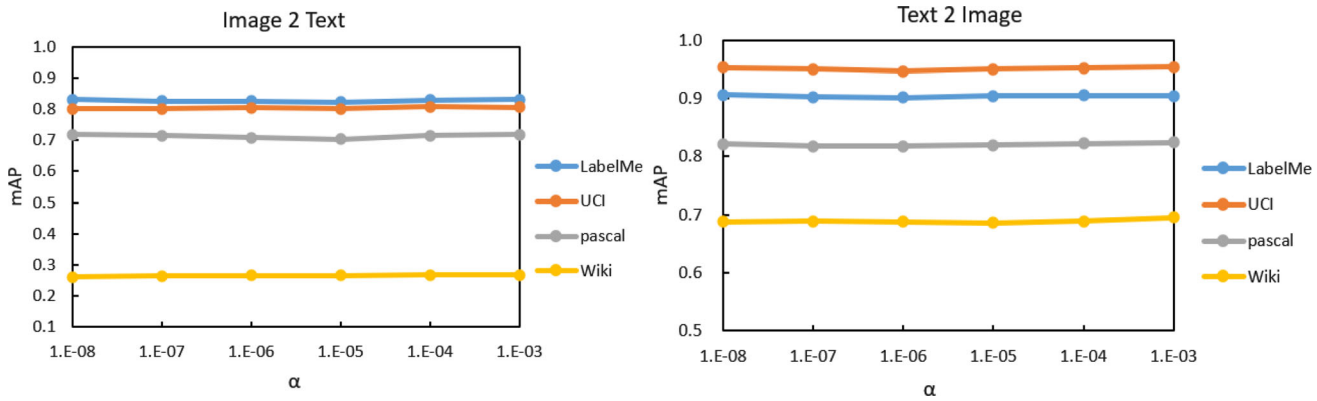

Fig. 8 mAP versus parameter α

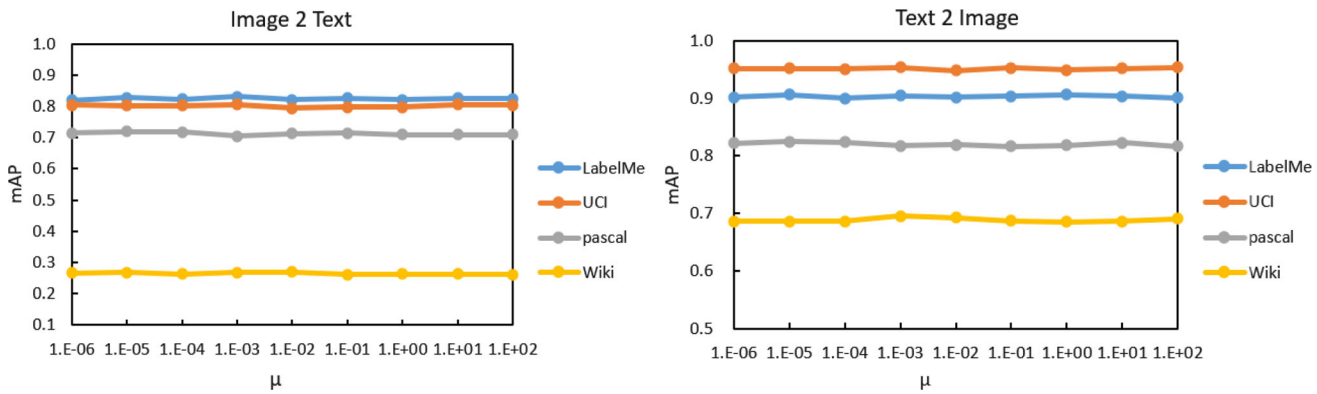

Fig. 9 mAP versus parameter μ

considering the label information in hash code learning can significantly improve the retrieval performance.

(3) RSMFH-E, RSMFH-E1 and RSMFH-E2 cannot outperform RSMFH on the four datasets. The main reason is that RSMFH-E method only learns the unified representation of different modalities. Therefore, it ignores the inconsistent representation of each modality, which affects the performance of cross-modal retrieval. In addition, RSMFH also outperform RSMFH-E1 and RSMFH-E2, which shows that the inner product constraint and the sparse constraints on the inconsistent expressions E_1 and E_2 can enlarge the gap of the inconsistent representation and effectively reduce the influence of noise and outliers, respectively.

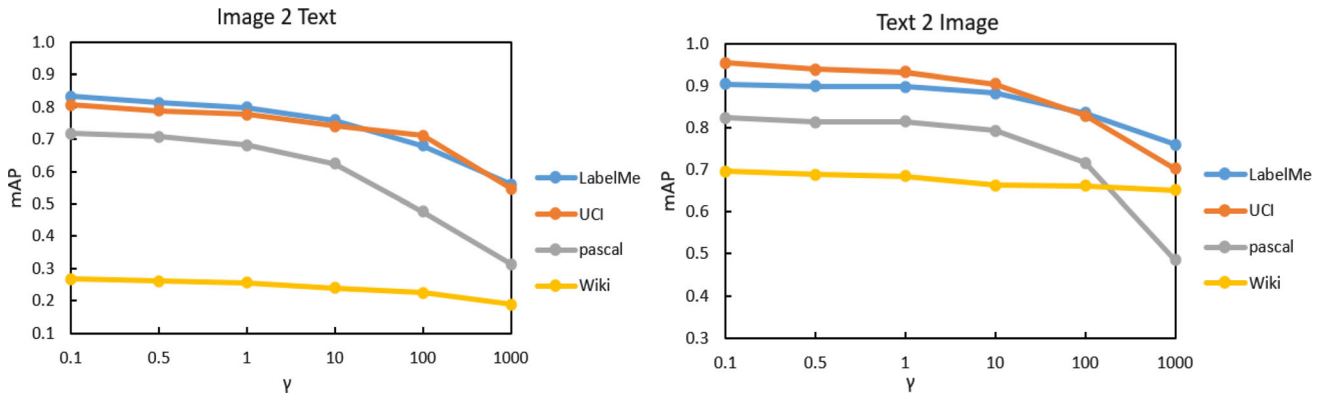

Fig. 10 mAP versus parameter γ

Table 6 Ablation study on the four datasets

Methods	Txt2Img				Img2Txt			
	16bits	32bits	64bits	128bits	16bits	32bits	64bits	128bits
Wiki								
RSMFH -L	0.2510	0.2548	0.2552	0.2565	0.6024	0.6225	0.6249	0.6313
RSMFH -E	0.2487	0.2588	0.2687	0.2781	0.6708	0.6842	0.6895	0.6939
RSMFH -E1	0.2496	0.2606	0.2703	0.2793	0.6738	0.6865	0.6911	0.6959
RSMFH -E2	0.2515	0.2634	0.2675	0.2743	0.6725	0.6888	0.6912	0.6949
RSMFH	**0.2598**	**0.2674**	**0.2742**	**0.2843**	**0.6745**	**0.6954**	**0.6989**	**0.7012**
LabelMe								
RSMFH -L	0.6763	0.7288	0.7722	0.7886	0.7370	0.8044	0.8450	0.8560
RSMFH -E	0.7987	0.8234	0.8333	0.8400	0.8905	0.9025	0.9024	0.9037
RSMFH -E1	0.8046	0.8266	0.8324	0.8381	0.8964	0.9010	0.9036	0.9095
RSMFH -E2	0.7915	0.8244	0.8339	0.8369	0.8874	0.9006	0.9010	0.9070
RSMFH	**0.8160**	**0.8318**	**0.8424**	**0.8467**	**0.9018**	**0.9046**	**0.9065**	**0.9112**
UCI								
RSMFH -L	0.6821	0.7461	0.7703	0.7846	0.7931	0.8821	0.9050	0.9182
RSMFH -E	0.7860	07,984	0.8034	0.8070	0.9354	0.9461	0.9511	0.9489
RSMFH -E1	0.7820	0.8016	0.8058	0.8096	0.9400	0.9472	0.9526	0.9516
RSMFH -E2	0.7856	0.8010	0.8036	0.8060	0.9384	0.9478	0.9502	0.9530
RSMFH	**0.7907**	**0.8058**	**0.8087**	**0.8102**	**0.9412**	**0.9487**	**0.9536**	**0.9556**
Pascal								
RSMFH -L	0.5557	0.5962	0.6014	0.6102	0.6385	0.6879	0.7018	0.7169
RSMFH -E	0.6688	0.7060	0.7310	0.7385	0.7986	0.8105	0.8170	0.8283
RSMFH -E1	0.6550	0.7054	0.7143	0.7370	0.8007	0.8112	0.8249	0.8230
RSMFH -E2	0.6680	0.7081	0.7304	0.7392	0.7974	0.8099	0.8225	0.8279
RSMFH	**0.6724**	**0.7182**	**0.7353**	**0.7406**	**0.8042**	**0.8184**	**0.8266**	**0.8370**

5.8 Visualization analysis

To better verify the effectiveness of the proposed RSMFH method, we employ the t-SNE tool to visualize the distribution of the original features and the learned representations. Specifically, we randomly select 600 image-text pairs from the LabelMe dataset for visualization experiments. The visualization results are shown in Fig. 11, where different colors represent different categories and different shapes represent different modalities, respectively. Figure 11a, c and e show the visual distribution of the original image features, the original text features and the mixed features of the two modalities, respectively. The results show that the original features of images and texts are scattered, and it is difficult to separate the categories. In addition, we can see from Fig. 11e that the scatterplots from the same category cannot correspond. This indicates that the distributions of the two modalities are also very

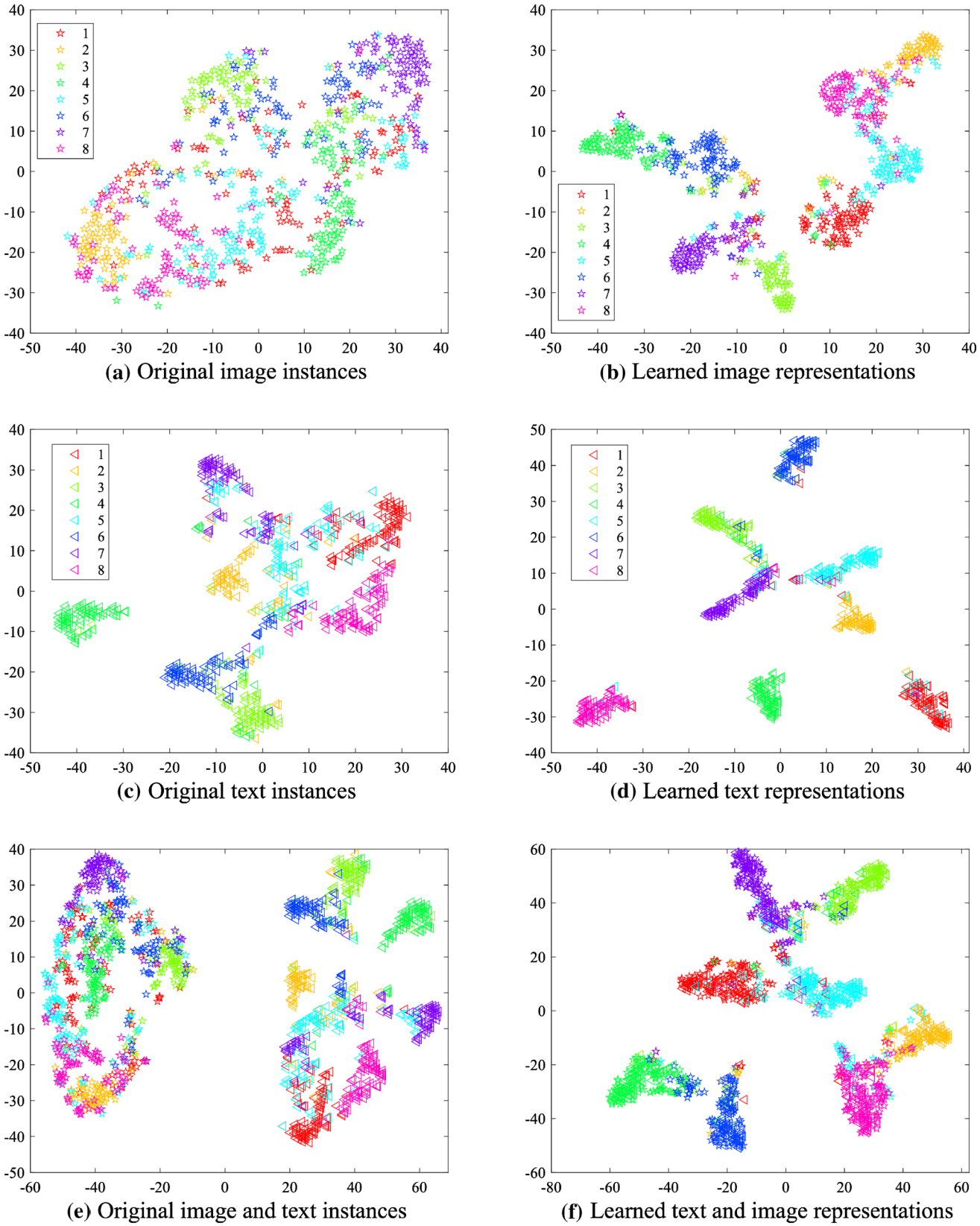

Fig. 11 t-SNE visualization of the raw features and the learned semantic features

different. It can be seen from Figs. 11 b and d that the representations of the image and text modality, respectively. This shows that the RSMFH method can effectively learn the discriminative semantic representation. In Fig. 11f, the learned image and text representations are mixed together. Therefore, we can know that the representations of the multimodality data obtained by our RSMFH method have stronger discriminative ability than those of the original multimodality data. In addition, the image and text samples from the same category are close, which demonstrates that our proposed model can effectively narrow the gap between different modalities.

6 Conclusions

In this paper, we propose a novel cross-modal retrieval approach, called RSMFH, which maintains both the shared and the specific properties by decomposing each modality into a shared semantic representation and an inconsistent representation. Meanwhile, the inconsistent representation of multi-modality data is imposed by the sparse constraints and their inner product is minimized simultaneously. Thus, it effectively improves the robustness of our approach. In addition, the hash codes are directly learned from the shared latent semantic representations and embedded in the supervised label information simultaneously. Therefore, our RSMFH approach can learn more discriminative hash codes. Experimental results on four benchmark datasets have shown the effectiveness of our RSMFH approach.

Author contributions All authors contributed to the study conception and design. Material preparation, data collection and experimental analysis were performed by ZS, KY and DZ. The first draft of the manuscript was written by KY and ZS. JY, ZY and XJW commented on previous versions of the manuscript. All authors read and approved the final manuscript.

Funding This work was supported by the National Natural Science Foundation of China [Grant Nos. 61603159, 62162033, U21B2027, U1836218], Yunnan Provincial Major Science and Technology Special Plan Projects [Grant Nos. 202002AD080001, 202103AA080015], Yunnan Foundation Research Projects [Grant Nos. 202101AT070438, 202101BE070001-056], Excellent Key Teachers of QingLan Project in Jiangsu Province.

Data availability The datasets analyzed during the current study are available in the LabelMe, UCL, Pascal sentences, Wiki repository http://labelme.csail.mit.edu/Release3.0/, https:// www.ucl.ac.uk/library, https://github.com/rupy/PascalSentenceDataset, http://www.svcl.ucsd.edu/projects/crossmodal/.

Declarations

Conflict of interest All authors declare that they have no conflicts of interest to this work.

Ethical approval Our study did not involve animals.

Informed consent Our study did not involve human participants.

References

1. Yang E, Deng C, Liu W et al (2017) Pairwise relationship guided deep hashing for cross-modal retrieval. In: Proceedings of the AAAI conference on artificial intelligence, pp 1618–1625
2. Yang E, Deng C, Li C et al (2018) Shared predictive cross-modal deep quantization. In: IEEE transactions on neural networks and learning systems, pp 1–12
3. Shu Z, Li L, Yu J et al (2022) Online supervised collective matrix factorization hashing for cross-modal retrieval. In: Applied intelligence, pp 1–18
4. Shu Z, Yong K, Yu J et al (2022) Discrete asymmetric zero-shot hashing with application to cross-modal retrieval. In: Neurocomputing, pp 366–379
5. Zhang D, Wu X, Yin H et al (2022) MOON: multi-hash codes joint learning for cross-media retrieval. In: Pattern recognition letters, pp 19–25
6. Shu Z, Bai Y, Zhang D et al (2022) Specific class center guided deep hashing for cross-modal retrieval. In: Information sciences, pp 304–318
7. Deng C, Yang E, Liu T et al (2019) Unsupervised semantic-preserving adversarial hashing for image search. In: IEEE transactions on image processing, pp 4032–4044
8. Deng C, Yang E, Liu T et al (2019) Two-stream deep hashing with class-specific centers for supervised image search. In: IEEE transactions on neural networks and learning systems, pp 1–13
9. Yu J, Zhang D, Shu Z et al (2022) Adaptive multi-modal fusion hashing via Hadamard matrix. In: Applied intelligence, pp 1–15
10. Gionis A, Indyk P, Motwani R et al (1999) Similarity search in high dimensions via hashing. In: Proceedings of the 25th VLDB conference, pp 518–529
11. Weiss Y, Torralba A, Fergus R (2009) Spectral hashing. In: Advances in neural information processing systems, pp 1753–1760
12. Zhu X, Huang Z, Cheng H et al (2013) Sparse hashing for fast multimedia search. In: ACM transactions on information systems, pp 1–24
13. Zhu X, Huang Z, Shen HT et al (2013) Linear cross-modal hashing for efficient multimedia search. In: ACM international conference on multimedia, pp 143–152
14. Song J, Yang Y, Yang Y et al (2013) Inter-media hashing for large-scale retrieval from heterogeneous data sources. In: ACM international conference on multimedia of data, pp 785–796
15. Zheng F, Tang Y, Shao L (2018) Hetero-manifold regularization for cross-modal hashing. In: IEEE transactions on pattern analysis and machine intelligence, pp 1059–1071
16. Wu F, Yu Z, Yang Y et al (2014) Sparse multi-modal hashing. In: IEEE transactions on multimedia, pp 427–439
17. Ding G, Guo Y, Zhou J (2014) Collective matrix factorization hashing for multimodal data. In: IEEE conference on computer vision and pattern recognition, pp 2083–2090

18. Wang D, Gao X, Wang X et al (2015) Semantic topic multimodal hashing for cross-media retrieval. In: International joint conference on artificial intelligence, pp 3890–3896
19. Wang D, Wang Q, He L et al (2020) Joint and individual matrix factorization hashing for large- scale cross-modal retrieval. In: Pattern recognition, pp 1–12
20. Zhou J, Ding G, Guo Y (2014) Latent semantic sparse hashing for cross-modal similarity search. In: ACM SIGIR conference on research and development in information retrieval, pp 415–424
21. Wang D, Wang Q, Gao X (2018) Robust and flexible discrete hashing for cross–modal similarity search. In: IEEE transactions on circuits and systems for video technology, pp 2703–2715
22. Yao T, Li Y, Guan W et al (2021) Discrete robust matrix factorization hashing for large-scale cross-media retrieval. In: IEEE transactions on knowledge and data engineering, pp 1–12
23. Bronstein MM, Bronstein AM, Michel F et al (2010) Data fusion through cross-modality metric learning using similarity-sensitive hashing. In: IEEE conference on computer vision and pattern recognition, pp 3594–3601
24. Kumar S, Udupa R (2011) Learning hash functions for cross-view similarity search. In: International joint conference on artificial intelligence, pp 1360–1367
25. Zhang D, Li W-J (2014) Large-scale supervised multimodal hashing with semantic correlation maximization. In: Proceedings of the AAAI conference on artificial intelligence, pp 2177–2183
26. Lin Z, Ding G, Hu M et al (2015) Semantics-preserving hashing for cross-view retrieval. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3864–3872
27. Tang J, Wang K, Shao L (2016) Supervised matrix factorization hashing for cross–modal retrieval. In: IEEE transactions on image processing, pp 3157–3166
28. Mandal D, Chaudhury KN, Biswas S (2017) Generalized semantic preserving hashing for n-label cross-modal retrieval. In: IEEE conference on computer vision and pattern recognition, pp 4076–4084
29. Wang D, Gao X, Wang X et al (2018) Label consistent matrix factorization hashing for large-scale cross-modal similarity search. In: IEEE transactions on pattern analysis and machine intelligence, pp 2466–2479
30. Xue F, Wang W, Zhou W et al (2020) Cross-modal retrieval via label category supervised matrix factorization hashing. In: Pattern recognition letters, pp 469–475
31. Zhang D, Wu X, Yu J (2021) Label consistent flexible matrix factorization hashing for efficient cross-modal retrieval. In: ACM transactions on multimedia computing communications and applications, pp 1–18
32. Obozinski G, Taskar B, Jordan MI (2010) Joint covariate selection and joint subspace selection for multiple classification problems. In: Statistics and computing, pp 231–252
33. Kong D, Huang H, Huang H (2011) Robust nonnegative matrix factorization using $L_{2,1}$-norm. In: ACM international conference on information and knowledge management, pp 673–682
34. Lai Z, Chen Y, Wu J et al (2018) Jointly sparse hashing for image retrieval. In: IEEE transactions on image processing, pp 6147–6158
35. Li C-X, Chen Z-D, Zhang P-F et al (2018) SCRATCH: a scalable discrete matrix factorization hashing for cross-modal retrieval. In: ACM international conference on multimedia, pp 1–9
36. Shen F, Shen C, Liu W et al (2015) Supervised discrete hashing. In: IEEE conference on computer vision and pattern recognition, pp 37–45
37. Russell BC, Torralba A, Murphy KP et al (2008) LabelMe: a database and web-based tool for image annotation. In: International Journal of Computer Vision, pp 157–173
38. Seewald AK (2005) Digits–a dataset for handwritten digit recognition. In: Austrian research institut for artificial intelligence technical report, Vienna (Austria)
39. Rashtchian C, Young P, Hodosh M et al (2010) Collecting image annotations using amazon's mechanical turk. In: Proceedings of the NAACL HLT 2010 workshop on creating speech and language data with Amazon's Mechanical Turk, pp 139–147
40. Rasiwasia N, Pereira J, Coviello E et al (2010) A new approach to cross-modal multimedia retrieval. In: ACM international conference on multimedia, pp 251–260
41. Rupnik J, Shawe-Taylor J (2010) Multi-view canonical correlation analysis. In: Proceedings of the conference on data mining and data warehouses, pp 1–4
42. Wang D, Wang Q, An Y et al (2020) Online collective matrix factorization hashing for large-scale cross-media retrieval. In: ACM SIGIR conference on research and development in information retrieval, pp 1409–1418
43. Xu X, Shen F et al (2017) Learning discriminative binary codes for large-scale cross-modal retrieval. In: IEEE transactions on image processing, pp 2494–2507
44. Baeza-Yates R, Ribeiro-Neto B (1999) Modern information retrieval. Addison Wesley