FDDH: Fast Discriminative Discrete Hashing for Large-Scale Cross-Modal Retrieval

Xin Liu, *Senior Member, IEEE*, Xingzhi Wang, and Yiu-ming Cheung, *Fellow, IEEE*

***Abstract*— Cross-modal hashing, favored for its effectiveness and efficiency, has received wide attention to facilitating efficient retrieval across different modalities. Nevertheless, most existing methods do not sufficiently exploit the discriminative power of semantic information when learning the hash codes while often involving time-consuming training procedure for handling the large-scale dataset. To tackle these issues, we formulate the learning of similarity-preserving hash codes in terms of orthogonally rotating the semantic data, so as to minimize the quantization loss of mapping such data to hamming space and propose an efficient fast discriminative discrete hashing (FDDH) approach for large-scale cross-modal retrieval. More specifically, FDDH introduces an orthogonal basis to regress the targeted hash codes of training examples to their corresponding semantic labels and utilizes the ε-dragging technique to provide provable large semantic margins. Accordingly, the discriminative power of semantic information can be explicitly captured and maximized. Moreover, an orthogonal transformation scheme is further proposed to map the nonlinear embedding data into the semantic subspace, which can well guarantee the semantic consistency between the data feature and its semantic representation. Consequently, an efficient closed-form solution is derived for discriminative hash code learning, which is very computationally efficient. In addition, an effective and stable online learning strategy is presented for optimizing modality-specific projection functions, featuring adaptivity to different training sizes and streaming data. The proposed FDDH approach theoretically approximates the bi-Lipschitz continuity, runs sufficiently fast, and also significantly improves the retrieval performance over the state-of-the-art methods. The source code is released at https://github.com/starxliu/FDDH.**

Manuscript received 11 June 2020; revised 21 December 2020 and 2 April 2021; accepted 26 April 2021. Date of publication 12 May 2021; date of current version 28 October 2022. This work was supported in part by the National Science Foundation of China under Grant 61673185 and Grant 61672444, in part by the National Science Foundation of Fujian Province under Grant 2020J01083 and Grant 2020J01084, in part by the Quanzhou City Science and Technology Program of China under Grant 2018C107R, in part by the Hong Kong Baptist University (HKBU) under Grant RC-FNRA-IG/18-19/SCI/03 and Grant RC-IRCMs/18-19/01, in part by the Innovation and Technology Fund of Innovation and Technology Commission of the Government of the Hong Kong under Project ITS/339/18, and in part by SZSTC under Grant SGDX20190816230207535. *(Corresponding author: Xin Liu.)*

Xin Liu is with the Department of Computer Science, Huaqiao University, Xiamen 361021, China, also with the Xiamen Key Laboratory of Computer Vision and Pattern Recognition, Huaqiao University, Xiamen 361021, China, and also with the Fujian Key Laboratory of Big Data Intelligence and Security, Xiamen 361021, China (e-mail: xliu@hqu.edu.cn).

Xingzhi Wang is with the School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou 510006, China (e-mail: wangxzh58@mail2.sysu.edu.cn).

Yiu-ming Cheung is with the Department of Computer Science, Hong Kong Baptist University, Hong Kong (e-mail: ymc@comp.hkbu.edu.hk).

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TNNLS.2021.3076684.

Digital Object Identifier 10.1109/TNNLS.2021.3076684

***Index Terms*— ε-dragging, bi-Lipschitz continuity, cross-modal hashing, online strategy, orthogonal basis, semantic margin.**

I. Introduction

With the explosive growth of multimedia data, automated mechanisms are needed to establish a similarity link from one multimedia item to another if they are related to each other. In order to maximally benefit from the richness of multimodal media data, cross-modal retrieval has recently gained increasing attention to approximate nearest neighbors search across different modalities, such as using an image to search the relevant text documents or using text to search the relevant images. Nevertheless, in real multimedia searching, the multimodal data usually span in different feature spaces, and each feature characterizes the data contents from different aspects. Such modality heterogeneity has been widely considered as a great challenge to cross-modal retrieval. To alleviate this concern, early naive studies [1] learn a common latent subspace to minimize the modality heterogeneity, indicating its possibility to directly compare the features from different modalities. Although these methods have achieved impressive performance, there is still a serious limitation for them. That is, the existing subspace approaches are computationally expensive to deal with large-scale and high-dimensional media data.

Hashing, favored for its low storage cost and fast retrieval speed, has recently attracted much more attention due to its effectiveness for indexing large-scale multimedia data [2]. The main objective of cross-modal hashing is to learn the compact binary codes for representing multiple modalities, while faithfully preserving both intramodality similarity and intermodality similarity. In recent years, various cross-modal hashing researches have been devoted to compress multimodal data in an isomorphic Hamming space and bridge their heterogeneity gap, which can be roughly divided into unsupervised methods [3]–[6] and supervised methods [7]–[10]. Since the label information is helpful to construct the semantic correlations across different modalities, the supervised methods often leverage the semantic labels to further improve retrieval performance over unsupervised cases. In spite of some supervised methods that have achieved impressive retrieval performance, it still remains a challenging task to achieve efficient cross-modal retrieval mainly due to the complex integration of semantic gap, modality heterogeneity, and mixed binary-integer optimization problem. For instance, the utilization of label supervision in terms of large pairwise

similarity or affinity matrix [11], [12] inevitably increases the computational cost during the hash code learning process. Specifically, the discrete constraints imposed on the binary codes and hash objective functions often lead to NP-hard optimization problems. To simplify such optimization, some supervised methods relax the original discrete optimization problem into the continuous one [13], and such relaxation scheme may deteriorate the accuracy of the learned binary codes due to the accumulated quantization error. Besides, recent supervised discrete hashing methods attempt to learn the hash code bit by bit [10], [14], which often involves large iterations in the learning process. In recent years, deep cross-modal hashing approaches [15], [16], integrate the feature learning and hashing code learning together, which always yields outstanding performance on many benchmarks. Nevertheless, these deep works often involve the exhaustive search for learning optimum parameters, which is quite time-consuming.

To the best of our knowledge, it can be well found that existing supervised methods just consider the semantic-preserving property provided by the label supervision, which does not carefully explore the discriminative power of semantic information when learning the hash codes. Consequently, the learned hash codes are not discriminative enough for high retrieval performance. Besides, exiting methods rarely consider shortening the optimization iterations during the training process, making it unscalable to large-scale datasets. Therefore, it is still desirable to study a fast and discriminative cross-modal hashing method from a practical viewpoint.

In this article, we present a fast discriminative discrete hashing (FDDH) approach to facilitating efficient retrieval across different modalities. The main contributions of the proposed FDDH approach are fourfold as follows.

1) The efficient learning of similarity-preserving hash codes is newly formulated in terms of orthogonally rotating the semantic data, whereby the quantization loss of mapping such data to hamming space can be well minimized.
2) The label values are reasonably relaxed to increase the discrimination power of semantic information, and ε-dragging methodology is introduced to provide a large margin property for discriminative hash code learning. To the best of our knowledge, this strategy has yet to be studied thus far in cross-modal hashing.
3) A novel orthogonal regression method is proposed for learning semantic-preserving hash codes, and an efficient closed-form solution is derived in a discrete and discriminative manner, which accelerates the learning process and makes a less computational effort.
4) Extensive experiments on three public benchmarks highlight the advantages of FDDH under various cross-modal retrieval scenarios and show its improved retrieval performance over the state-of-the-art ones.

The remainder of this article is structured as follows. Section II briefly surveys the related works of cross-modal hashing. Section III-B elaborates on FDDH and its theory analysis. The experimental results and discussions are provided in Section IV. Finally, we draw a conclusion in Section V.

II. Related Work

The primary issue of cross-modal retrieval lies in that the features of different modalities often span in different feature spaces, indicating its impossibility to be compared directly. To tackle this problem, the canonical correlation analysis (CCA) [17] is probably the most popular method that aims to learn a common latent subspace from two modalities, where the features of different modalities can be directly correlated and compared. Similarly, partial least-squares (PLS) [18] and bilinear model (BLM) [19] also learn a common latent subspace for cross-modal retrieval. Remarkably, these methods do not utilize the semantic labels for the discriminative analysis. Therefore, some extensions leverage the valuable label information to improve the retrieval performance. For instance, the multiview CCA framework [20] directly links the image and text views under the semantic class labels, and other extensions, e.g., cluster-CCA [21] and multilabel CCA [22], have also been developed to address cross-modal retrieval problem. Besides, multimodal deep models, e.g., multimodal autoencoder [23] and deep CCA [24], have been recently proposed to construct more powerful subspace in the hidden layers of the neural network while capturing the nonlinear correlation between the heterogeneous modalities. Remarkably, these deep methods are computationally expensive to process large-scale and high-dimensional media data.

Cross-modal hashing has received wide attention due to its effectiveness in reducing memory cost and improving query speed, and its main difficulty is to learn compact hash codes that have an additional property to preserve the semantic relationship between different modalities. It is noted that the recent multimodal hashing [25], [26] or multiview hashing [27] works often fuse the features from different modalities or views to learn the comprehensive hash codes, which are essentially different from cross-modal hashing that concentrates on discovering the shared hash codes to correlate multiple modalities. Furthermore, multimodal hashing is designed for multimedia search when multimodal features are all provided at the query stage, while cross-modal hashing aims to retrieve the most relevant objects represented by other modalities for a given query characterized by one modality. In the following, we mainly survey the cross-modal hashing works. In the past, various cross-modal hashing attempts have been proposed, mostly in either an unsupervised manner where the labels are unavailable or a supervised manner where the labels are explicitly provided. Unsupervised cross-modal hashing methods mainly learn the projection functions to map the original feature spaces into hamming spaces. Accordingly, intermedia hashing (IMH) [3] obtains a common hamming space by preserving the interview and intraview consistency, while collective matrix factorization hashing (CMFH) [4], [28] jointly learns the unified hash codes and hash functions by collective matrix factorization. Similarly, latent semantic sparse hashing (LSSH) [5] first utilizes sparse coding and matrix factorization to extract latent semantic features and then quantizes such latent semantic features for hash code generation. Although these methods are able to capture the semantic correlations between heterogeneous modalities, the available

class label information remains unexplored, and the derived hash codes are not discriminative enough for high retrieval performance.

Supervised cross-modal hashing methods primarily exploit the available label information to learn the compact hash codes, which can well mitigate the semantic gap between heterogeneous modalities and generally show improved performance than that of unsupervised ones. For instance, semantic correlation maximization (SCM) [7] seamlessly integrates the semantic labels into hash code learning procedures, while semantic preserved hashing (SePH) [11] and its extension [29] generate the unified binary code by modeling an affinity matrix in a probability distribution. Besides, supervised matrix factorization hashing (SMFH) [13] utilizes the label supervision to produce unified hash codes while maintaining the label consistency and local geometric consistency. It is noted that hashing is essentially a discrete learning problem, and these methods utilize relaxation-based continuous schemes to simplify the original binary optimization problem. Nevertheless, the approximated solutions with relaxation are suboptimal, which often degrades the discriminative power of the final hash codes, possibly due to the accumulated quantization error. In contrast to this, discrete methods try to directly solve the discrete problem without continuous relaxation. Along this way, discrete cross-modal hashing (DCH) [10], discrete latent factor model hashing (DLFH) [30], sequential discrete hashing [31], cross-modal discrete hashing (CMDH) [32], asymmetric discrete cross-modal hashing (ADCH) [33], and nonlinear robust discrete hashing (NRDH) [34] directly update hash codes while retaining the discrete constraints for more compact hash codes. In addition, generalized semantic preserving hashing (GSePH) [12] constructs an affinity matrix by label supervision to discretely approximate hash codes, while scalable discrete matrix factorization hashing (SCRATCH) [35], subspace relation learning for cross-modal hashing (SRLCH) [36], and scalaBle Asymmetric discreTe cross-modal hashing (BATCH) [37] improve the collective matrix factorization to discretely learn the hash codes. Although these supervised methods are able to achieve efficient cross-modal retrieval, they do not fully exploit the discriminative power of semantic information when learning hash codes and often involve a bit large iterations in the training procedures.

In recent years, multimodal deep learning has proven to be effective in capturing the high-level correlation in different modalities. Accordingly, recent deep cross-modal hashing works [16], [38]–[40] jointly learn the high-level features and hash code in an integrated way, whereby the hash codes can be optimized with feature representation learning through the multilayer neural networks. Although these deep methods have shown outstanding performance on many benchmarks, they are always constrained by computational complexity and exhaustive search for learning optimum network parameters. Another potential limitation is that these deep methods still employ the binary quantization functions to generate hash codes from the feature space, which cannot guarantee the learned binary codes to be semantically discriminative for characterizing the heterogeneous modalities. Therefore, it is still desirable to study the fast and discriminative indexing techniques for efficient cross-modal retrieval practically.

III. Fast Discriminative Discrete Hashing

Without loss of generality, this section mainly focuses on fast discriminative discrete hashing with only two modalities (i.e., image and text), and the proposed cross-modal hashing framework can be easily extended to three or more modalities.

A. Notation and Problem Formulation

Throughout this article, the uppercase bold font characters are utilized to denote matrices, while the lowercase bold font characters are select to represent data vectors. For simplicity, let $\mathbf{X}^t = \{\mathbf{x}_i^t\}_{i=1}^n$, $t = 1, 2$ be the training image–text examples, where $\mathbf{x}_i^t \in \mathbb{R}^{d_t}$ is the ith sample, d_t is the feature length in the tth modality, and n is the training number. Without loss of generality, the data points are assumed to be zero-centered, which means that $\sum_{i=1}^n \mathbf{x}_i^t = 0$. In practice, the zero-one matrix $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_n] \in \mathbb{R}^{c\times n}$ is utilized to represent the label matrix corresponding to the training samples, where each column vector $\mathbf{y}_i \in \mathbb{R}^{c\times 1}$ is simply defined as follows: if the tth training sample comes from the jth class (in general, each sample belongs to no less than one class), then the jth element of such column vector is 1, while the remaining elements are 0. The goal of cross-modal hashing is to learn binary codes matrix $\mathbf{H} = \{\mathbf{h}_i\}_{i=1}^n \in \mathbb{R}^{q\times n}$ for all training instances and modality-specific projection matrix $\mathbf{P}_t$ for linking the original feature space and the common hamming space, where $\mathbf{h}_i \in \{-1, 1\}^{q\times 1}$ is q bits hash code of the ith instance.

B. Proposed FDDH Methodology

1) Semantic-Preserving Hash Code Learning: For hashing representation learning, compactness is a critical criterion to guarantee its performance in efficient similarity search. Therefore, it is imperative to produce an efficient code in which the variance of each bit is maximized and the bits are pairwise uncorrelated. To learn the compact hash code, the orthogonal transformation is popular for discriminative hash code learning, and Wang *et al.* [25] present an orthogonal learning structure to reduce the redundant information lying in the hash representation. Note that this work is designed to learn modality-specific hash codes jointly for multimodal data representation, while cross-modal hashing learns common hash codes to preserve the semantic relationship across different modalities. Benefit from the observations of work [41], the learning of similarity-preserving binary codes can be successfully formulated in terms of orthogonally rotating zero-centered PCA-projected data, so as to decompose correlations among hash bits and minimize the quantization error of mapping that data to the vertices of a zero-centered binary hypercube. In practice, the length of hash bits is often larger than the number of semantic categories, i.e., $q \geq c$. Geometrically, it is not difficult to find that hamming space is consistent with the vertices of the unit hypercube, and we heuristically introduce an orthogonal basis

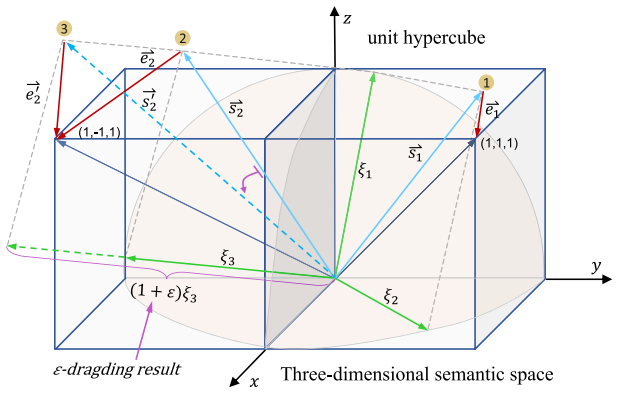

Fig. 1. Geometric example of 3-D semantic subspace illustration for semantic-preserving and quantization error reduction.

$\mathbf{C} = \{\boldsymbol{\xi}_i\}_{i=1}^{c} \in \mathbb{R}^{q\times c}$ to learn the semantic representation. More specifically, we propose to orthogonally rotate the label vector $\mathbf{y}_i$ to approximate its semantic data $\mathbf{s}_i$ (i.e., $\mathbf{s}_i = \mathbf{C}\mathbf{y}_i$) while ensuring $\mathbf{s}_i$ to be as close as possible to the vertices of unit hypercube. Accordingly, the optimal orthogonal basis $\mathbf{C}$ can be obtained by minimizing the following quantization loss:

$$\min_{\mathbf{C}} \sum_{i=1}^{n} \|\mathrm{sgn}(\mathbf{C}\mathbf{y}_i) - \mathbf{C}\mathbf{y}_i\|_2^2, \quad \text{s.t. } \mathbf{C}^{\mathrm{T}}\mathbf{C} = \mathbf{I}_c \tag{1}$$

where $\mathbf{I}_c$ is a c-order identity matrix. A geometric example for 3-D semantic subspace illustration is shown in Fig. 1, where $\vec{s_1}$ and $\vec{s_2}$ are semantic vectors of instances that formed by an orthogonal basis $\mathbf{C} = \{\xi_1, \xi_2, \xi_3\}$ and label vectors $\mathbf{y}_1 = (1, 1, 0)^{\mathrm{T}}$, $\mathbf{y}_2 = (1, 0, 1)^{\mathrm{T}}$. The quantization errors are attainted by computing the sum of the length of $\vec{e_1}$ and $\vec{e_2}$. As indicated in (1), the smaller quantization loss indicates the better binary code, which can well preserve the structure property of the semantic data. As hash codes are often obtained by quantizing the semantic data, (1) can be rewritten as

$$\min_{\mathbf{C}} \sum_{i=1}^{n} \|\mathbf{h}_i - \mathbf{C}\mathbf{y}_i\|_2^2, \quad \text{s.t. } \mathbf{C}^{\mathrm{T}}\mathbf{C} = \mathbf{I}_c. \tag{2}$$

Therefore, the learning of semantic-preserving binary codes can be formulated in terms of orthogonally rotating the semantic data to minimize the quantization loss

$$\min_{\mathbf{H},\mathbf{C}} \|\mathbf{H} - \mathbf{C}\mathbf{Y}\|_F^2, \quad \text{s.t. } \mathbf{H} \in \{-1, 1\}^{q\times n}, \ \mathbf{C}^{\mathrm{T}}\mathbf{C} = \mathbf{I}_c. \tag{3}$$

Remarkably, the optimization problem in (2) ensures that $\mathbf{C}\mathbf{y}_i \to \mathbf{h}_i$. Considering the linear structural equation $\mathbf{A}\mathbf{x} = \mathbf{b}$, the solution of $\mathbf{x}$ can be determined when $\mathrm{rank}(\mathbf{A}) = \mathrm{rank}(\mathbf{A}, \mathbf{b})$. Formally, we replace $\mathbf{A}$ with $\mathbf{C}$ and $\mathbf{b}$ with $\mathbf{h}_i$ and obtain $\mathbf{C}\mathbf{x} = \mathbf{h}_i$. Since $\mathbf{C}$ is constructed from the orthogonal basis in semantic space, and $\mathbf{h}_i$ can be linearly represented with these basis, resulting $\mathrm{rank}(\mathbf{C}) = \mathrm{rank}(\mathbf{C}, \mathbf{h}_i)$. Since $\mathbf{C}\mathbf{y}_i \to \mathbf{h}_i$, $\mathbf{y}_i$ is in close proximity to the solution of $\mathbf{x}$ in equation $\mathbf{C}\mathbf{x} = \mathbf{h}_i$, resulting $\|\mathbf{y}_i - \mathbf{x}\|_2 = \|\mathbf{y}_i - \mathbf{C}^{\mathrm{T}}\mathbf{h}_i\|_2 \to 0$, and we obtain the following equivalent equation:

$$\min_{\mathbf{C}} \sum_{i=1}^{n} \|\mathbf{y}_i - \mathbf{C}^{\mathrm{T}}\mathbf{h}_i\|_2^2, \quad \text{s.t. } \mathbf{C}^{\mathrm{T}}\mathbf{C} = \mathbf{I}_c. \tag{4}$$

TABLE I
IMPACTS OF ε-DRAGGING ON CLASS LABEL VECTORS

samples	label $\mathbf{y}^{\mathrm{T}}$	$\mathbf{y}^{\mathrm{T}}$ after ε-dragging	constraint
$\mathbf{x}_1$	$[1, 0, 0]$	$[1+\varepsilon_{11}, -\varepsilon_{12}, -\varepsilon_{13}]$	$\{\varepsilon_{11}, \varepsilon_{12}, \varepsilon_{13}\} \geq 0$
$\mathbf{x}_2$	$[1, 0, 0]$	$[1+\varepsilon_{21}, -\varepsilon_{22}, -\varepsilon_{23}]$	$\{\varepsilon_{21}, \varepsilon_{22}, \varepsilon_{23}\} \geq 0$
$\mathbf{x}_3$	$[0, 1, 0]$	$[-\varepsilon_{31}, 1+\varepsilon_{32}, -\varepsilon_{33}]$	$\{\varepsilon_{31}, \varepsilon_{32}, \varepsilon_{33}\} \geq 0$
$\mathbf{x}_4$	$[0, 1, 0]$	$[-\varepsilon_{41}, 1+\varepsilon_{42}, -\varepsilon_{43}]$	$\{\varepsilon_{41}, \varepsilon_{42}, \varepsilon_{43}\} \geq 0$
$\mathbf{x}_5$	$[0, 0, 1]$	$[-\varepsilon_{51}, -\varepsilon_{52}, 1+\varepsilon_{53}]$	$\{\varepsilon_{51}, \varepsilon_{52}, \varepsilon_{53}\} \geq 0$
$\mathbf{x}_6$	$[0, 0, 1]$	$[-\varepsilon_{61}, -\varepsilon_{62}, 1+\varepsilon_{63}]$	$\{\varepsilon_{61}, \varepsilon_{62}, \varepsilon_{63}\} \geq 0$

As indicated in work [42], the hash code can also be regressed to its corresponding label, and (4) is also in accordance with such feasibility. By combing all the instances, (4) can be rewritten in the matrix representation

$$\min_{\mathbf{H},\mathbf{C}} \|\mathbf{Y} - \mathbf{C}^{\mathrm{T}}\mathbf{H}\|_F^2, \quad \text{s.t. } \mathbf{H} \in \{-1, 1\}^{q\times n}, \ \mathbf{C}^{\mathrm{T}}\mathbf{C} = \mathbf{I}_c. \tag{5}$$

The above formulation is a typical regression problem, which regress $\mathbf{H}$ to $\mathbf{Y}$. That is, the zero-one class label vectors stipulate a type of binary regression with target "1" for positive class and target "0" for the negative classes. Evidently, for the rigid zero-one label matrix $\mathbf{Y}$, the Euclidean distances of regression responses between samples from different classes are a constant value, i.e., $\sqrt{2}$ for single label data. This is contrary to the expectation that the samples from different classes should be as far as possible. To alleviate this problem, we propose to utilize the ε-dragging technique to force the regression targets of different classes moving along the opposite directions, whereby the margin between different classes can be enlarged. That is, with a positive slack variable ε_i, we hope the output will become $1+\varepsilon_i$ for the sample grouped into "1" and $-\varepsilon_i$ for the sample categorized into "0." As shown in Fig. 1, the label $\mathbf{y}_2 = (1, 0, 1)^{\mathrm{T}}$ is further relaxed as $\mathbf{y}_2 = (1, 0, 1+\varepsilon)^{\mathrm{T}}$, and the semantic vector $\vec{s_2}$ is updated to $\vec{s_2'}$. Accordingly, the resulted length of $\vec{e_2'}$ is smaller than the original one of $\vec{e_2}$, and the total quantization error is reduced by $|\vec{e_2}| - |\vec{e_2'}|$.

Further interpretation of ε-dragging motivation is shown in Table I, which reports six single label data points in three classes, and their one-hot class label vectors are listed in the second column. It can be observed that $\mathbf{x}_1$ and $\mathbf{x}_2$ are marked within the same class, while $\mathbf{x}_3$, $\mathbf{x}_4$, $\mathbf{x}_5$, and $\mathbf{x}_6$ are categorized into other different classes. Specifically, if the first components of the class label vectors are gathered, we can get values "1, 1, 0, 0, 0, and 0," and their values will be relaxed into "$1+\varepsilon_{11}$, $1+\varepsilon_{21}$, $-\varepsilon_{31}$, $-\varepsilon_{41}$, $-\varepsilon_{51}$, and $-\varepsilon_{61}$." As all ε values are nonnegative, such ε-dragging technique could help to enlarge the distance between different classes in case where the data points are mapped. It is noted that the real datasets may have large volume of data points and involve multiple semantic labels, and the proposed ε-dragging operation can be well utilized in these datasets as well. To develop a unique compact model, we utilize a constant matrix $\mathbf{B} \in \mathbb{R}^{c\times n}$ to characterize the dragging direction, in which the ith row and the jth column element $\mathbf{B}_{ij}$ are defined as

$$\mathbf{B}_{ij} = \begin{cases} +1, & \text{if } \mathbf{y}_{ij} = 1 \\ -1, & \text{otherwise} \end{cases} \tag{6}$$

where "+1" represents the positive direction and "−1" means the negative direction. Performing the above ε-dragging operation on each element of $\mathbf{Y}$ and recording these ε values by matrix $\mathbf{E} = \{\varepsilon_{ij} \geq 0\} \in \mathbb{R}^{c\times n}$, (5) can be rewritten the as following optimization problem:

$$\begin{aligned} \min_{\mathbf{Y},\mathbf{C},\mathbf{E}} \ & \|\mathbf{Y} + \mathbf{B} \odot \mathbf{E} - \mathbf{C}^{\mathrm{T}}\mathbf{H}\|_F^2 + \delta\|\mathbf{E}\|_F^2 \\ \text{s.t. } & \mathbf{H} \in \{-1, 1\}^{q\times n}, \quad \mathbf{C}^{\mathrm{T}}\mathbf{C} = \mathbf{I}_c \end{aligned} \tag{7}$$

where $\odot$ is a Hadamard product operator of matrices and δ is the weight coefficient to control the degree of relaxation. In contrast to (5), we add an ε-dragging term $\mathbf{B} \odot \mathbf{E}$ in (7) to enlarge the distances between different classes. In this way, each sample can be regressed efficiently with a large margin between the true and false classes. Accordingly, the learning model is converted to be an equivalently constrained optimization problem. For ease of representation, let $\bar{\mathbf{Y}} = \mathbf{Y} + \mathbf{B} \odot \mathbf{E}$; it can be easily found that the term $\min_{\mathbf{E}} \|\mathbf{E}\|_F^2$ is equivalent to $\min_{\bar{\mathbf{Y}}} \|\bar{\mathbf{Y}}\|_F^2$. Therefore, (5) can be further rewritten as

$$\begin{aligned} \min_{\mathbf{H},\mathbf{C},\bar{\mathbf{Y}}} \ & \|\bar{\mathbf{Y}} - \mathbf{C}^{\mathrm{T}}\mathbf{H}\|_F^2 + \delta\|\bar{\mathbf{Y}}\|_F^2 \\ \text{s.t. } & \mathbf{H} \in \{-1, 1\}^{q\times n}, \quad \mathbf{C}^{\mathrm{T}}\mathbf{C} = \mathbf{I}_c. \end{aligned} \tag{8}$$

2) Semantic Embedding Learning: On the one hand, any suitable embedding learning algorithms, linear or nonlinear, can be utilized for mapping the data into the semantic space. In general, the semantic correlation of multiple modalities often exists in the high-level space, and the mapping functions from the raw feature space to the high-level space are highly nonlinear [11], [43]. On the other hand, for hash representation learning, it is necessary to produce an efficient code, in which the variance of each bit is maximized and the bits are pairwise uncorrelated. To integrate these issues, we propose a simpler idea of orthogonally transforming the data and utilize following simple yet powerful nonlinear form:

$$\min_{\mathbf{R}_t} \ \left\|\mathbf{S} - \mathbf{R}_t^{\mathrm{T}}\phi(\mathbf{X}^t)\right\|_F^2, \quad \text{s.t. } \mathbf{R}_t^{\mathrm{T}}\mathbf{R}_t = \mathbf{I}_q \tag{9}$$

where $\mathbf{S} \in \mathbb{R}^{q\times n}$ represents the semantic data, and $\phi(\cdot)$ is the RBF kernel [43], which could better capture the underlying nonlinear property in feature space. Since the semantic data is approximated by orthogonally rotating label vector, the semantic subspace can be further approximated by $\mathbf{S} = \mathbf{C}\bar{\mathbf{Y}}$. Similar to the relationship between (3) and (5), (9) can be transformed into following equivalent form:

$$\min_{\mathbf{R}_t} \ \|\phi(\mathbf{X}^t) - \mathbf{R}_t\mathbf{C}\bar{\mathbf{Y}}\|_F^2, \quad \text{s.t. } \mathbf{R}_t^{\mathrm{T}}\mathbf{R}_t = \mathbf{I}_q. \tag{10}$$

3) Overall Objective Function: According to the orthogonal relationship, the first item $\min_{\mathbf{H},\mathbf{C}} \|\bar{\mathbf{Y}} - \mathbf{C}^{\mathrm{T}}\mathbf{H}\|_F^2$ in (8) is also equivalent to $\min_{\mathbf{H},\mathbf{C}} \|\mathbf{H} - \mathbf{C}\bar{\mathbf{Y}}\|_F^2$. By integrating the semantic-preserving learning and semantic embedding learning, the process of learning the discriminative hash codes can be conducted by minimizing the following objective function:

$$\begin{aligned} \min_{\mathbf{H},\mathbf{R}_1,\mathbf{R}_2,\mathbf{C},\bar{\mathbf{Y}}} \ & \|\mathbf{H} - \mathbf{C}\bar{\mathbf{Y}}\|_F^2 + \mu\|\phi(\mathbf{X}^1) - \mathbf{R}_1\mathbf{C}\bar{\mathbf{Y}}\|_F^2 \\ & + \theta\|\phi(\mathbf{X}^2) - \mathbf{R}_2\mathbf{C}\bar{\mathbf{Y}}\|_F^2 + \delta\|\bar{\mathbf{Y}}\|_F^2 \\ \text{s.t. } & \mathbf{C}^{\mathrm{T}}\mathbf{C} = \mathbf{I}_c, \quad \mathbf{H} \in \{-1, 1\}^{q\times n} \\ & \mathbf{R}_1^{\mathrm{T}}\mathbf{R}_1 = \mathbf{I}_q, \quad \mathbf{R}_2^{\mathrm{T}}\mathbf{R}_2 = \mathbf{I}_q. \end{aligned} \tag{11}$$

C. Discrete Optimization for FDDH

The discrete constraints imposed in (11) lead to mixed-integer optimization problems, which are generally NP-hard. In the literature, some hashing methods discard discrete constraints and solve a relaxed problem to simplify the optimization steps. Nevertheless, this relaxation strategy may accumulate large quantization error during the hash code learning process. To solve (11), the discrete optimization method is selected. For all matrix variables $\mathbf{H}$, $\mathbf{R}_1$, $\mathbf{R}_2$, $\mathbf{C}$, and $\bar{\mathbf{Y}}$, it is convex with respect to any single matrix variable while fixing the other ones, and an alternating optimization technique can be adopted to iteratively solve such optimization problem until the convergence is reached. The details of the discrete optimization steps are elaborated as follows.

1) Update $\mathbf{C}$*:* Remove the items that are irrelevant to $\mathbf{C}$, and fix $\mathbf{H}$, $\mathbf{R}_1$, $\mathbf{R}_2$, and $\bar{\mathbf{Y}}$. Then, the suboptimization problem derived in (11) is simplified as

$$\begin{aligned} \min_{\mathbf{C}} \ & \|\mathbf{H} - \mathbf{C}\bar{\mathbf{Y}}\|_F^2 + \mu\|\phi(\mathbf{X}^1) - \mathbf{R}_1\mathbf{C}\bar{\mathbf{Y}}\|_F^2 \\ & + \theta\|\phi(\mathbf{X}^2) - \mathbf{R}_2\mathbf{C}\bar{\mathbf{Y}}\|_F^2 \\ \text{s.t. } & \mathbf{C}^{\mathrm{T}}\mathbf{C} = \mathbf{I}_c. \end{aligned} \tag{12}$$

By expanding each item and removing the irrelevant ones, we can rewrite (12) as follows:

$$\begin{aligned} \min_{\mathbf{C}} \ & \|\mathbf{C}\bar{\mathbf{Y}}\|_F^2 + \mu\|\mathbf{R}_1\mathbf{C}\bar{\mathbf{Y}}\|_F^2 + \theta\|\mathbf{R}_2\mathbf{C}\bar{\mathbf{Y}}\|_F^2 \\ & - 2\mathrm{Tr}(\bar{\mathbf{Y}}(\mathbf{H}^{\mathrm{T}} + \mu\phi(\mathbf{X}^1)^{\mathrm{T}}\mathbf{R}_1 + \theta\phi(\mathbf{X}^2)^{\mathrm{T}}\mathbf{R}_2)\mathbf{C}) \\ \text{s.t. } & \mathbf{C}^{\mathrm{T}}\mathbf{C} = \mathbf{I}_c \end{aligned} \tag{13}$$

where $\mathrm{Tr}(\cdot)$ is the trace norm. Since $\mathbf{C}^{\mathrm{T}}\mathbf{C} = \mathbf{I}_c$, $\mathbf{R}_1^{\mathrm{T}}\mathbf{R}_1 = \mathbf{I}_q$, and $\mathbf{R}_2^{\mathrm{T}}\mathbf{R}_2 = \mathbf{I}_q$, it can be easily obtained that $\|\mathbf{C}\bar{\mathbf{Y}}\|_F^2 = \|\bar{\mathbf{Y}}\|_F^2$, $\|\mathbf{R}_1\mathbf{C}\bar{\mathbf{Y}}\|_F^2 = \|\bar{\mathbf{Y}}\|_F^2$, and $\|\mathbf{R}_2\mathbf{C}\bar{\mathbf{Y}}\|_F^2 = \|\bar{\mathbf{Y}}\|_F^2$, and all these values are constant. Therefore, the optimization problem in (13) is equal to maximize the following trace function:

$$\begin{aligned} \max_{\mathbf{C}} \ & \mathrm{Tr}(\bar{\mathbf{Y}}(\mathbf{H}^{\mathrm{T}} + \mu\phi(\mathbf{X}^1)^{\mathrm{T}}\mathbf{R}_1 + \theta\phi(\mathbf{X}^2)^{\mathrm{T}}\mathbf{R}_2)\mathbf{C}) \\ \text{s.t. } & \mathbf{C}^{\mathrm{T}}\mathbf{C} = \mathbf{I}_c. \end{aligned} \tag{14}$$

It is noted that the problem in (14) corresponds to the classic orthogonal procrustes problem [44], which can be approximated by singular value decomposition (SVD). For simplicity, let $\mathbf{Q} = \bar{\mathbf{Y}}(\mathbf{H}^{\mathrm{T}} + \mu\phi(\mathbf{X}^1)^{\mathrm{T}}\mathbf{R}_1 + \theta\phi(\mathbf{X}^2)^{\mathrm{T}}\mathbf{R}_2)$; we utilize SVD to decompose $\mathbf{Q}$, i.e., $\mathbf{Q} \approx \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\mathrm{T}}$, where $\boldsymbol{\Sigma} = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \boldsymbol{\sigma}_r)$, $r \leq \min(q, c)$, $\mathbf{U} \in \mathbb{R}^{c\times r}$, and $\mathbf{V} \in \mathbb{R}^{q\times r}$ are the transformation matrices. Let $\mathbf{Z} = \mathbf{V}^{\mathrm{T}}\mathbf{C}\mathbf{U}$, the following properties are obtained:

$$\begin{aligned} \mathrm{Tr}(\mathbf{Q}\mathbf{C}) &= \mathrm{Tr}(\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\mathrm{T}}\mathbf{C}) = \mathrm{Tr}(\mathbf{V}^{\mathrm{T}}\mathbf{C}\mathbf{U}\boldsymbol{\Sigma}) \\ &= \mathrm{Tr}(\mathbf{Z}\Sigma) = \sum_{i=1}^{r} z_{ii}\sigma_i \leq \sum_{i=1}^{r}\sigma_i. \end{aligned} \tag{15}$$

Evidently, the upper bound in (15) can be achieved if $\mathbf{Z} = \mathbf{I}_r$, whereby the optimal solution of $\mathbf{C}$ can be obtained by

$$\mathbf{C} = \mathbf{V}\mathbf{U}^{\mathrm{T}}. \tag{16}$$

In general, the length of hash code is often larger than the number of semantic categories, i.e., $q \geq c$, and it is reasonable to set $r = c$ during the learning process.

2) Update $\mathbf{R_1}$ *and* $\mathbf{R_2}$*:* Removing the items that are irrelevant to $\mathbf{R_1}$ and $\mathbf{R_2}$ and fix $\mathbf{C}$ and $\bar{\mathbf{Y}}$, (11) can be further simplified as

$$\min_{\mathbf{R}_1} \|\phi(\mathbf{X}^1) - \mathbf{R}_1\mathbf{C}\mathbf{Y}\|_F^2, \quad \text{s.t. } \mathbf{R}_1^{\mathrm{T}}\mathbf{R}_1 = \mathbf{I}_q \tag{17}$$

$$\min_{\mathbf{R}_2} \|\phi(\mathbf{X}^2) - \mathbf{R}_2\mathbf{C}\mathbf{Y}\|_F^2, \quad \text{s.t. } \mathbf{R}_2^{\mathrm{T}}\mathbf{R}_2 = \mathbf{I}_q. \tag{18}$$

Equivalently, the optimization problems in (17) and (18) are equal to maximize the following trace functions:

$$\max_{\mathbf{R}_1} \ \mathrm{Tr}(\mathbf{C}\bar{\mathbf{Y}}\phi(\mathbf{X}^1)^{\mathrm{T}}\mathbf{R}_1), \quad \text{s.t. } \mathbf{R}_1^{\mathrm{T}}\mathbf{R}_1 = \mathbf{I}_q \tag{19}$$

$$\max_{\mathbf{R}_2} \ \mathrm{Tr}(\mathbf{C}\bar{\mathbf{Y}}\phi(\mathbf{X}^2)^{\mathrm{T}}\mathbf{R}_2), \quad \text{s.t. } \mathbf{R}_2^{\mathrm{T}}\mathbf{R}_2 = \mathbf{I}_q. \tag{20}$$

Similarly, the optimal solutions of $\mathbf{R}_1$ and $\mathbf{R}_2$ can be approximated by respectively computing the SVD of $\mathbf{C}\bar{\mathbf{Y}}\phi(\mathbf{X}^1)^{\mathrm{T}}$ and $\mathbf{C}\bar{\mathbf{Y}}\phi(\mathbf{X}^2)^{\mathrm{T}}$, i.e., $\mathbf{C}\bar{\mathbf{Y}}\phi(\mathbf{X}^1)^{\mathrm{T}} \approx \mathbf{U}_1\mathbf{\Sigma}_1\mathbf{V}_1^{\mathrm{T}}$ and $\mathbf{C}\bar{\mathbf{Y}}\phi(\mathbf{X}^2)^{\mathrm{T}} \approx \mathbf{U}_2\mathbf{\Sigma}_2\mathbf{V}_2^{\mathrm{T}}$. By referring to (15) and (16), the solution of $\mathbf{R}_1$ and $\mathbf{R}_2$ can be achieved by

$$\mathbf{R}_1 = \mathbf{V}_1\mathbf{U}_1^{\mathrm{T}}, \quad \mathbf{R}_2 = \mathbf{V}_2\mathbf{U}_2^{\mathrm{T}}. \tag{21}$$

3) Update $\bar{\mathbf{Y}}$*:* Removing the items that are irrelevant to $\bar{\mathbf{Y}}$ and fixing $\mathbf{C}, \mathbf{H}, \mathbf{R}_1$, and $\mathbf{R}_2$, (11) can be rewritten as

$$\begin{aligned}\min_{\bar{\mathbf{Y}}} \ & (1+\mu+\theta+\delta)\mathrm{Tr}(\bar{\mathbf{Y}}^{\mathrm{T}}\bar{\mathbf{Y}}) - 2\mathrm{Tr}(\mathbf{H}^{\mathrm{T}}\mathbf{C}\bar{\mathbf{Y}}) \\ & - 2\mu\mathrm{Tr}(\phi(\mathbf{X}^1)^{\mathrm{T}}\mathbf{R}_1\mathbf{C}\bar{\mathbf{Y}}) - 2\theta\mathrm{Tr}(\phi(\mathbf{X}^2)^{\mathrm{T}}\mathbf{R}_2\mathbf{C}\bar{\mathbf{Y}}).\end{aligned} \tag{22}$$

Furthermore, the subproblem of (22) can be simplified as

$$\min_{\bar{\mathbf{Y}}} \ (1+\mu+\theta+\delta)\mathrm{Tr}(\bar{\mathbf{Y}}^{\mathrm{T}}\bar{\mathbf{Y}}) - 2\mathrm{Tr}(\mathbf{W}^{\mathrm{T}}\bar{\mathbf{Y}}) \tag{23}$$

where $\mathbf{W} = \mathbf{C}^{\mathrm{T}}(\mathbf{H} + \mu\mathbf{R}_1^{\mathrm{T}}\phi(\mathbf{X}^1) + \theta\mathbf{R}_2^{\mathrm{T}}\phi(\mathbf{X}^2))$. To solve such minimization problem, the gradient descent method is selected. More specifically, the derivative of all the terms in (23) with respect to $\bar{\mathbf{Y}}$ is derived, and its optimal value is attained at $\bar{\mathbf{Y}}' = (1/(1+\mu+\theta+\delta))\mathbf{W}$. It is noted that the original label matrix $\mathbf{Y}$ is now extended to be $\bar{\mathbf{Y}} = \mathbf{Y} + \mathbf{B} \odot \mathbf{E}$, where $\mathbf{E}$ is the ε-dragging matrix that utilized to enlarge the distances between different classes. Evidently, the updating result for the false class should be smaller than zero, and the regression result for the true class should be larger than one. Therefore, the updating scheme of optimized target matrix $\bar{\mathbf{Y}}$ is further regularized as

$$\bar{\mathbf{Y}}_{ij} = \begin{cases} \min(\bar{\mathbf{Y}}'_{ij},\ 0), & \text{if } \mathbf{y}_{ij} = 0 \\ \max(\bar{\mathbf{Y}}'_{ij},\ 1), & \text{if } \mathbf{y}_{ij} = 1. \end{cases} \tag{24}$$

4) Update $\mathbf{H}$*:* Removing the items that are irrelevant to $\mathbf{H}$ and fixing $\mathbf{C}$ and $\bar{\mathbf{Y}}$, (11) can be simplified as

$$\min_{\mathbf{H}} \|\mathbf{H} - \mathbf{C}\bar{\mathbf{Y}}\|_F^2, \quad \text{s.t. } \mathbf{H} \in \{-1, 1\}^{q\times n}. \tag{25}$$

The discrete solution of $\mathbf{H}$ can be computed from the embedding matrix $\mathbf{C}\bar{\mathbf{Y}}$, and an efficient close-form solution is approximated by thresholding such data matrix as

$$\mathbf{H} = \mathrm{sgn}(\mathbf{C}\bar{\mathbf{Y}}). \tag{26}$$

It is noted that the proposed FDDH approach has a closed-form solution for hash code learning and only requires a single step to obtain all bits, which is highly efficient in comparison with bit by bit learning scheme [10]. The main procedures of the proposed FDDH method are summarized in Algorithm 1.

Algorithm 1 Learning Algorithm for FDDH

Input: Training data $\mathbf{X}^1$, $\mathbf{X}^2$; code length q; semantic labels $\mathbf{L}$, parameters μ, θ, δ and γ.

1: Initialize $\mathbf{C}, \mathbf{R}_1, \mathbf{R}_2$ as random matrix respectively.
2: Initialize $\bar{\mathbf{Y}} = \mathbf{Y}$ and $\mathbf{H} \in \{-1, 1\}^{q\times n}$ randomly.
3: **repeat**
4: Update $\mathbf{C}$ via Eq (16);
5: Update $\mathbf{R}_1$ and $\mathbf{R}_2$ via (21);
6: Update $\mathbf{H}$ via Eq (25);
7: Update $\mathbf{Y}$ via Eq (24);
8: **until** convergency or reaching maximum iterations.

Output: Obtain hash code matrix $\mathbf{H}$ via Eq (26).

D. Out-of-Sample Extension

The hash function is designed to project high-dimensional real-value features to low-dimensional binary space. For any unseen query sample, it is straightforward to predict its hash codes via the modality-specific hash function. On the one hand, the off-line learning method is the standard way to learn such modality-specific hashing projections, which keeps unchanged for all new coming data. Similar to most existing methods, the off-line learning strategy is configured within the proposed FDDH framework. On the other hand, the multimedia data points often continuously arrive in a streaming fashion. If the training data are increasingly accumulated, the off-line learning method needs to recalculate the hash functions on the whole database, which is computationally inefficient. Therefore, it is particularly important to develop an efficient online learning strategy to deal with the new query data. In the following, we elaborate on the off-line strategy and the newly proposed online strategy in tandem.

1) Off-Line Strategy: As introduced in Section III-B, the discriminative hash codes can be well obtained by optimizing the objective function in (11). Since the linear hash function cannot characterize the nonlinearity embedded in real-world data, we refer to the training process and first utilize the RBF kernel to capture the underlying nonlinear information in feature space. Then, the modality-specific projection $\mathbf{P}_t$ can be obtained by minimizing the following formulation:

$$G_{\mathrm{offline}}(\mathbf{P}_t) = \|\mathbf{H} - \mathbf{P}_t\phi(\mathbf{X}^t)\|_F^2 + \gamma\|\mathbf{P}_t\|_F^2, \quad t = 1, 2 \tag{27}$$

where γ is the hyperparameter for the regularization term. The solution of $\mathbf{P}_t$ can be computed by a regularized linear regression method, and its optimal solution is obtained when the gradient of (27) is equal to zero

$$\mathbf{P}_t = \mathbf{H}\phi(\mathbf{X}^t)^{\mathrm{T}}(\phi(\mathbf{X}^t)\phi(\mathbf{X}^t)^{\mathrm{T}} + \gamma\mathbf{I})^{-1}, \quad t = 1, 2. \tag{28}$$

For any query sample $\mathbf{x}_i^t$ in the tth modality, the corresponding hash code $\mathbf{h}_i^t$ can be directly generated by

$$\mathbf{h}_i^t = \mathrm{sgn}(\mathbf{P}_t\phi(\mathbf{x}_i^t)). \tag{29}$$

2) Online Strategy: In practice, the multimedia data often come in a streaming fashion, and the hashing projection functions derived from the off-line strategy keep unchanged for all new data points. If the new data are increasingly accumulated,

the off-line learning method may accumulate large quantization errors during the hash code learning process. To alleviate this concern, we present a simple but effective online learning strategy to adaptively learn the modality-specific projections, by minimizing the following formulation:

$$G_{\text{online}}(\mathbf{P}_t) = G_{\text{offline}}(\mathbf{P}_t) + \left\|\mathbf{H}_s^t - \mathbf{P}_t\phi\left(\mathbf{X}_s^t\right)\right\|_F^2 \tag{30}$$

where $\mathbf{H}_s^t$ is the corresponding hash code matrix of the stream data $\mathbf{X}_s^t$. The optimal solution of $\mathbf{P}_t$ and $\mathbf{H}_s^t$ can be obtained when the gradients of (30) are equal to zeros. Accordingly, the modality-specific projection $\mathbf{P}_t$ can be computed by

$$\mathbf{P}_t = \left(\mathbf{H}\phi(\mathbf{X}^t)^{\mathrm{T}} + \mathbf{H}_s^t\phi\left(\mathbf{X}_s^t\right)^{\mathrm{T}}\right) \times \left(\phi(\mathbf{X}^t)\phi(\mathbf{X}^t)^{\mathrm{T}} + \phi\left(\mathbf{X}_s^t\right)\phi\left(\mathbf{X}_s^t\right)^{\mathrm{T}} + \gamma\,\mathbf{I}\right)^{-1}. \tag{31}$$

It is noted that the computation results of $\mathbf{H}\phi(\mathbf{X}^t)^{\mathrm{T}}$ and $\phi(\mathbf{X}^t)\phi(\mathbf{X}^t)^{\mathrm{T}}$ can be stored as constants during the learning process. For the tth modality of stream data $\mathbf{X}_s^t$, the corresponding hash codes $\mathbf{H}_s^t$ can be obtained by

$$\mathbf{H}_s^t = \operatorname{sgn}\left(\mathbf{P}_t\phi\left(\mathbf{X}_s^t\right)\right), \quad t = 1, 2. \tag{32}$$

The final solutions of $\mathbf{P}_t$ and $\mathbf{H}_s^t$ are obtained iteratively by repeating (31) and (32) until the procedure converges.

E. Theoretical Analysis

The proposed FDDH approach aims to produce discriminative semantic-preserving hash codes in a fast way while reducing the quantization error. This section shows the theoretical analysis to prove its effectiveness. In addition, the theoretical analysis of the online strategy is also given to prove its stability.

1) Efficiency of Semantic-Preserving: To facilitate the analysis of the relationship between the semantic information and hash code, we utilize single data point for illustration. According to (2) and (4), we can obtain the following relationship:

$$\mathbf{h}_i = \mathbf{C}\mathbf{y}_i + \mathbf{e}_i; \quad \mathbf{y}_i = \mathbf{C}^{\mathrm{T}}\mathbf{h}_i + \mathbf{e}_i' \tag{33}$$

where $\mathbf{y}_i$ denotes the label value for the ith sample, and $\mathbf{e}_i$ and $\mathbf{e}_i'$ denote the quantization error and the regression error, respectively. Accordingly, given two label vectors $\mathbf{y}_i$ and $\mathbf{y}_j$ from the semantic space and their corresponding hash representations $\mathbf{h}_i$ and $\mathbf{h}_j$, we can obtain the following equations:

$$\begin{aligned} \|\mathbf{h}_i - \mathbf{h}_j\|_2 &= \|\mathbf{C}(\mathbf{y}_i - \mathbf{y}_j) + (\mathbf{e}_i - \mathbf{e}_j)\|_2 \\ \left\|\mathbf{y}_i - \mathbf{y}_j - \left(\mathbf{e}_i' - \mathbf{e}_j'\right)\right\|_2 &= \|\mathbf{C}^{\mathrm{T}}(\mathbf{h}_i - \mathbf{h}_j)\|_2. \end{aligned} \tag{34}$$

For any two vectors $\mathbf{u}$ and $\mathbf{v}$, there are two norm properties $\|\mathbf{u}\mathbf{v}\|_2 \le \|\mathbf{u}\|_2\|\mathbf{v}\|_2$ and $\|\mathbf{u}+\mathbf{v}\|_2 \le \|\mathbf{u}\|_2 + \|\mathbf{v}\|_2$, and the similar properties can also be found in matrix representation. According to the definition of the Frobenius norm, we can further obtain that $\|\mathbf{C}^{\mathrm{T}}\|_F = \|\mathbf{C}\|_F = \kappa$, where κ is a constant. According to the norm property, we can find $\|\mathbf{C}\|_2 \le \|\mathbf{C}\|_F$ and obtain the following semantic structure preservation property:

$$\frac{1}{\kappa}\|\mathbf{y}_i - \mathbf{y}_j\|_2 - \epsilon_2(i,j) \ll \|\mathbf{h}_i - \mathbf{h}_j\|_2 \ll \kappa\|\mathbf{y}_i - \mathbf{y}_j\|_2 + \epsilon_1(i,j) \tag{35}$$

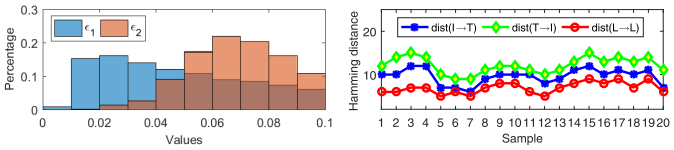

Fig. 2. Left: histogram distribution of bound errors. Right: Hamming distances for semantic preserving illustration.

where $\epsilon_1(i,j) = \|\mathbf{e}_i - \mathbf{e}_j\|_2$ and $\epsilon_2(i,j) = (1/\kappa)\|\mathbf{e}_i' - \mathbf{e}_j'\|_2$. Remarkably, if the bound error terms ϵ_1 and ϵ_2 are removed, it is not difficult to find that the relationship in (35) is in accordance with the bi-Lipschitz continuity. Equivalently, if the quantization error $(\mathbf{e}_i, \mathbf{e}_j)$ and the regression error $(\mathbf{e}_i', \mathbf{e}_j')$ are reduced perfectly, the error terms ϵ_1 and ϵ_2 shall closely equate to 0, whereby (35) approximates the relationship of the bi-Lipschitz continuity. That is, the smaller error bounds guarantee that $\mathbf{h}_i$ is similar to $\mathbf{h}_j$ when $\mathbf{y}_i$ is similar to $\mathbf{y}_j$. Remarkably, minimizing the objective function in (3) and (5) can well reduce $\mathbf{e}_i$ and $\mathbf{e}_i'$ because they aim to find $\mathbf{h}_i$ with small quantization error and regression error. Therefore, the semantic consistency between the semantic information and hash code is well preserved within the FDDH framework.

Next, we investigate the value distributions concerning to the error terms. Specifically, 5000 instances are randomly selected from the MIRFlickr dataset to compute $\mathbf{e}_i$ and $\mathbf{e}_i'$ by (33). In order to eliminate the influence of magnitude of input data pairs, we refer to the magnitude of $\|\mathbf{y}_i - \mathbf{y}_j\|_2$ and, respectively, compute the normalized relative errors of $|\epsilon_1(i,j)|/\kappa\|\mathbf{y}_i - \mathbf{y}_j\|_2$ and $\kappa|\epsilon_2(i,j)|/\|\mathbf{y}_i - \mathbf{y}_j\|_2$, which can be efficiently utilized to show the impacts of the error terms resulted by (35). As shown in the left part of Fig. 2, we draw the value histograms of error terms and record their number proportions among the 25M data pairs. It can be observed that all the data errors ϵ_1 and ϵ_2 fall into the small range $[0, 0.1]$, which means that these errors terms have very little impacts to the right-hand side of (35). In addition, we compute the Hamming distances (32 bits) between one randomly selected instance and other 20 different data instances. As shown in the right part of Fig. 2, it can be clearly observed that the Hamming distances of Image to Text (I $\rightarrow$ T), Text to Image (T $\rightarrow$ I), and Label to Label (L $\rightarrow$ L) often have a similar tendency, which indicates that the hash codes derived from the proposed FDDH framework can well hold the semantic-preserving property between heterogeneous modalities.

2) Stability of Online Strategy: We further discuss another potential benefit of the newly proposed online hash function learning strategy, which stabilizes the hash code generation for out-of-sample extensions. In a stable algorithm, the output hash codes do not change significantly if a training example is replaced with an independent and identically distributed (i.i.d.) one. According to (32), the hash code learning for new data is closely related to the modality-specific projections. More specifically, let $\mathbf{X}_s^t$ be the new streaming dataset of the tth modality and $\mathbf{X}_s^{t/i}$ be the dataset with the ith example in $\mathbf{X}_s^t$ replaced with an i.i.d. one; the proposed online hash projection

strategy holds the following stable property:

$$\left\|\mathbf{P}_t\left(\mathbf{X}_s^t\right)-\mathbf{P}_t\left(\mathbf{X}_s^{t/i}\right)\right\|_F \leq \boldsymbol{\Theta}(n) \tag{36}$$

where $\boldsymbol{\Theta}(n)$ converges to zero as the sample size n goes to infinity. In supplementary material, we provide its detailed proof.

3) Efficiency of Complexity Analysis: The computational complexity of FDDH mainly involves RBF mapping and the optimization in (11). For RBF mapping, whose complexity is $\mathcal{O}(m^2+kdn)$, where $d=\max(d_1,d_2)$, m is the number of instances selected to compute the kernel width, and k is the number of anchor points. For optimization in (11), whose complexity is $\mathcal{O}(n(q+c+qd+qc+d^2+q^2)+q^3+qd^2+d^3)$, since $c \leq q < d \ll n$, the optimization complexity can be simplified as $\mathcal{O}(n(q+c+d^2)+d^3)$. Let t be the iterative number to converge; the overall complexity is approximated as $\mathrm{O}(m^2+kdn+((q+c+d^2)n+d^3)t)$, which is linear to n, and it is very competitive to existing methods. In practice, the iteration number is always less than 15, and more illustrations about the learning speed will be discussed in Section IV.

IV. Experiments

A. Experimental Settings

1) Datasets: The popular multimodal PASCAL-VOC-2007 [45], MIRFlickr [46], and NUS-WIDE [47] datasets are selected for evaluation. Similar to [16], the PASCAL-VOC-2007 dataset (abbreviated as PASCAL-VOC) is divided into train, val, and test subsets. By dropping those pairs without text annotation, we conduct experiments on trainval and test splits, which, respectively, contain 5000 and 4919 pairs. For MIRFlickr, we keep 20015 image–text pairs whose textual tags appear more than 20 times and randomly select 2000 instances as a query set while choosing the rest as the training set. For the NUS-WIDE dataset, we select 186 577 annotated instances from the top-ten most frequent concepts to guarantee that each concept has abundant training samples and randomly select 1866 instances as a query set. For these three datasets, the image is respectively represented as the 512-d GIST feature vector, the 512-d SIFT feature vector, and the 500-d BoW feature vector, and the text is characterized by the 399-d, 1386-d, and 1000-d BoW feature vectors. These handcrafted features are very useful for evaluating the effectiveness of different learning algorithms. In addition, the recent convolutional neural network (CNN) has been popularized for visual feature extraction, and we also extract 4096-d CNN visual features from the last fully connected layer of the classic VGG19 model [48], [49].

2) Baseline Methods: For meaningful comparisons, the state-of-the-art unsupervised methods (i.e., CMFH [28], CCQ [6], and IMH [3]) and supervised methods (SePH [11], GSePH [9], DCH [10], DLFH [30], SCRATCH [35], SRLCH [36], and BATCH [37]) are selected for evaluation. It is noted that the recent deep cross-modal hashing methods [38]–[40] jointly learn the high-level feature representations and hash code in an integrated way, and the proposed framework is totally different from those works. In that sense, it is really difficult to perform a relatively fair and meaningful comparison with these approaches appropriately. In spite of such differences, we also select one representative deep cross-modal hashing method (DCMH [15]) for comparison. As CMFH [28], SePH [11], and GSePH [9] methods are computationally expensive in the training process, it is difficult to learn their corresponding hash functions on the whole NUS-WIDE dataset. Following the training strategy adopted in the literature [10], [11], we randomly select 10 000 instances from its retrieval set to learn the hash functions in the training process and then utilize the learned hash functions to generate binary codes for all instances in the dataset. For all the baselines, we utilize the source codes and initialize the relevant parameters kindly provided by the respective authors.

3) Evaluation Metric: The goal of cross-modal hashing is to index the relevant neighbors from the database of another modality, and the relevant instances are defined as those sharing at least one semantic label with the query. Similar to most works [9], [10], mean average precision (mAP), top-K precision, and precision–recall curves are selected for quantitative analysis, including retrieving text with given image (I $\rightarrow$ T) and retrieving image with given text (T $\rightarrow$ I). In general, the larger mAP scores and top-K precision values often indicate the better retrieval performance. In the experiments, parameters μ, θ, and δ are empirically set at $\{10^0, 10^{-3}, 10^3\}$, $\{10^{-2}, 10^{-3}, 10^3\}$, and $\{10^{-3}, 10^{-3}, 10^3\}$, respectively, for the PASCAL-VOC, MIRFlickr, and NUS-WIDE multimodal datasets.

B. Results and Discussion

1) Results of Retrieval Performances: The mAP scores and top-50 precision values tested with different datasets are summarized in Tables II–IV, respectively. For handcrafted features, it can be observed that the proposed FDDH approach has achieved very promising cross-modal retrieval performances in different hash length settings and outperforms most baselines. For instance, the proposed FDDH approach has delivered much better retrieval performances than that generated by unsupervised methods, i.e., CMFH, CCQ, and IMH, and also yielded comparable or even the better retrieval performances than that generated by the competing supervised methods, (i.e., SePH_km, GSePH_km, DCH, DLFH, SCRATCH, SRLCH, and BATCH). The main reason lies in that those unsupervised methods intuitively learn the hash codes from the original feature space to the Hamming space, and the hash codes learned in an unsupervised way are not discriminative enough. As a result, the corresponding semantic similarity is not well preserved in the Hamming space, and the relevant retrieval performances are a bit poor. In contrast to this, the supervised cross-modal hashing methods often deliver better retrieval performances, and the proposed FDDH method always yields the highest mAP scores in most cases and delivers the best top-50 precisions in all hash lengths. For instance, the mAP scores obtained by FDDH (32 bits on T $\rightarrow$ I) reach up to 0.9048, 0.8022, and 0.8133, respectively, evaluated on the PASCAL-VOC, MIRFlickr, and NUS-WIDE datasets. Specifically, DLFH [30] proposes a novel discrete latent factor model to learn the binary hash codes without continuous relaxation, which performs well on some T $\rightarrow$ I retrieval task,

TABLE II

QUANTITATIVE COMPARISONS OF MAP AND TOP-50 PRECISION ON PASCAL-VOC-2007, AND THE BEST RESULTS ARE HIGHLIGHTED IN BOLD

Method/Dataset		Handcrafted Features						CNN Visual Features					
		mAP			Top-50 precision			mAP			Top-50 precision		
		32 bits	64 bits	128 bits	32 bits	64 bits	128 bits	32 bits	64 bits	128 bits	32 bits	64 bits	128 bits
I→T	CMFH [28]	0.3337	0.3270	0.3244	0.4309	0.4340	0.4388	0.4406	0.4645	0.4799	0.6390	0.6664	0.6913
	CCQ [6]	0.3100	0.2987	0.3062	0.3997	0.3663	0.4062	0.3760	0.3899	0.4194	0.4910	0.5160	0.5685
	IMH [3]	0.2888	0.2759	0.2653	0.4021	0.3875	0.3697	0.3543	0.3174	0.2948	0.5505	0.5209	0.4921
	SePH_km [11]	0.5274	0.5375	0.5480	0.5683	0.5820	0.5992	0.7233	0.7265	0.7348	0.8215	0.8305	0.8382
	GSePH_km [9]	0.5468	0.5612	0.5672	0.5973	0.6136	0.6107	0.7582	0.7746	0.7841	0.8404	0.8466	0.8499
	DCH [10]	0.4287	0.4300	0.4238	0.4684	0.4720	0.4780	0.4120	0.4032	0.3874	0.4688	0.4661	0.4741
	DLFH [30]	0.4433	0.4710	0.4858	0.5223	0.5519	0.5662	0.4330	0.4707	0.4927	0.5072	0.5575	0.5865
	SCRATCH [35]	0.4869	0.4984	0.5000	0.5442	0.5508	0.5424	0.7092	0.7097	0.7067	0.8168	0.8157	0.8210
	SRLCH [36]	0.4827	0.4776	0.4966	0.5994	0.5977	0.6212	0.6892	0.6802	0.7229	0.8440	0.8433	0.8607
	BATCH [37]	0.5530	0.5640	0.5801	0.6108	0.6203	0.6256	0.7450	0.7576	0.7726	0.8302	0.8536	0.8588
	DCMH [15]	–	–	–	–	–	–	0.5248	0.5514	0.5899	0.6408	0.6645	0.7298
	FDDH	**0.5615**	**0.5728**	**0.5832**	**0.6231**	**0.6261**	**0.6306**	**0.7722**	**0.7970**	**0.8029**	**0.8508**	**0.8604**	**0.8615**
T→I	CMFH [28]	0.3369	0.3380	0.3364	0.5423	0.5735	0.5755	0.4142	0.4364	0.4547	0.6810	0.7194	0.7420
	CCQ [6]	0.2952	0.3123	0.3072	0.4307	0.4524	0.4681	0.4622	0.4864	0.5092	0.7214	0.7677	0.7828
	IMH [3]	0.4385	0.4009	0.3630	0.7406	0.7185	0.6712	0.4777	0.4010	0.3561	0.7961	0.7352	0.6700
	SePH_km [11]	0.8077	0.8183	0.8319	0.9381	0.9479	0.9537	0.8253	0.8248	0.8369	0.9440	0.9504	0.9554
	GSePH_km [9]	0.8631	0.8842	0.8939	0.9534	0.9577	0.9582	0.8751	0.8928	0.8999	0.9544	0.9576	0.9583
	DCH [10]	0.8160	0.8288	0.8213	0.9188	0.9290	0.9299	0.8094	0.8243	0.8158	0.9177	0.9250	0.9290
	DLFH [30]	0.7607	0.8175	0.8418	0.8625	0.9210	0.9444	0.7672	0.8257	0.8405	0.8726	0.9309	0.9462
	SCRATCH [35]	0.8278	0.8371	0.8233	0.9437	0.9435	0.9419	0.8309	0.8208	0.8076	0.9445	0.9417	0.9426
	SRLCH [36]	0.8085	0.7827	0.8274	0.9589	0.9559	0.9607	0.8015	0.7840	0.8320	0.9579	0.9573	0.9608
	BATCH [37]	0.8569	0.8763	0.8902	0.9507	0.9569	0.9592	0.8645	0.8748	0.8881	0.9517	0.9590	0.9563
	DCMH [15]	–	–	–	–	–	–	0.5477	0.5818	0.6345	0.6604	0.6938	0.7839
	FDDH	**0.9048**	**0.9182**	**0.9275**	**0.9589**	**0.9585**	**0.9606**	**0.9002**	**0.9216**	**0.9258**	**0.9573**	**0.9619**	**0.9596**

TABLE III

QUANTITATIVE COMPARISONS OF MAP AND TOP-50 PRECISION ON MIRFLICKR, AND THE BEST RESULTS ARE HIGHLIGHTED IN BOLD

Method/Dataset		Handcrafted Features						CNN Visual Features					
		mAP			Top-50 precision			mAP			Top-50 precision		
		32 bits	64 bits	128 bits	32 bits	64 bits	128 bits	32 bits	64 bits	128 bits	32 bits	64 bits	128 bits
I→T	CMFH [28]	0.5722	0.5582	0.5581	0.6361	0.5924	0.5879	0.5650	0.5650	0.5652	0.5970	0.5973	0.5980
	CCQ [6]	0.5848	0.5850	0.5812	0.6662	0.6721	0.6635	0.6560	0.6588	0.6645	0.8049	0.8255	0.8364
	IMH [3]	0.5718	0.5685	0.5650	0.6383	0.6321	0.6288	0.6134	0.5994	0.5878	0.7831	0.7791	0.7647
	SePH_km [11]	0.6679	0.6713	0.6710	0.7240	0.7344	0.7312	0.7885	0.7913	0.7949	0.8915	0.8980	0.9051
	GSePH_km [9]	0.6570	0.6649	0.6694	0.7204	0.7325	0.7391	0.7879	0.8002	0.8035	0.8960	0.9045	0.9073
	DCH [10]	0.6906	0.7017	0.6974	0.7523	0.7523	0.7598	0.7710	0.7919	0.7798	0.8928	0.8942	0.8983
	DLFH [30]	0.7194	0.7284	0.7325	0.7239	0.7433	0.7478	0.8259	0.8414	0.8464	0.8836	0.8965	0.8963
	SCRATCH [35]	0.7109	0.7259	0.7282	0.8039	0.8156	0.8289	0.8126	0.8259	0.8319	0.9261	0.9173	0.9190
	SRLCH [36]	0.6341	0.6263	0.6525	0.7572	0.7806	0.8117	0.7172	0.7367	0.7447	0.9101	0.9234	0.9183
	BATCH [37]	0.7316	0.7403	0.7511	0.8309	0.8326	0.8340	0.8345	0.8454	0.8752	0.9397	0.9486	0.9407
	DCMH [15]	–	–	–	–	–	–	0.7246	0.7306	0.7420	0.8355	0.8573	0.8866
	FDDH	**0.7392**	**0.7578**	**0.7631**	**0.9074**	**0.9127**	**0.9256**	**0.8515**	**0.8741**	**0.8843**	**0.9537**	**0.9590**	**0.9607**
T→I	CMFH [28]	0.5718	0.5562	0.5560	0.6231	0.5891	0.5960	0.5627	0.5625	0.5626	0.5963	0.5967	0.5962
	CCQ [6]	0.5782	0.5799	0.5748	0.6248	0.6501	0.6424	0.6553	0.6556	0.6617	0.8105	0.8203	0.8333
	IMH [3]	0.5710	0.5685	0.5651	0.6441	0.6422	0.6350	0.6189	0.6041	0.5919	0.8093	0.8124	0.7962
	SePH_km [11]	0.7102	0.7158	0.7206	0.8122	0.8295	0.8392	0.7533	0.7550	0.7584	0.8483	0.8586	0.8603
	GSePH_km [9]	0.7004	0.7100	0.7166	0.8210	0.8289	0.8405	0.7440	0.7557	0.7614	0.8450	0.8500	0.8581
	DCH [10]	0.7760	0.7963	0.7923	0.8779	0.8787	0.8914	0.7746	0.7946	0.7864	0.8743	0.8789	0.8924
	DLFH [30]	0.8161	**0.8267**	0.8347	0.8659	0.8603	0.8713	**0.8150**	**0.8292**	0.8340	0.8610	0.8668	0.8608
	SCRATCH [35]	0.7694	0.7855	0.7896	0.8886	0.8982	0.9032	0.7783	0.7889	0.7939	0.8883	0.8734	0.8772
	SRLCH [36]	0.6747	0.6635	0.6957	0.8517	0.8545	0.9157	0.6755	0.6827	0.6908	0.8608	0.8993	0.8918
	BATCH [37]	0.8037	0.8156	0.8297	0.9131	0.9080	0.9191	0.8097	0.8218	0.8304	0.8858	0.9177	0.9127
	DCMH [15]	–	–	–	–	–	–	0.7660	0.7735	0.7769	0.8760	0.8879	0.8934
	FDDH	**0.8022**	0.8250	**0.8357**	**0.9485**	**0.9521**	**0.9588**	0.8125	0.8247	**0.8402**	**0.9495**	**0.9566**	**0.9578**

mainly tested on the MIRFlickr and NUS-WIDE datasets. Comparatively speaking, the proposed FDDH approach outperforms the DLFH method in most cases. It is noted that the top-50 precision values obtained by the proposed FDDH approach are all higher than that produced by the DLFH method, which indicates that the proposed FDDH approach is able to search much more similar samples at the top-ranked instances. That is, the hash codes derived from the proposed framework are more discriminative and semantically meaningful, which can well guarantee the semantic consistency between the data and its semantic representation for better cross-modal retrieval.

For the CNN visual features, it can also be found that almost all competing baselines yield the improved retrieval

TABLE IV
QUANTITATIVE COMPARISONS OF MAP AND TOP-50 PRECISION ON THE NUS-WIDE DATASET, AND THE BEST RESULTS ARE HIGHLIGHTED IN BOLD

Method/Dataset		Handcrafted Features						CNN Visual Features					
		mAP			Top-50 precision			mAP			Top-50 precision		
		32 bits	64 bits	128 bits	32 bits	64 bits	128 bits	32 bits	64 bits	128 bits	32 bits	64 bits	128 bits
I→T	CMFH [28]	0.3429	0.3433	0.3431	0.4201	0.4235	0.4240	0.3462	0.3463	0.3463	0.4159	0.4159	0.4181
	CCQ [6]	0.3867	0.3928	0.3944	0.5200	0.5283	0.5313	0.4913	0.4967	0.5062	0.6878	0.7148	0.7379
	IMH [3]	0.3738	0.3609	0.3548	0.5086	0.4770	0.4632	0.4347	0.4011	0.3865	0.6588	0.6660	0.6576
	SePH_km [11]	0.5628	0.5752	0.5811	0.5908	0.6115	0.6217	0.7402	0.7511	0.7575	0.8128	0.8257	0.8340
	GSePH_km [9]	0.5565	0.5710	0.5769	0.6045	0.6129	0.6202	0.7445	0.7561	0.7634	0.8286	0.8333	0.8374
	DCH [10]	0.6398	0.6443	0.6626	0.6633	0.6209	0.6515	0.7829	0.8172	0.8404	0.8609	0.8541	0.8661
	DLFH [30]	0.6636	0.6768	0.6849	0.7752	0.7697	0.7785	0.8332	0.8548	0.8624	0.9086	0.9265	0.9298
	SCRATCH [35]	0.6649	0.6674	0.6747	0.7382	0.7406	0.7695	0.8081	0.8197	0.8277	0.8872	0.8880	0.8908
	SRLCH [36]	0.6144	0.6202	0.6411	0.7899	0.8032	0.7848	0.7850	0.7830	0.8073	0.9123	0.9153	0.9239
	BATCH [37]	0.6820	0.6908	0.7019	0.8048	0.8026	0.8158	0.8298	0.8368	0.8432	0.9169	0.9219	0.9205
	DCMH [15]	–	–	–	–	–	–	0.6344	0.6389	0.6514	0.7270	0.7456	0.7755
	FDDH	**0.6970**	**0.6910**	**0.7118**	**0.8315**	**0.8288**	**0.8596**	**0.8452**	**0.8578**	**0.8689**	**0.9237**	**0.9391**	**0.9359**
T→I	CMFH [28]	0.3418	0.3422	0.3421	0.4101	0.4116	0.4107	0.3439	0.3440	0.3442	0.4305	0.4359	0.4415
	CCQ [6]	0.3881	0.3945	0.3976	0.5345	0.5520	0.5564	0.5016	0.5087	0.5195	0.7080	0.7336	0.7415
	IMH [3]	0.3705	0.3605	0.3530	0.5224	0.5017	0.4698	0.4405	0.4033	0.3884	0.7152	0.7048	0.7012
	SePH_km [11]	0.6670	0.6738	0.6705	0.7528	0.7685	0.7701	0.7045	0.7138	0.7226	0.7663	0.7760	0.7894
	GSePH_km [9]	0.6523	0.6700	0.6776	0.7632	0.7623	0.7792	0.7026	0.7205	0.7222	0.7732	0.7876	0.7883
	DCH [10]	0.7822	0.8018	0.8172	0.8339	0.8185	0.8219	0.7597	0.7936	0.8121	0.8211	0.8105	0.8287
	DLFH [30]	0.8032	**0.8228**	**0.8265**	0.8927	0.8973	0.8934	0.7987	**0.8217**	**0.8258**	0.8750	0.8851	0.8855
	SCRATCH [35]	0.7879	0.7946	0.8013	0.8520	0.8620	0.8655	0.7830	0.7928	0.7928	0.8616	0.8589	0.8490
	SRLCH [36]	0.7383	0.7465	0.7692	0.8827	0.8744	0.8694	0.7484	0.7504	0.7666	0.8853	0.8794	0.8732
	BATCH [37]	0.7949	0.8099	0.8105	0.8892	0.8962	0.9062	0.7945	0.8035	0.8073	0.8876	0.8897	0.8896
	DCMH [15]	–	–	–	–	–	–	0.6722	0.6769	0.6863	0.7251	0.7389	0.7609
	FDDH	**0.8133**	0.8111	0.8244	**0.9024**	**0.9024**	**0.9234**	**0.8054**	0.8173	0.8167	**0.8923**	**0.9130**	**0.9143**

performances over the handcrafted features in most cases. Accordingly, the proposed FDDH approach often boosts the retrieval performances in different hash length settings and significantly outperforms most state-of-the-art baselines. For instance, if the hash length is set at 64 bits, the mAP scores obtained by FDDH and tested on I → T task are higher than 0.79, 0.87, and 0.85, respectively, evaluated on the PASCAL-VOC, MIRFlickr, and NUS-WIDE datasets. Remarkably, the proposed FDDH method is designed to explicitly learn the discriminative semantic-preserving hash codes, which can achieve very competitive and even better performances compared to the deep leaning method, i.e., DCMH. For instance, the mAP scores obtained by FDDH are higher than the results generated by DCMH in all cases. For some cases, the handcrafted features embedded within the proposed FDDH approach yield better performance than the CNN visual features. The possible reasons are twofold: 1) the handcrafted features associated with the semantic supervision, orthogonal regression, and ε-dragging operation are able to produce more discriminative hash codes for better retrieval performances and 2) the mapping functions from the raw feature space to the high-level semantic space are highly nonlinear, and the handcrafted features associated with RBF mapping are able to capture the underlying nonlinear information within the visual data and, therefore, perform comparably with the CNN visual features.

Furthermore, the precision–recall curves and top-K precision curves tested on different feature representations are shown in Figs. 3 and 4, respectively. On the one hand, the precision–recall curves show that the proposed FDDH approach has achieved the comparable cross-modal retrieval performances in different hash length settings and outperformed most baselines. On the other hand, top-K precision indicates the change of precision with respect to the number of top-ranked K instances exhibited to the users. As shown in Fig. 4, the top-K precision curves indicate that the proposed FDDH method always yields the highest precision scores than the baselines with the number of retrieved instance (K) changes, both in handcrafted visual features and CNN visual features. This indicates that the proposed FDDH approach is able to index much more similar samples at the beginning, which is very important for building a practical retrieval system. The main superiority contributed to these very competitive performances lies in that the hash codes derived from FDDH are more discriminative and interpretable to characterize the heterogeneous data samples while faithfully preserving both intramodality similarity and intermodality similarity.

2) Results of Ablation Studies: Within the proposed FDDH framework, the label relaxing and RBF mapping schemes are carefully considered for efficient cross-modal hashing. Next, we further evaluate the effectiveness of each learning module and heuristically validate the performance of different learning modules, i.e., FDDH without label relaxation (FDDH_NR) and its further extension without RBF mapping (FDDH_NRM). To be specific, their main objective formulations, simplified directly from (11), are denoted as $\|\mathbf{H}-\mathbf{CY}\|_F^2+\mu\|\phi(\mathbf{X}^1)-\mathbf{R}_1\mathbf{CY}\|_F^2+\theta\|\phi(\mathbf{X}^2)-\mathbf{R}_2\mathbf{CY}\|_F^2$ and $\|\mathbf{H}-\mathbf{CY}\|_F^2+\mu\|\mathbf{X}^1-\mathbf{R}_1\mathbf{CY}\|_F^2+\theta\|\mathbf{X}^2-\mathbf{R}_2\mathbf{CY}\|_F^2$, respectively. Accordingly, the mean mAP scores (m-mAP) and mean top-50 precision (m-top50) values, averaged on I → T and T → I tasks with all hash bits (i.e., 32, 64, and 128), are recorded to validate these different learning mechanisms.

As illustrated in Table V, it can be found that the m-mAP scores and m-top50 values attained by FDDH_NR and FDDH_NRM have also delivered very competitive performances. On the one hand, the reasonable relaxation of label

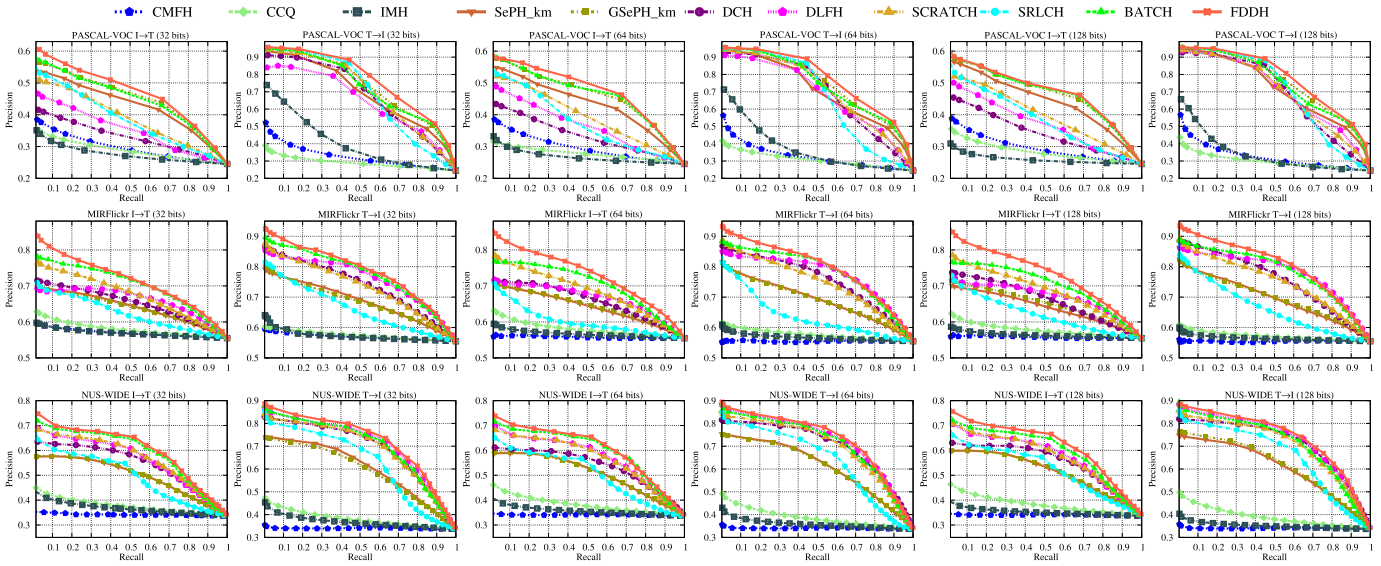

Fig. 3. Precision–recall curves obtained by different approaches and tested on different datasets.

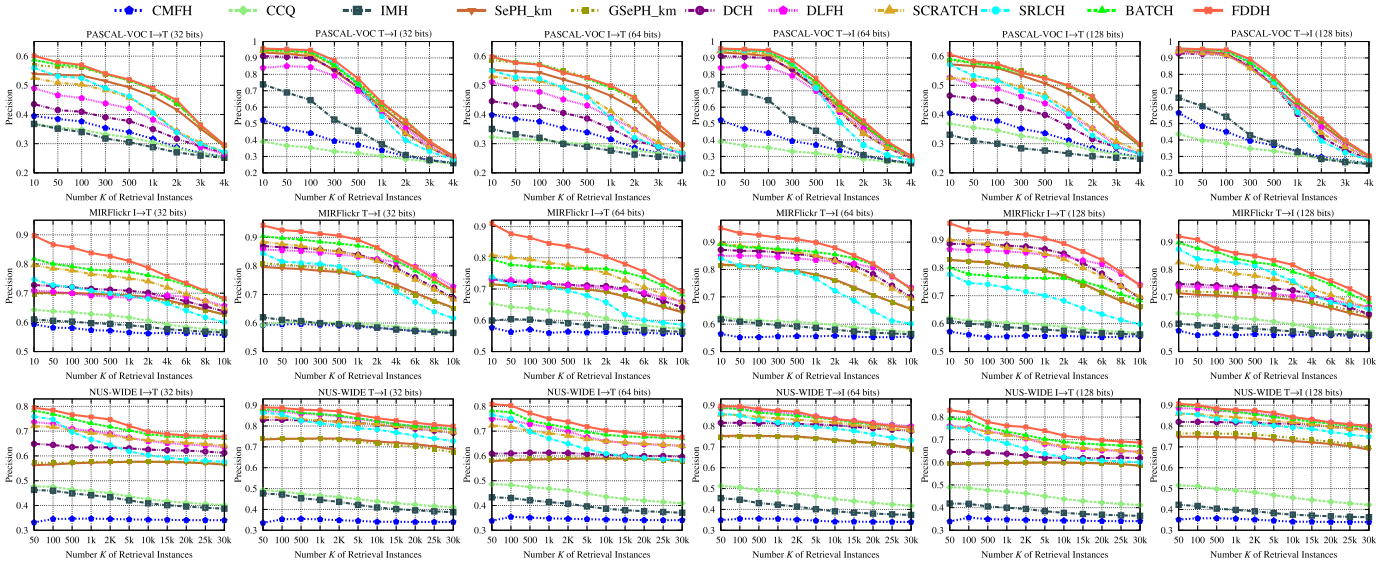

Fig. 4. Representative top-K precision curves obtained by different approaches and tested on handcrafted features.

TABLE V
ABLATION STUDIES OF FDDH ON THE VOC-PASCAL DATASET

Dataset	Method	Handcrafted feature		CNN visual feature	
		m-MAP	m-top50	m-MAP	m-top50
PASCAL-VOC	FDDH_NR	0.7440	0.7911	0.8410	0.9003
	FDDH_NRM	0.6559	0.7092	0.6506	0.7142
	FDDH	**0.7447**	**0.7930**	**0.8533**	**0.9086**
MIRFlickr	FDDH_NR	0.7463	0.8693	0.8110	0.9212
	FDDH_NRM	0.7396	0.8535	0.8047	0.9100
	FDDH	**0.7872**	**0.9342**	**0.8479**	**0.9562**
NUS-WIDE	FDDH_NR	0.7521	0.8698	**0.8364**	0.9186
	FDDH_NRM	0.7378	0.7663	0.8267	0.8676
	FDDH	**0.7581**	**0.8747**	0.8352	**0.9197**

values is able to offer a large class margin for discriminative analysis, which can promote the discriminative power of hash codes. On the other hand, the utilization of RBF mapping could capture the nonlinear structure of input data to improve retrieval performance. Remarkably, the m-mAP scores obtained by FDDH are higher than that produced by FDDH_NR and FDDH_NRM in all cases, while the m-top50 values generated by FDDH yield the best retrieval precisions. That is, the integration of relaxed label value learning and RBF mapping could yield more discriminative hash codes and, therefore, significantly boost the retrieval performance.

3) Results of Training Time: The computational complexity of the proposed FDDH framework mainly accumulates from the training process, in which the processing time of RBF kernel mapping is a constant during the learning process. For a fair comparison, we evaluate the competing algorithms on the NUS-WIDE dataset with handcrafted features and record the execution time of different training sizes with 128 hash

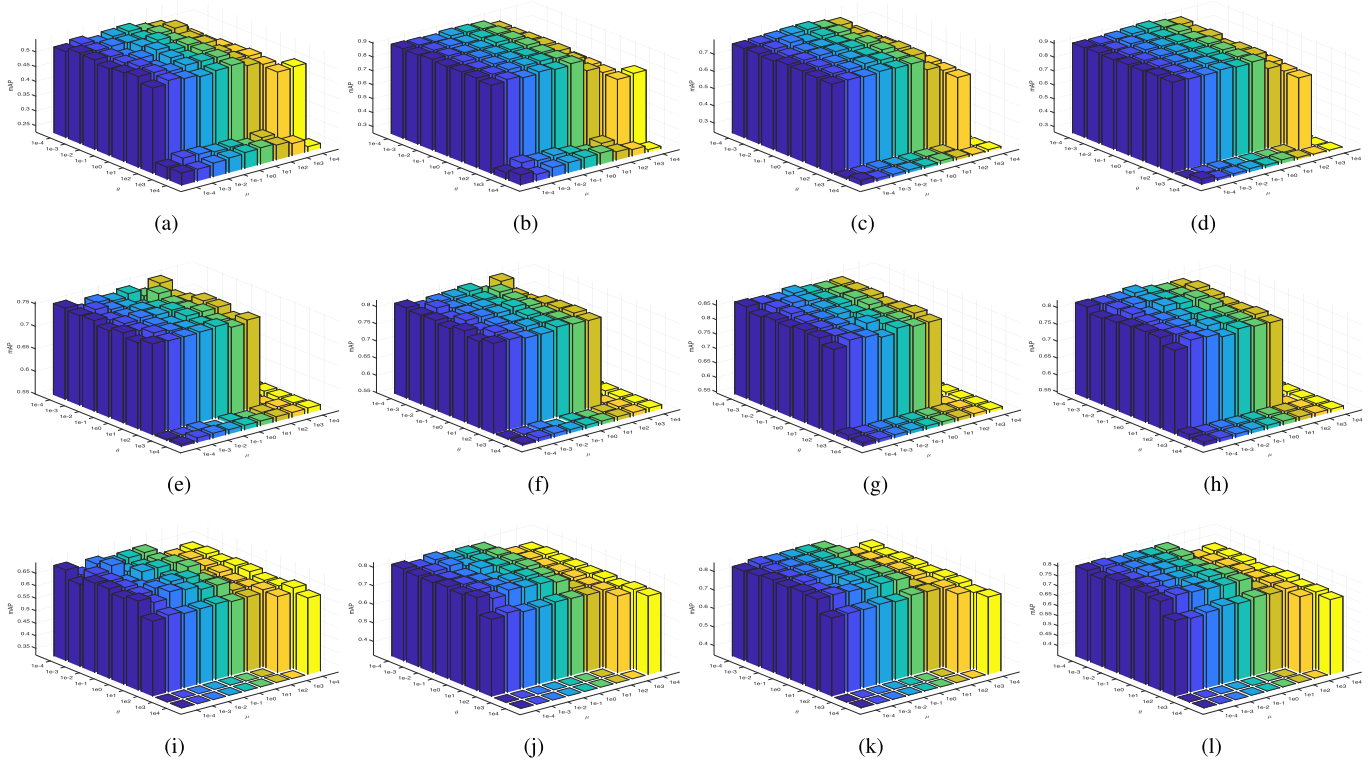

Fig. 5. Sensitivity analysis of the hyperparameters μ and θ tested on handcrafted and CNN visual features. (a) PASCAL-VOC I $\rightarrow$ T (32 bits, handcrafted). (b) PASCAL-VOC T $\rightarrow$ I (32 bits, handcrafted). (c) PASCAL-VOC I $\rightarrow$ T (32 bits, CNN). (d) PASCAL-VOC T $\rightarrow$ I (32 bits, CNN). (e) MIRFlikcr I $\rightarrow$ T (32 bits, handcrafted). (f) MIRFlikcr T $\rightarrow$ I (32 bits, handcrafted). (g) MIRFlikcr I $\rightarrow$ T (32 bits, CNN). (h) MIRFlikcr T $\rightarrow$ I (32 bits, CNN). (i) NUS-WIDE I $\rightarrow$ T (32 bits, handcrafted). (j) NUS-WIDE T $\rightarrow$ I (32 bits, handcrafted). (k) NUS-WIDE I $\rightarrow$ T (32 bits, CNN). (l) NUS-WIDE T $\rightarrow$ I (32 bits, CNN).

TABLE VI
TRAINING TIME (IN SECOND) ON SUBSETS OF NUS-WIDE

Method	1K	5K	10K	50K	184K	Kernel time
CMFH [28]	43.03	345.14	977.17	-	-	-
IMH [3]	0.59	21.33	140.26	-	-	-
SePH_km [11]	2.27	64.32	250.36	-	-	323.50
GSePH_km [9]	135.23	1343.94	4182.59	-	-	352.82
DCH [10]	1.08	2.15	4.45	40.21	242.84	-
DLFH [30]	21.20	86.92	186.34	1071.59	3701.65	-
SCRATCH [35]	1.27	4.83	8.68	39.98	141.12	40.48
SRLCH [36]	3.55	3.86	5.01	14.21	44.43	40.05
BATCH [37]	3.16	3.08	3.41	7.47	20.78	40.82
FDDH	**0.25**	**0.83**	**1.74**	**6.14**	**19.75**	**40.01**

bits. It is noted that the representative CMFH, IMH, SePH, and GSePH methods are computationally intractable on the large-scale training dataset, and we only implement these methods on relatively small training size, i.e., the data sizes are less than 10k. The training times tested on different data sizes and conducted on different methods are shown in Table VI; it can be observed that the unsupervised CMFH and supervised GSePH methods often require large training time to learn the hash codes. The main reason lies in that CMFH often involves large iterations to convergence, while the GSePH approach takes a massive amount of computations to factorize the large affinity matrix. Meanwhile, the deep learning method, i.e., DLFH, often requires large training time to perform the learning process. Evidently, these methods could be quite time-consuming when the training database is too large.

Comparatively speaking, the competing IMH, DCH, SCRATCH, SRLCH, BATCH, and proposed FDDH methods have significantly reduced the training time. Since the proposed FDDH approach considers more variables to discriminatively learn the hash codes, the execution time of training time could be much higher than the relevant IMH, DCH, SCRATCH, SRLCH, and BATCH methods. Fortunately, the proposed FDDH method not only significantly reduces the training time but also achieves the best cross-modal retrieval performance in most cases. For instance, the execution time obtained by the proposed FDDH method is only around 1.74 m, in contrast to 140.26, 4.45, 8.68, 5.01, and 3.41 m, respectively, evaluated on IMH, DCH, SCRATCH, SRLCH, and BATCH methods. The main advantages contributed to such a fast training process are threefold: 1) the utilization of orthogonal constraint to hash code learning can well reduce the quantization error, which, therefore, speed up the learning process; 2) the ε-dragging technique is able to provide large class margins for fast regression, which can make the optimization procedure converge within fewer iterations; and 3) the proposed FDDH framework has a close-form solution to hash code learning and only requires a single step to update the whole hash codes. As a result, the FDDH algorithm performs sufficiently fast to process the large-scale database.

4) Effects of Model Parameters: There are three main parameters involved within the proposed FDDH framework, i.e., μ, θ, and δ. Specifically, δ is the weight coefficient to control the degree of semantic label relaxation, and the extensive

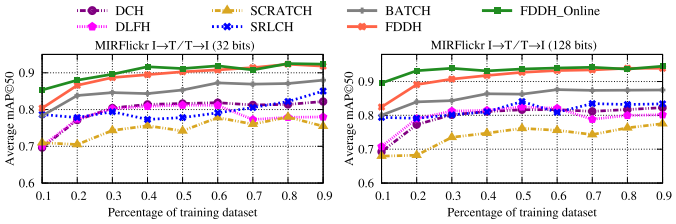

Fig. 6. Retrieval results tested with different training sizes.

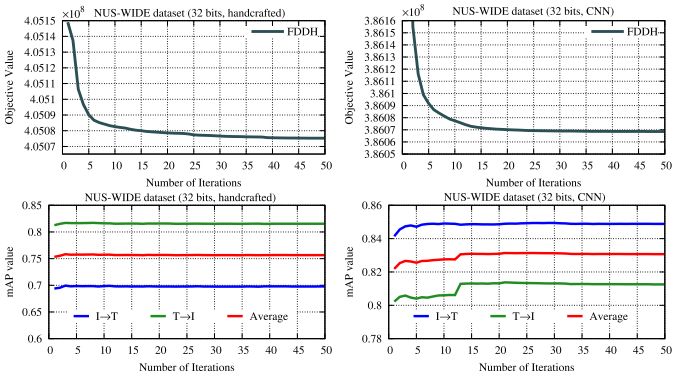

Fig. 7. Results of objective function values and mAP scores recorded in different iterations and tested on the NUS-WIDE dataset.

experiments find that the retrieval performance is always stable when the value of δ is within the range of $[10^2, 10^3]$. Meanwhile, we fix the code length to be 32 bits and set δ at 10^2 to further attempt different values of μ and θ, by varying the value of one parameter while fixing another one. More specifically, we vary the parameter μ within $\{10^{-4}, \ldots, 10^4\}$ and θ within $\{10^{-4}, \ldots, 10^4\}$. The mAP scores tested on different $\{\mu, \theta\}$ values and evaluated on different retrieval tasks are shown in Fig. 6; it can be observed that the mAP scores obtained by the FDDH are very stable in most cases, and the performance deteriorates only when the $\{\mu, \theta\}$ values are very large, e.g., μ greater than 10 and θ greater than 10^3. Overall, the results perform well when μ is selected within the range of $[10^{-4}, 10]$, and θ is chosen within the range of $[10^{-4}, 10^3]$. Besides, the different $\{\mu, \theta\}$ values only induce a minor fluctuation in the retrieval performance. Therefore, these parameters are generally insensitive to the cross-modal retrieval performances within a wide range of values.

5) Efficiency of Online Strategy: For the out-of-sample extensions, the off-line and online strategies are both proposed to learn the modality-specific projections. Similar to most competing works, the off-line learning strategy is selected as the standard way of learning. Besides, an efficient online learning strategy is newly proposed to adapt the streaming data. For simplicity, the proposed FDDH algorithm associated with the online learning strategy is abbreviated as FDDH_online. Furthermore, we vary the training size of the MIRFlickr dataset from 10% to 90% and select the handcrafted features of the MIRFlickr dataset for illustration. Accordingly, the average mAP@50 scores tested on I $\rightarrow$ T and T $\rightarrow$ I tasks are shown in the Fig. 5; it can be observed that both the proposed FDDH and FDDH_online methods always outperform other state-of-art competing methods on different training sizes. Remarkably, the proposed FDDH_online scheme has yielded very promising retrieval performance when the training size is relatively small (less than 30%). This impressive achievement can be attributed to the stable ability of the online learning scheme illustrated in Section III-D2, which can adaptively learn the modality-specific projections within the different training sizes. Therefore, the proposed online learning scheme is able to well learn discriminative hash codes with limited training dataset, which can be efficiently utilized to facilitate large-scale cross-modal retrieval tasks.

6) Convergence Analysis: Algorithm 1 shows the main procedures of the FDDH algorithm, and its complexity has been theoretically analyzed in Section III-E. Besides, we further study the convergence of the proposed FDDH algorithm.

TABLE VII
DIFFERENT OPTIMIZATIONS TESTED ON THE NUS-WIDE DATASET (128 bits)

Method	MAP		Top50 precison		Training time
	I→T	T→I	I→T	T→I	second (s)
DPLM-M	0.4062	0.4300	0.5077	0.5701	126.23 (s)
FSDH-M	0.6511	0.7961	0.7712	0.8745	96.05 (s)
FDDH-DPLM	0.4620	0.4791	0.6852	0.7364	102.63 (s)
FDDH	**0.7118**	**0.8244**	**0.8596**	**0.9234**	**59.76** (s)

By fixing the code length to be 32 bits, we record the objective value and mAP score at each iteration and select the NUS-WIDE dataset for illustration. As shown in Fig. 7, it can be observed that the FDDH converges very fast during the learning process. On the one hand, the objective values almost converge within 30 and 20 iterations, respectively, tested on handcrafted and CNN visual features. On the other hand, the corresponding mAP scores converge to a stable value within very limited iterations, e.g., five iterations for handcrafted features and 15 iterations for CNN visual features, respectively. Therefore, the overall computational load of the proposed FDDH can be significantly reduced due to the fewer iterations.

7) Different Optimizations: In the literature, several fast unimodal hashing methods have been presented, e.g., DPLM [50] and FSDH [51]. Similar to the multimodal extension from SDH [42] to DCH [10], we heuristically extend DPLM and FSDH to adapt the multimodal data (abbreviated, respectively, as DLPM-M and FSDH-M) and compare the proposed FDDH approach with these fast hashing schemes. Specifically, the large NUS-WIDE dataset associated with handcrafted features is selected for illustration, and we also utilize the optimization scheme within DPLM to update $\mathbf{H}$ in (11) (abbreviated as FDDH-DPLM). Accordingly, we record the retrieval performances and training times by different optimizations in Table VII; it can be found that the proposed FDDH approach not only produces the highest mAP scores and top-50 precisions but also involves the lowest training time. It is noted that the DPLM-M and FSDH-M methods, respectively, share the similar optimization algorithms with FDDH-DPLM and FDDH to achieve cross-modal retrieval tasks, and it is easy to find that the results of FDDH-DPLM

and FDDH involve the lower training times while producing the better retrieval performances. The main reason lies in that the proposed FDDH method imposes an orthogonal constraint to reduce the quantization error while employing the relaxed semantic label values for fast convergence. Remarkably, the optimization algorithm within the FDDH framework significantly speeds up the learning process and surprisingly contributes to the fastest implementation. Therefore, the proposed FDDH algorithm runs sufficiently fast, and it is particularly suitable for processing large-scale multimedia datasets.

V. Conclusion

This article has proposed a novel FDDH method for large-scale cross-modal retrieval. The proposed framework introduces an orthogonal basis to regress the targeted hash codes of training examples to their corresponding reasonably relaxed class label values, which offers provable large margin property to efficiently reduce the quantization error. Meanwhile, an orthogonal transformation scheme is further proposed to guarantee the semantic consistency between the data feature vector and its semantic representation. Through the joint exploitation of the above, an efficient closed-form solution is derived for discriminative hash code learning, while an effective online strategy is newly proposed for modality-specific projection function learning. Extensive experiments empirically show that the proposed FDDH approach performs sufficiently fast and brings substantial improvements over the state-of-the-art methods.

References

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

[2] M. Yu, L. Liu, and L. Shao, "Binary set embedding for cross-modal retrieval," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 12, pp. 2899–2910, Dec. 2017.

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

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

[5] 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. Retr.*, Jul. 2014, pp. 415–424.

[6] M. Long, Y. Cao, J. Wang, and P. S. Yu, "Composite correlation quantization for efficient multimodal retrieval," in *Proc. 39th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2016, pp. 579–588.

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

[8] H. Liu, R. Ji, Y. Wu, F. Huang, and B. Zhang, "Cross-modality binary code learning via fusion similarity hashing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6345–6353.

[9] D. Mandal, K. N. Chaudhury, and S. Biswas, "Generalized semantic preserving hashing for N-label cross-modal retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4076–4084.

[10] 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, Mar. 2017.

[11] 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)*, Jun. 2015, pp. 3864–3872.

[12] 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.

[13] 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.

[14] X. Liu, Z. Hu, H. Ling, and Y.-M. 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.

[15] Q.-Y. Jiang and W.-J. Li, "Deep cross-modal hashing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3132–3240.

[16] E. Yang, C. Deng, C. Li, W. Liu, J. Li, and D. Tao, "Shared predictive cross-modal deep quantization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 11, pp. 5292–5303, Nov. 2018.

[17] N. Rasiwasia *et al.*, "A new approach to cross-modal multimedia retrieval," in *Proc. Int. Conf. Multimedia (MM)*, 2010, pp. 251–260.

[18] A. Li, S. Shan, X. Chen, and W. Gao, "Cross-pose face recognition based on partial least squares," *Pattern Recognit. Lett.*, vol. 32, no. 15, pp. 1948–1955, Nov. 2011.

[19] J. B. Tenenbaum and W. T. Freeman, "Separating style and content with bilinear models," *Neural Comput.*, vol. 12, no. 6, pp. 1247–1283, Jun. 2000.

[20] Y. Gong, Q. Ke, M. Isard, and S. Lazebnik, "A multi-view embedding space for modeling Internet images, tags, and their semantics," *Int. J. Comput. Vis.*, vol. 106, no. 2, pp. 210–233, Jan. 2014.

[21] N. Rasiwasia, D. Mahajan, V. Mahadevan, and G. Aggarwal, "Cluster canonical correlation analysis," in *Proc. Int. Joint Conf. Artif. Intell.*, 2014, pp. 823–831.

[22] V. Ranjan, N. Rasiwasia, and C. V. Jawahar, "Multi-label cross-modal retrieval," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4094–4102.

[23] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, "Multimodal deep learning," in *Proc. IEEE Int. Conf. Mach. Learn.*, Jan. 2011, pp. 689–696.

[24] G. Andrew, R. Arora, J. Bilmes, and K. Livescu, "Deep canonical correlation analysis," in *Proc. IEEE Int. Conf. Mach. Learn.*, May 2013, pp. 1247–1255.

[25] D. Wang, P. Cui, M. Ou, and W. Zhu, "Learning compact hash codes for multimodal representations using orthogonal deep structure," *IEEE Trans. Multimedia*, vol. 17, no. 9, pp. 1404–1416, Sep. 2015.

[26] L. Zhu, X. Lu, Z. Cheng, J. Li, and H. Zhang, "Flexible multi-modal hashing for scalable multimedia retrieval," *ACM Trans. Intell. Syst. Technol.*, vol. 11, no. 2, pp. 1–20, Mar. 2020.

[27] X. Lu, L. Zhu, J. Li, H. Zhang, and H. T. Shen, "Efficient supervised discrete multi-view hashing for large-scale multimedia search," *IEEE Trans. Multimedia*, vol. 22, no. 8, pp. 2048–2060, Aug. 2020.

[28] G. Ding, Y. Guo, J. Zhou, and Y. Gao, "Large-scale cross-modality search via collective matrix factorization hashing," *IEEE Trans. Image Process.*, vol. 25, no. 11, pp. 5427–5440, Nov. 2016.

[29] 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.

[30] Q.-Y. Jiang and W.-J. Li, "Discrete latent factor model for cross-modal hashing," *IEEE Trans. Image Process.*, vol. 28, no. 7, pp. 3490–3501, Jul. 2019.

[31] 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.

[32] V. E. Liong, J. Lu, and Y.-P. Tan, "Cross-modal discrete hashing," *Pattern Recognit.*, vol. 79, pp. 114–129, Jul. 2018.

[33] X. Luo, P.-F. Zhang, Y. Wu, Z.-D. Chen, H.-J. Huang, and X.-S. Xu, "Asymmetric discrete cross-modal hashing," in *Proc. ACM Int. Conf. Multimedia Retr.*, Jun. 2018, pp. 204–212.

[34] Z. Yang, J. Long, L. Zhu, and W. Huang, "Nonlinear robust discrete hashing for cross-modal retrieval," in *Proc. 43rd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2020, pp. 1349–1358.

[35] Z.-D. Chen, C.-X. Li, X. Luo, L. Nie, W. Zhang, and X.-S. Xu, "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.

[36] H. T. Shen *et al.*, "Exploiting subspace relation in semantic labels for cross-modal hashing," *IEEE Trans. Knowl. Data Eng.*, early access, Jan. 29, 2020, doi: 10.1109/TKDE.2020.2970050.

[37] Y. Wang, X. Luo, L. Nie, J. Song, W. Zhang, and X.-S. Xu, "BATCH: A scalable asymmetric discrete cross-modal hashing," *IEEE Trans. Knowl. Data Eng.*, early access, Feb. 18, 2020, doi: 10.1109/TKDE.2020.2974825.

[38] L. Jin, K. Li, Z. Li, F. Xiao, G.-J. Qi, and J. Tang, "Deep semantic-preserving ordinal hashing for cross-modal similarity search," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 5, pp. 1429–1440, May 2019.
[39] G. Wu, J. Han, Z. Lin, G. Ding, B. Zhang, and Q. Ni, "Joint image-text hashing for fast large-scale cross-media retrieval using self-supervised deep learning," *IEEE Trans. Ind. Electron.*, vol. 66, no. 12, pp. 9868–9877, Dec. 2019.
[40] X. Zou, X. Wang, E. M. Bakker, and S. Wu, "Multi-label semantics preserving based deep cross-modal hashing," *Signal Process., Image Commun.*, vol. 93, pp. 116–131, Apr. 2021.
[41] Y. Gong, S. Lazebnik, A. Gordo, and F. Perronnin, "Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 12, pp. 2916–2929, Dec. 2013.
[42] F. Shen, C. Shen, W. Liu, and H. T. Shen, "Supervised discrete hashing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 37–45.
[43] B. Kulis and K. Grauman, "Kernelized locality-sensitive hashing for scalable image search," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep. 2009, pp. 2130–2137.
[44] P. H. Schönemann, "A generalized solution of the orthogonal procrustes problem," *Psychometrika*, vol. 31, no. 1, pp. 1–10, Mar. 1966.
[45] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jun. 2010.
[46] M. J. Huiskes, B. Thomee, and M. S. Lew, "New trends and ideas in visual concept detection: The MIR flickr retrieval evaluation initiative," in *Proc. Int. Conf. Multimedia Inf. Retr. (MIR)*, 2010, pp. 527–536.
[47] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. Zheng, "NUS-WIDE: A real-world Web image database from national university of Singapore," in *Proc. ACM Int. Conf. Image Video Retr. (CIVR)*, 2009, pp. 1–9.
[48] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–14.
[49] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.
[50] F. Shen, X. Zhou, Y. Yang, J. Song, H. T. Shen, and D. Tao, "A fast optimization method for general binary code learning," *IEEE Trans. Image Process.*, vol. 25, no. 12, pp. 5610–5621, Dec. 2016.
[51] J. Gui, T. Liu, Z. Sun, D. Tao, and T. Tan, "Fast supervised discrete hashing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 2, pp. 490–496, Feb. 2018.

Xin Liu (Senior Member, IEEE) received the Ph.D. degree in computer science from Hong Kong Baptist University, Hong Kong, in 2013.

He was a Visiting Scholar with the Computer and Information Sciences Department, Temple University, Philadelphia, PA, USA, from 2017 to 2018. He is currently a Full Professor with the Department of Computer Science and Technology, Huaqiao University, Xiamen, China, and also a Research Fellow with the State Key Laboratory of Integrated Services Networks of Xidian University, Xi'an, China. His present research interests include multimedia analysis, computational intelligence, computer vision, pattern recognition, and machine learning.

Xingzhi Wang received the M.S. degree in computer science from Huaqiao University, Xiamen, China, in 2020. He is currently pursuing the Ph.D. degree with the School of Electrics and Information Technology, Sun Yat-sen University, Guangzhou, China.

His present research interests include multimedia signal processing, pattern recognition, and affective computing.

Yiu-ming Cheung (Fellow, IEEE) received the Ph.D. degree in computer science and engineering from the Chinese University of Hong Kong, Hong Kong, in 2000.

He is currently a Full Professor with the Department of Computer Science, Hong Kong Baptist University, Hong Kong. His current research interests include machine learning, pattern recognition, and visual computing.

Prof. Cheung is an IET Fellow, RSA Fellow, and BCS Fellow. He is the Founding Chairman of the Computational Intelligence Chapter of the IEEE Hong Kong Section. He serves as an Associate Editor for the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, the IEEE TRANSACTIONS ON CYBERNETICS, *Pattern Recognition*, *Knowledge and Information Systems*, and *Neurocomputing*.