Proceedings of the 2023 International Conference on Machine Learning and Automation
DOI: 10.54254/2755-2721/43/20230850

# Single-loss hash image retrieval method based on improved visual transformer

**Huanjie Pei[1,2], Zhijie Wang[1,3]**

[1]Donghua University

[2]woshipeihuanjie@163.com
[3]wangzj@dhu.edu.cn

**Abstract.** Deep hashing methods have gained popularity in image retrieval due to their advantages such as low storage requirements and high efficiency. However, existing deep hashing methods for large-scale image retrieval tasks suffer from issues including low discriminative power of binary hash codes, difficult optimization of losses, and low retrieval accuracy. This paper proposes a single-loss hash image retrieval method based on an improved visual transformer to address these issues. The proposed method utilizes a pre-trained Vision Transformer (ViT) on ImageNet as the backbone network, augmented with a hash coding layer to extract image features more comprehensively. Additionally, we design a single learning objective loss function that addresses the discriminative power of hash codes and quantization errors, thereby eliminating the complexity of adjusting various loss weights. Experimental evaluations on ImageNet100, NUS-WIDE, CIFAR10, and MS-COCO datasets demonstrate the superior performance of the proposed method compared to contemporary methods, indicating its adaptability to diverse data.

**Keywords:** Hashing; Image Retrieval; Discriminative Power; Quantization Error.

## 1. Introduction

In image retrieval tasks, the hash features extracted by deep neural networks possess advantages such as high storage efficiency and fast querying speed, attracting widespread attention from researchers [1-2]. Currently, image hashing retrieval methods primarily rely on supervised deep learning techniques for representation learning [3]. These methods combine the strengths of deep learning and hashing learning, enhancing retrieval accuracy while maintaining retrieval efficiency. In recent years, significant progress has been made in deep hashing methods [4] compared to traditional hashing methods [5], with deep hashing methods allowing for grouping based on the similarity measure of hash codes. This implies that deep hashing methods represent a new research direction for large-scale problems [6].

This paper proposes a single-loss hash image retrieval method based on an improved visual transformer. The method leverages a pre-trained ViT [7] model to extract features from input images and utilizes an additional hash coding module to fine-tune the backbone network for more comprehensive feature extraction. Simultaneously, we design a single learning objective loss function that maximizes the cosine similarity between continuous codes and binary codes to maximize the inter-class Hamming distance, while minimizing quantization errors. The experimental results on public

DOI: 10.54254/2755-2721/43/20230850

datasets demonstrate the significant advantages of the proposed method compared to other advanced hash image retrieval methods.

## 2. Related Work

### *2.1. Binary Optimization*

Hashing is an NP-hard binary optimization problem [8], and due to the discrete and non-differentiable nature of binary hash functions, the problem of gradient vanishing often arises during model training. Early methods attempted to alleviate the gradient vanishing problem by abandoning discrete constraints [9], while some methods used gradient descent for training [10] in an effort to overcome the issue. However, these methods required balancing hyperparameters between different learning objectives, increasing the complexity of model learning. To address the issue of gradient vanishing and simplify the model complexity, Su et al. proposed the concept of Greedy Hash [11], which involved designing a new encoding layer that generates binary hash codes using a sign function during forward propagation and utilizes a straight-through estimator [12] for gradient backpropagation during optimization. However, this method resulted in high losses, exacerbating the difficulty of model learning. Li et al. [13] designed a parameter-less encoding layer, Bi-half, maximizing bit capacity by shifting the network output using the median. However, these methods often required modifications to the computation graph, making the original graph no longer end-to-end trainable, thereby increasing the complexity of optimization objectives. To address these issues, we propose a single learning objective loss function to eliminate the problem of gradient vanishing and reduce the complexity of model learning, further enhancing the retrieval performance of the model.

### *2.2. Cosine Similarity*

While most current work focuses on image hashing with various constraints, this paper rephrases the problem of deep hashing under cosine similarity, inspired by Zhang et al. [14], who utilized cosine similarity to find the closest approximations between binary and ternary representations. This paper uses cosine similarity to interpret quantization errors. Additionally, deep hypersphere embedding learning methods (such as SphereFace, CosFace, and ArcFace) impose discriminative constraints on hypersphere manifolds and propose improving decision boundaries by utilizing cosine or angular margins. Inspired by the above, we utilize the concept of decision boundaries in the loss function to further improve intra-class variance, minimizing intra-class Hamming distance while maximizing inter-class Hamming distance.

## 3. Model Architecture

This paper introduces an improved visual Transformer model that can be trained end-to-end. By utilizing a pre-trained ViT model as a universal feature extraction module, the model removes the MLP module and replaces it with a hash coding module to fine-tune the main network for comprehensive feature extraction.

The overall process of the model is as follows: first, the input image $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^{N} \mathbb{R}^{N \times d}$ (where d is the dimension of the retrieved image and N is the number of training samples) is fed into the backbone network to obtain continuous code $\mathbf{V} = \{\mathbf{v}_n\}_{n=1}^{N} \in \mathbb{R}^{N \times K}$ (where K is the number of binary encoding bits). Subsequently, the zero-mean continuous code is obtained through batch normalization in the hash coding layer, and the scaled cosine similarity between the continuous code and its binary orthogonal target $\mathbf{o}_i \in [\mathbf{o}_1, \cdots, \mathbf{o}_C]^{\top} = \mathbf{O} \in \{-1, +1\}^{C \times K}$ is computed, where C is the number of classes. Finally, the scaled cosine similarity serves as the model output and is fed into the cross-entropy loss for forward propagation and backward derivation.

DOI: 10.54254/2755-2721/43/20230850

## 4. Loss Function Design

### *4.1. Expressing Hamming Distance as Cosine Similarity*

Calculating the Hamming distance between binary codes $\mathbf{b}_i$ and $\mathbf{b}_j$ is accomplished using the logical XOR operation. If b is represented by $\{-1,+1\}^K$, the Hamming distance can also be mathematically computed as follows:

$$D(\mathbf{b}_i, \mathbf{b}_j) = \frac{K - \mathbf{b}_i^\top \mathbf{b}_j}{2} \tag{1}$$

The dot product $\mathbf{b}_i^\top \mathbf{b}_j$ can be reinterpreted from a geometric perspective as:

$$\mathbf{b}_i^\top \mathbf{b}_j = \| \mathbf{b}_i \| \| \mathbf{b}_j \| \cos\theta_{ij} \tag{2}$$

where $\| \cdot \|$ denotes the Euclidean norm, $\theta_{ij}$ represents the angle between $\mathbf{b}_i$ and $\mathbf{b}_j$. As both $\| \mathbf{b}_i \|$ and $\| \mathbf{b}_j \|$ are constants, equation (1) can be represented as:

$$D(\mathbf{b}_i, \mathbf{b}_j) = \frac{K - K\cos\theta_{ij}}{2} = \frac{K}{2}(1 - \cos\theta_{ij}) \tag{3}$$

Due to $\frac{K}{2}$ being a constant, we observe that the current retrieval will be based solely on the angle between the two hash codes, indicating that similar hash codes will have similar directions, resulting in smaller angles and consequently smaller Hamming distances.

Typically, the conversion of continuous codes ***v*** into binary codes ***b*** leads to information loss, also known as quantization error. Consequently, most existing hashing methods include quantization error minimization in their learning objectives, such as L1 norm, L2 norm, and p norms (e.g., p = 3 in Greedy Hash), generally expressed as:

$$minL + \lambda Q$$

where L represents supervised learning objectives such as Cross Entropy and Q denotes the quantization error between v and b. However, controlling the scale λ is difficult, with low λ potentially being ineffective and high λ leading to underfitting. To overcome this cumbersome practice, this study first offers a geometric explanation of quantization error:

$$min \| \mathbf{v} - \mathbf{b} \|^2 \qquad \mathrm{b} \in \{-1,1\}^K \tag{4}$$

where **v** is in continuous space and $\mathbf{b} = \mathrm{sgn}(\mathbf{v})$ is in binary space. Expanding equation (4) yields:

$$\| \mathbf{v} - \mathbf{b} \|^2 = \| \mathbf{v} \|^2 + \| \mathbf{b} \|^2 - 2 \| \mathbf{v} \| \| \mathbf{b} \| \cos\theta_{vb} \tag{5}$$

From equation (5), it can be seen that retrieval is based solely on the similarity of the two hash code directions. Therefore, we can ignore the magnitude of v, normalize it to have the same norm as b, i.e., $\| \mathrm{v} \| = \sqrt{\mathrm{K}}$, and interpret the quantization error solely as the angle θ_vb between v and b.

$$\| \mathbf{v} - \mathbf{b} \|^2 = 2K - 2K\cos\theta_{vb} = 2K(1 - \cos\theta_{vb}) \tag{6}$$

As $2K$ is a constant, we can conclude that maximizing the cosine similarity between v and b will lead to lower quantization error, thereby achieving better approximations in the hash codes.

### *4.2. Discriminative Hash Codes with Orthogonal Targets*

Using the random hyperplane technique, under a hash function family F, the probability that two samples $\mathbf{x}_i$ and $\mathbf{x}_j$ have the same hash code can be described as $\Pr_{h\in\mathcal{F}}[h(\mathbf{x}_i) = h(\mathbf{x}_j)] = 1 - \frac{\theta_{ij}}{\pi}$, where $h(.)$ is a hash function and $\theta_{ij}$ is the angle between $\mathbf{x}_i$ and $\mathbf{x}_j$. Therefore, based on the same principle, it can be deduced that if two continuous codes $\mathbf{v}_i$ and $\mathbf{v}_j$ from the hidden layer H have high cosine similarity, then the hash codes b_i and b_j should also have a high probability of having high cosine similarity.

DOI: 10.54254/2755-2721/43/20230850

Additionally, cosine similarity can be further optimized for retrieval performance by representing it as the quantization error between continuous codes and hash codes.

Considering these two cases, we propose to maximize the cosine similarity between the continuous code $\mathbf{v}_n$ and its corresponding binary orthogonal target, $\mathbf{o}_{y_n} \in [\mathbf{o}_1, \cdots, \mathbf{o}_C]^\top = \mathbf{O} \in \{-1, +1\}^{C \times K}$. This can be achieved by maximizing the posterior probability of the true class through the cross-entropy loss, as shown in equation (7):

$$L = -\frac{1}{N}\sum_{n=1}^{N} \log \frac{\exp(\mathbf{o}_{y_n}^{\top}\mathbf{v}_n)}{\sum_{i=1}^{C}\exp(\mathbf{o}_i^{\top}\mathbf{v}_n)} \tag{7}$$

where $\mathbf{v}_n$ represents the deep-layer continuous encoding of the nth sample of $\mathbf{o}_{y_n}$, and $\mathbf{o}_i \in \mathbf{O}$ represent the true class of the binary orthogonal target and the true value y_n, respectively. For simplicity, by omitting the bias term in equation (7), under the deep hypersphere embedding framework, the transformation $\mathbf{o}_i^{\top}\mathbf{v}_n = \| \mathbf{o}_i \| \| \mathbf{v}_n \| \cos \theta_{ni}$ is obtained, where $\theta_{ni}$ is the angle between the continuous code v_n and the binary orthogonal target $\mathbf{o}_i$. Next, $\mathbf{v}_n$ is L2 normalized to have $\| \mathbf{v}_n \| = 1$, $\| \mathbf{o}_i \| = \sqrt{K}$, since it is in binary form. Consequently, the loss function can be rewritten as:

$$L = -\frac{1}{N}\sum_{n=1}^{N} \log \frac{\exp(\sqrt{K}\cos\ \theta_{y_n})}{\exp(\sqrt{K}\cos\ \theta_{y_n}) + \sum_{i=1, i\neq y_n}^{C}\exp(\sqrt{K}\cos\ \theta_{ni})} \tag{8}$$

Thus, the method proposed in this study does not introduce quantization error minimization in the learning objective but unifies the learning objective and quantization error minimization under a cosine similarity minimization objective. Moreover, as the binary orthogonal target achieves the maximum inter-class Hamming distance and our loss function also aims to minimize intra-class error by utilizing the cosine similarity, further minimizing the within-class variance. The method proposed in this study can perform end-to-end training, learning highly discriminative hash codes without the need for complex training objectives and computational graph modifications.

## 5. Experiments

### *5.1. Dataset Settings*

To evaluate the retrieval performance of our method, we used four widely applicable datasets for image retrieval. The CIFAR10 dataset consists of 60,000 images across 10 categories. The ImageNet dataset is a subset of the Large-Scale Visual Recognition Challenge (ILSVRC 2015). We utilized the standard retrieval protocol [15] on the ImageNet dataset and evaluated images from 100 of the most common categories, with query, training, and retrieval sets containing 5,000, 10,000, and 128,495 images, respectively. NUS WIDE is a multi-label image dataset, from which we selected images from 21 of the most common categories for evaluation, with the query, training, and retrieval sets containing 2,040, 10,000, and 149,685 images, respectively. The MS-COCO dataset consists of 80 categories, with the query, training, and retrieval sets containing 5,000, 10,000, and 117,218 images, respectively.

### *5.2. Network and Training Settings*

The model batch size was set to 32, with 50 epochs using the Adam optimizer and a learning rate of 0.001. We applied the cosine annealing algorithm for learning rate optimization. The experiments utilized a pre-trained Visual Transformer (ViT) model, with the main network weights pre-trained on ImageNet. The hash codes generated by the hashing algorithm had lengths of 16, 32, and 64 bits. Testing was performed every 10 epochs, with the best results reported. The Mean Average Precision (MAP) across all categories was employed as the evaluation metric.

### *5.3. Comparative Experiments*

To validate the effectiveness of the proposed method, we compared it with various hashing learning methods, including shallow model-based hashing methods and deep learning-based hashing methods. The shallow methods were ITQ-CCA, BRE, KSH, and SDH. The deep methods included CNNH,

DOI: 10.54254/2755-2721/43/20230850

DNNH, DHN, HashNet, DCH, GreedyHash, CSQ, DPN, and OrthoCos. According to Table 1, our method exhibited superior MAP performance results for 16, 32, and 64-bit hash codes across the four mainstream datasets. This indicates that our method demonstrates better performance and effectiveness in image retrieval tasks.

**Table 1.** Comparison of MAP for Different Bit Hamming Rankings in Image Retrieval

| *Method* | *MSCOCO* | | | *ImageNet100* | | | *CIFAR10* | | | *NUS WIDE* | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 16 | 32 | 64 | 16 | 32 | 64 | 16 | 32 | 64 | 16 | 32 | 64 |
| *ITQ-CCA* | 0.56 | 0.56 | 0.50 | 0.26 | 0.43 | 0.57 | - | - | - | 0.43 | 0.43 | 0.43 |
| *BRE* | 0.59 | 0.62 | 0.63 | 0.06 | 0.25 | 0.35 | - | - | - | 0.48 | 0.52 | 0.54 |
| *KSH* | 0.52 | 0.53 | 0.53 | 0.16 | 0.29 | 0.39 | - | - | - | 0.39 | 0.40 | 0.39 |
| *SDH* | 0.55 | 0.56 | 0.58 | 0.29 | 0.45 | 0.58 | - | - | - | 0.57 | 0.59 | 0.61 |
| *CNNH* | 0.55 | 0.56 | 0.58 | 0.31 | 0.47 | 0.59 | | - | - | 0.65 | 0.65 | 0.64 |
| *DNNH* | 0.64 | 0.65 | 0.64 | 0.35 | 0.52 | 0.61 | - | - | - | 0.70 | 0.73 | 0.75 |
| *DHN* | 0.71 | 0.73 | 0.74 | 0.36 | 0.52 | 0.62 | - | - | - | 0.71 | 0.75 | 0.77 |
| *DCH* | 0.75 | 0.80 | 0.82 | 0.65 | 0.73 | 0.75 | - | 0.66 | 0.67 | 0.77 | 0.79 | 0.81 |
| *HashNet* | 0.74 | 0.77 | 0.78 | 0.62 | 0.70 | 0.73 | 0.64 | 0.67 | 0.68 | 0.66 | 0.69 | 0.71 |
| *GreedyHash* | 0.67 | 0.72 | 0.74 | 0.62 | 0.66 | 0.68 | 0.78 | 0.81 | 0.81 | 0.77 | 0.79 | 0.81 |
| *CSQ* | 0.79 | 0.83 | 0.86 | 0.85 | 0.86 | 0.87 | 0.84 | 0.83 | 0.85 | 0.81 | 0.82 | 0.83 |
| *DPN* | 0.71 | 0.80 | 0.85 | 0.61 | 0.69 | 0.73 | 0.77 | 0.80 | 0.81 | 0.84 | 0.85 | 0.82 |
| *OrthoCos* | 0.70 | 0.78 | 0.79 | 0.61 | 0.67 | 0.71 | 0.85 | 0.87 | 0.89 | 0.80 | 0.83 | 0.85 |
| *Ours* | 0.80 | 0.84 | 0.88 | 0.90 | 0.91 | 0.92 | 0.95 | 0.96 | 0.97 | 0.78 | 0.84 | 0.85 |

### *5.4. Ablation Experiments*

o verify the effectiveness of the feature extraction module and the loss module in our approach, we replaced the feature extraction module with AlexNet, ResNet50, and VGG modules in the model. We conducted experiments on the ImageNet100 dataset (using 64-bit encoding) and the CIFAR10 dataset (using 64-bit encoding), using MAP as the performance metric, as shown in Table 2. To evaluate the efficacy of the loss function, while keeping other model modules unchanged, we replaced the loss function module with Softmax and CrossEntropy modules, respectively, instead of the loss function module designed in this study. We conducted experiments and compared the results on the ImageNet100 dataset (using 64-bit encoding) and the CIFAR10 dataset (using 64-bit encoding) using MAP as the performance metric, as shown in Table 2.

**Table 2.** Comparison of MAP with Replaced Feature Extraction Module

| *Dataset* | *Feature Extraction Module* | *MAP* | *Loss Function* | *MAP* |
|---|---|---|---|---|
| *ImageNet100* | AlexNet | 0.61 | Softmax Loss | 0.87 |
| | ResNet50 | 0.88 | Cross Entropy Loss | 0.89 |
| | VGG | 0.85 | Ours | 0.92 |
| | Ours | 0.92 | / | / |
| *CIFAR10* | AlexNet | 0.78 | Softmax Loss | 0.95 |
| | ResNet50 | 0.90 | Cross Entropy Loss | 0.93 |
| | VGG | 0.84 | Ours | 0.97 |
| | Ours | 0.97 | / | / |

### *5.5. Visualization*

To visually demonstrate the effectiveness of our model, we utilized the confusion matrix method to visualize the experimental results of the model on the CIFAR-10 dataset (with encodings of 16, 32, and 64 bits). In the confusion matrix, shades of blue represent the accuracy of identification, with the color depth directly proportional to the model's recognition accuracy. The horizontal direction represents the

DOI: 10.54254/2755-2721/43/20230850

predicted labels of the samples, while the vertical direction represents the true labels of the samples. The results are depicted in Figure 3, and a comprehensive analysis of the model's output prediction accuracy indicates a satisfactory performance.

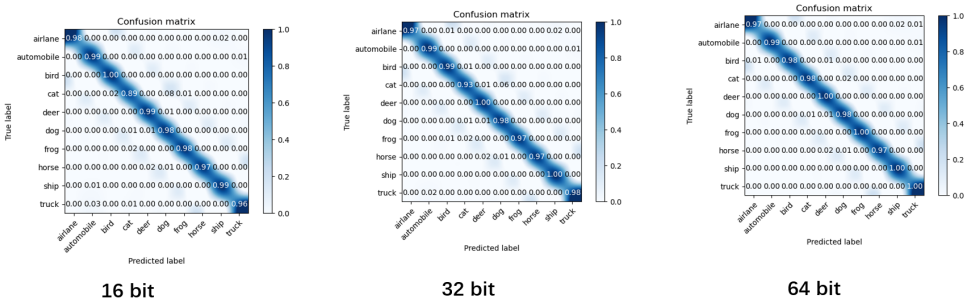


**Figure 1.** Confusion matrix for results on CIFAR-10

## 6. Conclusion and Future Work

We proposed a unified training objective for deep hashing under a single classification target. It was demonstrated that this could be achieved by maximizing the cosine similarity between continuous codes and binary orthogonal targets under cross-entropy loss. To this end, we first redefined the deep hashing problem through the lens of cosine similarity and then demonstrated that end-to-end training of deep hashing is feasible without any additional complex constraints if we perform L2 normalization on the continuous codes. As part of future work, we are exploring how to use hash codes through unsupervised learning to improve retrieval performance by learning better feature representations.

## References

[1] Bengio Y, Nicholas Léonard, Courville A C. Estimating or Propagating Gradients Through Stochastic Neurons for Conditional Computation: arXiv 2013.

[2] Hanjiang Lai, Yan Pan, Ye Liu, and Shuicheng Yan. Simultaneous feature learning and hash coding withdeep neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition,pages 3270–3278, 2015.

[3] Yair Weiss, Antonio Torralba, and Rob Fergus. Spectral hashing. In D. Koller, D. Schuurmans, Y. Bengio,and L. Bottou, editors, Advances in Neural Information Processing Systems, volume 21. Curran Associates,Inc., 2009.

[4] Yunchao Gong, Svetlana Lazebnik, Albert Gordo, and Florent Perronnin. Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval. IEEE transactions on pattern analysis and machine intelligence, 35(12):2916–2929, 2012.

[5] Piotr Indyk and Rajeev Motwani. Approximate nearest neighbors: Towards removing the curse of dimensionality. In Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing, STOC ’98, page 604–613, New Y ork, NY, USA, 1998. Association for Computing Machinery.

[6] Brian Kulis and Kristen Grauman. Kernelized locality-sensitive hashing for scalable image search. In 2009 IEEE 12th international conference on computer vision, pages 2130–2137. IEEE, 2009.

[7] BL Lu, L Zhang, J Kwok. Proceedings of the 18th international conference on Neural Information Processing - Volume Part II[C]// International Conference on Neural Information Processing. Springer-Verlag, 2011.

[8] Cao Z, Long M , Wang J , et al. HashNet: Deep Learning to Hash by Continuation[J]. IEEE Computer Society, 2017.

[9] Su S, Tian Y . Greedy Hash: Towards Fast Optimization for Accurate Hash Coding in CNN[C]// Neural Information Processing Systems. 2018.

DOI: 10.54254/2755-2721/43/20230850

[10] Zheng X, Zhang Y, Lu X. Deep Balanced Discrete Hashing for Image Retrieval[J]. Neurocomputing, 2020, 403(3).
[11] Dubey S R, Singh S K, Chu W T. Vision Transformer Hashing for Image Retrieval[J]. arXiv e-prints, 2021.
[12] Dosovitskiy A, Beyer L, Kolesnikov A, et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale[C]// International Conference on Learning Representations. 2021.
[13] Chen X, Yan B, Zhu J, et al. High-Performance Transformer Tracking[J]. 2022.
[14] Zhang T, Zhu L, Zhao Q, et al. Neural Networks Weights Quantization: Target None-retraining Ternary (TNT)[J]. 2019.
[15] Kulis B, Darrell T. Learning to Hash with Binary Reconstructive Embeddings[C]// International Conference on Neural Information Processing Systems. Curran Associates Inc. 2009.