CIMON: Towards High-quality Hash Codes

Xiao Luo[1,2,*] , **Daqing Wu**[1,2,*] , **Zeyu Ma**[3] , **Chong Chen**[1,2,†] , **Minghua Deng**[1,†] ,
Jinwen Ma[1] , **Zhongming Jin**[2] , **Jianqiang Huang**[2] and **Xian-Sheng Hua**[2]
[1]School of Mathematical Sciences, Peking University, China
[2]DAMO Academy, Alibaba Group, Hangzhou, China
[3]School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China
{xiaoluo,wudq}@pku.edu.cn, zeyu.ma@stu.hit.edu.cn, {dengmh,jwma}@math.pku.edu.cn,
{cheung.cc,zhongming.jinzm, jianqiang.hjq,xiansheng.hxs}@alibaba-inc.com

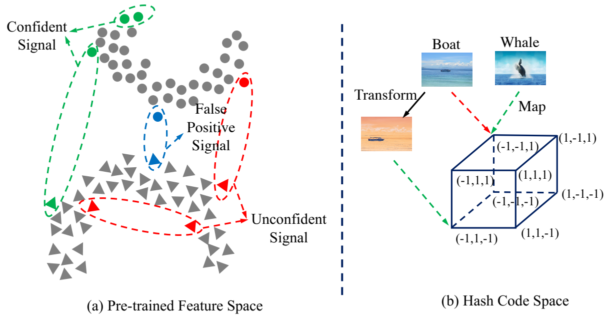
Abstract

Recently, hashing is widely used in approximate nearest neighbor search for its storage and computational efficiency. Most of the unsupervised hashing methods learn to map images into semantic similarity-preserving hash codes by constructing local semantic similarity structure from the pre-trained model as the guiding information, i.e., treating each point pair similar if their distance is small in feature space. However, due to the inefficient representation ability of the pre-trained model, many false positives and negatives in local semantic similarity will be introduced and lead to error propagation during the hash code learning. Moreover, few of the methods consider the robustness of models, which will cause instability of hash codes to disturbance. In this paper, we propose a new method named **C**omprehensive s**I**milarity **M**ining and c**O**nsistency lear**N**ing (CIMON). First, we use global refinement and similarity statistical distribution to obtain reliable and smooth guidance. Second, both semantic and contrastive consistency learning are introduced to derive both disturb-invariant and discriminative hash codes. Extensive experiments on several benchmark datasets show that the proposed method outperforms a wide range of state-of-the-art methods in both retrieval performance and robustness.

1 Introduction

Hashing-based Approximate Nearest Neighbour search has attracted ever-increasing attention in the era of big data due to their high retrieval efficiency and low storage cost. The main idea of hashing methods is to project high dimensional datapoints into compact binary codes while preserving the semantic similarity of original datapoints.

Hashing methods include supervised hashing [Luo *et al.*, 2020; Fan *et al.*, 2020; Wang *et al.*, 2020; Xie *et al.*, 2020] and unsupervised hashing. However, it is difficult to apply supervised hashing methods in practice since large-scale data annotations are unaffordable. To address this problem, several deep unsupervised methods were proposed and provided a cost-effective solution to practical applications [Lin *et al.*, 2016; Yang *et al.*, 2018; Tu *et al.*, 2020]. Recently, most unsupervised hashing methods employ a two-step framework: Firstly, the local semantic similarity structure is reconstructed from the pre-trained neural network. To be specific, the local semantic similarity relationships are often derived from the Euclidean distance or the cosine similarity of deep features extracted from the pre-trained model. Secondly, a hashing network is optimized to generate compact and similarity-preserving hash codes by incorporating the defined similarity structure as the guiding information.



(a) Pre-trained Feature Space

(b) Hash Code Space

Figure 1: Motivation of our model. (a) The "triangle" points and the "circle" points are belong to different categories. False signals (blue) and unconfidence signals (red) will misguide the hash code learning. (b) Two different images (the first line) are mapped to the same hash code (collision) and the hash code is sensitive to the transformation, which implies the hash code is of low quality.

However, the existing methods have two significant drawbacks that will harm the quality of hash codes. First, many false positives and negatives will be introduced in the similarity matrix for the insufficient representation ability of the pre-trained model, which will misguide the hashing model during hash code learning and further reduce the retrieval performance. As shown in Figure 1(a), false similar pairs can occur between the boundary points of two manifolds (blue points). Moreover, most methods treat the confident signals and uncertain signals equally (green and red points), which will also accumulate a lot of errors. Second, few of them consider the stability of models, which will cause hash codes

[*] Equal Contribution. This work was done when Xiao Luo and Daqing Wu interned in Alibaba Group. [†] Corresponding authors.

to be unstable under disturbances and greatly influence the quality. For example, images of different classes with similar background could be mapped to the same hash code (called hash collisions) while the transformed image could be quite far away from the original image in the hash code space (Figure 1(b)).

To address the above two issues, we propose a new method named CIMON, which comprehensively explores semantic similarity structure to achieve reliable semantic guidance and considers the stability of hash codes by introducing consistency learning. Specifically, CIMON firstly takes advantage of the global information to remove false positives between boundary points and smooths the unconfident signals by confidence adjustment. Secondly, CIMON generates two groups of deep features by data augmentation and constructs two similarity matrices and both parallel semantic consistency and cross semantic consistency are encouraged to generate robust hash codes. Furthermore, contrastive consistency between hash codes is adopted to improve the quality of hash codes. Through these improvements, CIMON could obtain high-quality hash codes in both retrieval performance and robustness, which is also demonstrated by extensive experiments on several challenging benchmark datasets. Our main contributions can be summarized as following:

- CIMON not only utilizes global refinement to refine the initial local semantic similarity structure, but also explores the similarity statistical distribution to adjust the weight for each image pair, which generates a reliable and smooth guide for hash code learning.
- A novel consistency loss including semantic consistency and contrastive consistency is proposed to optimize the hashing network, which helps to generate robust and discriminative hash codes.
- Experiments on several public datasets demonstrate that CIMON outperforms existing state-of-the-art unsupervised hashing techniques by a large margin.

2 Related Work

Deep Unsupervised Hashing. Most of the unsupervised deep hashing methods extract deep features to construct a semantic structure, by which unsupervised problems can be turned into supervised problems. In a quite different way, DeepBit [Lin *et al.*, 2016] regards the original images and the corresponding rotated images as similar pairs and attempts to preserve the similarities when learning related hash codes. Stochastic generative hashing [Dai *et al.*, 2017] tries to learn hash functions by using a generative model based on the minimum description length principle. SSDH [Yang *et al.*, 2018] makes use of a specific truncated function on the pairwise distances and constructs the similarity structures. The hashing model is then trained by supervised hashing techniques. Afterwards, the performance of SSDH is improved by DistillHash, which distills the image pairs with confident similarity signals. Clustering-driven Unsupervised Deep Hashing [Gu *et al.*, 2019] recursively learns discriminative clusters by soft clustering model and produces binary code with high similarity responds. MLS^{3}RDUH [Tu *et al.*, 2020] reconstructs the local semantic similarity structure by taking advantage of the manifold structure in feature space, achieving the state-of-the-art performance.

Contrastive Learning. [Hadsell *et al.*, 2006] is the first work to learn representations by contrasting positive pairs against negative pairs. To solve the storage of large scale dataset, [Wu *et al.*, 2018] proposes to utilize a memory bank for class representation vectors. Various pretext work is based on several forms of contrastive loss function, which is related to the exemplar-based task and noise-contrastive estimation [Dosovitskiy *et al.*, 2014]. Recently, Momentum Contrast [He *et al.*, 2020] proposes to build a dynamic dictionary with a queue and a moving-averaged encoder, which enables building a large and consistent dictionary on-the-fly that facilitates contrastive unsupervised learning. SimCLR [Chen *et al.*, 2020] further simplifies the learning algorithms without requiring specialized architectures or a memory bank and achieves better performance on ImageNet.

3 The Proposed Model

In this section, we first formally define the problem and feature our model with two parts as shown in Figure 2: (1) **Semantic information generating**. A pre-trained VGG-F [Simonyan and Zisserman, 2015] without the last fully-connected layer $F(\cdot)$ is adopted to extract deep features, which will be used to generate the similarity graph and the confidence-based weight matrix. (2) **Consistency learning**. The hashing network $G(\cdot)$ is modified from VGG-F by replacing the last fully-connected layer with a fully-connected layer with L hidden units to incorporate the hash code learning process. A novel consistency learning framework is adopted to learn high-quality hash codes.

3.1 Problem Formulation

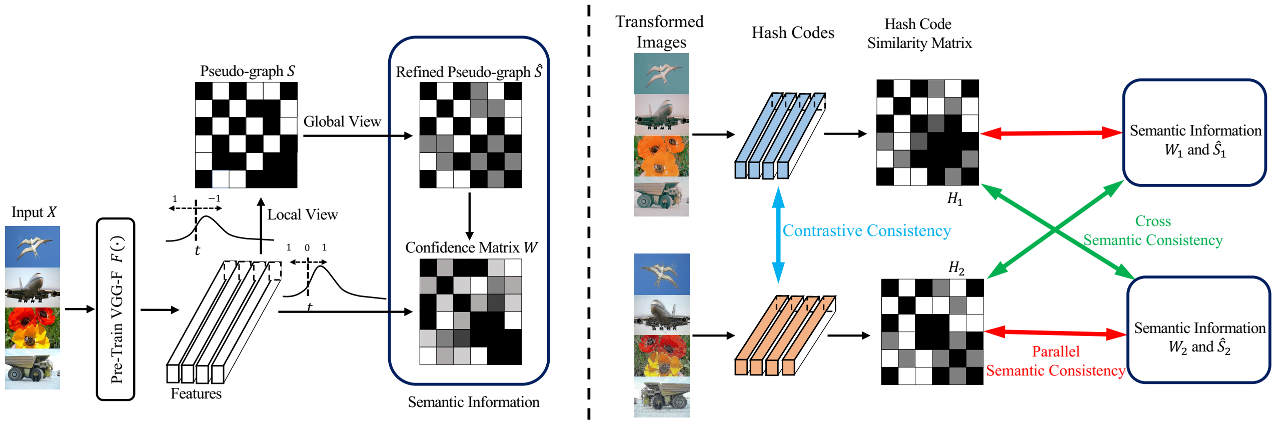
In the unsupervised hashing settings, $\mathcal{X} = \{x_i\}_{i=1}^N$ is the training set with N samples without label information. We aim to learn a hash function $\mathcal{H} : x \rightarrow b \in \{-1, 1\}^L$, in which x is the input image and b is a compact L-bit hash code. This map should preserve similarity, i.e., images with similar ground truth labels should correspond to hash codes with small Hamming distances.

3.2 Semantic Information Generating

In our model, semantic information is composed of the similarity pseudo-graph and the similarity confidence matrix.

From the local perspective, the pseudo-graph aims to capture pairwise similarity information. Given the pre-trained deep features $\{F(x_i)\}_{i=1}^N$, the cosine distance between the i-th and the j-th images is obtained by $d_{ij} = 1 - \frac{F(x_i) \cdot F(x_j)}{||F(x_i)||_2 ||F(x_j)||_2}$. Since most pairs should be negative in ground-truth, we set a relatively small threshold $t = 0.1$ following [Wu *et al.*, 2019], and consider pairs with the cosine distance smaller than t as potential positives ($S_{ij} = 1$) and pairs with the cosine distance larger than t as potential negatives ($S_{ij} = -1$). Mathematically, the pseudo-graph S can be constructed as:

$$S_{ij} = \begin{cases} 1 & d_{ij} \leq t, \\ -1 & d_{ij} > t \end{cases} \tag{1}$$


Input X
Pre-Train VGG-F $F(\cdot)$
Features
Pseudo-graph S
Global View
Local View
1
−1
t
1
0
1
t
Refined Pseudo-graph $\hat{S}$
Confidence Matrix W
Semantic Information
Transformed Images
Hash Codes
Hash Code Similarity Matrix
H_1
H_2
Contrastive Consistency
Semantic Information W_1 and $\hat{S}_1$
Semantic Information W_2 and $\hat{S}_2$
Cross Semantic Consistency
Parallel Semantic Consistency

(a) Generation of Semantic Information

(b) Consistency Learning

Figure 2: Overview of CIMON. (a) CIMON generates semantic information including refined pseudo-graph and confidence matrix for reliable guidance. (b) With data augmentation, CIMON generates two groups of semantic information. The parallel and cross semantic consistency is constructed between features and hash codes under the same group and crossing the different groups respectively. The contrastive consistency is based on two groups of hash codes.

Global Refinement As mentioned in Figure 1, the features of images with the same semantic information should lay on a high-dimensional manifold following [Yang *et al.*, 2019; Tu *et al.*, 2020] and many false positives and negatives will be introduced in local semantic similarity S [Luo *et al.*, 2021]. Hence, we propose to use the global clustering results to refine the semantic similarity by removing contradictory results. Since spectral clustering has been proven to be suitable for clustering high-dimensional manifold data, we take advantage of it to perform global refinement. Specifically, assume $c_i \in \{1, ..., K\}, i = 1, \cdots, N$ is the i-th cluster label of spectral clustering (K is the number of clusters). Then each pair of two points with the same class is considered as global potential similar and vice versa. If an image pair has a different potential similarity signal with pseudo-graph S, its similarity is considered as unreliable. After removing the unreliable signals, the final refined pseudo-graph is $\hat{S}$ is formulated as:

$$\hat{S}_{ij} = \begin{cases} 1 & c_i = c_j \ \&\ S_{ij} = 1 \\ -1 & c_i \neq c_j \ \&\ S_{ij} = -1 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

Confidence Adjustment Note that the semantic confidence of similarity signal for each pair is different, we further construct the confidence matrix for pseudo-graph S based on the semantic confidence. Inspired by recent works [Yang *et al.*, 2018], we observe that the distribution of cosine distances for deep feature pairs can be estimated by two half Gaussian distributions, denoted as $\mathbf{N}(m_1, \sigma_1^2)$ and $\mathbf{N}(m_2, \sigma_2^2)$ respectively, in which m_1, m_2 and σ_1^2 and σ_2^2 are the corresponding means and variances. Intuitively, we hypothesize that image pairs with distances obviously smaller than others are semantically 'similar' and those with obviously larger distances are semantically 'dissimilar', which are denoted as highly confident pairs. Moreover, the confidence weights are approximated by Cumulative Distribution Function (CDF) of two half Gaussian distributions. Specifically, for potential similar signals, distances have a more confident similarity signal if d_{ij} is closer to 0, and for potential dissimilar signals, distances have a more confident similarity signal if d_{ij} is closer to 2. Therefore, confidence-based weight matrix is computed as following:

$$W_{ij} = \begin{cases} \frac{\Phi_1(t) - \Phi_1(d_{ij})}{\Phi_1(t) - \Phi_1(0)} & d_{ij} \leq t \ \&\ \hat{S}_{ij} \neq 0, \\ \frac{\Phi_2(d_{ij}) - \Phi_2(t)}{\Phi_2(2) - \Phi_2(t)} & t < d_{ij} \ \&\ \hat{S}_{ij} \neq 0, \\ 0 & \hat{S}_{ij} = 0 \end{cases} \tag{3}$$

in which $\Phi_k(x) = \Phi(\frac{x - m_k}{\sigma_k})$ and $\Phi(\cdot)$ is the CDF of standard normal distribution. Under this setting, all $W_{ij} \in [0, 1]$ and the confidence weights on both ends are relatively larger.

3.3 Consistency Learning

In order to preserve the similarity structure of input images, similar (dissimilar) images are expected to be mapped into similar (dissimilar) hash codes. Different from previous models, here we adopt two groups of semantic information under two different kinds of data augmentation.

Semantic Consistency For each image x_i, there are two transformed samples $x_i^{(1)}$ and $x_i^{(2)}$. At the semantic information generating stage, two refined similarity graphs with confidence matrices $\{W^{(1)}, \hat{S}^{(1)}\}$, $\{W^{(2)}, \hat{S}^{(2)}\}$ are generated with extracted features $\{F(x_i^{(1)})\}_{i=1}^N$ and $\{F(x_i^{(2)})\}_{i=1}^N$ as the guiding information. Simultaneously, images $x_i^{(1)}$ and $x_i^{(2)}$ are the inputs of the hashing network $G(\cdot)$, and hash codes $b_i^{(1)}$ and $b_i^{(2)}$ are obtained through activation function $sign(\cdot)$. Therefore, we derive two similarity outputs $H^{(1)}$ and $H^{(2)}$ from hash codes, which is formulated as

$$H_{ij}^{(m)} = \frac{1}{L} b_i^{(m)\top} b_j^{(m)}, \ \ b_i^{(m)} = sign(G(x_i^{(m)}; \Theta)) \tag{4}$$

in which $m = 1$ or 2, and Θ represents the set of parameters of hashing network. For the purpose of preserving the semantic structures, we first minimize weighted L_2 loss between the hash code similarity and the corresponding pseudo-graph from the same group. The parallel semantic consistency loss can be formulates as:

$$\mathcal{L}_{PSC} = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} W_{ij}^{(1)}(H_{ij}^{(1)} - \hat{S}_{ij}^{(1)})^2 + W_{ij}^{(2)}(H_{ij}^{(2)} - \hat{S}_{ij}^{(2)})^2 \tag{5}$$

Inspired by the cross-attention mechanism [Boussaha *et al.*, 2019], we also match the hash code similarity with the pseudo-graph from the different groups. To be specific, the cross semantic consistency loss can be written as:

$$\mathcal{L}_{CSC} = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} W_{ij}^{(1)}(H_{ij}^{(2)} - \hat{S}_{ij}^{(1)})^2 + W_{ij}^{(2)}(H_{ij}^{(1)} - \hat{S}_{ij}^{(2)})^2 \tag{6}$$

Contrastive Consistency Self-supervised learning has been proved to generate high-quality representations for downstream tasks [He *et al.*, 2020]. From this points, we randomly sample a minibatch of M images, producing $2M$ random transformed images $\{x_i^{(1)}, x_i^{(2)}\}_{i=1}^{M}$. Given a positive pair $x_i^{(1)}$ and $x_i^{(2)}$, we treat the other $2(M-1)$ augmented images within a minibatch as negative examples. Denote $b_i \star b_j$ as cosine similarity of b_i and b_j ($\frac{b_i \cdot b_j}{||b_i||_2||b_j||_2}$), the contrastive consistency loss is defined as

$$\ell_{CC} = -\frac{1}{2M}\sum_{i=1}^{M}\Big(\log\frac{e^{b_i^{(1)}\star b_i^{(2)}/\tau}}{Z_i^{(1)}} + \log\frac{e^{b_i^{(1)}\star b_i^{(2)}/\tau}}{Z_i^{(2)}}\Big) \tag{7}$$

where $Z_i^{(m)} = \sum_{j\neq i}(e^{b_i^{(m)}\star b_j^{(1)}/\tau} + e^{b_i^{(m)}\star b_j^{(2)}/\tau})$, $m = 1$ or 2, and τ denotes a temperature parameter set to 0.5 following [Chen *et al.*, 2020]. Note that the numerator of each term punishes the distance between hash codes of samples under different transformation while the denominator encourages to enlarge the distance between hash codes of different samples, which encourages the hash codes to be uniformly distributed in the hash code space from [Wang and Isola, 2020]. This point helps to maximize the capacity of each hash bit [Shen *et al.*, 2018], preserving as much information of the data as possible.

Finally, the loss of consistency learning is formulated as

$$\mathcal{L} = \mathcal{L}_{PSC} + \mathcal{L}_{CSC} + \eta\mathcal{L}_{CC} \tag{8}$$

in which η is a balance coefficient to balance different consistency loss. However, the $sign(\cdot)$ is in-differentiable at zero and the derivation of it will be zero for every non-zero input, with the result that the parameters of the hashing model will not be updated by the back-propagation algorithm when minimizing the Equation 8. Thus, we use $tanh(\cdot)$ to approximate the sign function and generate the approximate hash code $v_i^{(m)} = tanh(G(x_i^{(m)}))$ to replace $b_i^{(m)}$ in loss function. Our loss function is optimized by the mini-batch standard stochastic gradient descent (SGD) method. The whole learning procedure are shown in Algorithm 1.

Algorithm 1 CIMON's Training Algorithm

```
Require: Training images $\mathcal{X} = \{x_i\}_{i=1}^{N}$; Code length $L$;
Ensure: Parameters $\Theta$ for the neural network $G(\cdot)$;
        Hash codes $B = \{b_i\}_{i=1}^{N}$ for training images.
Generate two transformed images via data augmentation for each image: $\mathcal{X}^{(1)}$ and $\mathcal{X}^{(2)}$;
for $m = 1, 2$ do
    Get pre-train features of $\mathcal{X}^{(m)}$ through $F(\cdot)$;
    Construct the pseudo-graph $S^{(m)}$ by Equation 1;
    Perform global refinement to obtain refined pseudo-graph $\hat{S}^{(m)}$ by Equation 2;
    Construct the confidence matrix $W^{(m)}$ by Equation 3;
end for
repeat
    Sample $M$ images from $\mathcal{X}$ and obtain their augmentation to construct a mini-batch;
    Calculate loss function by Equation 8;
    Update parameters of $G(\cdot)$ through back propagation;
until convergence
Generate image hash codes B
```

4 Experiments

We conduct extensive experiments on three popular benchmark datasets to evaluate our CIMON by comparisons with various unsupervised hashing methods.

4.1 Datasets and Setup

FLICKR25K [Huiskes and Lew, 2008] contains 25,000 images labeled by some of the 24 categories. We randomly sample 2,000 images as the query set and use the other images as the retrieval set. 10000 images are randomly selected from the retrieval set as the training set. **CIFAR-10** [Krizhevsky *et al.*, 2009] contains 60000 images of ten different categories. We randomly sample 1,000 images as the query set for each class, and take the rest as the retrieval set. We sample 500 images per class in the retrieval set as the training set. **NUSWIDE** [Chua *et al.*, 2009] contains 269,648 images of 81 concepts. Here, we use the subset that contains the 10 most popular concepts. We randomly select 5,000 images as the query set and the remaining images make up the retrieval set. 5000 images randomly sampled from the retrieval set serve as the training set.

Our CIMON is compared with various state-of-the-art unsupervised hashing methods including both traditional methods and deep learning methods. Traditional methods include ITQ [Gong *et al.*, 2012], DSH [Jin *et al.*, 2013], SpH [Heo *et al.*, 2012] and SGH [Dai *et al.*, 2017]. Deep unsupervised hashing methods include DeepBits [Lin *et al.*, 2016], SSDH [Yang *et al.*, 2018], DistillHash [Yang *et al.*, 2019], CUDH [Gu *et al.*, 2019], and MLS^{3}RUDH [Tu *et al.*, 2020]. For deep learning-based methods, we use raw pixels as inputs. For traditional methods, we extract 4096-dimensional feature vectors by the VGG-F model which is pre-trained on ImageNet for fair comparison.

As for evaluation, the ground-truth similarity information is obtained according to the ground-truth image labels. Specifically, two data points are considered similar if they

Methods	FLICKR25K				CIFAR-10				NUS-WIDE			
	16bits	32bits	64bits	128bits	16bits	32bits	64bits	128bits	16bits	32bits	64bits	128bits
ITQ	0.6492	0.6518	0.6546	0.6577	0.1942	0.2086	0.2151	0.2188	0.5270	0.5241	0.5334	0.5398
DSH	0.6452	0.6547	0.6551	0.6557	0.1616	0.1876	0.1918	0.2055	0.5123	0.5118	0.5110	0.5267
SpH	0.6119	0.6315	0.6381	0.6451	0.1439	0.1665	0.1783	0.1840	0.4458	0.4537	0.4926	0.5000
SGH	0.6362	0.6283	0.6253	0.6206	0.1795	0.1827	0.1889	0.1904	0.4994	0.4869	0.4851	0.4945
DeepBit	0.5934	0.5933	0.6199	0.6349	0.2204	0.2410	0.2521	0.2530	0.3844	0.4341	0.4461	0.4917
SSDH	0.7240	0.7276	0.7377	0.7343	0.2568	0.2560	0.2587	0.2601	0.6374	0.6768	0.6829	0.6831
DistillHash	-	-	-	-	0.2844	0.2853	0.2867	0.2895	-	-	-	-
CUDH	0.7332	0.7426	0.7549	0.7561	0.2856	0.2903	0.3025	0.3000	0.6996	0.7222	0.7451	0.7418
MLS^{3}RDUH	0.7587	0.7754	0.7870	0.7927	0.2876	0.2962	0.3139	0.3117	0.7056	0.7384	0.7629	0.7818
CIMON	**0.8049**	**0.8195**	**0.8281**	**0.8321**	**0.4506**	**0.4723**	**0.4944**	**0.4981**	**0.7883**	**0.8060**	**0.8214**	**0.8243**

Table 1: MAP for different methods on FLICKR25K, CIFAR-10 and NUS-WIDE.

share the same label or share at least one common label. The retrieval quality is evaluated by Mean Average Precision (MAP), Precision-Recall curve and Top N precision curve. MAP is widely-used to evaluate the retrieval accuracy. Given a query and a list of R ranked retrieval results, the average precision (AP) for the given query can be obtained. MAP is defined to be the average of APs for all queries. For experiments on datasets FLICKR25K and NUSWIDE, we set R as 5000. For CIFAR-10, we set R as 50000. Precision-recall curve can reveal the precision at different recall levels, which is a good indicator of overall performance. As a popular way of the performance visualization, Top N precision curve is the precision curve in regard to the top N retrieved instances.

In our implementation, we optimize our model by mini-batch SGD with momentum. The mini-batch size is set to 24. The learning rate is fixed at 0.001. For all three datasets, training images are resized to 224 × 224 as inputs. Data augmentation we adopt includes random cropping and resizing, rotation, cutout, color distortion and Gaussian blur. As two introduced hyper-parameters, η and the number of clusters K in spectral clustering are set to 0.3 and 70 as default.

4.2 Experimental Results

The MAPs of different methods on datasets FLICKER25K, CIFAR-10 and NUSWIDE with hash code lengths varying from 16 to 128 are shown in Table 1. The following observations can be derived: 1) Deep learning-based algorithms overall perform better than the traditional methods, which shows that the strong representation-learning ability of deep learning helps to improve the performance of unsupervised hashing methods. 2) The methods that reconstruct semantic similarity structure with global information (CUDH, MLS^{3}RUDH) perform better than other deep unsupervised hashing methods, which indicates that semantic similarity reconstructed only by local information (i.e. pairwise distance of features) is inaccurate and unreliable. 3) We can find that CIMON has a significant improvement over the previous the-state-of-art MLS^{3}RUDH in all cases by a large margin. Specifically, the improvements of our model over the best baseline are 5.51%, 58.4% and 8.39% for average MAP on datasets FLICKER25K, CIFAR-10 and NUS-WIDE respectively, which shows the superiority of our model. We plot the precision-recall curves of SSDH, CUDH, MLS^{3}RDUH and CIMON on three datasets in the first column of Figure 3. We find that the curve of CIMON is always on top of the other three models' curves, which demonstrates that the hash codes obtained by CIMON are also more appropriate for hash table lookup search strategy. The second column of Figure 3 shows that the Top-N precision curves of these four models on the same datasets. Our CIMON significantly outperforms the comparison methods by large margins. Since the precision curves are based on the ranks of Hamming distance, CIMON achieves superior performance under Hamming ranking-based evaluations.

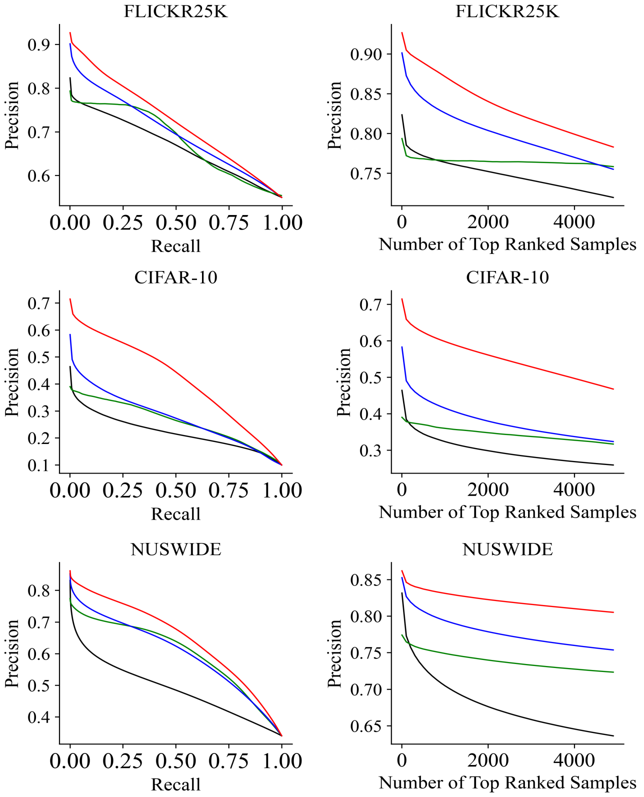


Figure 3: The first column plots the precision-recall curves, and the second column plots the Top-N precision curves. ($L = 128$, — SSDH, — CUDH, — MLS^{3}RDUH, — CIMON)

To demonstrate the robustness of CIMON, we add perturbation or transformation noise to the query set, which doesn't

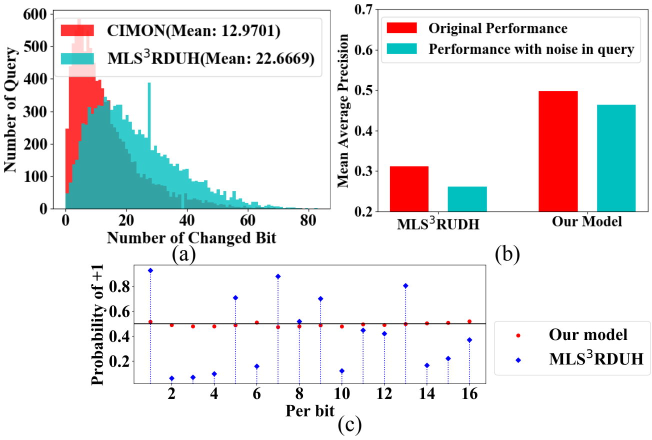


Figure 4: (a) Distribution of changed bits number after adding noise in query images on CIFAR-10 for MLS^{3}RUDH and CIMON, respectively. (b) MAP results before and after adding noise to MLS^{3}RUDH and CIMON. (c) The per-bit probability of code +1 over all images for 16 bits on CIFAR-10.

break the semantic information. Figure 4(a) shows the distribution of changed bits number before and after adding noise in query images for MLS^{3}RUDH and our model. It is observed that the mean of changed bits number of CIMON is significantly smaller than that of MLS^{3}RUDH, which implies that CIMON can learn more disturb-invariant hash codes. The MAP of CIMON also decreases less after the noise attack compared with the baseline in Figure 4(b). Moreover, CIMON is able to generate informative hash bits because hash bits distribution of CIMON approximates a uniform distribution, making good use of full bit capacity in Figure 4(c).

	Correlations				Results			
	GR	CW	SC	CC	16bits	32bits	64bits	128bits
M_1					0.1907	0.2253	0.2428	0.2592
M_2	√				0.2101	0.2552	0.2640	0.2796
M_3	√	√			0.3251	0.3260	0.3354	0.3452
M_4	√	√	√		0.3815	0.4315	0.4399	0.4437
M_5	√	√	√	√	**0.4506**	**0.4723**	**0.4944**	**0.4981**

Table 2: Ablation analysis on CIFAR-10. GR, CW, SC and CC correspond to Global Refinement, Confidence-based Weight, Semantic Consistency and Contrastive Consistency, respectively.

Ablation Study We investigate the effectiveness of various correlations in Table 2. M_1 uses the local similarity structure as guiding information, and trains the hashing network with the degraded loss following [Yang *et al.*, 2018]. The difference between M_2 and M_1 lies in whether to use the global refinement or not. It can be seen that M_2 surpasses M_1 significantly, demonstrating the effectiveness of global refinement for reconstructing the accurate similarity graph. After considering the confidence of semantic similarity, M_3 achieves better results than M_2 because the refined similarity-graph is still noisy and M_3 further accounts for the variations in confident and unconfident pairs, which eases the effect of false similarity signals and enlarges the effect of highly confident signals in the similarity graph. M_4 makes use of the data augmentation and our novel semantic consistency loss function. We can see that M_4 performs much better than M_3, which demonstrates the strength of data augmentation and our well-designed semantic consistency loss. By comparing the results of M_5 and M_4, we can see that the contrastive consistency can further improve the performance of our model.

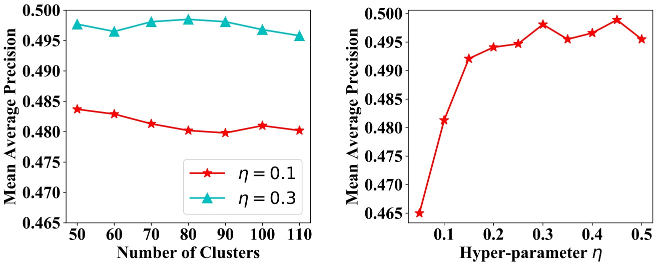


Figure 5: Sensitivity of K and η with 128-bits on CIFAR-10

Parameter Sensitivity We study the influence of balance coefficient η and the number of clusters K in Figure 5. We first fix η to 0.1 and 0.3 and evaluate the MAP by varying K from 50 to 110. The performance of our model is not sensitive to the number of clusters in the range of $[50, 110]$ and we can set K as any values in that interval. Furthermore, we show the MAP by varying η from 0.05 to 0.5 with K fixed to 70. The MAP of our model first increases and then keeps at a relatively high level. The result is not sensitive to η in the range of $[0.2, 0.5]$. Then for the proposed model, K and η are set to 70 and 0.3 respectively.

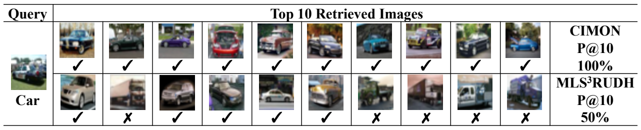


Figure 6: Examples of the top 10 images and Precision@10

Visualization In Figure 6, we visualize the top 10 returned images of our model and the best baseline for the query image of CIFAR-10, which demonstrates that our model can retrieve much more relevant and user-desired images.

5 Conclusion

Here we propose a novel deep hashing method named CIMON, which generates reliable semantic information by comprehensive similarity mining from local and global views. Then a novel consistency loss function from the view of semantic matching and contrastive learning is proposed to optimize the hashing model by incorporating the semantic information into the training process. Extensive experiments reveal that CIMON boosts the state-of-the-art unsupervised hashing schemes in both image retrieval and robustness.

Acknowledgements

This work was supported by The National Key Research and Development Program of China (No.2016YFA0502303) and the National Natural Science Foundation of China (No.31871342).

References

[Boussaha *et al.*, 2019] Basma El Amel Boussaha, Nicolas Hernandez, Christine Jacquin, and Emmanuel Morin. Multi-level context response matching in retrieval-based dialog systems. In *Proceedings of the 7th edition of the Dialog System Technology Challenges Workshop at AAAI*, 2019.

[Chen *et al.*, 2020] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

[Chua *et al.*, 2009] Tat-Seng Chua, Jinhui Tang, Richang Hong, Haojie Li, Zhiping Luo, and Yantao Zheng. Nus-wide: a real-world web image database from national university of singapore. In *Proceedings of the ACM international Conference on Image and Video Retrieval*, page 48, 2009.

[Dai *et al.*, 2017] Bo Dai, Ruiqi Guo, Sanjiv Kumar, Niao He, and Le Song. Stochastic generative hashing. In *ICML*, pages 913–922, 2017.

[Dosovitskiy *et al.*, 2014] Alexey Dosovitskiy, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with convolutional neural networks. In *NIPS*, pages 766–774. 2014.

[Fan *et al.*, 2020] Lixin Fan, Kam Woh Ng, Ce Ju, Tianyu Zhang, and Chee Seng Chan. Deep polarized network for supervised learning of accurate binary hashing codes. In *IJCAI*, pages 825–831, 2020.

[Gong *et al.*, 2012] 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.

[Gu *et al.*, 2019] Yifan Gu, Shidong Wang, Haofeng Zhang, Yazhou Yao, and Li Liu. Clustering-driven unsupervised deep hashing for image retrieval. *Neurocomputing*, 368:114–123, 2019.

[Hadsell *et al.*, 2006] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, pages 1735–1742, 2006.

[He *et al.*, 2020] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, pages 9729–9738, 2020.

[Heo *et al.*, 2012] Jae-Pil Heo, Youngwoon Lee, Junfeng He, Shih-Fu Chang, and Sung-Eui Yoon. Spherical hashing. In *CVPR*, pages 2957–2964, 2012.

[Huiskes and Lew, 2008] Mark J Huiskes and Michael S Lew. The mir flickr retrieval evaluation. In *ACMMM*, pages 39–43, 2008.

[Jin *et al.*, 2013] Zhongming Jin, Cheng Li, Yue Lin, and Deng Cai. Density sensitive hashing. *IEEE transactions on cybernetics*, 44(8):1362–1371, 2013.

[Krizhevsky *et al.*, 2009] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[Lin *et al.*, 2016] Kevin Lin, Jiwen Lu, Chu-Song Chen, and Jie Zhou. Learning compact binary descriptors with unsupervised deep neural networks. In *CVPR*, pages 1183–1192, 2016.

[Luo *et al.*, 2020] Xiao Luo, Chong Chen, Huasong Zhong, Hao Zhang, Minghua Deng, Jianqiang Huang, and Xiansheng Hua. A survey on deep hashing methods. *arXiv preprint arXiv:2003.03369*, 2020.

[Luo *et al.*, 2021] Xiao Luo, Zeyu Ma, Daqing Wu, Huasong Zhong, Jinwen Ma, and Minghua Deng. Deep unsupervised hashing by distilled smooth guidance. In *ICME*, 2021.

[Shen *et al.*, 2018] Fumin Shen, Yan Xu, Li Liu, Yang Yang, Zi Huang, and Heng Tao Shen. Unsupervised deep hashing with similarity-adaptive and discrete optimization. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):3034–3044, 2018.

[Simonyan and Zisserman, 2015] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, May 2015.

[Tu *et al.*, 2020] Rong-Cheng Tu, Xian-Ling Mao, and Wei Wei. Mls3rduh: Deep unsupervised hashing via manifold based local semantic similarity structure reconstructing. In *IJCAI*, pages 3466–3472, 2020.

[Wang and Isola, 2020] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *ICML*, pages 9929–9939, 2020.

[Wang *et al.*, 2020] Weiwei Wang, Yuming Shen, Haofeng Zhang, Yazhou Yao, and Li Liu. Set and rebase: determining the semantic graph connectivity for unsupervised cross modal hashing. In *IJCAI*, pages 853–859, 2020.

[Wu *et al.*, 2018] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, pages 3733–3742, 2018.

[Wu *et al.*, 2019] Jianlong Wu, Keyu Long, Fei Wang, Chen Qian, Cheng Li, Zhouchen Lin, and Hongbin Zha. Deep comprehensive correlation mining for image clustering. In *CVPR*, pages 8150–8159, 2019.

[Xie *et al.*, 2020] Yanzhao Xie, Yu Liu, Yangtao Wang, Lianli Gao, Peng Wang, and Ke Zhou. Label-attended hashing for multi-label image retrieval. In *IJCAI*, pages 955–962, 2020.

[Yang *et al.*, 2018] Erkun Yang, Cheng Deng, Tongliang Liu, Wei Liu, and Dacheng Tao. Semantic structure-based unsupervised deep hashing. In *IJCAI*, pages 1064–1070, 2018.

[Yang *et al.*, 2019] Erkun Yang, Tongliang Liu, Cheng Deng, Wei Liu, and Dacheng Tao. Distillhash: Unsupervised deep hashing by distilling data pairs. In *CVPR*, pages 2946–2955, 2019.