Deep Multimodal Hashing with Orthogonal Regularization

Daixin Wang[1], **Peng Cui**[1], **Mingdong Ou**[1], **Wenwu Zhu**[1]
[1]Tsinghua National Laboratory for Information Science and Technology
Department of Computer Science and Technology, Tsinghua University. Beijing, China
dxwang0826@gmail.com,cuip@tsinghua.edu.cn,oumingdong@gmail.com,wwzhu@tsinghua.edu.cn

Abstract

Hashing is an important method for performing efficient similarity search. With the explosive growth of multimodal data, how to learn hashing-based compact representations for multimodal data becomes highly non-trivial. Compared with shallow-structured models, deep models present superiority in capturing multimodal correlations due to their high nonlinearity. However, in order to make the learned representation more accurate and compact, how to reduce the redundant information lying in the multimodal representations and incorporate different complexities of different modalities in the deep models is still an open problem. In this paper, we propose a novel deep multimodal hashing method, namely Deep Multimodal Hashing with Orthogonal Regularization (**DMHOR**), which fully exploits intra-modality and inter-modality correlations. In particular, to reduce redundant information, we impose orthogonal regularizer on the weighting matrices of the model, and theoretically prove that the learned representation is guaranteed to be approximately orthogonal. Moreover, we find that a better representation can be attained with different numbers of layers for different modalities, due to their different complexities. Comprehensive experiments on WIKI and NUS-WIDE, demonstrate a substantial gain of **DMHOR** compared with state-of-the-art methods.

1 Introduction

Nowadays, people have generated huge volumes of multimodal contents on the Internet, such as texts, videos and images. Multimodal data carries different aspects of information, and many real-world applications need to integrate multimodal information to obtain comprehensive results. For example, recommendation systems aim to find preferred multimodal items (e.g. web posts with texts and images) for users, and image search systems aim to search images for text queries. Among these important applications, multimodal search, which integrates multimodal information for similarity search is a fundamental problem.

Faced with such huge volumes of multimodal data, multimodal hashing is a promising way to perform efficient multimodal similarity search. The fundamental problem of multimodal hashing is to capture the correlation of multiple modalities to learn compact binary hash codes. Despite the success of these hashing methods [Kumar and Udupa, 2011; Zhang *et al.*, 2011; Zhen and Yeung, 2012], most existing multimodal hashing adopts shallow-structured models. As [Srivastava and Salakhutdinov, 2012a] argued, correlation of multiple modalities exists in the high-level space and the mapping functions from the raw feature space to the high level space are highly nonlinear. Thus it is difficult for shallow models to learn such a high-level correlation [Bengio, 2009].

Recently, multimodal deep learning [Ngiam *et al.*, 2011b; Srivastava and Salakhutdinov, 2012a; 2012b] has been proposed to capture the high-level correlation in multimodal information. [Kang *et al.*, 2012] also proposed a multimodal deep learning scheme to perform hashing. However, existing works do not consider the redundancy problem in the learned representation, which makes it incompact or imprecise, and significantly affects the performance of efficient similarity search. In shallow-structured models, it is common to directly impose the orthogonal regularization on the global dataset to decorrelate different bits. However, due to the high nonlinearity, the objective function of deep learning is highly non-convex of parameters, thus potentially causes many distinct local minima in the parameter space [Erhan *et al.*, 2010]. In order to alleviate this problem, mini-batch training is commonly adopted for deep learning [Ngiam *et al.*, 2011a], but this intrinsically prohibits the possibility to impose regularization directly on the final output representation of global dataset. Therefore, how to solve the redundancy problem of hashing representation learning by deep models is still a challenging and unsolved problem. Furthermore, most of the previously proposed deep models adopt symmetric structures, with the assumption that different modalities possess the same complexity. However, it is intuitive that visual data has much larger semantic gap than textual data, which results in different complexities in different modalities. How to address the imbalanced complexity problem in deep learning models is also critical for multimodal hashing.

To address the above problems, we propose a Deep Multimodal Hashing model with Orthogonal Regularization (as

shown in Figure 1) for mapping multimodal data into a common hamming space. The model fully captures intra-modality and inter-modality correlations to extract useful information from multimodal data. In particular, to address the redundancy problem, we impose orthogonal regularizers on the weighting matrices, and theoretically prove that the learned representation is approximately guaranteed to be orthogonal. For the problem of imbalanced complexities, we empirically tune the number of layers for each modality, and find that a better representation can be attained with different number of layers for different modalities.

The contributions of our paper are listed as follows:

- We propose a novel deep learning framework to generate compact and precise hash codes for multimodal data, by fully exploiting the intra-modality and inter-modality correlation and incorporating different complexities of different modalities.
- We propose a novel method with theoretical basis to reduce the redundancy in the learned hashing representation by imposing orthogonal regularization on the weighting parameters.
- Experiments on two real-world datasets demonstrate a substantial gain of our **DMHOR** model compared to other state-of-the-art hashing methods.

2 Related work

In recent years hashing methods have experienced great success in many real-world applications because of their superiority in searching efficiency and storage requirements. In general, there are mainly two different ways for hashing to generate hash codes: data-independent and data-dependent ways.

Data-independent hashing methods often generate random projections as hash functions. Locality Sensitive Hashing (**LSH**) [Datar *et al.*, 2004] is one of the most well-known representative. It uses a set of random locality sensitive hashing functions to map examples to hash codes. Further improvements such as multi-probe **LSH** [Lv *et al.*, 2007] are proposed but the performance is still limited by the random projection technique. Data-dependent hashing methods were then proposed. They use machine learning to utilize the distribution of data to help improve the retrieval quality. Spectral Hashing [Weiss *et al.*, 2008] is a representative. Then some other hashing methods are proposed, including shallow structured methods [Norouzi and Blei, 2011; Liu *et al.*, 2012; Wang *et al.*, 2010] and deep learning based methods [Salakhutdinov and Hinton, 2009; Xia *et al.*, 2014].

Most of the above methods are designed for single modality data. However, we often need to process multimodal information in real-world applications. Therefore, some recent works have focused on encoding examples represented by multimodal features. For example, Cross-View Hashing (**CVH**) [Kumar and Udupa, 2011] extends spectral hashing to multiview. Predictable Dual-view Hashing (**PDH**) [Rastegari *et al.*, 2013] applies max-margin theory to perform multimodal hashing. Relation-aware Heterogeneous Hashing (**RaHH**) [Ou *et al.*, 2013] incorporates a heterogeneous relationship to help learning the multimodal hash function and so on [Bronstein *et al.*, 2010; Zhai *et al.*, 2013; Song *et al.*, 2013; Zhang and Li, 2014; Ou *et al.*, 2015]. However, these multimodal hashing models adopt shallow-layer structures. It is difficult for them to capture the correlation between different modalities.

Only a few recent works focus on multimodal deep learning. [Ngiam *et al.*, 2011b; Srivastava and Salakhutdinov, 2012a; 2012b] target at learning high-dimensional latent features to perform discriminative classification task. [Wang *et al.*, 2014a; Feng *et al.*, 2014] apply autoencoder to perform cross-modality retrieval. However, these methods are all different from our task of learning compact hash codes for multimodal data. From the angle of targeted problem, the most related work with ours is [Kang *et al.*, 2012], which proposed a multimodal deep learning scheme to perform hashing. However, in their scheme, it does not consider the redundant information between different bits of hash codes. The redundancy in hash codes will badly influence the performance of similarity search due to the compact characteristic of hashing representations. In addition, they fail to consider the different complexity of different modalities.

3 The Methodology

In this section, we present Deep Multimodal Hashing with Orthogonal Regularization (**DMHOR**) in detail and analyze its complexity to prove the scalability.

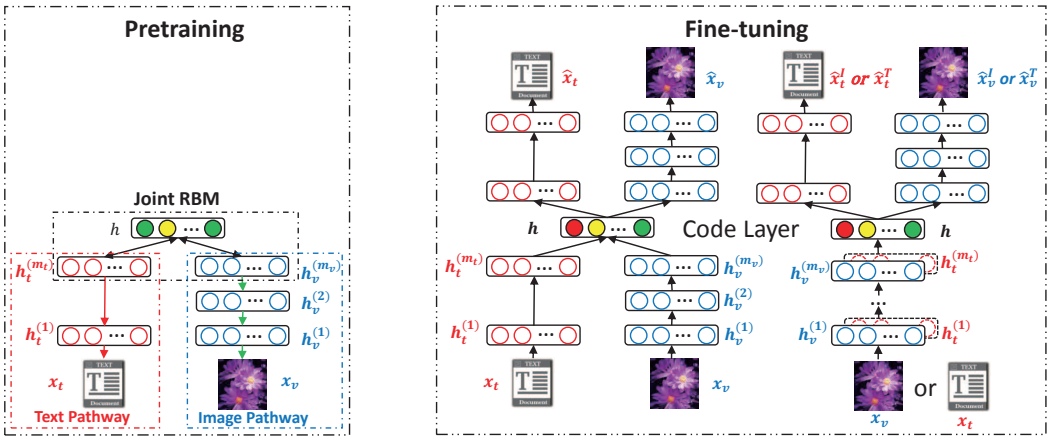
3.1 Notations and Problem Statement

In this paper, we use image and text as the input of two different modalities without loss of generality. Note that it is easy for the model to be extended to incorporate other forms of representations and more modalities. The terms and notations of our model are listed in Table 1. Note that the subscript v represents image modality and t represents text modality.

Table 1: Terms and Notations

Symbol	Definition
m_v,m_t	number of hidden layers in image or text pathway
n	number of samples
M	the length of the hash codes
$\mathbf{x}_v$, $\mathbf{x}_t$	image or text input
$\mathbf{h}_v^{(l)}$, $\mathbf{h}_t^{(l)}$	the l-th hidden layer for image or text pathway
$\mathbf{h}$	top joint layer
$W_v^{(l)}$, $W_t^{(l)}$	l-th layer's weight matrix for image or text pathway
$\mathbf{b}_v^{(l)}$, $\mathbf{b}_t^{(l)}$	l-th biases for image or text pathway
θ	$\{W_v^{(l)}, \mathbf{b}_v^{(l)}, \mathbf{c}_v^{(l)}\}_{l=1}^{m_v+1} \cup \{W_t^{(l)}, \mathbf{b}_t^{(l)}, \mathbf{c}_t^{(l)}\}_{l=1}^{m_t+1}$
$s_v^{(l)}$,$s_t^{(l)}$	l-th layer's unit numbers for image or text pathway

Suppose that we have n training examples with image-text pairs, represented by $X_v = \{\mathbf{x}_{v,1}, ..., \mathbf{x}_{v,n}\}$ and $X_t = \{\mathbf{x}_{t,1}, ..., \mathbf{x}_{t,n}\}$. The main objective is to find M-bit binary hashing codes H for training examples, as well as a corresponded hash function $f(\cdot)$, which satisfies $H = f(X_v, X_t)$. Moreover, if two objects O_1 and O_2 are semantic similar, the hash functions should satisfy that the distance of hash codes $H(O_1)$ and $H(O_2)$ is small. After learning the hash function, given any out-of-sample image-text pair denoted as $(\mathbf{x}_v, \mathbf{x}_t)$, its corresponding hashing codes are calculated as $f(\mathbf{x}_v, \mathbf{x}_t)$.



(a) (b)

Figure 1: (a) The multimodal DBN in pretraining (b) The multimodal AutoEncoder and the cross-modality Autoencoder in fine-tuning. They are optimized together. The same color in different bits means they may contain redundant information.

3.2 Modality-specific Deep Multimodal Hashing

Pretraining

Due to the high nonlinearity, deep-structured models often suffer from many local minima in the parameter space. In order to find a good region of parameter space, we use pretraining in our multimodal hashing.

As the correlation of multiple modalities exists in the high-level space, we propose a multimodal Deep Belief Network (mDBN) as shown in Figure 1(a). The mDBN is composed of two DBNs and a joint Restricted Boltzmann Machine (RBM). The two DBNs map individual low-level features to high-level space and the joint RBM captures the correlation of multiple modalities.

To train the mDBN, we adopt the popular method of greedy layer-wise training [Hinton *et al.*, 2006]. We perform approximately learning of the RBM with 1-step Contrastive Divergence [Hinton, 2002] and adopt dropout [Hinton *et al.*, 2012] over the entire network to prevent overfitting.

Fine-Tuning

After Pre-training, the parameters lie in a good region of the parameter space, but are not optimal. Thus we do fine-tuning to refine the parameters by performing local gradient search.

To learn an accurate representation, we need to incorporate both intra-modality and inter-modality correlation to extract more discriminative information. To preserve the intra-modality correlation, we unroll the mDBN to form the multimodal Autoencoder (MAE) as shown in the left part of Figure 1(b). Given input of two modalities, the joint representation is demanded to reconstruct both modalities.

In this way, the intra-modality correlation is maintained, but the cross-modality cannot be well captured. Inspired by [Ngiam *et al.*, 2011b], we further propose a cross-modality Autoencoder (CAE) as shown in right part of Figure 1(b). In this model, when only one modality is present and the other is absent, the learned representation is still required to be able to reconstruct both modalities. In this way, the common semantic lying in both modalities is strengthened and the modality-specific information is weakened, which results in the effect of capturing the inter-modality correlation.

Modality-specific Structure

Even if different modalities describe the same object, they will have different statistical properties in the low-level raw feature space, while they have high correlation in the high-level semantic space. Thus, deep-structured models adopt multiple layers to map low-level raw feature space to high-level space. However, the gap between low-level feature space to high-level semantic space varies for different modalities. For example, the gap between visual pixels and object categories is much larger than the gap between textual words to topics, which means that the visual modality possess higher complexity than text modality. Therefore, we propose a modality-specific structure on all of the above models to incorporate different complexities of different modalities. In particular, we endow the model with the flexibility of varying the number of layers for different modalities independently. The experimental results clearly demonstrate that the optimal solution is achieved with different number of layers for different modalities.

Hash Function

To define the hash function, we first define the representation of the l-th hidden layer $\mathbf{z}_i^{(l)}$, $i \in \{v, t\}$ for the image or text pathway, and the joint representation $\mathbf{z}$ as follows:

$$\begin{aligned}\mathbf{z}_i^{(1)} &= \sigma(W_i^{(1)T}\mathbf{x}_i + \mathbf{b}_i^{(1)}) \\ \mathbf{z}_i^{(l)} &= \sigma(W_i^{(l)T}\mathbf{z}_i^{(l-1)} + \mathbf{b}_i^{(l)}), l = 2, ..., m_i \\ \mathbf{z} &= \sigma\Big[\sum_{i\in\{v,t\}}(W_i^{(m_i+1)T}\mathbf{z}_i^{(m_i)} + \mathbf{b}_i^{(m_i+1)})\Big] \end{aligned} \quad (1)$$

where $\sigma(\cdot)$ is the sigmoid function. Specifically for CAE, we set the missing modality to zero when calculating Eq. 1. For the decoder part of CAE or MAE, the representation is calculated reversely in a similar way.

Then the hash function $f(\cdot)$ and hash codes $\mathbf{h}$ for input $\mathbf{x}_v$ and $\mathbf{x}_t$ are defined as follows:

$$\mathbf{h} = f(\mathbf{x}_v, \mathbf{x}_t; \theta) = \mathbf{I}[\mathbf{z} \geq \delta] \in \{0, 1\}^M \quad (2)$$

where $\mathbf{I}$ is the indicator function and δ is the threshold.

To obtain a hash function which both preserves intra-modality and inter-modality correlation, we need to learn the parameters θ by optimizing MAE and CAE together. For MAE, the loss function is defined as:

$$L_{vt}(\mathbf{x}_v, \mathbf{x}_t; \theta) = \frac{1}{2}(\|\hat{\mathbf{x}}_v - \mathbf{x}_v\|_2^2 + \|\hat{\mathbf{x}}_t - \mathbf{x}_t\|_2^2) \quad (3)$$

where $\hat{\mathbf{x}}_i$ is the reconstruction of $\mathbf{x_i}$, $i \in \{v, t\}$.

For CAE with only Image input (Image-only CAE), the loss function is defined as:

$$L_{v\bar{t}}(x_v, x_t; \theta) = \frac{1}{2}(\|\hat{\mathbf{x}}_v^I - \mathbf{x}_v\|_2^2 + \|\hat{\mathbf{x}}_t^I - \mathbf{x}_t\|_2^2) \quad (4)$$

where the superscript I denotes the image-only CAE. $\hat{\mathbf{x}}_i^I$ is the reconstruction of $\mathbf{x}_i$, $i \in \{v, t\}$ in image-only CAE. The loss function for text-only CAE $L_{\bar{v}t}$ is defined similarily.

Then a straightforward solution to the hash function is proposed as follows:

$$\min_{\theta} \quad L_1(X_v, X_t; \theta) = \frac{1}{n}\sum_{i=1}^{n}(L_{vt} + \lambda L_{\bar{v}t} + \mu L_{v\bar{t}}) + \nu L_{reg}$$

where L_{reg} is an $\mathcal{L}2$-norm regularizer term of the weight matrix to prevent overfitting.

3.3 Orthogonal Regularization

For hashing representation learning, compactness is a critical criterion to guarantee its performance in efficient similarity search. Given a certain small length of binary codes, the redundancy lies in different bits would badly affect its performance. By removing the redundancy, we can either incorporate more information in the same length of binary codes, or shorten the binary codes while maintaining the same amount of information. Thus to alleviate the redundancy problem, we impose the orthogonal constraints to decorrelate different bits. Since H is non-negative as is shown in Eq. 2, we first transform H to $\tilde{H} = 2H - \mathbf{1} \in \{-1, 1\}$. We formulate the problem as the following objective function:

$$\min_{\theta} \quad L_1(X_v, X_t; \theta) = \frac{1}{n}\sum_{i=1}^{n}(L_{vt} + \lambda L_{\bar{v}t} + \mu L_{v\bar{t}}) + \nu L_{reg}$$
$$s.t. \quad \frac{1}{n}\tilde{H}^T \cdot \tilde{H} = I$$

The above objective function is a hard problem in two aspects. Firstly, the value of $\tilde{H}$ is discrete, which makes $\tilde{H}$ non-differentiable. To solve it, we follow [Weiss *et al.*, 2008] to remove the discrete constraint. Secondly, the orthogonality constraint constrains the hash codes of the global dataset. However, mini-batch training is commonly adopted for deep learning. Thus it prevents us directly solving the constraint on the global dataset. To solve this, we provide the following lemma to transform the objective function.

Lemma 3.1. *Suppose $H = \sigma(W_x X^T + W_y Y^T)$. If X,Y,W_x,W_y are orthogonal matrices, $W_x^T W_y = 0$, then the matrix $\tilde{H} = 2H - 1$ satisfies $\tilde{H}^T \cdot \tilde{H} \propto I$.*

Proof. The one-order Taylor Series of H at $X = 0, Y = 0$ is:

$$\begin{aligned} H &\approx H(0,0) + (\frac{\partial}{\partial X}X^T + \frac{\partial}{\partial Y}Y^T)H(0,0) \\ &= \frac{1}{2} + \frac{1}{4}(W_x X^T + W_y Y^T) \end{aligned}$$

Therefore, $\tilde{H} = 2H - 1 = \frac{1}{2}(W_x X^T + W_y Y^T)$.

$$\begin{aligned} \tilde{H}^T \cdot \tilde{H} &\propto X W_x^T W_x X^T + Y W_y^T W_y Y^T \\ &\quad + X W_x^T W_y Y^T + Y W_y^T W_x X^T \\ &\propto I \end{aligned}$$

□

Under the assumption that input X_v and X_t are orthogonal [Wang *et al.*, 2010], if we impose the orthogonal constraints on each layer's weight matrix as shown in Eq.5, the representations of the layer $h_v^{(m_v)}$ and $h_t^{(m_t)}$ are approximately orthogonal. In this case the input of the joint RBM is orthogonal, if we further impose the constraint of Eq. 6, the Lemma 3.1 guarantees the orthogonality of the hash codes. Thus we can impose the orthogonal regularization on the weighting matrices instead of the hashing bits, which significantly facilitate the optimization process. Therefore, we propose the following new objective function:

$$\min_{\theta} \quad L_1$$
$$s.t. \quad W_i^{(l)^T} \cdot W_i^{(l)} = I, \quad l = 1, ..., m_i + 1 \quad i \in \{v, t\} \quad (5)$$
$$W_v^{(m_v+1)} \cdot W_t^{(m_t+1)^T} = 0 \quad (6)$$

Inspired by [Wang *et al.*, 2010], the hard orthogonality constraints may reduce the quality. Thus instead of imposing hard orthogonality constraints, we add penalty terms on the objective function and propose the following final overall objective function:

$$\begin{aligned} \min_{\theta} \quad & L(X_v, X_t; \theta) = L_1 + \gamma \|W_v^{(m_v+1)} \cdot W_t^{(m_t+1)^T}\|_F^2 \\ & + \sum_{l=1}^{m_v+1} \alpha_l \|W_v^{(l)^T} W_v^{(l)} - I\|_F^2 + \sum_{l=1}^{m_t+1} \beta_l \|W_t^{(l)^T} W_t^{(l)} - I\|_F^2 \end{aligned} \quad (7)$$

3.4 Solution

We adopt back-propagation on Eq. 7 to fine-tune the parameters. We calculate the derivative of $W_v^{(m_v+1)}$ as an example [1]:

$$\frac{\partial L}{\partial W_v} = \frac{\partial L_1}{\partial W_v} + \gamma \frac{\partial \|W_v W_t^T\|_F^2}{\partial W_v} + \alpha \frac{\partial \|W_v^T W_v - I\|_F^2}{\partial W_v} \quad (8)$$

The calculation of the first term is the same as most basic autoencoders. The second and third terms are calculated as follows:

$$\begin{aligned} \frac{\partial \|W_v W_t^T\|_F^2}{\partial W_v} &= \frac{\partial \mathbf{tr}[(W_v W_t^T)^T (W_v W_t^T)]}{\partial W_v} = 2 W_v W_t^T W_t \\ \frac{\partial \|W_v^T W_v - I\|_F^2}{\partial W_v} &= \frac{\partial \mathbf{tr}[(W_v^T W_v - I)^T (W_v^T W_v - I)]}{\partial W_v} \\ &= 4 \times (W_v W_v^T - I) W_v \end{aligned} \quad (9)$$

The update of other parameters follows a similar way. The fine-tuning algorithm is presented in Algorithm 1.

After finishing training all of the parameters, we can use Eq.2 to derive representations for any samples. We use the median value of all the training samples as the threshold δ to perform binarization and generate hash codes.

[1] Here, for simplicity, $W_v^{(m_v+1)}$ and $W_t^{(m_t+1)}$ is simply denoted as W_v and W_t

Algorithm 1 Fine-tuning for **DMHOR**

Input: X_t, X_v, θ;
Output: New Parameters: θ

repeat
  for batch (B_v,B_t) in (X_v,X_t) **do**
    Apply Eq.4 to get $L_{v\bar{t}}(B_v, B_t; \theta)$, $L_{\bar{v}t}(B_v, B_t; \theta)$.
    Apply Eq.3 to get $L_{vt}(B_v, B_t; \theta)$.
    Apply Eq.7 to get $L(B_v, B_t; \theta)$
    Use $\partial L(B_v, B_t; \theta)/\partial\theta$ to back-propagate through the entire network to get new θ
  end for
until converge

3.5 Complexity Analysis

The overall complexity is composed of training complexity and online test complexity. For online complexity, the calculation of the hash codes can be performed using a few matrix multiplications, which is fast and is linear with the number of query data and irrelevant with the size of training data [Salakhutdinov and Hinton, 2009].

For pre-training, we suppose that each RBM is pre-trained for k_1 epochs. Thus, the computational cost of updating the weights and bias for the l-th RBM in the image pathway is $O\big(nk_1(s_v^{(l)}s_v^{(l+1)})\big)$. The cost is similar for other layers or text pathway. For fine-tuning with k_2 epochs, the process is almost the same. Then the overall training complexity is:

$$O\big(n(k_1+k_2)\cdot\sum_{i\in\{v,t\}}(\sum_{l=1}^{m_i}s_i^{(l)}s_i^{(l+1)}+M\cdot s_i^{(m_i+1)})\big)$$

Therefore, the training time complexity is linear to the size of the training data. These complexities guarantee the good scalability of **DMHOR**.

4 EXPERIMENTS

4.1 Dataset

In our work, two real-world datasets are used for evaluation.

NUS-WIDE [Chua *et al.*, 2009] is a public web image dataset, which consists of 269,648 images from Flickr. These images are surrounded by tags, with a total of 5,018 unique tags. The ground-truth for 81 concepts is available. Two images are regarded to be similar if they share common concept and vice versa. In our experiment, we select images belonging to the 10 largest concepts and the 1000 most frequent tags. We randomly choose 30,000 images for training, 2,000 images for test and 100,000 images as database. For image features, many features are proposed and some research discussed about their own characteristics [Wang *et al.*, 2014b]. Since our paper does not focus on the comparison of features, we use one of the best known image features SIFT [Lowe, 1999] to form 500-dimensional bag of words. Texts are represented by 1000-dimensional tag occurrence vectors.

WIKI [Rasiwasia *et al.*, 2010] is a web document dataset, which has 2,866 documents from Wikipedia. Each document is accompanied by an image and is labelled with one of the ten semantic classes. If two documents share the same class, they are regarded to be similar. The images are represented by 128-dimensional SIFT vectors and texts are represented by 100-dimensional vectors based on LDA[Blei *et al.*, 2003]. 80% samples of the dataset are chosen as training set and the rest is used for testing. The training set is also used as the database due to the limited samples in this dataset.

4.2 Experiment Settings

For both datasets, the model consists of a 6-layer image pathway, a 4-layer text pathway and a joint RBM. The number of units in each layer is summarized in Table 2. The RBMs for the first layer are different and corresponded with input types. In NUS-WIDE, the RBM is a Gaussian RBM [Welling *et al.*, 2004] for real-valued image input and a Bernoulli RBM for binary text input. In WIKI, the RBMs for image and text input are both Gaussian RBMs. Other layer's RBMs are all Bernoulli RBMs. We run the following experiments with implementation in Matlab on a machine running Windows Server 2008 with 12 2.39GHz cores and 192 GB of memory. The hyper-parameters of λ, μ and ν are set as 0.5, 0.5 and 0.001 by using grid search. The value of α_l, β_l and γ are discussed later.

Table 2: Number of units on NUS-WIDE and WIKI

Dataset	Image Pathway	Text Pathway
NUS-WIDE	500-512-256-128-64-32	1000-1024-512-128
WIKI	100-256-128-64-32-32	100-256-128-32

4.3 Baseline and Evaluation Metrics

Our task mainly focuses on the multi-source hashing, which is defined in [Zhang and Li, 2014]. As most hash methods apply, we use *precision*, *recall* and *Mean Average Precision (MAP)* as evaluation metrics. Their definitions and calculations are the same as in [Ou *et al.*, 2013]. For multi-source hashing task, we choose **Bimodal DBN** [Srivastava and Salakhutdinov, 2012a], **Cross-modality AE** [Ngiam *et al.*, 2011b], **DMVH** [Kang *et al.*, 2012], **CHMIS** [Zhang *et al.*, 2011], **CVH** [Kumar and Udupa, 2011] and **PDH** [Rastegari *et al.*, 2013] as baseline methods. The first three baseline methods are deep learning methods and the rest are shallow-layer models. Note that **DMVH** is supervised in fine-tuning. Therefore, we apply multimodal autoencoder to make a fair comparison. Parameters and experiment settings of all deep-structured methods are the same as those for our method. We run each algorithm five times and report the following results.

4.4 Results

First, Table 3 shows the MAP when we vary the number of hashing bits in $\{8, 12, 16, 20, 32\}$ in both dataset.

From these comparison results, some observations and analysis are included as follows:

- The deep-structured models can consistently and obviously outperform other shallow-structured models, which demonstrates that the multimodal correlations cannot be well captured by shallow-structured model, and deep models have much merit in this aspect due to its intrinsic nonlinearity.
- The result shows that **DMHOR** outperforms **Cross-modality AE**, which demonstrates that removing the redundancy from hashing is critical in improving the per-

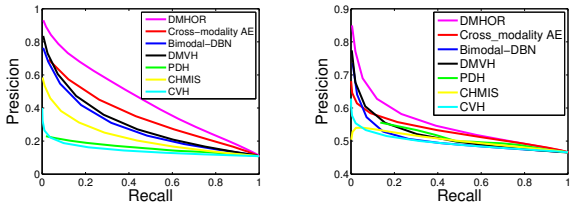
Table 3: MAP on WIKI and NUS-WIDE with varying length of hash codes

Method	WIKI					NUS-WIDE				
	8 bit	12 bit	16 bit	20 bit	32 bit	8 bit	12 bit	16 bit	20 bit	32 bit
DMHOR	**0.3424**	**0.4693**	**0.489**	**0.5033**	**0.5268**	**0.5472**	**0.5543**	**0.5618**	**0.5702**	**0.5791**
Cross-modality AE	0.306	0.3795	0.3922	0.4251	0.4478	0.5314	0.5397	0.5476	0.5482	0.5508
Bimodal-DBN	0.2695	0.3344	0.3727	0.3941	0.4035	0.5252	0.536	0.5392	0.5339	0.5386
DMVH	0.2847	0.3241	0.3543	0.3843	0.3960	0.5211	0.5274	0.5288	0.5342	0.5336
CHMIS	0.2171	0.2491	0.2672	0.2731	0.2697	0.5199	0.5276	0.5284	0.5308	0.5294
CVH	0.17	0.1665	0.1649	0.1615	0.1824	0.5088	0.5009	0.4961	0.4928	0.4868
PDH	0.1618	0.1622	0.1602	0.2412	0.2532	0.5164	0.507	0.5169	0.5204	0.5223

formance, and the proposed orthogonal regularization method can well address the redundancy problem.

- When the length of codes increases, the performance of **DMHOR** improves significantly than other baseline methods improve. The reason is that our methods well reduce redundant information. Thus we can make use of the increasing bits to represent more useful information.

By fixing the length of hash codes to 16, we report the precision-recall curve as shown in Figure 2. It is clear that **DMHOR** performs best among the baseline methods.



(a) WIKI (b) NUS-WIDE

Figure 2: Precision-recall curve for WIKI and NUS-WIDE dataset, with a fixed bit number of 16.

4.5 Insights

Table 4: MAP for WIKI with varying orthogonality constraints with 16-bit codes

	α_6	α_5	α_4	α_3	α_2	α_1	γ	MAP
Exp1	0.5	0.5	0.5	0.5	2	5	0.5	**0.489**
Exp2	0.5	0.5	0.5	0.5	2	0	0.5	0.485
Exp3	0.5	0.5	0.5	0.5	0	0	0.5	0.478
Exp4	0.5	0.5	0.5	0	0	0	0.5	0.467
Exp5	0.5	0.5	0	0	0	0	0.5	0.45
Exp6	0.5	0	0	0	0	0	0.5	0.429
Exp7	0	0	0	0	0	0	0.5	0.391
Exp8	0.5	0.5	0.5	0.5	2	5	0	0.475

Here we give some insights about our proposed methods. Firstly, we evaluate how the orthogonality constraints affect the performance. In Eq. 7, we fix the values of β_l to 0.5 and change the value of α_l and γ to observe the change of MAP for WIKI as shown in Table 4. All of the parameters are optimally chosen.

The result shows that from Exp1 to Exp7 the performance gradually decreases, which demonstrates the effectiveness of all the orthogonal constraints on each layer's matrix as shown in Eq. 5. Furthermore, the result that Exp1 outperforms Exp8 demonstrates that the cross-modality constraint as shown in Eq. 6 is also necessary and effective.

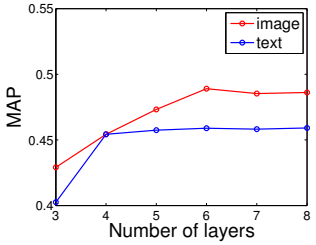


Figure 3: MAP for WIKI when setting the number of layers for one modality to 4 and varying the number of layers of another modality.

Now we will evaluate how the number of layers affects the performance. We set one modality to be 4 layers and change the number of layers for another modality to see the MAP. As shown in Figure 3, we find that the curve of text modality stabilizes faster than that for image modality. The explanation is that image features have larger semantic gap, thus we need to assign more layers to attain a better performance. However, for text modality, the structure needs not to be very deep otherwise it will waste time and space but obtain almost the same performance. Therefore, we need to assign an appropriate number of layers for different modalities. 4 layers for text modality and 6 layers for image modality is optimal for us.

5 CONCLUSIONS

In this paper, we propose a novel Deep Multimodal Hashing with Orthogonal Regularization (**DMHOR**) for performing similarity search on multimodal data. The proposed model with orthogonal regularization solves the redundancy problem. Furthermore, our strategy of applying different numbers of layers to different modalities makes a more precise representation and more compact learning process. Experimental results demonstrate a substantial gain of our method compared with state-of-the-art on two widely used public datasets. Our future work will aim at automatically determining the ideal number of layers for different modalities.

6 Acknowledgement

This work was supported in part by the National Basic Research Program of China under Grant No. 2015CB352300; National Natural Science Foundation of China, No. 61370022 and No. 61210008. Thanks for the support of NExT Research Center funded by MDA, Singapore, under the research grant, WBS:R-252-300-001-490. Thanks for the support of Beijing Key Laboratory of Networked Multimedia, Tsinghua University, China.

References

[Bengio, 2009] Yoshua Bengio. Learning deep architectures for ai. *Foundations and trends® in Machine Learning*, 2(1):1–127, 2009.

[Blei *et al.*, 2003] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *JMLR*, 3:993–1022, 2003.

[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. IEEE, 2010.

[Chua *et al.*, 2009] Tat-Seng Chua, Jinhui Tang, Richang Hong, Haojie Li, Zhiping Luo, and Yantao Zheng. Nus-wide: a real-world web image database from national university of singapore. In *CIVR*, page 48. ACM, 2009.

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

[Erhan *et al.*, 2010] Dumitru Erhan, Yoshua Bengio, Aaron Courville, Pierre-Antoine Manzagol, Pascal Vincent, and Samy Bengio. Why does unsupervised pre-training help deep learning? *JMLR*, 11:625–660, 2010.

[Feng *et al.*, 2014] Fangxiang Feng, Xiaojie Wang, and Ruifan Li. Cross-modal retrieval with correspondence autoencoder. In *ACM MM*, pages 7–16. ACM, 2014.

[Hinton *et al.*, 2006] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.

[Hinton *et al.*, 2012] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.

[Hinton, 2002] Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.

[Kang *et al.*, 2012] Yoonseop Kang, Saehoon Kim, and Seungjin Choi. Deep learning to hash with multiple representations. In *ICDM*, pages 930–935, 2012.

[Kumar and Udupa, 2011] Shaishav Kumar and Raghavendra Udupa. Learning hash functions for cross-view similarity search. In *IJCAI*, volume 22, page 1360, 2011.

[Liu *et al.*, 2012] Wei Liu, Jun Wang, Rongrong Ji, Yu-Gang Jiang, and Shih-Fu Chang. Supervised hashing with kernels. In *CVPR*, pages 2074–2081. IEEE, 2012.

[Lowe, 1999] David G Lowe. Object recognition from local scale-invariant features. In *ICCV*, volume 2, pages 1150–1157. Ieee, 1999.

[Lv *et al.*, 2007] Qin Lv, William Josephson, Zhe Wang, Moses Charikar, and Kai Li. Multi-probe lsh: efficient indexing for high-dimensional similarity search. In *VLDB*, pages 950–961, 2007.

[Ngiam *et al.*, 2011a] Jiquan Ngiam, Adam Coates, Ahbik Lahiri, Bobby Prochnow, Quoc V Le, and Andrew Y Ng. On optimization methods for deep learning. In *ICML*, pages 265–272, 2011.

[Ngiam *et al.*, 2011b] Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y Ng. Multimodal deep learning. In *ICML*, pages 689–696, 2011.

[Norouzi and Blei, 2011] Mohammad Norouzi and David M Blei. Minimal loss hashing for compact binary codes. In *ICML*, pages 353–360, 2011.

[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. ACM, 2013.

[Ou *et al.*, 2015] Mingdong Ou, Peng Cui, Jun Wang, Fei Wang, and Wenwu Zhu. Probabilistic attributed hashing. In *AAAI*, 2015.

[Rasiwasia *et al.*, 2010] Nikhil Rasiwasia, Jose Costa Pereira, Emanuele Coviello, Gabriel Doyle, Gert RG Lanckriet, Roger Levy, and Nuno Vasconcelos. A new approach to cross-modal multimedia retrieval. In *ACM MM*, pages 251–260. ACM, 2010.

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

[Salakhutdinov and Hinton, 2009] Ruslan Salakhutdinov and Geoffrey Hinton. Semantic hashing. *IJAR*, 50(7):969–978, 2009.

[Song *et al.*, 2013] Jingkuan Song, Yang Yang, Yi Yang, Zi Huang, and Heng Tao Shen. Inter-media hashing for large-scale retrieval from heterogeneous data sources. In *SIGMOD*, pages 785–796. ACM, 2013.

[Srivastava and Salakhutdinov, 2012a] Nitish Srivastava and Ruslan Salakhutdinov. Learning representations for multimodal data with deep belief nets. In *ICML Workshop*, 2012.

[Srivastava and Salakhutdinov, 2012b] Nitish Srivastava and Ruslan Salakhutdinov. Multimodal learning with deep boltzmann machines. In *NIPS*, pages 2231–2239, 2012.

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

[Wang *et al.*, 2014a] Wei Wang, Beng Chin Ooi, Xiaoyan Yang, Dongxiang Zhang, and Yueting Zhuang. Effective multi-modal retrieval based on stacked auto-encoders. *Proceedings of the VLDB Endowment*, 7(8), 2014.

[Wang *et al.*, 2014b] Zhiyu Wang, Peng Cui, Fangtao Li, Edward Chang, and Shiqiang Yang. A data-driven study of image feature extraction and fusion. *Information Sciences*, 281:536–558, 2014.

[Weiss *et al.*, 2008] Yair Weiss, Antonio Torralba, and Robert Fergus. Spectral hashing. In *NIPS*, volume 9, page 6, 2008.

[Welling *et al.*, 2004] Max Welling, Michal Rosen-Zvi, and Geoffrey E Hinton. Exponential family harmoniums with an application to information retrieval. In *NIPS*, pages 1481–1488, 2004.

[Xia *et al.*, 2014] Rongkai Xia, Yan Pan, Hanjiang Lai, Cong Liu, and Shuicheng Yan. Supervised hashing for image retrieval via image representation learning. In *AAAI*, 2014.

[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*, 2014.

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

[Zhen and Yeung, 2012] Yi Zhen and Dit-Yan Yeung. A probabilistic model for multimodal hash function learning. In *SIGKDD*, pages 940–948. ACM, 2012.