Adaptive Marginalized Semantic Hashing for Unpaired Cross-Modal Retrieval

Kaiyi Luo ⊙, Chao Zhang ⊙, Graduate Student Member, IEEE, Huaxiong Li ⊙, Senior Member, IEEE, Xiuyi Jia ⊙, Member, IEEE, and Chunlin Chen ⊙, Senior Member, IEEE

Abstract—In recent years, Cross-Modal Hashing (CMH) has attracted much attention due to its fast query speed and efficient storage. Previous studies have achieved promising results for Cross-Modal Retrieval (CMR) by discovering discriminative hash codes and modality-specific hash functions. Nonetheless, most existing CMR works are subjected to some restrictions: 1) It is assumed that data of different modalities are fully paired, which is impractical in real applications due to sample missing and false data alignment, and 2) binary regression targets including the label matrix and binary codes are too rigid to effectively learn semantic-preserving hash codes and hash functions. To address these problems, this paper proposes an Adaptive Marginalized Semantic Hashing (AMSH) method which not only enhances the discrimination of latent representations and hash codes by adaptive margins, but can also be used for both paired and unpaired CMR. As a two-step method, in the first step, AMSH generates semantic-aware modality-specific latent representations with adaptively marginalized labels, thereby enlarging the distances between different classes, and exploiting the labels to preserve the inter-modal and intra-modal semantic similarities into latent representations and hash codes. In the second step, adaptive margin matrices are embedded into the hash codes, and enlarge the gaps between positive and negative bits, which improves the discrimination and robustness of hash functions. On this basis, AMSH generates similarity-preserving hash codes and robust hash functions without the strict one-to-one data correspondence requirement. Experiments are conducted on several benchmark datasets to demonstrate the superiority and flexibility of AMSH over some state-of-the-art CMR methods.

Index Terms—Cross-modal retrieval, unpaired hashing, adaptive margins.

Manuscript received 24 July 2022; revised 11 October 2022 and 29 January 2023; accepted 11 February 2023. Date of publication 15 February 2023; date of current version 15 December 2023. This work was supported in part by the National Natural Science Foundation of China under Grants 62176116, 62176123, 62073160, and 62276136, in part by the Natural Science Foundation of the Jiangsu Higher Education Institutions of China under Grant 20KJA520006, and in part by the National Key Research and Development Program of China under Grant 2018YFB1404200. The Associate Editor coordinating the review of this manuscript and approving it for publication was Dr. Palaiahnakoti Shivakumar. (Corresponding author: Huaxiong Li.)

Kaiyi Luo, Chao Zhang, Huaxiong Li, and Chunlin Chen are with the Department of Control Science and Intelligence Engineering, Nanjing University, Nanjing 210093, China (e-mail: ky_luo@smail.nju.edu.cn; chzhang@smail.nju.edu.cn; huaxiongli@nju.edu.cn; clchen@nju.edu.cn).

Xiuyi Jia is with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210004, China (e-mail: jiaxy@njust.edu.cn).

The source code is available at https://github.com/LKYLKYZ/AMSH.

Digital Object Identifier 10.1109/TMM.2023.3245400

I. INTRODUCTION

The recent years have witnessed the explosive growth of multimedia such as texts and images. Cross-Modal Retrieval (CMR) has attracted much interest as a method to search relevant items of different modalities [1], such as searching desired images with text descriptions. Unlike uni-modal retrieval, which searches within one modality, the biggest challenge of CMR is to bridge the heterogeneous gaps among modalities, which is caused by the different feature distributions and dimensionality across modalities [2], [3], [4]. Some CMR methods [5], [6], [7], [8] were proposed to eliminate the heterogeneity by learning a common real-valued representation for all modalities. Despite the effectiveness of these methods, query and training efficiency is compromised, and much memory is required for data storage. Thus, how to efficiently and accurately search semantically similar items across modalities remains a challenge [9], [10].

The hashing technique has been widely applied in CMR due to its fast query speed and efficient storage. Many Cross-Modal Hashing (CMH) methods have been proposed in recent years [11], [12], [13], [14], [15], [16], [17]. The basic idea of hashing methods is to map data of different modalities into a common Hamming space, where each sample can be represented by binary codes. In this way, distances between data points can be measured by XOR operation, which requires low computational costs. CMH methods can be roughly categorized into supervised methods [2], [9], [18], [19], [20], [21] and unsupervised ones [22], [23], [24], [25]. Unsupervised CMH methods seek to generate robust hash codes and hash functions without label information by exploiting data structures. Typical works include Collective Matrix Factorization Hashing (CMFH) [22], Fusion Similarity Hashing (FSH) [24] and Collective Reconstructive Embeddings (CRE) [25]. Supervised CMH methods utilize labels to guide the hash learning, with inter-modal and intra-modal similarities well preserved. Due to the heterogeneous gaps among different modalities, a common strategy of supervised CMH methods is to seek a shared latent representation, with data across modalities fully paired. To preserve semantic similarities, some methods [2], [3] train a linear classifier between labels and hash codes for semantic embedding. Typical works include Supervised Matrix Factorization Hashing (SMFH) [9], Discrete Cross-modal Hashing (DCH) [19], Scalable disCReTe mATrix faCtorization Hashing (SCRATCH) [2] and Subspace Relation Learning for Cross-modal Hashing (SRLCH) [20]. Generally,
supervised methods outperform unsupervised ones, since labels are high-level semantic information, which can better represent data similarities and correlations [3].

Although most supervised CMH methods have achieved promising retrieval results, there are still some problems to be solved:

1) One crucial problem is the strict inter-modal data alignment assumption. Most supervised CMH methods assume that data of all modalities are fully paired, which facilitates the elimination of heterogeneous gaps by projecting multi-modal data into a shared latent representation. However, in real practice, collected data may suffer from data missing and misalignment, making the one-to-one correspondence assumption invalid. This scenario is referred to as Unpaired Cross-Modal Retrieval (UCMR). Recently, some works have been proposed to address Semi-paired Cross-Modal Retrieval [10], [26], [27], [28], where only partial data correspondence is given. However, semi-paired hashing methods cannot solve the UCMR problem due to their reliance on paired data. To the best of our knowledge, few studies are dedicated to the UCMR problem [29], [30], [31], [32].

2) Some supervised CMH methods train a classifier between latent features and labels to preserve high-level semantic information based on the original logical label matrix, which ignores the inter-class discrimination. In the multi-class classification, instances of different classes should be as far from each other as possible after they are transformed, to increase data separability and discrimination. Besides, the labels are strictly discrete, while latent features are continuous, which hinders effective semantic embedding.

3) To obtain modality-specific hash functions, most previous methods directly adopt least squares regression between multi-modal features and binary codes, thereby compromising the robustness of hash functions, because the binary codes are too rigid for robust and discriminative regression.

To tackle these limitations, we propose an Adaptive Marginalized Semantic Hashing (AMSH) method for CMR with unpaired data. The overall framework of AMSH is illustrated in Fig. 1. AMSH is a two-step method. In the first step, to tackle the UCMR problem, AMSH generates modality-specific latent representations, which are embedded with semantic information by preserving inter-modal and intra-modal similarities. Besides, AMSH introduces the adaptive margin matrices into the linear classifier between labels and latent representations to alleviate the rigid zero-one linear regression, which enhances the discrimination of latent representations and hash codes. In the second step, to address the rigid binary linear regression problem, AMSH incorporates the adaptive margin matrices to adaptively enlarge the distances between positive and negative bits. Thus, discriminative semantic-aware hash codes and robust hash functions can be obtained. The main contributions are summarized as follows:

- We present a novel supervised method, i.e., AMSH, which can not only learn discriminative hash codes and hash functions, but also solve unpaired CMR problems.
- We introduce the adaptive margin matrices which are utilized to enlarge the distances between different classes, and the distances between positive and negative hash code bits, to enhance the discrimination of hash codes and hash functions.
- Compared with some state-of-the-art CMH methods, extensive experiments are conducted on several benchmark datasets, demonstrating the effectiveness of the proposed AMSH.

The remainder of this paper is organized as follows. Section II briefly reviews some existing related works. Section III presents the formulation of AMSH and the optimization scheme. Experiments are conducted in Section IV. Section V concludes the paper.
II. RELATED WORK

A. Supervised Cross-Modal Hashing

Supervised CMH methods bridge the semantic gaps among different modalities by embedding label information into hash codes, which generally outperform unsupervised ones. Semantic Correlation Maximization (SCM) hashing [33] constructs the similarity matrix with label information, and further applies spectral relaxation to solve the binary NP-hard problem. Semantics-Preserving Hashing (SePH) method [34] constructs a probability distribution with semantic affinities, and approximates the distribution to the target binary codes. These methods relax the binary constraints into continuous ones. However, relaxation strategies may cause quantization error. Other discrete hashing methods have been proposed without relaxation. DCH [19] views binary codes as features to train classifiers, constructs modality, where $l$ is the feature dimensionality and Two-stEp exploits labels to pre-

$ \begin{align*} 
\text{min} & \quad \frac{1}{2} \| x - y \|_2^2 + \lambda \| \theta - \phi \|_2^2 \\
\text{subject to} & \quad \| x \|_2 = 1, \| y \|_2 = 1, \\
& \quad 0 \leq x_i \leq 1, 0 \leq y_i \leq 1, \\
& \quad 0 \leq \phi_i \leq 1, 0 \leq \theta_i \leq 1, \\
& \quad \eta \text{ is the code length.} 
\end{align*} $ 

is an $x \in \mathbb{R}$.

adaptively enlarges semantic distance of different classes to increase label discrimination. However, all these methods also explore data structures for better hash learning. Luo et al. proposed a Supervised Discrete Manifold-Embedded Cross-Modal Hashing (SDMCH) [37] which explores manifold correlations among data of different modalities. Adaptive Label correlation based asymmetric Cross-modal Hashing (ALECH) [3] and Two-stEp Cross-modal Hashing (TECH) [38] both consider label correlation for better semantic discrimination. Fast Cross-Modal Hashing (FCMH) [39] preserves not only global similarities but also local similarities. Fast Discriminative Discrete Hashing (FDDH) [40] adaptively enlarges semantic distance of different classes to increase label discrimination. However, all these methods cannot address the UCMR problem, for they are based on the assumption that data across modalities are fully paired, which is impractical in real applications.

Recently, with the development of deep learning, hashing methods have been greatly boosted by Deep Neural Networks (DNNs). Many DNN-based CMR methods have been proposed [41, 42, 43, 44]. These methods generally outperform shallow models due to the representation capacity of DNNs. Typical works include Deep Cross-Modal Hashing (DCMH) [41], Pairwise Relationship Guided Deep Hashing (PRDH) [42] and Attention-Aware Deep Adversarial Hashing (AADAH) [43]. Nevertheless, DNN-based methods suffer from high computational costs, and it is challenging to determine the network structures for different modalities. Besides, most DNN-based methods cannot be applied to the UCMR problem.

$ \begin{align*} 
& \text{min} \quad \frac{1}{2} \| x - y \|_2^2 + \lambda \| \theta - \phi \|_2^2 \\
& \text{subject to} \quad \| x \|_2 = 1, \| y \|_2 = 1, \\
& \quad 0 \leq x_i \leq 1, 0 \leq y_i \leq 1, \\
& \quad 0 \leq \phi_i \leq 1, 0 \leq \theta_i \leq 1, \\
& \quad \eta \text{ is the code length.} 
\end{align*} $ 

B. Unpaired Cross-Modal Hashing

Many CMH methods are based on the assumption that data of different modalities have one-to-one correspondence. However, in real applications, data missing and misalignment may occur, making fully paired data unobtainable. One realistic challenge is to address the UCMR problem, which is less explored by researchers. To our knowledge, there are several UCMR-oriented methods. Generalized Semantic Preserving Hashing (GSPH) [29] seeks to factorize the similarity matrix into the dot product of two binary matrices, thus embedding semantic information into the to-be-learned binary codes, and applies kernel logistic regression to obtain modality-specific hash functions. Unpaired Cross-Modal Hashing (UCMH) [30] utilizes matrix factorization to generate modality-specific latent representations. To preserve semantic similarities, UCMH constructs the Laplacian matrices with labels, which is time-consuming. Robust Unsupervised Cross-Modal Hashing (RUCMH) [31] considers the data reconstruction relations in the subspace to bridge the heterogeneous gaps between modalities, and adopts $\ell_2,1$ norm to learn hash functions, which can alleviate the model sensitivity to data noise. Nonetheless, RUCMH is an unsupervised method that ignores embedding the label information into the hash functions. Enhanced Discrete Multi-modal Hashing (EDMH) [45] and Flexible Multi-modal Hashing (FMH) [32] both adopt an asymmetric strategy to keep similarities. However, they do not consider label discrimination, with the distances between different classes the same. Besides, most of these methods directly adopt linear regression to obtain modality-specific hash functions without considering the rigid binary problem, which impairs the robustness of hash functions.

III. PROPOSED METHOD

A. Notations

In this paper, matrices are denoted by boldface uppercase letters, and vectors by boldface lowercase letters. For a matrix $M$, $M_{ij}$ is the entry of the $i^{th}$ row and $j^{th}$ column, and $M_{ik}$ is the $i^{th}$ column of $M$. Given a multimodal dataset $\mathcal{D} = \{X^{(1)}, X^{(2)}, \ldots, X^{(m)}\}$ with $m$ modalities, $X^{(i)} = [x_{1}^{(i)}, x_{2}^{(i)}, \ldots, x_{n}^{(i)}] \in \mathbb{R}^{d_i \times n_i}$ is the training data of the $i^{th}$ modality, where $d_i$ is the feature dimensionality and $n_i$ is the data size. We assume that the samples are zero-centered in each modality, i.e., $\sum_{j=1}^{n_i} x_{j}^{(i)} = 0, i = 1, \ldots, m$. $L^{(i)} = [l_{1}^{(i)}, l_{2}^{(i)}, \ldots, l_{m}^{(i)}] \in \{0, 1\}^{c \times n_i}$ represents the label matrix of the $i^{th}$ modality, where $c$ is the number of classes. $L_{jk}^{(i)} = 1$ if the $k^{th}$ sample from the $i^{th}$ modality is associated with the $j^{th}$ category and 0 otherwise. Hash code matrices are denoted by $B^{(i)} \in \{-1, 1\}^{r \times n_i}$, where $r$ is the code length. $F^{(i)}_{\alpha(i)}$ are the modality-specific hash functions that connect the original data space to the Hamming space. $\text{tr}()$ is the trace operation. $\text{sgn}(\cdot)$ is the sign function that returns 1 if the input is positive and -1 otherwise. $I_n$ is an $n \times n$ identity matrix, and $0_n$ is an $n$-dimensional column vector with all items being 0.

B. Hash Codes Learning

1) Adaptive Marginalized Regression: To eliminate the heterogeneous gap across modalities and leverage the semantic information, most previous methods learn a common latent representation, and construct a linear regression between the latent
feature space and the label space, which can be described as follows:

\[
\min_{\mathbf{P}, \mathbf{V}} \| \mathbf{L} - \mathbf{P} \mathbf{V} \|_F^2, \tag{1}
\]

where \( \mathbf{L} \in \{0, 1\}^{c \times n} \) is the label matrix, \( \mathbf{P} \in \mathbb{R}^{c \times r} \) is a projection matrix, and \( \mathbf{V} \in \mathbb{R}^{r \times n} \) is the shared latent representation, where \( n \) is the data size. \( \mathbf{L} \) is not explicitly computed, \( \mathbf{P} \) and \( \mathbf{V} \) are the modality-specific projection matrix and the latent representation, respectively, and \( m \) is the number of modalities. The above objective function aims to generate semantic-preserving latent representations for each modality by establishing linear transform relationships between the representation and the strict zero-one label matrix. However, the zero-one label matrix is too rigid to learn an effective projection matrix for fitting class labels, which may degrade the representation quality and limit the flexibility. Besides, from the perspective of robust classification, it is desired that instances of different classes should be as far from each other as possible after they are transformed, to increase data separability and discrimination, as shown in Fig. 2. Inspired by [46], [47], to alleviate this problem, we introduce adaptive margin factors to refine the regression targets so that the margins of different classes are enlarged.

For its purpose, we introduce an index matrix \( \mathbf{R}^{(i)} \in \mathbb{R}^{c \times n} \) for each modality that satisfies:

\[
P^{(i)}_{jk} = \begin{cases} +1, & \text{if } L_{jk}^{(i)} = 1 \\ -1, & \text{otherwise} \end{cases}, \tag{3}
\]

where ‘+1’ denotes the positive direction and ‘-1’ the negative direction. The margin matrix \( \mathbf{E}^{(i)} = \{ \varepsilon_{jk} \geq 0 \} \in \mathbb{R}^{c \times n} \) is used to perform the marginalization operation on every element of \( \mathbf{L}^{(i)} \). For example, \( \mathbf{I}^{(i)} = [0, 1, 0, 1] \) is marginalized into \( \mathbf{I}^{(i)} = [-\varepsilon_1, 1 + \varepsilon_2, -\varepsilon_3, 1 + \varepsilon_4] \), where \( \varepsilon_1, \varepsilon_2, \varepsilon_3 \) and \( \varepsilon_4 \) are adaptive margin factors. \( \mathbf{E}^{(i)} \) can be rewritten as:

\[
\min_{\mathbf{P}^{(i)}, \mathbf{V}^{(i)}} \sum_{i=1}^{m} \| \mathbf{L}^{(i)} - \mathbf{P}^{(i)} \mathbf{V}^{(i)} \|_F^2, \tag{2}
\]

\[
\text{s.t. } \mathbf{E}^{(i)} \geq 0, \tag{4}
\]

where \( \odot \) is the Hadamard product operator. In this way, the latent representations can be regressed with adaptive marginalized targets. Consequently, the distances between positive and negative classes are enlarged, and the discrimination of features is enhanced.

2) Semantic Similarity Embedding: With the supervision of refined regression targets, \( \mathbf{E}^{(i)} \) can embed semantic information into latent representations to some extent. However, the equation focuses mainly on the inter-class separability for classification, while the intra-class compactness is not fully explored. For effective representation learning, semantically similar data points should also be close in the latent subspace. In this paper, we use pairwise semantic similarities to improve the intra-class compactness. The semantic similarity \( S^{(ij)} \) between two samples, \( \mathbf{x}^{(i)}_u \) and \( \mathbf{x}^{(j)}_k \), is defined as follows:

\[
S^{(ij)}_{uk} = \frac{\mathbf{L}^{(i)}_u \mathbf{L}^{(j)}_k}{\| \mathbf{L}^{(i)}_u \|_2 \cdot \| \mathbf{L}^{(j)}_k \|_2}, \tag{5}
\]

where the similarity is measured with the cosine distance. To have a compact form, we define the column-normalized label matrix as:

\[
\mathbf{L}^{(i)} = \begin{bmatrix} \mathbf{L}^{(i)}_1 \\ \| \mathbf{L}^{(i)}_1 \|_2 \\ \| \mathbf{L}^{(i)}_2 \|_2 \\ \vdots \\ \| \mathbf{L}^{(i)}_n \|_2 \end{bmatrix}, \tag{6}
\]

Thus, the pairwise similarity matrix between two modalities can be obtained by \( \mathbf{S}^{(ij)} = \mathbf{L}^{(i)} \mathbf{L}^{(j)} \). \( \mathbf{S}^{(ij)} \) is not explicitly computed during the optimization to reduce computational costs. \( S^{(ij)}_{uk} \in [0, 1] \) characterizes the fine-grained semantic similarities among samples, especially for multi-label data. Two samples that have more shared labels will have larger \( S^{(ij)}_{uk} \). To improve the semantic information of modality-specific representations, we preserve the inter-modal similarities by solving the following problem:

\[
\min_{\mathbf{V}^{(i)}} \sum_{i \neq j} \| \mathbf{V}^{(j)} \mathbf{V}^{(i)} - r \mathbf{S}^{(ij)} \|_F^2. \tag{7}
\]

Apart from inter-modal semantic similarity preservation, the intra-modal similarities should also be maintained:

\[
\min_{\mathbf{V}^{(i)}} \sum_{i=1}^{m} \| \mathbf{V}^{(i)} \mathbf{V}^{(i)} - r \mathbf{S}^{(ii)} \|_F^2. \tag{8}
\]

However, the symmetric dot product in \( \mathbf{B}^{(i)} \mathbf{V}^{(i)} - r \mathbf{S}^{(ii)} \mathbf{V}^{(i)} \) makes the optimization difficult. To reduce the quantization error and accelerate the optimization, we adopt an asymmetric strategy:

\[
\min_{\mathbf{V}^{(i)}, \mathbf{B}^{(i)}} \sum_{i=1}^{m} \left( \| \mathbf{V}^{(i)} \mathbf{V}^{(i)} - r \mathbf{S}^{(ii)} \|_F^2 + \| \mathbf{B}^{(i)} - \mathbf{V}^{(i)} \|_F^2 \right). \tag{9}
\]
In this way, the hash codes $B^{(i)}$ can be obtained by approximating $V^{(i)}$, and the semantic information in $V^{(i)}$ can be transferred into $B^{(i)}$.

3) Overall Objective Function: Combining (4), (7) and (9), the overall objective function is

$$\min_{B^{(i)}} \sum_{i=1}^{m} (\|L^{(i)} + R^{(i)} \odot E^{(i)} - P^{(i)} V^{(i)}\|^2_F + \eta \|B^{(i)} - V^{(i)}\|^2_F + \lambda \|B^{(i)T} V^{(i)} - r S^{(i)}\|^2_F + \beta \|\eta V^{(i)} - \rho S^{(i)}\|^2_F/\|S^{(i)}\|^2_F)$$

s.t. $\left\{ \begin{array}{l} E^{(i)} \geq 0, B^{(i)} \in \{-1, 1\}^{r \times n_i}, \\ V^{(i)} V^{(i)T} = n_i I_r, V^{(i)} 1_{n_i} = 0_r. \end{array} \right.$ (10)

In (10), two widely used constraints, $V^{(i)} V^{(i)T} = n_i I_r$, $V^{(i)} 1_{n_i} = 0_r$ are imposed for bit decorrelation and balance. By optimizing the above problem, $V^{(i)}$ containing semantic information can be obtained by utilizing labels, thus generating discriminative hash codes for all modalities.

C. Optimization

To address (10), we develop an alternate discrete optimization method by updating one variable with others fixed in each step. **Update $P^{(i)}$.** Remove the irrelevant terms, and the subproblem w.r.t $P^{(i)}$ is

$$\min_{P^{(i)}} \|L^{(i)} + R^{(i)} \odot E^{(i)} - P^{(i)} V^{(i)}\|^2_F.$$ (11)

With the constraint $V^{(i)} V^{(i)T} = n_i I_r$, the closed-form solution for $P^{(i)}$ can be obtained by setting the derivative w.r.t $P^{(i)}$ to zero:

$$P^{(i)} = (L^{(i)} + R^{(i)} \odot E^{(i)}) V^{(i)T} / n_i.$$ (12)

**Update $V^{(i)}$.** Constrained by $V^{(i)} V^{(i)T} = n_i I_r$, (10) can be transformed into the following problem:

$$\max_{V^{(i)}} \text{Tr} (Z V^{(i)T})$$

s.t. $V^{(i)} V^{(i)T} = n_i I_r, V^{(i)} 1_{n_i} = 0_r.$ (13)

where $Z = P^{(i)T} (L^{(i)} + R^{(i)} \odot E^{(i)}) + \eta B^{(i)} + \lambda \rho B^{(i)T} L^{(i)} + \beta \rho \eta V^{(i)} V^{(i)T} / \|S^{(i)}\|^2_F$. The above problem can be optimized with the following theorem.

**Theorem 3.1:** [48] Given the optimization problem

$$\max_{V^{(i)}} \text{Tr} (Z V^{(i)T}) \text{ s.t. } V^{(i)T} V^{(i)} = n_i I_r, V^{(i)} 1_{n_i} = 0_r,$$ (14)

the eigendecomposition is performed on $ZJZ^T$ with $J$ defined as $J = I_n - \frac{1}{n} 1_n 1_n^T$.

$$ZJZ^T = [N, N] [V \ 0 \ 0] [N, N]^T,$$ (15)

where $V \in \mathbb{R}^{r \times r'}$ is the diagonal eigenvalue matrix and $N \in \mathbb{R}^{r \times r'}$ is comprised of the corresponding eigenvectors, with $r'$ the rank of $ZJZ^T$. Then Gram-Schmidt process is applied on $N$ to obtain the orthogonal matrix $N \in \mathbb{R}^{r \times (r - r')}$.

**D. Hash Functions Learning**

With the optimal hash codes generated from the first step, AMSH then learns the modality-specific hash functions to perform CMR with new queries. Previous methods usually adopt the traditional least squares regression for hash functions learning, which can be described as:

$$\min_{F^{(i)}} \sum_{i=1}^{m} \|B^{(i)} - F^{(i)} X^{(i)}\|^2_F,$$ (22)

where $F^{(i)} \in \mathbb{R}^{r \times d_i}$ is the hash function. Nonetheless, the binary codes in (22) are also too strict for effective regression, and make the hash functions sensitive to noise. Thus, we introduce adaptive margin matrices, and the hash functions can be obtained
by minimizing the following problem:

$$\min_{F(i), M(i)} \sum_{i=1}^{m} \|B(i) + B(i) \odot M(i) - F(i)X(i)\|_F^2$$

s.t. $M(i) \geq 0$. \hspace{1cm} (23)

Intuitively, the margin matrix makes the regression target more flexible and robust by enlarging the distances between positive and negative bits, which makes AMSH more robust to noise and eases the learning of hash functions.

The kernel method is widely adopted to capture the nonlinear data structure. We map the original data from the $i^{th}$ modality to the kernel space with the following function:

$$\phi(x) = \left[ \exp \left( - \frac{\|x - x_1\|^2}{2\delta^2} \right), \ldots, \exp \left( - \frac{\|x - x_k\|^2}{2\delta^2} \right) \right]^T,$$

where $x_1, x_2, \ldots x_k$ are $k$ randomly selected samples and $\delta$ is the bandwidth defined as the mean distance between other training samples and the $k$ samples. The kernel function is applied to all modalities, and the objective function is

$$\min_{F(i), M(i)} \sum_{i=1}^{m} \|B(i) + B(i) \odot M(i) - F(i)\phi(X(i))\|_F^2$$

s.t. $M(i) \geq 0$. \hspace{1cm} (25)

The optimal $F(i)$ and $M(i)$ can be efficiently obtained by performing the alternate optimization scheme, similar to hash codes learning.

**Update $F(i)$**: Set the derivative of (25) w.r.t $F(i)$ to zero, and the optimal solution can be computed by

$$F(i) = (B(i) + B(i) \odot M(i))\phi(X(i))^T(\phi(X(i))\phi(X(i))^T)^{-1}.$$

**Update $M(i)$**: Similar to optimizing $E(i)$ in hash codes learning, the optimal solution can be obtained by

$$M(i) = \max \left( B(i) \odot C(i), 0 \right),$$

where $C(i) = F(i)\phi(X(i)) - B(i)$.

In hash codes learning, $P(i), V(i), B(i)$ and $E(i)$ for each modality are updated alternately to generate the optimal $B(i)$. In hash functions learning, similar to the first step, $F(i)$ and $M(i)$ are alternately updated until convergence. The training procedure is summarized in Algorithm 1. After obtaining the modality-specific hash functions, for a new query from the $i^{th}$ modality, $x(i)$, its hash code can be determined by $b(i) = \text{sgn}(F(i)(\phi(x(i))))$.

**E. Complexity Analysis**

For simplicity, we assume that the numbers of samples of all modalities are the same, i.e., $n_1 = n_2 = \ldots = n_m = n$. As Algorithm 1 shows, the optimization is designed in two steps. Concretely, for hash codes learning, the time complexity includes $O(\tau(cn + cn))$ for updating $P(i), O(\tau(r^3 + cn + mcn))$ for updating $V(i), O(\tau(r^3 + cn))$ for updating $B(i)$ and $O(\tau(r^3 + cn))$ for updating $E(i)$, where $\tau$ is the number of iterations. To obtain hash functions, updating $M(i)$ requires $O(\tau(rd_i + n))$, and obtaining $F(i)$ requires $O(\tau((r + rd_i + d^2_1)n + d^2_2 + d^2_3)))$, where $d_i$ is the kernelized feature length of the $i^{th}$ modality. With $r, c, d \ll n$, the complexity required for optimizing AMSH is linear to the size of datasets. Thus, AMSH is an efficient algorithm and is scalable to large-scale datasets.

**IV. EXPERIMENT**

In this section, experiments are conducted on several benchmark datasets to demonstrate the superiority of AMSH.
A. Datasets

Four datasets are used to conduct experiments, including MIRFlickr-25K [50], NUS-WIDE [51], IAPR TC-12 [52] and XMedia [53], [54], which are widely adopted for CMR evaluation.

MIRFlickr-25K originally consists of 25,000 image-text pairs, collected from the Flickr website, with each one annotated with at least one of the 24 semantic labels. Each image is represented with a 150-D edge histogram feature vector, and each text a 500-D vector. We select textual tags that appear at least 20 times, and remove the samples without textual tags. Finally, the dataset contains 16,738 samples. Following [19], we randomly select 5% as the query set, and the remaining 95% as the training set.

NUS-WIDE is a large database with 269,684 instances associated with 81 semantic concepts. Following [20] and [2], we select 186,577 instances with the 10 most frequent labels. The images and texts are represented as 500-D SIFT feature vectors and 1000-D binary tagging vectors, respectively. 2,000 samples are randomly selected as the query set, and the rest form the training set.

IAPR TC-12 comprises 20,000 worldwide images with related textual descriptions. Each instance is classified as at least one of the 255 semantic labels. An image and a text are denoted by a 512-D GIST feature vector and a 2,912-D BOW vector, respectively. 2,000 image-text pairs are randomly selected as the query set, and the rest as the training set.

XMedia consists of 5,000 texts, 5,000 images, 500 videos, 1,000 audio clips and 500 3D models collected from various websites. Each sample corresponds to a specific label out of 20 categories. We choose three modalities, i.e., images, audios and videos for the experiment, which are represented by 128-D feature vectors, 29-D Mel Frequency Cepstrum Coefficient (MFCC) and 128-D BoVW vectors, respectively. We randomly select 80% as the training set and the rest as the retrieval set.

B. Baselines and Implementation

To demonstrate the effectiveness of AMSH, we evaluate it in two different settings, i.e., the fully paired scenario and the unpaired scenario. For the fully paired scenario, we compare our AMSH with several state-of-the-art cross-modal hashing methods, including CMFH [22], FSH [24], SMFH [9], CRE [25], DCH [19], SCRATCH [2], SRLCH [20], BATCH [35], DAH [36] and ALECH [3]. Among these methods, CMFH, FSH and CRE are unsupervised ones and the rest are supervised ones. These baselines are implemented with the codes provided by the authors, and the parameters of these baselines are carefully tuned according to the original papers to reach their best performance. For the unpaired scenario, SAPMH [27], GSPH [29], UCMH [30], RUCMH [31], EDMH [45] and FMH [32] are adopted for comparison. Among these baselines, RUCMH is an unsupervised method and the others are supervised ones. For AMSH, the retrieval performance is relatively insensitive to the model parameters in appropriate ranges, which will be shown in Section IV-F; and we set $\eta = 1, \lambda = 1e - 3$ and $\beta = 1e - 3$ on all datasets. The number of instance anchors for kernel feature learning is set to 1500 for both the image modality and the textual modality, 500 for the audio modality and 200 for the video modality. The maximum iteration number $\tau$ is set to 15. All the experiments are conducted on a personal computer with MATLAB R2021b, Win10 system, Intel i7-10875H CPU @ 2.3 GHz, 16 GB RAM.

C. Evaluation Metrics

For evaluation, two basic CMR tasks are tested on MIRFlickr-25K, NUS-WIDE and IAPR TC-12, i.e., Image-to-Text (i.e., $I \rightarrow T$) using images as query to search texts, and Text-to-Image (i.e., $I \rightarrow T$) using texts as query to search images. Besides, an experiment involving images, audios ($A$) and videos ($V$) is conducted on XMedia. Mean Average Precision (MAP) and the Precision-Recall (PR) curve are adopted for evaluation. Average Precision (AP) is defined as:

$$AP(x) = \frac{1}{N_g} \sum_{k=1}^{K} P_x(k) \phi_x(k),$$

where $N_g$ is the number of true neighbors, and $P_x(k)$ denotes the precision of the top $k$ retrieved instances. $\phi_x(k) = 1$ if the $k^{th}$ retrieved instance is the true neighbor of the query and 0 otherwise. Ground-truth neighbors of a query are defined as samples with at least one shared semantic label with the query. $K$ is the size of the retrieval set [2], [35]. MAP is defined as:

$$MAP = \frac{1}{N_q} \sum_{i=1}^{N_q} AP(i),$$

where $N_q$ is the size of the query set. Higher MAP values indicate better performance.

D. Evaluation on Fully Paired Data

We first evaluate AMSH in the fully paired scenario and vary the code length from 16 to 128 bits (i.e., 16, 32, 64 and 128). Table I reports the MAP results of AMSH and the baselines on MIRFlickr-25K, NUS-WIDE and IAPR TC-12, and Fig. 3 illustrates the corresponding PR curves. From the experimental results, the following observations can be summarized:

- Compared with all the baselines, AMSH obtains higher MAP results on Image-to-Text and Text-to-Image tasks on three datasets with different code lengths, and the PR curves of AMSH are generally above these baselines. AMSH considers enlarging the distances between different classes to enhance label discrimination, contributing to more discriminative hash codes. For hash functions learning, adaptive marginalization eases the rigid binary linear regression to generate more robust hash functions. The experimental results demonstrate the effectiveness and superiority of AMSH over these methods.

- Generally, supervised methods outperform unsupervised ones, which shows the effectiveness of using semantic information for hash learning. AMSH, ALECH, DAH and BATCH embed high-level semantic information into hash codes provided by the authors, and the parameters of these baselines are carefully tuned according to the original papers to reach their best performance.
codes by constructing the similarity matrix with labels. SCRATCH, SRLCH and DCH construct a linear classifier between hash codes and labels, which generates semantic-aware hash codes. Among the unsupervised methods, CRE and FSH achieve promising results, for CRE adopts a strategy similar to that used by BATCH except for replacing the labels with raw data, and FSH preserves fusion similarity for more discriminative hash codes.

- For most baselines, the performance improves as the code length increases, since more semantic information can be stored with longer binary codes.

- The MAP results of these methods drop drastically on NUS-WIDE, compared to MIRFlickr-25K. One possible reason is that the size of NUS-WIDE is much larger than MIRFlickr-25K with large intra-class variations, making it hard to learn discriminative hash codes. Similar results can be observed on IAPR TC-12, with bigger drop in the MAP results.

- For most methods, MAP results on Text-to-Image task are generally higher than those on Image-to-Text task. one possible factor is that the textual modality is more semantically similar to label information than the image modality.

---

**TABLE I**

| Task       | Method | MIRFlickr-25K | NUS-WIDE | IAPR TC-12 |
|------------|--------|---------------|----------|------------|
| I → T      |        |               |          |            |
|            | CMPH [22] | 0.5838 0.5789 0.5825 0.5858 | 0.3478 0.3450 0.3447 0.3516 | 0.2966 0.2963 0.2965 0.2977 |
|            | FSH [24]  | 0.6185 0.6234 0.6288 0.6280 | 0.4735 0.4771 0.4802 0.4951 | 0.3887 0.3969 0.4087 0.4149 |
|            | CRE [25]  | 0.6029 0.6122 0.6244 0.6213 | 0.4606 0.4774 0.4923 0.4833 | 0.3930 0.4141 0.4234 0.4154 |
|            | DCH [19]  | 0.6796 0.7171 0.7226 0.7318 | 0.5977 0.6110 0.6388 0.6537 | 0.4169 0.5450 0.5492 0.4803 |
|            | SMPIH [9] | 0.5641 0.5556 0.5593 0.5628 | 0.4178 0.4248 0.4270 0.4313 | 0.3341 0.3365 0.3398 0.3440 |
|            | SCRATCH [2] | 0.7076 0.7088 0.7207 0.7251 | 0.6202 0.6193 0.6356 0.6287 | 0.4940 0.4613 0.4714 0.4853 |
|            | SRLCH [20] | 0.6381 0.6474 0.6874 0.6809 | 0.5918 0.6185 0.6330 0.6470 | 0.3560 0.3520 0.3823 0.4065 |
|            | BATCH [35] | 0.7406 0.7468 0.7504 0.7519 | 0.6464 0.6490 0.6593 0.6659 | 0.4807 0.5084 0.5281 0.5388 |
|            | DAH [36] | 0.6929 0.7054 0.7030 0.7107 | 0.6026 0.6250 0.6402 0.6409 | 0.4338 0.4704 0.4721 0.4820 |
|            | ALBCH [3] | 0.7271 0.7430 0.7489 0.7527 | 0.6383 0.6553 0.6567 0.6665 | 0.4790 0.5029 0.5204 0.5321 |
|            | AMSH     | 0.7455 0.7516 0.7572 0.7591 | 0.6544 0.6589 0.6642 0.6690 | 0.4888 0.5163 0.5375 0.5507 |
| T → I      |        |               |          |            |
|            | CMPH [22] | 0.5902 0.5899 0.5949 0.5985 | 0.3362 0.3363 0.3367 0.3365 | 0.3487 0.3500 0.3476 0.3453 |
|            | FSH [24]  | 0.6103 0.6162 0.6183 0.6185 | 0.4991 0.5076 0.5105 0.5192 | 0.3987 0.4083 0.4245 0.4338 |
|            | CRE [25]  | 0.6124 0.6245 0.6416 0.6398 | 0.4712 0.4878 0.5076 0.5034 | 0.4090 0.4405 0.4474 0.4402 |
|            | DCH [19]  | 0.7689 0.7900 0.8072 0.8233 | 0.7147 0.7408 0.7728 0.7821 | 0.5128 0.5513 0.5702 0.6064 |
|            | SMPIH [9] | 0.5678 0.5554 0.5598 0.5633 | 0.3974 0.4031 0.4079 0.4138 | 0.3327 0.3357 0.3394 0.3426 |
|            | SCRATCH [2] | 0.7008 0.7773 0.7999 0.8050 | 0.7611 0.7821 0.7908 0.7826 | 0.5323 0.5767 0.5989 0.6285 |
|            | SRLCH [20] | 0.6884 0.7028 0.7525 0.7491 | 0.7413 0.7656 0.7797 0.7953 | 0.4014 0.4098 0.4654 0.4777 |
|            | BATCH [35] | 0.8134 0.8326 0.8344 0.8405 | 0.7695 0.7906 0.8001 0.7995 | 0.5720 0.6183 0.6531 0.6655 |
|            | DAH [36] | 0.8060 0.8149 0.8243 0.8336 | 0.7445 0.7750 0.7721 0.7894 | 0.5731 0.5999 0.5915 0.6151 |
|            | ALBCH [3] | 0.8101 0.8212 0.8397 0.8457 | 0.7656 0.7865 0.7916 0.7995 | 0.5788 0.6150 0.6455 0.6664 |

Fig. 3. The PR curves of AMSH and baselines on MIRFlickr-25K (top), NUS-WIDE (middle) and IAPR TC-12 (bottom) with paired data.
TABLE II
THE MAP RESULTS OF AMSH AND BASELINES ON MIRFLICKR-25K, NUS-WIDE AND IAPR TC-12 WITH UNPAIRED DATA

| Task | Method | MIRFlickr-25K | NUS-WIDE | IAPR TC-12 |
|------|--------|--------------|----------|------------|
| I → T | | 16 b | 32 b | 64 b | 128 b | 16 b | 32 b | 64 b | 128 b | 16 b | 32 b | 64 b | 128 b |
| | SAPMH [27] | 0.5643 | 0.5636 | 0.5673 | 0.5684 | 0.3384 | 0.3386 | 0.3380 | 0.3384 | 0.3041 | 0.3044 | 0.3042 | 0.3042 |
| | UCMH [30] | 0.5981 | 0.5997 | 0.5912 | 0.5972 | 0.4091 | 0.4092 | 0.4061 | 0.3937 | 0.3070 | 0.3070 | 0.3070 | 0.3071 |
| | RUCMH [31] | 0.5620 | 0.5607 | 0.5626 | 0.5614 | 0.3399 | 0.3487 | 0.3467 | 0.3396 | 0.3252 | 0.3018 | 0.3040 | 0.3031 |
| | GSPH [29] | 0.6541 | 0.6600 | 0.6695 | 0.6705 | 0.5062 | 0.5199 | 0.5349 | 0.5409 | 0.4014 | 0.4222 | 0.4459 | 0.4572 |
| | EDMH [45] | 0.6738 | 0.6728 | 0.6876 | 0.6979 | 0.4594 | 0.5209 | 0.5030 | 0.5052 | 0.3775 | 0.4089 | 0.4255 | 0.4314 |
| | FMH [32] | 0.7130 | 0.7338 | 0.7354 | 0.7377 | 0.6473 | 0.6480 | 0.6297 | 0.6409 | 0.3791 | 0.4192 | 0.4331 | 0.4904 |
| | AMSH | 0.7456 | 0.7515 | 0.7582 | 0.7575 | 0.6580 | 0.6544 | 0.6565 | 0.6641 | 0.4908 | 0.5182 | 0.5370 | 0.5482 |
| T → I | | 16 b | 32 b | 64 b | 128 b | 16 b | 32 b | 64 b | 128 b | 16 b | 32 b | 64 b | 128 b |
| | SAPMH [27] | 0.5625 | 0.5629 | 0.5667 | 0.5672 | 0.3378 | 0.3378 | 0.3378 | 0.3381 | 0.3042 | 0.3044 | 0.3049 | 0.3030 |
| | UCMH [30] | 0.5852 | 0.5556 | 0.5804 | 0.5838 | 0.3982 | 0.3915 | 0.3978 | 0.3895 | 0.3064 | 0.3065 | 0.3066 | 0.3066 |
| | RUCMH [31] | 0.5642 | 0.5613 | 0.5621 | 0.5616 | 0.3293 | 0.3407 | 0.3401 | 0.3383 | 0.3143 | 0.3029 | 0.3042 | 0.3046 |
| | GSPH [29] | 0.6420 | 0.6529 | 0.6607 | 0.6616 | 0.4812 | 0.4916 | 0.5010 | 0.5090 | 0.4021 | 0.4183 | 0.4382 | 0.4503 |
| | EDMH [45] | 0.6622 | 0.6498 | 0.6828 | 0.6898 | 0.5362 | 0.5809 | 0.5990 | 0.6507 | 0.3879 | 0.3906 | 0.4502 | 0.4565 |
| | FMH [32] | 0.7914 | 0.8109 | 0.8169 | 0.8246 | 0.7344 | 0.7289 | 0.7249 | 0.7493 | 0.3821 | 0.4259 | 0.4433 | 0.5080 |
| | AMSH | 0.8310 | 0.8309 | 0.8400 | 0.8436 | 0.7915 | 0.7860 | 0.8048 | 0.8060 | 0.5830 | 0.6328 | 0.6598 | 0.6771 |

**E. Evaluation on Unpaired Data**

In this subsection, we further explore the performance of AMSH in the unpaired setting. Specifically, we adopt the same benchmark datasets used in the last subsection, with all data randomly shuffled. Thus, the data have no one-to-one correspondence. Six baselines are adopted for comparison, i.e., UCMH [30], GSPH [29], RUCMH [31], SAMPH [27], EDMH [45] and FMH [32], which can be used for unpaired data.

Table II shows the MAP results of all the methods, and the PR curves are illustrated in Fig. 4. Some observations can be summarized as follows:

- AMSH greatly outperforms these UCMR methods on both Image-to-Text and Text-to-Image tasks with different code lengths, and the PR curves of AMSH are generally above those of other methods, demonstrating its effectiveness and superiority in handling unpaired multi-modal data.
- AMSH does not suffer from data correspondence, and the performance in the unpaired setting is almost the same as that in the paired setting, for AMSH considers both the inter-modal and intra-modal similarity, making it more robust under all circumstances. FMH and EDMH are both similarity-preserving methods. However, unlike AMSH, they fail to incorporate label discrimination with adaptive margins. GSPH directly utilizes hash codes for similarity preservation without using latent representations, which may limit the representation capacity of hash codes. RUCMH bridges different modalities by linear reconstruction, which is insufficient to capture latent discriminative.
TABLE III
MAP VALUES OF AMSH AND BASELINES ON XMEDIA WITH DIFFERENT CODE LENGTHS IN SIX CROSS-MODAL RETRIEVAL TASKS

| Method | $I \rightarrow A$ | $I \rightarrow V$ | $A \rightarrow I$ |
|--------|------------------|------------------|------------------|
|        | 16b | 32b | 64b | 128b | 16b | 32b | 64b | 128b | 16b | 32b | 64b | 128b |
| SAPMH [27] | 0.0562 | 0.0564 | 0.0575 | 0.0575 | 0.0679 | 0.0756 | 0.0556 | 0.0575 | 0.0516 | 0.0511 | 0.0520 | 0.0508 |
| UCMPH [30] | 0.1008 | 0.1087 | 0.1045 | 0.1099 | 0.0679 | 0.0785 | 0.0884 | 0.0889 | 0.0871 | 0.0906 | 0.0962 | 0.1021 |
| RUCMH [31] | 0.0578 | 0.0600 | 0.0627 | 0.0570 | 0.0612 | 0.0671 | 0.0586 | 0.0679 | 0.0548 | 0.0578 | 0.0552 | 0.0512 |
| GSMPH [29] | 0.3491 | 0.4388 | 0.4904 | 0.5256 | 0.3030 | 0.3908 | 0.4608 | 0.5251 | 0.2701 | 0.4333 | 0.5183 | 0.5590 |
| EDMH [45] | 0.1843 | 0.1536 | 0.2154 | 0.2704 | 0.0833 | 0.1266 | 0.2599 | 0.3029 | 0.1394 | 0.1248 | 0.1486 | 0.1668 |
| FMII [32] | 0.2663 | 0.3422 | 0.3436 | 0.3326 | 0.2387 | 0.3128 | 0.3248 | 0.3352 | 0.2986 | 0.3788 | 0.3968 | 0.4107 |
| AMSH | 0.5122 | 0.5674 | 0.5854 | 0.5948 | 0.5213 | 0.5777 | 0.5972 | 0.6051 | 0.4651 | 0.5077 | 0.5581 | 0.5895 |

F. Parameter Analysis

In this subsection, we discuss the parameter sensitivity on MIRFlickr-25K, NUS-WIDE and IAPR TC-12. There are three parameters in AMSH, i.e., $\eta$, $\lambda$, and $\beta$, which adjust the weights of hash codes learning, intra-modal similarity and inter-modal similarity, respectively. When analyzing each parameter, the others are fixed to default values specified in Section IV-B.

Parameter $\eta$ The MAP results of Image-to-Text and Text-to-Image tasks are given in Fig. 5. $\eta$ controls the distance of binary codes and continuous representations. As $\eta$ ranges from $1e-3$ to $1e0$, the performance remains steady and slightly improves, which shows the robustness and effectiveness of the second term in (10). However, when $\eta$ is set to $1e2$ and $1e3$, the MAP results drop drastically on two tasks. One possible reason is that larger $\eta$ causes $V(i)$ to be as close to $B(i)$ as possible, which limits the representation capacity of $V(i)$ to learn discriminative hash codes.

Parameter $\lambda$ and $\beta$ The MAP results of AMSH versus $\lambda$ and $\beta$ are given in Fig. 5. $\lambda$ influences the intra-modal similarity preservation. It can be observed that when $\beta$ is fixed, with the increase in $\lambda$, the performance is alleviated, which demonstrates the effectiveness of intra-modal similarity preservation. $\beta$ controls the inter-modal similarity preservation. With the increase of $\beta$, the performance of AMSH is improved. When the value of $\lambda$ is large, the MAP drastically drops, for which one possible reason is that larger value of $\lambda$ disturbs the inter-modal similarity preservation, thus compromising the learning of discriminative hash codes. It can also be observed that the sensitivity of AMSH to $\beta$ is less than that to $\lambda$. Even when $\beta$ is large, the MAP results only slightly drop, which demonstrates the effectiveness of
TABLE IV  
THE TRAINING TIME (IN SECOND) OF AMSH AND BASELINES ON TWO DATASETS WITH DIFFERENT CODE LENGTHS

| Method  | Computational Complexity | MIRFlicker-25K | NUS-WIDE |
|---------|--------------------------|---------------|----------|
|         |                          | 16 b         | 32 b     | 64 b | 128 b | 16 b  | 32 b | 64 b | 128 b |
| CBM [22]| $O(r^2 + \sum_{i=1}^{m}(d_i + d_i^2)n)$ | 2.12         | 2.44     | 2.65 | 3.16 | 85.82 | 20.81 | 21.34 | 20.81 |
| FSH [24]| $O(r^2 + \sum_{i=1}^{m}(d_i + d_i^2)n^2)$ | 76.89       | 78.62    | 80.36 | 81.29 | 6708.35 | 7610.14 | 6714.35 |
| CRE [25]| $O(r^2 + \sum_{i=1}^{m}(d_i + d_i^2)n)$ | 56.86       | 61.03    | 58.91 | 63.07 | 520.59 | 541.53 | 563.83 |
| DCH [19]| $O(r^2 + \sum_{i=1}^{m}(d_i^2 + d_i cr)n)$ | 1.94       | 3.01     | 6.97 | 21.34 | 20.81 | 31.60 | 75.48 | 245.37 |
| SMFH [9]| $O(r^2 + \sum_{i=1}^{m}d_in)$ | 4.71        | 4.94     | 13.54 | 16.94 | 11.81 | 12.20 | 13.33 | 14.88 |
| SCRATCH [2]| $O(r^2 + \sum_{i=1}^{m}d_ir + cr)n)$ | 0.76       | 0.93     | 1.28 | 2.22 | 24.16 | 26.78 | 33.34 | 46.49 |
| SRLCH [20]| $O(r^2 + \sum_{i=1}^{m}d_in)$ | 0.93       | 1.26     | 2.11 | 3.63 | 21.14 | 26.95 | 43.94 | 79.74 |
| BATCH [35]| $O(r^2 + \sum_{i=1}^{m}d_ir + cr)n)$ | 0.61       | 0.80     | 1.30 | 2.09 | 5.77 | 7.44 | 11.96 | 21.78 |
| GSPH [29]| $O(r^2 + \sum_{i=1}^{m}d_ir)$ | 75.25       | 93.82    | 184.92 | 360.08 | 903.25 | 1125.84 | 2219.04 | 4529.60 |
| EDMH [45]| $O(r^2 + \sum_{i=1}^{m}(d_i^2 + d_i r)n)$ | 1.13       | 1.54     | 2.26 | 3.05 | 8.37 | 9.22 | 12.27 | 18.00 |
| DAH [36]| $O(r^2 + \sum_{i=1}^{m}(d_i^2 + d_i r)n)$ | 0.40       | 0.67     | 1.35 | 2.05 | 4.25 | 7.96 | 15.06 | 31.85 |
| ALECH [3]| $O(r^2 + \sum_{i=1}^{m}(d_i^2 + d_i r)n)$ | 0.42       | 0.60     | 1.34 | 2.65 | 10.12 | 12.18 | 16.12 | 23.81 |
| AMSH     | $O(r^2 + \sum_{i=1}^{m}(d_i^2 + d_i r)n)$ | 2.39       | 2.48     | 3.07 | 4.12 | 21.89 | 24.05 | 30.36 | 42.64 |

inter-modal similarity preservation. From the parameter analysis, it can be summarized that the suitable parameter ranges for $\lambda$ and $\beta$ are from $1e^{-3}$ to $1e0$.

G. Training Time and Convergence

To solve AMSH, we develop an alternate optimization algorithm. Table IV reports the training time of AMSH and some baselines on MIRFlickr-25K and NUS-WIDE, including hash codes learning and hash functions learning. We do not compare the query time, for it is nearly the same for all methods to obtain hash codes by well-learned functions and compute Hamming distances. Generally speaking, BATCH reaches the fastest training speed for its efficient matrix factorization and semantic information preservation. Among all the methods, FSH is the slowest, in which fusion similarity embedding is time-consuming. For GSPH, the discrete optimization procedure is implemented in an one-by-one manner, which is inefficient. It can be observed that AMSH achieves a promising training speed, though it is not the fastest method. One important factor is that AMSH is a two-step method, where in the second step the adaptive margin matrix and the hash function are updated with an alternate optimization method, and each subproblem can be solved efficiently. Traditional two-step hash methods, i.e., SCRATCH [2] and BATCH [35], obtain the hash functions with direct linear regression, which is fast, and yet compromises the performance, as discussed in Section III. AMSH can handle paired and unpaired scenarios while achieving better performance than these baselines, with an acceptable increase in training time. Table IV also reports the computational complexity of these methods from their original papers, most of which is linear to the size of datasets. FSH and GSPH contain complexity with $n^2$, making them unsuitable.

In Section III, we theoretically prove the convergence property of AMSH. In this subsection, we present the convergence curves with experiments conducted on MIRFlicker-25K, NUS-WIDE and IAPR TC-12. For better observation, we normalize the objective function values by dividing each point by the largest value. As shown in Fig. 6, the objective function value monotonously decreases with each iteration, and after 15 iterations, it reaches a stable state.

H. Ablation Study

For AMSH, adaptive marginalization and similarity preservation are adopted for hash codes and hash functions learning. In this subsection, we analyze their effects by ablation study. From AMSH, we derive four variants, i.e., AMSH-s, AMSH-t, AMSH-k and AMSH-m. AMSH-s discards the intra-modal similarity preservation, i.e., the third term in (10). AMSH-t drops the inter-modal similarity preservation term. AMSH-k adopts no kernel function and utilizes the original data. AMSH-m discards the adaptive margin matrix.

The performance of AMSH is compared with that of these four variants on three datasets in the paired scenario with different code lengths. The MAP results are reported in Table V. It can be observed that AMSH generally outperforms the four variants in all cases. AMSH-s and AMSH-t both achieve low MAP results, which demonstrates the importance of similarity-preserving regularization terms. AMSH-k achieves promising results. However, it is still inferior to AMSH, which proves the effectiveness of the kernel methods to capture nonlinear data structures. The MAP results of AMSH-m are comparably high yet still lower than those of AMSH, showing that the adaptive marginalization helps improve performance.

I. Comparison With Deep Hashing Methods

Over the recent years, due to the great capacity of neural networks to capture high-level semantic information, deep learning based methods have contributed to improving the performance of cross modal retrieval. To further demonstrate the superiority of AMSH, we compare it with three state-of-the-art supervised

Fig. 6. The convergence analysis of AMSH on MIRFlicker-25K (left), NUS-WIDE (middle) and IAPR TC-12 (right) with various code lengths.
TABLE V
THE MAP RESULTS OF AMSH AND ITS FOUR VARIANTS ON THREE DATASETS WITH PAIRED DATA

| Task  | Method       | MIRFlickr-25K | NUS-WIDE | LAPR TC-12 |
|-------|--------------|---------------|----------|------------|
|       |              | 16 b  | 32 b  | 64 b  | 128 b | 16 b  | 32 b  | 64 b  | 128 b | 16 b  | 32 b  | 64 b  | 128 b |
| I → T| AMSH-s       | 0.6425 | 0.6166 | 0.6012 | 0.5919 | 0.5346 | 0.4499 | 0.5151 | 0.4720 | 0.4334 | 0.4078 | 0.3876 | 0.3741 |
|       | AMSH-H       | 0.6192 | 0.5767 | 0.6376 | 0.6625 | 0.4704 | 0.3101 | 0.3744 | 0.4371 | 0.3148 | 0.3444 | 0.3407 | 0.3030 |
|       | AMSH-k       | 0.7115 | 0.7226 | 0.7240 | 0.7258 | 0.6386 | 0.6391 | 0.6512 | 0.6552 | 0.4642 | 0.4898 | 0.5085 | 0.5206 |
|       | AMSH-m       | 0.7381 | 0.7490 | 0.7533 | 0.7580 | 0.6432 | 0.6502 | 0.6534 | 0.6678 | 0.4859 | 0.5128 | 0.5352 | 0.5498 |
|       | AMSH         | 0.7455 | 0.7516 | 0.7572 | 0.7591 | 0.6544 | 0.6589 | 0.6642 | 0.6690 | 0.4888 | 0.5163 | 0.5375 | 0.5507 |
| T → I| AMSH-s       | 0.6915 | 0.6562 | 0.6327 | 0.6144 | 0.6412 | 0.5266 | 0.5679 | 0.5043 | 0.5146 | 0.4955 | 0.4753 | 0.4476 |
|       | AMSH-H       | 0.6307 | 0.5666 | 0.6342 | 0.6902 | 0.4387 | 0.2677 | 0.4528 | 0.4424 | 0.3067 | 0.3571 | 0.3378 | 0.3224 |
|       | AMSH-k       | 0.8208 | 0.8332 | 0.8372 | 0.8374 | 0.7748 | 0.7844 | 0.8026 | 0.8015 | 0.5701 | 0.6085 | 0.6397 | 0.6522 |
|       | AMSH-m       | 0.8178 | 0.8288 | 0.8351 | 0.8382 | 0.7715 | 0.7772 | 0.7876 | 0.8013 | 0.5752 | 0.6167 | 0.6514 | 0.6709 |
|       | AMSH         | 0.8217 | 0.8359 | 0.8407 | 0.8477 | 0.7784 | 0.7975 | 0.8040 | 0.8092 | 0.5914 | 0.6309 | 0.6632 | 0.6787 |

V. CONCLUSION

In this paper, we propose a novel Adaptive Marginalized Semantic Hashing method (AMSH) to address the CMR problem in both paired and unpaired cases. AMSH is a two-step method. In the first step, to enhance label discrimination, AMSH incorporates adaptive margin matrices to enlarge the distances between different classes, which alleviates the rigid zero-one linear regression. To handle unpaired data, AMSH generates semantic-aware modality-specific latent representations, and exploits labels to preserve the inter-modal and intra-modal semantic similarities into latent representations and hash codes. In the second step, AMSH utilizes adaptive margin matrices to enlarge the gaps between positive and negative bits, which increases the discrimination and robustness of hash functions. A discrete alternate optimization scheme is proposed to solve AMSH. Experiments on several datasets demonstrate that AMSH outperforms some state-of-the-art cross-modal hashing methods in terms of accuracy and computational costs.

AMSH adopts linear regression to obtain hash codes and hash functions, which is not enough to preserve the intrinsic data correlations. AMSH can be further improved by integrating it with deep neural networks for better performance. Besides, accurate label information of different modal data is not always available. Thus, in future work, we will extend the proposed AMSH to handle unsupervised unpaired data.

REFERENCES

[1] F. Shen, C. Shen, W. Liu, and H. T. Shen, “Supervised discrete hashing,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2015, pp. 37–45.
[2] Z. Chen et al., “SCRATCH: A scalable discrete matrix factorization hashing framework for cross-modal retrieval,” IEEE Trans. Circuits Syst. Video Technol., vol. 30, no. 7, pp. 2262–2275, Jul. 2020.
[3] H. Li, C. Zhang, X. Jia, Y. Gao, and C. Chen, “Adaptive label correlation based asymmetric discrete hashing for cross-modal retrieval,” IEEE Trans. Knowl. Data Eng., vol. 35, no. 2, pp. 1185–1199, Feb. 2023.
[4] C. Zhang, H. Li, Y. Gao, and C. Chen, “Weakly-supervised enhanced semantic-aware hashing for cross-modal retrieval,” IEEE Trans. Knowl. Data Eng., vol. 35, no. 2, pp. 1185–1199, Feb. 2023.
[5] V. Ranjan, N. Rasiwasia, and C. Jawahar, “Multi-label cross-modal retrieval,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2015, pp. 4094–4102.
[6] K. Wang, R. He, L. Wang, W. Wang, and T. Tan, “Joint feature selection and subspace learning for cross-modal retrieval,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 38, no. 10, pp. 2010–2023, Oct. 2016.
Chao Zhang (Graduate Student Member, IEEE) received the B.E. and M.E. degrees in control science and engineering in 2018 and 2021, respectively from Nanjing University, Nanjing, China, where he is currently working toward the Ph.D. degree with the Department of Control Science and Intelligence Engineering. His research interests include machine learning and pattern recognition.

Huaxiong Li (Senior Member, IEEE) received the M.E. degree in control theory and control engineering from Southeast University, Nanjing, China, in 2006, and the Ph.D. degree from Nanjing University, Nanjing, in 2009. He was a Visiting Scholar with the Department of Computer Science, University of Regina, Regina, SK, Canada, from 2007 to 2008, and a Visiting Scholar with the University of Hong Kong, Hong Kong, in 2010. He is currently a Professor with the Department of Control Science and Intelligence Engineering, Nanjing University. His research interests include machine learning, data mining, and computer vision. Dr. Li is a Committee Member of the China Association of Artificial Intelligence (CAAI) Machine Learning Committee/Granular Computing and Knowledge Discovery (GCKD) Committee, a CAAI Senior Member, a China Computer Federation Distinguished Member, and a Committee Member of the JiangSu Association of Artificial Intelligence Pattern Recognition Committee.

Xiuyi Jia (Member, IEEE) received the Ph.D. degree in computer science from Nanjing University, Nanjing, China, in 2011. He is currently a Professor with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing. His research interests include machine learning, granular computing, and data mining.

Chunlin Chen (Senior Member, IEEE) received the B.E. degree in automatic control and the Ph.D. degree in control science and engineering from the University of Science and Technology of China, Hefei, China, in 2001 and 2006, respectively. He was a Visiting Scholar with Princeton University, Princeton, NJ, USA, from 2012 to 2013. He was a Visiting Positions with the University of New South Wales at Canberra, Canberra, NSW, Australia, and the City University of Hong Kong, Hong Kong. He is currently a Professor and the Vice Dean of the School of Management and Engineering, Nanjing University, Nanjing, China. His research interests include machine learning, pattern recognition, intelligent information processing, and quantum control. Dr. Chen is the Chair of the Technical Committee on Quantum Cybernetics and the IEEE Systems, Man and Cybernetics Society.