Deep Visual-Semantic Hashing for Cross-Modal Retrieval

Yue Cao[†], Mingsheng Long[†*], Jianmin Wang[†], Qiang Yang[♯], and Philip S. Yu[†‡]
[†]School of Software, Tsinghua National Laboratory (TNList), Tsinghua University, Beijing, China
[‡]Institute for Data Science, Tsinghua University & University of Illinois at Chicago, IL, USA
[♯]Department of Computer Science and Engineering, Hong Kong University of Science and Technology
caoyue10@gmail.com, {mingsheng, jimwang}@tsinghua.edu.cn
qyang@cse.ust.hk, psyu@uic.edu

ABSTRACT

Due to the storage and retrieval efficiency, hashing has been widely applied to approximate nearest neighbor search for large-scale multimedia retrieval. Cross-modal hashing, which enables efficient retrieval of images in response to text queries or vice versa, has received increasing attention recently. Most existing work on cross-modal hashing does not capture the spatial dependency of images and temporal dynamics of text sentences for learning powerful feature representations and cross-modal embeddings that mitigate the heterogeneity of different modalities. This paper presents a new Deep Visual-Semantic Hashing (DVSH) model that generates compact hash codes of images and sentences in an end-to-end deep learning architecture, which capture the intrinsic cross-modal correspondences between visual data and natural language. DVSH is a hybrid deep architecture that constitutes a visual-semantic fusion network for learning joint embedding space with images and text sentences, and two modality-specific hashing networks for learning hash functions with compact binary codes. Our architecture effectively unifies joint multimodal embedding with cross-modal hashing, which is based on a novel combination of Convolutional Neural Networks over images, Recurrent Neural Networks over sentences, and a structured max-margin objective that integrates all things together to enable learning of similarity-preserving and high-quality hash codes. Extensive empirical evidence shows that our DVSH approach yields state of the art results in cross-modal retrieval experiments on image-sentences datasets, i.e. standard IAPR TC-12 and large-scale Microsoft COCO.

Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing—*Indexing methods*; I.2.6 [**Artificial Intelligence**]: Learning—*Deep learning*

Keywords

Deep hashing, cross-modal retrieval, multimodal embedding

KDD '16, August 13–17, 2016, San Francisco, California

DOI: 10.1145/1235

1. INTRODUCTION

While multimedia big data of massive volumes and high dimensions are pervasive in search engines and social networks, it has attracted increasing attention to approximate nearest neighbors search across different media modalities that brings both computation efficiency and search quality. Since correspondence data from different modalities may endow semantic correlations, it is desirable to support cross-modal retrieval that returns relevant results of one modality in response to a query of different modality, e.g. retrieval of images with text queries. An advantageous solution to cross-modal retrieval is hashing methods, which compress high-dimension data in compact binary codes and generate similar binary codes for similar objects [32]. This paper focuses on cross-modal hashing that builds isomorphic hash codes for efficient cross-media retrieval. To date, effective and efficient cross-modal hashing remains a challenge, due to the heterogeneity across different modalities [34], and the semantic gap between feature descriptors and semantics [28].

Many cross-modal hashing methods have been proposed to exploit shared structures across different modalities in the process of hash function learning and indexes cross-modal data into an isomorphic Hamming space [3, 20, 40, 41, 29, 33, 37, 25, 39, 35, 23, 26]. These cross-modal hashing methods based on shallow architectures cannot effectively exploit the *heterogeneous* correlation structure to bridge different modalities. Several recent deep models for multimodal embedding [8, 18, 16, 5, 9] show that deep learning can capture heterogeneous cross-modal correlations more effectively than shallow learning methods. Although these deep models have been successfully applied to image captioning and retrieval, they cannot generate compact hash codes for efficient cross-modal retrieval. Meanwhile, latest deep hashing methods [36, 21, 4] have yielded state of art results on many datasets, but these methods are limited to single-modal retrieval.

In this work, we strive to take a step further towards the goal of efficient cross-modal retrieval of images in response to sentence queries or vice versa, as shown in Figure 1. This novel hashing scenario is more desirable for practical applications, as it is usually easier for users to describe the images one intends to search by a free-style text sentence instead of a couple of keywords. The primary challenge towards this goal is in the design of a model that is rich enough to simultaneously reason about contents of images and their representation in the domain of natural language. Additionally, the model should be able to generate compact hash codes that capture the rich features of the images and sentences as well as the cross-modal correlation structures to enable

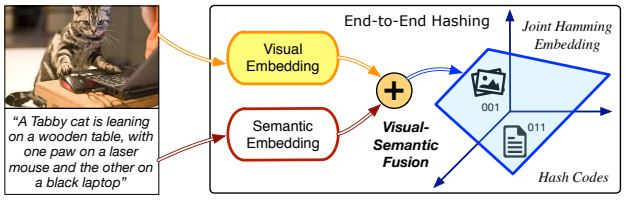


Figure 1: Deep visual-semantic hashing (DVSH) for cross-modal retrieval of images and text sentences.

efficient cross-modal retrieval. To our best knowledge, this work is the first end-to-end learning approach to cross-modal hashing that enables efficient cross-modal retrieval of images in response to sentence queries and vice versa.

This paper presents a new Deep Visual-Semantic Hashing (DVSH) model that generates compact hash codes of images and sentences in an end-to-end deep learning architecture, which capture the spatial dependency of images and temporal dynamics of text sentences for learning powerful feature representations and cross-modal embeddings that mitigate the heterogeneity of different modalities. DVSH is a hybrid deep architecture that constitutes a visual-semantic fusion network for learning joint embedding space with images and text sentences, and two modality-specific hashing networks for learning hash functions with compact binary codes. Our architecture effectively unifies joint multimodal embedding with cross-modal hashing, which is based on a novel combination of Convolutional Neural Networks over images, Recurrent Neural Networks over sentences, and a structured max-margin objective that integrates all things together to enable the learning of similarity-preserving and high-quality hash codes. Extensive empirical evidence shows that our DVSH model yields state of the art results in cross-modal retrieval experiments on popular image-sentences datasets, i.e. standard IAPR TC-12 and large-scale Microsoft COCO.

2. RELATED WORK

This work is related to cross-modal hashing, which has been an increasingly popular research topic in machine learning, computer vision, and multimedia retrieval communities [3, 20, 40, 41, 29, 27, 33, 37, 7, 14, 39, 25, 35, 23]. We refer readers to [32] for a comprehensive and up-to-date survey.

Prior cross-modal hashing methods can be roughly organized into unsupervised methods and supervised methods. Unsupervised hashing methods learn hash functions that can encode input data points to binary codes only using the unlabeled training data. Typical learning criteria include reconstruction error minimization [7, 33], neighborhood preserving as graph-based hashing [20, 29], and quantization error minimization as correlation quantization [26]. Supervised hashing explores supervised information (e.g., class labels, relative similarity, or relevance feedback) to learn compact hash coding. Typical learning methods include metric learning [3, 23], neural network [27], and correlation analysis [39, 35]. Since supervised methods can explore the semantic labels to enhance the cross-modal correlations and reduce the semantic gap [28], they can achieve superior accuracy than unsupervised methods for cross-modal similarity retrieval.

Most of previous cross-modal hashing methods based on shallow architectures cannot effectively exploit the *heterogeneous* correlation structure across different modalities. Some latest deep models for multimodal embedding [8, 18, 16, 5, 9] have shown that deep learning can capture heterogeneous cross-modal correlations more effectively for image captioning and cross-modal retrieval, but it remains unclear how to explore these deep models to cross-modal hashing. Recent deep hashing methods [36, 21, 4] have given the latest state of the art results on many datasets, but these methods can only be used for single-modal retrieval. To our knowledge, this work is the first end-to-end learning approach to cross-modal deep hashing that enables efficient cross-modal retrieval of images given text-sentences queries and vice versa.

3. PRELIMINARY ON DEEP NETWORKS

3.1 Convolutional Neural Network (CNN)

To learn deep representation of visual data, we start with AlexNet [19], the deep convolutional network (CNN) architecture which won the ImageNet ILSVRC 2012 challenge. AlexNet comprises five convolutional layers ($conv1$–$conv5$) and three fully connected layers ($fc6$–$fc8$), as in Figure 3. Each fully-connected layer ℓ learns a nonlinear mapping $\boldsymbol{h}^\ell = a^\ell\left(\boldsymbol{W}^\ell \boldsymbol{h}^{\ell-1} + \boldsymbol{b}^\ell\right)$, where $\boldsymbol{h}^\ell$ is the ℓ-th layer activation of image $\boldsymbol{x}$, $\boldsymbol{W}^\ell$ and $\boldsymbol{b}^\ell$ are the weight and bias parameters of the ℓ-th layer, and a^ℓ is the activation function, taken as rectifier units (ReLU) $a^\ell(\boldsymbol{x}) = \max(\boldsymbol{0}, \boldsymbol{x})$ for all feature layers $conv1$–$fc7$. Different from fully connected layers, each convolutional layer is a three-dimensional array of size $h \times w \times d$, where h and w are spatial dimensions, and d is the feature or channel dimension. The first layer is input image, with pixel size $h \times w$ and d color channels. Locations in higher convolutional layers correspond to the locations in the image they are connected to, which are called the receptive fields.

CNNs are built on translation invariance [5]. Their basic components (convolution, pooling, and activation functions) operate on local input regions, and depend only on relative spatial coordinates. Writing $\boldsymbol{x}_{ij}$ for the image vector at location (i, j) in a particular layer, and $\boldsymbol{h}_{ij}$ for the following layer, these functions in convolutional layers compute $\boldsymbol{h}_{ij}$ by

$$\boldsymbol{h}_{ij} = f_{ks}\left(\{\boldsymbol{x}_{si+\delta i, sj+\delta j}\}_{0 \le \delta i, \delta j \le k}\right), \tag{1}$$

where k is called the kernel size, s is the stride or subsampling factor, and f_{ks} determines the layer type: a matrix multiplication for convolution or average pooling, a spatial max for max pooling, or an elementwise nonlinearity for an activation function, and so on for other types of layers. This functional form is maintained under composition, with the kernel size and stride obeying the transformation rule

$$f_{ks} \circ g_{k's'} = (f \circ g)_{k'+(k-1)s', ss'}. \tag{2}$$

While a general deep network computes a general nonlinear function, a network with only layers of this form computes a nonlinear filter, which we call a deep filter or feature map.

3.2 Long Short-Term Memory (LSTM)

To learn deep representation of sequential data, we adopt Long Short-Term Memory (LSTM) recurrent neural network [13]. Though recurrent neural networks (RNNs) have proven successful on tasks such as speech recognition and text generation, it can be difficult to train them to learn long-term dynamics, likely due in part to the vanishing and exploding gradients problem that can result from propagating the

gradients down through the many layers of the recurrent network, each corresponding to a particular timestep. LSTMs provide a solution by incorporating memory units that allow the network to learn when to forget previous hidden states and when to update hidden states given new information.

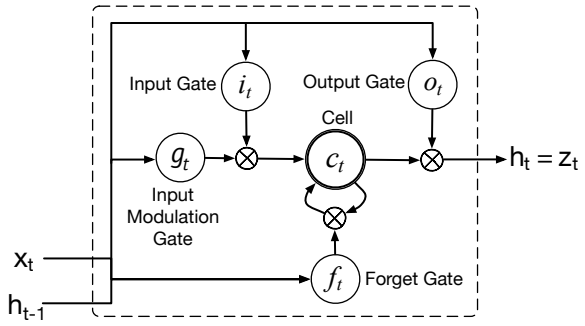


Figure 2: A diagram of an LSTM memory cell.

In this paper, we adopt the LSTM unit as described in [31, 38, 5], which is a slight simplification of the one described in [10], as shown in Figure 2. Let $\sigma(x) = \frac{1}{1+\exp^{-x}}$ be the sigmoid function that maps real-valued inputs to $[0, 1]$, and let $\phi(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} = 2\sigma(2x) - 1$ be the hyperbolic tangent (tanh) function, similarly mapping its inputs to $[-1, 1]$, the LSTM updates for timestep t given inputs $\boldsymbol{x}_t$, $\boldsymbol{h}_{t-1}$ and $\boldsymbol{c}_{t-1}$:

$$\begin{aligned}
\boldsymbol{i}_t &= \sigma\left(\boldsymbol{W}_{xi}\boldsymbol{x}_t + \boldsymbol{W}_{hi}\boldsymbol{h}_{t-1} + \boldsymbol{b}_i\right) \\
\boldsymbol{f}_t &= \sigma\left(\boldsymbol{W}_{xf}\boldsymbol{x}_t + \boldsymbol{W}_{hf}\boldsymbol{h}_{t-1} + \boldsymbol{b}_f\right) \\
\boldsymbol{o}_t &= \sigma\left(\boldsymbol{W}_{xo}\boldsymbol{x}_t + \boldsymbol{W}_{ho}\boldsymbol{h}_{t-1} + \boldsymbol{b}_o\right) \\
\boldsymbol{g}_t &= \phi\left(\boldsymbol{W}_{xc}\boldsymbol{x}_t + \boldsymbol{W}_{hc}\boldsymbol{h}_{t-1} + \boldsymbol{b}_c\right) \\
\boldsymbol{c}_t &= \boldsymbol{f}_t \odot \boldsymbol{c}_{t-1} + \boldsymbol{i}_t \odot \boldsymbol{g}_t \\
\boldsymbol{h}_t &= \boldsymbol{o}_t \odot \phi\left(\boldsymbol{c}_t\right),
\end{aligned} \tag{3}$$

where $\boldsymbol{i}_t$, $\boldsymbol{f}_t$, $\boldsymbol{o}_t$, $\boldsymbol{g}_t$, $\boldsymbol{c}_t$, $\boldsymbol{h}_t$ are respectively *input gate*, *forget gate*, *output gate*, *input modulation gate*, *memory cell* and *hidden unit* for timestep t. The weight matrix has the obvious meaning that $\boldsymbol{W}_{xf}$ is the input-forget gate matrix and $\boldsymbol{W}_{hi}$ is the hidden-input gate matrix, etc. Because the activation function of $\boldsymbol{f}_t$ and $\boldsymbol{i}_t$ is sigmoid function, their values are in $[0, 1]$, and they are learned to control how much of the memory cell to forget its previous memory or consider their current inputs. Similarly, the output gate $\boldsymbol{o}_t$ learns that how much the memory cell transfers to hidden unit. Considering the memory cell, which is a summation of two parts: the previous memory cell $\boldsymbol{c}_{t-1}$ which is modulated by the forget gate $\boldsymbol{f}_t$, and $\boldsymbol{g}_t$ which is modulated by the input gate $\boldsymbol{i}_t$. These additional gates enable LSTM to learn much complex and long-term temporal dynamics which cannot gain from RNN. Additional depth can be added to LSTMs by stacking them on top of each other, using the hidden state of the LSTM in layer $(\ell - 1)$ as the input to the LSTM in layer ℓ.

The advantages of LSTMs for modeling sequential data in vision and natural language problems are: (1) when integrated with current vision systems, LSTM models are straightforward to fine-tune end-to-end. (2) LSTMs are not confined to fixed length inputs or outputs allowing simple modeling for sequential data of varying lengths, such as text or video.

4. DEEP VISUAL-SEMANTIC HASHING

In cross-modal retrieval systems, the database consists of objects from one modality and the query consists of objects from another modality. In this paper, we study a novel cross-modal hashing scheme, where we are given image-sentence pairs each corresponding to an image and a text sentence that correctly describes the image. We uncover the correlation structure between images and texts by learning from a training set of N bimodal objects $\{\boldsymbol{o}_i = (\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^N$, where $\boldsymbol{x}_i \in \mathbb{R}^{D_x}$ denotes the D_x-dimensional feature vector of the image modality, and $\boldsymbol{y}_i = <\boldsymbol{y}_{i1}, \boldsymbol{y}_{i2}, ..., \boldsymbol{y}_{iT}> \in \mathbb{R}^{D_y \times T}$ denotes sentence i consisting of word sequences, where $\boldsymbol{y}_{it} \in \mathbb{R}^{D_y}$ is a one-hot vector that denotes a word of time t in sentence i (the nonzero element of $\boldsymbol{y}_{it}$ denotes the index of the word in the vocabulary of size D_y). Some pairs of the bimodal objects are associated with similarity labels s_{ij}, where $s_{ij} = 1$ implies $\boldsymbol{o}_i$ and $\boldsymbol{o}_j$ are similar and $s_{ij} = -1$ indicates $\boldsymbol{o}_i$ and $\boldsymbol{o}_j$ are dissimilar. In supervised cross-modal hashing, $\mathcal{S} = \{s_{ij}\}$ is constructed from the semantic labels of data points or the relevance feedback from click-through data.

We propose a novel Deep Visual-Semantic Hashing (DVSH) approach to cross-modal retrieval, which learns end-to-end (1) a bimodal fusion function $f(\boldsymbol{x}, \boldsymbol{y}) : \left(\mathbb{R}^{D_x}, \mathbb{R}^{D_y \times T}\right) \mapsto \{-1, 1\}^K$, which maps images and texts into a K-dimensional joint embedding space $\mathcal{H}$ such that the embeddings of each image-sentence pair are tightly fused to bridge different modalities whilst the similarity information conveyed in the given bimodal object pairs $\mathcal{S}$ is preserved; and (2) two modality-specific hashing functions $f_x(\boldsymbol{x}) : \mathbb{R}^{D_x} \mapsto \{-1, 1\}^K$ and $f_y(\boldsymbol{y}) : \mathbb{R}^{D_y \times T} \mapsto \{-1, 1\}^K$, which encode each image $\boldsymbol{x}$ and sentence $\boldsymbol{y}$ from database and query to compact binary hash codes $\boldsymbol{u} \in \{-1, 1\}^K$ and $\boldsymbol{v} \in \{-1, 1\}^K$ in the joint embedding space $\mathcal{H}$ to enable efficient cross-modal retrieval.

The proposed cross-modal deep hashing approach (DVSH) in Figure 3 is an end-to-end deep architecture for cross-modal hashing, which comprises both convolutional neural network (AlexNet) for learning image representations and recurrent neural network (LSTM) for learning text representations. The architecture accepts pairwise input $(\boldsymbol{o}_i, \boldsymbol{o}_j, s_{ij})$ and processes them in an end-to-end deep representation learning and hash coding pipeline: (1) a deep visual-semantic fusion network for learning isomorphic hash codes in the joint embedding space such that the representations of each image-sentence pair is tightly fused and correlated; (2) an image hashing network and a sentence hashing network for learning nonlinear modality-specific hash functions that encode each unseen image and sentence to compact hash codes in the joint embedding space; (3) a new cosine max-margin loss to preserve the pairwise similarity information and enhance the robustness to outliers; (4) a novel bitwise max-margin loss to control the quality of the binary hash codes.

4.1 Visual-Semantic Fusion Network

The challenge of cross-modal retrieval arises in that cross-modal data (images and texts) have significantly different statistical properties (heterogeneous), which makes it very difficult to capture the correlation across modalities based on hand-crafted features. Recently, it has been witnessed that deep learning methods [2], such as deep convolutional networks (CNNs) [19] and deep recurrent networks (RNNs) [31], have made performance breakthroughs on many real-world perception problems. Deep architectures are very powerful for extracting the multimodal embedding shared by different modalities since they can extract nonlinear feature represen-

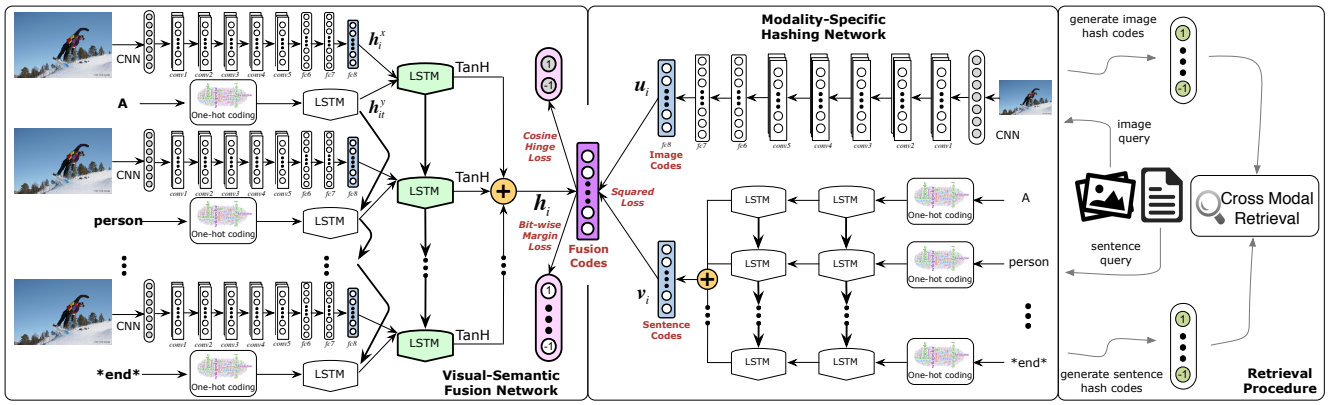


Figure 3: The architecture of *Deep Visual-Semantic Hashing* (DVSH), an end-to-end deep hashing approach to image-sentence cross-modal retrieval. The architecture comprises four key components: (1) a deep visual-semantic fusion network (unifying CNN and LSTM) for learning isomorphic hash codes in the joint embedding space; (2) an image hashing network (CNN) and a sentence hashing network (LSTM) for learning nonlinear modality-specific hash functions that map inputs to the joint embedding space; (3) a new cosine max-margin loss to preserve the pairwise similarity information; (4) a novel bitwise max-margin loss to control the quality of binary hash codes. Colored ones are modules modified or newly-crafted in this paper. *Best viewed in color.*

tations to bridge different modalities effectively [1, 8, 30, 17, 18, 5, 16]. We thus leverage deep networks for cross-modal joint embedding by designing a deep visual-semantic fusion network as illustrated in the left part of Figure 3, which maps the deep feature representations of images and texts in the shared visual-semantic embedding space such that the correspondence relation conveyed in the image-sentence pair can be maximized whilst the pairwise similarity information conveyed in the similarity labels can be preserved.

The proposed deep visual-semantic fusion network works by passing each visual input $\boldsymbol{x}_i$ (an image in our case) through the deep convolutional neural network (CNN) to produce a fixed-length vector representation $\boldsymbol{h}_i^x$. Note that, we replace the softmax classifier in the $fc8$ layer of the original AlexNet [19] with a feature map, which maps the image features from the $fc7$ layer to new features of K-dimension. We adopt the LSTM as our sequence model, which maps an input $\boldsymbol{y}_{it}$ of each sequence (a sentence in our case) at timestep t and a hidden state $\boldsymbol{h}_{i(t-1)}^y$ of previous timestep $(t-1)$ to an output $\boldsymbol{z}_{it}^y$ and updates hidden state $\boldsymbol{h}_{it}^y$. Therefore, inference must be run sequentially (i.e. from top to bottom in Figure 3), by computing the activations in order using Equation (3), that is, updating the t-th state based on the $(t-1)$-th state.

To integrate CNN and LSTM into a unified deep visual-semantic embedding model, the computed feature-space representation $\boldsymbol{h}_i^x$ of the visual input $\boldsymbol{x}_i$ is fused into the *second* layer of the LSTM model over each state, as illustrated in Figure 3. Specifically, the activation $\boldsymbol{h}_i$ of the fusion layer (the LSTMs with green color) for the t-th state (a word) in the sequence (text sentence) can be calculated as follows:

$$\boldsymbol{h}_{it} = f\left(\boldsymbol{h}_i^x + \boldsymbol{h}_{it}^y\right), \tag{4}$$

where $f(\cdot)$ denotes the updates made to the timestep t of the second-layer LSTM by substituting $\boldsymbol{x}_t \triangleq \boldsymbol{h}_i^x + \boldsymbol{h}_{it}^y$ into Equation (3). Note that, to reduce the gap between the activation $\boldsymbol{h}_{it}$ of the fusion layer and the final binary hash codes $\boldsymbol{u}_i$ and $\boldsymbol{v}_i$, we first squash the activations $\boldsymbol{h}_{it}$ to $[-1, 1]$ using the hyperbolic tangent (tanh) activation function $\phi(\boldsymbol{x}) = \tanh(\boldsymbol{x})$ in Equation (3). This fusion operation is very important to embody the multimodal visual-semantic embedding space.

The aforementioned timestep-wise fusion tights the visual and textual embeddings $\boldsymbol{h}_i^x$ and $\boldsymbol{h}_{it}^y$ to a unified embedding. However, each timestep t produces a joint embedding $\boldsymbol{h}_{it}$, while we would expect that each image-text pair produces only one fusion code to make cross-modal retrieval efficient. To this end, we are motivated by the technique of mean embeddings of distributions [11] and generate pair-level fusion code $\boldsymbol{h}_i$ for each image-sentence pair by weighted averaging:

$$\boldsymbol{h}_i = \frac{\sum_{t=1}^{T} \pi_{it} \boldsymbol{h}_{it}}{\sum_{t=1}^{T} \pi_{it}} = \frac{\sum_{t=1}^{T} \pi_{it} f\left(\boldsymbol{h}_i^x + \boldsymbol{h}_{it}^y\right)}{\sum_{t=1}^{T} \pi_{it}}, \tag{5}$$

where $\pi_{it} \in \{1, 0\}$ is the indicator variable, $\pi_{it} = 1$ if word t is present in timestep t, and $\pi_{it} = 0$ otherwise. We handle these cases because the text sentences are of variable-length and some sentences are shorter than the number T of states in the LSTMs. It is important to note that, the derived joint visual-semantic embedding $\boldsymbol{h}_i$ not only captures the spatial dependencies over images and temporal dynamics over sentences using CNN and LSTM respectively, but also captures the cross-modal relationship in a multimodal Hamming embedding space. To achieve an optimal joint embedding space for binary coding, the joint embeddings should be made to preserve the pairwise similarity information in training data $\mathcal{S}$ and to be separated well by bitwise hyperplane $h_{ik} = 0$.

4.1.1 Cosine Max-Margin Loss

In order to make the learned joint visual-semantic embeddings maximally preserve the similarity information across different modalities, we propose the following criterion: for each pair of objects $(\boldsymbol{o}_i, \boldsymbol{o}_j, s_{ij})$, if $s_{ij} = 1$, indicating that $\boldsymbol{o}_i$ and $\boldsymbol{o}_j$ are similar, then their hash codes $\boldsymbol{u}_i$ and $\boldsymbol{v}_j$ must be similar across different modalities (image and sentence), which is equivalent to requiring that their joint visual-semantic embeddings $\boldsymbol{h}_i$ and $\boldsymbol{h}_j$ should be similar. Correspondingly, if

$s_{ij} = -1$, indicating that $\boldsymbol{o}_i$ and $\boldsymbol{o}_j$ are dissimilar, then their joint visual-semantic embeddings $\boldsymbol{h}_i$ and $\boldsymbol{h}_j$ should be dissimilar. We use the cosine similarity $\cos(\boldsymbol{h}_i, \boldsymbol{h}_j) = \frac{\mathbf{h}_i \cdot \boldsymbol{h}_j}{\|\boldsymbol{h}_i\| \|\boldsymbol{h}_j\|}$ for measuring the closeness between $\boldsymbol{h}_i$ and $\boldsymbol{h}_j$, where $\boldsymbol{h}_i \cdot \boldsymbol{h}_j$ is the inner-product of $\boldsymbol{h}_i$ and $\boldsymbol{h}_j$, and $\|\cdot\|$ denotes the Euclidean norm of a vector. For similarity-preserving learning, we propose to minimize the following cosine max-margin loss

$$L = \sum_{s_{ij} \in \mathcal{S}} \max\left(0, \mu_c - s_{ij} \frac{\boldsymbol{h}_i \cdot \boldsymbol{h}_j}{\|\boldsymbol{h}_i\| \, \|\boldsymbol{h}_j\|}\right), \tag{6}$$

where $\mu_c > 0$ is the margin parameter, which is fixed to $\mu_c = 0.5$. This objective encourages similar image-sentences pairs to have a higher cosine similarity than dissimilar pairs, by a margin. Similar to the support vector machines (SVMs), the max-margin loss enhances the robustness to outliers. The cosine max-margin loss is particularly powerful for cross-modal correlation analysis, since the vector lengths are very diverse in different modalities and may make many distance metrics (e.g. Euclidean distance) as well as loss functions (e.g. squared loss) misspecified. To date this problem has not been studied in cross-modal deep hashing methods [32].

4.1.2 Bitwise Max-Margin Loss

For each image-sentence pair $\boldsymbol{o}_i = (\boldsymbol{x}_i, \boldsymbol{y}_i)$, to reduce the gap between its joint embedding $\boldsymbol{h}_i$ and its modality-specific binary codes $\boldsymbol{u}_i$ and $\boldsymbol{v}_i$, we require that the joint embedding $\boldsymbol{h}_i$ is close to its signed code $\text{sgn}(\boldsymbol{h}_i) \in \{-1, 1\}^K$, which is equivalent to minimizing $\||\boldsymbol{h}_i| - \mathbf{1}\|^2$. However, as a common knowledge, such squared loss is not robust to outliers. Thus we propose to minimize a novel bitwise max-margin loss as

$$Q = \sum_{i=1}^{N} \sum_{k=1}^{K} \max\left(0, \mu_b - |h_{ik}|\right) \tag{7}$$

where $\mu_b > 0$ is the bitwise margin parameter, which is fixed to $\mu_b = 0.5$. This objective encourages the joint embedding to separate apart from the hyperplane $h_{ik} = 0$ corresponding to the k-th bit, by a margin, hence we call it bitwise max-margin. Note that, minimizing the bitwise max-margin loss will lead to lower quantization error when binarizing the continuous embeddings $\boldsymbol{u}_i \in \mathbb{R}^K$ and $\boldsymbol{v}_i \in \mathbb{R}^K$ to binary hash codes, which allows us to learn high-fidelity binary codes.

4.2 Modality-Specific Hashing Network

The proposed deep visual-semantic fusion network will produce isomorphic joint embeddings that are sharable as the bridge to correlate different modalities, which effectively mitigates the cross-modal heterogeneity by deep representations of images and texts and the deep fusion between them. However, two major problems remain: (1) the fusion network cannot extend the embedding model to out-of-sample images and texts; (2) the fusion network require bimodal objects (both image and text modalities should be available) to predict the joint embeddings. In other words, the fusion network cannot be directly applied to cross-modal retrieval, where only one modality is available for the database or the query. Most importantly, it does not provide a mechanism to map each unimodal input to the joint embedding space. This thus motivates us to craft two more hashing networks for directly learning the modality-specific hashing functions. The key difference between the hashing network and the fusion network is: in the fusion network, we map each input to its modality-specific representation and then unify all modalities by elementwise summation in Equation (4); in the hashing network, however, we directly map each input to the joint embedding space learned by the fusion network. Hence the hashing network can address the above two problems.

4.2.1 Image Hashing Network

The image hashing network is crafted to learn the hashing function for the image modality. It is similar to the CNN module of the fusion network: we directly copy the $conv1$–$fc7$ layers from AlexNet [19], and replace the softmax classifier in $fc8$ layer with a hash function that transforms the feature representation of input image $\boldsymbol{x}_i$ to hash code $\boldsymbol{u}_i$. To guarantee that the hash code $\boldsymbol{u}_i$ produced by the hashing network lie in the joint embedding space, we require the hash code $\boldsymbol{u}_i$ and the joint embedding $\boldsymbol{h}_i$ corresponding to the same training image $\boldsymbol{x}_i$ to be close with the squared loss:

$$L^x = \frac{1}{2N} \sum_{i=1}^{N} \left(\boldsymbol{u}_i - \frac{\sum_{t=1}^{T} \pi_{it} \boldsymbol{h}_{it}}{\sum_{t=1}^{T} \pi_{it}} \right)^2. \tag{8}$$

4.2.2 Sentence Hashing Network

The sentence hashing network is crafted to learn the hashing function for the text modality. It is similar to the LSTM module of the fusion network, but by removing the visual input branch. We replace the softmax classifier in the output layer of the LSTM with a hash function that transforms the feature representation of input sentence $\boldsymbol{y}_i$ to hash code $\boldsymbol{v}_i$. Again, to guarantee that the hash code $\boldsymbol{v}_i$ lie in the joint embedding space, we require the hash code $\boldsymbol{v}_i$ and the joint embedding $\boldsymbol{h}_i$ corresponding to the same training sentence $\boldsymbol{y}_i$ to be similar in each timestep t under the squared loss:

$$L^y = \frac{1}{2N} \sum_{i=1}^{N} \frac{\sum_{t=1}^{T} \pi_{it} (\boldsymbol{v}_{it} - \boldsymbol{h}_{it})^2}{\sum_{t=1}^{T} \pi_{it}}. \tag{9}$$

Note that for both hashing networks, bimodal objects are only required in the training phase. After the hash functions are learned, we can directly encode any out-of-sample input.

4.3 Deep Visual-Semantic Hashing

In this paper, we enable joint representation learning and hash coding in an end-to-end deep architecture. Specifically, (1) we guarantee robust similarity-preserving representation learning by minimizing the cosine max-margin loss (6); (2) we guarantee the high-quality of compact binary hash codes by minimizing the bitwise max-margin loss (7); (3) we enable effective and efficient out-of-sample code generation by minimizing the squared losses (8)–(9). Integrating these loss functions in a joint optimization problem that is taken over the deep visual-semantic hashing (DVSH) network, it yields

$$\min_{\Theta} O = L + \lambda Q + \beta \left(L^x + L^y\right), \tag{10}$$

where $\Theta \triangleq \left\{\boldsymbol{W}_*^\ell, \boldsymbol{b}_*^\ell\right\}_{*\in\{x,y,u,v\}}$ denotes the set of network parameters, λ and β are the penalty parameters for trading off the relative importance of the bitwise max-margin loss and modality-specific squared loss. Through joint optimization (10) over the deep visual-semantic hashing network, we can jointly learn an isomorphic joint embedding space that effectively bridges the image and text modalities, and two

modality-specific hashing functions that respectively map the image and text inputs to compact binary codes in the joint embedding space, which enables effective and efficient cross-modal retrieval. With the trained fusion network and hashing network, we can obtain K-bit binary hash codes by simple sigh thresholding $\text{sgn}(\boldsymbol{u})$ and $\text{sgn}(\boldsymbol{v})$ for each modality, where $\text{sgn}(\cdot)$ is the element-wise sign function that for $i = 1, \ldots, K$, $\text{sgn}(z_i) = 1$ if $z_i > 0$, otherwise $\text{sgn}(z_i) = -1$. It is worth noting that, since we have minimized the bitwise max-margin loss in Equation (10) during training, this final binarization step will incur relatively small loss of retrieval quality, which will also be validated in the empirical study.

4.4 Algorithms and Training Details

The CNN module is pre-trained on the ImageNet classification task [19]. The LSTM module is pre-trained on the MS COCO dataset [22] using the neural language model [31]. These two components are fined-tuned during the training of the proposed DVSH model. We jointly train the new layers (colored modules in Figure 3) of visual-semantic fusion network and modality-specific hashing network with mini-batch stochastic gradient descent (SGD) method. And the hyper-parameters of the model are selected by cross-validation.

We derive the learning algorithms for the DVSH model in Equation (10), and show rigorously that both cosine max-margin loss and bitwise max-margin quantization loss can be optimized efficiently through the standard back-propagation (BP). For notation brevity, we define the point-wise loss as

$$O_i \triangleq \sum_{j:s_{ij}\in\mathcal{S}} L_{ij} + \lambda \sum_{k=1}^{K} Q_{ik} + \beta \left(L_i^x + L_i^y\right). \tag{11}$$

To improve the convergence stableness, we let the loss of hashing network makes no effect to the updates of the fusion network during the training of DVSH. We derive the gradient of point-wise loss O_i w.r.t. $\boldsymbol{W}_{x,k}^{\ell}$, the network parameter of the k-th unit of ℓ-th layer for the fusion network:

$$\begin{aligned}\frac{\partial O_i}{\partial \boldsymbol{W}_{x,k}^{\ell}} &= \sum_{j:s_{ij}\in\mathcal{S}} \frac{\partial L_{ij}}{\partial \boldsymbol{W}_{x,k}^{\ell}} + \lambda \frac{\partial Q_{ik}}{\partial \boldsymbol{W}_{x,k}^{\ell}} \\ &= \left(\sum_{j:s_{ij}\in\mathcal{S}} \frac{\partial L_{ij}}{\partial \hat{h}_{ik}^{\ell}} + \lambda \frac{\partial Q_{ik}}{\partial \hat{h}_{ik}^{\ell}}\right) \frac{\partial \hat{h}_{ik}^{\ell}}{\partial \boldsymbol{W}_{x,k}^{\ell}} \\ &= \delta_{x,ik}^{\ell} \boldsymbol{h}_i^{\ell-1},\end{aligned} \tag{12}$$

where $\hat{\boldsymbol{h}}_i^{\ell} = \boldsymbol{W}_x^{\ell} \boldsymbol{h}_i^{\ell-1} + \boldsymbol{b}_x^{\ell}$ is the out of the ℓ-th layer before activation $a^{\ell}(\cdot)$, $\delta_{x,ik}^{\ell} \triangleq \sum_{j:s_{ij}\in\mathcal{S}} \frac{\partial L_{ij}}{\partial \hat{h}_{ik}^{\ell}} + \lambda \frac{\partial Q_{ik}}{\partial \hat{h}_{ik}^{\ell}}$ is the point-wise *residual* term that measures how much the k-th unit in the ℓ-th layer is responsible for the error of point $\boldsymbol{x}_i$ in the network output. For an output unit k, we can measure the difference between the network's activation and the true target value, and use that to define the residual $\delta_{x,ik}^{l}$ as

$$\begin{aligned}\delta_{x,ik}^{l} = &\sum_{j\neq i:s_{ij}\in\mathcal{S}} \mathbb{I}\left(\mu_c - s_{ij} \frac{\boldsymbol{h}_i^l \cdot \boldsymbol{h}_j^l}{\left\|\boldsymbol{h}_i^l\right\| \left\|\boldsymbol{h}_j^l\right\|} > 0\right) \\ &\cdot \left[-s_{ij}\left(\frac{h_{jk}^l}{\left\|\boldsymbol{h}_i^l\right\| \left\|\boldsymbol{h}_j^l\right\|} - \frac{h_{ik}^l \left\langle \boldsymbol{h}_i^l, \boldsymbol{h}_j^l \right\rangle}{\left\|\boldsymbol{h}_i^l\right\|^3 \left\|\boldsymbol{h}_j^l\right\|}\right)\right] \dot{a}^l(\hat{h}_{ik}^l) \\ &+ \lambda \mathbb{I}\left(h_{ik} < 0\right) \mathbb{I}\left(\mu_b - |h_{ik}| > 0\right) \dot{a}^l(\hat{h}_{ik}^l),\end{aligned} \tag{13}$$

where l is the output layer of LSTMs, $\dot{a}^l(\cdot)$ is the derivative of the l-th layer activation function, and $\mathbb{I}(A)$ is an indicator function, $\mathbb{I}(A) = 1$ if A is true and $\mathbb{I}(A) = 0$ otherwise. For a hidden unit k in the $(\ell-1)$-th layer, we compute the residual $\delta_{x,ik}^{\ell-1}$ based on a weighted average of the errors of all the units $k' = 1, \ldots, n_{\ell-1}$ in the $(\ell-1)$-th layer that use $\boldsymbol{h}_i^{\ell-1}$ as an input, which is consistent with standard back-propagation,

$$\delta_{x,ik}^{\ell-1} = \left(\sum_{k'=1}^{n_{\ell-1}} \delta_{x,ik'}^{\ell} W_{x,kk'}^{\ell-1}\right) \dot{a}_x^{\ell-1}\left(\hat{h}_{ik}^{\ell-1}\right), \tag{14}$$

where $n_{\ell-1}$ is number of units in the $(\ell-1)$-th layer. The residuals in all layers can be computed by back-propagation.

For the hashing network, we derive the gradient of point-wise loss O_i w.r.t. $\boldsymbol{W}_{u,k}^{\ell}$ and $\boldsymbol{W}_{v,k}^{\ell}$, the network parameter of the k-th unit of ℓ-th layer in the hashing networks for image and sentence, respectively. The derivatives are as follows,

$$\begin{aligned}\frac{\partial O_i}{\partial \boldsymbol{W}_{u,k}^{\ell_u}} &= \beta \frac{\partial L_i^x}{\partial \boldsymbol{W}_{u,k}^{\ell_u}} = \beta \delta_{u,ik}^{\ell_u} \hat{\boldsymbol{u}}_i^{\ell_u - 1}, \\ \frac{\partial O_i}{\partial \boldsymbol{W}_{v,k}^{\ell_v}} &= \beta \frac{\partial L_i^y}{\partial \boldsymbol{W}_{v,k}^{\ell_v}} = \beta \delta_{v,ik}^{\ell_v} \hat{\boldsymbol{v}}_i^{\ell_v - 1},\end{aligned} \tag{15}$$

where $\hat{\boldsymbol{u}}_i^{\ell} = \boldsymbol{W}_u^{\ell} \boldsymbol{u}_i^{\ell-1} + \boldsymbol{b}_u^{\ell}$ is the ℓ-th layer output before activation $a^{\ell}(\cdot)$, $\delta_{u,ik}^{\ell} \triangleq \frac{\partial L_i^x}{\partial \hat{u}_{ik}^{\ell}}$ is the point-wise *residual* term that measures how much the k-th unit in the ℓ-th layer is responsible for the error of point $\boldsymbol{u}_i$ in the network output (similar definitions apply to the sentence hashing network):

$$\begin{aligned}\delta_{u,ik}^{l_u} &= \boldsymbol{u}_i^{l_u} - \boldsymbol{h}_{it}^{l} = \left(\boldsymbol{u}_i^{l_u} - \frac{\sum_{t=1}^{T} \pi_{it} \boldsymbol{h}_{it}^{l}}{\sum_{t=1}^{T} \pi_{it}}\right), \\ \delta_{v,ik}^{l_v} &= \frac{\sum_{t=1}^{T} \pi_{it} \left(\boldsymbol{v}_{it}^{l_v} - \boldsymbol{h}_{it}^{l}\right)}{\sum_{t=1}^{T} \pi_{it}},\end{aligned} \tag{16}$$

where l_u is the output layer of the image hashing network, and $\dot{a}^{l_u}(\cdot)$ is the derivative of the l_u-th layer activation function. For a hidden unit k in the $(\ell_u - 1)$-th layer, we compute the residual $\delta_{u,ik}^{\ell_u - 1}$ based on a weighted average of the errors of all the units $k' = 1, \ldots, n_{\ell_u - 1}$ in the $(\ell_u - 1)$-th layer that use $\boldsymbol{u}_i^{\ell_u - 1}$ as an input, which is consistent with standard BP.

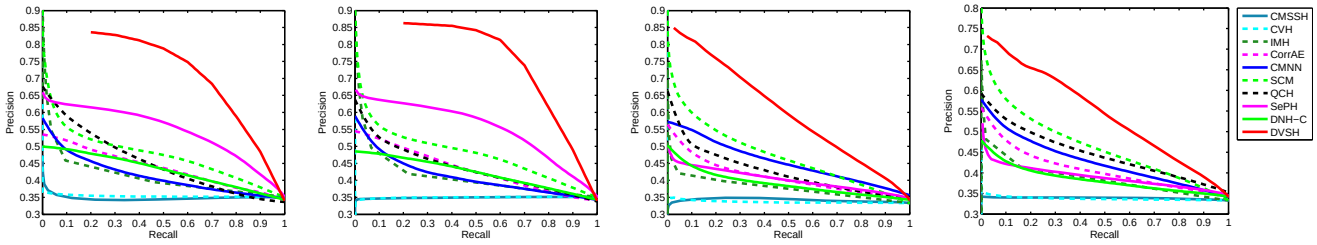
As the only differences between standard back-propagation (BP) and our algorithm are the residual terms defined in Equations (13)(16), we analyze the computational complexity for (13) and (16). Denote the number of similarity pairs $\mathcal{S}$ available for training as $|\mathcal{S}|$ and the number of bimodal objects available for training as N, then it is easy to verify that the overall computational complexity is $O(|\mathcal{S}| + N)$.

5. EXPERIMENTS

We conduct extensive experiments to evaluate the efficacy of the proposed DVSH model with several state of the art hashing methods on two widely-used benchmark datasets. Datasets, codes and configurations will be publicly available.

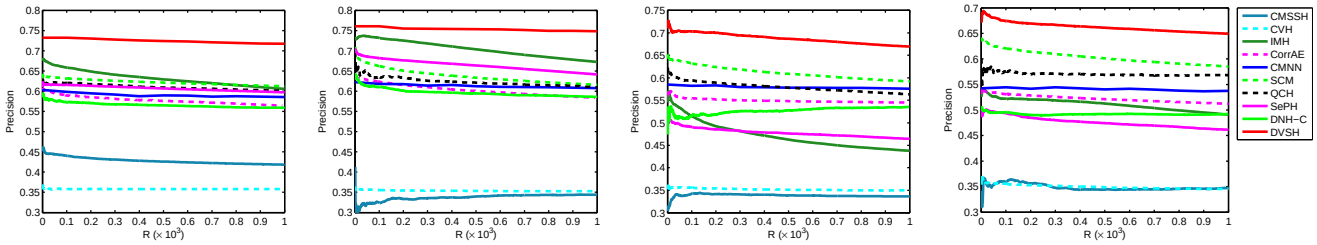
5.1 Evaluation Setup

The evaluation is conducted on two benchmark cross-modal datasets: **Microsoft COCO** [22] and **IAPR TC-12** [12].



(a) $I \rightarrow T$ on COCO (b) $T \rightarrow I$ on COCO (c) $I \rightarrow T$ on IAPR TC-12 (d) $T \rightarrow I$ on IAPR TC-12

Figure 4: Precision-recall curves of cross-modal retrieval on Microsoft COCO and IAPR TC-12 @ 32 bits.



(a) $I \rightarrow T$ on COCO (b) $T \rightarrow I$ on COCO (c) $I \rightarrow T$ on IAPR TC-12 (d) $T \rightarrow I$ on IAPR TC-12

Figure 5: Precision@top-R curves of cross-modal retrieval on Microsoft COCO and IAPR TC-12 @ 32 bits.

Microsoft COCO[1] The current release of this recently proposed dataset contains 82,783 training images and 5000 testing images. For each image, it provides five sentences annotations, belonging to 90 most frequent categories as ground truth labels. After pruning images with no category information, we get 82,120 training images and 4,960 testing images, from which we generate 410,600 training image-sentence pairs and 24,800 testing image-sentence pairs.

IAPR TC-12[2] This dataset consists of 20,000 images collected from a wide variety of domains, such as sports and actions, people, animals, cities, landscapes, and so forth. For each image, it provides at least one sentence annotation. On average there are about 1.7 sentence annotations for each image. Besides, it provides category annotations generated from segmentation tasks[3] with 275 concepts. For evaluation, we prune the original IAPR TC-12 to form a new dataset, which consists of 18715 images belonging to 22 most frequent concepts, and then generate 33447 image-sentence pairs.

For the propose deep-hashing approach DVSH, we directly use the raw pixels as the image input and word sequences as the sentence input, which consists of one-hot vectors each representing a word of the sentence. As a common practice for fair comparison, for traditional shallow-hashing methods, we use AlexNet [19, 6] to extract deep $fc7$ features for each image in two benchmark dataset by a 4096-dimensional vector, and represent each sentence by a bag-of-word vector.

All image and text features are available at the datasets' website. For Microsoft COCO, we randomly select 25,000 image-sentence pairs as training set, 5000 pairs as validation set and 5000 pairs as query set. For IAPR TC-12 dataset, we randomly select 5000 pairs as the training set, 1000 pairs as the validation set and 100 pairs per class as the test query set. The pairwise similarity labels for training are randomly constructed using semantic labels or concepts, and each pair is considered similar (dissimilar) if they share at least one (none) semantic label, a common protocol used by [23, 21].

We compare the cross-modal retrieval performance of our approach with eight state of the art cross-modal hashing methods, including three unsupervised methods **IMH**[4] [29], **CVH**[5] [20] and **CorrAE**[6] [7], and five supervised methods **CMSSH**[5] [3], **CM-NN**[7] [27], **SCM**[7] [39], **QCH**[7] [35] and **SePH**[8] [23], where **CorrAE** and **CM-NN** are deep methods and the rest are shallow methods. To our best knowledge, there is no cross-modal deep hashing method based either on CNNs or RNNs, hence we extend the state of the art deep network hashing (DNH) method for image retrieval [21] to cross-modal retrieval as a strong baseline, denoted as **DNH-C**. This baseline is modified by applying multi-layer perceptrons to the sentence modality with the same triplet hinge loss as image modality, and adding a least square loss to reduce the gap between the codes of different modalities.

We follow [35, 39, 23, 21] to evaluate the retrieval performance based on three metrics: Mean Average Precision(MAP), *precision-recall* curves, and *precision@top-R* curves. We adopt MAP@$R = 500$ following the baseline methods [35, 23].

We implement the DVSH model in the open-source **Caffe** framework [15]. For training network, we employ the AlexNet architecture [19] and a factored-2-layer LSTM [18], fine-tune convolutional layers $conv1$–$conv5$ and fully-connected layers $fc6$–$fc7$ that were copied from the pre-trained model, and train LSTM layers and feature-map layer $fc8$, all via back-propagation. As the $fc8$ layer is trained from scratch, we

[1] http://mscoco.org
[2] http://imageclef.org/photodata
[3] http://imageclef.org/SIAPRdata
[4] http://staff.itee.uq.edu.au/shenht/UQ_IMH
[5] http://www.cse.ust.hk/~dyyeung/code/mlbe.zip
[6] https://github.com/fangxiangfeng/deepnet
[7] Since code is not publicly available, we implement it by ourselves.
[8] We thank the authors for kindly providing the codes.

Table 1: Mean Average Precision (MAP) Comparison of Cross-modal Retrieval Tasks on Two Datasets

Task	Method	Microsoft COCO [22]				IAPR TC-12 [12]			
		16 bits	32 bits	64 bits	128 bits	16 bits	32 bits	64 bits	128 bits
$I \rightarrow T$	CMSSH [3]	0.4047	0.4886	0.4405	0.4480	0.3445	0.3371	0.3478	0.3738
	CVH [20]	0.3731	0.3677	0.3657	0.3570	0.3788	0.3686	0.3620	0.3540
	IMH [29]	**0.6154**	0.6505	0.6573	0.6770	0.4632	0.4901	0.5104	0.5212
	CorrAE [7]	0.5498	0.5559	0.5695	0.5809	0.4951	0.5252	0.5578	0.5890
	CM-NN [27]	0.5557	0.5602	0.5847	0.5938	0.5159	0.5419	0.5766	0.6003
	SCM [39]	0.5699	0.6002	0.6307	0.6487	**0.5880**	0.6110	0.6282	0.6370
	QCH [35]	0.5723	0.5954	0.6132	0.6345	0.5259	0.5546	0.5785	0.6054
	SePH [23]	0.5813	0.6134	0.6253	0.6339	0.5070	0.5130	0.5151	0.5309
	DNH-C [21]	0.5353	0.5560	0.5693	0.5824	0.4801	0.5093	0.5259	0.5349
	DVSH	0.5870	**0.7132**	**0.7386**	**0.7552**	0.5696	**0.6321**	**0.6964**	**0.7236**
$T \rightarrow I$	CMSSH [3]	0.3747	0.3838	0.3400	0.3601	0.3633	0.3770	0.3645	0.3482
	CVH [20]	0.3734	0.3686	0.3645	0.3711	0.3790	0.3674	0.3636	0.3560
	IMH [29]	0.6068	0.6793	0.7280	0.7403	0.5157	0.5259	0.5337	0.5274
	CorrAE [7]	0.5593	0.5807	0.6109	0.6262	0.4975	0.5195	0.5329	0.5495
	CM-NN [27]	0.5793	0.5984	0.6195	0.6448	0.5119	0.5394	0.5487	0.5649
	SCM [39]	0.5581	0.6188	0.6583	0.6858	0.5876	0.6045	0.6200	0.6262
	QCH [35]	0.5742	0.6057	0.6375	0.6669	0.4997	0.5364	0.5652	0.5885
	SePH [23]	**0.6127**	0.6496	0.6723	0.6929	0.4712	0.4801	0.4812	0.4955
	DNH-C [21]	0.5250	0.5592	0.5902	0.6339	0.4692	0.4838	0.4905	0.5053
	DVSH	0.5906	**0.7365**	**0.7583**	**0.7673**	**0.6037**	**0.6395**	**0.6806**	**0.6751**

set its learning rate to be 10 times that of the lower layers. For hashing networks, we employ AlexNet for image network and a 2-layer LSTM for sentence network, with the feature-map layers ($fc8$ of AlexNet and the output layer of LSTM) trained from scratch. We use the mini-batch stochastic gradient descent (SGD) with 0.9 momentum and the learning rate annealing strategy implemented in Caffe, cross-validate learning rate from 10^{-5} to 1 with a multiplicative step-size 10, and fix mini-batch size as 50. Following [5], we adopt 20 and 25 as the maximum number of words in each sentence for Microsoft COCO and IAPR-TC12 datasets, respectively.

The DVSH approach involves two penalty parameters λ and β for trading off the relative importance of bitwise max-margin loss (7) and squared losses (8) and (9), which can be automatically selected using cross-validation. And we can always achieve good empirical results with $\lambda = 0.1$ and $\beta = 1$. For comparison methods, we use cross-validation to carefully tune their parameters for best results. Each experiment repeats five runs and the average results are reported.

5.2 Results and Discussions

We compare our approach DVSH with the nine state of the art methods on the two datasets in terms of MAP, precision-recall curves and precision@top-R curves of two cross-modal retrieval tasks: image query on sentence database ($I \rightarrow T$), and sentence query on image database ($T \rightarrow I$).

We evaluate all methods with different lengths of hash codes, i.e. 16, 32, 64 and 128 bits, and report their MAP results in Table 1. From the experimental results, we can observe that DVSH substantially outperforms all state of the art methods for most cross-modal tasks on the benchmark datasets which well demonstrates its effectiveness. Specifically, compared to the best shallow baseline SCM with deep AlexNet-$fc7$ features as input, DVSH achieves absolute increases of 8.6%/8.3% and 3.9%/4.0% in average MAP for two cross-modal tasks $I \rightarrow T$ and $T \rightarrow I$ on Microsoft COCO and IAPR TC-12 datasets. SePH does not perform well in comparison to SCM, due to its assumption of t-distribution in the learning procedure, which does not hold on our datasets. Compared to the cross-modal deep hashing methods, DVSH outperform CM-NN by large margins 12.5%/10.3% and 9.7%/10.9%. As we expected, DVSH also outperforms the cross-modal extension of the state of the art deep hashing method DNH-C. But DNH-C cannot outperform the shallow methods with deep features as input (SCM, QCH and SePH), which implies that different architectures and loss functions should be crafted together to achieve optimal performance. This motivates us to craft an end-to-end deep hashing architecture for cross-modal retrieval.

The precision-recall curves with 32 bits for the two cross-modal tasks $I \rightarrow T$ and $T \rightarrow I$ on two datasets Microsoft COCO and IAPR TC-12 are shown in Figure 4, respectively. DVSH shows the best cross-modal retrieval performance at all recall levels. Figure 5 shows the precision@top-R curves of all comparison methods with 32 bits on the two datasets, which shows how the precision changes with the number R of top-retrieved results. Again, we can observe that DVSH significantly outperforms all state of the art methods, which shows that DVSH is also suitable for applications that prefer higher precision by torlerating fewer top-retrieved results.

5.3 Empirical Analysis

To extensively evaluate the effectiveness of the components newly-crafted in this paper, including the cosine max-margin loss for similarity-preserving learning (6), the bitwise max-margin loss for controlling the quality of binary codes (7), and the modality-specific hashing networks for generating out-of-sample hash codes (8)–(9), we design four variants of the DVSH approach: **(a) DVSH-B** is the DVSH variant without binarization ($\text{sgn}(\boldsymbol{h}^l)$ is not performed), which may serve as the upper bound of performance. **(b) DVSH-Q** is the DVSH variant without bitwise max-margin loss (7); **(c) DVSH-I** is the DVSH variant by replacing the cosine max-margin loss (6) with the widely-used inner-product squared

Table 2: Mean Average Precision (MAP) of DVSH Variants for Cross-Modal Retrieval Tasks on Two Datasets

Task	Method	Microsoft COCO [22]				IAPR TC-12 [12]			
		16 bits	32 bits	64 bits	128 bits	16 bits	32 bits	64 bits	128 bits
$I \rightarrow T$	DVSH-B	**0.6658**	**0.7408**	**0.7532**	**0.7645**	**0.6260**	**0.6761**	**0.7359**	**0.7554**
	DVSH	0.5870	0.7132	0.7386	0.7552	0.5696	0.6321	0.6964	0.7236
	DVSH-Q	0.5746	0.7019	0.7145	0.7505	0.5385	0.6113	0.6869	0.7097
	DVSH-I	0.5264	0.5745	0.6056	0.6391	0.4792	0.5035	0.5583	0.5890
	DVSH-H	0.4856	0.5244	0.5545	0.5786	0.4575	0.4975	0.5493	0.5690
$T \rightarrow I$	DVSH-B	**0.7605**	**0.8192**	**0.8034**	**0.8194**	**0.6285**	**0.6728**	**0.6922**	**0.6756**
	DVSH	0.5906	0.7365	0.7583	0.7673	0.6037	0.6395	0.6806	0.6751
	DVSH-Q	0.5530	0.7105	0.7541	0.7569	0.5684	0.6153	0.6618	0.6693
	DVSH-I	0.5185	0.5353	0.5805	0.6136	0.4903	0.5496	0.5890	0.6012
	DVSH-H	0.5025	0.5368	0.5688	0.5939	0.4396	0.4853	0.5185	0.5337

loss $L = \sum_{s_{ij} \in \mathcal{S}} \left(s_{ij} - \frac{1}{K} \langle \boldsymbol{h}_i, \boldsymbol{h}_j \rangle\right)^2$ [24, 36]; **(d) DVSH-H** is the DVSH variant without using the hashing networks (8) and (9), which means that we use the fusion network with single-modal features (image or sentence) to generate hash codes. MAP results of all variants are shown in Table 2.

From Table 2, we may have the following observations:

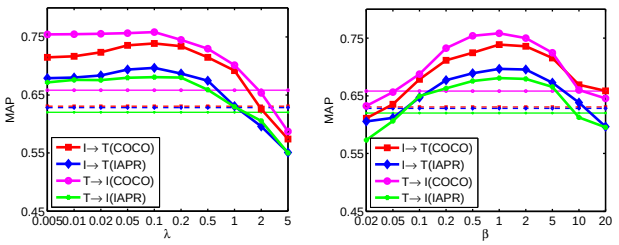
(a) By using cosine max-margin loss, DVSH outperforms DVSH-I by large margins of 11.2%/15.1% and 12.3%/9.2% in average MAP on the two benchmark datasets. The squared inner-product loss has been widely adopted in previous work [24, 36]. However, this loss does not link well the pairwise distances between points (taking values in $(-\infty, +\infty)$ when using continuous relaxation) to the pairwise similarity labels (taking binary values {-1,1}). In contrast, the proposed cosine max-margin loss (6) is inherently consistent with the training pairs. Besides, the margin μ_c in (6) can also control the robustness of similarity-preserving learning to outliers.

(b) By optimizing bitwise max-margin loss (7), DVSH incurs small decreases 3.3%/8.7% and 4.3%/1.8% in average MAP when quantizing continuous embeddings of DVSH-B into binary codes. In contrast, without optimizing bitwise max-margin loss, DVSH-Q incurs larger decreases 4.6%/10.7% and 6.2%/3.9% in average MAP. Especially for shorter length of hash codes (16 bits), DVSH-Q suffers from huge decreases of 9.1%/20.8% and 8.8%/6.0% while DVSH incurs smaller MAP decreases 7.9%/17.0% and 5.6%/2.5%. This validates that the bitwise max-margin loss (7) can effectively reduce the quantization error and achieve higher-quality hash codes.

(c) As we have expected, the performance of DVSH-H drops by huge decreases 16.3%/16.3% and 13.7%/15.5% in average MAP w.r.t. the carefully-crafted DVSH approach. This validates that the visual-semantic fusion network cannot perform well if it is used to generate out-of-sample hash codes which may have only single-modal inputs. This result motivates us to integrate the modality-specific hashing networks into DVSH, our end-to-end deep hashing architecture.

5.4 Parameter Sensitivity

In this section, we further discuss the performance of DVSH w.r.t the two model parameters λ and β to validate the robustness of our approach. Here we compute the MAP score @ 64 bits on both the cross-modal retrieval tasks by varying λ between 0.005 and 5 and β between 0.02 and 20. The sensitivity performance of DVSH with respect to two parameters is illustrated in Figure 6(a) and 6(b). From the figure, we see that DVSH can consistently outperform all the baseline methods by large margins when varying λ between 0.005 and 1, and β between 0.1 and 5. When $\lambda \rightarrow 0$, DVSH deprecates to DVSH-Q which learns hash codes without bitwise max-margin loss (7). We observer the retrieval performance of DVSH first increases and then decreases as λ and β vary and demonstrates a desirable bell-shaped curve. This justifies our motivation of jointly learning deep features whilst minimizing the bitwise max-margin loss (7) and squared losses (8) and (9), since a good trade-off between them can enable effective learning of high-quality hash codes. The results also validate that DVSH is robust against parameter selection.



(a) 64bits w.r.t λ (b) 64bits w.r.t β

Figure 6: The MAP of DVSH @ 64 bits versus the parameter $\lambda \in$ [0.005, 5] and $\beta \in$ [0.02, 20] for the two cross-modal retrieval tasks ($I \rightarrow T$ and $T \rightarrow I$).

6. CONCLUSION

This paper presented a novel deep visual-semantic hashing (DVSH) model to enable efficient cross-modal retrieval of images in response to text sentences and vice versa. Our DVSH model generates compact hash codes of images and sentences in an end-to-end deep learning architecture, which effectively unifies joint multimodal embedding with cross-modal hashing. In particular, by embedding convolutional neural networks over images into recurrent neural networks over sentences, we jointly capture the spatial dependency of images and temporal dynamics of text sentences for learning powerful feature representations and cross-modal embeddings that mitigate the heterogeneity of different modalities. Comprehensive empirical evidence shows that our DVSH model yields state of the art performance in cross-modal retrieval experiments on image-sentences datasets, i.e. standard IAPR TC-12 and large-scale Microsoft COCO. In the future, we plan to extend DVSH to data from social media and mobile computing, and to heterogeneous scenarios where inter-modal relationship information is not available.

7. REFERENCES

[1] G. Andrew, R. Arora, J. Bilmes, and K. Livescu. Deep canonical correlation analysis. In *ICML*, 2013.

[2] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *TPAMI*, 35, 2013.

[3] M. Bronstein, A. Bronstein, F. Michel, and N. Paragios. Data fusion through cross-modality metric learning using similarity-sensitive hashing. In *CVPR*. IEEE, 2010.

[4] Y. Cao, M. Long, J. Wang, H. Zhu, and Q. Wen. Deep quantization network for efficient image retrieval. In *AAAI*, 2016.

[5] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, 2015.

[6] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *ICML*, 2014.

[7] F. Feng, X. Wang, and R. Li. Cross-modal retrieval with correspondence autoencoder. In *MM*. ACM, 2014.

[8] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, T. Mikolov, et al. Devise: A deep visual-semantic embedding model. In *NIPS*, pages 2121–2129, 2013.

[9] H. Gao, J. Mao, J. Zhou, Z. Huang, L. Wang, and W. Xu. Are you talking to a machine? dataset and methods for multilingual image question answering. In *NIPS*, 2015.

[10] A. Graves and N. Jaitly. Towards end-to-end speech recognition with recurrent neural networks. In *ICML*, pages 1764–1772, 2014.

[11] A. Gretton, K. Borgwardt, M. Rasch, B. Schölkopf, and A. Smola. A kernel two-sample test. *JMLR*, 13:723–773, Mar. 2012.

[12] M. Grubinger, P. Clough, H. Müller, and T. Deselaers. The iapr tc-12 benchmark: A new evaluation resource for visual information systems. In *International Workshop OntoImage*, pages 13–23, 2006.

[13] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[14] Y. Hu, Z. Jin, H. Ren, D. Cai, and X. He. Iterative multi-view hashing for cross media indexing. In *MM*. ACM, 2014.

[15] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *MM*. ACM, 2014.

[16] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, pages 3128–3137, 2015.

[17] R. Kiros, R. Salakhutdinov, and R. Zemel. Multimodal neural language models. In T. Jebara and E. P. Xing, editors, *ICML*, pages 595–603. JMLR Workshop and Conference Proceedings, 2014.

[18] R. Kiros, R. Salakhutdinov, and R. S. Zemel. Unifying visual-semantic embeddings with multimodal neural language models. In *NIPS*, 2014.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[20] S. Kumar and R. Udupa. Learning hash functions for cross-view similarity search. In *IJCAI*, 2011.

[21] H. Lai, Y. Pan, Y. Liu, and S. Yan. Simultaneous feature learning and hash coding with deep neural networks. In *CVPR*. IEEE, 2015.

[22] T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312, 2014.

[23] Z. Lin, G. Ding, M. Hu, and J. Wang. Semantics-preserving hashing for cross-view retrieval. In *CVPR*, 2015.

[24] W. Liu, J. Wang, R. Ji, Y.-G. Jiang, and S.-F. Chang. Supervised hashing with kernels. In *CVPR*. IEEE, 2012.

[25] X. Liu, J. He, C. Deng, and B. Lang. Collaborative hashing. In *CVPR*. IEEE, 2014.

[26] M. Long, J. Wang, and P. S. Yu. Compositional correlation quantization for large-scale multimodal search. *arXiv preprint arXiv:1504.04818*, 2015.

[27] J. Masci, M. M. Bronstein, A. M. Bronstein, and J. Schmidhuber. Multimodal similarity-preserving hashing. *TPAMI*, 36, 2014.

[28] A. W. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *TPAMI*, 22, 2000.

[29] J. Song, Y. Yang, Y. Yang, Z. Huang, and H. T. Shen. Inter-media hashing for large-scale retrieval from heterogeneous data sources. In *SIGMOD*. ACM, 2013.

[30] N. Srivastava and R. Salakhutdinov. Multimodal learning with deep boltzmann machines. *JMLR*, 15, 2014.

[31] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *NIPS*, pages 3104–3112, 2014.

[32] J. Wang, H. T. Shen, J. Song, and J. Ji. Hashing for similarity search: A survey. *arXiv preprint arXiv:1408.2927*, 2014.

[33] W. Wang, B. C. Ooi, X. Yang, D. Zhang, and Y. Zhuang. Effective multi-modal retrieval based on stacked auto-encoders. In *VLDB*. ACM, 2014.

[34] Y. Wei, Y. Song, Y. Zhen, B. Liu, and Q. Yang. Scalable heterogeneous translated hashing. In *KDD*, pages 791–800. ACM, 2014.

[35] B. Wu, Q. Yang, W. Zheng, Y. Wang, and J. Wang. Quantized correlation hashing for fast cross-modal search. In *IJCAI*, 2015.

[36] R. Xia, Y. Pan, H. Lai, C. Liu, and S. Yan. Supervised hashing for image retrieval via image representation learning. In *AAAI*. AAAI, 2014.

[37] Z. Yu, F. Wu, Y. Yang, Q. Tian, J. Luo, and Y. Zhuang. Discriminative coupled dictionary hashing for fast cross-media retrieval. In *SIGIR*. ACM, 2014.

[38] W. Zaremba and I. Sutskever. Learning to execute. *CoRR*, abs/1410.4615, 2014.

[39] D. Zhang and W. Li. Large-scale supervised multimodal hashing with semantic correlation maximization. In *AAAI*, 2014.

[40] Y. Zhen and D.-Y. Yeung. Co-regularized hashing for multimodal data. In *NIPS*, 2012.

[41] Y. Zhen and D.-Y. Yeung. A probabilistic model for multimodal hash function learning. In *SIGKDD*. ACM, 2012.