Deep Hashing with Attribute Guidance for Image Retrieval

Yao Lu\textsuperscript{a} and Heyan Huang\textsuperscript{b}

Beijing Engineering Research Center of High Volume Language Information Processing and Cloud Computing Applications, Department of Computer Science and Technology, Beijing Institute of Technology

\textsuperscript{a}luyao@bit.edu.cn, \textsuperscript{b}hhy63@bit.edu.cn

Abstract. Similarity-preserving hashing is an important method to solve approximate nearest neighbour search problem for image retrieval. Lots of works have been proposed on supervised hashing and image category labels are utilized as supervised information. However, current works ignore a fact that images can be described by a set of attributes. Intuitively, images of different categories with close visual features can be separated by their attribute features because attributes keep more detailed information. In this paper, we propose a novel supervised deep hashing method with image attribute guidance. Specifically, hash codes are learnt through image visual features and guided by image attributes by maintaining pair wise similarities between images as well as the corresponding attribute descriptions. Extensive experimental results on two benchmark datasets show that our proposed method achieves better performance compared with the state of the art hashing methods.

1. Introduction

Similarity-preserving hashing is an important method to solve approximate nearest neighbour search problem. The key idea of similarity preserving hashing is to learn a hashing function to project data points from the original feature space into binary codes in the Hamming space, aiming to minimize Hamming distance of similar data points and maximize Hamming distance of dissimilar data points [1]. Lots of works have been proposed on hashing for image retrieval. Among various hashing methods, supervised hashing has achieved remarkable progress [2-10]. By taking advantage of image category, supervised hashing methods keep pair wise similarity between images, i.e., two images with same category labels are similar while two images with different category labels are dissimilar [10].

However, images can be described by a set of attributes. Take biological images for example, attributes are used to describe precise biological features thus keep more detailed information. Images of different categories with close visual features can be separated via attribute descriptions, since images of different categories have more dissimilar biological information. Hash codes should not only preserve information from image visual features but also from the corresponding attribute descriptions of images. Current works [2-10] only take image categories into consideration and fail to focus on image attribute descriptions, thus catch less information. For example, as is shown in Figure 1, the hash code of image of Wilson Warbler have a distant Hamming distance with the hash code of image of Black Bird, which implies that two images are dissimilar. If attribute descriptions are not kept by hash codes, the Hamming distance between the hash code of image of Yellow Warbler and the hash code of image of Wilson Warbler is small, indicating two images are similar and should be classified into the same category, which is wrong. If attribute descriptions are maintained, the
Hamming distance between the hash code of image of Yellow Warbler and the hash code of image of Wilson Warbler is larger, which implies that two images are dissimilar and are from different categories.

Figure 1. Comparison between hash code with no attribute and hash code with attributes.

In this paper, we propose a novel deep hashing method with image attribute guidance. Specifically, hash codes are learnt through image visual features and guided by the corresponding attribute descriptions. We first learn refined attribute features for the corresponding attribute descriptions of images, which is supervised by image category labels. Then hash codes of images are generated through visual features and guided by the refined attribute features. To take advantage of image attributes, pairwise similarities are maintained among images, as well as image attributes, by an objective function based on Weighted Maximum Likelihood estimation [11]. In this way, our proposed method can generate hash code that keeps information from both image visual feature and image attribute description.

Our major contributions are summarized as follows:

- We propose a novel deep hashing method with image attribute guidance. To maintain similarity among images as well as image attributes, our proposed method can learn a hash function that keeps information from both image visual features and attribute descriptions. For images of different categories with close visual features, the learnt hash function can generate hash codes with distant Hamming distance.

- Extensive experimental results on two datasets, including Caltech-UCSD Birds 200 [12] and Animal with attribute 2 [13], show that our proposed method can achieve better performance compared with the state of the art hashing methods.

2. Related Work

Existing hashing methods for image retrieval can be categorized into data independent hashing and data dependent hashing. Data independent hashing methods, like Locality Sensitive Hashing (LSH) [14] use no data for training and achieve less satisfactory performance while data dependent hashing methods utilize image data to generate hashing codes thus achieve better results. Generally, data dependent hashing methods consist of two kinds: unsupervised hashing methods and supervised hashing methods. Unsupervised hashing methods use no data labels for training and maintain image similarity in the Euclidean space, like Iterative Quantization Hashing (ITQ) [15] and PCA Hashing (PCAH) [16]. Supervised hashing methods use labeled data for training and take label information to supervise hash code learning. FSSH [17] adopts image labels together with images visual features to generate hash codes. Recent researches try to combine deep learning with hashing for image retrieval. Convolutional neural networks are applied to extract image features for hash code learning, which is supervised by image category labels [18-22]. CNNH [18] proposes the first work to combine deep
learning with hashing. DSH [19] proposes to optimize real value code to get discrete hashing code. DPSH [20] proposes to maintain pair wise similarity among images for hash code learning. DCMH [21] focuses on cross modal hashing for images and texts. HashNet [22] proposes a new activation function to learn discrete hash codes.

Existing works mainly focus on utilize image category labels as supervised information for hash code learning. However, they ignore a fact that images can be described by attributes which maintain more detailed information. In this paper, we propose a novel deep hashing method with image attribute guidance to keep information from both images visual features and images attributes.

3. Deep Hashing with Attribute Guidance

3.1. Problem formulation

Suppose we have n data points \( D = \{d_i\}_{i=1}^n \), \( d_i = (x_i, a_i, t_i) \). For each data point \( d_i \), \( x_i \) represents image visual feature. \( a_i = [a_{i1}, a_{i2}, \ldots, a_{ik}] \) is the attribute description annotation, where \( k \) is the number of attributes. \( a_{im} = 1 \) means the data point \( d_i \) has the \( m \)-th attribute where \( a_{im} = 0 \) means the data point \( d_i \) does not have the \( m \)-th attribute. \( t_i \) represents its category label. For our proposed method, similarity information is provided by a set of pair wise labels \( S = \{s_{ij}\} \) with \( s_{ij} \in \{0, 1\} \). \( s_{ij} = 1 \) means data point \( d_i \) and data point \( d_j \) have same category labels and are similar images while \( s_{ij} = 0 \) means data point \( d_i \) and data point \( d_j \) have different category labels and are dissimilar.

The goal of the proposed method is to learn a hashing function \( H(x_i) \) which generates a binary hash code \( b_i \in \{-1, 1\}^c \) for each image \( x_i \), where \( c \) is the code length. The binary codes \( B = \{b_i\}_{i=1}^n \) should perverse similarity in \( S \). If \( s_{ij} = 1 \), \( b_i \) and \( b_j \) have a small Hamming distance. Otherwise if \( s_{ij} = 0 \), \( b_i \) and \( b_j \) have a distant Hamming distance. The hashing function to learn is denoted as \( H(x_i) \rightarrow b_i \).

For two binary hash codes \( b_i \) and \( b_j \), their inner product \( < b_i, b_j > \) and their Hamming distance \( dis_B(b_i, b_j) \) can be formulated as:

\[
dis_H(b_i, b_j) = \frac{1}{2} (c - < b_i, b_j >)
\]

If two binary hash codes have a distant Hamming distance, their inner product should be small, and vice versa. The pair wise similarity probability of two data points is defined by a likelihood function [20]:

\[
P(s_{ij}|b_i, b_j) = \left\{ \begin{array}{ll} \sigma(b_i^T b_j), & s_{ij} = 1 \\ 1 - \sigma(b_i^T b_j), & s_{ij} = 0 \end{array} \right. \]

(2)

Where \( \sigma(x) = \frac{1}{1+\exp(-x)} \). A large condition probability \( p(1|b_i, b_j) \) implies a large inner product \( < b_i, b_j > \) and a small Hamming distance \( dis_B(b_i, b_j) \), which indicates image \( x_i \) and image \( x_j \) are similar, and vice versa.

Similar to hash codes, the pair wise similarity probability between two features \( f_i \) and \( f_j \) can be calculated by their inner product \( < f_i, f_j > \), which can be formulated as:

\[
P(s_{ij}|f_i, f_j) = \left\{ \begin{array}{ll} \sigma(f_i^T f_j), & s_{ij} = 1 \\ 1 - \sigma(f_i^T f_j), & s_{ij} = 0 \end{array} \right. \]

(3)

3.2. Proposed Method

As is shown in Figure 2, our hashing learning network consists of two parts: attribute refinement learning and image hash code learning. The former part learns refined attribute features for the corresponding attribute descriptions of images. The latter part learns a hash function to generate hash codes that keep information from both image visual features and image attribute descriptions.
3.2.1. Attribute Refinement Learning. In order to discover abundant information from image attribute description, we adopt a two fc layers network as an attribute refinement network denoted as $R(\cdot)$ and take $relu$ as the activation function to obtain refined features. For images $x_i$, the refined attribute features $e_i$ can be learnt from attribute descriptions $a_i$ which can be formulated as:

$$e_i = R(a_i, \theta^a)$$  \hspace{1cm} (4)

Where $\theta^a$ is the network parameter. For image $x_j$, a refined attribute feature $e_j$ can also be learnt from the corresponding attribute description $a_j$. The learnt refined attribute features $e_i$ and $e_j$ should keep the similarity from $S$. For all training data points, the Weighted Maximum Likelihood estimation of the refined attribute features $\{e_i\}_{i=1}^N$ for all data points can be formulated as:

$$logP(S|E) = \sum_{s_{ij}\in S} w_{ij} logP(s_{ij}|e_i, e_j)$$ \hspace{1cm} (5)

$$w_{ij} = \begin{cases} 1 & s_{ij} = 1 \\ 0 & s_{ij} = 0 \end{cases}$$ \hspace{1cm} (6)

Where $S = \{s_{ij} \in S : s_{ij} = 1\}$ is the set of similar pairs for all training datapoints. By taking equation (3) into equation (5), the objective optimization function of attribute refinement learning for all data points can be formulated as:

$$\min_{\theta^a} J_1 = \sum_{s_{ij}\in S} w_{ij} log \left( 1 + \exp(e_i^T e_j) \right) - s_{ij}(e_i^T e_j)$$ \hspace{1cm} (7)

By minimizing $J_1$, we can get refined features $e_i$ and $e_j$ that preserve attribute information as well as similarity between image $x_i$ and $x_j$.

3.2.2. Image Hash Code Learning. To obtain image visual feature, we employ Alexnet [23] denoted as $G(\cdot)$ and take the last fc layer output as image visual feature, since Alexnet has been proven effective in many visual tasks [24]. For image $x_i$, its visual feature $v_i$ can be obtained as follow:

$$v_i = G(x_i, \theta^v)$$ \hspace{1cm} (8)

Where $\theta^v$ is the network parameter. Next, we adopt a fc layer as hashing layer denoted as $H(\cdot)$ to get an intermediate representation $u_i$ for hash code learning and take $tanh$ as the activation function as follow:

$$u_i = H(tanh(v_i), \theta^h)$$ \hspace{1cm} (9)

Where $\theta^h$ is the network parameter. The binary code $b_i$ of image $x_i$ are obtained as follow:

$$b_i = sign(u_i)$$ \hspace{1cm} (10)
Similar to image \( x_i \), we can get an intermediate representation \( u_j \) and its binary code \( b_j \) for image \( x_j \). For images \( x_i \) and \( x_j \), their learned intermediate representations \( u_i \) and \( u_j \) should preserve similarity from \( S \). Similar to attribute refinement learning, the objective optimization problem for all data points can be formulated as follow:

\[
\min_{\theta_v, \theta_h} J_2 = \sum_{s_{ij} \in S} w_{ij} \log \left( 1 + \exp \left( u_i^T u_j \right) \right) - s_{ij} \left( u_i^T u_j \right) \tag{11}
\]

To take advantage of image attributes, intermediate representations should be learnt by the guidance of attribute refined features so that information from attributes can be maintained. The learning of representation \( u_i \) should be guided by the corresponding attribute feature \( e_i \) of image \( x_i \) so that its own attribute information can be kept. The objective function can be formulated as follow:

\[
\min_{\theta_v, \theta_h} J_3 = \sum_{s_{ij} \in S} w_{ij} \log \left( 1 + \exp \left( u_i^T e_i \right) \right) - s_{ij} \left( u_i^T e_i \right) \tag{12}
\]

The learning of intermediate representation \( u_i \) should also be guided by the attribute feature \( e_j \) of image \( x_j \) to keep similarity between images \( x_i \) and \( x_j \) from \( S \). The objective function can be formulated as follow:

\[
\min_{\theta_v, \theta_h} J_4 = \sum_{s_{ij} \in S} w_{ij} \log \left( 1 + \exp \left( u_i^T e_j \right) \right) - s_{ij} \left( u_i^T e_j \right) \tag{13}
\]

Similar to the learning of \( u_i \), the learning of intermediate representation \( u_i \) should be guided by its attribute refined features \( e_j \). The objective function can be formulated as follow:

\[
\min_{\theta_v, \theta_h} J_5 = \sum_{s_{ij} \in S} w_{ij} \log \left( 1 + \exp \left( u_i^T e_j \right) \right) - s_{ij} \left( u_i^T e_j \right) \tag{14}
\]

The learning of intermediate representation \( u_j \) should also be guided by the attribute feature \( e_i \) of image \( x_i \) to keep similarity between images \( x_i \) and \( x_j \) from \( S \). The objective function can be formulated as follow:

\[
\min_{\theta_v, \theta_h} J_6 = \sum_{s_{ij} \in S} w_{ij} \log \left( 1 + \exp \left( u_j^T e_i \right) \right) - s_{ij} \left( u_j^T e_i \right) \tag{15}
\]

For binary code learning, we take regularization terms as follow:

\[
\min_{\theta_v, \theta_h} J_7 = \| b_i - u_i \|_2^2 + \| b_j - u_j \|_2^2 \tag{16}
\]

The overall objective function is as follow:

\[
\min_{\theta_v, \theta_h} J = J_1 + J_2 + \beta (J_3 + J_4 + J_5 + J_6) + \eta J_7 \tag{17}
\]

Where \( \beta \) and \( \eta \) are hyper-parameters.

4. Experiments and Results

4.1. Datasets

The experiments are conducted on two benchmark image retrieval datasets: Caltech-UCSD Birds 200 [12] and Animal with attribute 2 [13]:

- Caltech-UCSD Birds 200 is an image dataset with photos of 200 bird categories. It has 11788 images in total. Each bird category has a 312-dimension attribute description vector and each dimension represents one attribute. We take the official dataset split for training and test. 5994 images are used as training data and retrieval base data while 5794 images are used as queries for test.

- Animal with attribute 2 is an image dataset with photos of 50 animal categories. It has 37322 images in total. Each animal category has a 80-dimension attribute description vector and each dimension represents one attribute. We randomly select 10000 images for training. For test, we select 5000 images as queries and take the rest 32322 images as retrieval base data.
4.2. Evaluation Metrics

The retrieval performance of hashing method is evaluated based on two metrics: Mean Average Precision (MAP) and Precision curves with respect to different numbers of top returned samples. Specially, we evaluate MAP on top 10 returned samples for Caltech-UCSD Birds 200 and on top 50 returned samples for Animal with attribute 2 with respect to 16 bit, 32 bit, 48 bit and 64 bit hash code. We evaluate precision on top 100, top 200, top 300 and top 400 returned samples for both datasets with respect to 32 bit and 64 bit hash code.

We compare our proposed method with 7 hashing methods: FSSH [17], HashNet [22], DPSH [20], DSH [8], ITQ [15], PCAH [16] and LSH [14].

4.3. Implementation Details

For our proposed methods, we utilize the Pytorch package [25] to construct our network architecture and perform backpropagation to update network parameters. Stochastic gradient decent (SGD) is applied for attribute refinement learning and image hash code learning. The initial learning rate is set to 0.001, momentum is set to 0.9 and weight decay is set to 0.0005. For the hyper-parameters, $\beta$ is set to 0.01 and $\eta$ is set to 0.0001.

For compared hashing methods, we adopt their publicly released code and the same experimental settings. For FSSH, ITQ, PCAH and LSH, we apply the pretrained Alexnet in Pytorch torchvision implementation Details to construct our network architecture and take the last fc layer outputs as image features for hash code learning.

4.4. Results

We conduct our experiments on two benchmark image retrieval datasets, Caltech-UCSD Birds 200 and Animal with attribute 2. The experimental results of MAP and Precision curves are demonstrated in Table 1, Figure 3 and Figure 4.

4.4.1. Map Results. The MAP results on two datasets are reported in Table 1. We can find that our method can achieve better performance compared with other 7 hashing methods. For instance, on Caltech-UCSD bird 200, with different length of hash codes, the MAP results of our proposed method shows a relative improvement of 1.36% ~ 3.10% against the second best result. On Animal with Attribute 2, our proposed method performs better and gains a relative improvement of 0.47% ~ 0.77%. Compared with other 7 methods, our proposed hashing method takes advantage of image attribute descriptions and utilizes an objective function based on pair wise similarity to maintain similarity among images as well as the corresponding attribute descriptions. For our proposed method, the learnt hashing function can generate hash codes with more information kept from image visual features and attributes.

| Methods  | Caltech-UCSD Birds 200 | Animal with attribute 2 |
|----------|------------------------|-------------------------|
|          | 16 bits | 32 bits | 48 bits | 64 bits | 16 bits | 32 bits | 48 bits | 64 bits |
| Ours     | 27.85   | 29.34   | 32.05   | 34.27   | 52.64   | 64.15   | 67.49   | 67.75   |
| HashNet  | 26.33   | 27.98   | 28.95   | 31.25   | 52.17   | 63.67   | 66.85   | 66.98   |
| DPSH     | 11.45   | 11.55   | 13.01   | 14.31   | 51.90   | 57.83   | 62.91   | 63.55   |
| DSH      | 3.24    | 4.28    | 4.48    | 4.7     | 13.12   | 14.85   | 15.53   | 15.84   |
| FSSH     | 1.18    | 1.4     | 1.43    | 1.64    | 3.35    | 6.03    | 5.71    | 6.4     |
| ITQ      | 7.81    | 8.07    | 15.43   | 14.11   | 33.57   | 42.77   | 46.43   | 55.08   |
| PCAH     | 7.33    | 9.35    | 11.64   | 12.01   | 22.38   | 35.05   | 35.71   | 41.07   |
| LSH      | 3.13    | 5.06    | 6.44    | 6.65    | 12.15   | 14.76   | 21.51   | 24.86   |

4.4.2. Precision curves. Figure 3 demonstrates the Precision curves with respect to top 100, top 200, top 300 and top 400 returned samples for 32 bit hash code and Figure 4 demonstrates the Precision curves for 64 bit hash code. It can be observed that our hashing method achieves better performance compared with other 7 hashing methods consistently on different number of top returned numbers for
both Caltech-UCSD Birds 200 and Animal with attribute 2. Compared with other 7 hashing methods, our proposed method utilizes the corresponding attribute descriptions to guide hash code learning, thus images of the same category can have hash codes with smaller Hamming distance while images of different categories can have hash codes with more distant Hamming distance.

5. Conclusion
In this paper, we propose a novel deep hashing method with image attribute guidance. By keeping similarity among images as well as image attribute descriptions, our proposed hashing method is able to generate hash codes that maintain information from both image visual features and image attribute descriptions. Extensive experimental results on two datasets, Caltech-UCSD Birds 200 and Animal with attribute 2, show that our proposed method can achieve better performance.

Reference
[1] Erin Liong V, Lu J, Wang G, Moulin P, Zhou J. Deep hashing for compact binary codes learning. In Proceedings of the IEEE conference on computer vision and pattern recognition 2015 (pp. 2475-2483).
[2] Lu J, Liong VE, Zhou J. Deep hashing for scalable image search. IEEE transactions on image processing. 2017 Mar 3;26(5):2352-67.
[3] Li WJ, Wang S, Kang WC. Feature learning based deep supervised hashing with pairwise labels, arXiv preprint arXiv:1511.03855. 2015 Nov 12.
[4] Lai H, Pan Y, Liu Y, Yan S. Simultaneous feature learning and hash coding with deep neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition 2015 (pp. 3270-3278).
[5] Lai H, Yan P, Shu X, Wei Y, Yan S. Instance-aware hashing for multi-label image retrieval. IEEE Transactions on Image Processing. 2016 Mar 22;25(6):2469-2479

[6] Zhao W, Guan Z, Luo H, Peng J, Fan J. Deep Multiple Instance Hashing for Object-based Image Retrieval. In IJCAI 2017 Aug 19 (pp. 3504-3510).

[7] Sohn S, Kim H, Kim J. Supervised hashing via uncorrelated component analysis. In Thirtieth AAAI Conference on Artificial Intelligence 2016 Feb 21.

[8] Zhang Z, Chen Y, Saligrama V. Efficient training of very deep neural networks for supervised hashing. In Proceedings of the IEEE conference on computer vision and pattern recognition 2016 (pp. 1487-1495).

[9] Zhu H, Long M, Wang J, Cao Y. Deep hashing network for efficient similarity retrieval. In Thirtieth AAAI Conference on Artificial Intelligence 2016 Mar 2.

[10] Yang E, Deng C, Liu W, Liu X, Tao D, Gao X. Pairwise relationship guided deep hashing for cross-modal retrieval. In Thirty-First AAAI Conference on Artificial Intelligence 2017 Feb 12.

[11] Dmochowski JP, Sajda P, Parra LC. Maximum likelihood in cost-sensitive learning: Model specification, approximations, and upper bounds. Journal of Machine Learning Research. 2010;11(Dec):3313-32.

[12] Wah C, Branson S, Welinder P, Perona P, Belongie S. The caltech-ucsd birds-200-2011 dataset.

[13] Xian Y, Lampert CH, Schiele B, Akata Z. Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly. IEEE transactions on pattern analysis and machine intelligence. 2018 Jul 19.

[14] Datar M, Immorlica N, Indyk P, Mirrokni VS. Locality-sensitive hashing scheme based on p-stable distributions. In Proceedings of the twentieth annual symposium on Computational geometry 2004 Jun 8 (pp. 253-262). ACM.

[15] Gong Y, Lazebnik S, Gordo A, Perronnin F. Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2012 Sep 6;35(12):2916-29.

[16] Kong W, Li WJ. Isotropic hashing. In Advances in neural information processing systems 2012. (pp. 1646-1654).

[17] Luo X, Nie L, He X, Wu Y, Chen ZD, Xu XS. Fast scalable supervised hashing. In The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval 2018 Jun 27 (pp. 735-744). ACM.

[18] Li WJ, Wang S, Kang WC. Feature learning based deep supervised hashing with pairwise labels. arXiv preprint arXiv:1511.03855. 2015 Nov 12.

[19] Liu H, Wang R, Shan S, Chen X. Deep supervised hashing for fast image retrieval. In Proceedings of the IEEE conference on computer vision and pattern recognition 2016 (pp. 2064-2072).

[20] Li WJ, Wang S, Kang WC. Feature learning based deep supervised hashing with pairwise labels. arXiv preprint arXiv:1511.03855. 2015 Nov 12.

[21] Jiang QY, Li WJ. Deep cross-modal hashing. In Proceedings of the IEEE conference on computer vision and pattern recognition 2017 (pp. 3232-3240).

[22] Cao Z, Long M, Wang J, Yu PS. Hashnet: Deep learning to hash by continuation. In Proceedings of the IEEE international conference on computer vision 2017 (pp. 5608-5617).

[23] Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems 2012 (pp. 1097-1105).

[24] Yuan ZW, Zhang J. Feature extraction and image retrieval based on AlexNet. In Eighth International Conference on Digital Image Processing (ICDIP 2016) 2016 Aug 29 (Vol. 10033, p. 100330E).International Society for Optics and Photonics.

[25] Paszke A, Gross S, Chintala S, Chanan G, Yang E, DeVito Z, Lin Z, Desmaison A, Antiga L, Lerer A. Automatic differentiation in pytorch.