Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015)

Learning to Hash on Partial Multi-Modal Data

Qifan Wang, Luo Si and **Bin Shen**
Computer Science Department, Purdue University
West Lafayette, IN 47907, US
wang868@purdue.edu, lsi@purdue.edu, bshen@purdue.edu

Abstract

Hashing approach becomes popular for fast similarity search in many large scale applications. Real world data are usually with multiple modalities or having different representations from multiple sources. Various hashing methods have been proposed to generate compact binary codes from multi-modal data. However, most existing multimodal hashing techniques assume that each data example appears in all modalities, or at least there is one modality containing all data examples. But in real applications, it is often the case that every modality suffers from the missing of some data and therefore results in many partial examples, i.e., examples with some modalities missing. In this paper, we present a novel hashing approach to deal with Partial Multi-Modal data. In particular, the hashing codes are learned by simultaneously ensuring the data consistency among different modalities via latent subspace learning, and preserving data similarity within the same modality through graph Laplacian. We then further improve the codes via orthogonal rotation based on the orthogonal invariant property of our formulation. Experiments on two multi-modal datasets demonstrate the superior performance of the proposed approach over several state-of-the-art multi-modal hashing methods.

1 Introduction

With the explosive growth of the Internet, a huge amount of data has been generated, which indicates that efficient similarity search becomes more important. Traditional similarity search methods are difficult to be directly used for large scale applications since linear scan between query example and all candidates in the database is impractical. Moreover, the similarity between data examples is usually conducted in high dimensional space. Hashing methods [Datar *et al.*, 2004; Bergamo *et al.*, 2011; Weiss *et al.*, 2008; Liu *et al.*, 2011; Rastegari *et al.*, 2013; Salakhutdinov and Hinton, 2009; Wang *et al.*, 2010; Ye *et al.*, 2013; Wang *et al.*, 2014a; 2013a; Kong and Li, 2012; Raginsky and Lazebnik, 2009; Zhang *et al.*, 2013; Wang *et al.*, 2014c; 2015] have been proposed to address the similarity search problem within large scale data. Hashing techniques design compact binary code in a low-dimensional space for each data example so that similar examples are mapped to similar binary codes. The retrieval of similar data examples can then be completed in a sublinear or even constant time, using Hamming distance ranking based on fast binary operation (XOR) or hash table lookup within a certain Hamming distance. In addition, the storage cost can be significantly reduced due to the binary compression.

In many applications, data examples are usually represented by multiple modalities captured from different sources. For example, in web page search, the web page content and its linkage information can be regarded as two modalities. In web image retrieval, the image visual feature, text description and textual tags can be viewed as multiple modalities. Recently, several multi-modal hashing methods (also known as multi-view or cross-view) have been proposed to handle multi-modal data. Roughly speaking, these multimodal hashing approaches can be divided into two categories: modality-specific methods and modality-integrated ones.

The modality-specific hashing methods [Bronstein *et al.*, 2010; Kumar and Udupa, 2011; Ou *et al.*, 2013; Quadrianto and Lampert, 2011; Zhen and Yeung, 2012; Liu *et al.*, 2014; Zhai *et al.*, 2013] learn independent hashing codes for each modality of data examples, and then merge multiple binary codes from different modalities into the final hashing codes. A cross-modality similarity search hashing (CMSSH) method [Bronstein *et al.*, 2010] is proposed to embed data from different feature space into a common metric space. The hashing codes are learned through eigen-decomposition with AdaBoost framework. In work [Kumar and Udupa, 2011], a cross-view hashing method is designed based on spectral hashing, which generates the hashing codes by minimizing the distance of hashing codes for similar data and maximizing the distance for dissimilar data. Co-Regularized Hashing [Zhen and Yeung, 2012] method intends to project data from multiple sources, and at the same time, preserve the inter-modality similarity.

The modality-integrated hashing methods [Ding *et al.*, 2014; Gong *et al.*, 2014; 2012; Kim *et al.*, 2012; Zhang *et al.*, 2011; Zhang and Li, 2014] directly learn unified hashing codes for each data example. In the work of [Zhang *et al.*, 2011], a Composite Hashing with Multiple Information Sources (CHMIS) method is proposed to incorporate information from multiple sources into final integrated hashing codes by linearly combining the hashing codes from different modalities. Multi-View Spectral Hashing (MVSH) [Kim *et al.*, 2012] integrates multi-view information into binary codes, and uses product of codewords to avoid undesirable embedding. More recently, A Canonical Correlation Analysis with Iterative Quantization (CCA-ITQ) method has been proposed in [Gong *et al.*, 2014; 2012] which treats the data features and tags as two different modalities. The hashing function is then learned by extracting a common space from these two modalities. The work in [Ding *et al.*, 2014] introduces collective matrix factorization into multi-modal hashing (CMFH), which learns unified hashing codes by collective matrix factorization with latent factor model from different modalities. However, existing multi-modal hashing methods fail to handle the situation where only partial examples are available in different modalities.

Although existing multi-modal hashing methods generate promising results in dealing with multi-modal data, most of them assume that all data examples have full information in all modalities, or there exists at least one modality which contains all the examples. However, in real world tasks, it is often the case that every modality suffers from some missing information, which results in many partial examples [Li *et al.*, 2014]. For instance, in web page search, many web pages may not contain any linkage information. For web image retrieval, not all images are associated with tags or text descriptions. Moreover, the image itself may be inaccessible due to deletion or invalid url. Therefore, it is a practical and important research problem to design effective hashing methods for partial multi-modal data.

In order to apply existing multi-modal hashing methods to partial data, we can either remove the data examples that suffer from missing information, or preprocess the partial examples by first filling in the missing data. The first strategy is clearly not suitable since the purpose is to map all examples to their corresponding binary codes, whereas our experiments show that the second strategy does not achieve good performance either. In this paper, we propose a novel Partial Multi-Modal Hashing (PM^{2}H) approach to deal with such partial data. More specifically, a unified learning framework is developed to learn the binary codes, which simultaneously ensures the data consistency among different modalities via latent subspace learning, and preserves data similarity within the same modality through graph Laplacian. A coordinate descent algorithm is applied as the optimization procedure. We then further reduce the quantization error via orthogonal rotation based on the orthogonal invariant property of our formulation. Experiments on the datasets demonstrate the advantages of the proposed approach over several state-of-the-art multi-modal hashing methods. We summarize the contributions in this paper as follows:

1. We propose a unified hashing method to deal with partial multi-modal data scenario, which can generate effective hashing codes for all data examples. As far as we know, it is the first attempt to learn binary codes on partial multi-modal data.
2. We propose a coordinate descent method for the joint optimization problem. We prove the orthogonal invariant property of the optimal solution and learn an orthogonal rotation by minimizing the quantization error to further improve the code effectiveness.
3. Our extensive experiments demonstrate PM^{2}H is an effective hashing method when only partial multiple modality information sources are available.

2 Partial Multi-Modal Hashing

2.1 Problem Definition

For the convenience of discussion, assume we are dealing with two-modality data, i.e., given a data set of N data examples $\boldsymbol{X}$=$\{(x_i^1, x_i^2), i = 1, \ldots, N\}$, where $x_i^1 \in \mathbb{R}^{d_1}$ is the instance of the *i-th* example in the first modality and $x_i^2 \in \mathbb{R}^{d_2}$ is the *i-th* example in the second modality (usually $d_1 \neq d_2$). In the partial modality setting, a partial data set $\hat{\boldsymbol{X}}$=$\{\hat{\boldsymbol{X}}^{(1,2)}, \hat{\boldsymbol{X}}^{(1)}, \hat{\boldsymbol{X}}^{(2)}\}$ instead of $\boldsymbol{X}$ is given, where $\hat{\boldsymbol{X}}^{(1,2)}$=$\{(x_1^1, x_1^2), \ldots, (x_c^1, x_c^2)\} \in \mathbb{R}^{c\times(d_1+d_2)}$ denotes the common examples present in both modalities, $\hat{\boldsymbol{X}}^{(1)}$=$\{x_{c+1}^1, \ldots, x_{c+m}^1\} \in \mathbb{R}^{m\times d_1}$ denotes the examples only present in the first modality and $\hat{\boldsymbol{X}}^{(2)}$=$\{x_{c+m+1}^2, \cdots, x_{c+m+n}^2\} \in \mathbb{R}^{n\times d_2}$ denotes the examples only present in the second modality. Note that the number of examples present and only present in both modalities, the first modality, and the second modality are c, m and n (N=c+m+n). The purpose of PM^{2}H is to learn unified hashing codes $\boldsymbol{Y}$=$\{y_1, y_2, \ldots, y_N\} \in \{-1, 1\}^{N\times k}$ together with the modality-specific hashing functions $\boldsymbol{H}^1$ and $\boldsymbol{H}^2$ to map each data example x_i to the corresponding hashing codes y_i:

$$y_i = sgn(v_i) = sgn(\boldsymbol{H}^t x_i^t) \quad t = \{1, 2\} \tag{1}$$

where $\boldsymbol{H}^t \in \mathbb{R}^{k\times d_t}$ is the coefficient matrix representing the hashing function for the *t-th* modality and *sgn* is the sign function. k is the length of the code. v_i is the signed magnitude relaxation of binary code y_i, which is widely adopted in previous hashing approaches. The objective function of PM^{2}H is composed of two components: (1) Data consistency between modalities, latent subspace learning is utilized to ensure that the hashing codes generated from different modalities are consistent. (2) Similarity preservation within modality, graph Laplacian is applied to enforce that similar data examples within each modality are mapped into similar codes.

2.2 Data Consistency between Modalities

In the partial modality setting, $\hat{\boldsymbol{X}}^{(1,2)}, \hat{\boldsymbol{X}}^{(1)}, \hat{\boldsymbol{X}}^{(2)}$ are represented by heterogeneous features of dimensions $(d_1 + d_2)$, d_1, d_2, which makes it hard for their hashing codes learning. But investigating the problem from modality perspective, in each individual modality, the data instances are sharing the same feature space. The two different modalities are coupled/bridged by the shared common examples. If we can learn a common latent subspace for the two modalities, where instances belonging to the same example between different modalities are consistent, while at the same time for each modality, the representations for similar instances are close in the latent subspace. Then the hashing codes can be directly learned from this subspace, and we do not need to fill in or complete the partial modality examples. Let $\hat{\boldsymbol{X}}^{(1,2)} = [\hat{\boldsymbol{X}}_c^{(1)}, \hat{\boldsymbol{X}}_c^{(2)}]$, where $\hat{\boldsymbol{X}}_c^{(1)} \in \mathbb{R}^{c \times d_1}$, $\hat{\boldsymbol{X}}_c^{(2)} \in \mathbb{R}^{c \times d_2}$ are the instances of the common examples coming from the two modalities. We denote the instances of each modality as: $\bar{\boldsymbol{X}}^{(1)} = [\hat{\boldsymbol{X}}_c^{(1)}, \hat{\boldsymbol{X}}^{(1)}] \in \mathbb{R}^{(c+m) \times d_1}$, $\bar{\boldsymbol{X}}^{(2)} = [\hat{\boldsymbol{X}}_c^{(2)}, \hat{\boldsymbol{X}}^{(2)}] \in \mathbb{R}^{(c+n) \times d_2}$. Following the above idea, the latent subspace learning can be formulated as:

$$\min_{\bar{\boldsymbol{V}}^{(1)}, \boldsymbol{B}^{(1)}} \|\bar{\boldsymbol{X}}^{(1)} - \bar{\boldsymbol{V}}^{(1)} \boldsymbol{B}^{(1)}\|_F^2 + \lambda \, R(\bar{\boldsymbol{V}}^{(1)}, \boldsymbol{B}^{(1)}) \qquad (2)$$

$$\min_{\bar{\boldsymbol{V}}^{(2)}, \boldsymbol{B}^{(2)}} \|\bar{\boldsymbol{X}}^{(2)} - \bar{\boldsymbol{V}}^{(2)} \boldsymbol{B}^{(2)}\|_F^2 + \lambda \, R(\bar{\boldsymbol{V}}^{(2)}, \boldsymbol{B}^{(2)}) \qquad (3)$$

where $\boldsymbol{B}^{(1)} \in \mathbb{R}^{k \times d_1}$ and $\boldsymbol{B}^{(2)} \in \mathbb{R}^{k \times d_2}$ are the basis matrix for each modality's latent space. $\bar{\boldsymbol{V}}^{(1)} = [\hat{\boldsymbol{V}}_c^{(1)}, \hat{\boldsymbol{V}}^{(1)}] \in \mathbb{R}^{(c+m) \times k}$ and $\bar{\boldsymbol{V}}^{(2)} = [\hat{\boldsymbol{V}}_c^{(2)}, \hat{\boldsymbol{V}}^{(2)}] \in \mathbb{R}^{(c+n) \times k}$ are the latent representation of instances in the latent space, which can also be viewed as the relaxed representation of binary codes $\boldsymbol{Y}$. The same latent space dimension k is shared between the two modalities. $R(\cdot) = \|\cdot\|_F^2$ (sum over all matrices) is the regularization term and λ is the tradeoff parameter. By Eqn.2 and Eqn.3, the latent space basis $\boldsymbol{B}$ and corresponding instance latent representation $\boldsymbol{V}$ are simultaneously learned to minimize the reconstruction error from each individual modality.

In the above equations, the latent space are learned independently for each modality. But in the partial modality setting, for examples present in both modalities $\hat{\boldsymbol{X}}_c^{(1)}$, $\hat{\boldsymbol{X}}_c^{(2)}$, their latent representation $\hat{\boldsymbol{V}}_c^{(1)}$, $\hat{\boldsymbol{V}}_c^{(2)}$ should also be consistent. Incorporating the above formulations by ensuring $\hat{\boldsymbol{V}}_c^{(1)} = \hat{\boldsymbol{V}}_c^{(2)} = \hat{\boldsymbol{V}}_c$, we seek to minimize:

$$\begin{aligned} \min_{\boldsymbol{V}, \boldsymbol{B}} & \left\| \begin{bmatrix} \hat{\boldsymbol{X}}_c^{(1)} \\ \hat{\boldsymbol{X}}^{(1)} \end{bmatrix} - \begin{bmatrix} \hat{\boldsymbol{V}}_c \\ \hat{\boldsymbol{V}}^{(1)} \end{bmatrix} \boldsymbol{B}^{(1)} \right\|_F^2 \\ + & \left\| \begin{bmatrix} \hat{\boldsymbol{X}}_c^{(2)} \\ \hat{\boldsymbol{X}}^{(2)} \end{bmatrix} - \begin{bmatrix} \hat{\boldsymbol{V}}_c \\ \hat{\boldsymbol{V}}^{(2)} \end{bmatrix} \boldsymbol{B}^{(2)} \right\|_F^2 + \lambda \, R(\boldsymbol{V}, \boldsymbol{B}) \end{aligned} \qquad (4)$$

By solving the above problem, we can obtain the homogeneous feature (relaxed hashing) representation for all examples as $\boldsymbol{V} = [\hat{\boldsymbol{V}}_c, \hat{\boldsymbol{V}}^{(1)}, \hat{\boldsymbol{V}}^{(2)}] \in \mathbb{R}^{(c+m+n) \times k}$, whether they are originally partial or not. Then the hashing codes $\boldsymbol{Y}$ can be directly achieved via binarization from this relaxed latent representation. Note that Eqn.4 is different from previous subspace based multi-modal hashing approaches, which either requires $\bar{\boldsymbol{V}}^{(1)}$ and $\bar{\boldsymbol{V}}^{(2)}$ to be the same or do not require $\bar{\boldsymbol{V}}^{(1)}$ and $\bar{\boldsymbol{V}}^{(2)}$ to share any common part. In the above formulation, $\bar{\boldsymbol{V}}^{(1)}$ and $\bar{\boldsymbol{V}}^{(2)}$ share one common representation $\hat{\boldsymbol{V}}_c$, while at the same time have their own individual components. Moreover, the individual basis matrix $\boldsymbol{B}^{(1)}$ and $\boldsymbol{B}^{(2)}$, which are learned from all available instances from both modalities, are connected by the common $\hat{\boldsymbol{V}}_c$.

2.3 Similarity Preservation within Modality

One of the key problems in hashing algorithms is similarity preserving, which indicates that similar data examples should be mapped to similar hashing codes within a short Hamming distance. Therefore, besides the data consistency between different modalities, we also preserve the data similarity within each individual modality. In other words, we want the learned relaxed representation $\boldsymbol{V}$ to preserve the similarity structure in each modality. In this work, we use the L_2 distance to measure the similarity between v_i and v_j as $\|v_i - v_j\|^2$, which is consistent with the Hamming distance between the binary codes y_i and y_j ($\frac{1}{4}\|y_i - y_j\|^2$). Then one natural way to preserve the similarity in each modality is to minimize the weighted average distance as follows:

$$\sum_{i,j} \boldsymbol{S}_{ij}^{(t)} \|v_i - v_j\|^2 \quad t = \{1, 2\} \qquad (5)$$

Here, $\boldsymbol{S}^{(t)}$ is the similarity matrix in $t\text{-}th$ modality, which can be calculated from the instances $\bar{\boldsymbol{X}}^{(t)}$. In this paper, we adopt the local similarity [Wang *et al.*, 2014b; Zhang *et al.*, 2011], due to its nice property in many machine learning applications. To meet the similarity preservation criterion, we seek to minimize this quantity in each modality since it incurs a heavy penalty if two similar examples have very different latent representations.

By introducing a diagonal $n \times n$ matrix $\boldsymbol{D}^{(t)}$, whose entries are given by $\boldsymbol{D}_{ii}^{(t)} = \sum_{j=1}^n \boldsymbol{S}_{ij}^{(t)}$. Eqn.5 can be rewritten as:

$$\begin{gathered} tr\left(\bar{\boldsymbol{V}}^{(t)^T} (\boldsymbol{D}^{(t)} - \boldsymbol{S}^{(t)}) \bar{\boldsymbol{V}}^{(t)}\right) = tr\left(\bar{\boldsymbol{V}}^{(t)^T} \boldsymbol{L}^{(t)} \bar{\boldsymbol{V}}^{(t)}\right) \\ t = \{1, 2\} \end{gathered} \qquad (6)$$

where $\boldsymbol{L}$ is called graph $Laplacian$ [Weiss *et al.*, 2008] and $tr(\cdot)$ is the matrix trace function. By minimizing the above objective in all modalities, the similarity between different examples can be preserved in the latent representation.

2.4 Overall Objective and Optimization

The entire objective function consists of two components: the data consistency between modalities in Eqn.4 and similarity preservation within modality given in Eqn.6 as follows:

$$\begin{aligned} \min_{\boldsymbol{V}, \boldsymbol{B}} O \; = & \; \|\bar{\boldsymbol{X}}^{(1)} - \bar{\boldsymbol{V}}^{(1)} \boldsymbol{B}^{(1)}\|_F^2 + \|\bar{\boldsymbol{X}}^{(2)} - \bar{\boldsymbol{V}}^{(2)} \boldsymbol{B}^{(2)}\|_F^2 \\ & + \alpha \left(tr\left(\bar{\boldsymbol{V}}^{(1)^T} \boldsymbol{L}^{(1)} \bar{\boldsymbol{V}}^{(1)}\right) + tr\left(\bar{\boldsymbol{V}}^{(2)^T} \boldsymbol{L}^{(2)} \bar{\boldsymbol{V}}^{(2)}\right) \right) \\ & + \lambda \, R(\boldsymbol{V}, \boldsymbol{B}) \end{aligned} \qquad (7)$$

where α and λ are trade-off parameters to balance the weights among the terms. Note that $\bar{\boldsymbol{V}}^{(1)}$ and $\bar{\boldsymbol{V}}^{(2)}$ share an identical part $\hat{\boldsymbol{V}}_c$ corresponding to the common examples present in both modalities. Directly minimizing the objective function in Eqn.7 is intractable since it is a non-convex optimization problem with $\boldsymbol{V}$ and $\boldsymbol{B}$ coupled together. We propose to use coordinate descent scheme by iteratively solving the optimization problem with respect to $\boldsymbol{V}$ and $\boldsymbol{B}$ as follows:

(1) Optimizing O with respect to $\hat{\boldsymbol{V}}_c$, $\hat{\boldsymbol{V}}^{(1)}$ and $\hat{\boldsymbol{V}}^{(2)}$ by fixing $\boldsymbol{B}$. Given the basis matrix $\boldsymbol{B}^{(t)}$ for both modalities, we can decompose the objective since $\hat{\boldsymbol{V}}_c$ and $\hat{\boldsymbol{V}}^{(t)}$ will not depend on each other.

$$\min_{\hat{\boldsymbol{V}}^{(t)}} O(\hat{\boldsymbol{V}}^{(t)}) = \|\hat{\boldsymbol{X}}^{(t)} - \hat{\boldsymbol{V}}^{(t)}\boldsymbol{B}^{(t)}\|_F^2 + \alpha\, tr\left(\hat{\boldsymbol{V}}^{(t)^T}\hat{\boldsymbol{L}}^{(t)}\hat{\boldsymbol{V}}^{(t)}\right) + \lambda\, R(\hat{\boldsymbol{V}}^{(t)}) + const \quad t = \{1,2\} \tag{8}$$

$$\min_{\hat{\boldsymbol{V}}_c} O(\hat{\boldsymbol{V}}_c) = \|\hat{\boldsymbol{X}}_c^{(1)} - \hat{\boldsymbol{V}}_c\boldsymbol{B}^{(1)}\|_F^2 + \|\hat{\boldsymbol{X}}_c^{(2)} - \hat{\boldsymbol{V}}_c\boldsymbol{B}^{(2)}\|_F^2 + \alpha\, tr\left(\hat{\boldsymbol{V}}_c{}^T(\hat{\boldsymbol{L}}_c^{(1)} + \hat{\boldsymbol{L}}_c^{(2)})\hat{\boldsymbol{V}}_c\right) + \lambda\, R(\hat{\boldsymbol{V}}_c) + const \tag{9}$$

where $\hat{\boldsymbol{L}}^{(t)}$ and $\hat{\boldsymbol{L}}_c^{(t)}$ can be simply derived from $\boldsymbol{L}^{(1)}$ with some addition mathematical operation. $const$ is the constant value independent with the parameter that to be optimized with. Although Eqn.8 and Eqn.9 are still non-convex, but they are smooth and differentiable which enables gradient descent methods for efficient optimization. We use L-BFGS quasi-Newton method [Liu and Nocedal, 1989] to solve Eqn.8 and Eqn.9 with the obtained gradients. Due to space limitation, we will present the gradients in supplementary material.

(2) Optimizing O with respect to $\boldsymbol{B}^{(t)}$ by fixing $\boldsymbol{V}$. It is equivalent to solve the following least square problems:

$$\min_{\boldsymbol{B}^{(t)}} O(\boldsymbol{B}^{(t)}) = \|\bar{\boldsymbol{X}}^{(t)} - \bar{\boldsymbol{V}}^{(t)}\boldsymbol{B}^{(t)}\|_F^2 + \lambda\|\boldsymbol{B}^{(t)}\|_F^2 \quad t = \{1,2\} \tag{10}$$

By taking the derivative of Eqn.10 w.r.t. $\boldsymbol{B}^{(t)}$ and setting it to $\mathbf{0}$, a closed form solution can be simply obtained. We then alternate the process of updating $\boldsymbol{V}$ and $\boldsymbol{B}$ for several iterations to find a locally optimal solution.

2.5 Orthogonal Rotation

After obtaining the optimal latent representation $\boldsymbol{V}$, the hashing codes $\boldsymbol{Y}$ and modality-specific hashing functions $\boldsymbol{H}^t$ can be generated using Eqn.1. It is obvious that the quantization error can be measured as $\|\boldsymbol{Y} - \boldsymbol{V}\|_F^2$. Inspired by [Gong *et al.*, 2012], we propose to further improve the hashing codes by minimizing this quantization error using an orthogonal rotation. We first prove the following orthogonal invariant theorem.

Theorem 1. *Assume $\boldsymbol{Q}$ is a $k \times k$ orthogonal matrix, i.e., $\boldsymbol{Q}^T\boldsymbol{Q} = \boldsymbol{I}$. If $\boldsymbol{V}$ and $\boldsymbol{B}$ are an optimal solution to the problem in Eqn.7, then $\boldsymbol{VQ}$ and $\boldsymbol{Q}^T\boldsymbol{B}$ are also an optimal solution.*

Proof. By substituting $\boldsymbol{VQ}$ and $\boldsymbol{Q}^T\boldsymbol{B}$ into Eqn.7, it is obvious that: $\|\bar{\boldsymbol{X}}^{(t)} - \bar{\boldsymbol{V}}^{(t)}\boldsymbol{Q}\boldsymbol{Q}^T\boldsymbol{B}^{(t)}\|_F^2 = \|\bar{\boldsymbol{X}}^{(t)} - \bar{\boldsymbol{V}}^{(t)}\boldsymbol{B}^{(t)}\|_F^2$, $tr\left((\bar{\boldsymbol{V}}^{(t)}\boldsymbol{Q})^T\boldsymbol{L}^{(t)}\bar{\boldsymbol{V}}^{(t)}\boldsymbol{Q}\right) = tr\left(\boldsymbol{Q}^T\bar{\boldsymbol{V}}^{(t)^T}\boldsymbol{L}^{(t)}\bar{\boldsymbol{V}}^{(t)}\boldsymbol{Q}\right) = tr\left(\bar{\boldsymbol{V}}^{(t)^T}\boldsymbol{L}^{(t)}\bar{\boldsymbol{V}}^{(t)}\right)$, and $\|\boldsymbol{VQ}\|_F^2 = \|\boldsymbol{V}\|_F^2$, $\|\boldsymbol{Q}^T\boldsymbol{B}\|_F^2 = \|\boldsymbol{B}\|_F^2$. Thus, the value of the objective function in Eqn.7 does not change by the orthogonal rotation. □

According to the above theorem, we propose to seek for better hashing codes by minimizing the quantization error between the binary hashing codes $\boldsymbol{Y}$ and the orthogonal rotation of the latent representation $\boldsymbol{VQ}$ as follows:

$$\min_{\boldsymbol{Y},\boldsymbol{Q}} \|\boldsymbol{Y} - \boldsymbol{VQ}\|_F^2 \quad s.t. \quad \boldsymbol{Y} \in \{-1,1\}^{N \times k}, \;\; \boldsymbol{Q}^T\boldsymbol{Q} = \boldsymbol{I} \tag{11}$$

Intuitively, we seek binary codes that are close to some orthogonal transformation of the latent representation. The orthogonal rotation not only preserves the optimality of the solution but also provides us more flexibility to achieve better hashing codes with low quantization error. The above optimization problem can be solved by minimizing Eqn.11 with respect to $\boldsymbol{Y}$ and $\boldsymbol{Q}$ alternatively.

Fix Q and update Y. The closed form solution can be expressed as:

$$\boldsymbol{Y} = sgn\,(\boldsymbol{VQ}) \tag{12}$$

which is identical with Eqn.1 except the rotation.

Fix Y and update Q. The objective function becomes:

$$\min_{\boldsymbol{Q}^T\boldsymbol{Q}=\boldsymbol{I}} \|\boldsymbol{Y} - \boldsymbol{VQ}\|_F^2 \tag{13}$$

In this case, the objective function is essentially the classic Orthogonal Procrustes problem [Schonemann, 1966], which can be solved efficiently by singular value decomposition using the following theorem (detailed proof in [Schonemann, 1966]).

Theorem 2. *Let $\boldsymbol{S\Lambda U}^T$ be the singular value decomposition of $\boldsymbol{Y}^T\boldsymbol{V}$. Then $\boldsymbol{Q} = \boldsymbol{US}^T$ minimizes the objective function in Eqn.13.*

We perform the above two steps alternatively to obtain the optimal hashing codes and the orthogonal rotation matrix. The modality-specific hashing functions can be then derived by minimizing the projection error as:

$$\min_{\boldsymbol{H}^t} \|\bar{\boldsymbol{X}}^{(t)}(\boldsymbol{H}^t)^T - \bar{\boldsymbol{V}}^{(t)}\boldsymbol{Q}\|_F^2 + \gamma\|\boldsymbol{H}^t\|_F^2 \quad t = \{1,2\} \tag{14}$$

where γ is the tradeoff parameter of the regularization term. The full learning algorithm is described in Algorithm 1.

Algorithm 1 Partial Multi-Modal Hashing (PM^{2}H)

```
Input: Partial data $\{\hat{X}^{(1,2)}, \hat{X}^{(1)}, \hat{X}^{(2)}\}$, trade-off parameters $\alpha$, $\lambda$ and $\gamma$
Output: Unified hashing codes $Y$ and hashing functions $H^1$, $H^2$
  Initialize $B$, Calculate $L$.
  repeat
      Optimize Eqns.8 and 9 and update $\hat{V}_c$, $\hat{V}^{(1)}$ and $\hat{V}^{(2)}$.
      Optimize Eqn.10 and update $B^{(1)}$ and $B^{(2)}$.
  until the solution converges
  repeat
      Update $Y$ using Eqn.12
      Update $Q = US^T$ according to Theorem 2.
  until the solution converges
  Obtain the hashing functions $H^1$ and $H^2$ from Eqn.14.
```

modality 1	NUS-WIDE					MIRFLICKR-25k				
# of bits	8	16	32	64	128	8	16	32	64	128
PM^{2}H	**0.455**	**0.476**	**0.514**	**0.522**	**0.533**	**0.548**	**0.567**	**0.582**	**0.601**	**0.614**
CMFH	0.432	0.448	0.463	0.476	0.484	0.519	0.533	0.545	0.560	0.568
CCA-ITQ	0.397	0.415	0.428	0.436	0.443	0.451	0.475	0.488	0.496	0.513
CMSSH	0.368	0.380	0.403	0.411	0.414	0.402	0.417	0.421	0.426	0.429
CVH	0.285	0.307	0.324	0.336	0.331	0.438	0.456	0.472	0.470	0.475

modality 2	NUS-WIDE					MIRFLICKR-25k				
# of bits	8	16	32	64	128	8	16	32	64	128
PM^{2}H	**0.422**	**0.445**	**0.462**	**0.473**	**0.479**	**0.550**	**0.571**	**0.595**	**0.608**	**0.618**
CMFH	0.386	0.403	0.414	0.427	0.431	0.504	0.521	0.536	0.547	0.549
CCA-ITQ	0.347	0.361	0.377	0.385	0.392	0.498	0.515	0.526	0.535	0.541
CMSSH	0.353	0.372	0.391	0.386	0.382	0.470	0.493	0.502	0.499	0.504
CVH	0.312	0.338	0.348	0.355	0.351	0.456	0.468	0.481	0.475	0.472

Table 1: Precision of top 100 retrieved examples with PDR=0.4.

2.6 Analysis

This section provides some complexity analysis on the training cost of the learning algorithm. The optimization algorithm of PM^{2}H consists of two main loops. In the first loop, we iteratively solve $\boldsymbol{V}$ and $\boldsymbol{B}$ to obtain the optimal solution, where the time complexities for solving $\boldsymbol{V}$ and $\boldsymbol{B}$ are bounded by $O(Nkd_1 + Nkd_2 + Nk^2 + N^2k)$ and $O(Nk^2 + Nkd_1 + Nkd_2)$ respectively. The second loop iteratively optimizes the binary hashing codes and the orthogonal rotation matrix, where the time complexities for updating $\boldsymbol{Y}$ and $\boldsymbol{Q}$ are bounded by $O(Nk^2 + k^3)$. Thus, the total time complexity of the learning algorithm is bounded by $O(Nkd_1 + Nkd_2 + N^2k + Nk^2 + k^3)$. For each query, the hashing time is constant $O(d_1k)$ and $O(d_2k)$.

3 Experimental Results

3.1 Datasets and Setting

We evaluate our method on two image datasets: NUS-WIDE and MIRFLICKR-25k. NUS-WIDE[1] contains 270k images associated with more than 5k unique tags. 81 ground-truth concepts are annotated on these images. We filter out those images with less than 10 tags, resulting in a subset of 110k image examples. Visual features are represented by 500-dimension SIFT [Lowe, 2004] histograms, and text features are represented by index vectors of the most common 2k tags. We use 90% of the data as the training set and the rest 10% as the query set. MIRFLICKR-25k[2] is collected from Flicker images for image retrieval tasks. This dataset contains 25k image examples associated with 38 unique labels. 100-dimensional SIFT descriptors and 512-dimensional GIST descriptors [Oliva and Torralba, 2001] are extracted from these images as the two modalities. We randomly choose 23k image examples as the training set and 2k for testing. Two image examples are considered to be similar if they share at least one ground-truth concept/label. In our experiments, SIFT feature is viewed as modality 1, while text and GIST features are viewed as modality 2.

To simulate the partial modality setting, we randomly select a fraction of training examples to be partial examples, i.e., they are represented by either of the modality but not both, and the remaining ones appear in both modalities. We refer the fraction number of partial examples as Partial Data Ratio (PDR), i.e., $\frac{m+n}{N}$.

The proposed PM^{2}H approach is compared with four different multi-modal hashing methods, i.e., CVH [Kumar and Udupa, 2011], CMSSH [Bronstein *et al.*, 2010], CCA-ITQ [Gong *et al.*, 2014; 2012] and CMFH [Ding *et al.*, 2014].[3] We implement our algorithm using Matlab on a PC with Intel Duo Core i5-2400 CPU 3.1GHz and 8GB RAM. The parameters α, λ and γ are tuned by 5-fold cross validation on the training set. We set the maximum number of iterations to 100. To remove any randomness caused by random selection of training set and random initialization, all of the results are averaged over 10 runs.

3.2 Results and Discussion

We first evaluate the performance of different methods by varying the number of hashing bits in the range of $\{8, 16, 32, 64, 128\}$, with fixed PDR 0.4. To apply the compared multi-modal hashing methods to the partial data, a simple way is to fill in the missing data with 0. However, this may result in large fitting errors between two modalities for the multi-modal methods, since the hashing code for the missing instance will be 0. Therefore, to achieve stronger baseline results, we replace the missing instance using the linear combination of its 5 nearest neighbor examples (weighed by their similarities) which appear in both modalities[4]. Then the baseline multi-modal hashing methods can be directly applied on these extended data.

The precisions for the top 100 retrieved examples are reported in Table 1. From these comparison results, we can see that PM^{2}H provides the best results among all five

[1] http://lms.comp.nus.edu.sg/research/NUS-WIDE.htm

[2] http://press.liacs.nl/mirflickr/

[3] We implement CVH and obtain the codes of CMSSH and CMFH from the authors. The code of CCA-ITQ is public available.

[4] We empirically choose 5 in our experiments. But other numbers can also be applied.

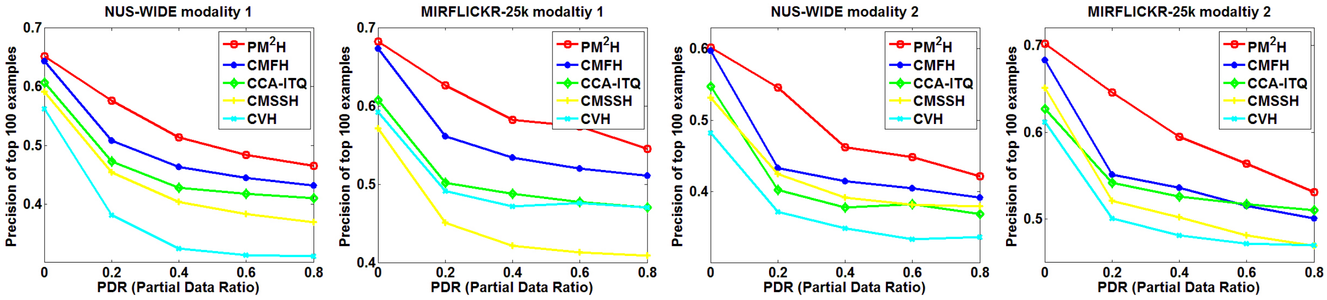

Figure 1: Precision of top 100 retrieved examples under different PDRs with 32 bits.

mod 1	NUS-WIDE			MIRFLICKR-25k		
bits	16	32	64	16	32	64
After	0.476	0.514	0.522	0.567	0.582	0.601
Before	0.463	0.504	0.515	0.552	0.566	0.587

Table 2: Precision of top 100 examples before and after orthogonal rotation with PDR=0.4 on modality 1.

hashing methods on both datasets. For example, the precision of PM^{2}H increases over 8% and 15% on average compared with CMFH and CCA-ITQ on NUS-WIDE under modality 1. The reason is that PM^{2}H can effectively handle the partial data by common subspace learning between modalities and similarity preservation within modality, while the compared methods fail to accurately extract a common space from the partial examples. It can be seen from Table 1 that CMSSH and CVH do not perform well especially with 64 or 128 bits. This phenomenon has also been observed in [Ding *et al.*, 2014; Wang *et al.*, 2013b]. Actually, in CMSSH and CVH methods, the hashing codes are learned by eigenvalue decomposition under the hard bit orthogonality constraint, which makes the first few projection directions very discriminative with high variance. However, the hashing codes will be dominated by bits with very low variance when the code length increases, resulting in many meaningless and ambiguous bits. Another interesting observation is that the retrieval result from modality 1 is better than that from modality 2 on NUS-WIDE. This coincides with our expectation that the image modality is more informative than the tag modality since tags are usually noisy and incomplete.

To evaluate the effectiveness of the proposed PM^{2}H under different partial data ratios, we progressively increase the PDR from $\{0,\ 0.2,\ 0.4,\ 0.6,\ 0.8\}$ and compare our method with the other baselines by fixing the hashing bits to 32. The precision results of top 100 retrieved examples are shown in Fig.1. It can be seen from the figure that when the partial data ratio PDR is 0, the data actually becomes the traditional multi-modal setting with each example appears in both modalities. In this case, PM^{2}H is also able to perform better than most baselines and is comparable with CMFH. As the PDR increases from 0 to 0.8, our PM^{2}H approach always achieves the best performance among all compared methods. Although the missing instances are recovered from the common examples in both modalities, the baseline methods seem less effective in the modality missing case. Our hypothesis is that the missing data may not be accurately recovered when the data are missing blockwise for the partial data setting. In other words, the missing examples can be dissimilar to all the examples appear in both modalities.

We also evaluate the code effectiveness with and without orthogonal rotation. The comparison results (before and after rotation) in Table 2 demonstrate that the orthogonal rotation can further improve the effectiveness of the codes, which is consistent with our expectation since the quantization error is minimized through the rotation. Similar results on modality 2 are observed. Furthermore, we conduct parameter sensitivity experiment on α and λ by tuning only one parameter while fixing the other one to the optimal values obtained from the previous experiments. We identify that the performance of PM^{2}H is relatively stable with respect to $\alpha \in (2, 100)$ and $\lambda \in (0.001, 0.1)$.

4 Conclusions

This paper propose a novel hashing approach to deal with partial multi-modal data. We formulate a unified learning framework by simultaneously ensuring data consistency among different modalities via latent subspace learning, and preserving data similarity within the same modality through graph Laplacian. A coordinate descent algorithm is applied to solve the optimization problem. We then utilize orthogonal rotation to further reduce the quantization error. Experiments on two datasets demonstrate the advantages of the proposed approach in dealing with partial multi-modal data over several multi-modal hashing methods. There are several possibilities to explore in the future research. For example, we plan to apply some sequential learning approach to accelerate the training process. We also plan to extend this subspace based partial modality learning idea to nonlinear latent subspace cases.

5 Acknowledgments

This work is partially supported by NSF research grants IIS-0746830, DRL-0822296, CNS-1012208, IIS-1017837, CNS-1314688 and a research grant from Office of Naval Research (ONR-11627465). This work is also partially supported by the Center for Science of Information (CSoI), an NSF Science and Technology Center, under grant agreement CCF-0939370.

References

[Bergamo *et al.*, 2011] Alessandro Bergamo, Lorenzo Torresani, and Andrew W. Fitzgibbon. Picodes: Learning a compact code for novel-category recognition. In *NIPS*, pages 2088–2096, 2011.

[Bronstein *et al.*, 2010] Michael M. Bronstein, Alexander M. Bronstein, Fabrice Michel, and Nikos Paragios. Data fusion through cross-modality metric learning using similarity-sensitive hashing. In *CVPR*, pages 3594–3601, 2010.

[Datar *et al.*, 2004] Mayur Datar, Nicole Immorlica, Piotr Indyk, and Vahab S. Mirrokni. Locality-sensitive hashing scheme based on p-stable distributions. In *Symposium on Computational Geometry*, pages 253–262, 2004.

[Ding *et al.*, 2014] Guiguang Ding, Yuchen Guo, and Jile Zhou. Collective matrix factorization hashing for multimodal data. In *CVPR*, pages 2083–2090, 2014.

[Gong *et al.*, 2012] Yunchao Gong, Svetlana Lazebnik, Albert Gordo, and Florent Perronnin. Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval. *IEEE TPAMI*, 2012.

[Gong *et al.*, 2014] Yunchao Gong, Qifa Ke, Michael Isard, and Svetlana Lazebnik. A multi-view embedding space for modeling internet images, tags, and their semantics. *International Journal of Computer Vision*, 106(2):210–233, 2014.

[Kim *et al.*, 2012] Saehoon Kim, Yoonseop Kang, and Seungjin Choi. Sequential spectral learning to hash with multiple representations. In *ECCV*, pages 538–551, 2012.

[Kong and Li, 2012] Weihao Kong and Wu-Jun Li. Isotropic hashing. In *NIPS*, pages 1655–1663, 2012.

[Kumar and Udupa, 2011] Shaishav Kumar and Raghavendra Udupa. Learning hash functions for cross-view similarity search. In *IJCAI*, pages 1360–1365, 2011.

[Li *et al.*, 2014] Shao-Yuan Li, Yuan Jiang, and Zhi-Hua Zhou. Partial multi-view clustering. In *AAAI*, pages 1968–1974, 2014.

[Liu and Nocedal, 1989] Dong C. Liu and Jorge Nocedal. On the limited memory bfgs method for large scale optimization. *Mathematical Programming*, 45:503–528, 1989.

[Liu *et al.*, 2011] Wei Liu, Jun Wang, Sanjiv Kumar, and Shih-Fu Chang. Hashing with graphs. In *ICML*, pages 1–8, 2011.

[Liu *et al.*, 2014] Xianglong Liu, Junfeng He, Cheng Deng, and Bo Lang. Collaborative hashing. In *CVPR*, pages 2147–2154, 2014.

[Lowe, 2004] David G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[Oliva and Torralba, 2001] Aude Oliva and Antonio Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *IJCV*, 42(3):145–175, 2001.

[Ou *et al.*, 2013] Mingdong Ou, Peng Cui, Fei Wang, Jun Wang, Wenwu Zhu, and Shiqiang Yang. Comparing apples to oranges: a scalable solution with heterogeneous hashing. In *SIGKDD*, pages 230–238, 2013.

[Quadrianto and Lampert, 2011] Novi Quadrianto and Christoph H. Lampert. Learning multi-view neighborhood preserving projections. In *ICML*, pages 425–432, 2011.

[Raginsky and Lazebnik, 2009] Maxim Raginsky and Svetlana Lazebnik. Locality-sensitive binary codes from shift-invariant kernels. In *NIPS*, pages 1509–1517, 2009.

[Rastegari *et al.*, 2013] Mohammad Rastegari, Jonghyun Choi, Shobeir Fakhraei, Hal Daumé III, and Larry S. Davis. Predictable dual-view hashing. In *ICML*, pages 1328–1336, 2013.

[Salakhutdinov and Hinton, 2009] Ruslan Salakhutdinov and Geoffrey E. Hinton. Semantic hashing. *Int. J. Approx. Reasoning*, 50(7):969–978, 2009.

[Schonemann, 1966] Peter Schonemann. A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 31(1):1–10, 1966.

[Wang *et al.*, 2010] Jun Wang, Ondrej Kumar, and Shih-Fu Chang. Semi-supervised hashing for scalable image retrieval. In *CVPR*, pages 3424–3431, 2010.

[Wang *et al.*, 2013a] Jun Wang, Wei Liu, Andy Sun, and Yu-Gang Jiang. Learning hash codes with listwise supervision. In *ICCV*, 2013.

[Wang *et al.*, 2013b] Qifan Wang, Dan Zhang, and Luo Si. Semantic hashing using tags and topic modeling. In *SIGIR*, pages 213–222, 2013.

[Wang *et al.*, 2014a] Qifan Wang, Bin Shen, Shumiao Wang, Liang Li, and Luo Si. Binary codes emmbedding for fast image tagging with incomplete labels. In *ECCV*, 2014.

[Wang *et al.*, 2014b] Qifan Wang, Luo Si, and Dan Zhang. Learning to hash with partial tags: Exploring correlation between tags and hashing bits for large scale image retrieval. In *ECCV*, pages 378–392, 2014.

[Wang *et al.*, 2014c] Qifan Wang, Luo Si, Zhiwei Zhang, and Ning Zhang. Active hashing with joint data example and tag selection. In *SIGIR*, 2014.

[Wang *et al.*, 2015] Qifan Wang, Luo Si, and Bin Shen. Learning to hash on structured data. In *AAAI*, 2015.

[Weiss *et al.*, 2008] Yair Weiss, Antonio Torralba, and Robert Fergus. Spectral hashing. In *NIPS*, pages 1753–1760, 2008.

[Ye *et al.*, 2013] Guangnan Ye, Dong Liu, Jun Wang, and Shih-Fu Chang. Large scale video hashing via structure learning. In *ICCV*, 2013.

[Zhai *et al.*, 2013] Deming Zhai, Hong Chang, Yi Zhen, Xianming Liu, Xilin Chen, and Wen Gao. Parametric local multimodal hashing for cross-view similarity search. In *IJCAI*, 2013.

[Zhang and Li, 2014] Dongqing Zhang and Wu-Jun Li. Large-scale supervised multimodal hashing with semantic correlation maximization. In *AAAI*, pages 2177–2183, 2014.

[Zhang *et al.*, 2011] Dan Zhang, Fei Wang, and Luo Si. Composite hashing with multiple information sources. In *SIGIR*, pages 225–234, 2011.

[Zhang *et al.*, 2013] Lei Zhang, Yongdong Zhang, Jinhui Tang, Ke Lu, and Qi Tian. Binary code ranking with weighted hamming distance. In *CVPR*, pages 1586–1593, 2013.

[Zhen and Yeung, 2012] Yi Zhen and Dit-Yan Yeung. Co-regularized hashing for multimodal data. In *NIPS*, pages 1385–1393, 2012.