Binary Set Embedding for Cross-Modal Retrieval

Mengyang Yu, *Student Member, IEEE*, Li Liu, and Ling Shao, *Senior Member, IEEE*

***Abstract*— Cross-modal retrieval is such a challenging topic that traditional global representations would fail to bridge the semantic gap between images and texts to a satisfactory level. Using local features from images and words from documents directly can be more robust for the scenario with large intraclass variations and small interclass discrepancies. In this paper, we propose a novel unsupervised binary coding algorithm called binary set embedding (BSE) to obtain meaningful hash codes for local features from the image domain and words from text domain. Understanding image features with the word vectors learned from the human language instead of the provided documents from data sets, BSE can map samples into a common Hamming space effectively and efficiently where each sample is represented by the sets of local feature descriptors from image and text domains. In particular, BSE explores relationship among local features in both feature level and image (text) level, which can balance the sensitivity of each other. Furthermore, a recursive orthogonalization procedure is applied to reduce the redundancy of codes. Extensive experiments demonstrate the superior performance of BSE compared with state-of-the-art cross-modal hashing methods using either image or text queries.**

***Index Terms*— Cross-modal retrieval, hashing, local descriptor, multimedia, word vector.**

I. Introduction

IN THE current multimedia era, an image always appears with a description of text content on public knowledge websites such as the Wikipedia or photo sharing/social media websites such as Flickr and Facebook. Due to the diversity of the query and the multiple modalities of input, the multimedia similarity search is becoming a critical problem and a ubiquitous searching method on the Internet [1]–[5]. Nonetheless, the traditional nearest neighbor search (NN-search) in information retrieval is neither scalable nor efficient when facing the explosion of multimedia data. To conquer this problem, binary code representations [6]–[9], or hashing methods [10]–[18], provide a fast search mechanism through the bit XOR operation and the time complexity of similarity search is simply $O(1)$ if all the binary codes are stored. In addition, a more discriminative representation could be acquired if the algorithm sufficiently learns the intrinsic structure and the semantic information of multimedia data. In order to generate effective but compact hash codes, many learning systems have been involved in hashing methods. Most of them obtain the hash function through mining the data structure and solving an optimization problem associated with the objective function. For instance, Spectral hashing (SpH) [18] learns compact binary codes by employing the balanced and uncorrelated constraints into the learning phase. Evolutionary compact embedding [19] combines genetic programming with the boosting scheme to generate high-quality binary codes for large-scale data classification tasks. Furthermore, Semantic hashing [20] was proposed to learn hash codes based on the deep neural network.

Since locality sensitive hashing [10] introduced the idea of preserving the similarity in the original data, a number of cross-modal hashing schemes have been proposed to discover the relationship among different modalities of multimedia data. Cross-modality similarity search hashing (CMSSH) [21] embeds incommensurable data into a common metric space by a boosting algorithm. With extended SpH [18], Kumar and Udupa [11] proposed cross-view hashing (CVH) to generate binary codes for each modality via canonical correlation analysis (CCA). Multimodal latent binary embedding (MLBE) [22] is another cross-modal hashing method considering both the intermodal and intramodal similarity via a probabilistic model. To learn the hash function with good generalization, co-regularized hashing [13] was proposed to project data far from zero. Zhu *et al.* [23] proposed a linear method for multimedia search to reduce the computational complexity. Recently, intermedia hashing (IMH) [15] was proposed to explore the correlations among different modalities and learn hashing functions by a linear regression model. Instead of learning codes for each specific view, both composite hashing with multiple information sources (CHMISs) [14] and collective matrix factorization hashing (CMFH) [16] learn unified hash codes for each sample.

Notwithstanding the successful results achieved by the methods aforementioned, the lack of incorporating the visual features with the corresponding linguistic understanding makes them uncompetitive for the challenging cross-modal tasks. A major drawback is the use of global histogram-based representations, which would bring the quantization error during the codebook construction and lose the structure of local features and words. The document-oriented representations, such as latent Dirichlet allocation (LDA) [24], need to be retrained when the text is modified, a new paragraph is added to the data set or a new data set is built. This operation largely increases the computational complexity and the aforementioned cross-modality algorithms are also required to be implemented again. In addition, single-vector representations cannot comprehensively and precisely characterize the samples, which have

Manuscript received May 10, 2016; revised August 12, 2016; accepted September 12, 2016. This work was supported in part by Northumbria University and in part by National Natural Science Foundation of China under Grant 61528106. *(Corresponding author: Ling Shao).*

M. Yu and L. Shao are with the School of Computer and Information Science, Southwest University, Chongqing 400715, China, and also with the Department of Computer and Information Sciences, Northumbria University, Newcastle upon Tyne, NE1 8ST, UK (e-mail: m.y.yu@ieee.org; ling.shao@ieee.org).

L. Liu is with the Department of Computer and Information Sciences, Northumbria University, Newcastle upon Tyne, NE1 8ST, UK (e-mail: li2.liu@northumbria.ac.uk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNNLS.2016.2609463

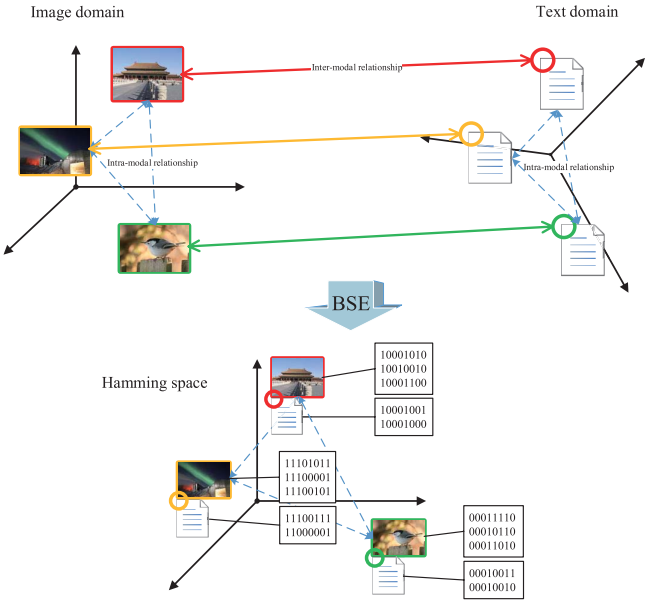


Fig. 1. Illustration of BSE. BSE encodes all local features in the image and text domains into a common Hamming space.

multiple tags or topics and the scenario with large intraclass variations and small interclass discrepancies.

In this paper, we aim to exploit the semantic connection between images and their corresponding documents in low-level features, either visual or textual, i.e., local features. The local feature descriptors for images, such as SIFT [25] and even deep features [26], have been well studied. The construction of local features for texts can be done by the *word vector* techniques [27]–[29] in natural language processing, which have been shown the superiority in machine translation. Once the learning phase for local features is completed, the coding function is fixed for any new sample (image-text pair) since each word has been assigned a unique hash code. Apparently, one of the requirements for our algorithm is that the cross-modal links based on local features need to be established. However, it is impossible and unrealistic to build a one-to-one correspondence between local feature points from different modalities. Therefore, we consider the relationship between the sets composed of local features from image and text domains. Taking the image-text pair of a car as an example, two SIFT features are close to each other if they are visually similar and two word vectors are close to each other if they are semantically similar. Meanwhile, the cross-modal algorithm must also connect the local feature set of the image "car" and the word vector set of the corresponding description of the car for semantic understanding of images.

To address the above problem, we propose a novel cross-modal hashing scheme called binary set embedding (BSE), which is shown in Fig. 1. Due to the different distributions of image and text data, BSE learns two orthogonal projections and projects local features (image or text) into a common low-dimensional Hamming space. In this way, for each sample, the image features and the corresponding linguistic features are encoded to similar hash codes by BSE. In the meantime, we also take the geometric structures of each modality into account for preserving the intramodal similarity. Given a local feature, its source information, i.e., the image (text) from which it is extracted, is also provided. Consequently, relationships in two layers: element-to-element and set-to-set which are equivalent to the structures of data points and images (texts) represented by local feature sets, respectively, are simultaneously preserved in the lower-dimensional Hamming space. It is worthwhile to highlight the several properties of the proposed approach.

1) This paper associates images with semantic information in a fundamental level. The binary codes learned from local image features are semantically more robust than the word-frequency histogram.
2) BSE assigns a binary code for each local feature. The encoding of local features reduces the sparsity of the final hash table and improves the usage of hash codes, which enables hash codes to achieve competitive performance with a short length.
3) Last but not least, the local features for the text domain, i.e., word vectors, are independent of any specific data sets and can be trained offline, which makes BSE more universal in realistic applications.

The rest of this paper is organized as follows. The BSE algorithm is proposed in Section II with an orthogonal constraint in Section II-E. In Section III, we introduce a voting scheme for indexing, as the traditional indexing method is not suitable for local features. Experimental results are shown in Section IV. Finally, we conclude this paper in Section V.

II. Binary Set Embedding

In this section, we introduce our BSE algorithm. We first describe the intramodal and intermodal structures and then associate them into one objective function. With the orthogonality constraint, BSE outputs the orthogonal projections for each modality.

A. Notations and Problem Statement

Since our task is the similarity search between the image domain and the text domain, we consider s image-text sample pairs $S_1, \ldots, S_s$ containing the local feature sets from the image and text domains. For the ith sample pair S_i, we denote its local feature sets in the image and text domains by $X_i = \{\mathbf{x}_{i1}, \ldots, \mathbf{x}_{in_i}\}$ with $\mathbf{x}_{ij} \in \mathbb{R}^{D_1}$, and $Y_i = \{\mathbf{y}_{i1}, \ldots, \mathbf{y}_{im_i}\}$ with $\mathbf{y}_{ij} \in \mathbb{R}^{D_2}$, respectively. In this way, we have the union of the local feature sets $X = \bigcup_{i=1}^{s} X_i$ and $Y = \bigcup_{i=1}^{s} Y_i$. Without the loss of generality, we denote $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ and $Y = \{\mathbf{y}_1, \ldots, \mathbf{y}_M\}$, where $N = \sum_{i=1}^{s} n_i$ and $M = \sum_{i=1}^{s} m_i$.

Considering the different properties of image and text domains, we aim to seek two projections $\Theta_1 \in \mathbb{R}^{D_1 \times d}$ and $\Theta_2 \in \mathbb{R}^{D_2 \times d}$ for X and Y, respectively, to build the hash functions with the same code length

$$H_1(\mathbf{x}) = \operatorname{sgn}\left(\Theta_1^T \mathbf{x}\right) \quad \text{and} \quad H_2(\mathbf{y}) = \operatorname{sgn}\left(\Theta_2^T \mathbf{y}\right). \tag{1}$$

It is noticeable that during the code learning stage, we use $\{-1, +1\}$ to encode local features and employ centralized data $\mathbf{x}_i - (1/N)\sum_{j=1}^{N} \mathbf{x}_j$ and $\mathbf{y}_i - (1/M)\sum_{j=1}^{M} \mathbf{y}_j$ instead of $\mathbf{x}_i$ and $\mathbf{y}_i$, respectively, $i = 1, \ldots, s$. In the indexing phase, we use $\{0, 1\}$ to represent codes for hash lookup.

B. Intramodal Relationship

For the unsupervised analysis based on local feature descriptors, we are given not only the local features themselves, but also their source information, i.e., the sample from which they are extracted. We first discuss the connection between local features and the connection between images for the image domain. Then, we have the similar objective functions for the text domain.

1) Element-to-Element Structure: We hope that the pairwise structure of local features in the original space could be preserved in the lower-dimensional Hamming space. Without class information, we employ the K-means clustering on X to divide the set $\mathcal{P}_1 = \{(i, j)|\mathbf{x}_i, \mathbf{x}_j \in X\}$ into two categories, i.e., positive pairs and negative pairs. Specifically, we divide X into K clusters by the K-means clustering and define the pairwise label for $(\mathbf{x}_i, \mathbf{x}_j)$ as follows:

$$\ell_{ij}^X = \begin{cases} +1, & \text{if } \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ are in the same cluster} \\ -1, & \text{otherwise.} \end{cases}$$

Moreover, we also expect that, for a positive pair, the effect on the objective function will increase when their distance decreases, and for a negative pair, conversely, its importance will be reduced when the paired features are closer to each other. Then, by Gaussian function, we assign the following weight for each pair with parameter σ:

$$W_{ij}^X = \begin{cases} \exp\left(-\dfrac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\sigma^2}\right), & \ell_{ij}^X = 1 \\ \exp\left(-\dfrac{1}{\sigma^2\|\mathbf{x}_i - \mathbf{x}_j\|^2}\right), & \ell_{ij}^X = -1 \end{cases}$$

where $\|\cdot\|$ is the L^2-norm. It is easy to find that $W_{ij}^X \in (0, 1)$ and satisfies our requirement. Hence, preserving the feature-to-feature structure in the image domain is to maximize

$$\sum_{(i,j)\in\mathcal{P}_1} W_{ij}^X \ell_{ij}^X \langle H_1(\mathbf{x}_i), H_1(\mathbf{x}_j)\rangle. \tag{2}$$

Similarly, for the text domain, we also have the following objective function to be maximized:

$$\sum_{(i,j)\in\mathcal{P}_2} W_{ij}^Y \ell_{ij}^Y \langle H_2(\mathbf{y}_i), H_2(\mathbf{y}_j)\rangle \tag{3}$$

where $\mathcal{P}_2$ is the pair set for the text domain, and W_{ij}^Y and ℓ_{ij}^Y are the pairwise weights and the pairwise labels in the text domain, respectively.

2) Set-to-Set Structure: The set-to-set structure can be regarded as a higher-level connection among local features to balance the sensitivity of the clustering information in the above element-to-element structure. This structure is constructed on the samples when each of them is represented by a set of local features. For image i, X_i represents its local feature set $\{\mathbf{x}_{i1}, \ldots, \mathbf{x}_{in_i}\}$. We use the image-to-image (I2I) distance derived from [30] to measure the set-to-set distance from image i to image j, which can be regarded as an approximation of the Kullback–Leibler divergence and is defined as

$$d_{ij} = \sum_{\mathbf{x}\in X_i} \|\mathbf{x} - \mathrm{NN}_j(\mathbf{x})\|^2 \tag{4}$$

where $\mathrm{NN}_j(\mathbf{x})$ is the nearest neighbor of local feature $\mathbf{x}$ in image j. Since generally $d_{ij} \neq d_{ji}$, we update the symmetric distance $D_{ij} = (d_{ij} + d_{ji})/2$ as the I2I distance between image i and image j. By Gaussian function, we can define the following I2I similarity with the smooth parameter σ_X:

$$S_{ij}^X = \exp\left(-\frac{D_{ij}^2}{2\sigma_I^2}\right), \quad i, j = 1, \ldots, s. \tag{5}$$

Although the number of local features in one image is much smaller than N, the NN-search for all images is still time-consuming. We hope to use the cluster information in the above element-to-element section for the reduction of complexity. We denote the clusters of the K-means on X by $C_1, \ldots, C_K$. Without the loss of generality, supposing the local features of image j are in $C_1, \ldots, C_{K_1}$ and the order of distances from corresponding centroids to $\mathbf{x} \in X_i$ is from the nearest to the farthest, the range of NN-search in $\mathcal{X}_j$ is reduced to $(C_1 \cup \cdots \cup C_{\lceil (K_1)^\delta \rceil}) \cap \mathcal{X}_j$, where $0 < \delta < 1$ and $\lceil\cdot\rceil$ is the ceiling function. This reduction of range is based on the assumption that the centroid of the cluster where the true nearest neighbor locates is also close to $\mathbf{x}$. In fact, it holds when $K \to N$. After the reduction of the searching range, the average complexity is reduced from $O(N^2)$ to $O(NK^{1+\delta})$ and we only need to compute the distances from $\mathbf{x}$ to the cluster centroids, which has been done in the K-means.

After applying the encoding algorithm, the I2I distance in Hamming space becomes

$$\hat{D}_{ij}^X = \frac{1}{2}\left(\sum_{\mathbf{x}\in X_i} \|H_1(\mathbf{x}) - \mathrm{NN}_j(H_1(\mathbf{x}))\|^2 + \sum_{\mathbf{x}\in X_j} \|H_1(\mathbf{x}) - \mathrm{NN}_i(H_1(\mathbf{x}))\|^2\right). \tag{6}$$

Therefore, to preserve the I2I structure of the original image domain by giving the penalty S_{ij}^X to the mapped distance $\hat{D}_{ij}^X$, a reasonable objective function is to minimize

$$\sum_{i,j} \hat{D}_{ij}^X \cdot S_{ij}^X. \tag{7}$$

Likewise, preserving the set-to-set structure in the text domain is to minimize the following similar objective function:

$$\sum_{i,j} \hat{D}_{ij}^Y \cdot S_{ij}^Y \tag{8}$$

where $\hat{D}_{ij}^Y$ and S_{ij}^Y are the encoded set-to-set distance and the set-to-set similarity in the text domain, respectively.

C. Intermodal Relationship

Local features in the image domain and the text domain have different distributions. For precise retrieval, we need to encode the local features from similar samples to close hash codes no matter they are in the image domain or the text domain. Without class information, we are only concerned about the relationship between the image local features and the text local features from the same sample.

For each sample pair S_i, the local feature set from the image domain and the local feature set from the text domain are denoted by X_i and Y_i, respectively, $i = 1, \ldots, s$. Generally speaking, it is impossible to construct a one-to-one correspondence between X_i and Y_i; even a nearest neighbor relationship in the Hamming space cannot be built since the correspondence between visual features and semantic information is unknown by the algorithm. Then, using the I2I distance to measure the connection between X_i and Y_i is not applicable. Therefore, we minimize the distance of all the local feature pairs in the set $\{(\mathbf{x}, \mathbf{y})|\mathbf{x} \in X_i, \mathbf{y} \in Y_i\}$ for the ith image-text pair in the Hamming space. In other words, our goal for the intermodal relationship is to maximize the following sum of the inner products:

$$\sum_{i=1}^{s} \sum_{\mathbf{x}\in X_i, \mathbf{y}\in Y_i} \langle H_1(\mathbf{x}), H_2(\mathbf{y})\rangle. \tag{9}$$

D. Objective Function and Optimization

1) Spectral Relaxation: First, let us transform (2), (3), (7), (8), and (9) to the functions on Θ_1 and Θ_2. Motivated by [18] and [31], we relax the discrete sign function to a real-valued continuous function by using its signed magnitude, i.e., $\text{sgn}(x) \approx x$. In this way, the objective function in the element-to-element part of the image domain, i.e., (2) becomes

$$\begin{aligned}
&\sum_{(i,j)\in\mathcal{P}_1} W_{ij}^X \ell_{ij}^X \langle \Theta_1^T \mathbf{x}_i, \Theta_1^T \mathbf{x}_j\rangle \\
&\quad= \sum_{(i,j)\in\mathcal{P}_1} W_{ij}^X \ell_{ij}^X \left(\Theta_1^T \mathbf{x}_i\right)^T \Theta_1^T \mathbf{x}_j \\
&\quad= \sum_{(i,j)\in\mathcal{P}_1} W_{ij}^X \ell_{ij}^X \operatorname{Tr}\left(\Theta_1^T \mathbf{x}_i \left(\Theta_1^T \mathbf{x}_j\right)^T\right) \\
&\quad= \sum_{(i,j)\in\mathcal{P}_1} W_{ij}^X \ell_{ij}^X \operatorname{Tr}\left(\Theta_1^T \mathbf{x}_i \mathbf{x}_j^T \Theta_1\right) \\
&\quad= \operatorname{Tr}\left(\Theta_1^T L_X \Theta_1\right)
\end{aligned} \tag{10}$$

where $L_X = \sum_{(i,j)\in\mathcal{P}_1} W_{ij}^X \ell_{ij}^X \mathbf{x}_i \mathbf{x}_j^T$. With a similar transformation, (3) for the text domain becomes

$$\operatorname{Tr}\left(\Theta_2^T L_Y \Theta_2\right) \tag{11}$$

where $L_Y = \sum_{(i,j)\in\mathcal{P}_2} W_{ij}^Y \ell_{ij}^Y \mathbf{y}_i \mathbf{y}_j^T$.

In addition, for the I2I distance, we make a statistical approximation on the computation of projected I2I distances due to the large amount of local features. That is, we exchange the operation of NN-search and $H_1(\cdot)$ for all $\mathbf{x} \in X_i$ during the optimization, i.e., $\sum_{\mathbf{x}\in X_i} \|H_1(\mathbf{x}) - \text{NN}_j(H_1(\mathbf{x}))\|^2 \approx \sum_{\mathbf{x}\in X_i} \|H_1(\mathbf{x}) - H_1(\text{NN}_j(\mathbf{x}))\|^2$. In fact, the pairwise structure has been preserved in the objective function (2), which ensures the correctness of the exchange operation. Then, we have the following projected distance $\hat{d}_{ij}$ in the optimization:

$$\begin{aligned}
\hat{d}_{ij} &\approx \sum_{\mathbf{x}\in X_i} \left\|\Theta_1^T \mathbf{x} - \Theta_1^T \text{NN}_j(\mathbf{x})\right\|^2 \\
&= \sum_{\mathbf{x}\in X_i} \left\|\Theta_1^T (\mathbf{x} - \text{NN}_j(\mathbf{x}))\right\|^2 \\
&= \sum_{\mathbf{x}\in X_i} \left(\Theta_1^T (\mathbf{x} - \text{NN}_j(\mathbf{x}))\right)^T \Theta_1^T (\mathbf{x} - \text{NN}_j(\mathbf{x})) \\
&= \sum_{\mathbf{x}\in X_i} \operatorname{Tr}\left(\Theta_1^T (\mathbf{x} - \text{NN}_j(\mathbf{x}))(\Theta_1 (\mathbf{x} - \text{NN}_j(\mathbf{x})))^T\right) \\
&= \sum_{\mathbf{x}\in X_i} \operatorname{Tr}\left(\Theta_1^T (\mathbf{x} - \text{NN}_j(\mathbf{x}))(\mathbf{x} - \text{NN}_j(\mathbf{x}))^T \Theta_1\right).
\end{aligned}$$

If we denote

$$D_X = \frac{1}{2} \sum_{i,j} S_{ij}^X \left(\sum_{\mathbf{x}\in X_i} (\mathbf{x} - \text{NN}_j(\mathbf{x}))(\mathbf{x} - \text{NN}_j(\mathbf{x}))^T + \sum_{\mathbf{x}\in X_j} (\mathbf{x} - \text{NN}_i(\mathbf{x}))(\mathbf{x} - \text{NN}_i(\mathbf{x}))^T \right)$$

then the objective function in the set-to-set part of the image domain, i.e., (7) can be written as

$$\operatorname{Tr}\left(\Theta_1^T D_X \Theta_1\right). \tag{12}$$

Certainly, for the text domain, we also have the similar trace form

$$\operatorname{Tr}\left(\Theta_2^T D_Y \Theta_2\right) \tag{13}$$

where

$$D_Y = \frac{1}{2} \sum_{i,j} S_{ij}^Y \left(\sum_{\mathbf{y}\in Y_i} (\mathbf{y} - \text{NN}_j(\mathbf{y}))(\mathbf{y} - \text{NN}_j(\mathbf{y}))^T + \sum_{\mathbf{y}\in Y_j} (\mathbf{y} - \text{NN}_i(\mathbf{y}))(\mathbf{y} - \text{NN}_i(\mathbf{y}))^T \right).$$

And for the intermodal relationship, (9) is simply relaxed to

$$\operatorname{Tr}\left(\Theta_1^T A \Theta_2\right) \tag{14}$$

where $A = \sum_{i=1}^{n} \sum_{\mathbf{x}\in X_i, \mathbf{y}\in Y_i} \mathbf{x}\mathbf{y}^T$.

2) Objective Function: Without the loss of generality, we let the objective dimension (code length) $d = 1$, i.e., Θ_1 and Θ_2 are column vectors. Furthermore, we place the intramodal relationship and the intermodal relationship at equally important positions. Therefore, combining the above functions on Θ_1 and Θ_2 and the norm constraint $\|\Theta_1\| = \|\Theta_2\| = 1$, we have our final optimization problem

$$\underset{\|\Theta_1\|=\|\Theta_2\|=1}{\arg\max} \frac{\Theta_1^T A \Theta_2}{\left(\Theta_1^T (\lambda D_X - L_X)\Theta_1\right)\left(\Theta_2^T (\lambda D_Y - L_Y)\Theta_2\right)} \tag{15}$$

where λ is the parameter for balancing the effect of the element-to-element and set-to-set structures.

3) Optimization: Let us denote $B_X = \lambda D_X - L_X$ and $B_Y = \lambda D_Y - L_Y$ which are two symmetric matrices. We change the norm constraints to $\Theta_1^T B_X \Theta_1 = 1$ and $\Theta_2^T B_Y \Theta_2 = 1$, since it is always possible to restore the final norm to $\|\Theta_1\| = \|\Theta_2\| = 1$. Then, we can define the Lagrangian function

$$\begin{aligned}
L(\Theta_1, \Theta_2) = \Theta_1^T A \Theta_2 &- \alpha\left(\Theta_1^T B_X \Theta_1 - 1\right) \\
&- \beta\left(\Theta_2^T B_Y \Theta_2 - 1\right)
\end{aligned}$$

where α and β are the Lagrangian coefficients. To find the optimal solution, we let the derivatives of L with respect to Θ_1 and Θ_2 be zeros to obtain

$$\frac{\partial L}{\partial \Theta_1} = A\Theta_2 - 2\alpha B_X \Theta_1 = 0 \tag{16}$$

$$\frac{\partial L}{\partial \Theta_2} = A^T \Theta_1 - 2\beta B_Y \Theta_2 = 0. \tag{17}$$

Multiplying Θ_1^T and Θ_2^T on the left-hand-side of the above equations, respectively, we have

$$\Theta_1^T A \Theta_2 - 2\alpha = 0$$
$$\Theta_2^T A^T \Theta_1 - 2\beta = 0.$$

Then, we only need to find the maximum α. From (16) and (17), we also have

$$A\Theta_2 = 2\alpha B_X \Theta_1 \quad \text{and} \quad A^T \Theta_1 = 2\alpha B_Y \Theta_2. \tag{18}$$

By transforming the above equations to the form of block matrix, we have

$$\begin{pmatrix} 0 & A \\ A^T & 0 \end{pmatrix} \begin{pmatrix} \Theta_1 \\ \Theta_2 \end{pmatrix} = 2\alpha \begin{pmatrix} B_X & 0 \\ 0 & B_Y \end{pmatrix} \begin{pmatrix} \Theta_1 \\ \Theta_2 \end{pmatrix}. \tag{19}$$

As a result, to find the optimal solution of (15) is equivalent to solve the generalized eigen-decomposition problem (19).

E. Orthogonality Constraint

Until now we have only computed the projection vector for the first dimension. It is noticeable that our objective function (15) is similar to the CCA. However, the relative orthogonality constraints in CCA cannot reflect realistic intention for our scheme. In this section, we use the orthogonalization method in [32] to compute the remaining vectors successively and make them mutually orthogonal by using the matrix composed by previous output vectors. Previous works [33], [34] have highlighted the benefits of orthogonality constraints, for instance, avoidance of overfitting and redundancy in representing the subspace. With this orthogonalization procedure, we can realize our whole algorithm.

Suppose, we have gained first p vectors $\Theta_1 = [\mathbf{a}_1, \ldots, \mathbf{a}_p]$ and $\Theta_2 = [\mathbf{b}_1, \ldots, \mathbf{b}_p]$. We need to find the solutions $\mathbf{a}_{p+1}$ and $\mathbf{b}_{p+1}$ to the optimization problem (15) with the orthogonal constraints

$$\mathbf{a}_1^T \mathbf{a}_{p+1} = \cdots = \mathbf{a}_p^T \mathbf{a}_{p+1} = \mathbf{b}_1^T \mathbf{b}_{p+1} = \cdots = \mathbf{b}_p^T \mathbf{b}_{p+1} = 0.$$

If we project all the local features in the image and text domains onto the subspaces $\text{span}(\mathbf{a}_1, \ldots, \mathbf{a}_p)^\perp$ and $\text{span}(\mathbf{b}_1, \ldots, \mathbf{b}_p)^\perp$, respectively, then the optimization process will be in these two subspaces and the output vectors will satisfy the orthogonal constraints. In fact, we only need to solve the linear equations $\Theta_1^T Z = 0$ and $\Theta_2^T Z = 0$ with the unknown variable Z to obtain the orthonormal basis $P_1 \in \mathbb{R}^{D_1 \times (D_1 - p)}$ and $P_2 \in \mathbb{R}^{D_2 \times (D_2 - p)}$ of the spaces $\text{span}(\mathbf{a}_1, \ldots, \mathbf{a}_p)^\perp$ and $\text{span}(\mathbf{b}_1, \ldots, \mathbf{b}_p)^\perp$, respectively, which is commonly used in linear algebra. With the basis P_1 and P_2, the projections are simply as follows:

$$\mathbb{R}^{D_1} \to \mathbb{R}^{D_1 - p} \cong \text{span}(\mathbf{a}_1, \ldots, \mathbf{a}_p)^\perp$$
$$\mathbf{x} \mapsto P_1^T \mathbf{x}$$

Algorithm 1 Binary Set Embedding

Input: The local feature sets X and Y from image and text domains respectively, the number of centroids K in the K-means, the parameter δ for the NN-search, the balancing parameter λ and the objective dimension (code length) d.

Output: The projection matrices Θ_1 and Θ_2 for the local features in image and text domains respectively.

1: Preprocessing: centralize $\mathbf{x}_i \leftarrow \mathbf{x}_i - \frac{1}{N}\sum_{k=1}^{N} \mathbf{x}_k$, $\mathbf{y}_j \leftarrow \mathbf{y}_j - \frac{1}{M}\sum_{k=1}^{M} \mathbf{y}_k$ for $i = 1, \ldots, N$, $j = 1, \ldots, M$;
2: Construct local feature pairing sets $\mathcal{P}_1$ and $\mathcal{P}_2$, and their corresponding pairwise labels ℓ_{ij}^X and ℓ_{ij}^Y by K-means clustering for image and text domains, respectively;
3: Compute the weights W_{ij}^X and W_{ij}^Y for the element-to-element structure and the similarities S_{ij}^X and S_{ij}^Y for the set-to-set structure;
4: Initialization: $\Theta_1 \leftarrow \emptyset$, $\Theta_2 \leftarrow \emptyset$, $P_1 \leftarrow I_{D_1}$ and $P_2 \leftarrow I_{D_2}$;
5: **for** $i = 1$ to d **do**
6: Project training local features in image and text domains onto the subspaces $span(\Theta_1)^\perp$ and $span(\Theta_2)^\perp$ by using the basis P_1 and P_2, respectively;
7: Solve the generalized eigen-decomposition problem (19) to obtain the vector $\begin{pmatrix} \mathbf{a}_i \\ \mathbf{b}_i \end{pmatrix}$ corresponding to the largest generalized eigenvalue;
8: Recover $\mathbf{a}_i \leftarrow P_1 \mathbf{a}_i$ and $\mathbf{b}_i \leftarrow P_2 \mathbf{b}_i$;
9: Update $\Theta_1 \leftarrow [\Theta_1, \mathbf{a}_i]$ and $\Theta_2 \leftarrow [\Theta_2, \mathbf{b}_i]$, and let P_1 and P_2 be the orthonormal basis of $span(\Theta_1)^\perp$ and $span(\Theta_2)^\perp$ by solving the corresponding linear equations respectively.
10: **end for**

and

$$\mathbb{R}^{D_2} \to \mathbb{R}^{D_2 - p} \cong \text{span}(\mathbf{b}_1, \ldots, \mathbf{b}_p)^\perp$$
$$\mathbf{y} \mapsto P_2^T \mathbf{y}.$$

In this case, we need to update all the matrices related to the local feature data

$$A \leftarrow P_1^T A P_2, \quad B_X \leftarrow P_1^T B_X P_1, \quad B_Y \leftarrow P_2^T B_Y P_2.$$

Now, we can repeat the eigen-decomposition procedure described in the above optimization section and output the optimal solutions $\mathbf{a}_{p+1} \in \mathbb{R}^{D_1 - p}$ and $\mathbf{b}_{p+1} \in \mathbb{R}^{D_2 - p}$. Finally, we recover $\mathbf{a}_{p+1}$ and $\mathbf{b}_{p+1}$ to the vectors in $\mathbb{R}^{D_1}$ and $\mathbb{R}^{D_2}$ by updating $\mathbf{a}_{p+1} \leftarrow P_1 \mathbf{a}_{p+1}$ and $\mathbf{b}_{p+1} \leftarrow P_2 \mathbf{b}_{p+1}$, respectively. We summarize BSE in Algorithm 1.

III. Voting Scheme for Local Feature Indexing

Having obtained the projection matrices Θ_1 and Θ_2, we can easily embed the training local features into binary hash codes by (1). For the query local features $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$, their hash codes are obtained by $H(\hat{\mathbf{x}}) = \text{sgn}(\Theta_1^T(\hat{\mathbf{x}} - (1/N)\sum_{j=1}^{N} \mathbf{x}_j))$ and $H(\hat{\mathbf{y}}) = \text{sgn}(\Theta_2^T(\hat{\mathbf{y}} - (1/M)\sum_{j=1}^{M} \mathbf{y}_j))$, respectively. Nevertheless, traditional linear search (e.g., Hamming distance ranking) with complexity $O(N)$ is not fast any more for our local feature hashing scenario, since N denotes the total

Algorithm 2 Voting Scheme for Local Feature Indexing

Input: The local feature sets X and Y from image and text domains respectively, the local feature set of query text (image) $Q = \{\mathbf{q}_1, \ldots, \mathbf{q}_m\}$, Hamming radius r and the learned projections Θ_1 and Θ_2.

Output: The retrieved images (texts) ranked by similarity.

1: Encoding all the local features into Hamming space via the Eq. (1) with Θ_1 and Θ_2;
2: Construct Hamming lookup table over the training set;
3: **for** $i = 1$ to m **do**
4: For the query hash code $H(\mathbf{q}_i)$, store all the possible image (text) indices fall into the Hamming lookup table within Hamming radius r;
5: Assign vector $\mathbf{v} = (v_1, \ldots, v_n) \in \mathbb{R}^n$ for the query Q and update $v_i \leftarrow v_i + 1$ if image (text) i appears in one Hamming lookup;
6: **end for**
7: Sort $(v_1, \ldots, v_n)$ in decreasing order;
8: **return** All the relevant images (texts) as the retrieved results.

number (at least 3M for a large-scale database) of local features. To accomplish the local feature-based visual retrieval, in this paper, we introduce a fast voting scheme for local feature indexing [35]. We first build the Hamming lookup table (also known as. the hashing table) for all the hash codes from the image and text domains. Given a query, we can find the bucket of corresponding hash codes in near constant time $O(1)$, and return all the data in the bucket as the retrieved results whether they are in the image and text domains.

After construction of the Hamming lookup table over the training set, we store the corresponding indices for all the hash codes of local features. In this way, for a text query Q, we search the hash code $H(\mathbf{q}_i)$ for each local feature $\mathbf{q}_k \in Q$ in the query $Q = \{\mathbf{q}_1, \ldots, \mathbf{q}_m\}$ over the Hamming lookup table within Hamming radius r and return the possible images' indices. It is noteworthy that the same bucket in the Hamming lookup table may store the indices from different images. Therefore, we vote and accumulate the times of each image's index appearing in relevant buckets and then rank them in a decreasing order. Specifically, we assign a vector $\mathbf{v} = (v_1, \ldots, v_n) \in \mathbb{R}^n$ for the query with the subscripts corresponding to the indices of the images in the gallery. Then, we update $v_i \leftarrow v_i + 1$ if there exists a local feature from sample i, which is within Hamming radius r in one Hamming lookup. The final retrieved samples are returned according to the descending order of $(v_1, \ldots, v_n)$. And for the image query, retrieval for the text results is performed by the same voting procedure. We summarize the above voting scheme in Algorithm 2.

IV. Experiments

In this section, we evaluate the proposed BSE method on two public data sets: the Wiki data set and the NUS-WIDE data set for cross-modal retrieval tasks. The relevant results show that our BSE significantly outperforms several state-of-the-art methods.

A. Data Sets

The Wiki data set [36] collects samples from Wikipedia "featured articles," containing 2866 image-text pairs in ten semantic classes. For each image, a set of 128-d SIFT [25] local features are extracted around salient points. For each text, we utilize the novel word-to-vector technique [27] to extract the 200-d semantic *word vectors* trained from the *first billion characters from Wikipedia*[1] for each word. Following the setting in the original paper [36], we take 2173 image-text pairs as the training set and the remaining 693 image-text pairs as the query set.

The NUS-WIDE data set [37] contains around 270 000 Web images associated with 81 ground truth concept classes. As in [38], we only use the most frequent 21 concept classes, each of which has abundant relevant images ranging from 5000 to 30 000. Unlike other data sets, each image in the NUS-WIDE data set is assigned with multiple semantic labels (tags). In this paper, two images belong to the same class, only if they share at least one common tag. Similarly, each image or text sample is represented by a set of SIFT features or a set of word vectors, respectively, as in the Wiki data set. We further sample randomly 100 images from each of the selected 21 tags to form a query set of 2100 images with the rest serving as the training set, since some of the remaining 60 tags contain too few images for the retrieval task.

B. Compared Methods and Experimental Settings

In our experiments, since few works focus on the local feature representation-based hashing scheme for cross-modal retrieval, we can only systematically compare the proposed BSE method with six prevailing global hashing methods for cross-modal retrieval tasks: CVH [11], MLBE [22], IMH [15], CMSSH [21], CHMIS [14], CMFH [16], and QCH [39]. For fair comparison, all the methods are implemented on the same SIFT features and word vectors in the image and text domains, respectively. Specifically, for the global methods, we use the vector of locally aggregated descriptors[2] (VLAD) [40] to embed sets of SIFT/word vectors from each image/text into an integrated representation. For CVH, IMH, CMSSH, and CMFH, the view-specific hashing codes can be learned while CHMIS is a cross-view fusion method, which learns integrated hash codes. We implement CVH, IMH ourselves and utilize the public codes of MLBE, CMSSH, CHMIS, CMFH, and QCH to calculate the results. All the parameters in compared methods are strictly selected according to their original publications.

For BSE, the parameter K for K-means is chosen from one of $\{100, 200, \ldots, 1000\}$ via tenfold cross validation on the training data and the best performed value of K is selected. Furthermore, the balancing parameter λ is also selected from one of $\{0.05, 0.1, \ldots, 0.5\}$, which yields the best performance by tenfold cross validation on the training set. δ for the NN-search is always fixed at 0.5 and the Hamming radius r is equal to 3.

[1] https://code.google.com/p/word2vec/

[2] The best number of clusters K used in VLAD is selected via tenfold cross-validation on the training data from $K = 100$ to $K = 1000$ with step 100.

TABLE I

MAP COMPARISON ON THE WIKI AND NUS-WIDE DATA SETS

Task	Method	Wiki						NUS-WIDE					
		16 bits	32 bits	48 bits	64 bits	80 bits	96 bits	16 bits	32 bits	48 bits	64 bits	80 bits	96 bits
Image to Text	CVH	0.210	0.163	0.142	0.129	0.137	0.132	0.371	0.382	0.426	0.413	0.405	0.393
	IMH	0.221	0.224	0.232	0.220	0.213	0.208	0.498	0.492	0.473	0.477	0.468	0.466
	MLBE	0.242	0.237	0.231	0.235	0.223	0.210	0.483	0.472	0.465	0.463	0.474	0.472
	CMSSH	0.231	0.233	0.238	0.242	0.245	0.247	0.501	0.504	0.510	0.513	0.515	0.518
	CHMIS	0.237	0.240	0.245	0.248	0.248	0.251	0.492	0.497	0.513	0.515	0.521	0.524
	CMFH	0.256	0.259	0.261	0.263	0.265	0.270	0.551	0.562	0.568	0.570	0.574	0.583
	QCH	0.238	0.251	0.253	0.257	0.261	0.264	0.517	0.538	0.546	0.555	0.561	0.566
	BSE	**0.260**	**0.268**	**0.272**	**0.277**	**0.281**	**0.284**	**0.572**	**0.574**	**0.574**	**0.580**	**0.583**	**0.597**
Text to Image	CVH	0.310	0.202	0.187	0.153	0.140	0.137	0.422	0.403	0.395	0.390	0.427	0.438
	IMH	0.503	0.496	0.483	0.493	0.462	0.467	0.493	0.508	0.512	0.504	0.492	0.497
	MLBE	0.483	0.432	0.319	0.262	0.231	0.220	0.510	0.501	0.472	0.486	0.488	0.493
	CMSSH	0.305	0.312	0.320	0.323	0.328	0.331	0.508	0.514	0.523	0.527	0.529	0.533
	CHMIS	0.237	0.240	0.245	0.248	0.248	0.251	0.492	0.497	0.513	0.515	0.521	0.524
	CMFH	0.601	0.605	0.612	0.618	0.625	0.633	0.650	0.674	0.688	0.707	0.707	0.711
	QCH	0.316	0.357	0.369	0.427	0.456	0.471	0.554	0.583	0.588	0.601	0.624	0.633
	BSE	**0.614**	**0.618**	**0.625**	**0.633**	**0.638**	**0.640**	**0.671**	**0.684**	**0.710**	**0.721**	**0.728**	**0.732**

All the compared methods (except "BSE") utilize vector of locally aggregated descriptors (VLAD) in this table.

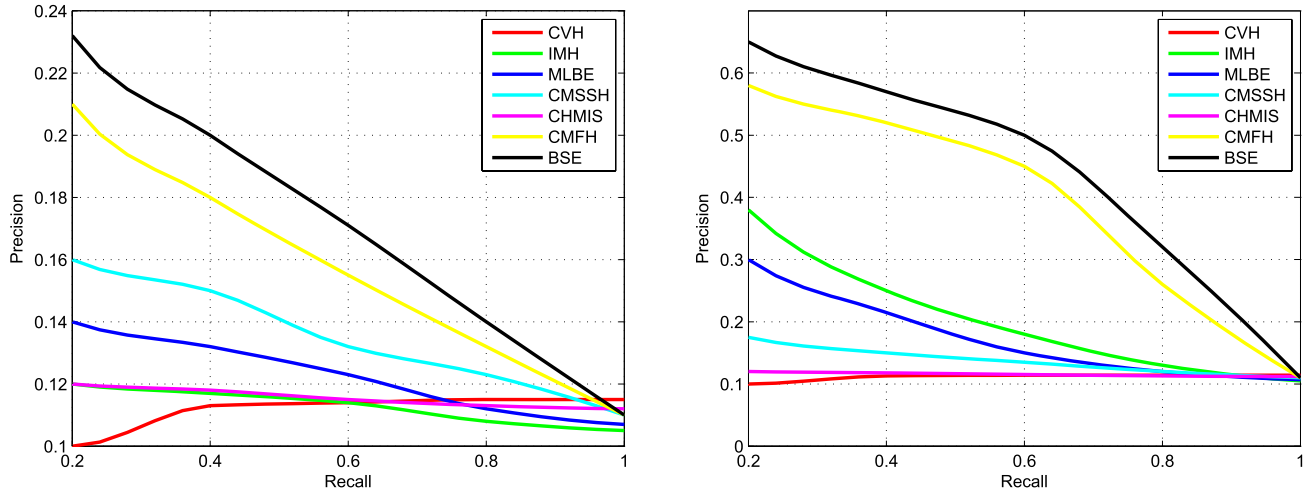


Fig. 2. Precision-recall curves of all compared algorithms on the Wiki data set with the code length of 32 b. (a) Wiki (I2T). (b) Wiki (T2I).

For the query phase, we use the voting scheme introduced in Section III to retrieve the neighbors of the query. We further report the mean average precision (MAP) of the top 50 retrieved images/documents for both of the data sets. It is defined as

$$\text{MAP} = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{50}\sum_{j=1}^{50}P(ij)$$

where $|Q|$ is the size of the query set and $P(ij)$ indicates the precision of the top j retrieved texts (images) of the ith image (text). In addition, all the methods are evaluated on six different lengths of codes $\{16, 32, 48, 64, 80, 96\}$. The selection of training and test samples is repeated five times for all the data sets and the compared methods, and we report the averages as the final results.

C. Results and Discussion

In this section, we will show the compared results of BSE and other methods, the parameter sensitivity analysis and the training size sensitivity analysis, respectively.

Table I shows the MAP on both Wiki and NUS-WIDE data sets. Since we focus on the cross-modal retrieval task, we show the corresponding results on two aspects, respectively: image query versus text database (I2T) and text query versus image database (T2I). From the table, we can observe that the MAP of a text query is generally higher than that of an image query. The reason is that the text can better describe the semantic meaning of the image-text pairs than the image. Given an image query, since it only describes the low-level visual information, it is difficult to find semantically similar texts for it accurately.

From Table I, it is easy to discover that the searching accuracies from CVH, IMH, and MLBE are always fluctuant with the increase of the code length. Specifically, in terms of IMH and MLBE, the best performances are usually achieved with small bits (i.e., 16 and 32 b, respectively) for both I2T and T2I on two data sets. For CVH, the highest results constantly appear with 16 b on the Wiki data set, while the best results are obtained with 96 b on the NUS-WIDE data set. Besides, we can also find that with the code length increasing, the results calculated by CMSSH, CHMIS, and CMFH

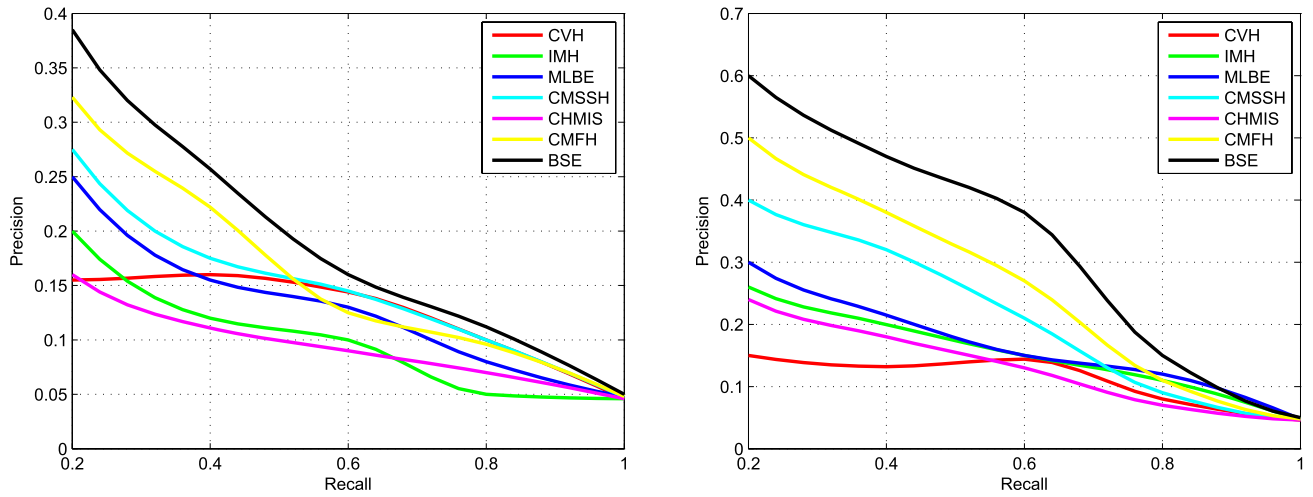


Fig. 3. Precision-recall curves of all compared algorithms on the NUS-WIDE data set with the code length of 32 b. (a) NUS-WIDE (I2T). (b) NUS-WIDE (T2I).

TABLE II
COMPUTATIONAL COMPLEXITY FOR THE TEST PHASE WITH THE 48-b CODES ON THE WIKI AND NUS-WIDE DATA SETS

Complexity / Datasets	Task	Training time (s)	Average coding time (ms) for each local feature	Average querying time (ms) for each image/text	HashTable size (MB)
Wiki (4×10^4 pairs)	Image to text	987.16	1.62	203.46	6.75
	Text to image	-	1.70	97.24	21.63
NUS-WIDE (8×10^5 pairs)	Image to text	2172.50	1.53	157.20	3.61
	Text to image	-	1.79	21.4	254.34

are getting higher on both data sets. In particular, CHMIS always achieves better performance than CMSSH for I2T search, but obtains lower accuracies than CMSSH for T2I search. Since CHMIS is regarded as a cross-view fusion method, it cannot directly compute the separated codes for the image and text domains, respectively. Thus, the same integrated codes are used for I2T and T2I and give the same performance on these two domains. Different from all the above conventional methods, our proposed BSE method successfully considers the relationship between local features on inter/intra data structures and completes retrieval via the local hash-based feature-indexing scheme. The related results demonstrate that our BSE can achieve significantly better performance than CVH, IMH, MLBE, CMSSH, CHMIS, and QCH for both I2T and T2I on Wiki and NUS-WIDE data sets and even outperforms the recent CMFH method. It is noticeable that CMFH's results are slightly lower than the results in the original paper [16]. The reason is the use of word vectors, which can be trained offline and independent of any specific data set unlike the provided data set-oriented LDA representation. In addition, we used 21 most frequent classes of the NUS-WIDE data set, which is larger than the ten largest concepts used in their paper. Beyond those, the precision-recall curves with the code length of 32 are also shown in Figs. 2 and 3. By measuring the area under curve, it can be obviously observed that BSE consistently performs better than other state-of-the-art methods. Moreover, the computational complexity for the test phase with the 48-b codes on the Wiki and NUS-WIDE data sets is in Table II. To make our method more convincing, Table III gives a comparison between the proposed BSE and other cross-modal metric learning methods which also map multiple modalities into a shared space. In particular, we use VLAD to construct global representations for images and texts as mentioned earlier and

TABLE III
MAP COMPARISON WITH STATE-OF-THE-ART CROSS-MODAL METRIC LEARNING METHODS ON BOTH DATA SETS

Dataset	Task	Code length	CCA	SCCA	BSE
Wiki	Image to Text	8	0.171	0.224	0.237
		16	0.178	0.218	0.260
		24	0.180	0.213	0.265
		32	0.179	0.210	0.268
		48	0.175	0.212	0.272
Wiki	Text to Image	8	0.201	0.460	0.608
		16	0.214	0.427	0.614
		24	0.233	0.401	0.615
		32	0.246	0.388	0.618
		48	0.244	0.372	0.625
NUS-WIDE	Image to Text	8	0.428	0.465	0.567
		16	0.420	0.460	0.572
		24	0.413	0.454	0.573
		32	0.404	0.451	0.574
		48	0.397	0.446	0.574
NUS-WIDE	Text to Image	8	0.433	0.472	0.665
		16	0.427	0.470	0.671
		24	0.419	0.465	0.677
		32	0.405	0.453	0.684
		48	0.401	0.448	0.710

All the compared methods (except "BSE") utilize vector of locally aggregated descriptors (VLAD) in this table.

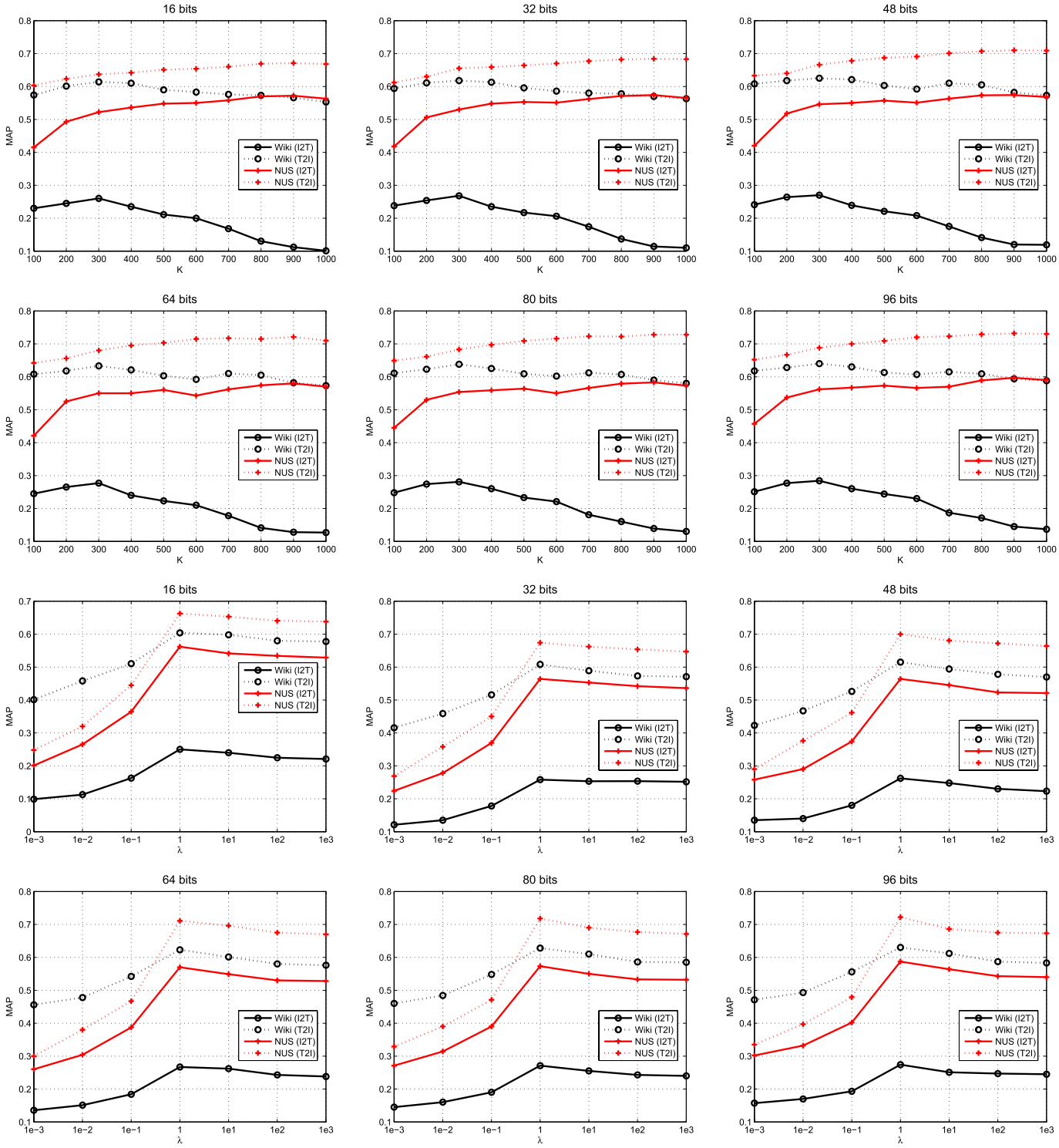


Fig. 4. Comparison of the MAP of BSE with respect to parameters K and λ on the Wiki and NUS-WIDE data sets with different bit lengths.

then CCA and supervised CCA (SCCA) [39] are utilized to learn the real-valued low-dimensional data for cross-modal retrieval.

D. Parameter Sensitivity

In this section, we illustrate the sensitivity of two parameters: the number of clusters K and the balance parameter λ, on the Wiki and NUS-WIDE data sets with different bit lengths. We report the best results for a fixed parameter with varying other parameters in Fig. 4. As we can see from the figure, the results on two data sets at all different code lengths have the similar tendency. For the parameter K, we can observe that a small value of K ($K = 300$) in the K-means works better for the Wiki data set with all bit lengths, since it is a relatively small data set containing only 2173 image-text data with ten semantic classes for training.

TABLE IV
MAP COMPARISON WITH DIFFERENT SETTINGS OF THE PROPOSED BSE

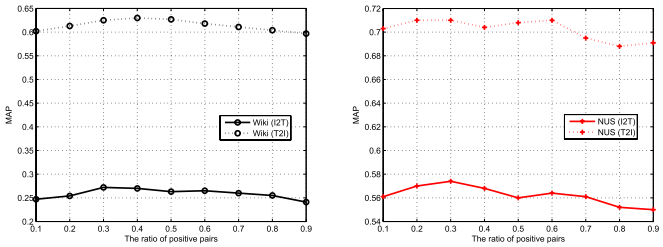
Task	Method	Wiki						NUS-WIDE					
		16 bits	32 bits	48 bits	64 bits	80 bits	96 bits	16 bits	32 bits	48 bits	64 bits	80 bits	96 bits
Image to Text	Only element-to-element structure preserving	0.098	0.121	0.135	0.136	0.145	0.157	0.201	0.224	0.258	0.260	0.271	0.302
	Only set-to-set structure preserving	0.202	0.251	0.223	0.238	0.240	0.245	0.528	0.536	0.521	0.528	0.532	0.540
	BSE without orthogonality constraint	0.244	0.250	0.253	0.259	0.266	0.270	0.546	0.558	0.567	0.573	0.575	0.582
	BSE	**0.260**	**0.268**	**0.272**	**0.277**	**0.281**	**0.284**	**0.572**	**0.574**	**0.574**	**0.580**	**0.583**	**0.597**
Text to Image	Only element-to-element structure preserving	0.401	0.415	0.423	0.456	0.460	0.471	0.248	0.268	0.290	0.300	0.329	0.335
	Only set-to-set structure preserving	0.577	0.570	0.570	0.576	0.583	0.585	0.638	0.647	0.664	0.670	0.671	0.674
	BSE without orthogonality constraint	0.608	0.612	0.618	0.623	0.627	0.634	0.666	0.677	0.692	0.703	0.715	0.720
	BSE	**0.614**	**0.618**	**0.625**	**0.633**	**0.638**	**0.640**	**0.671**	**0.684**	**0.710**	**0.721**	**0.728**	**0.732**

"Only element-to-element structure preserving" refers to $\lambda = 0$ in Eq. (15). On the contrary, "Only set-to-set structure preserving" refers to $\lambda = +\infty$ in Eq. (15). "BSE without orthogonality constraint" indicates solving Eq. (15) under CCA-like solution without orthogonal projection optimization.

While for the NUS-WIDE data set, the best value of K always tends to be large ($K = 900$) for both I2T and T2I search. Furthermore, from the whole perspective, the tendency of the accuracies on NUS-WIDE with the change of K goes stably, which indicates that our final results are not sensitive to the choice of K. In Fig. 4, we also demonstrate the sensitivity of the balance parameter λ. It is discovered that with the increase of λ, the search results always rapidly grow and then slightly drop down with all bit lengths. The best results are usually achieved when $\lambda = 1$ on both Wiki and NUS-WIDE data sets. However, the final accuracies are more sensitive on the NUS-WIDE data set when λ takes various values compared with those on the Wiki data set. The comparison has shown the fact that we can reduce the range near the best point in the future tuning of parameters, i.e., the range for tuning K is proportional to the training size and the range for tuning λ is around 1. In addition, we evaluate the effectiveness of element-to-element structure preserving and set-to-set structure preserving in (15) on both data sets, respectively. From Table IV, we can observe that only preserving element-to-element structure (i.e., $\lambda = 0$) or set-to-set structure (i.e., $\lambda = +\infty$) individually cannot achieve the best performance. To further explore the advantages of the orthogonality constraint in (15), the results of BSE without orthogonal projection learning in Section II-E are also included in Table IV, where the learning procedure is similar to CCA.

E. Training Size Sensitivity

For the training phase, although we always fix the number of the training samples as mentioned in Section IV-A, the number of constructed local feature pairs for element-to-element preserving can be varied. Theoretically, more local pairs used in the training phase will lead to better results. If there exists N local features, the maximum number of pairs can be N^2. However, N for large data sets can be over a million. It is infeasible to utilize all the local pairs in training due to the computational costs. Thus, in our experiments, we randomly select a subset of the pairs, which contains 30% positive pairs and 70% negative pairs, similar to [41], during the training phase. Table V shows the corresponding results by varying the number of pairs. Obviously, the proposed BSE can achieve significantly better results when the pair number equals 8×10^4 and 1.6×10^6 on two data sets with the total numbers of local feature pairs 4×10^{11} and 1×10^{13}, respectively. In addition, different ratios for the number of positive and negative feature pairs with 48-b codes on both Wiki and NUS-WIDE data sets are shown in Fig. 5, as well. From Fig. 5, it is observed that the performance on Wiki is quite stable when varying the ratio of positive and negative pairs, while for NUS-WIDE there always exists fluctuation in terms of MAP.



(a) (b)

Fig. 5. MAP (48 b) via different ratios of positive/negative pair construction on the (a) Wiki and (b) NUS-WIDE data sets.

F. Generalization

In this experiment, we train the hash functions of different methods on the combination of the Wiki and NUS-WIDE data sets. Since the local features from the image and text domains that we used, i.e., SIFT and word vectors, are irrelevant to any specific data set, we can unite the features of the Wiki and NUS-WIDE data sets together to form a larger data set. For the global methods, we still use the above VLAD representations. As shown in Table VI, the results of almost every method are between the corresponding ones of the Wiki and NUS-WIDE data sets in Table I. Generally, the text-to-image results on the combined data set are better than the ones on the Wiki data set since more sufficient semantic information for images can be learned in the larger data set. In contrast, the image-to-text results on the combined data set are lower than the ones on the NUS-WIDE data set for the reason that the images in NUS-WIDE are only with several tags rather than documents and the retrieval results are possibly the words in Wiki. In addition, our method has significantly outperformed the other state-of-the-art cross-modality hashing methods and improved the text-to-image MAP compared with the results on both data sets.

TABLE V
EFFECT OF TRAINING PAIR SIZE ON MAP AT 48 b

Datasets	Pair Size	Image to Text	Text to Image
Wiki	1×10^4	0.202	0.558
	2×10^4	0.224	0.586
	4×10^4	0.251	0.608
	8×10^4	0.272	0.625
NUS-WIDE	2×10^5	0.493	0.640
	4×10^5	0.518	0.672
	8×10^5	0.541	0.695
	1.6×10^6	0.574	0.710

TABLE VI
MAP COMPARISON ON THE COMBINATION OF THE WIKI AND NUS-WIDE DATA SETS

Task	Code length	CVH	IMH	MLBE	CMSSH	CHMIS	CMFH	BSE
Image to Text	16	0.264	0.301	0.306	0.341	0.317	0.351	**0.441**
	32	0.251	0.296	0.313	0.350	0.329	0.358	**0.450**
	48	0.244	0.304	0.315	0.357	0.336	0.367	**0.457**
	64	0.237	0.310	0.311	0.368	0.348	0.407	**0.465**
	80	0.230	0.323	0.320	0.372	0.364	0.397	**0.479**
	96	0.227	0.335	0327	0.380	0.382	0.385	**0.487**
Text to Image	16	0.337	0.501	0.487	0.401	0.317	0.620	**0.681**
	32	0.279	0.493	0.445	0.408	0.329	0.627	**0.693**
	48	0.256	0.481	0.348	0.419	0.336	0.631	**0.703**
	64	0.233	0.458	0.364	0.400	0.348	0.630	**0.736**
	80	0.247	0.453	0.356	0.403	0.364	0.649	**0.738**
	96	0.252	0.444	0.348	0.398	0.382	0.664	**0.742**

V. CONCLUSION

In this paper, a novel cross-modal hashing scheme called BSE has been presented. Aiming for a general representation that is independent of any data set, we have employed local feature descriptors for both image and text modalities. BSE associates the local feature set of images with the semantic information of the corresponding documents and embeds them into a common Hamming space. Due to the nature of local features, BSE simultaneously preserves the element-to-element and set-to-set structures, which are correspondent to the data points and the source information of local features, respectively, in the intramodel relationship. Extensive results have shown that BSE outperforms the state-of-the-art methods in terms of cross-modal retrieval tasks. Our future work aims to generalize our approach to carry out the cross-modal task for data from multiple modalities.

REFERENCES

[1] J. C. Pereira *et al.*, "On the role of correlation and abstraction in cross-modal multimedia retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 3, pp. 521–535, Mar. 2014.

[2] R. Datta, D. Joshi, J. Li, and J. Z. Wang, "Image retrieval: Ideas, influences, and trends of the new age," *Comput. Surv.*, vol. 40, no. 2, pp. 1-60, Apr. 2008.

[3] Y. Luo, D. Tao, C. Xu, C. Xu, H. Liu, and Y. Wen, "Multiview vector-valued manifold regularization for multilabel image classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 5, pp. 709–722, May 2013.

[4] L. Duan, D. Xu, and I. W. Tsang, "Domain adaptation from multiple sources: A domain-dependent regularization approach," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 3, pp. 504–518, Mar. 2012.

[5] J. Liu, Y. Jiang, Z. Li, Z.-H. Zhou, and H. Lu, "Partially shared latent factor learning with multiview data," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 6, pp. 1233–1246, Jun. 2015.

[6] B. Kulis and T. Darrell, "Learning to hash with binary reconstructive embeddings," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, pp. 1042–1050.

[7] L. Liu, L. Shao, and X. Li, "Evolutionary compact embedding for large-scale image classification," *Inf. Sci.*, vol. 316, pp. 567–581, Sep. 2015.

[8] M. Yu, L. Liu, and L. Shao, "Structure-preserving binary representations for RGB-D action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 8, pp. 1651–1664, Aug. 2016.

[9] J. Qin, L. Liu, Z. Zhang, Y. Wang, and L. Shao, "Compressive sequential learning for action similarity labeling," *IEEE Trans. Image Process.*, vol. 25, no. 2, pp. 756–769, Feb. 2016.

[10] A. Gionis, P. Indyk, and R. Motwani, "Similarity search in high dimensions via hashing," in *Proc. VLDB*, Sep. 1999, pp. 518–529.

[11] S. Kumar and R. Udupa, "Learning hash functions for cross-view similarity search," in *Proc. Int. Joint Conf. Artif. Intell.*, Jul. 2011, pp. 1360–1365.

[12] L. Liu, M. Yu, and L. Shao, "Projection bank: From high-dimensional data to medium-length binary codes," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 2821–2829.

[13] Y. Zhen and D.-Y. Yeung, "Co-regularized hashing for multimodal data," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1376–1384.

[14] D. Zhang, F. Wang, and L. Si, "Composite hashing with multiple information sources," in *Proc. Conf. Res. Develop. Inf. Retr.*, 2011, pp. 225–234.

[15] J. Song, Y. Yang, Y. Yang, Z. Huang, and H. T. Shen, "Inter-media hashing for large-scale retrieval from heterogeneous data sources," in *Proc. Int. Conf. Manage. Data*, 2013, pp. 785–796.

[16] G. Ding, Y. Guo, and J. Zhou, "Collective matrix factorization hashing for multimodal data," in *Proc. IEEE Conf. Comput. Vis. Patt. Recognit.*, 2014, pp. 2075–2082.

[17] L. Liu, M. Yu, and L. Shao, "Multiview alignment hashing for efficient image search," *IEEE Trans. Image Process.*, vol. 24, no. 3, pp. 956–966, Mar. 2015.

[18] Y. Weiss, A. Torralba, and R. Fergus, "Spectral hashing," in *Proc. Adv. Neural Inf. Process. Syst.*, 2008, pp. 1753–1760.

[19] L. Liu and L. Shao, "Sequential compact code learning for unsupervised image hashing," *IEEE Trans. Neural Netw. Learn. Syst.*, doi: 10.1109/TNNLS.2015.2495345.

[20] R. Salakhutdinov and G. Hinton, "Semantic hashing," *Int. J. Approx. Reasoning*, vol. 50, no. 7, pp. 969–978, Jul. 2009.

[21] M. Bronstein, A. Bronstein, F. Michel, and N. Paragios, "Data fusion through cross-modality metric learning using similarity-sensitive hashing," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, p. 5.

[22] Y. Zhen and D.-Y. Yeung, "A probabilistic model for multimodal hash function learning," in *Proc. SIGKDD*, 2012, pp. 940–948.

[23] X. Zhu, Z. Huang, H. T. Shen, and X. Zhao, "Linear cross-modal hashing for efficient multimedia search," in *Proc. Int. Conf. Multimedia*, 2013, pp. 143–152.

[24] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar. 2003.

[25] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[26] L. Liu, C. Shen, L. Wang, A. van den Hengel, and C. Wang, "Encoding high dimensional local features by sparse coding based Fisher vectors," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 1143–1151.

[27] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, Oct. 2013, pp. 3111–3119.

[28] P. D. Turney and P. Pantel, "From frequency to meaning: Vector space models of semantics," *J. Artif. Intell. Res.*, vol. 37, no. 1, pp. 141–188, 2010.

[29] J.-T. Chien and Y.-C. Ku, "Bayesian recurrent neural network for language modeling," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 2, pp. 361–374, Feb. 2016.

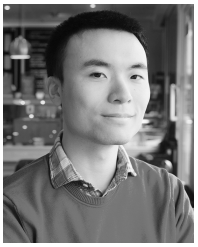
[30] O. Boiman, E. Shechtman, and M. Irani, "In defense of nearest-neighbor based image classification," in *Proc. IEEE Conf. Comput. Soc. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[31] W. Liu, J. Wang, R. Ji, Y.-G. Jiang, and S.-F. Chang, "Supervised hashing with kernels," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2074–2081.

[32] M. Yu, L. Shao, X. Zhen, and X. He, "Local feature discriminant projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 9, pp. 1908–1914, Sep. 2016.

[33] G. Hua, M. Brown, and S. A. J. Winder, "Discriminant embedding for local image descriptors," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2007, pp. 1–8.

[34] J. Duchene and S. Leclercq, "An optimal transformation for discriminant and principal component analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 10, no. 6, pp. 978–983, Nov. 1988.
[35] H. Jegou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *Proc. Eur. Conf. Comput. Vis.*, Oct. 2008, pp. 304–317.
[36] N. Rasiwasia *et al.*, "A new approach to cross-modal multimedia retrieval," in *Proc. Int. Conf. Multimedia*, 2010, pp. 251–260.
[37] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. Zheng, "Nus-wide: A real-world Web image database from national university of singapore," in *Proc. ACM Int. Conf. Image Video Retr.*, 2009, p. 48.
[38] W. Liu, J. Wang, S. Kumar, and S. Chang, "Hashing with graphs," in *Proc. Int. Conf. Mach. Learn.*, 2011, pp. 1–8.
[39] B. Wu, Q. Yang, W.-S. Zheng, Y. Wang, and J. Wang, "Quantized correlation hashing for fast cross-modal search," in *Proc. 24th Int. Conf. Artif. Intell.*, 2015, pp. 3946–3952.
[40] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 3304–3311.
[41] G. Shakhnarovich, "Learning task-specific similarity," Ph.D. dissertation, Dept. Elect. Eng. Comput. Sci., Massachusetts Inst. Technol., Cambridge, MA, USA, 2005.

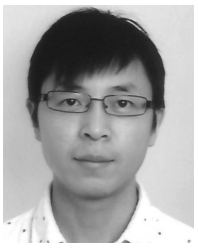


Mengyang Yu (S'14) received the B.S. and M.S. degrees from the School of Mathematical Sciences, Peking University, Beijing, China, in 2010 and 2013, respectively. He is currently pursuing the Ph.D. degree with the Department of Computer and Information Sciences, Northumbria University, Newcastle upon Tyne, U.K.

His current research interests include computer vision, machine learning, and data mining.

Li Liu received the B.Eng. degree in electronic information engineering from Xi'an Jiaotong University, Xi'an, China, in 2011, and the Ph.D. degree from the Department of Electronic and Electrical Engineering, University of Sheffield, Sheffield, U.K., in 2014.

He is currently a Research Fellow with the Department of Computer and Information Sciences, Northumbria University, Newcastle upon Tyne, U.K. His current research interests include computer vision, machine learning, and data mining.



Ling Shao (M'09–SM'10) is a Professor with the Department of Computer and Information Sciences at Northumbria University, Newcastle upon Tyne, U.K. and a Visiting Professor with Southwest University, Chongqing, China. Previously, he was a Senior Lecturer (2009–2014) with the Department of Electronic and Electrical Engineering at the University of Sheffield and a Senior Scientist (2005–2009) with Philips Research, The Netherlands. His research interests include Computer Vision, Image/Video Processing and Machine Learning. He is an associate editor of IEEE Transactions on Image Processing, IEEE Transactions on Cybernetics and several other journals. He is a Fellow of the British Computer Society and the Institution of Engineering and Technology.