Semantic Transfer-Based Unsupervised Deep Hashing for Web Image Retrieval Postprint

Authors: Xu Sheng, Chen Shengshuang, Xie Liang

Date: 2018-05-24T00:00:00+00:00

Abstract

Current mainstream Web image retrieval methods consider only visual features, fail to fully utilize the textual information accompanying Web images, and neglect the valuable semantics involved in the related text, resulting in weak image representation capability. To address this issue, we propose a novel unsupervised image hashing method: Semantic Transfer Deep Visual Hashing (STDVH). This method first employs spectral clustering to mine semantic information from training texts; then constructs a deep convolutional neural network to transfer textual semantic information into the learning of image hash codes; and finally simultaneously learns image hash codes and hash functions within a unified framework, enabling efficient retrieval of large-scale Web image data in low-dimensional Hamming space. Experiments on two public Web image datasets, Wiki and MIR Flickr, demonstrate the superiority of the proposed method compared to other state-of-the-art hashing algorithms.

Full Text

Preamble

Unsupervised Deep Hashing Based on Semantic Transfer for Web Image Retrieval

Xu Sheng, Chen Shengshuang, Xie Liang
(School of Science, Wuhan University of Technology, Wuhan 430070, China)

Abstract: Most existing Web image retrieval approaches only consider visual features, ignoring the valuable semantics involved in associated texts and failing to take advantage of textual information. This paper proposes a new unsupervised visual hashing approach called Semantic Transfer Deep Visual Hashing (STDVH). Firstly, it extracts semantic information from training texts through spectral clustering. Then, it constructs a deep convolutional neural network to

transfer the text semantic information into the learning of image hash codes. Finally, it trains image hash codes and hash functions within a unified framework, enabling effective retrieval of large-scale Web image data in low-dimensional Hamming space. Experiments on two publicly available image datasets, Wiki and MIR Flickr, demonstrate that the proposed approach achieves superior performance compared to other state-of-the-art hashing algorithms.

Keywords: semantic transfer; image hashing; Web image retrieval; deep learning

0 Introduction

Over the past decade, the availability of Web images has grown tremendously with the continuous advancement of social media and mobile computing technologies. Consequently, intelligent image retrieval technology has attracted increasing attention in the fields of information retrieval and multimedia computing. In particular, Content-Based Image Retrieval (CBIR) [1], as a technique that uses only visual images as queries, has become increasingly important due to its broad application prospects.

To provide high-quality content-based search services over massive image collections, both efficiency and effectiveness are critical research issues that urgently need to be addressed. The simplest CBIR method performs sequential comparison between the query image and each sample stored in the database. Its linear complexity leads to low efficiency and poor scalability in practical applications. Moreover, visual features typically have very high dimensionality. How to address the "curse of dimensionality" remains an open research problem that has not yet been properly solved. However, in most practical CBIR applications, approximate retrieval results can fully satisfy users' information needs, which demonstrates the feasibility of approximate nearest neighbor search. Inspired by this phenomenon, various indexing methods have been developed in recent years, such as inverted documents [2], tree structures [3], and hashing [4]. Inverted documents only perform well for indexing high-dimensional sparse features [2]. The performance of tree structures degrades significantly when the dimensionality of the indexed features becomes high. Furthermore, both inverted documents and tree structures consume substantial memory when storing the corresponding data structures, a problem that becomes more severe when the image collection scale is large.

As one of the emerging technologies supporting fast and accurate image retrieval, image hashing algorithms have received great attention in the past decade and have become a very active research area. The basic idea is to map original high-dimensional visual features into binary codes in low-dimensional Hamming space, enabling visual similarity between images to be measured through simple and efficient bit operations. Generally, image hashing has two main advantages: (a) fast query response, as bitwise operations can be executed efficiently,

allowing the retrieval process to be completed quickly; and (b) low storage consumption, as binary embedding results can significantly reduce the storage of high-dimensional features.

However, due to the "semantic gap" between visual features and human understanding, hashing based on visual features lacks certain semantic information, thereby reducing CBIR performance. To enrich the semantics of image hash codes, many machine learning strategies have been applied and various hashing schemes have been proposed, including unsupervised image hashing [5], supervised image hashing [6], and semi-supervised image hashing [7]. Both supervised and semi-supervised image hashing can improve the semantic discriminability of hash codes. However, these two modes require labeled images during training. In practice, this requirement may not be satisfied in CBIR because high-quality labeled images are scarce in real-world scenarios, and they require substantial manual labor and expert knowledge. On the other hand, images (such as pictures in social networks) are often associated with informative text tags or descriptions. Therefore, it is necessary to consider using auxiliary text to improve the quality of image hashing through unsupervised learning. The core challenge of this approach is how to develop effective unsupervised learning schemes that can intelligently extract and integrate semantics from relevant text information into image hash codes.

This paper proposes a new unsupervised image hashing scheme called Semantic Transfer Deep Visual Hashing (STDVH). The key idea of this method is to automatically extract semantics from image-associated texts and transfer them into image hash codes through deep learning, thereby improving the performance of image hashing. STDVH works as follows: First, semantic information is obtained through spectral clustering on text information and transferred to the subsequent learning of visual hash codes. Then, a deep convolutional neural network model is constructed to obtain hash codes. Finally, semantic transfer, hash code learning, and hash function learning are integrated into a unified framework, enabling the learned deep hash functions to simultaneously preserve the semantic information corresponding to original images and their visual similarity. Hash codes for database and query images can be obtained through the hash function for image retrieval.

The main contributions of this paper are summarized as follows: 1. STDVH does not only consider image features or simultaneously process images and text, but specifically utilizes semantic transfer learning from text information to assist image hashing. Through spectral clustering, text semantics are obtained as clusters, and a model is built based on the text clustering results and images, effectively integrating semantics into hash codes. 2. STDVH adopts a unified unsupervised learning framework that integrates hash function learning, hash code learning, and semantic transfer learning into a single framework, using deep convolutional neural networks to learn the semantic information from original images to text. 3. STDVH can be trained using multimodal data from Web images including both visual and textual features, but requires only visual

information from images as input for queries. This conforms to the practical requirements of CBIR, where Web images in the database are usually accompanied by text information, while users do not need to provide text queries.

Comprehensive experiments on publicly available image databases fully demonstrate the superiority of STDVH and prove that it significantly outperforms several state-of-the-art content-based or cross-modal hashing methods in various aspects.

1 Related Work

1.1 Single-Modal Image Hashing

According to how hash functions are generated, existing single-modal image hashing (SFVH) can be further divided into two categories: data-independent hashing [8] and data-dependent hashing [5]. Locality-Sensitive Hashing (LSH) [8] is one of the most typical data-independent hashing methods. It maps similar points to the same Hamming space with high probability based on random vectors from specific distributions such as standard Gaussian distribution. On the other hand, data-dependent hashing learns hash functions based on the characteristics of the underlying data distribution through machine learning methods. Spectral Hashing (SH) [5] is a typical unsupervised learning-based hashing method that learns hash functions by preserving the similarity of images in hash codes. With the development of hashing technology, Sparse Embedding Hashing [9], manifold-based hashing [10], and deep learning-based hashing [7] have been proposed to learn effective binary hash codes.

1.2 Multi-Modal Image Hashing

In multi-modal image hashing, one image modality can be replaced by text modality. Integrating multiple modalities comprehensively is crucial for fully interpreting image content and achieving optimal learning effectiveness [11]. Many researchers have designed various schemes considering multi-feature fusion for hashing for different purposes. For example, Multi-View Latent Hashing (MVLH) [11] combines multi-modal features into binary representation learning by discovering shared latent factors among multiple views, learning the weights of multi-feature fusion according to the reconstruction error of each view. Multi-View Alignment Hashing (MVAH) [12] learns hash codes through regularized kernel non-negative matrix factorization, considering the implicit semantics and joint probability distribution of multiple visual features.

The most significant limitation of both single-modal and multi-modal image hashing is that they only consider features from visual modalities. Due to the semantic gap, image relationships characterized by low-level visual features cannot effectively describe rich image semantics, resulting in hash codes with less semantic meaning.

1.3 Cross-Modal Hashing

The core idea of cross-modal hashing is to map heterogeneous modal features into a common Hamming space and calculate similarity in that space to return cross-modal retrieval results. Zhou et al. [13] proposed Latent Semantic Sparse Hashing (LSSH) by employing sparse coding and matrix factorization. Collective Matrix Factorization Hashing (CMFH) [14] learns hash codes from multiple modalities of a sample using collective matrix factorization with a latent factor model.

Since the projection space of cross-modal hashing may embed more semantics than single-modal visual feature space, cross-modal hashing can improve CBIR performance. However, the main design goal of various cross-modal hashing methods is multimedia retrieval across different modalities. They assume that each involved modality contributes equally to cross-modal retrieval. This assumption makes them share the same Hamming space, so they cannot effectively discriminate specialized image visual information. Additionally, the unique discriminative information in original image visual features may be lost during hashing due to forced heterogeneous modality association.

summarizes the key features of state-of-the-art hashing methods and the proposed STDVH. Based on the above analysis, it can be found that it is very important to specifically design an intelligent hashing method that effectively utilizes relevant modalities (such as text information) to assist image hashing.

2 Unsupervised Deep Hashing Based on Semantic Transfer

2.1 System Overview

[Figure 1: see original paper] illustrates the system framework of unsupervised deep hashing based on semantic transfer for image retrieval. The system mainly consists of two parts: offline learning and online retrieval.

a) Offline Learning. The purpose of this part is to learn hash codes from database images and generate hash functions for query images. It contains three main steps: First, spectral clustering is performed on text features in the training set to enhance the learning effect of image hash codes using the spectral clustering results of text information. Then, a convolutional neural network is constructed with original images from the training set as input and spectral clustering results of text information as output. The entire network is trained, and the output of the third-to-last layer of the convolutional neural network is transformed into hash codes. Finally, semantic transfer, hash code learning, and hash function learning are integrated into a unified framework, enabling the learned deep hash functions to simultaneously preserve the semantic information corresponding to original images and their visual similarity.

b) Online Retrieval. Query images are extracted and mapped into binary codes using the hash function. Finally, the Hamming distance between the query

image and database images is calculated, and database images are returned in ascending order of distance.

2.2 Notation and Problem Description

This paper uses bold uppercase letters to denote matrices and bold lowercase letters to denote vectors. The transpose of matrix X is denoted as $X^\top$, the inverse of matrix X is denoted as X^{-1}, $\mathrm{tr}(X)$ denotes the trace of matrix X, and $\|X\|_F$ denotes the Frobenius norm. $\mathrm{sgn}(\cdot)$ denotes the sign function, returning 1 if positive and -1 otherwise.

Let $\{(x_i^{(1)}, x_i^{(2)})\}_{i=1}^N$ denote N pairs of data points, where $x_i^{(1)}$ represents the i-th image and $x_i^{(2)}$ represents the corresponding text features. The goal of unsupervised image hashing is to learn hash codes $Y = \{y_i\}_{i=1}^N \in \{-1, 1\}^{L\times N}$ for database images $X^{(1)} = \{x_i^{(1)}\}_{i=1}^N$, where $y_i \in \{-1, 1\}^L$ is the hash code of the i-th image and L is the length of the image hash code. Additionally, we need to learn a hash function $h(x^{(1)})$ such that $y_i = h(x_i^{(1)})$.

2.3 Semantic Learning

This paper transfers semantic information from text features to the learning of visual hash codes to enhance the effectiveness of visual hashing. In the STDVH model, considering that spectral clustering can cluster on sample spaces of arbitrary shapes and converges to the global optimum, this paper uses spectral clustering to generate semantic information from text features. Specifically, for text features $X^{(2)} = \{x_i^{(2)}\}_{i=1}^N$ in the training set, spectral clustering is used to generate a category matrix $C = \{c_i\}_{i=1}^N$, where c_i represents the category corresponding to the text feature of the i-th data point, i.e., the semantic information. Then, based on the category matrix C, a text similarity matrix $S = \{s_{ij}\}$ is generated and transferred to visual hash learning.

The text features $X^{(2)} = \{x_i^{(2)}\}_{i=1}^N$ are divided into K classes. A graph $G = (X^{(2)}, E)$ is constructed based on the data, where vertices represent text information and weighted edges represent similarity between texts. Let $A_1, A_2, \ldots, A_K$ be several subsets of the graph with no intersections. To minimize the Cut of the partition, spectral clustering aims to minimize the following objective function:

$$\mathrm{cut}(A_1, \ldots, A_K) = \frac{1}{2}\sum_{i=1}^{K} \mathrm{cut}(A_i, \bar{A}_i) = \sum_{i=1}^{K} \sum_{j\in A_i, k\in \bar{A}_i} w_{jk}$$

where $\bar{A}_i$ represents the complement of A_i, and $\mathrm{cut}(A_i, \bar{A}_i)$ represents the sum of weights of all edges between A_i and $\bar{A}_i$.

The graph is represented using an adjacency matrix $M = \{m_{ij}\}_{i,j=1}^N$, where similarity m_{ij} is calculated according to equation (2):

$$m_{ij} = e^{-\frac{2(1-\mathrm{COS}(x_i^{(2)}, x_j^{(2)}))}{\sigma^2}}$$

where σ is a hyperparameter, and $\mathrm{COS}(\cdot,\cdot)$ represents cosine distance. To exclude the influence of self-similarity, diagonal elements are assigned a value of 0, i.e., $m_{ii} = 0$.

To improve the effectiveness of spectral clustering, the similarity matrix M is sparsified according to equation (3):

$$m_{ij} = \begin{cases} m_{ij}, & m_{ij} > \lambda \cdot \bar{m} \\ 0, & m_{ij} \leq \lambda \cdot \bar{m} \end{cases}$$

where λ is the sparsity degree and $\bar{m}$ represents the average of all elements.

To prevent any single node from being more easily removed, this paper considers a normalized diagonal matrix D, where diagonal elements are the sum of all elements in a row (or column, as they are the same due to symmetry) of the similarity matrix, i.e., $d_{ii} = \sum_{j=1}^{N} m_{ij}$.

Then, the normalized Laplacian graph matrix is calculated:

$$L = I - D^{-1/2} M D^{-1/2}$$

The eigenvalues and eigenvectors of the Laplacian graph matrix L are computed. The eigenvectors corresponding to the top K largest eigenvalues are selected and combined column-wise into a matrix $V = [v_1, v_2, \ldots, v_K]$, where $v_1, v_2, \ldots, v_K$ are the eigenvectors corresponding to the K largest eigenvalues.

Each row of matrix V is treated as a point in K-dimensional space. Using a traditional clustering algorithm, K-means [15] is employed here to cluster them into K classes. The category to which each row in the clustering result belongs represents the category of the original text features, yielding the category matrix C. The semantic learning process is summarized in Algorithm 1.

Algorithm 1: Semantic Learning
Input: Training set text features $X^{(2)} = \{x_i^{(2)}\}_{i=1}^{N}$, number of clusters K, hyperparameter σ, sparsity degree λ
Output: Semantic category matrix $C = \{c_i\}_{i=1}^{N}$

1. Compute similarity matrix $M = \{m_{ij}\}_{i,j=1}^{N}$
2. Set diagonal values of M to 0, $m_{ii} = 0$
3. Sparsify M according to formula (3)

4. Compute normalized matrix D

5. Compute normalized Laplacian graph matrix $L = I - D^{-1/2}MD^{-1/2}$

6. Compute eigenvectors of L, combine the top K eigenvectors corresponding to the largest eigenvalues column-wise into a matrix $V = [v_1, v_2, \dots, v_K]$

7. Normalize V to form matrix $P = D^{-1/2}V$

8. Treat each row of matrix P as a data point, perform K-means clustering to obtain C

2.4 Unsupervised Deep Hashing

The STDVH model in this paper includes a CNN model. The unsupervised deep hashing is a 9-layer convolutional neural network, where the first 7 layers are the same as AlexNet [16]. Of course, other CNN structures can also replace AlexNet in STDVH, but the purpose of this paper is not to study different networks. Therefore, only AlexNet is used as part of the deep hashing in the STDVH model, with other networks to be studied in the future. Based on the previous semantic learning results, original images are used as input to the CNN and the corresponding semantic categories are used as output.

shows the detailed configuration of the deep hashing part of STDVH. This part includes 5 convolutional layers (conv 1-5) and 4 fully connected layers (full 6-9). Each convolutional layer is described as follows: "filter" indicates the number of convolutional filters and the size of their receptive field and number of channels, in the form "number size×size×channels" ; "stride" indicates the convolution stride, i.e., the interval at which the filter is applied to the input; "pad" indicates the number of pixels to be added to each side of the input; "LRN" indicates whether to apply a local response normalization layer; "pool" indicates the downsampling factor; "4096" in the fully connected layer indicates the output dimension; "L" indicates the length of the hash code; "K" indicates the number of categories generated from text semantics. The activation function for all layers is the Rectification Linear Unit (RELU).

2.5 Hashing Learning Based on Unified Framework

This paper builds a unified unsupervised framework that integrates semantic transfer, hash code learning, and hash function learning.

Based on the semantic learning result $C = \{c_i\}_{i=1}^{N}$, the similarity matrix $S = \{s_{ij}\}$ of data points can be obtained. $s_{ij} = 1$ when x_i and x_j are similar, and $s_{ij} = 0$ when x_i and x_j are not similar. It is defined as follows:

$$s_{ij} = \begin{cases} 1, & c_i = c_j \\ 0, & \text{otherwise} \end{cases}$$

For given binary codes $Y = \{y_i\}_{i=1}^N$ of all images, the maximum likelihood estimation for S is defined as:

$$p(S|Y) = \prod_{s_{ij}\in\Omega} p(s_{ij}|Y), \quad p(s_{ij}|Y) = \begin{cases} \sigma(\Omega_{ij}), & s_{ij} = 1 \\ 1 - \sigma(\Omega_{ij}), & s_{ij} = 0 \end{cases}$$

where $\Omega_{ij} = \frac{1}{2} y_i^\top y_j$ and $\sigma(\Omega_{ij}) = \frac{1}{1+e^{-\Omega_{ij}}}$. Note that $y_i \in \{-1, 1\}^L$.

By taking the negative log-likelihood, the following optimization problem is obtained:

$$\min_Y -\log p(S|Y) = \sum_{s_{ij}\in\Omega} \log(1 + e^{\Omega_{ij}}) - \sum_{s_{ij}\in\Omega} s_{ij}\Omega_{ij}$$

Obviously, the above optimization problem (9) can make the Hamming distance between two similar points as small as possible, while making the Hamming distance between two dissimilar points as large as possible, which aligns with the goal of unsupervised image hashing.

This paper solves this problem in a discrete manner by converting it into the following equivalent form:

$$\min_{Y\in\{-1,1\}^{L\times N}} \sum_{s_{ij}\in\Omega} \log(1 + e^{U_{ij}}) - \sum_{s_{ij}\in\Omega} s_{ij}U_{ij} \quad \text{s.t.} \quad Y = \operatorname{sgn}(U), \quad U \in \mathbb{R}^{L\times N}$$

where $U_{ij} = \frac{1}{2} u_i^\top u_j$. $U = \{u_i\}_{i=1}^N$ is the continuous state expression of binary hash code Y under relaxed conditions, and the discrete Y can be obtained using $y_i = \operatorname{sgn}(u_i)$.

To optimize problem (10), the equality constraints in (10) can be moved to a regularization term to optimize the regularized problem:

$$\min_{Y\in\{-1,1\}^{L\times N}, U} \sum_{s_{ij}\in\Omega} \log(1 + e^{U_{ij}}) - \sum_{s_{ij}\in\Omega} s_{ij}U_{ij} + \frac{\eta}{2} \sum_{i=1}^N \|y_i - u_i\|^2$$

where η is the regularization term.

Traditional hashing methods usually rely on handcrafted feature extraction, and the feature extraction stage is separate from the hash learning stage, causing the extracted features to not be optimally adapted to the hashing process. These features often cannot maintain semantic similarity. To enable mutual promotion between image feature representation and hash codes, deep hashing is aligned with the objective function by letting:

$$u_i = W^\top \phi(x_i^{(1)}; \theta) + v$$

where θ represents all parameters of the first 7 layers of the CNN network in the deep hashing part; $\phi(x_i^{(1)}; \theta)$ represents the output of the 7th layer for image $x_i^{(1)}$; and v is a bias vector. This means that in the deep hashing part, a fully connected layer with weight matrix $W \in \mathbb{R}^{4096 \times L}$ and bias vector $v \in \mathbb{R}^L$ connects the image feature part and the semantic transfer result. The problem is then formulated as:

$$\min_{Y \in \{-1,1\}^{L \times N}, W, v, \theta} \sum_{s_{ij} \in \Omega} \log(1 + e^{U_{ij}}) - \sum_{s_{ij} \in \Omega} s_{ij} U_{ij} + \frac{\eta}{2} \sum_{i=1}^{N} \|y_i - u_i\|^2$$

where $u_i = W^\top \phi(x_i^{(1)}; \theta) + v$.

In this way, semantic transfer, hash function learning, and hash code learning are integrated into the same framework. In the STDVH model, the parameters of the learning part include W, v, and θ. A mini-batch learning strategy is adopted, where in each iteration, a small portion of data points is sampled from the entire training set, and learning is performed based on these sampled points. An alternating learning optimization method is used, optimizing one parameter while fixing the others.

For y_i, it is directly optimized as follows:

$$y_i = \mathrm{sgn}(u_i) = \mathrm{sgn}(W^\top \phi(x_i^{(1)}; \theta) + v)$$

For parameters W, v, and θ, the backpropagation algorithm is used for learning. The derivative of the loss function with respect to u_i is calculated according to equation (15). Then, backpropagation is used to update parameters W, v, and θ respectively.

Algorithm 2: Deep Image Hashing Learning
Input: Training set images $X^{(1)} = \{x_i^{(1)}\}_{i=1}^N$, semantic similarity $S = \{s_{ij}\}$
Output: Hash codes $Y = \{y_i\}_{i=1}^N$

Initialization: Initialize parameters θ of the CNN network in the initial deep hashing model, and initialize each item of W and v by random sampling from a Gaussian distribution with mean 0 and variance 0.01.

Randomly sample mini-batch points from $X^{(1)}$, and for each sampled point $x_i^{(1)}$ perform the following operations: 1. Compute u_i through forward propagation, and compute the binary code $y_i = \mathrm{sgn}(u_i)$ 2. Compute the derivative of u_i according to formulas (16), (17), and (18) 3. Update parameters W, v, and θ using backpropagation

End for a fixed number of iterations.

After completing the learning process, only hash codes for points in the training data can be obtained. Out-of-sample extension is still needed to predict hash codes for points not present in the training set. The deep hashing framework of STDVH can be naturally extended to out-of-sample cases. For any image $x_q \notin X^{(1)}$, its hash code is predicted using the following hash function:

$$h(x_q) = \text{sgn}(W^\top \phi(x_q; \theta) + v)$$

The hash function learning part is summarized in Algorithm 2.

3 Experiments

3.1 Experimental Datasets

To verify the effectiveness of unsupervised deep hashing based on semantic transfer, comprehensive experiments were conducted on two publicly available image datasets: Wiki [17] and MIR Flickr [18]. All datasets consist of image-text pairs and have been widely used to evaluate multimedia retrieval performance in past work. Under the same settings, all datasets are divided into query sets, learning sets, and database sets. This experimental setup conforms to the practical application scenarios of CBIR.

Wiki contains 2,866 multimedia documents from 10 semantic categories collected from Wikipedia. Visual content is represented by raw images, and text content is represented by 10-dimensional topic vectors generated through Latent Dirichlet Allocation. For the Wiki dataset, since images are already labeled with 10 different categories, images in this dataset are considered relevant only when they belong to the same category.

MIR Flickr contains 25,000 images from 38 categories obtained from Flickr. Each image has text tags. To exclude the influence of tags irrelevant to image content, text tags that appear fewer than 50 times are removed, resulting in a vocabulary of 457 tags [19]. Visual content is represented by raw images, and text content is represented by 457-dimensional binary vectors, where each dimension indicates whether the corresponding tag is attached to the image. Since images in MIR Flickr typically belong to multiple categories, they are considered relevant only when they share at least one common category.

shows the statistics of the experimental data.

3.2 Evaluation Metrics

In this experimental study, mean Average Precision (mAP) is adopted as the evaluation metric [14]. For a given query q, Average Precision (AP) is calculated according to the formula:

$$\mathrm{AP}(q) = \frac{1}{R}\sum_{r=1}^{R} \mathrm{precision}(r) \cdot \mathrm{rel}(r)$$

where R is the number of returned results; precision(r) represents the precision of the top r retrieved images, defined as the ratio of relevant images to the number of retrieved images; and rel(r) is an indicator function that equals 1 if the r-th image is relevant to the query and 0 otherwise. mAP is defined as the average of AP values for all queries, with larger values indicating better retrieval performance. In the experiments, R is set to 50 to obtain results. Additionally, Precision-Scope curves are provided to reflect changes in retrieval performance relative to the number of retrieved images.

3.3 Comparison Methods

The proposed method is specifically designed for CBIR without using any supervised information from images. Therefore, for fair comparison, this paper compares with some state-of-the-art single-modal and cross-modal (multi-modal) hashing methods. The single-modal hashing methods used for comparison include Iterative Quantization (ITQ) [20], Locality-Sensitive Hashing (LSH) [8], PCA Hashing (PCAH) [21], Spectral Hashing (SH) [5], Random Rotation PCA Hashing (PCA-RR) [20], and Density Sensitive Hashing (DSH) [22]. The unsupervised cross-modal hashing methods used for comparison include Canonical Correlation Analysis Hashing (CCA) [20] and Collective Matrix Factorization Hashing (CMFH) [14]. CMFH learns a latent semantic subspace shared across multiple modalities, mapping both visual and text features into a unified hash code.

Note that CCA and CMFH can generate hash codes for both query visual images and text. The purpose of the experiment is to test their performance for CBIR, so this paper removes the text hash codes. In this case, the CBIR retrieval process for all comparison methods is based on the Hamming distance of visual hash codes. Parameters for all comparison methods are adjusted according to relevant literature and reports for optimal performance.

3.4 Implementation Details

In the experiments, parameters are selected through multiple comparative experiments. For the Wiki dataset, the number of spectral clusters $K = 15$, the hyperparameter in the similarity matrix $\sigma = 1$, and the sparsity degree of the similarity matrix $\lambda = 0.1$ are selected, under which conditions the performance is optimal. For the MIR Flickr dataset, since the text data is binary vectors and the similarity matrix is sufficiently sparse, no sparsification is needed during semantic extraction, so the sparsity degree $\lambda = 0$ is selected. In the experiments, the hash code length L ranges in $[16, 32, 64, 128]$ for observing performance on all datasets. The retrieval range is set from 100 to 1000 with a step size of 100.

In the deep hashing part, all images are first resized to 227×227 pixels. Then, raw images are directly used as input, and semantic information generated by spectral clustering of text features is used as output. The AlexNet network pretrained on the ImageNet [16] dataset is used to initialize the first 7 layers of the STDVH framework.

4 Results and Discussion

The mAP results of STDVH and all comparison methods on different datasets with different hash code lengths are shown in . The Precision-Scope curves for 128-bit hash code length on the two datasets are shown in [Figure 2: see original paper]. Based on the obtained results, it can be clearly seen that STDVH outperforms all comparison methods. As the hash code length increases, the retrieval performance of STDVH steadily improves. However, for some other compared methods, the improvement in retrieval performance with increasing length is not obvious, indicating that the hash codes learned by STDVH have less information redundancy. Moreover, STDVH can achieve better performance using fewer hash bits than other methods using longer bits. The reason is that with the help of text semantic transfer, STDVH can compress more semantic information into short hash codes. This means that CBIR based on STDVH can have faster retrieval processes and lower storage costs at the same performance level.

4.1 Effect of Semantic Transfer

This paper verifies through experiments that text semantic transfer effectively improves the semantic effectiveness of visual hashing. Deep visual hashing models are built using spectral clustering results from text features and image features respectively, and the performance of STDVH is compared with that of a version that ignores text information and only considers visual features. [Figure 3: see original paper] shows detailed experimental results. It can be observed that semantic transfer can improve the retrieval performance of CBIR. The reason for better performance is that with the help of semantics, the relationship between images and potential common themes can be better modeled and associated. The extracted valuable semantics can be effectively encoded in binary hash codes. The performance gap varies across different datasets and hash code lengths, mainly due to different auxiliary effects of text on visual hashing.

4.2 Effect of Training Set Size

This section constructs experiments to observe performance changes with training set size on MIR Flickr. The hash code length is set to 128 bits, and performance changes are recorded as the training set size varies from 1,000 to 10,000. shows the main results. When more training data is utilized, the mAP of STDVH increases slightly, demonstrating the stability of STDVH in learning hash functions. Further observation verifies that with limited training data, text

semantic transfer can effectively alleviate the semantic deficiency of visual hash codes.

4.3 Parameter Sensitivity

This section observes the effect of parameters in STDVH on performance changes through experiments. K represents the number of clustering categories in text information spectral clustering, σ is the hyperparameter, and λ is the sparsity degree of the similarity matrix. The hash code length is set to 128 bits, and experiments are conducted on the Wiki dataset. Parameter K is varied from 2 to 12, parameter σ is varied across orders of magnitude (0.01, 0.1, 1, 10, 100), and parameter λ is varied from 0 to 5. In the experiments, one parameter is fixed while the remaining two parameters are varied. Detailed experimental results are shown in [Figure 4: see original paper]. From Figure 4: see original paper, it can be seen that performance is relatively better when the number of clusters $K = 10$; from Figure 4: see original paper, it can be seen that performance is optimal when σ reaches a certain point, i.e., $\sigma = 0.1$ or $\sigma = 1$, and $K = \lambda = 10$.

5 Conclusion

Most existing hashing methods for CBIR only consider visual features. They ignore the valuable semantics involved in associated texts. This study proposes an effective hashing framework, STDVH, which utilizes associated texts of images to extract semantics and transfer them to unsupervised visual hash learning. A deep convolutional neural network is constructed to integrate additional discriminative semantic information into visual hash codes and functions while preserving image visual similarity. Offline learning can effectively utilize semantics involved in texts, while online hashing requires only visual images as input, conforming to the practical application scenarios of CBIR. Comprehensive experiments on several standard image datasets verify that visual hashing performance can be improved with the assistance of texts, and STDVH achieves better performance compared to some existing hashing techniques.

References

[1] Su Jahwung, Huang Weijyun, Yu Philip S, et al. Efficient relevance feedback for content-based image retrieval by mining user navigation patterns [J]. IEEE Trans on Knowledge & Data Engineering, 2011, 23 (3): 360-372.

[2] Sivic J, Zisserman A. Video Google: a text retrieval approach to object matching in videos [C]// Proc of IEEE International Conference on Computer Vision. [S. l.] : IEEE Computer Society, 2003: 1470.

[3] Ciaccia P, Patella M, Zezula P. M-tree: an efficient access method for similarity search in metric spaces [C]// Proc of International Conference on Very Large Data Bases. [S. l.] : Morgan Kaufmann Publishers Inc, 1997.

[4] Wang Jingdong, Shen Hengtao, Song Jingkuan, et al. Hashing for similarity search: a survey [EB//OL]. (2014) . arXiv preprint arXiv: 1408. 2927.

[5] Weiss Y, Torralba A, Fergus R. Spectral hashing [C]// Proc of International Conference on Neural Information Processing Systems. [S. l.] : Curran Associates Inc, 2008: 1753-1760.

[6] Liu Wei, Wang Jun, Ji Rongrong, et al. Supervised hashing with kernels [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. [S. l.] : IEEE Computer Society, 2012: 2074-2081.

[7] Gao Lianli, Song Jingkuan, Zou Fuhao, et al. Scalable multimedia retrieval by deep learning hashing with relative similarity learning [J]. Environmental Modelling & Software, 2015, 39 (39): 903-906.

[8] Raginsky M, Lazebnik S. Locality-sensitive binary codes from shift-invariant kernels [C]// Advances in Neural Information Processing Systems. 2009: 1509-1517.

[9] Zhu Xiaofeng, Zhang Lei, Huang Zi. A sparse embedding and least variance encoding approach to hashing [J]. IEEE Trans on Image Processing A Publication of the IEEE Signal Processing Society, 2014, 23 (9): 3737-50.

[10] Shen Fumin, Shen Chunhua, Shi Qinfeng, et al. Hashing on nonlinear manifolds [J]. IEEE Trans on Image Processing A Publication of the IEEE Signal Processing Society, 2015, 24 (6): 1839-51.

[11] Shen Xiaobo, Shen Fumin, Sun Quansen, et al. Multi-view latent hashing for efficient multimedia Search [C]// Proc of ACM Multimedia. 2015: 831-840.

[12] Liu Li, Yu Mengyang, Shao Ling. Multiview alignment hashing for efficient image search [J]. IEEE Trans on Image Processing A Publication of the IEEE Signal Processing Society, 2015, 24 (3): 956-966.

[13] Zhou Jie, Ding Guiguang, Guo Yuchen. Latent semantic sparse hashing for cross-modal similarity search [C]// Proc of International ACM SIGIR Conference on Research & Development in Information Retrieval. 2014: 415-424.

[14] Ding Guiguang, Guo Yuchen, Zhou Jile. Collective matrix factorization hashing for multimodal data [C]// Proc of Computer Vision and Pattern Recognition. [S. l.] : IEEE Computer Society, 2014: 2083-2090.

[15] Hartigan J A. A K-means clustering algorithm [J]. Appl Stat, 1979, 28 (1): 100-108.

[16] Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks [C]// Proc of International Conference on Neural Information Processing Systems. [S. l.] : Curran Associates Inc, 2012: 1097-1105.

[17] Rasiwasia N, Pereira J C, Coviello E, et al. A new approach to cross-modal multimedia retrieval [C]// Proc of International Conference on Multi-

media. 2010: 251-260.

[18] Huiskes M J, Lew M S. The MIR flickr retrieval evaluation [C]// Proc of ACM International Conference on Multimedia Information Retrieval. 2008: 39-43.

[19] Guillaumin M, Verbeek J, Schmid C. Multimodal semi-supervised learning for image classification [C]// Proc of IEEE Computer Vision and Pattern Recognition. 2010: 902-909.

[20] Gong Yunchao, Lazebnik S, Gordo A, et al. Iterative quantization: a procrustean approach to learning binary codes for large-scale image retrieval [J]. IEEE Trans on Pattern Analysis & Machine Intelligence, 2013, 35 (12): 2916-2929.

[21] Yu Xiang, Zhang Shaoting, Liu Bo, et al. Large scale medical image search via unsupervised PCA hashing [C]// Proc of IEEE Computer Vision and Pattern Recognition Workshops. 2013: 393-398.

[22] Jin Zhongming, Li Cheng, Lin Yue, et al. Density sensitive hashing [J]. IEEE Trans on Cybernetics, 2014, 44 (8): 1362-1371.

Note: Figure translations are in progress. See original paper for figures.