Average Approximate Hashing-Based Double Projections Learning for Cross-Modal Retrieval

Xiaozhao Fang, Kaihang Jiang, Na Han, Shaohua Teng, Guoxu Zhou, and Shengli Xie, *Fellow, IEEE*

***Abstract*—Cross-modal retrieval has attracted considerable attention for searching in large-scale multimedia databases because of its efficiency and effectiveness. As a powerful tool of data analysis, matrix factorization is commonly used to learn hash codes for cross-modal retrieval, but there are still many shortcomings. First, most of these methods only focus on preserving locality of data but they ignore other factors such as preserving reconstruction residual of data during matrix factorization. Second, the energy loss of data is not considered when the data of cross-modal are projected into a common semantic space. Third, the data of cross-modal are directly projected into a unified semantic space which is not reasonable since the data from different modalities have different properties. This article proposes a novel method called average approximate hashing (AAH) to address these problems by: 1) integrating the locality and residual preservation into a graph embedding framework by using the label information; 2) projecting data from different modalities into different semantic spaces and then making the two spaces approximate to each other so that a unified hash code can be obtained; and 3) introducing a principal component analysis (PCA)-like projection matrix into the graph embedding framework to guarantee that the projected data can preserve the main energy of data. AAH obtains the final hash codes by using an average approximate strategy, that is, using the mean of projected data of different modalities as the hash codes. Experiments on standard databases show that the proposed AAH outperforms several state-of-the-art cross-modal hashing methods.**

***Index Terms*—Average approximation, cross-modal retrieval, different semantic spaces, hashing, residual preservation.**

I. Introduction

Information technology, with its superior computing power and human–computer interaction capability, has risen rapidly and has become the main force of rapid growth of social productivity. The main advantage of information technology is the ability to process large amounts of data in a short time, which is later pushed to the peak by the rise of artificial intelligence. To allow computers to retrieve the information at minimal cost, hashing retrieval techniques have emerged. Hashing retrieval has a series of huge advantages, such as fast retrieval speed and small storage space, so it is compatibility used for large-scale data retrieval. The principle is to map the feature space of data to the semantic space and obtain hash codes from it for retrieval. The initial appearance of hashing is unimodal methods [1]–[3]. They can only learn hash codes from unimodal data, such as retrieving images with images. They convert data into uniform and compact binary codes for retrieval, which greatly improve the retrieval speed [4], [5]. However, with the advent of the Internet era, we have to confront more multimodal data in real life. Therefore, it is rare to design a cross-modal retrieval method over multimodal datasets. As we have known, multimedia data of the same kind or similar things have similar semantic descriptions. For example, the pictures of cats in Wiki have corresponding text descriptions. Our purpose is to find the similarities and differences between the texts and pictures for constructing the semantic space and learning the uniform hash codes.

In recent years, many excellent methods have emerged to facilitate the development of cross-modal retrieval [6]–[11]. Existing hashing methods can be roughly classified into two categories: 1) unsupervised and 2) supervised methods [12]. Unsupervised methods commonly explore correlations of heterogeneous data to learn binary codes. For example, collective matrix factorization hashing (CMFH) [13] utilizes the collective matrix factorization method to effectively extract features from original data, and transforms features into hash codes quickly. Since it lacks the supervision of semantic labels, the overall accuracy of it is low.

To overcome this problem, supervised cross-modal hashing methods utilize semantic labels of data to obtain higher accuracies. Therefore, more supervised hashing methods have

Manuscript received December 10, 2020; revised March 23, 2021; accepted May 15, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 62006048, Grant 61772141, Grant 61972102, and Grant U191140003; in part by the Guangdong Provincial Natural Science Foundation under Grant 2021A1515012017; in part by the Science and Technology Planning Project of Guangdong Province, China, under Grant 2019B020208001 and Grant 2019B110210002; and in part by the Guangzhou Science and Technology Planning Project under Grant 201903010107 and Grant 201802010042. This article was recommended by Associate Editor C.-T. Lin. *(Corresponding author: Na Han.)*

Xiaozhao Fang is with the School of Automation and the Key Laboratory of Intelligent Detection and The Internet of Things in Manufacturing, Ministry of Education, Guangdong University of Technology, Guangzhou 510006, China (e-mail: xzhfang168@126.com).

Kaihang Jiang and Shaohua Teng are with the School of Computer Science and Technology, Guangdong University of Technology, Guangzhou 510006, China (e-mail: jkh1650290810@163.com; shteng@gdut.edu.cn).

Na Han is with the College of Computer Science, Guangdong Polytechnic Normal University, Guangzhou 510665, China (e-mail: hannagdut@126.com).

Guoxu Zhou is with the School of Automation and the Key Laboratory of Intelligent Information Processing and System Integration of IoT, Ministry of Education, Guangdong University of Technology, Guangzhou 510006, China (e-mail: gx.zhou@gdut.edu.cn).

Shengli Xie is with the School of Automation and the Guangdong–HongKong–Macao Joint Laboratory for Smart Discrete Manufacturing, Guangdong University of Technology, Guangzhou 510006, China (e-mail: shlxie@gdut.edu.cn).

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TCYB.2021.3081615.

Digital Object Identifier 10.1109/TCYB.2021.3081615

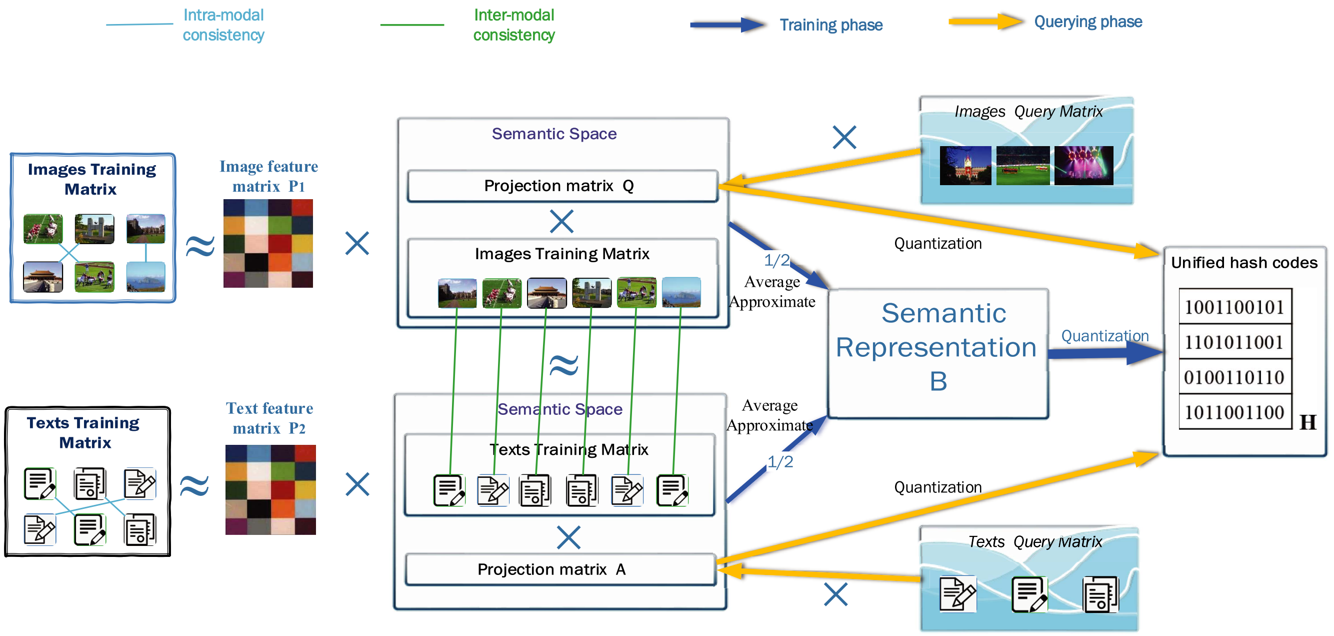

Fig. 1. Framework of AAH. In the training phase, the multimodal data of image and text can be extracted to different semantic spaces with intramodal consistency (referred to in Section III-B). And the unified hash codes can be quantified by averaging different semantic spaces. In the querying phase, we directly multiply the query matrix (image or text) by the projection matrix, and the binary codes of the query matrix are produced by quantifying it.

been proposed in recent years [14]–[19]. Discrete cross-modal hashing (DCH) learns independent binary codes for each modality of one instance by explicitly capturing the discriminative ability [20]. Supervised matrix factorization for cross-modal hashing (SMFH) exploits the label similarity matrix and graph regularization to guarantee the discriminative ability and similarities among multimodal for the learned binary codes [21]. However, SMFH and DCH only preserve the intermodal correlations while ignoring the intramodal correlations. Semantic correlation maximization (SCM) integrates semantic labels into the hashing learning procedure for large-scale multimodal data [22]. But the hash codes generated by SCM do not require independence from each hash bit and, thus, it cannot guarantee that the learned binary codes are semantically discriminative. Semantics-preserving hashing (SePH) tries to transform a semantic affinity matrix into a probability distribution and approximate it in Hamming space via minimizing the Kullback–Leibler divergence (KLD) which may lead to high training time cost [23].

In general, most supervised cross-modal hashing methods share some common properties. First, they intend to learn binary codes that preserve the global and local structure information of data, while ignoring the other factors such as preserving reconstruction residual of data. Second, they always utilize matrix factorization to decompose data of different modalities to the semantic space directly. However, the matrix factorization methods may lose the main energy of data since the dimensionality of data may be reduced. Third, most cross-modal retrieval methods decompose data of different modalities into a common semantic space, then learn the unified hash codes from the common space. However, it is not reasonable that the data from different modalities are directly projected into a common semantic space because the data from different modalities have different properties.

To address the aforementioned challenges, we propose a novel cross-modal hashing method called average approximate hashing (AAH), which projects data of different modalities into different semantic spaces while preserves reconstruction residual and main energy of data, then obtains the hash codes by approximating the average of semantic spaces from different modalities. The main concept of AAH is that the data from different modalities should be projected into different semantic spaces and hash codes should preserve the properties of original data. In AAH, we unify the locality and reconstruction residual preserving terms into a graph embedding framework to extract features from different modalities, respectively. Moreover, we utilize a principal component analysis (PCA)-like projection matrix to guarantee that the projected data can preserve the main energy of data. Fig. 1 depicts the workflow of the proposed AAH. First, the intramodal and intermodal consistency is constructed from the label information. Second, intramodal consistency combines with the PCA-like projection matrix to extract the features from different modalities, respectively, which also projects heterogeneous data into different semantic spaces. Third, we utilize another graph embedding constraint term along with intermodal consistency to make the different semantic spaces approximate to each other. Finally, the unified hash codes are generated by quantizing the average of semantic spaces from different modalities.

Comparing with existing supervised cross-modal retrieval methods, the main contributions of the proposed AAH are summarized as follows.

1) The proposed AAH method integrates the locality, residual, and energy preserving into a graph embedding framework, which, to our knowledge, may be the first

work that integrates these three different components into a unified optimization objective.

2) Instead of directly making the binary codes of the image modal and the text modal approximate to each other, AAH uses the projected data to replace the binary codes and requires the projected data of the image modal and the text modal to be close. This makes the projected data have more freedom to learn more accurate binary codes.
3) The average approximate strategy is utilized to obtain the final hash codes. In other words, the final binary codes and the average of different projected data of image and text modalities are required to be equal to each other.
4) The proposed AAH is extensively evaluated on three benchmark datasets with various retrieval scenarios and the results show that our proposed AAH is competitive with several state-of-the-art methods.

The remainder of this article is organized as follows. Section II introduces the main principles of this article and its related theoretical analysis. Section III presents the detail of the proposed method. In Section IV, we present the details of optimization in the algorithm. Section V presents the experimental results and comparisons using three real-world datasets. Finally, the conclusions are presented in Section VI.

II. Related Works

In this section, we introduce the related works of our AAH method, that is, PCA and locality preserving projections (LPP), which are highly related to our work in this article.

A. Principal Component Analysis

As we knew, PCA can be utilized to perform the data processing and dimensionality reduction [24], [25]. Given a dataset with m dimensions and n samples, that is, $X = [x_1, x_2 \cdots x_n] \in \Re^{m\times n}$, PCA aims to find a linear subspace with dimension d ($d \ll n$) that the data points lie on this linear subspace. Assume $U \in \Re^{m\times d}$ is a projection matrix with the d orthogonal unit vectors as columns. We can use matrix U to project the original data into a low-dimensional subspace and the low-dimensional feature representation can preserve the main energy of data. The general objective function of PCA is as follows:

$$\min_{U^TU=I} \left\|X - UU^TX\right\|_F^2. \tag{1}$$

We can use $Y = U^TX$ to obtain the low-dimensional representation Y of X. By minimizing the reconstruction error of $\|X - UU^TX\|_F^2$, the obtained Y can preserve the main energy of X.

B. Locality Preserving Projections

The LPP method has been widely used in dimensionality reduction, data analysis and information retrieval, and so on [26], [27]. It first uses the neighborhood information of the data to build a similarity matrix and then exploit such similarity matrix to perform many data analysis task. Given a dataset $X = [x_1, x_2, \ldots, x_n] \in \Re^{m\times n}$, we utilize k nearest neighbors or ϵ-neighborhoods to construct the similarity matrix $W \in \Re^{n\times n}$. To obtain the low-dimensional representation $Y \in \Re^{d\times n}$ of X and make Y preserve the locality of data, LPP learns a projection matrix $A \in \Re^{m\times d}$ ($d \ll m$) to minimize the following objective:

$$\sum_{ij}(y_i - y_j)^2 W_{ij} = \sum_{ij}\left(A^Tx_i - A^Tx_j\right)^2 W_{ij}. \tag{2}$$

We can obtain Y by using $Y = A^TX$ ($y_i = A^Tx_i$). Recently, LPP has been further used for cross-modal retrieval. For example, SMFH [15] uses $W_{ij} = s_{ij}^{(1)} + s_{ij}^{(2)} + c_{ij}$ as the similarity matrix, where c_{ij} is label similarity of two samples and s_{ij} is local similarity to model the intramodal similarity of two samples in each modality. In this way, W_{ij} can preserve both local information and label information. But it only preserves the local information of data while ignores global information.

III. Proposed Method

In this section, we present the details of our method. To simplify the presentation, we focus on learning hash codes from cross-modal data (i.e., image and text).

A. Problem Formulation

Supposing that we have a set of n cross-modal training samples. Let $X = [x_1, x_2, \ldots, x_n] \in \Re^{d_1\times n}$ be the image data matrix and $Y = [y_1, y_2, \ldots, y_n] \in \Re^{d_2\times n}$ be the text data matrix, respectively, where d_1 and d_2 are the number of dimensionality of image and text data and n is the number of samples. Each sample x_i or y_i has a class label vector $\ell_i \in \Re^{c\times 1}$, where c is the number of classes. We define ℓ_i as follows: if x_i or y_i is from the kth class ($k = 1, 2, \ldots, c$), then the kth entry of ℓ_i is one and all the other entries are zero. Our objective is to learn hash codes B from X and Y, where $B = [b_1, b_2, \ldots, b_n] \in \{-1, 1\}^{r\times n}$ is the unified hash codes for cross-modal retrieval, and r is the length of the bit of the hash code.

B. Similarity Graph Construction

Due to its success in information revealing the local geometric structure preserving, similarity graph is widely utilized in pattern recognition and computer vision [8], [28]–[31]. Since the image or text has multitag and thus we define the similarity graph of intramodal S^m($m = v, t$; when $m = v$ means images modality, and $m = t$ means texts modality) as follows. Suppose two different samples x_i^m and x_j^m from the same modality as follows:

$$S_{ij}^m = \begin{cases} 1, & d_{ij} \ge 1 \\ 0, & d_{ij} < 1. \end{cases} \tag{3}$$

We define S_{ij}^m by calculating the distance between the labels. Suppose that the label of x_i^m is $\ell_i^{(m)}$ and the label of x_j^m is $\ell_j^{(m)}$, we obtain $d_{ij} = (\ell_i^{(m)})^T\ell_j^{(m)}$ by evaluating the similarities between x_i^m and x_j^m. We consider $S_{ij}^m = 1$ when $d_{ij} \ge 1$, $S_{ij}^m = 0$ when $d_{ij} < 1$. In this way, we can preserve the semantic similarity information of data from the intramodal. So we call it intramodal consistency.

Meanwhile, we define the similarity graph of intermodal L_{ij} of two different samples (e.g., x_i and y_j) from different modalities as follows:

$$L_{ij} = \begin{cases} 1, & d_{ij} \geq 1 \\ 0, & d_{ij} < 1. \end{cases} \tag{4}$$

The calculation way of L_{ij} is the same as S_{ij}^m while the difference is that L_{ij} deal with samples from intermodal. $L_{ij} = 1$ indicates that two samples from different modalities are similar or are from the same class. Because it evaluates the similarities between the samples from intermodal, we call it intermodal consistency.

C. Energy and Residual Preserving

PCA [see (1)] uses a single projection matrix to perform dimensionality reduction and simultaneously preserve the main energy of data. It is worth noting that we normalize the X and Y matrices. And the normalization of X is $X = ([X - E(X)]/[D(X)])$ and the normalization of Y is $Y = ([Y - E(Y)]/[D(Y)])$, where $E(x)$ is the mean of x and $D(x)$ is the variance of x. Inspired by PCA, AAH uses two different matrices, that is, P_1 and Q for image data (or P_2 and A for text data) to guarantee that the data representation of $Q^T X$ and $A^T Y$ can preserve the main energy of data. This provides a more flexible way than PCA to perform dimensionality reduction and energy preserving. Moreover, to make the data representation of $Q^T X$ and $A^T Y$ can preserve the locality of data, we further propose a novel reconstruction residual preserving method. By considering these issues, we propose the following objective:

$$\begin{aligned} \min_{P_1, P_2, Q, A} \quad & \sum_{i=1}^{n} \sum_{j=1}^{n} \left\| x_i - P_1 Q^T x_j \right\|_2^2 S_{ij}^v \\ & + \theta \sum_{i=1}^{n} \sum_{j=1}^{n} \left\| y_i - P_2 A^T y_j \right\|_2^2 S_{ij}^t \\ \text{s.t.} \quad & P_1^T P_1 = I, \; P_2^T P_2 = I. \end{aligned} \tag{5}$$

$P_1 \in \Re^{d_1 \times r}$ and $Q \in \Re^{d_1 \times r}$ are two different matrices for image data, and $P_2 \in \Re^{d_2 \times r}$ and $A \in \Re^{d_2 \times r}$ are also two different matrices for text data, and $\theta \geq 0$ is a tradeoff parameter. Suppose two samples x_i and x_j from the image modality, and $S_{ij}^v = 1$ means x_i is similar with x_j. Minimizing $\|x_i - P_1 Q^T x_j\|_2^2 S_{ij}^v$ means that the term of $\|x_i - P_1 Q^T x_j\|_2^2$ should be minimized. This demonstrates that the Euclid distance between sample x_i and the reconstruction sample $P_1 Q^T x_j$ is minimized. Suppose x_i is the nearest neighbor of x_j and x_i and x_j can be represented by $P_1 Q^T x_k$ and $P_1 Q^T x_h$ as $x_i = P_1 Q^T x_k$ and $x_j = P_1 Q^T x_h$, respectively, we have $\|x_i - x_j\|_2^2 = \|x_i - P_1 Q^T x_h\|_2^2 = \|P_1 Q^T x_k - x_j\|_2^2$. This demonstrates that the Euclidean distance between the reconstructed sample and the original sample is the same with that of the corresponding two original samples. We impose S on the reconstruction residual to preserve the nearest neighbor relationships among the original samples and the reconstructed samples. To achieve the minimum of objective (5), we have to minimize $\sum_{i=1}^n \sum_{j=1}^n \|x_i - P_1 Q^T x_j\|_2^2 S_{ij}^v$ which means $x_i \approx P_1 Q^T x_j$. This indicates that $P_1 Q^T x_i \approx P_1 Q^T x_j$ approximately holds. Therefore, similar samples have similar reconstruction data. That is, to say, similar samples have similar reconstruction residuals. For the text modality, we have the similar observation.

D. Global Approximation and Label Consistency

The purpose of cross-modal retrieval is to learn the unified binary codes. Thus, the binary codes of image modality and text modality are required to be approximately equal. In our method, we refer to it as a global approximation. We formulate the global approximation as $\min \|Q^T X - A^T Y\|_F^2$. When two different samples from different modalities but with the same label come, we expect these two samples to be close together which is also called label consistency among different modalities. Suppose two different samples x_i and y_j are, respectively, from image and text modalities but they share the same label (i.e., $L_{ij} = 1$), we can minimize $\sum_{i=1}^n \sum_{j=1}^n \|Q^T x_i - A^T y_j\|_2^2 L_{ij}$ to guarantee that these different projection samples ($Q^T x_i$ and $A^T y_j$) from different modalities but with the same label can be pushed together. By considering these two factors, we have the following objective:

$$\min_{Q,A} \alpha \sum_{i=1}^{n} \sum_{j=1}^{n} \left\| Q^T x_j - A^T y_j \right\|_2^2 L_{ij} + \beta \left\| Q^T X - A^T Y \right\|_F^2 \tag{6}$$

where $\alpha \geq 0$ and $\beta \geq 0$ are the balance parameters.

E. Average Approximation

To learn unified binary codes, we use the strategy of average approximation to ensure that the information of image and text can be all used to learn the binary codes. We formulate the average approximation as follows:

$$\begin{aligned} \min_{Q,A} \quad & \left\| B - \tfrac{Q^T X + A^T Y}{2} \right\|_F^2 \\ \text{s.t.} \quad & B \in \{-1, 1\}^{r \times n}. \end{aligned} \tag{7}$$

In (7), we obtain hash codes B which can not only preserve the different properties of two modalities ($Q^T X$ and $A^T Y$) but also fuse the information of different modalities to learn the unified binary codes.

F. Overall Objective Function

The overall objective function of AAH that combines energy and residual preserving, global approximation, label consistency, and average approximation is as follows:

$$\begin{aligned} \min_{P_1, P_2, Q, A, B} \quad & \sum_{i=1}^{n} \sum_{j=1}^{n} \left\| x_i - P_1 Q^T x_j \right\|_2^2 S_{ij}^v \\ & + \theta \sum_{i=1}^{n} \sum_{j=1}^{n} \left\| y_i - P_2 A^T y_j \right\|_2^2 S_{ij}^t \\ & + \alpha \sum_{i=1}^{n} \sum_{j=1}^{n} \left\| Q^T x_i - A^T y_j \right\|_2^2 L_{ij} \\ & + \beta \left\| Q^T X - A^T Y \right\|_F^2 + \left\| B - \tfrac{Q^T X + A^T Y}{2} \right\|_F^2 \\ \text{s.t.} \quad & P_1^T P_1 = I, \; P_2^T P_2 = I, \; B \in \{-1, 1\}^{r \times n}. \end{aligned} \tag{8}$$

In objective function (8), we jointly learn Q, A, P_1, P_2, B and other parameters. In the training phase, the hash codes of training data are obtained by quantifying B as $\mathrm{sgn}(B)$, where $\mathrm{sgn}(\cdot)$ denotes the elementwise sign function.

In addition, if we have image query set X' and text query set Y', by quantifying Q^TX' as $\mathrm{sgn}(Q^TX')$ to obtain images query hash codes and quantifying A^TY' as $\mathrm{sgn}(A^TY')$ to obtain texts query hash codes in the query phase.

IV. Optimization Algorithm

A. *Optimization*

It is easy to prove that the optimization objective of (8) is nonconvex for each variable, but it is convex with respect to any one of these variables when the others are fixed. Therefore, we can use the alternating direction method of multipliers (ADMMs) [32] to solve the optimization problem (8).

Specifically, the first term of (8) can be further simplified as

$$\begin{aligned}
&\sum_{i=1}^{n}\sum_{j=1}^{n}\left\|x_i - P_1Q^Tx_j\right\|_2^2 S_{ij}^v\\
&= \sum_{i=1}^{n}\sum_{j=1}^{n}\mathrm{Tr}\Big(\big(x_i - P_iQ^TX_j\big)\big(x_i - P_iQ^TX_j\big)^T\Big)S_{ij}^v\\
&= \sum_{i=1}^{n}\sum_{j=1}^{n}\mathrm{Tr}\Big(x_ix_i^T - 2x_ix_j^TQP_1^T\Big)S_{ij}^v + \mathrm{Tr}\Big(Q^Tx_jx_j^TQ\Big)S_{ij}^v\\
&= \mathrm{Tr}\big(XD^vX^T - 2XS^vX^TQP_1^T\big) + \mathrm{Tr}\big(Q^TXD^vX^TQ\big) \qquad (9)
\end{aligned}$$

where $D^v = \mathrm{diag}(D_{11}^v, D_{22}^v, \ldots, D_{nn}^v)$ is a diagonal matrix, and $D_{ii}^v = \sum_j S_{ij}^v$, and S^v is the matrix representation of S_{ij}^v. $\mathrm{Tr}(\cdot)$ is the trace operator. Similarly, the second term of (8) can be simplified as

$$\begin{aligned}
\theta\sum_{i=1}^{n}\sum_{j=1}^{n}\left\|y_i - P_2A^Ty_j\right\|_2^2 S_{ij}^t &= \theta\mathrm{Tr}\big(YD^tY^T - 2YS^tY^T\mathrm{AP}_2^T\big)\\
&\quad + \theta\mathrm{Tr}\big(A^TYD^tY^TA\big) \qquad (10)
\end{aligned}$$

where $D^t = \mathrm{diag}(D_{11}^t, D_{22}^t, \ldots, D_{nn}^t)$ is a diagonal matrix, and $D_{ii}^t = \sum_j S_{ij}^t$, and S^t is the matrix representation of S_{ij}^t. Again, the third term of (8) can be rewritten as.

$$\begin{aligned}
&\alpha\sum_{i=1}^{n}\sum_{j=1}^{n}\left\|Q^Tx_i - A^Ty_j\right\|_2^2 L_{ij}\\
&= \alpha\sum_{i=1}^{n}\sum_{j=1}^{n}\mathrm{Tr}\Big(\big(Q^Tx_i - A^Ty_i\big)\big(Q^Tx_i - A^Ty_i\big)^T\Big)L_{ij}\\
&= \alpha\sum_{i=1}^{n}\sum_{j=1}^{n}\mathrm{Tr}\Big(Q^Tx_ix_i^TQ - 2Q^Tx_iy_j^TA + A^Ty_jy_j^TA\Big)L_{ij}\\
&= \alpha\mathrm{Tr}\Big(Q^TXD^lX^TQ - 2A^TYLX^TQ\Big)\\
&\quad + \alpha\mathrm{Tr}\Big(A^TYD^lY^TA\Big) \qquad (11)
\end{aligned}$$

where $D^l = \mathrm{diag}(D_{11}^l, D_{22}^l, \ldots, D_{nn}^l)$ is a diagonal matrix, and $D_{ii}^l = \sum_j L_{ij}$. L is the matrix representation of L_{ij}

Then, the objective function (8) can be simplified as

$$\begin{aligned}
\min_{P_1,P_2,Q,A,B}\ & \mathrm{Tr}\big(XD^vX^T - 2XS^vX^TQP_1^T\big) + \mathrm{Tr}\big(Q^TXD^vX^TQ\big)\\
& + \theta\mathrm{Tr}\big(YD^tY^T - 2YS^tY^T\mathrm{AP}_2^T\big) + \theta\mathrm{Tr}\big(A^TYD^tY^TA\big)\\
& + \alpha\mathrm{Tr}\Big(Q^TXD^lX^TQ - 2A^TYLX^TQ\Big)\\
& + \alpha\mathrm{Tr}\Big(A^TYD^lY^TA\Big)\\
& + \beta\mathrm{Tr}\big(Q^TXX^TQ - 2A^TYX^TQ\big) + \beta\mathrm{Tr}\big(A^TYY^TA\big)\\
& + \mathrm{Tr}\bigg(BB^T - \big(Q^TX + A^TY\big)B^T\\
&\qquad + \frac{(Q^TX+A^TY)(Q^TX+A^TY)^T}{4}\bigg)\\
\text{s.t. }\ & P_1^TP_1 = I, P_2^TP_2 = I, B \in \{-1, 1\}^{r\times n}. \qquad (12)
\end{aligned}$$

To optimize objective function (12), we convert problem (12) into the following formula by introducing two auxiliary variables $U = Q^TX$ and $V = A^TY$ as follows:

$$\begin{aligned}
\min_{P_1,P_2,Q,A}\ & \mathrm{Tr}\big(XD^vX^T - 2XS^vU^TP_1^T\big) + \mathrm{Tr}\big(UD^vU^T\big)\\
& + \theta\mathrm{Tr}\big(YD^vY^T - 2YS^tV^TP_2^T\big) + \theta\mathrm{Tr}\big(VD^tV^T\big)\\
& + \alpha\mathrm{Tr}\Big(UD^lU^T - 2VLU^T + VD^lV^T\Big)\\
& + \beta\mathrm{Tr}\big(UU^T - 2VU^T + VV^T\big)\\
& + \mathrm{Tr}\Big(BB^T - (U+V)B^T + \tfrac{UU^T+2VU^T+VV^T}{4}\Big)\\
\text{s.t. }\ & P_1^TP_1 = I, P_2^TP_2 = I, U = Q^TX\\
& V = A^TY, B \in \{-1, 1\}^{r\times n}. \qquad (13)
\end{aligned}$$

According to [33], (13) can be further extended to the augmented Lagrangian function

$$\begin{aligned}
&L(P_1, P_2, Q, A, U, V, C_1, C_2)\\
&= \mathrm{Tr}\big(XD^vX^T - 2XS^VU^TP_1^T\big) + \mathrm{Tr}\big(UD^vU^T\big)\\
&\quad + \theta\mathrm{Tr}\big(YD^vY^T - 2YS^tV^TP_2^T\big) + \theta\mathrm{Tr}\big(VD^tV^T\big)\\
&\quad + \alpha\mathrm{Tr}\Big(UD^lU^T - 2VLU^T + VD^lV^T\Big)\\
&\quad + \beta\mathrm{Tr}\big(UU^T - 2VU^T + VV^T\big)\\
&\quad + \mathrm{Tr}\Big(BB^T - (U+V)B^T + \tfrac{UU^T+2VU^T+VV^T}{4}\Big)\\
&\quad + \tfrac{\mu}{2}\Big(\left\|Q^TX - U\right\|_F^2 + \left\|A^TY - V\right\|_F^2\Big)\\
&\quad + \langle C_1, Q^TX - U\rangle + \langle C_2, A^TY - V\rangle\\
&\text{s.t. }\ P_1^TP_1 = I, P_2^TP_2 = I, B \in \{-1, 1\}^{r\times n} \qquad (14)
\end{aligned}$$

where $\langle A, B\rangle = \mathrm{Tr}(A^TB)$. C_1 and C_2 are the Lagrangian multipliers. And μ is a positive penalty parameter. By solving (14) alternately, we can obtain the solutions of all variables $P_1, P_2, Q, A, U, V, C_1,$ and C_2. The detailed steps are as follows.

Step 1 (Update Q and A): For problem (14), fix P_1, P_2, A, U, V, B and update Q by minimizing the following formula:

$$\begin{aligned}
L(Q) &= \tfrac{\mu}{2}\left\|Q^TX - U\right\|_F^2 + \big\langle C_1, Q^TX - U\big\rangle\\
&= \tfrac{\mu}{2}\left\|Q^TX - U + \tfrac{C_1}{\mu}\right\|_F^2. \qquad (15)
\end{aligned}$$

By setting the derivative $([\partial L(Q)]/\partial Q) = 0$, we obtain

$$\frac{\partial L(Q)}{\partial Q} = 2XX^TQ - 2XH_1^T = 0 \qquad (16)$$

where $H_1 = U - (C_1/\mu)$. Then, we can obtain the closed solution of Q

$$Q = \left(XX^T\right)^{-1} XH_1^T. \tag{17}$$

We can obtain the solution of A by solving the following problem:

$$L(A) = \tfrac{\mu}{2}\left\|A^T Y - V + \tfrac{C_2}{\mu}\right\|_F^2. \tag{18}$$

We notice that Q and A are mathematically symmetric, so we can obtain the solution of A is as follows:

$$A = \left(YY^T\right)^{-1} YH_2^T \tag{19}$$

where $H_2 = V - (C_2/\mu)$.

Step 2 (Update U and V): If we fix P_1, P_2, Q, A, V, and B, (14) can be converted into

$$\begin{aligned} L(U) = {} & \mathrm{Tr}\left(-2XS^v U^T P_1^T\right) + \mathrm{Tr}\left(UD^v U^T\right) \\ & + \alpha \mathrm{Tr}\left(UD^l U^T - 2VLU^T\right) + \beta \mathrm{Tr}\left(UU^T - 2VU^T\right) \\ & + \mathrm{Tr}\left(\tfrac{-4UB^T + UU^T + 2VU^T}{4}\right) + \tfrac{\mu}{2}\left\|Q^T X - U + \tfrac{C_1}{\mu}\right\|_F^2. \end{aligned} \tag{20}$$

By setting the derivative $([\partial L(U)]/\partial U) = 0$, and define $H_3 = Q^T X + (C_1/\mu)$, we obtain

$$U = EF^{-1} \tag{21}$$

where $E = (2P_1^T XS^v + \mu H_3 + 2\alpha VL + (2\beta - [1/2])V + B)$ and $F = (2(1+\alpha)D^l + (2\beta + 1/2 + \mu)I)$.

Similarly, we can obtain the solution of V by minimizing the following objective:

$$\begin{aligned} L(V) = {} & \mathrm{Tr}\left(-2YS^t V^T P_2^T\right) + \mathrm{Tr}\left(VD^t V^T\right) \\ & + \alpha \mathrm{Tr}\left(VD^l V^T - 2VLU^T\right) + \beta \mathrm{Tr}\left(VV^T - 2VU^T\right) \\ & + \mathrm{Tr}\left(\tfrac{-4VB^T + VV^T + 2VU^T}{4}\right) + \tfrac{\mu}{2}\left\|A^T Y - V + \tfrac{C_2}{\mu}\right\|_F^2. \end{aligned} \tag{22}$$

By setting the derivative $([\partial L(V)]/\partial V) = 0$, and define $H_4 = A^T Y + (C_2/\mu)$, we obtain

$$V = JK^{-1} \tag{23}$$

$J = (2\theta P_2^T YS^t + \mu H_4 + 2\alpha UL + (2\beta - [1/2])U + B)$ and $K = (2(\theta + \alpha)D^l + (2\beta + 1/2 + \mu)I)$.

Step 3 (Update P_1 and P_2): Fix P_2, Q, A, U, V, and B and update P_1 by solving the following minimization problem:

$$\min_{P_1^T P_1 = I} \mathrm{Tr}\left(-2XS^v U^T P_1^T\right). \tag{24}$$

Minimizing (24) is equivalent to the following maximization problem:

$$\max_{P_1^T P_1 = I} \mathrm{Tr}\left(XS^v U^T P_1^T\right). \tag{25}$$

Problem (25) is an orthogonal Procrustes problem and can be simply solved by performing SVD, where SVD is the singular value decomposition (SVD) operation [24]. For example, let $\mathrm{SVD}(XS^v U^T) = U_1 S V_1^T$, then we obtain P_1 as

$$P_1 = U_1 V_1^T. \tag{26}$$

Similarly, we can obtain P_2 by solving the following maximization problem:

$$\max_{P_2^T P_2 = I} \mathrm{Tr}\left(YS^t V^T P_2^T\right). \tag{27}$$

Let $\mathrm{SVD}(YS^t V^T) = U_2 S V_2^T$, then we obtain P_2 as

$$P_2 = U_2 V_2^T. \tag{28}$$

Step 4 (Update B): The final step is to solve for the unified hash codes B by fixing other variables, the problem in (14) becomes

$$\begin{aligned} & L(B) = \mathrm{Tr}\left(BB^T - (U+V)B^T + \tfrac{UU^T + 2VU^T + VV^T}{4}\right) \\ & \quad \text{s.t. } B \in \{-1, 1\}^{r \times n}. \end{aligned} \tag{29}$$

The problem in (29) is NP-hard for directly optimizing the unified binary codes B. However, we can solve a relaxed problem through discarding the discrete constraints, reconstraining B as $0 \le B \le 1$. Then, by setting the derivative $([\partial L(B)]/\partial B) = 0$, we obtain

$$B = \frac{U + V}{2}. \tag{30}$$

Finally, the approximate binary codes for the training samples can be obtained, then we obtain unified binary codes B by quantization as $B = \mathrm{sgn}(B)$.

Step 5 (Update C_1, C_2, and μ): Lagrangian multipliers C_1 and C_2, and penalty parameter μ can be updated by using the following formulas:

$$C_1 = C_1 + \mu\left(Q^T X - U\right) \tag{31}$$

$$C_2 = C_2 + \mu\left(A^T Y - V\right) \tag{32}$$

$$\mu = \min(\rho\mu, \mu_{\max}). \tag{33}$$

To obtain the final solution, we alternately update all parameters according to the above steps until it converges. The detailed procedures of the proposed algorithm are summarized in Algorithm 1.

B. Multiple Modalities

The proposed AAH can be easily extended for multimodal (more than two modalities) scenario by using two different strategies.

The first strategy is simple to repeat AAH for each of two modalities. For example, there would be $\binom{2}{3} = 3$ combinations for three modalities (i.e., text, video, and image). However, this strategy requires high time complexity, due to it has to repeat AAH three times for obtaining three hash codes.

The second strategy is more efficient than the first one. We build a new joint model based on the principle of AAH. Suppose that there are instances X consisting of $m (m \ge 2)$ modalities' data, denoted by $X^{(t)} (t = 1, 2, \ldots, m)$. We used matrices $S^{(t)}$ to represent the intramodal consistency for the t-th modality, matrices $L^{(t,l)}$ to represent the intermodal consistency for the t-th modality and lth modality and P_t to represent the feature matrix for the t-th modality. Also, we used W_t to

Algorithm 1 Optimization Algorithm in AAH

```
procedure
  Input: Images training matrix $X$, texts training matrix $Y$; label of samples $Z$, code length $r$; model parameters $\theta$, $\beta$, $\alpha$.
  Initialization: $Q$, $A$, $P_1$, $P_2$ and $B$ by random matrices respectively. Construct the matrix of $S^v$, $S^t$, $L$ and $D^v$, $D^t$, $D^l$. $C_1 = C_2 = 0$, $\mu = 0.1$, $\rho = 1.01$, $\mu_{max} = 10^8$, $U = Q^T X$ ,$V = A^T Y$
  while not converged do
    1. Update $Q$ using (17);
    2. Update $A$ using (19);
    3. Updata $U$ by solving (21);
    4. Updata $V$ by solving (23);
    5. Update $P_1$ using (26);
    6. Update $P_2$ using (28);
    7. Update $B$ using (30);
    8. Update $C_1$, $C_2$, $\mu$ by (31), (32), and (33), respectively;
  End while
  Output: The hash codes matrix $B$, projection matrices $Q$ and $A$.
end procedure
```

represent the projection matrix for the t-th modality. Then, the objective of AAH can be written as

$$\begin{aligned}\min_{\{P_t, W_t, B\}_{t=1,2,\ldots,m}} \quad & \sum_{t=1}^{m} \lambda_t \left(\sum_{i=1}^{n} \sum_{j=1}^{n} \left\| x_i^{(t)} - P_t W_t^T x_j^{(t)} \right\|_2^2 S_{ij}^{(t)} \right) \\ & + \alpha \sum_{t \neq l}^{m} \sum_{i=1}^{n} \sum_{j=1}^{n} \left\| W_t^T x_i^{(t)} - W_l^T x_j^{(l)} \right\|_2^2 L_{ij}^{(t,l)} \\ & + \beta \sum_{t \neq l}^{m} \left\| W_t^T X^{(t)} - W_l^T X^{(l)} \right\|_F^2 \\ & + \left\| B - \frac{1}{m} \sum_{t=1}^{m} W_t^T X^{(t)} \right\|_F^2 \\ \text{s.t.} \quad & P_t^T P_t = I, \;\; B \in \{-1, 1\}^{r \times n} \end{aligned} \tag{34}$$

where λ_t is the parameter to balance the influences of different modalities and α and β are the same parameters with the original objective function (8). Regarding the solutions of W_t, P_t, and B, it is not difficult to observe that they could be solved by the previously described optimization algorithm of AAH.

C. Computational Complexity Analysis

In this section, we show the computational complexity of the optimization algorithm of AAH. Specifically, for an $m \times m$ matrix, the computational complexity is $O(m^3)$ in the inverse operation, such as (17) and (19). And the computational complexity of the conventional SVD operation is $O(m^3)$ for an $m \times n$ matrix, such as (26) and (28). It is worth nothing that in (21) and (23), the inverse operation of F and K can be directly obtained by calculating the reciprocal of its elements because they are diagonal matrices. Suppose T is the number of iterations of the algorithm, the overall computational complexity of Algorithm 1 includes $O(T(d_1(d_1^2 + 2d_1 n + nr)))$ for solving (17), $O(T(d_2(d_2^2 + 2d_2 n + nr)))$ for solving (19), $O(T(nr(2d_1 + 2n)))$ for solving (21), $O(T(nr(2d_2 + 2n)))$ for solving (23), $O(T(d_1(n^2 + nr + d_1^2)))$ for solving (26), $O(T(d_2(n^2 + nr + d_2^2)))$ for solving (28), $O(T(d_1 nr))$ for solving (31), and $O(T(d_2 nr))$ for solving (32), respectively. Since $r, d_1, d_2, T \ll n$, and we set $d = \max\{d_1, d_2\}$ and $r < d$, the overall computational complexity of the training stage is $O(dn^2 T)$, where T is usually less than 10 in practice.

TABLE I
STATISTICS OF THREE BENCHMARK DATASETS

Data sets name	Dimensionality		Data set size	Training Size	Query Size	Class
	Image	Text				
Wiki	128	10	2866	2173	693	10
MIR-Flickr	150	500	16738	15902	836	24
NUS-WIDE	128	1000	186577	25000	1866	10

In the search phase, the complexity is $O(rdm)$, where m is the number of query samples. So the query computational complexity of AAH is low.

Therefore, the optimization rules in AAH are suitable for practical cross-modal hashing retrievals, and its effectiveness and convergence will be discussed in the next section.

V. EXPERIMENT

To validate the effectiveness of the proposed AAH, we conducted experiments on three benchmark datasets, that is: 1) Wiki; 2) MIR-Flickr; and 3) NUS-WIDE, which are widely used in several state-of-the-art cross-modal retrieval methods [34]–[36]. So far, these datasets are the largest publicly available multimodal datasets which are fully paired and labeled. In this article, we compared AAH with seven state-of-the-art shallow cross-modal hashing methods. Further, we analyzed the differences between different algorithms from more indicators, that is, Precision–Recall, TopK–precision, parameter sensitivity, running time comparison, and so on. Finally, we draw a conclusion of AAH through comparative analysis. Due to space limitation, the more experimental analyses (i.e., effects of training size, effects of PCA-like function, t-SNE visualization, and distance of reconstruction residual and different modalities) are moved to *Supplement*.

A. Datasets

Wiki: It consists of 2866 image–text pairs which are collected from Wikipedia, and each pair is labeled with one of 10 semantic classes. It includes 2173 training instances and 693 test instances. And each instance is an image–text pair with a label. Following the method in [6], each image was represented as 4096-D features extracted by the Caffe implementation of AlexNet and then represented as 128-D features by PCA. Each text is obtained by the probability distribution over ten topics learned from the latent Dirichlet allocation (LDA) model. So, the Wiki dataset contains 2173 training set pairs with training labels and 693 testing set pairs with testing labels.

MIR-Flickr: It contains 25 000 instances downloaded from Flicker. Each image or text is associate with a number of users assigned labels and it uses manual annotations from the

24 unique provided tags. According to [23], after preprocessing, we selected textual tags that appear at least 20 times and deleted image tag pairs without text tags or manual annotation tags. For each image–text pair, we used a 150-D edge histogram feature vector to represent the image, and a 500-D feature vector derived from PCA on the text label index vector to represent the text. Following the setting of [13], [14], and [34], we randomly selected 5% image–tag pairs as the query set, and the remaining part of image–tag pairs as the training set.

NUS-WIDE: It is a real-world Web image datasets including 260 648 instances downloaded from the Flickr website, and each instance contains an image and its associated textual tags and textual descriptions. Tagging ground truth for 81 semantic concepts is provided for evaluation. Each image is first extracted from the Caffe implementation of VGG Net [37], and then represented as 128-D features by PCA, and each text is represented by an index vector of the most frequent 1000 tags. Due to the scarcity of some tags, we selected only the ten most common tags and their corresponding 186 577 images. Here, we randomly selected 1% of the datasets as query sets and the remaining 184 711 as the training sets. However, as the NUS-WIDE dataset is too large and the training time is too long, we only randomly select a part of it (25 000 image–text pairs) in the actual experiment.

Table I introduces many properties of these datasets.

B. Evaluation Metrics

The mean average precision (mAP) is used to measure the performance of the cross-modal hashing methods. Given a query with a list of ϕ retrieved instances, the average precision (AP) is defined as

$$\text{AP} = \frac{1}{N}\sum_{i=1}^{\phi}\text{Precision}(i)\delta(i) \tag{35}$$

where N is the number of ground-truth relevant instances in retrieved set ϕ, and $\delta(i) = 1$ if the ith retrieved instances are relevant to query and $\delta(i) = 0$ otherwise. Precision(i) is the precision of the top i retrieved instances. Then, we can average the AP of all queries to obtain mAP.

To further analyze AAH, we used another baseline to evaluate retrieval quality. Regarding the metric of TopK–precision, it can reflect the accuracy of the number of top K images/texts presents to the user. TopK–precision returns the points given any hamming radius, and it is widely used to measure the performance of information retrieval.

C. Baseline Methods

We compared the AAH model with seven state-of-the-art multimodal hashing methods, which not only includes the conventional work CMFH but also the recent work generalized semantic preserving hashing (GSPH). We briefly introduced all the compared methods as follows.

1) The latent semantic sparse hashing (LSSH) method [38] utilizes sparse coding and matrix factorization to learn latent semantic features for images and texts, respectively, and then projects them to a joint space for generating unified hash codes.
2) The CMFH method [13] obtains unified hash codes by utilizing collective matrix factorization in the latent factor model from different modalities of one instance.
3) The SMFH method [21] tackles the multimodal hashing problem with collective non-negative matrix factorization and graph regularization across the different modalities.
4) The SCM method [22] aims to reconstruct the similarity matrix and integrate the semantic labels into the hashing learning procedure by learning multimodal hash codes.
5) The SePH method [23] transforms a semantic affinity matrix into a probability distribution and approximates it in the Hamming space via minimizing the KLD.
6) The DCH method [20] utilizes class labels to learn modality-specific hash functions and obtains the unified binary codes by retaining the discrete constraints.
7) The GSPH method [39] makes use of the integrated hashing methods to solve different scenarios, such as SL-P, SL-U, ML-P, and ML-U, in the same framework, and generates the unified binary codes with the probability that different modalities produce hash values.

Among these above seven methods, CMFH and LSSH are unsupervised methods whereas the others are supervised methods. The reason for comparing the proposed AAH with CMFH and LSSH is because they both use matrix factorization methods to generate hash codes.

D. Discussion

For the parameter sensitivity, we presented the analysis of the experimental results with different parameters, which verifies that AAH can achieve stable performance in a wide range of parameter values. Specifically, we used the same parameter settings for MIR-Flickr and NUS-WIDE datasets, that is, $\theta = 10$, $\alpha = 1$, and $\beta = 10$, and another parameter setting $\theta = 1$, $\alpha = 10$, and $\beta = 10$ for the Wiki dataset. We almost quoted mAP results of these compared methods from the original paper, or we quoted them from other papers if the mAP results do not appear in the original paper, such as we cited the results from other papers [20], [22] for the LSSH method. We also implemented almost all of the baseline methods on the personal computer, and obtained the Precision–Recall curve and TopK–Precision curve according to the experimental results. The experiments are conducted on a personal computer with an Intel Core i9-9900K running at 3.6 GHz with 16 cores, 32-GB RAM, and a 64-bit Windows-10 operating system.

E. Results on Wiki

The mAP values of AAH and seven baseline methods on the Wiki dataset are reported in Table II. The Precision–Recall curves and TopK–Precision curves are plotted in Figs. 2 and 3, respectively. From Table II, we have the following observations. First, DCH yields the best mAP results of image query text on the Wiki dataset. Second, the mAP values of the text query image of the proposed AAH outperform the other methods. Third, the mAP values of the image query text are

TABLE II
mAP PERFORMANCE OF AAH AND BASELINES ON THE BENCHMARK DATASETS WITH VARIED CODE LENGTHS. THE BEST PERFORMANCE IS SHOWN IN BOLDFACE

Task	Method	Wiki				MIR-Flicker				NUS-WIDE			
		16bits	32bits	64bits	128bits	16bits	32bits	64bits	128bits	16bits	32bits	64bits	128bits
Image query Text	LSSH [38]	0.2330	0.2340	0.2387	0.2340	0.5784	0.5804	0.5797	0.5816	0.4933	0.5006	0.5096	0.5084
	CMFH [13]	0.2538	0.2582	0.2619	0.2648	0.6480	0.6597	0.6693	0.6752	0.5591	0.5698	0.5780	0.5837
	SMFH [21]	0.2276	0.2516	0.2581	0.2496	0.5688	0.5917	0.5953	0.5961	0.5660	0.5938	0.6325	0.6175
	SCM-seq [22]	0.2210	0.2337	0.2442	0.2596	0.6237	0.6343	0.6448	0.6489	0.4842	0.4941	0.4947	0.4965
	SePH [23]	0.2787	0.2956	0.3064	0.3134	0.6723	0.6771	0.6783	0.6817	0.5421	0.5499	0.5537	0.5601
	DCH [20]	0.3317	**0.3686**	**0.3762**	**0.3748**	0.6589	0.6801	0.6970	0.6993	0.5789	0.5985	0.5886	0.5775
	GSPH [39]	0.2900	0.3100	0.3200	0.3260	0.6710	0.6830	0.6910	0.6950	0.6170	0.6300	0.6420	0.6480
	AAH	**0.3337**	0.3498	0.3535	0.3578	**0.7145**	**0.7230**	**0.7271**	**0.7283**	**0.6409**	**0.6439**	**0.6515**	**0.6549**
Text query Image	LSSH [38]	0.5571	0.5743	0.5710	0.5577	0.5784	0.5804	0.5797	0.5816	0.6250	0.6578	0.6823	0.6913
	CMFH [13]	0.6116	0.6298	0.6398	0.6477	0.6174	0.6241	0.6311	0.634	0.6641	0.6921	0.7164	0.7185
	SMFH [21]	0.5242	0.6039	0.6602	0.6658	0.5586	0.5727	0.5841	0.5828	0.5787	0.5462	0.6633	0.6247
	SCM-seq [22]	0.2134	0.2366	0.2479	0.2573	0.6133	0.6209	0.6295	0.634	0.4536	0.4620	0.4630	0.4644
	SePH [23]	0.6318	0.6577	0.6646	0.6709	0.7197	0.7271	0.7309	0.7354	0.6302	0.6425	0.6506	0.6580
	DCH [20]	0.7006	0.7087	0.7241	0.7093	0.7381	0.7782	0.7943	0.8127	0.7140	0.7303	0.7162	0.6914
	GSPH [39]	0.6460	0.6680	0.6770	0.6810	0.7210	0.7380	0.7490	0.7530	**0.7620**	**0.7780**	**0.7809**	**0.7980**
	AAH	**0.7102**	**0.7373**	**0.7413**	**0.7457**	**0.8137**	**0.8198**	**0.8251**	**0.8281**	0.7397	0.7553	0.7595	0.7629

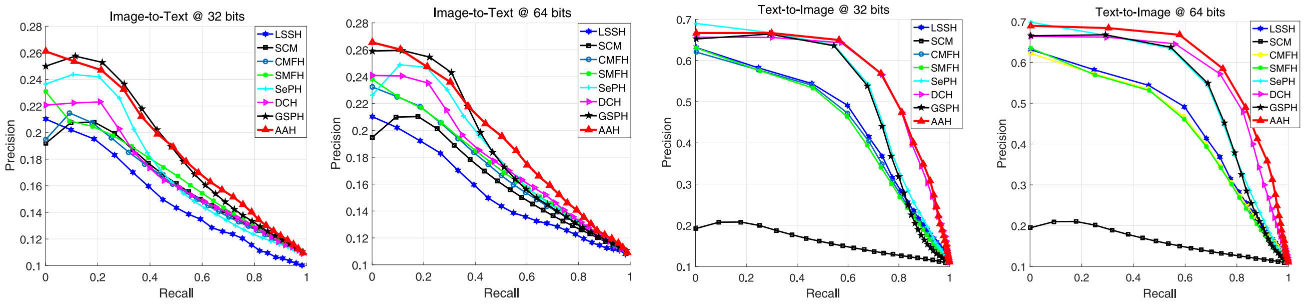

Fig. 2. Precision–Recall curves on the Wiki dataset by varying code length.

quite lower than those of the text query image for all the methods. There are possible reasons for the above observations. First, there is a quite large semantic gap between the two modalities of Wiki and images are quite different from their texts [38]. Therefore, it is more difficult to retrieve relevant text and images between different modalities, resulting in poor search accuracy for the image–query–text tasks. Second, due to the Wiki dataset has only one tag per sample, some images are not closely related to their assigned tags. In such cases, the extracted features of the image cannot reflect the semantic properties appropriately. Because DCH directly decomposes label and heterogeneous data into a common semantic space, which allows them to relate with each other when the label information is weak. Therefore, DCH obtains better mAP values than other methods for the image–query–text tasks in the Wiki dataset.

For the Precision–Recall curves and TopK–Precision curves shown in Figs. 2 and 3, we can see that our AAH is competitive with GSPH, SePH, and DCH. In general, the proposed AAH achieves satisfactory performance with different code lengths under different evaluation criteria on the Wiki dataset.

F. Results on MIR-Flickr

The mAP values of AAH and seven baseline methods on the MIR-Flickr datasets are shown in Table II. Apparently, AAH obtains a substantial improvement over the other state-of-the-art methods. In addition, we can observe that the mAP values of all methods improve with the increase of the code length. This is clear that longer hash codes can preserve more semantic feature. Moreover, we observed that AAH can preserve semantic information stably even with fewer bits. The Precision–Recall curves and TopK–Precision curves of all baseline methods on the MIR-Flickr dataset are plotted in Figs. 4 and 5, respectively. We can observe that AAH outperforms all baseline methods with different code lengths.

We clearly noticed that the results of all the baseline methods can produce better results on the MIR-Flickr dataset while achieving poor performance on the Wiki dataset, including our AAH. One possible reason for this phenomenon is that the images and texts of MIR-Flickr are quite semantic related to each other, and multiple tags make image–text pairs more compatible with labels.

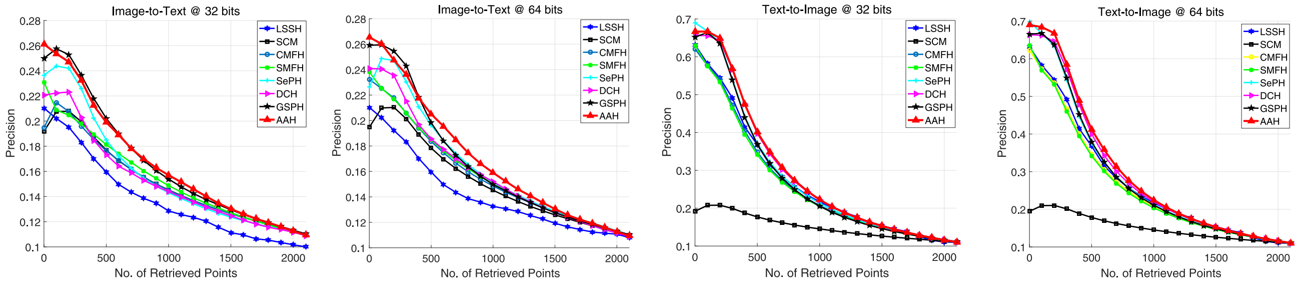

Fig. 3. TopK–Precision curves on the Wiki dataset by varying code length.

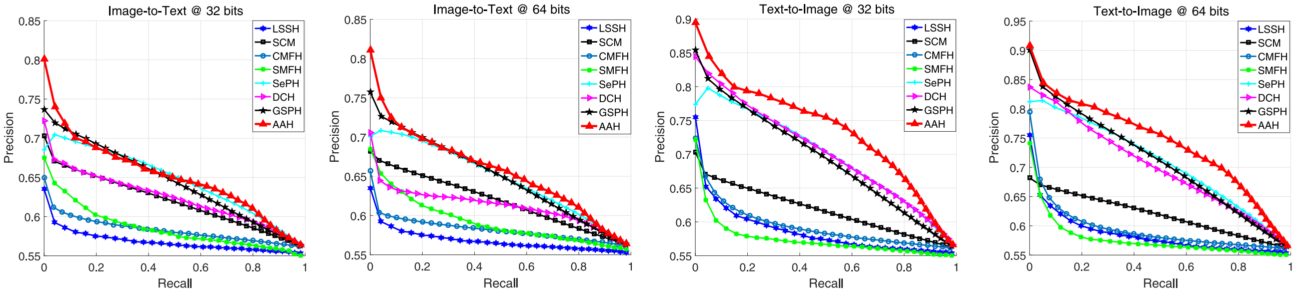

Fig. 4. Precision–Recall curves on MIR-Flickr by varying code length.

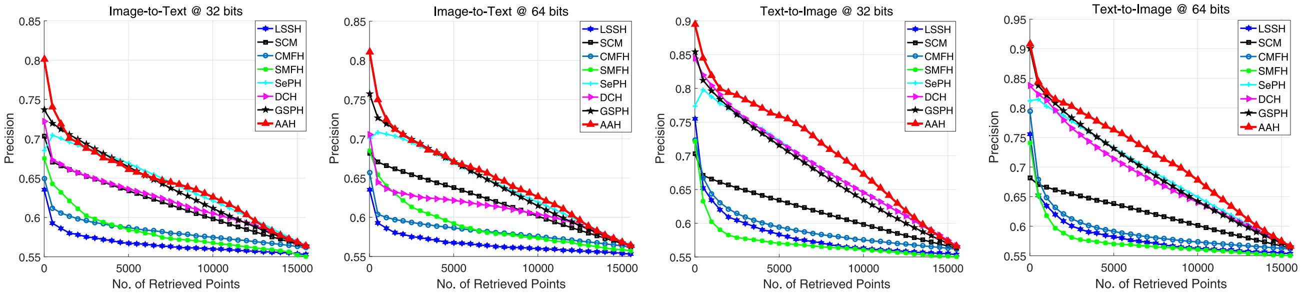

Fig. 5. TopK–Precision curves on MIR-Flickr by varying code length.

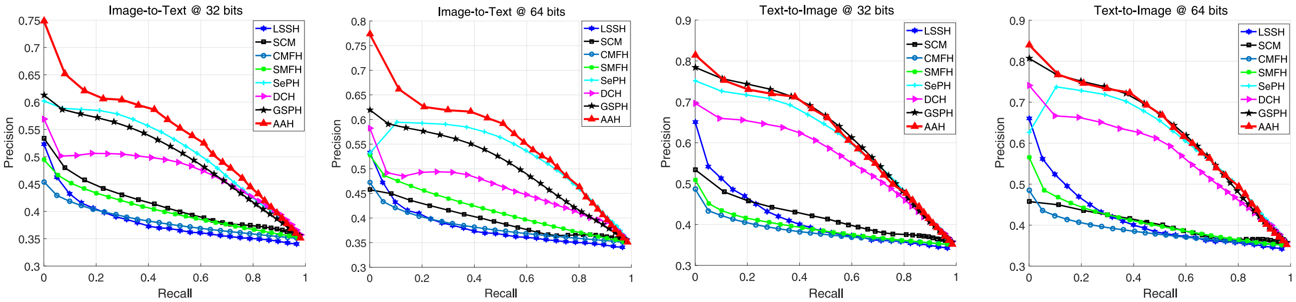

Fig. 6. Precision–Recall curves on NUS-WIDE by varying code length.

G. *Results on NUS-WIDE*

The mAP values for AAH and seven baseline methods on the NUS-WIDE dataset are reported in Table II. The TopK–Precision curves and Precision–Recall curves are plotted in Figs. 6 and 7, respectively. On the one hand, according to the experimental results, AAH is slightly lower than GSPH when using texts to query for images. This is because GSPH utilizes a semantic similar matrix to learn hash codes of two modalities simultaneously. Thus, the similarities between different modalities can be measured by the semantic similar matrix easily, that the original high-dimensional and discrete text features in the NUS-WIDE dataset can be embedded into the hash codes well. On the other hand, the results of the image–query–text show that AAH is better than all of the other methods. The reason may be that the image features in NUS-WIDE are complex, the pairwise similarity information of intramodal is hard to be measured and preserved for generating hash codes. AAH makes use of the labels similarity information

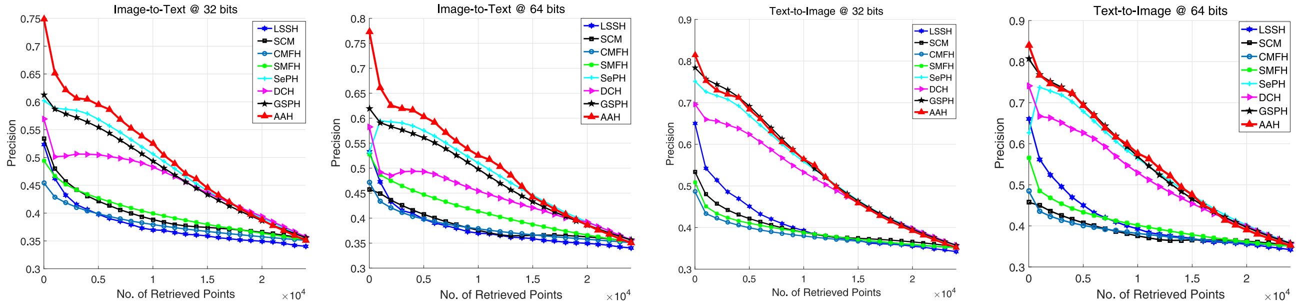

Fig. 7. TopK–Precision curves on NUS-WIDE by varying code length.

of intramodal and intermodal to preserve the discriminative information of each category, which guarantees that hash codes can both preserve the intermodal and intramodal information, so that the hash codes of samples with the same labels have similar representations. However, GSPH only extracts the intermodal information for cross-modal retrieval but ignores the intramodal information. Therefore, the hash codes learned by AAH have more discriminative ability to perform better in the image–query–text tasks than GSPH in the NUS-WIDE dataset.

H. Parameters Sensitivity Analysis

In the experiments, θ was first fixed in advance and an attempt was made to find a candidate interval where the optimal parameters α and β may exist. Then, by fixing the value of α and β in the candidate interval, the candidate interval of θ was determined. Finally, the optimal parameters in the 3-D candidate space of (θ, α, and β) with a fixed step length were searched. Figs. 8–10 show that mAP versus the value of different parameters on different datasets in which we adjusted the relevant parameter by fixing other parameters. For example, Fig. 8 shows that mAP versus the value of θ when we adjusted θ by fixing α and β (we set the length of the hash code to 64 bits and all mAP values are calculated by means of ten experimental results).

θ is the parameter that balances the influence of each modality in (5). When θ is larger than 1, texts can be more influential on the performance of AAH than images do, and vice versa. The results of using different θ are shown in Fig. 8 which illustrates that AAH achieves better performance when θ in the range of 1–10. When θ is in the range of other values, the mAP value fluctuates obviously, especially in the NUS-WIDE dataset. It can be inferred that texts have a greater impact on the performance of AAH than images overall. One possible reason is that the dimension of the text is lower than that of the image, so the features of the text are easier to be extracted than those in the image. Another possible reason is that the distribution of the original feature of the texts is more regular than images and, thus, the properties of their features are easier to be preserved, which can be seen in the t-SNE visualization section. In general, θ can be chosen in the range of [1:10].

α controls the influences of the label consistency in the approximation function in (6). For AAH, the semantic spaces of different modalities will approximate each other with the increase of α. It can be found from Fig. 9 that as α increases, the mAP values are stable on the whole, but it still fluctuates a little. This may be that the label consistency is also utilized to preserve the intramodal information in (6), which may weaken the influence of α. However, α is still significant for AAH, we will discuss it in the component analysis section again. The experimental results in Fig. 9 indicate that α works best when it is in the range of [1:10].

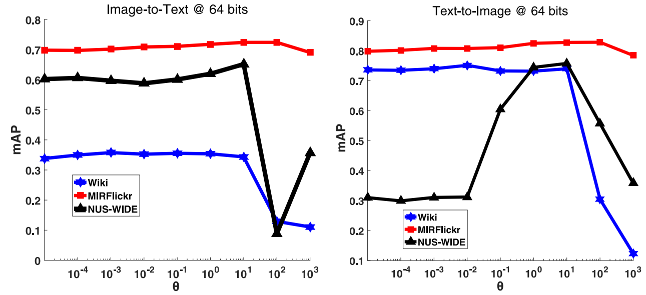

Fig. 8. mAP versus parameter θ.

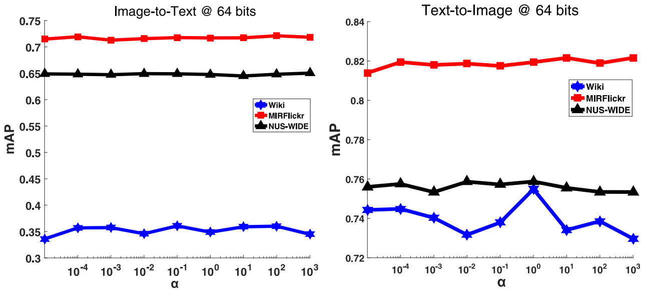

Fig. 9. mAP versus parameter α.

β controls the influences of the global approximation function in (6). We adjusted β to make semantic spaces of different modalities approximate to each other globally. It can be observed from Fig. 10 that AAH can achieve better performances when β is greater than 1. The learned hash codes in (7) cannot preserve the information of different semantic spaces very well when β is less than 1. Therefore, β can be chosen from the range between [1, 1000].

In general, the above results demonstrate that AAH is sufficiently robust against the parameters.

I. Running Time Comparison

Table III shows the training time of different methods on NUS-WIDE by varying the size of the training set from 1500

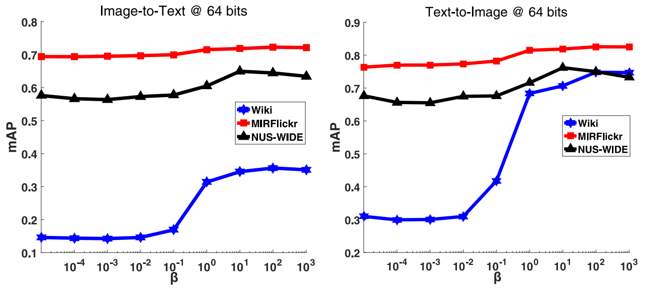

Fig. 10. mAP versus parameter β.

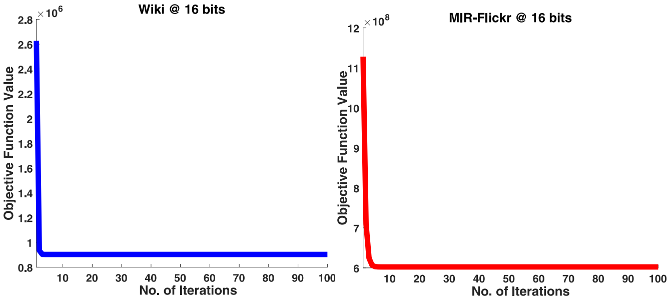

Fig. 11. Convergence behavior analysis of AAH.

TABLE III
TRAINING TIME (IN SECONDS) ON THE NUS-WIDE DATASET BY VARYING THE SIZE OF TRAINING SET

Methods\Size of data set	1500	2000	3000	5000	8000	10000
CMFH	40.6	32.665	49.704	85.413	148.857	198.872
SMFH	1.343	1.632	2.305	3.573	5.376	6.682
SCM	53.635	53.178	55.042	56.587	57.932	58.715
SePH	83.515	116.727	213.332	527.375	1268.607	1971.884
DCH	1.247	1.591	2.243	3.37	5.237	6.287
GSPH	27.893	33.079	38.161	63.667	98.198	131.136
AAH	5.354	7.636	10.877	22.561	46.754	68.346

TABLE IV
INDISPENSABILITY OF EACH COMPONENT IN THE PROPOSED AAH

Task	Methods\Data set	Wiki	MIR-Flickr	NUS-WIDE
Image query Text	AAH_knn	0.2181	0.5446	0.3449
	AAH_nl	0.3255	0.7118	0.6275
	AAH_ng	0.3272	0.5764	0.5734
	AAH_na	0.2453	0.6482	0.5431
	AAH	**0.3337**	**0.7145**	**0.6409**
Text query Image	AAH_knn	0.4646	0.5471	0.3432
	AAH_nl	0.7003	0.8100	0.7237
	AAH_ng	0.6750	0.5851	0.6551
	AAH_na	0.5650	0.7191	0.6402
	AAH	**0.7102**	**0.8137**	**0.7397**

to 10 000. The code length is also fixed to 16 bits in this experiment. As shown in Table III, SePH requires a large amount of time for training. During these methods, SMFH and DCH are the fastest ones than others and the reason may be that they both directly utilize collective matrix factorization to obtain the hash codes. As the size of the dataset increases, our AAH will spend more time on the construction of a similar semantic matrix and optimization algorithm. However, the proposed AAH still achieves a much better performance than the others, such as CMFH, GSPH, and SePH. Therefore, AAH possesses a competitive computational speed as well as better performance compared with existing multimodal hashing methods.

J. Component Analysis

To evaluate the indispensability of each component in the proposed AAH, we gave four variants of the proposed algorithm for comparison. In this section, we evaluated their performance in the case of 16 bits length.

We first replaced label with k nearest neighbors to reconstruct similarly graph construction, so we redefined $S_{ij} = W_{ij}$ in (8) where $W_{ij} = 1$ if $x_i \in N_k(x_j)$ and $x_j \in N_k(x_i)$, or $W_{ij} = 0$ otherwise, $N(\cdot)$ denotes the set of k nearest neighbors. Here, we called it AAH_knn in Table IV. From Table IV, we can observe that the mAP results of AAH are much better than AAH_knn. This is because the corresponding information between samples obtained by using labels is more accurate than that obtained by using k nearest neighbors.

Second, we set parameter $\alpha = 0$ to remove the influence of label consistency approximate function, that is, the third term in (8), we marked it as AAH_nl. As mentioned in the parameter sensitivity analysis section, although α has a relatively small impact on the performance of AAH, it is also indispensable. We can observe that the performance of AAH is slightly better than that of AAH_nl in Table IV, which also proves that the label consistency approximate function can improve the overall results of AAH.

Third, we set the parameter $\beta = 0$ to remove the influence of the global approximation function in AAH, that is, the fourth term in (8), we called it AAH_ng. From the comparison of Table IV, it can be observed that the global approximation function has a great influence on the results of AAH. Therefore, lacking global constraints on the approximation of different semantic spaces, the unified hash code obtained from different semantic spaces will cause deviation to the retrieval results. Thus, the global approximation function is indispensable for AAH.

The last one we removed the influence of average approximate function in AAH, that is, the last term of (8), we marked it as AAH_na. It is clear from Table IV that the average approximate function produces a positive effect in AAH. Since the unified hash codes obtained from the average approximate function preserve the information of original data from each modality, which greatly improves the effectiveness of retrieval tasks. Therefore, the experimental results produced by AAH are better than those produced by AAH_na.

TABLE V
COMPARISON WITH DEEP HASHING METHODS. THE BEST PERFORMANCE IS SHOWN IN BOLDFACE

method	Dataset	MIR-Flickr			NUS-WIDE		
	Task	16bits	32bits	64bits	16bits	32bits	64bits
DCMH	Image query Text	0.7756	0.7813	0.7774	0.6153	0.6205	0.6223
	Text query Image	0.7969	0.8014	0.7911	0.6819	0.6901	0.6910
	Time(in seconds)	21569.36	23751.63	25025.88	23178.25	23517.29	24508.56
SSAH	Image query Text	**0.7800**	**0.7820**	**0.7950**	**0.6310**	0.6250	0.6390
	Text query Image	0.7830	0.7930	0.8050	0.6690	0.6630	0.6710
	Time(in seconds)	441620.5	414740.7	437002.2	406951.9	422341.5	426584.5
AAH	Image query Text	0.7048	0.7148	0.7204	0.6266	**0.6365**	**0.6399**
	Text query Image	**0.8051**	**0.8120**	**0.8168**	**0.7154**	**0.7335**	**0.7422**
	Time(in seconds)	**41.57**	**43.228**	**47.12**	**71.968**	**73.418**	**78.492**

Notably, we can observe the mAP results of AAH are better than AAH_knn, AAH_nl, AAH_ng, and AAH_na, which verifies the effectiveness of the proposed strategy.

K. Convergence Study

Because AAH is solved by iterative update rules in Algorithm 1, in this section, we evaluated its convergence property empirically by using 16 bits codes on Wiki and MIR-Flickr datasets. In Section IV, we have given an iterative updating rule which was proven to be convergent in solving (8). Here, we showed the convergence property of AAH in Fig. 11 which includes the convergence curves on the Wiki and MIR-Flickr datasets. We can observe that the update rules of AAH converge very quickly on both datasets within ten iterations. Furthermore, the average time cost for each iteration on Wiki and MIR-Flick25k is 0.320 and 7.65 s, respectively, which validates the high efficiency of AAH on large-scale datasets.

L. Comparison With Deep Hashing

Recently, deep hashing methods have achieved cracking performance. In this section, we conducted this experiment to compare our proposed AAH with two state-of-the-art methods, that is: 1) deep cross-modal hashing (DCMH) [40] and 2) self-supervised adversarial hashing networks for cross-modal retrieval (SSAH) [41]. Following the parameters setting of DCMH and SSAH, we set the optimal parameters for them to obtain the best performance. In addition, we also selected the size of the training set and the query set of our method as the same as that of DCMH and SSAH. Specifically, the size of the training set and query set is 10 000 and 2000 for the MIR-Flickr dataset, respectively. Table V shows the experimental results of mAP and time consumption of these methods. It is observed from Table V that our proposed AAH can obtain better mAP results with less time consumption. Therefore, the proposed AAH method can be more adaptable for the large-scale data dataset with less training time.

It is worth noting that our proposed AAH is not an end-to-end deep model. However, it can still achieve competitive results. The main reason is that AAH can use the PCA-like item to preserve the main energy of data and thus the obtained hash code can devotedly preserve the main information of data.

VI. CONCLUSION

In this article, we proposed an effective cross-modal hashing method called AAH. This method utilizes a PCA-like projection matrix to preserve the main energy and reconstruction residual of data from different modalities and then the projected data are embedded into different semantic spaces. We utilized the global approximation to make the two semantic spaces approximate to each other and the label consistency to make the samples from different modalities but with the same label close together. By averaging two semantic spaces, the obtained hash codes can preserve the properties of different modalities. To the best of our knowledge, AAH is the first work to use the PCA-like item and average approximation strategy for addressing the problem of cross-modal retrieval. Extensive experiments on three benchmark datasets demonstrate that AAH outperforms several state-of-the-art methods.

REFERENCES

[1] M. Datar, N. Immorlica, P. Indyk, and V. S. Mirrokni, "Locality-sensitive hashing scheme based on *p*-stable distributions," in *Proc. 20th ACM Symp. Comput. Geometry*, Brooklyn, NY, USA, Jun. 2004, pp. 253–262.

[2] B. Kulis and K. Grauman, "Kernelized locality-sensitive hashing for scalable image search," in *Proc. IEEE 12th Int. Conf. Comput. Vis. (ICCV)*, Kyoto, Japan, Sep./Oct. 2009, pp. 2130–2137.

[3] M. Raginsky and S. Lazebnik, "Locality-sensitive binary codes from shift-invariant kernels," in *Proc. Adv. Neural Inf. Process. Syst. 23rd Annu. Conf. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, Dec. 2009, pp. 1509–1517.

[4] X. Nie, W. Jing, C. Cui, C. J. Zhang, L. Zhu, and Y. Yin, "Joint multi-view hashing for large-scale near-duplicate video retrieval," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 10, pp. 1951–1965, Oct. 2020.

[5] S. He *et al.*, "Bidirectional discrete matrix factorization hashing for image search," *IEEE Trans. Cybern.*, vol. 50, no. 9, pp. 4157–4168, Sep. 2020.

[6] G. Irie, H. Arai, and Y. Taniguchi, "Alternating co-quantization for cross-modal hashing," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, Chile, Dec. 2015, pp. 1886–1894.

[7] B. Wu, Q. Yang, W. Zheng, Y. Wang, and J. Wang, "Quantized correlation hashing for fast cross-modal search," in *Proc. 24th Int. Joint Conf. Artif. Intell. (IJCAI)*, Buenos Aires, Argentina, Jul. 2015, pp. 3946–3952.

[8] Y. Wang, X. Lin, L. Wu, W. Zhang, and Q. Zhang, "LBMCH: Learning bridging mapping for cross-modal hashing," in *Proc. 38th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, Santiago, Chile, Aug. 2015, pp. 999–1002.

[9] X. Luo, X. Yin, L. Nie, X. Song, Y. Wang, and X. Xu, "SDMCH: Supervised discrete manifold-embedded cross-modal hashing," in *Proc. 2nd Int. Joint Conf. Artif. Intell. (IJCAI)*, Stockholm, Sweden, Jul. 2018, pp. 2518–2524.

[10] L. Liu, Z. Lin, L. Shao, F. Shen, G. Ding, and J. Han, "Sequential discrete hashing for scalable cross-modality similarity retrieval," *IEEE Trans. Image Process.*, vol. 26, no. 1, pp. 107–118, Jan. 2017.

[11] T. Yao, X. Kong, H. Fu, and Q. Tian, "Discrete semantic alignment hashing for cross-media retrieval," *IEEE Trans. Cybern.*, vol. 50, no. 12, pp. 4896–4907, Dec. 2020.

[12] Y. Peng, X. Huang, and Y. Zhao, "An overview of cross-media retrieval: Concepts, methodologies, benchmarks, and challenges," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 9, pp. 2372–2385, Sep. 2018.

[13] G. Ding, Y. Guo, and J. Zhou, "Collective matrix factorization hashing for multimodal data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Columbus, OH, USA, Jun. 2014, pp. 2083–2090.

[14] X. Liu, A. Li, J. Du, S. Peng, and W. Fan, "Efficient cross-modal retrieval via flexible supervised collective matrix factorization hashing," *Multimedia Tools Appl.*, vol. 77, no. 21, pp. 28665–28683, 2018.

[15] J. Tang, K. Wang, and L. Shao, "Supervised matrix factorization hashing for cross-modal retrieval," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3157–3166, Jul. 2016.

[16] D. Wang, X. Gao, X. Wang, and L. He, "Label consistent matrix factorization hashing for large-scale cross-modal similarity search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 10, pp. 2466–2479, Oct. 2019.

[17] D. Mandal and S. Biswas, "Label consistent matrix factorization based hashing for cross-modal retrieval," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Beijing, China, Sep. 2017, pp. 2901–2905.

[18] Y. Cui, J. Jiang, Z. Lai, Z. Hu, and W. K. Wong, "Supervised discrete discriminant hashing for image retrieval," *Pattern Recognit.*, vol. 78, pp. 79–90, Jun. 2018.

[19] X. Nie, X. Liu, X. Xi, C. Li, and Y. Yin, "Fast unmediated hashing for cross-modal retrieval," *IEEE Trans. Circuits Syst. Video Technol.*, early access, Dec. 7, 2020, doi: 10.1109/TCSVT.2020.3042972.

[20] X. Xu, F. Shen, Y. Yang, H. T. Shen, and X. Li, "Learning discriminative binary codes for large-scale cross-modal retrieval," *IEEE Trans. Image Process.*, vol. 26, no. 5, pp. 2494–2507, May 2017.

[21] H. Liu, R. Ji, Y. Wu, and G. Hua, "Supervised matrix factorization for cross-modality hashing," in *Proc. 25th Int. Joint Conf. Artif. Intell. (IJCAI)*, New York, NY, USA, Jul. 2016, pp. 1767–1773.

[22] D. Zhang and W. Li, "Large-scale supervised multimodal hashing with semantic correlation maximization," in *Proc. 28th AAAI Conf. Artif. Intell.*, Jul. 2014, pp. 2177–2183.

[23] Z. Lin, G. Ding, M. Hu, and J. Wang, "Semantics-preserving hashing for cross-view retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 3864–3872.

[24] H. Zou, T. Hastie, and R. Tibshirani, "Sparse principal component analysis," *J. Comput. Graph. Stat.*, vol. 15, no. 2, pp. 265–286, 2006.

[25] J. Franklin, "The elements of statistical learning: Data mining, inference and prediction," *Publ. Amer. Stat. Assoc.*, vol. 99, no. 466, p. 567, 2010.

[26] S. Thrun, L. K. Saul, and B. Schölkopf, Eds., *Advances in Neural Information Processing Systems 16 [Neural Information Processing Systems, NIPS 2003, December 8-13, 2003, Vancouver and Whistler, British Columbia, Canada]*. Cambridge, MA, USA: MIT Press, 2004.

[27] G. F. Lu, Y. Wang, J. Zou, and Z. Wang, "Matrix exponential based discriminant locality preserving projections for feature extraction," *Neural Netw*, vol. 97, pp. 127–136, Jan. 2018.

[28] Y. Zhen, Y. Gao, D. Yeung, H. Zha, and X. Li, "Spectral multimodal hashing and its application to multimedia retrieval," *IEEE Trans. Cybern.*, vol. 46, no. 1, pp. 27–38, Jan. 2016.

[29] B. Wu and Y. Wang, "Neighborhood-preserving hashing for large-scale cross-modal search," in *Proc. ACM Conf. Multimedia (MM)*, Amsterdam, The Netherlands, Oct. 2016, pp. 352–356.

[30] Z. Miao *et al.*, "Graph regularized sparse coding for image representation," *IEEE Trans. Image Process.*, vol. 20, no. 5, pp. 1327–1336, May 2011.

[31] X. Liu, B. Du, C. Deng, M. Liu, and B. Lang, "Structure sensitive hashing with adaptive product quantization," *IEEE Trans. Cybern.*, vol. 46, no. 10, pp. 2252–2264, Oct. 2016.

[32] S. P. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.

[33] J. Wen, N. Han, X. Fang, L. Fei, K. Yan, and S. Zhan, "Low-rank preserving projection via graph regularized reconstruction," *IEEE Trans. Cybern.*, vol. 49, no. 4, pp. 1279–1291, Apr. 2019.

[34] D. Wang, X. Gao, X. Wang, and L. He, "Semantic topic multimodal hashing for cross-media retrieval," in *Proc. 24th Int. Joint Conf. Artif. Intell. (IJCAI)*, Buenos Aires, Argentina, Jul. 2015, pp. 3890–3896.

[35] X. Liu, Z. Hu, H. Ling, and Y. Cheung, "MTFH: A matrix tri-factorization hashing framework for efficient cross-modal retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 3, pp. 964–981, Mar. 2021.

[36] Z. Lin, G. Ding, J. Han, and J. Wang, "Cross-view retrieval via probability-based semantics-preserving hashing," *IEEE Trans. Cybern.*, vol. 47, no. 12, pp. 4342–4355, Dec. 2017.

[37] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, San Diego, CA, USA, May 2015, pp. 1–6.

[38] J. Zhou, G. Ding, and Y. Guo, "Latent semantic sparse hashing for cross-modal similarity search," in *Proc. 37th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval (SIGIR)*, Gold Coast, QLD, Australia, Jul. 2014, pp. 415–424.

[39] D. Mandal, K. N. Chaudhury, and S. Biswas, "Generalized semantic preserving hashing for cross-modal retrieval," *IEEE Trans. Image Process.*, vol. 28, no. 1, pp. 102–112, Jan. 2019.

[40] Q. Jiang and W. Li, "Deep cross-modal hashing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 3270–3278.

[41] C. Li, C. Deng, N. Li, W. Liu, X. Gao, and D. Tao, "Self-supervised adversarial hashing networks for cross-modal retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Salt Lake City, UT, USA, Jun. 2018, pp. 4242–4251.