

Deep Collaborative Discrete Hashing with Semantic-Invariant Structure

Zijian Wang[†]

The University of Queensland
Brisbane, Australia
zijian.wang@uq.net.au

Yadan Luo

The University of Queensland
Brisbane, Australia
lyadanluo@gmail.com

Zheng Zhang^{†*}

The University of Queensland
Brisbane, Australia
darrenzz219@gmail.com

Zi Huang

The University of Queensland
Brisbane, Australia
huang@itee.uq.edu.au

ABSTRACT

Existing deep hashing approaches fail to fully explore semantic correlations and neglect the effect of linguistic context on visual attention learning, leading to inferior performance. This paper proposes a dual-stream learning framework, dubbed Deep Collaborative Discrete Hashing (DCDH), which constructs a discriminative common discrete space by collaboratively incorporating the shared and individual semantics deduced from visual features and semantic labels. Specifically, the context-aware representations are generated by employing the outer product of visual embeddings and semantic encodings. Moreover, we reconstruct the labels and introduce the focal loss to take advantage of frequent and rare concepts. The common binary code space is built on the joint learning of the visual representations attended by language, the semantic-invariant structure construction and the label distribution correction. Extensive experiments demonstrate the superiority of our method.

CCS CONCEPTS

• Information systems → Retrieval efficiency; Image search.

KEYWORDS

Learning to Hash; class encoding; semantic-preserving hashing.

ACM Reference Format:

Zijian Wang, Zheng Zhang, Yadan Luo, and Zi Huang. 2019. Deep Collaborative Discrete Hashing with Semantic-Invariant Structure. In *42nd Int'l ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '19)*, July 21–25, 2019, Paris, France. ACM, New York, NY, USA, 4 pages. <https://doi.org/10.1145/3331184.3331275>

* indicates corresponding author; † indicates co-first authors with equal contributions.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

SIGIR '19, July 21–25, 2019, Paris, France

© 2019 Association for Computing Machinery.

ACM ISBN 978-1-4503-6172-9/19/07...\$15.00

<https://doi.org/10.1145/3331184.3331275>

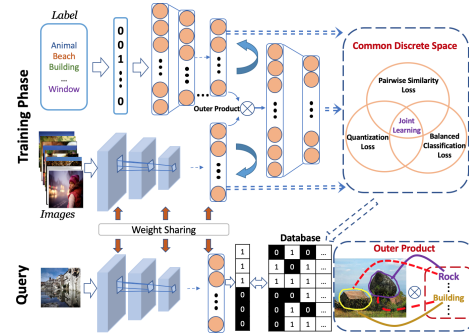


Figure 1: The proposed deep collaborative discrete hashing framework. A dual-stream network consists of feature embedding network and label encoding network. We sketch the strength of the outer product in the bottom right subfigure.

1 INTRODUCTION

In this big data era, large volume and high-dimensional multimedia data is ubiquitous in social networks and search engines. This leads to the major challenge of how to efficiently retrieve information from the large-scale database [15]. To guarantee retrieval efficiency and quality, approximate nearest neighbour (ANN) search has attracted increasing attention in recent years. Parallel to the traditional indexing methods, hashing is one of the most advantaged methods in existing ANN methods, as it transforms high dimensional multimedia data into compact binary codes and enables efficient xor operations to accelerate calculation in Hamming space. In this paper, we will focus on learning to hash methods which build upon data-dependent binary encoding schemes for efficient image retrieval, which have demonstrated superior performance over data-independent hashing methods, e.g. LSH [3].

Generally, learning to hash methods can be divided into unsupervised and supervised groups. Compared with unsupervised methods [4, 19], supervised methods [5, 14, 17, 18, 20] can yield better performance with the support of label supervision. With the rapid development of deep neural network, deep hashing methods [1, 2, 7, 11, 12, 16, 21] have demonstrated superior performance over non-deep hashing methods and achieved state-of-the-art results on public benchmarks.

However, among mainstream deep hashing frameworks, human-annotated labels purely supervise the distribution alignment of hash code embedding, yet fail to trigger context-aware visual representation learning, let alone optimal binary codes generation. Moreover, the correlations between features and semantics are not well-explored to generate semantic consistent binary codes. Furthermore, existing supervised methods are vulnerable to the imbalanced distribution of semantic labels. Models tend to grasp the frequently appeared concepts in the training data and disregard the infrequent ones, which highly restricts the expression capacity of hash codes. Hence, existing deep hashing methods may fail to generate optimal binary codes for efficient image retrieval.

In this paper, we propose a novel Deep Collaborative Discrete Hashing (DCDH) method, which constructs a discriminative common discrete space via dual-stream learning, as illustrated in Figure 1. The main idea of the proposed framework is to construct a semantic invariant space, via bridging the gap between visual space and semantic space. Specifically, (1) We develop a bilinear representation learning framework, which significantly fuses and strengthens visual-semantic correlations to learn context-aware binary codes. (2) We employ outer product on visual features and label embeddings to generate more expressive representations rather than element-wise product or plain concatenation. To the best of our knowledge, this is one of the first attempts to utilize the outer product to capture pairwise correlations between heterogeneous spaces. (3) We seamlessly integrate our framework with the focal loss to enhance the discriminant of generated binary codes and mitigate the class-imbalance problem by reducing weights on the well classified concepts and increasing weights on rare concepts. (4) Extensive experiments conducted on benchmark datasets demonstrate that DCDH is capable to generate more discriminative and informative binary codes and yield state-of-the-art performance.

2 THE PROPOSED APPROACH

2.1 Problem Formulation

Given a set of n images $\mathcal{D} = \{\mathbf{x}_i, \mathbf{l}_i\}_{i=1}^n$, where \mathbf{x}_i and $\mathbf{l}_i \in \{0, 1\}^c$ are the i -th image and corresponding one-hot label vector, respectively. Deep hashing aims to encode data $X \in \mathbb{R}^{n \times d}$ as k -bits binary codes $B \in \mathbb{R}^{n \times k}$. In our method, we mainly focus on the pairwise similarity-preserving hashing. In particular, we construct the similarity information based on ground-truth label. If two images i and j share at least one common label, we define i and j are semantically similar and $S_{ij} = 1$, otherwise $S_{ij} = -1$ indicating dissimilar.

2.2 Deep Visual Embedding Network

The purpose of the visual embedding network is to generate discriminative hash codes such that similar pairs can be distinguished from dissimilar pairs. Specifically, Hamming distance between b_i and b_j should be minimized when $S_{ij} = 1$, while maximized when $S_{ij} = -1$. To preserve the pairwise similarities [10], our work adopts smooth L_2 loss defined on the inner product between binary codes as:

$$\min_{b_i, b_j} \mathcal{L}_1 = \sum_{i=1}^n \sum_{j=1}^n \|b_i^T b_j - k S_{ij}\|_F^2 \text{ s.t. } b_i, b_j \in \{-1, 1\}^k, \quad (1)$$

However, it is difficult to generate the *discrete* outputs. We can set $b_i = \text{sgn}(F_v(x_i; \theta_v))$, where θ_v denotes the parameters of deep visual embedding network.

$$\min_{\theta, B} \mathcal{L}_1 = \|\text{sgn}(F_v(x_i; \theta_v))^T B - \rho S\|_F^2 \text{ s.t. } B \in \{-1, 1\}^{n \times k}, \quad (2)$$

where $S \in \{1, -1\}^{n \times n}$ is the binary similarity matrix. In this paper, we designed an end-to-end feature learning network which extends the pretrained AlexNet [6] model for discriminative visual embedding learning. Based on this backbone network, we replace the final classifier layer with a fully connected layer to transform the convolutional feature maps into the k -dimensional continuous codes U . Subsequently, we apply hyperbolic tangent (\tanh) as the activation function to approximate non-differential signum (sgn) function, i.e., $b_i = \text{sgn}(u_i)$. To control the quantization error and bridge the gap between the binary codes and its relaxation, we add an extra penalty term to keep U_v and B as close as possible. We adopt the following matrix-form loss function to facilitate the network back-propagate the gradient to θ_v . Hence, the problem in (2) is transformed into the following problem:

$$\begin{aligned} \min_{B, \theta_v} \mathcal{L}_1 &= \|U_v^T B - \rho S\|_F^2 + \alpha \|B - U_v\|_F^2 \\ \text{s.t. } B &\in \{-1, 1\}^{n \times k}, U_v = \tanh(F_v(x_i; \theta_v)), \end{aligned} \quad (3)$$

where α is a weighting parameter.

2.3 Deep Class Encoding Network

In pairwise-preserving hashing methods, labels are always exploited as similarity measurement between data points by applying the element-wise inner product. However, solely using similarity matrix to supervise hash codes learning inevitably results in severe information loss and thus highly restricts the expression ability of generated hash codes, especially in multi-label cases. To be more specific, one image annotated by multiple labels (such as 'ocean', 'beach' and 'water') contains underlying semantic connections in concepts, while single class vector may hinder the conceptual bridge at a fine-grained level. The purpose of the label encoding network is to capture the original semantic information and preserve them in k -dimensional flexible continuous space. Similarly, the loss function of the label encoding network can be defined as:

$$\begin{aligned} \min_{B, \theta_l} \mathcal{L}_2 &= \|\tanh(U_l)^T B - \rho S\|_F^2 + \alpha \|B - \tanh(U_l)\|_F^2 \\ \text{s.t. } U_l &= F_l(L; \theta_l), \end{aligned} \quad (4)$$

where $F_l(\cdot; \theta_l)$ denotes the label encoding network parameterized by θ_l . By providing with complementary views of semantics, the label encoding network potentially guides the visual embedding network to learn beneficial context-aware representations.

2.4 Semantic Invariant Structure Construction

To disentangle the relationships between the abstract concepts and the visual features, we apply the outer product to fuse visual and label embeddings. Being distinct from the conventional element-wise product or plain concatenation, the applied outer product allows high-level label encoding and low-level visual feature embeddings to interactively influence each other. In this way, we can capture the

pairwise correlations between the feature of an image and its corresponding label, enabling discovery of the common latent attributes. By applying the outer product, we first obtain the pairwise interaction between label and image features. After the training procedure, the semantic information is well separated by the related region in the image. The latent vector is obtained by reshaping the pairwise correlation matrix to a vector, which can project to discrete space to generate hash codes. The generated codes are more discriminative since the outer product operator ensures the bits truly reflect regions in the images to the corresponding semantic information. To construct the semantic invariant structure, the objective function can be formulated as:

$$\begin{aligned} \min_{B, \theta} \mathcal{L}_3 &= \|\tanh(U)^T B - \rho S\|_F^2 + \alpha \|B - \tanh(U)\|_F^2 \\ \text{s.t. } U &= F(U_v \otimes U_l; \theta), \end{aligned} \quad (5)$$

where \otimes denotes the outer product and $F(\cdot; \theta)$ denotes the fusion network parameterized by θ .

Furthermore, we introduce the focal loss [8] to mitigate the side effect from class imbalance, and the objective function is:

$$\begin{aligned} \min_{B, \theta, \theta_c} \mathcal{L}_3 &= \|\tanh(U)^T B - \rho S\|_F^2 + \alpha \|B - \tanh(U)\|_F^2 \\ &+ \sum_{i=1}^n \sum_{t=1}^c -(1 - p_{i,t})^\gamma \log(p_{i,t}) \\ \text{s.t. } p_i &= \text{softmax}(F_c(u_i; \theta_c)), \end{aligned} \quad (6)$$

where γ is the hyper-parameter. The $F_c(\cdot; \theta_c)$ denotes the classification layer parameterized by θ_c , and $p_{i,t}$ is the estimated probability of the i -th sample for the t -th class. The adaptive factor $p_{i,t}$ is:

$$p_{i,t} = \begin{cases} p_{i,t}, & \text{if } l_{i,t} = 1; \\ 1 - p_{i,t}, & \text{otherwise.} \end{cases} \quad (7)$$

2.5 Collaborative Learning

To learn context-aware and discriminative hash codes, we adopt the joint learning framework consisting of the visual embedding network, the label encoding network and the semantic-invariant structure construction, and we have:

$$\min_{B, \theta_v, \theta_l, \theta_c, \theta} \mathcal{L} = \mathcal{L}_1 + \lambda \mathcal{L}_2 + \mu \mathcal{L}_3, \quad (8)$$

where λ and μ are coefficients to weight the importance of different terms. \mathcal{L}_1 , \mathcal{L}_2 and \mathcal{L}_3 denote the visual embedding net loss, the label encoding net loss and semantic invariant structure construction loss, respectively.

2.6 Optimization

The proposed DCDH needs to jointly optimize Eqn. (3), (4), and (6). Due to similar forms, we only illustrate one detailed optimization on Eqn. (6), and the rest equations can be solved similarly. Specifically, we adopt the iterative alternating optimization manner, that is, we update one variable with others fixed.

Learning θ_v : The network parameters θ_v can be optimized via standard back-propagation algorithm by automatic differentiation techniques in Pytorch [13].

Learning B : We aim to optimize \tilde{B} with all hyper-parameters fixed,

Table 1: The averaged retrieval MAP comparison on NUS-WIDE, and MIRFlickr. The best performance are shown in boldface.

Methods	NUS-WIDE				MIRFlickr			
	12-bits	24-bits	32-bits	48-bits	12-bits	24-bits	32-bits	48-bits
LSH	0.3942	0.4049	0.4305	0.4331	0.5456	0.5501	0.5460	0.5523
ITQ	0.5435	0.5544	0.5536	0.5560	0.6243	0.6305	0.6318	0.6359
KSH	0.5701	0.5735	0.5797	0.5788	0.6135	0.6144	0.6213	0.6176
SDH	0.6769	0.6914	0.6981	0.7052	0.8018	0.8258	0.8267	0.8387
LFH	0.7152	0.7446	0.7512	0.7722	0.8258	0.8364	0.8281	0.8573
COSDISH	0.7398	0.7678	0.7819	0.7888	0.7736	0.7973	0.8589	0.8693
DHN	0.7719	0.8013	0.8051	0.8146	0.8092	0.8283	0.8290	0.8411
DVSQ	0.7856	0.7924	0.7976	0.8019	0.8112	0.8263	0.8288	0.8341
DPSH	0.7941	0.8249	0.8351	0.8442	0.6857	0.7058	0.7140	0.7182
DTQ	0.7951	0.7993	0.8012	0.8024	0.8098	0.8274	0.8456	0.8511
DCDH (ours)	0.8223	0.8526	0.8712	0.8974	0.8758	0.8970	0.9059	0.9079

and we rewrite the Eqn. (6) as follows:

$$\begin{aligned} \min_B \mathcal{L}_1 &= \|\tanh(U)^T B\|_F^2 - 2\rho \text{Tr}(B^T \tanh(U)S) \\ &- 2\alpha \text{Tr}(B^T \tanh(U)) + \text{const}, \end{aligned} \quad (9)$$

where $\text{Tr}(\cdot)$ is the trace norm. Since focal loss is independent to \tilde{B} update, we can consider focal loss as constant when learning \tilde{B} .

It is challenging to directly optimize \tilde{B} due to its discrete constraint. Inspired by [14], we learn binary codes B by the DCC strategy, in which non-differential variable can be solved in a bit-by-bit manner. Therefore, problem (9) can be reformulated to minimize

$$\begin{aligned} \|\tanh(U)^T B\|_F^2 &- 2\left(\rho \text{Tr}(B^T \tanh(U)S) + \gamma \text{Tr}(B^T \tanh(U))\right) + \text{const} \\ &= \text{Tr}\left(2(u^T(U')^T B' - q^T)z\right) + \text{const} \quad \text{s.t. } B \in \{-1, 1\}, \end{aligned} \quad (10)$$

where $Q = \rho \tanh(U)S + \gamma \tanh(U)$, and $\tilde{U} = \tanh(U)$. u^T is the row of \tilde{U} , U' denotes the matrix of \tilde{U} exclude u . z^T is the row of B , B' denotes the matrix of B exclude z . q^T is the row of Q , Q' denotes the matrix of Q exclude q . Eventually, we can get the following optimal solution of problem (10) that can be used to update z :

$$z = \text{sgn}(u^T(U')^T B' - q^T). \quad (11)$$

The network parameters can be efficiently optimized through standard back propagation algorithm by using automatic differentiation techniques by PyTorch [13].

2.7 Out of Sample Extension

Based on the proposed optimization method, we can obtain the optimal binary codes for all the training data and the optimized visual embedding learning network, i.e., $F_v(x_i; \theta_v)$. Our learning framework can easily generate the binary codes of a new query x_q by using the visual network followed by the signum function:

$$b_q = \phi(x_q; \theta_v) = \text{sgn}(F_v(x_q; \theta_v)) \quad (12)$$

3 EXPERIMENTS

We conduct extensive experiments to evaluate our method against several state-of-the-art hashing methods on NUS-WIDE and MIRFlickr. NUS-WIDE contains 269,648 images with 81 tags. Following [7], we select a subset of 195,834 images that are included in the 21 most frequent classes. MIRFlickr contains 25,000 images from Flickr website, in which each image is tagged by at least one of 38 concepts. Following evaluation splits in [5, 14], we randomly

Table 2: Ablation study of our DCDH method.

Methods	NUS-WIDE		MIRFlickr	
	12-bits	48-bits	12-bits	48-bits
DCDH-V	0.7477	0.8153	0.7842	0.8681
DCDH-S	0.7677	0.8523	0.8146	0.8875
DCDH	0.8223	0.8974	0.8758	0.9079

sample 2,100 and 1700 images as query sets for NUS-WIDE and MIRFlickr, respectively, and the rest are utilized for training.

We compare our DCDH with 10 state-of-the-art hashing methods, which include 4 non-deep hashing methods (i.e. KSH [10], SDH [14], COSDISH [5], LFH [17]), 4 deep hashing methods (i.e. DPSH [7], DHN [21], DVSQ [1], DTQ [9]), 1 unsupervised method (i.e. ITQ [4]) and 1 data independent method (i.e. LSH [3]). For fair comparison, we employ 4096-dim deep features extracted from AlexNet [6] for non-deep methods. Two evaluation metrics, *i.e.*, Mean Average Precision (MAP), and Precision@top K, are used for performance comparison.

3.1 Results

The MAP results of different methods on NUS-WIDE, and MIRFlickr are reported in Table 1. (1) Generally, taking advantage of semantic information, supervised methods can achieve better retrieval performance than unsupervised methods, while ITQ can obtain competitive results. (2) Deep hashing methods can outperform shallow methods in most cases, since deep hashing methods benefit from learning discriminative representations and non-linear hash functions. (3) From MAP results in Table 1 and percision@top K curves in Figure 2, we can observe DCDH outperforms other comparison methods by a large margin. Our proposed method always produces the best performance on both of the benchmarks, which emphasizes the importance of semantic invariant structure construction and excavating the underlying semantic correlation.

3.2 Ablation Study

We investigate two variants of DCDH: 1) DCDH-V utilizes the visual feature embedding net solely to generate hash codes. 2) DCDH-S leaves alone the semantic encoding and visual feature embedding to generate hash codes. We report the results of DCDH variants in Table 2 with 12 bits hash codes and 48 bits hash codes on NUS-WIDE, and MIRFlickr. Compared with full model DCDH, we observe that both DCDH-S and DCDH-V incur a descendant in MAP. DCDH-S can achieve better performance than DCDH-V after employing the supervision of encoded labels. The result further reveals that the importance of mining the semantic correlation between semantic information and local visual features.

4 CONCLUSION

In this paper, we proposed a novel deep supervised hashing framework, which collaboratively explores the visual feature representation learning, semantic invariant structure construction, and label distribution correction. A discriminative common discrete Hamming space was constructed by concurrently considering the shared and model-specific semantic information from visual features and context annotations. Moreover, the class imbalance problem was

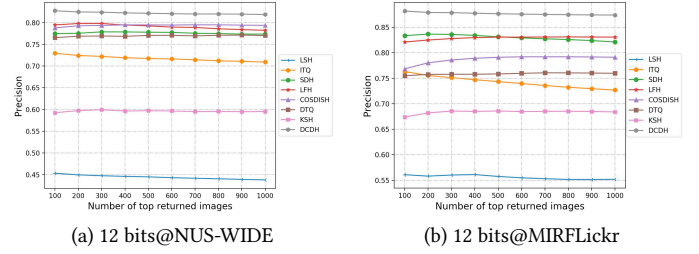


Figure 2: The precision curves of the top-N returned samples on the NUS-WIDE in (a) and MIRFlickr in (b).

addressed to leverage frequent and rare concepts. Extensive experimental results demonstrate the superiority of the proposed joint learning framework.

ACKNOWLEDGMENTS

This work is supported by ARC discovery project DP190102353.

REFERENCES

- [1] Yue Cao, Mingsheng Long, Jianmin Wang, and Shichen Liu. 2017. Deep visual-semantic quantization for efficient image retrieval. In *CVPR*. 1328–1337.
- [2] Yue Cao, Mingsheng Long, Jianmin Wang, Han Zhu, and Qingfu Wen. 2016. Deep quantization network for efficient image retrieval. In *AAAI*. 3457–3463.
- [3] Aristides Gionis, Piotr Indyk, Rajeev Motwani, et al. 1999. Similarity search in high dimensions via hashing. In *VLDB*. 518–529.
- [4] Yunchao Gong, Svetlana Lazebnik, Albert Gordo, and Florent Perronnin. 2013. Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval. *IEEE TPAMI* 35, 12 (2013), 2916–2929.
- [5] Wang-Cheng Kang, Wu-Jun Li, and Zhi-Hua Zhou. 2016. Column sampling based discrete supervised hashing. In *AAAI*. 1230–1236.
- [6] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. ImageNet classification with deep convolutional neural networks. In *NIPS*. Curran Associates, Inc., 1097–1105.
- [7] Wu-Jun Li, Sheng Wang, and Wang-Cheng Kang. 2015. Feature learning based deep supervised hashing with pairwise labels. (2015), 1711–1717.
- [8] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *ICCV*. 2980–2988.
- [9] Bin Liu, Yue Cao, Mingsheng Long, Jianmin Wang, and Jingdong Wang. 2018. Deep triplet quantization. *ACMM* (2018).
- [10] Wei Liu, Jun Wang, Rongrong Ji, Yu-Gang Jiang, and Shih-Fu Chang. 2012. Supervised hashing with kernels. In *CVPR*. 2074–2081.
- [11] Yadan Luo, Yang Li, Fumin Shen, Yang Yang, Peng Cui, and Zi Huang. 2018. Collaborative learning for extremely low bit asymmetric hashing. *CoRR* abs/1809.09329 (2018). arXiv:1809.09329
- [12] Yadan Luo, Yang Yang, Fumin Shen, Zi Huang, Pan Zhou, and Heng Tao Shen. 2018. Robust discrete code modeling for supervised hashing. *Pattern Recognition* 75 (2018), 128–135.
- [13] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in PyTorch. (2017).
- [14] Fumin Shen, Chunhua Shen, and Wei Liu. 2015. Supervised discrete hashing. In *CVPR*. 37–45.
- [15] Jingdong Wang, Ting Zhang, Nicu Sebe, Heng Tao Shen, et al. 2018. A survey on learning to hash. *IEEE Trans. PAMI* 40, 4 (2018), 769–790.
- [16] Yang Yang, Yadan Luo, Weilun Chen, Fumin Shen, Jie Shao, and Heng Tao Shen. 2016. Zero-shot hashing via transferring supervised knowledge. In *ACMM*.
- [17] Peichao Zhang, Wei Zhang, Wu-Jun Li, and Minyi Guo. 2014. Supervised hashing with latent factor models. In *SIGIR*. 173–182.
- [18] Zheng Zhang, Zhihui Lai, Zi Huang, W. Wong, Guosen Xie, and Li Liu. 2019. Scalable supervised asymmetric hashing with semantic and latent factor embedding. *IEEE Trans. IP* 99, 99 (2019), 1–16.
- [19] Zheng Zhang, Li Liu, Fumin Shen, Heng Tao Shen, and Shao Ling. 2018. Binary Multi-View Clustering. *IEEE Trans. PAMI* 99, 99 (2018), 1–9.
- [20] Zheng Zhang, Guosen Xie, Yang Li, Sheng Li, and Zi Huang. 2019. SADIH: Semantic-Aware Discrete Hashing. In *AAAI*. 12–19.
- [21] Han Zhu, Mingsheng Long, Jianmin Wang, and Yue Cao. 2016. Deep Hashing Network for Efficient Similarity Retrieval. In *AAAI*. 2415–2421.