IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 4, 2020, accepted July 19, 2020, date of publication July 22, 2020, date of current version August 14, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3011102

Image Retrieval Using a Deep Attention-Based Hash

XINLU LI[1], MENGFEI XU[1,3], JIABO XU[2,4], THOMAS WEISE[1],
LE ZOU[1], FEI SUN[1], AND ZHIZE WU[1]

[1]School of Artificial Intelligence and Big Data, Institute of Applied Optimization, Hefei University, Hefei 230601, China
[2]Department of Software, Nanchang Hangkong University, Nanchang 330063, China
[3]School of Software, Nanchang University, Nanchang 330047, China
[4]Information Engineering School, Nanchang University, Nanchang 330031, China

Corresponding author: Zhize Wu (wuzhize@mail.ustc.edu.cn)

This work was supported in part by the Science Foundation for Research Development of Hefei University under Grant 18ZR06ZDA, 19ZR05ZDA, and 2019d07020002, in part by the Anhui Natural Science Research Project under Grant 908085QF285 and 1908085MF184, and in part by the Hefei University Talent Research under Grant 18-19RC25 and 18-19RC26, and in part by the National Natural Science Foundation of China under Grant 61673359.

ABSTRACT Image retrieval is becoming more and more important due to the rapid increase of the number of images on the web. To improve the efficiency of computing the similarity of images, hashing has moved into the focus of research. This paper proposes a Deep Attention-based Hash (DAH) retrieval model, which combines an attention module and a convolutional neural network to obtain hash codes with strong representability. Our DAH has the following features: The Hamming distance between the hash codes generated by similar images is small and the Hamming distance of hash codes of dissimilar images has a larger constant value. The quantitative loss from Euclidean distance to Hamming distance is minimized. DAH has a high image retrieval precision: We thoroughly compare it with ten state-of-the-art approaches on the CIFAR-10 dataset. The results show that the Mean Average Precision (MAP) of DAH reaches more than 92% in terms of 12, 24, 36 and 48 bit hash codes on CIFAR-10, which is better than what the state-of- art methods used for comparison can deliver.

INDEX TERMS Content-based image retrieval, depth-wise separable convolution kernel, Hamming distance, pairwise loss.

I. INTRODUCTION

How to guarantee the efficiency and accuracy of image retrieval is a very challenging problem. Content-Based Image Retrieval (CBIR) [1], [2] is a promising computer vision technique, used to implement queries based on content-based visual similarity, for example, color, texture and shape. In CBIR, image representations and similarity measures are two critical design choices. How to quickly and accurately retrieve images from large-scale image data sets is particularly challenging. Traditional CBIR methods [3], [4] are not efficient on a large-scale corpus because of their high computational cost. Hashing is a practical strategy to speed-up this process [5], [6].

A hash function is applied to arbitrary data and produces data of a fixed, usually small, size. Hashing is being increasingly used for approximating the nearest instances for image retrieval, especially in large-scale scenarios. Hashing image retrieval presents high-dimensional raw images as compact low-dimensional codes, and calculates the similarity between images according to their Hamming distance [1], [2]. With the ability to represent richer information within little storage capacity, hashing can effectively reduce the memory requirement and computational load. Hashing receives increasing attention from the CBIR community and is widely used to approximate nearest neighbor retrieval [7], [8].

Deep hashing has become a research hotspot due to the advancement of deep learning and its ability to learn the semantic features and hash encoding for images simultaneously [9]–[14]. As the neural network deepens, the semantic information that it can represent becomes more

The associate editor coordinating the review of this manuscript and approving it for publication was Claudio Cusano.

comprehensive. The emergence of the convolutional neural network (CNN) AlexNet [15] in 2012 was a significant turning point in deep learning research. Subsequently, VGGNet [16], GoogLeNet [17], and ResNet [18] have successively increased the performance. Several CBIR methods using deep learning based on CNNs [19]–[23] have been proposed. By adopting CNNs to extract rich semantic features from images and using hashing technology to obtain binary codes for representing images, these methods provide high image retrieval precision. Deep learning is powerful, but also faces problems in terms of storage capacity and computing efficiency [24]. Liu *et al.* [9] presented a shallow neural network structure (DSH) for reducing the storage requirements and improving the efficiency.

Noh *et al.* [10] developed DELF, using convolution to strengthen the learning of local features, and Wei *et al.* [11] proposed the SCDA method for extracting fine-grained features. The improvement of the hash coding ability is mainly reflected in the design of the loss function. The Siamese Network [12] is a pair-based method, which encourages positive samples to approach and set apart the distance between negative samples. The triplet network was proposed in [13]: each triple contains a positive and a negative sample pair. This triplet model aims to make sure that the similarity of the negative pair is always lower than that of the positive pair. Ge *et al.* developed the hierarchical triple loss (HTL) [14]. HTL constructs a three-level hierarchical tree of all image categories and trains accordingly.

We propose to learn more effective hash functions and hash codes by improving the unsupervised learning module and loss function. Given the excellent performance of attention in visual recognition [25]–[30], some existing methods [54]–[56] try to introduce the visual attention mechanism into the deep model, so as to achieve more robust feature learning. In [54], Shu *et al.* adopt an attention mechanism to quantify the contribution of a certain motion by measuring the consistency between itself and the whole activity under the Global Context Coherence (GCC) constraint. Inspired by this, we integrate an attention module into the neural network and further study the problem of how to build a uniform and deep framework for CBIR by combining a hashing and an attention module.

We present the deep attention-based hash coding method (DAH) with pairwise tag information for large-scale image retrieval. Figure 1 illustrates the DAH framework. We use a deep residual network as the backbone and integrate the Convolutional Block Attention Module (CBAM) [30] to enhance the feature representation. The images and labels of the training set are organized into pairs. We train our model using a logarithmic loss function under the supervised information of paired images and labels. Finally, we build the classification layer after the hash encoding layer. Our model attains the ability of positive correlation optimization by jointly solving these two tasks. We evaluate the performance of DAH in a comprehensive experiment. We find that it has significant advantages regarding the retrieval precision of the *n* nearest neighbor images in comparison with the state-of-the-art.

Our contributions are summed up as follows:

1. We construct the deep attention-based hash network (DAH). The attention module is seamlessly integrated into the neural network. The proposed method effectively extracts the semantic features of the data, thus significantly improving the image retrieval precision.
2. We organize paired batches of images as input sources, which have both classified One-Hot tags and similarity information. The semantic similarity representation and classification recognition ability are interconnected and optimized, such that the proposed model can improve the level of classification while achieving strong hash encoding capabilities.
3. We improve upon the original sigmoid activation function and design a logarithmic hash lossfunction. This not only leads to a smooth gradient change between the hash layer and the classification layer, but also allows the trained model to produce similarity features.

The remaining sections of this paper is organized as follows. Section 2 provides a brief overview of the related work on hash retrieval methods. In Section 3, we introduce our novel deep attention-based hash retrieval model along with a thorough theoretical analysis. Section 4 provides the performance evaluation and discussion on the CIFAR-10 dataset. Finally, the conclusions and outlook on future work are discussed in Section 5.

II. RELATED WORK

A. CONVENTIONAL HASHING

The representation of image similarity has always been a hot spot for research, especially fast approximate nearest neighbor search. Unsupervised retrieval method provides a hash function that uses unlabeled samples for training. Locality Sensitive Hashing (LSH) [3] is a characteristic unsupervised neighbor hash search approach. LSH maps similar image to similar binary codes by utilizing random projections. However, in order to obtain high retrieval precision, LSH usually requires large codes, resulting in high memory consumption. Spectral Hashing (SH) [31] combines hashing and spectral analysis to produce compact binary codes by thresholding with non-linear functions along the principal component analysis directions of the image data. PCA-Hash (PCA-H) [4] constructs a hash function by extracting the intrinsic relationship between data. This hash function can be continuously updated in order to minimize errors. Iterative Quantization (ITQ) [32] minimizes the quantization error between the binary code and the original data via a rotation matrix.

Supervised retrieval methods generate a hash function by labeling the samples, which usually produces a compact binary-coded representation. Many studies have shown that supervised models can obtain richer semantic information from labeled samples, and have higher search precision than unsupervised methods [33]–[35]. Supervised Hashing with

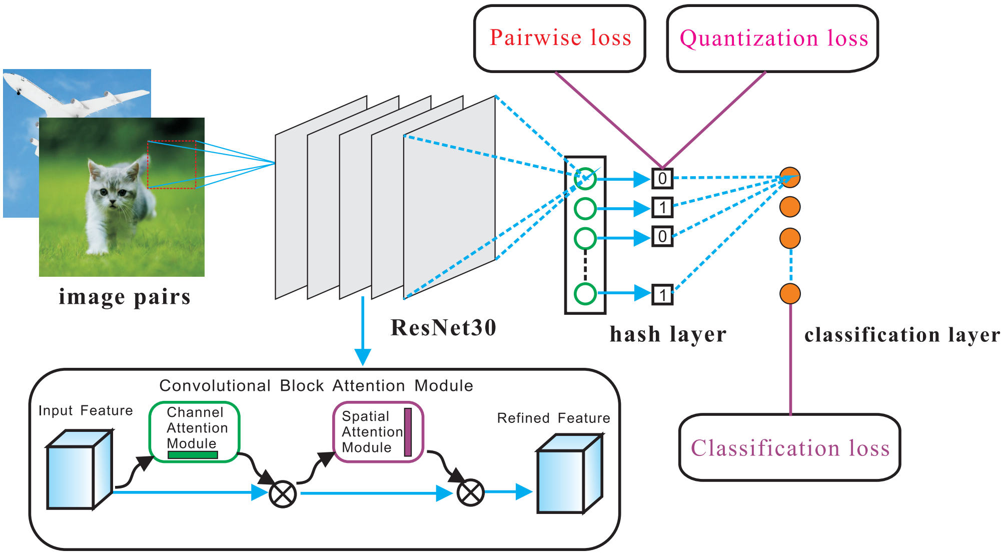


FIGURE 1. Framework of the proposed DAH.

Kernels (KSH) [34] is a representative supervised learning retrieval method. KSH minimizes the Hamming distance of the binary codes representing similar image pairs. At the same time, it also maximizes the Hamming distance between different pairs. Supervised Discrete Hashing (SDH) [35] integrates the generation of hash codes with the training of linear classifiers and the expected retrieval effect is also improved. Column Sampling Based Discrete Supervised Hashing (COSDISH) [33], another discrete supervised hashing approach, is implemented by iteratively sampling columns from a similarity matrix and optimizing the sampled information alternately.

B. DEEP HASHING

Hierarchical rich mid-level feature representation in deep learning often can capture the semantic information of images better than conventional algorithms. CNNH [19] is an early-stage deep learning-based hashing algorithm with excellent performance. It demonstrated the potential of CNNs as hashing methods. Network In Network Hashing (NINH) [20] preserves image similarity through triplet ranking loss. Deep Supervised Hashing (DSH) [9] and Deep Pairwise Supervised Hashing (DPSH) [36] are based on pairwise loss, which was shown to achieve high precision. Deep Regularized Similarity Comparison Hashing (DRSCH) [23] makes use of the similarity between image pairs as a regular term while also using triple loss. These methods are proposed to provide a universal solution for CBIR tasks. Deep hashing algorithms have received widespread attention. The works [37], [38] propose supervised hashing for face queries and fast image queries and [39], [40] conduct research on deep hashing for cross-modal image queries. Supervised Deep Hashing Perception (SDHP) [6] is a loss calculation method combining paired loss and quantized loss, which further improves the retrieval precision. The Deep Incremental Hashing Network (DIHN) [41] is a hash method for incremental retrieval. While maintaining retrieval precision and reducing training time, Deep Saliency Hashing (DSaH) [8] uses attention networks and combines semantic loss, saliency loss, and quantization loss to give the model fine-grained retrieval capabilities.

Recently, several advanced deep hash models have appeared, such as [53], [57], [58]. Sun *et al.* [53] propose an end-to-end supervised hierarchical cross-modal hashing method, consisting of two key components: the hierarchical discriminative learning and regularized cross-modal hashing. In [57], Jin *et al.* develop deep ordinal hashing (DOH), which learns ordinal representations to generate ranking-based hash codes by leveraging the ranking structure of feature space from both local and global views. Similar to [57], Lai *et al.* [58] propose a deep-networks-based hashing method for multi-label image retrieval, by incorporating automatically generated region proposals and label probability calculations in the hash learning process.

The goal of our new image retrieval method DAH is to improve retrieval efficiency, precision, and stability by using a high-quality hash coding. We propose to learn the hash function by integrating an attention module into the CNN model. We construct the backbone architecture of the DAH model based on the RestNet50 structure, while reducing the amount of network parameters to decrease the time consumption of model updates. By introducing an attention module into the DAH, our model avoids excessive computational loads, while improving the retrieval precision. Finally, we design a batch-based loss function for the end-to-end learning mechanism, thus optimizing the model as a whole.

III. DEEP ATTENTION-BASED HASH MODEL FOR IMAGE RETRIEVAL

We now discuss the details of our deep attention-based hash model for image retrieval (DAH). The structure of the DAH is the unified end-to-end CNN framework shown

TABLE 1. ResNet30 backbone network structure (serving CIFAR-10 data set).

<table>
<tr><td rowspan="7">Conv Block</td><td>SeparableConv2D</td><td rowspan="5">SeparableConv2D</td><td rowspan="7">Identity Block</td><td>SeparableConv2D</td><td rowspan="6">Input Feature</td></tr>
<tr><td>BN+Relu</td><td>BN+Relu</td></tr>
<tr><td>SeparableConv2D</td><td>SeparableConv2D</td></tr>
<tr><td>BN+Relu</td><td>BN+Relu</td></tr>
<tr><td>SeparableConv2D</td><td>SeparableConv2D</td></tr>
<tr><td>BN</td><td>BN</td><td>BN</td></tr>
<tr><td colspan="2">Add+Relu</td><td colspan="2">Add+Relu</td></tr>
<tr><td rowspan="23">Backbone</td><td>Type</td><td colspan="2">Input Size</td><td colspan="2">Output Size</td></tr>
<tr><td>BN</td><td colspan="2">32*32*3</td><td colspan="2">32*32*3</td></tr>
<tr><td>ZeroPadding</td><td colspan="2">32*32*3</td><td colspan="2">34*34*3</td></tr>
<tr><td>Conv2D:(3*3)/1</td><td colspan="2">34*34*3</td><td colspan="2">32*32*64</td></tr>
<tr><td>BN+Relu</td><td colspan="2">32*32*64</td><td colspan="2">32*32*64</td></tr>
<tr><td>ZeroPadding</td><td colspan="2">32*32*64</td><td colspan="2">34*34*64</td></tr>
<tr><td>Maxpolling:(3*3)/1</td><td colspan="2">34*34*64</td><td colspan="2">32*32*64</td></tr>
<tr><td>CBAM</td><td colspan="2">32*32*64</td><td colspan="2">32*32*64</td></tr>
<tr><td>Conv Block</td><td colspan="2">32*32*64</td><td colspan="2">16*16*256</td></tr>
<tr><td>Identity Block</td><td colspan="2">16*16*256</td><td colspan="2">16*16*256</td></tr>
<tr><td>Identity Block</td><td colspan="2">16*16*256</td><td colspan="2">16*16*256</td></tr>
<tr><td>CBAM</td><td colspan="2">16*16*256</td><td colspan="2">16*16*256</td></tr>
<tr><td>Conv Block</td><td colspan="2">16*16*256</td><td colspan="2">8*8*512</td></tr>
<tr><td>Identity Block</td><td colspan="2">8*8*512</td><td colspan="2">8*8*512</td></tr>
<tr><td>Identity Block</td><td colspan="2">8*8*512</td><td colspan="2">8*8*512</td></tr>
<tr><td>CBAM</td><td colspan="2">8*8*512</td><td colspan="2">8*8*512</td></tr>
<tr><td>Conv Block</td><td colspan="2">8*8*512</td><td colspan="2">4*4*1024</td></tr>
<tr><td>Identity Block</td><td colspan="2">4*4*1024</td><td colspan="2">4*4*1024</td></tr>
<tr><td>Identity Block</td><td colspan="2">4*4*1024</td><td colspan="2">4*4*1024</td></tr>
<tr><td>GlobalAvgPooling</td><td colspan="2">4*4*1024</td><td colspan="2">1*1024</td></tr>
<tr><td>Linear</td><td colspan="2">1*1024</td><td colspan="2">1*k</td></tr>
<tr><td>Shrink Sigmoid</td><td colspan="2">1*k</td><td colspan="2">1*k</td></tr>
<tr><td>Output+Softmax</td><td colspan="2">1*k</td><td colspan="2">1*10</td></tr>
</table>

in Figure 1. First, an attention-based deep hash network is presented to learn a hash function from the training set. Second, a batch-based logarithmic loss function is developed, with the aim of making the Hamming distance of image hashes of the same category as small as possible, while maintaining a constant distances for hashes of images from different categories. This improves the hash code quality. Finally, the hyper-parameters of the model and our method are explained.

A. FORWARD STRUCTURE

ResNet50 [18] shows excellent results in image recognition, detection and segmentation on the ImageNet [42] and COCO [43] data set. This kind of network with residual structure is often used and has achieved great performance in CBIR tasks [5], [7], [44], [45]. However, a large number of stacks at the network level will inevitably result in an explosive growth of the number of parameters. This results in greatly reduced computing efficiency. In our proposed DAH network, we only use a core structure of 30 layers of ResNet50 (see Table 1). Furthermore, DAH uses the depth-wise separable convolution kernel from Xception [46] to replace the traditional convolution structure. Xception has a higher recognition accuracy on the ImageNet dataset than Inception V3 [47], which uses the traditional convolution kernel with the same number of parameters.

In order to complement the feature representation of our model, we introduce the attention module CBAM [30] into three important intermediate nodes in ResNet30. Through emphasizing or suppressing intermediate features, the intermediate feature map is adaptively refined along the spatial and channel dimensions in CBAM. The number of parameters of ResNet30 with attention module are reduced by 87.48% compared to ResNet50. Therefore, the structure we used allows for a much faster computation of hashes.

The top-level structure in the forward structure is a crucial part in improving the precision of the hash search. We append a fully-connected layer between the classification layer and the global average pooling layer, to improve upon the original top-level structure of ResNet50. The amount of nodes in the added fully-connected layer is k, which is equal to

TABLE 2. Comparison of DAH and state-of-the-art methods based on MAP. The highest indicator is bold and placed on the first line. *means that the values are taken from the original papers.

Task	Methods	Code Length			
		12 bits	24 bits	36 bits	48 bits
MAP In CIFAR-10	**DAH**	**0.9251**	**0.9219**	**0.9267**	**0.9231**
	NoneA-DAH	0.9208	0.9206	0.9235	0.9189
	BGAN	0.884	0.889	-	0.894
	HashGAN	0.668	0.731	-	0.749
	PGDH	0.866	0.874	-	0.877
	LSH	0.1217	0.1218	0.1434	0.1417
	PCAH	0.1311	0.1290	0.1255	0.1235
	SH	0.1268	0.1242	0.1238	0.1282
	ITQ	0.1548	0.1649	0.1668	0.1684
	DSH	0.1454	0.1567	0.1589	0.1652
	SDH	0.4054	0.5139	0.5347	0.5377
	COSDISH	0.4804	0.5118	0.5413	0.5493
	CNNH*	0.4650	0.5210	-	0.5320
	NINH*	0.5520	0.5660	-	0.5810
	SDHP*	0.8318	0.8684	0.8755	0.8767

the length of the hash code. We explore the performance of DAH for 12, 24, 36, and 48 bits. The fully-connected layer is used to calculate the output, i.e., the hash code, and is also responsible for the model classification results, especially matching the gradient descent in the back structure. To attain a smooth distribution of the expected outcomes, we modify the activation function of the hash layer shown as follows:

$$shrink \cdot sigmoid(x) = \frac{1}{4} + \frac{1}{2\left(1 + e^{-x}\right)} \tag{1}$$

Here, x stands for the output of the hash layer in the DAH framework. The shrink sigmoid function above performs a non-linear conversion on the output layer. Compare with the original sigmoid version, the shrink sigmoid function reduces the mapping transformation range from [0, 1] to [1/4, 3/4]. This modification is beneficial to the transfer of parameters. Based on Equation (1), we enhance the dependency relationship between the hash layer and the classification layer. There is a positive correlation gradient optimization direction during the back propagation process in the output of these two layers.

Our network with the default size as given in Table 1 can represent RGB images of 32*32 pixels, i.e., 3072 numbers with 8 bits each (24576 bits in total), as hash codes of k bits length. Of course, other input sizes are also possible.

B. BACK STRUCTURE

The back structure determines the direction of the model change. It calculates partial derivatives of the loss function in the opposite direction of the forward structure, thereby iteratively tuning the parameters. In this procedure, the objective function implies the direction for the model update and has a huge impact on the performance. With the aim to get high-quality model parameters, we develop a novel loss function based on the characteristics of the classification, encoding, and quantization used in our scenario.

First, we utilize the softmax classification function in the classification layer. We use the cross-entropy loss method, which is often adopted in classification tasks.

Second, to obtain a high-quality binary encoding from the training, we organize pairs of image samples as input of the model. The output of the hash layer in the model is the source for obtaining binary encoded information. Assuming that ω represents the image space, then for the paired images I_1, I_2, the Euclidean space obtained through the hash layer can be expressed numerically as $\omega \rightarrow [1/4, 3/4]^k$. The hash code of each image can be represented as a vector of length k. The output for the paired samples I_1, I_2 of the hash layer be the vector x_1, x_2. Our design aims for assigning similar hash codes to similar images, while the hash codes of dissimilar images should have a larger constant Hamming distance. To ensure the smoothness of the training transition, we adopt

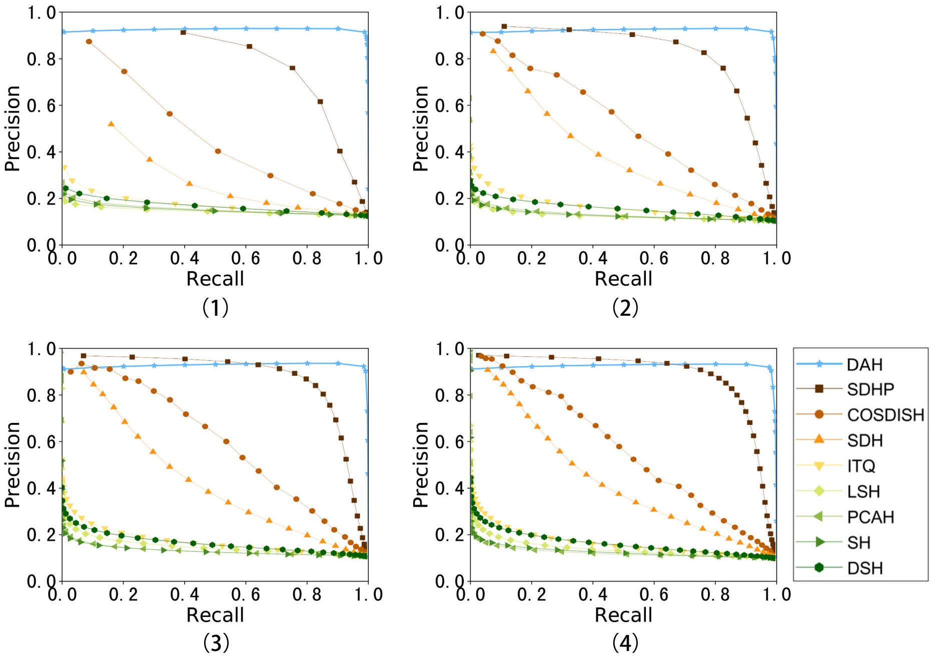


FIGURE 2. P-R curves with different hash bits. (1) 12 bits. (2) 24 bits. (3) 36 bits. (4) 48 bits.

a logarithmic loss function within (0.5, 1].

$$W_1(x_1, x_2) = \begin{cases} \|x_1 - x_2\|^2 & S = 1 \\ \left| t + \sum_{i=1}^{k} \log(1 - |x_1 - x_2|) \right| & S = 0 \end{cases} \quad (2)$$

$$S = \begin{cases} 1 & I_1 \text{ is similar to } I_2 \\ 0 & I_1 \text{ and } I_2 \text{ are not similar} \end{cases} \quad (3)$$

Here t denotes the threshold value, which we set to $k/4.8$ in the experimental analysis. This Equation indicates that the difference between two similar images in the output space before quantization is measured in terms of the Euclidean distance. When the Euclidean distance is smaller, the loss function is smaller. Also, the distance of dissimilar images is calculated by the logarithmic expression in Equation (2) and the loss value is minimal only when this distance is equal to the threshold t.

Finally, the real vectors x_1, x_2 need to be transformed to the binary hash codes b_1, b_2. Ideally the transformed binary codes b_1, b_2 over all images should be distributed uniformly in order to increase the information capacity of the hash function, which also reduces the quantization loss when mapping from the Euclidean to the binary space. With this in mind, we re-defined the loss function in the original space as follows.

$$W_2(x_1, x_2) = (\sum_{i=1}^{2} \sum_{j=1}^{j=k} (x_i^j - 0.5))^2 \quad (4)$$

Since the x_1, x_2 value range is [1/4, 3/4], Equation (4) ensures that the variable output through the hash layer fluctuates around 0.5. The mapping from x_1, x_2 to b_1, b_2 can be expressed as follows.

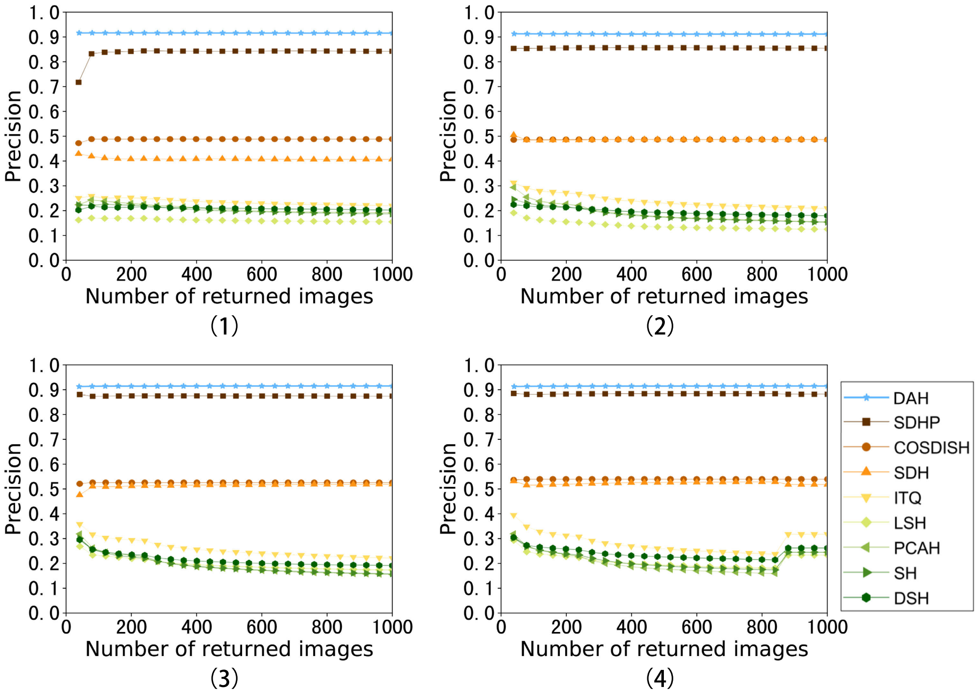
$$b^i = \begin{cases} 1 & \mathit{if}\ x^i > \frac{1}{2} \\ 0 & \mathit{if}\ x^i < \frac{1}{2} \end{cases} \quad (5)$$

Here i is the index of the bit to be encoded and W_1, W_2 are two loss functions, which are used for the hash layer optimization. W_1 enables the training of the neural network for the similarity features between images. W_2 is an auxiliary function to ensure the uniform distribution of the output results of the hash layer, such that the learned coding information has the maximum information capacity. We adopt the mini-batch gradient descent method (MBGD) to ensure smooth improvements when training the model.

C. TRAINING

Adopting the stochastic gradient descent method for training using single pairs of samples may result in a locally optimal configuration. Therefore, we choose the general mini-batch gradient descent method (MBGD) in the whole hash network. The batch size is set to twice the number of categories. Thus, each batch contains almost all possible permutations and combinations of different categories. The overall loss function of the hash layer can be expressed as:

$$W = \frac{1}{A_{2n}^2} \sum_{i=1}^{2n} \sum_{j=1}^{2n} [\varphi_1 W_1(x_i, x_j) + \varphi_2 W_2(x_i, x_j)] \quad i \neq j \quad (6)$$



(1)

(2)

(3)

(4)

FIGURE 3. Precision curve regarding top-n with different hash sizes. (1) 12 bits. (2) 24 bits. (3) 36 bits. (4) 48 bits.

Here, φ_1 and φ_1 are the hyper-parameters to control the influence of W_1 and W_2, respectively, and allow to establish a trade-off between feature similarity and uniform hash code distribution. We determined their appropriate values through experimental cross-validation and found that setting both to 1 gives good results in our experiments. Since n denotes the number of categories, the batch size value is $2n$, and A_{2n}^2 is the possible number of all combinations in the current batch.

For training the model, we utilize the gradient optimization method Adam [48] and follow the recommended settings: We set the exponential decay rate of the first and the second moment estimate in Adam to 0.9 and 0.999, respectively. The initial learning rate is 0.001. Our initial experiments proved that such a design has a positive effect on the training of model parameters.

IV. EXPERIMENTS AND DISCUSSIONS

In order to assess the performance of the proposed DAH method, we perform a large number of evaluations on the CIFAR-10 dataset. Of course, DAH is also applicable to other data sets or retrieval tasks. We first present the basic information and evaluation approach. Then we compare the proposed DAH with several representative hashing image retrieval models, such as LSH [3], SH [31], PCAH [4], ITQ [32], SDH [35], and COSDISH [33] from the field of conventional hashing, and the deep hashing models CNNH [19], NINH [20], SDHP [6], BGAN [50], HashGAN [51], and PGDH [52]. We also design experiments with and without using the attention module in DAH.

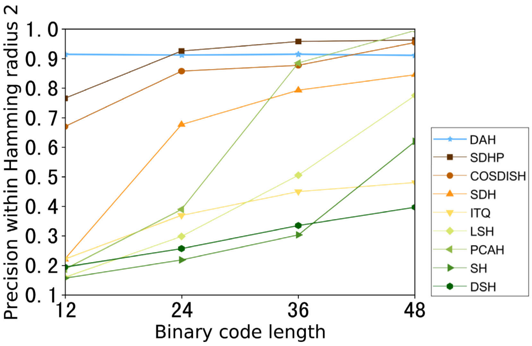


FIGURE 4. Retrieval precision of Hamming distance within 2.

The hyper-parameters φ_1, φ_2, are both set to 1 using cross-validation. The partition setting of CIFAR-10 is fixed. All the results reported for the comparison methods are with the same experimental setting. All the results reported in this paper follow the protocol used in [6].

A. DATA SET AND EVALUATION METHODS

The CIFAR-10 [49] dataset contains 60,000 color images with a size of 32 × 32 pixels. It is divided into ten categories, each of which contains 6000 images. The dataset has 50,000 images for training and the remaining 10,000 images are to be used for testing. The complexity and diversity of this data set are relatively high. If such a stress test shows good results, it proves that the proposed DAH model is reliable. In the CIFAR-10 dataset, two images are identified to be

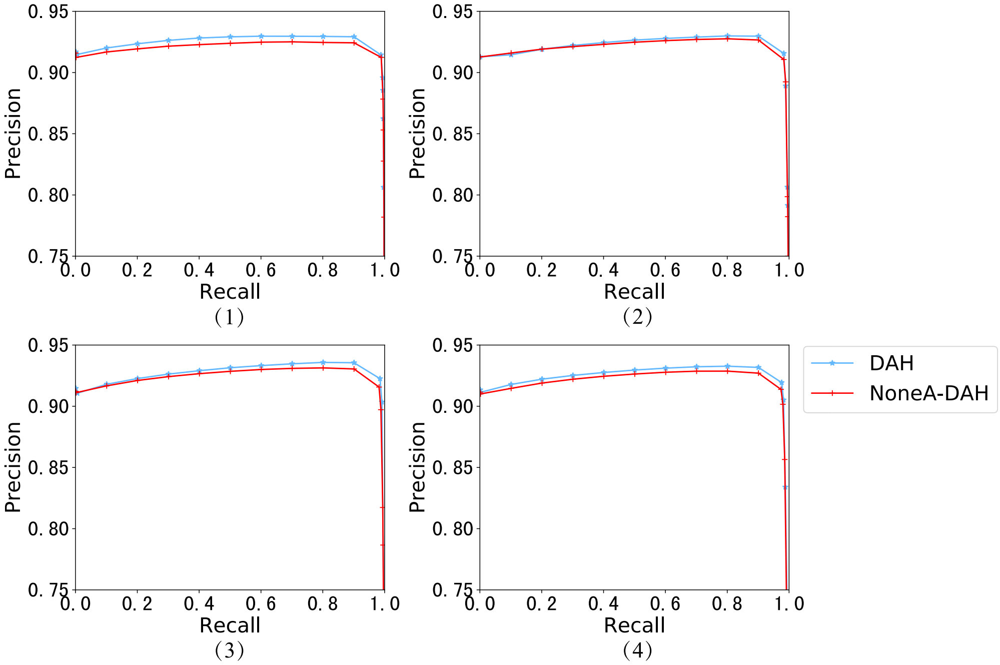


FIGURE 5. P-R curve regarding different number of bits. (1) 12 bits. (2) 24 bits. (3) 36 bits. (4) 48 bits.

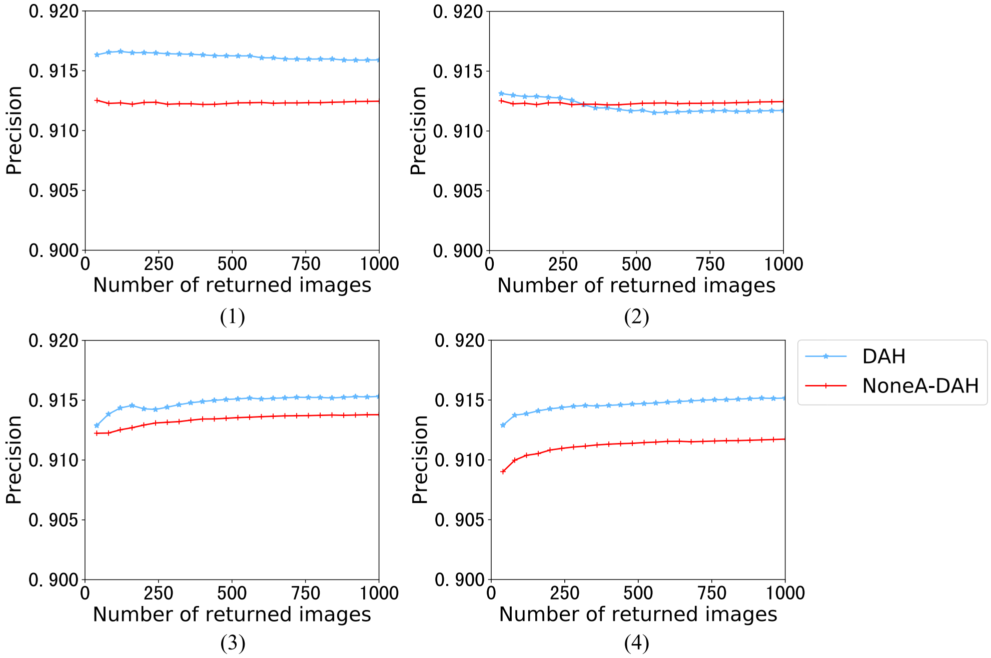


FIGURE 6. Precision curves regarding the top-n indicator for different hash sizes. (1) 12 bits. (2) 24 bits. (3) 36 bits. (4) 48 bits.

semantically similar if they have the same label. Their hash codes should then be similar as well. If two images have different labels, they are semantically dissimilar and their hash codes should have a larger constant Hamming distance.

We adopt four widely used evaluation criteria: (1) the Mean Average Precision (MAP), (2) the Precision-Recall (P-R) curve based on Hamming distance ranking, (3) the Top-1000 precision rate based on Hamming distance ranking, and (4) the search precision of the hashes within Hamming distances of at most 2. We also list the performance indicators for our DAH method without the attention module, which we call NoneA-DAH. The two variants are compared and discussed in detail in Section IV.C, while the following Section IV.B focusses on DAH vs. the related work.

TABLE 3. Comparisons in the top-1000 precision index. The highest indicator is bold and placed on the first line. * means that the values are taken from the original papers.

Task	Methods	Code Length			
		12 bits	24 bits	36 bits	48 bits
Precision In CIFAR-10	**DAH**	**0.9159**	0.9117	**0.9153**	**0.9152**
	NoneA-DAH	0.9125	**0.9124**	0.9138	0.9117
	LSH	0.1480	0.1579	0.1992	0.1994
	PCAH	0.1833	0.1863	0.1818	0.1784
	SH	0.1784	0.1852	0.1842	0.1925
	ITQ	0.2123	0.2400	0.2458	0.2523
	DSH	0.1946	0.2110	0.2165	0.2317
	SDH	0.3967	0.5079	0.5311	0.5363
	COSDISH	0.4775	0.5082	0.5382	0.5463
	SDHP*	0.8272	0.8645	0.8724	0.8748

It should be noted that for BGAN [50], HashGAN [51], and PGDH [52], only the MAP results on CIFAR-10 are reported in their corresponding publications.

B. RESULTS ON THE CIFAR-10 DATASET

Table 2 demonstrates the MAP values of the algorithms used in our experiment on the CIFAR-10 dataset. We set the length of the hash code to 12, 24, 36, and 48 bits, respectively. We find that DAH performs better than all other approaches and beats BGAN, which ranks second, by a 3% margin. The MAP of DAH is also 6.1% better in average than SDHP, which also uses paired sample inputs. This indicates that our DAH is feasible and has significant advantages.

Figures 2 (1) to (4) show the P-R curves for hash sizes of 12, 24, 36, and 48 bits, respectively. As can be seen, the DAH method is again superior to the other methods, especially on longer hash codes. We make four observations: (1) As the recall rate increases, the precision of DAH shows an overall upward trend. (2) Compared with the other methods, DAH has a clear lead in precision, especially when the recall rate is high. (3) When the recall rate is not high, there are several compared methods, such as SDHP and COSDISH, which have higher precision. However, the gaps are not too wide and still acceptable. In addition, good performance does not only mean a high precision, but also a high recall. (4) In terms of stability, our method performs best. With this, we can argue that the ResNet30 model structure we designed can extract image similarity features and the gradient optimization of the mini-batch loss function can make better use of the supervision information.

TABLE 4. Encoding time using different hashing methods for 48-bits hash codes.

Task	Method	Encoding time in μs
Encoding Time in CIFAR-10	LSH	2.86
	PCAH	3.28
	SH	2.37
	ITQ	7.99
	DSH	2.68
	COSDISH	1870
	SDH	21700
	SDHP+	9830
	DAH	5280
	NoneA-DAH	4860

Table 3 illustrates the precision of the top 1,000 queries in terms of Hamming on CIFAR-10. Here, DAH outperforms SDHP by 8.87%, 4.72%, 4.29%, and 4.04% for 12, 24, 36, and 48 bits, respectively. Figure 3 demonstrates similar results. The reason behind the high precision of DAH is that it adopts the loss function shown in Equation (2). This loss function reduces the distance between images of the same category and maintains a constant distance between dissimilar images. We can also see from Figure 3 that DAH is more stable. The retrieval precisions at different hash sizes all exceed 91%.

Figure 4 presents the retrieval precision for different hash sizes on CIFAR-10 when the Hamming radius is 2. The time complexity of such a Hamming ranking of n images is only

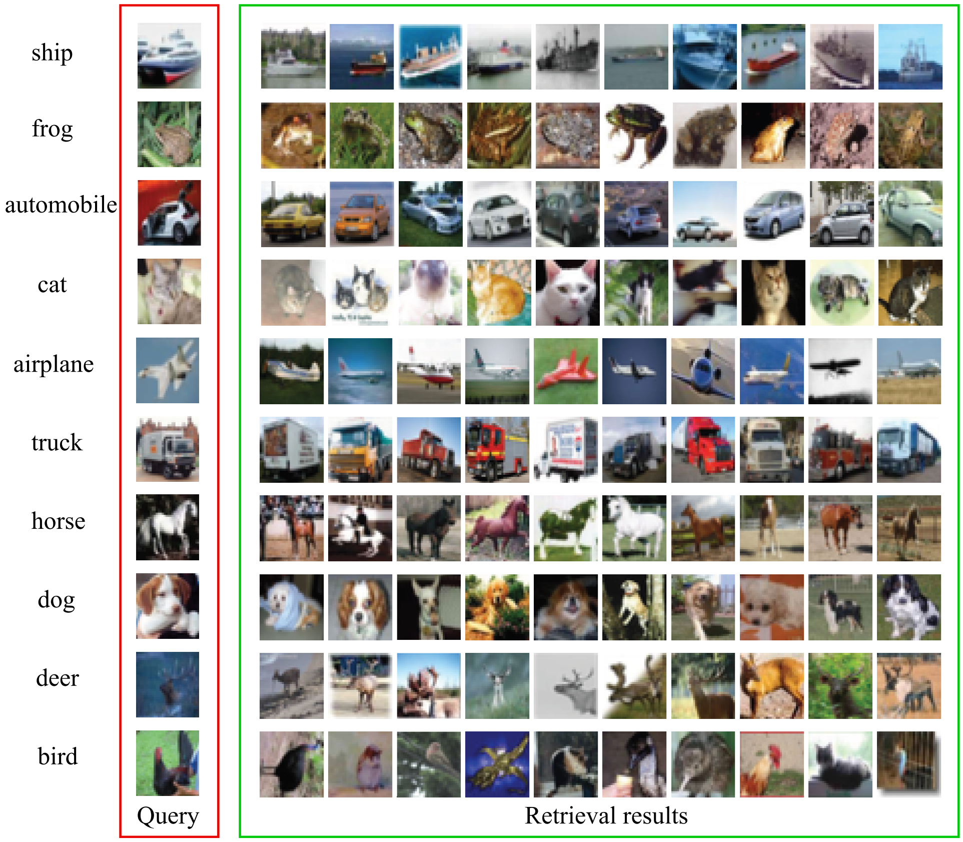


FIGURE 7. Retrieval samples using NoneA-DAH.

$O(n)$ [6], which makes it an efficient image retrieval tool. Our DAH has basically the same retrieval precision for all hash sizes. It outperforms the other methods for 12 bits and always maintains a high precision of over 90%. This is attributable to the shrink sigmoid activation function proposed in Equation 1, which leads to high precision even for small hash sizes. On larger hash sizes, SDHP, PCAH, and COSDISH perform slightly better in this performance metric. The performance of our DAH method is very stable over different hash lengths, which certainly is another important positive feature.

C. IMPACT OF THE ATTENTION MODULE

We now implement a set of experiments to verify the effectiveness of the attention module. For this purpose, we only retain our novel loss function, our proposed training method, and improved network structure — but remove the attention module. We refer to our method without the attention module as NoneA-DAH. Table 2 also lists the MAP of NoneA-DAH. We find that the attention module improves the precision of DAH over NoneA-DAH by 0.43%, 0.13%, 0.32%, and 0.42% at 12, 24, 36, and 48 bits, respectively. Even seemingly smaller improvements are important and can have a tangible impact on larger datasets. On the other hand, an MAP of 0.9267 is a very competitive result, even compared with the latest state-of-the-art method [50]. At this level, gaining another 0.5 percentage points can be considered a valuable improvement. Similar results can be found in Figure 5, which implies that the proposed attention module plays a positive role in terms of the P-R curves, too. It is noteworthy that the MAP values of both models peak at hash lengths of 36 bits. This indicates that 36 bits are more suitable for use in the hash-based image retrieval task if we only consider retrieval precision, at least on the CIFAR-10.

Table 3 also includes the top-1000 precision of NoneA-DAH on the CIFAR-10 dataset. DAH has better precision than NoneA-DAH in most cases (12, 36, and 48 bits). The exception are 24 bit hashes, for which NoneA-DAH is better than DAH, but only by 0.07%. Figure 6 shows the precision curves regarding the top-n indicator for different hash sizes. The precision of DAH is higher than NoneA-DAH in most cases, but NoneA-DAH still outperforms the compared state-of-the-art algorithms.

Inspired by the illustrations in [50], Figures 7 and 8 provide some visualized examples of image retrieval using NoneA-DAH and DAH, respectively. Some false positives appear in Figure 7. The retrieval effect of DAH is better than that of NoneA-DAH, for images with birds.

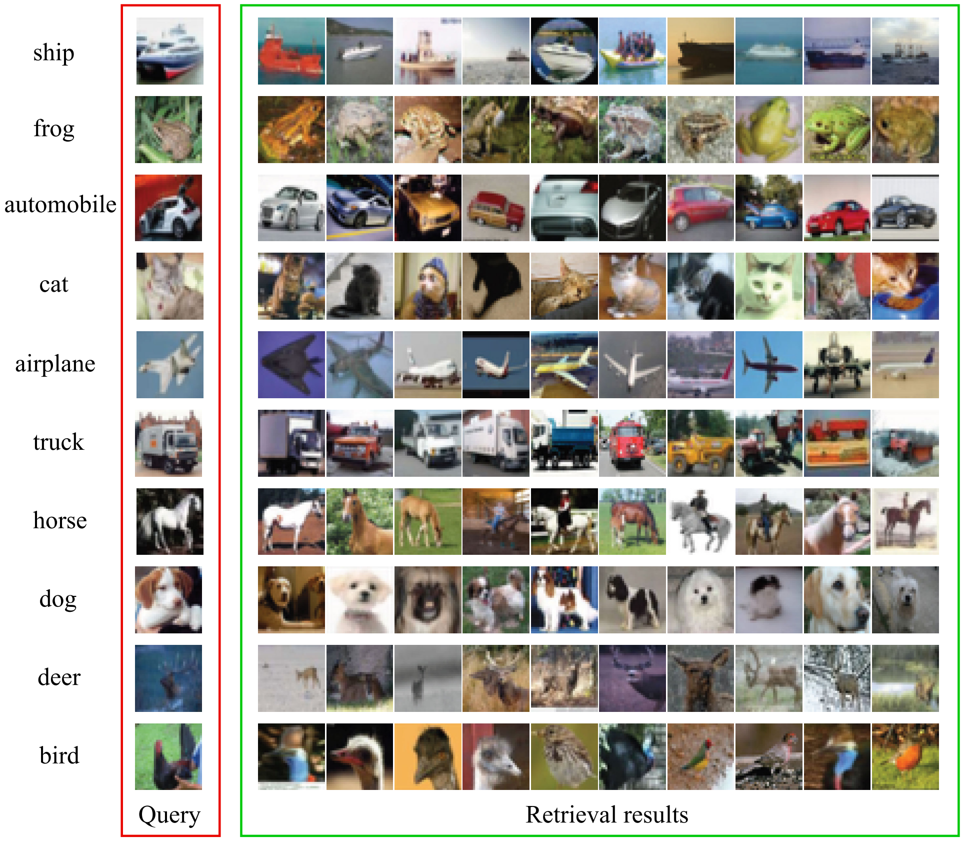


FIGURE 8. Retrieval samples using DAH.

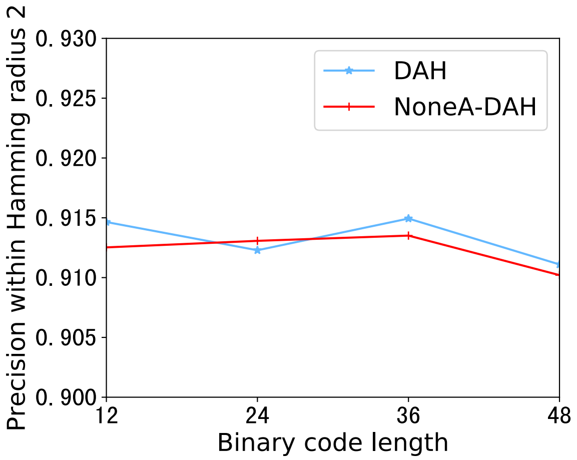


FIGURE 9. Retrieval precision of within Hamming distance 2.

Figure 9 reports the retrieval precision when the Hamming radius is 2. Overall, the precision of DAH is slightly better than NoneA-DAH for hash codes of 12, 36, and 48 bits.

D. COMPUTATIONAL TIME

We now investigate the encoding time to assess the computational requirements of DAH. The encoding time is determined as the average over the 50,000 training set images. Table 4 illustrates the encoding time of all investigated hashing methods on CIFAR-10. Compared to traditional hash retrieval algorithms, deep hash retrieval methods are generally slower. The proposed DAH approach has significantly better performance than similar methods such as SDHP +, which adopts GoogLeNet [17] as backbone network. The encoding time of NoneA-DAH is better than DAH. This indicates that in some cases, NoneA-DAH may be a good choice, as it already can improve retrieval efficiency and maintain a high retrieval precision.

TABLE 5. Uniformity of the distribution of the binary codes.

Methods	Code Length			
	12 bits	24 bits	36 bits	48 bits
DAH	1.4766	1.0603	1.0429	1.0134
NoneA-DAH	1.3142	1.0323	1.0595	1.0328

E. UNIFORM DISTRIBUTION OF BINARY CODES

One important criterion for testing the quality of hash coding procedures is whether they achieve a uniform distribution of the generated binary codes. Having a uniform distribution indicates a high information representation capacity.

We therefore compute the ratio of 1 and 0 values at each bit index. A loss function is effective when distributing the binary values evenly and the ratio approaches 1. The results on CIFAR-10 are summarized in Table 5. For DAH, the result approaches 1 more closely when larger hash sizes are used. NoneA-DAH, on the other hand, achieves a slightly more even distribution for shorter hash codes.

V. CONCLUSION

We presented a novel image retrieval model called Deep Attention-based Hash (DAH) Network. Our DAH learns supervised information end-to-end. We develop a pairwise loss function for training the ability of capturing correlation information between images. By this, we can obtain discriminative binary codes for image retrieval. In addition, we include an attention module into our network structure to further improve the precision. Our experimental results on the CIFAR-10 dataset show that our DAH method is superior to the state-of-the-art methods with respect to the most important performance metrics. DAH reaches mean average image retrieval precisions of 92.51%, 92.19%, 92.67%, and 92.31% for hash codes of sizes 12, 24, 36, and 48 bits, respectively. This exceeds the best result of the conventional methods, obtained by COSDISH with 54.93%, and the best result attained by deep hashing methods, which is 89.4% (by BGAN). We also confirm experimentally that the inclusion of the attention module in DAH is indeed helpful. We plan to consider different model structures and analyze the effects of these structures on the retrieval precision in our future work. In addition, we will evaluate other modules besides CBAM for inclusion and assess their impact on the CBIR task.

ACKNOWLEDGMENT

Dr. Weise is especially thankful for the support of the city Hefei under the Hefei Specially Recruited Foreign Expert program.

DISCLOSURE

All authors declare that there are no conflicts of interests. All authors declare that they have no significant competing financial, professional or personal interests that might have influenced the performance or presentation of the work described in this manuscript. Declarations of interest: none. All authors approve the final article.

REFERENCES

[1] D. Zhang, A. Wong, M. Indrawan, and G. Lu, "Content-based image retrieval using Gabor texture features," in *Proc. 1st IEEE PacificRim Conf. Multimedia*, Sydney, NSW, Australia, Dec. 2000, pp. 392–395.

[2] T. He, Y. Wei, Z. Liu, G. Qing, and D. Zhang, "Content based image retrieval method based on SIFT feature," in *Proc. Int. Conf. Intell. Transp., Big Data Smart City (ICITBS)*, Xiamen, China, Jan. 2018, pp. 649–652, doi: 10.1109/ICITBS.2018.00169.

[3] A. Gionis, P. Indyk, and R. Motwani, "Similarity search in high dimensions via hashing," in *Proc. VLDB*, Edinburgh, U.K., Sep. 1999, pp. 518–529.

[4] J. Wang, S. Kumar, and S. F. Chang, "Sequential projection learning for hashing with compact codes," in *Proc. ICML*, Haifa, Israel, 2010, pp. 1127–1134.

[5] B.-H. Qiang, P.-L. Wang, S.-P. Guo, Z. Xu, W. Xie, J.-L. Chen, and X.-J. Chen, "Large-scale multi-label image retrieval using residual network with hash layer," in *Proc. 11th Int. Conf. Adv. Comput. Intell. (ICACI)*, Guilin, China, Jun. 2019, pp. 262–267, doi: 10.1109/ICACI.2019.8778549.

[6] C. Yan, H. Xie, D. Yang, J. Yin, Y. Zhang, and Q. Dai, "Supervised hash coding with deep neural network for environment perception of intelligent vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 1, pp. 284–295, Jan. 2018, doi: 10.1109/TITS.2017.2749965.

[7] S. Ayyachamy, V. Alex, M. Khened, and G. Krishnamurthi, "Medical image retrieval using Resnet-18," presented at the Med. Imag. Imag. Informat. Healthcare, Res., Appl., San Diego, CA, USA, Mar. 15, 2019. [Online]. Available: https://doi.org/10.1117/12.2515588

[8] S. Jin, H. Yao, X. Sun, S. Zhou, L. Zhang, and X. Hua, "Deep saliency hashing for fine-grained retrieval," *IEEE Trans. Image Process.*, vol. 29, pp. 5336–5351, 2020, doi: 10.1109/TIP.2020.2971105.

[9] H. Liu, R. Wang, S. Shan, and X. Chen, "Deep supervised hashing for fast image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 2064–2072.

[10] H. Noh, A. Araujo, J. Sim, T. Weyand, and B. Han, "Large-scale image retrieval with attentive deep local features," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 3456–3465.

[11] X.-S. Wei, J.-H. Luo, J. Wu, and Z.-H. Zhou, "Selective convolutional descriptor aggregation for fine-grained image retrieval," *IEEE Trans. Image Process.*, vol. 26, no. 6, pp. 2868–2881, Jun. 2017, doi: 10.1109/TIP.2017.2688133.

[12] A. Gordo, J. Almazán, J. Revaud, and D. Larlus, "End-to-End learning of deep visual representations for image retrieval," *Int. J. Comput. Vis.*, vol. 124, no. 2, pp. 237–254, Sep. 2017, doi: 10.1007/s11263-017-1016-8.

[13] E. Hoffer and N. Ailon, "Deep metric learning using triplet network," in *Proc. Int. Workshop Similarity-Based Pattern Recognit.*, Copenhagen, Denmark, Oct. 2015, pp. 84–92, doi: 10.1007/978-3-319-24261-3_7.

[14] W. Ge, "Deep metric learning with hierarchical triplet loss," in *Proc. ECCV*, vol. 1. Boston, MA, USA, Jun. 2015, pp. 3270–3278.

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. NIPS*, Lake Tahoe, NV, USA, Dec. 2012, pp. 1097–1105.

[16] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556

[17] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 1–9.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778.

[19] R. Xia, Y. Pan, H. Lai, C. Liu, and S. Yan, "Supervised hashing for image retrieval via image representation learning," in *Proc. 28th AAAI Conf. Artif. Intell.*, Montreal, QC, Canada, Jul. 27-31, 2014.

[20] H. Lai, Y. Pan, Y. Liu, and S. Yan, "Simultaneous feature learning and hash coding with deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 3270–3278.

[21] F. Zhao, Y. Huang, L. Wang, and T. Tan, "Deep semantic ranking based hashing for multi-label image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 1556–1564.

[22] K. Lin, H.-F. Yang, J.-H. Hsiao, and C.-S. Chen, "Deep learning of binary hash codes for fast image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Boston, MA, USA, Jun. 2015, pp. 27–35.

[23] R. Zhang, L. Lin, R. Zhang, W. Zuo, and L. Zhang, "Bit-scalable deep hashing with regularized similarity learning for image retrieval and person re-identification," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 4766–4779, Dec. 2015, doi: 10.1109/TIP.2015.2467315.

[24] S. Han, J. Pool, J. Tran, and W. Dally, "Learning both weights and connections for efficient neural network," in *Proc. NIPS*, Montreal, QC, Canada, Dec. 2015, pp. 1135–1143.

[25] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 3156–3164.

[26] V. Mnih, N. Heess, and A. Graves, "Recurrent models of visual attention," in *Proc. NIPS*, Montreal, QC, Canada, Dec. 2014, pp. 2204–2212.

[27] B. Zhao, X. Wu, J. Feng, Q. Peng, and S. Yan, "Diversified visual attention networks for fine-grained object classification," *IEEE Trans. Multimedia*, vol. 19, no. 6, pp. 1245–1256, Jun. 2017, doi: 10.1109/TMM.2017.2648498.

[28] J. Hu, L. Shen, and G. Sun, "Squeeze-and-Excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 7132–7141.

[29] M. Jaderberg, K. Simonyan, and A. Zisserman, "Spatial transformer networks," in *Proc. NIPS*, Montreal, QC, Canada, Dec. 2015, pp. 2017–2025.

[30] S. Woo, J. Park, J. Y. Lee, and I. So Kweon, "CBAM: Convolutional block attention module," in *Proc. ECCV*, Munich, Germany, Sep. 2018, pp. 3–19.

[31] Y. Weiss, A. Torralba, and R. Fergus, "Spectral hashing," in *Proc. NIPS*, Vancouver, BC, Canada, Dec. 2008, pp. 1753–1760.

[32] Y. Gong, S. Lazebnik, A. Gordo, and F. Perronnin, "Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 12, pp. 2916–2929, Dec. 2013, doi: 10.1109/TPAMI.2012.193.

[33] W. C. Kang, W. J. Li, and Z. H. Zhou, "Column sampling based discrete supervised hashing," in *Proc. 13th AAAI Conf. Artif. Intell.*, Phoenix, AZ, USA, Feb. 2016, pp. 1230–1236.

[34] W. Liu, J. Wang, R. Ji, Y.-G. Jiang, and S.-F. Chang, "Supervised hashing with kernels," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Providence, RI, USA, Jun. 2012, pp. 2074–2081, doi: 10.1109/CVPR.2012.6247912.

[35] F. Shen, C. Shen, W. Liu, and H. T. Shen, "Supervised discrete hashing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 37–45.

[36] W.-J. Li, S. Wang, and W.-C. Kang, "Feature learning based deep supervised hashing with pairwise labels," 2015, *arXiv:1511.03855*. [Online]. Available: http://arxiv.org/abs/1511.03855

[37] J. Tang, J. Lin, Z. Li, and J. Yang, "Discriminative deep quantization hashing for face image retrieval," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 12, pp. 6154–6162, Dec. 2018, doi: 10.1109/TNNLS.2018.2816743.

[38] H. Zhu, M. Long, J. Wang, and Y. Cao, "Deep hashing network for efficient similarity retrieval," in *Proc. 13th AAAI Conf. Artif. Intell.*, Phoenix, AZ, USA, Feb. 2016, pp. 2415–2421.

[39] C. Deng, Z. Chen, X. Liu, X. Gao, and D. Tao, "Triplet-based deep hashing network for cross-modal retrieval," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 3893–3903, Aug. 2018, doi: 10.1109/TIP.2018.2821921.

[40] X. Shi, M. Sapkota, F. Xing, F. Liu, L. Cui, and L. Yang, "Pairwise based deep ranking hashing for histopathology image classification and retrieval," *Pattern Recognit.*, vol. 81, pp. 14–22, Sep. 2018, doi: 10.1016/j.patcog.2018.03.015.

[41] D. Wu, Q. Dai, J. Liu, B. Li, and W. Wang, "Deep incremental hashing network for efficient image retrieval," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 9069–9077.

[42] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015, doi: 10.1007/s11263-015-0816-y.

[43] T. Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, Zurich, Switzerland, Sep. 2014, pp. 740–755, doi: 10.1007/978-3-319-10602-1_48.

[44] L. Liu, H. Liu, T. Chen, Q. Shen, and Z. Ma, "Codedretrieval: Joint image compression and retrieval with neural networks," in *Proc. IEEE Vis. Commun. Image Process. (VCIP)*, Sydney, NSW, Australia, Dec. 2019, pp. 1–4, doi: 10.1109/VCIP47243.2019.8965918.

[45] P. Tschandl, G. Argenziano, M. Razmara, and J. Yap, "Diagnostic accuracy of content-based dermatoscopic image retrieval with deep classification features," *Brit. J. Dermatol.*, vol. 181, no. 1, pp. 155–165, 2019. [Online]. Available: https://doi.org/10.1111/bjd.17189.

[46] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 1251–1258.

[47] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 2818–2826.

[48] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980

[49] A. Krizhevsky and G. Hinton, *Learning Multiple Layers of Features From Tiny Images*. Toronto, ON, Canada: Toronto Univ., Apr. 2009.

[50] J. Song, T. He, L. Gao, X. Xu, and A. Hanjalic, "Unified binary generative adversarial network for image retrieval and compression," *Int. J. Comput. Vis.*, to be published. [Online]. Available: https://link.springer.com/article/10.1007/s11263-020-01305-2#citeas, doi: 10.1007/s11263-020-01305-2.

[51] Y. Cao, B. Liu, M. Long, and J. Wang, "HashGAN: Deep learning to hash with pair conditional wasserstein GAN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 1287–1296, doi: 10.1109/CVPR.2018.00140.

[52] X. Yuan, L. Ren, J. Lu, and J. Zhou, "Relaxation-free deep hashing via policy gradient," in *Proc. Eur. Conf. Comput. Vis.*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds. Munich, Germany: Springer, 2018, pp. 134–150, doi: 10.1007/978-3-030-01225-0_9.

[53] C. Sun, X. Song, F. Feng, W. X. Zhao, H. Zhang, and L. Nie, "Supervised hierarchical cross-modal hashing," in *Proc. 42nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, New York, NY, USA, 2019, pp. 725–734, doi: 10.1145/3331184.3331229.

[54] X. Shu, L. Zhang, Y. Sun, and J. Tang, "Host-parasite: Graph LSTM-in-LSTM for group activity recognition," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Apr. 2, 2020, doi: 10.1109/TNNLS.2020.2978942.

[55] J. Tang, X. Shu, R. Yan, and L. Zhang, "Coherence constrained graph LSTM for group activity recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Jul. 15, 2019, doi: 10.1109/TPAMI.2019.2928540.

[56] R. Yan, J. Tang, X. Shu, Z. Li, and Q. Tian, "Participation-contributed temporal dynamic model for group activity recognition," in *Proc. ACM Multimedia Conf. Multimedia Conf. (MM)*, 2018, pp. 1292–1300, doi: 10.1145/3240508.3240572.

[57] L. Jin, X. Shu, K. Li, Z. Li, G.-J. Qi, and J. Tang, "Deep ordinal hashing with spatial attention," *IEEE Trans. Image Process.*, vol. 28, no. 5, pp. 2173–2186, May 2019, doi: 10.1109/TIP.2018.2883522.

[58] H. Lai, P. Yan, X. Shu, Y. Wei, and S. Yan, "Instance-aware hashing for multi-label image retrieval," *IEEE Trans. Image Process.*, vol. 25, no. 6, pp. 2469–2479, Jun. 2016, doi: 10.1109/TIP.2016.2545300.

XINLU LI received the M.S. degree in computer science from Anhui University, Anhui, China, in 2009, and the Ph.D. degree in computer science from Technological University Dublin (TU Dublin), Dublin, Ireland, in 2019.

Since 2010, he has been a Lecturer with the School of Artificial Intelligence and Big Data, Hefei University. His research interests include swarm intelligence optimization algorithms and artificial intelligence.

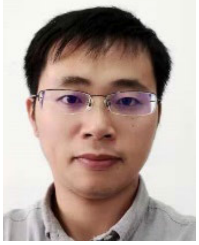
MENGFEI XU received the bachelor's degree in computer science and technology from Hefei University, in 2020. He is currently pursuing the master's degree in software engineering with Nanchang University. His current research interest includes computer vision.

JIABO XU received the bachelor's degree in software engineering from Nanchang Hangkong University, in 2019. He is currently pursuing the master's degree with the Information Engineering School, Nanchang University. His current research interest includes computer vision.

THOMAS WEISE received the Diplom Informatiker degree (equivalent to M.Sc. degree in computer science) from the Chemnitz University of Technology, in 2005, and the Ph.D. degree in computer science from the University of Kassel, in 2009. He then joined the University of Science and Technology of China (USTC) as a Postdoctoral Researcher and subsequently became an Associate Professor at the USTC-Birmingham Joint Research Institute in Intelligent Computation and Its Applications (UBRI), USTC. In 2016, he joined Hefei University as a Full Professor to found the Institute of Applied Optimization (IAO) at the Faculty of Computer Science and Technology. He has more than 80 scientific publications in international peer-reviewed journals and conferences. His book *Global Optimization Algorithms—Theory and Application* has been cited 840 times. He has acted as a Reviewer, an Editor, or a Program Committee Member at 70 different venues and is a member of the Editorial Board of the *Applied Soft Computing* journal.

LE ZOU received the M.S. degree from the Hefei University of Technology, Hefei, China, in 2008. He is currently pursuing the Ph.D. degree with the Institute of Intelligent Machine, Chinese Academy of Sciences. He is also an Associate Professor with Hefei University. His research interests include rational interpolation, image processing, and image segmentation.

FEI SUN received the M.S. degree from the Chinese Academy of Sciences, Hefei, China, in 2004. His main research interests include public safety technical prevention and automatic control. He is a member of the Anhui Information Standardization Technical Committee, the Anhui Safety Technology Prevention Association, the Anhui Safety Production, a National Registered Information Security Evaluator (CISP), an Anhui Provincial Laboratory Qualification Reviewer, and an Anhui Meteorological Bureau Member of the Expert Group, such as Lightning Protection and Grounding and Anhui Quality Association.



ZHIZE WU received the Ph.D. degree from the School of Computer Science and Technology (SCST), University of Science and Technology of China (USTC), in 2017. He is currently a Researcher with the School of Artificial Intelligence and Big Data, Institute of Applied Optimization, Hefei University. His research interests include image processing, neural networks, and machine learning.