Deep Hashing with Active Pairwise Supervision

Ziwei Wang[1,2,3], Quan Zheng[1,2,3], Jiwen Lu[1,2,3] ⋆, and Jie Zhou[1,2,3,4]

[1] Department of Automation, Tsinghua University, China
[2] State Key Lab of Intelligent Technologies and Systems, China
[3] Beijing National Research Center for Information Science and Technology, China
[4] Tsinghua Shenzhen International Graduate School, Tsinghua University, China
{wang-zw18,zhengq16}@mails.tsinghua.edu.cn,
{lujiwen,jzhou}@tsinghua.edu.cn

Abstract. In this paper, we propose a Deep Hashing method with Active Pairwise Supervision(DH-APS). Conventional methods with passive pairwise supervision obtain labeled data for training and require large amount of annotations to reach their full potential, which are not feasible in realistic retrieval tasks. On the contrary, we actively select a small quantity of informative samples for annotation to provide effective pairwise supervision so that discriminative hash codes can be obtained with limited annotation budget. Specifically, we generalize the structural risk minimization principle and obtain three criteria for the pairwise supervision acquisition: uncertainty, representativeness and diversity. Accordingly, samples involved in the following training pairs should be labeled: pairs with most uncertain similarity, pairs that minimize the discrepancy between labeled and unlabeled data, and pairs which are most different from the annotated data, so that the discriminality and generalization ability of the learned hash codes are significantly strengthened. Moreover, our DH-APS can also be employed as a plug-and-play module for semi-supervised hashing methods to further enhance the performance. Experiments demonstrate that the presented DH-APS achieves the accuracy of supervised hashing methods with only 30% labeled training samples and improves the semi-supervised binary codes by a sizable margin.

Keywords: Active Learning · Deep Hashing · Structural Risk Minimization

1 Introduction

Large scale image search, which aims to retrieve images with similar content from the database given a query image, has aroused extensive interest in computer vision due to its wide application [17, 1, 21]. Although conventional methods based on trees [49], nearest neighbor search [36] and quantization [20] have been broadly employed in low-dimensional data retrieval, they are not feasible for

⋆ Corresponding author

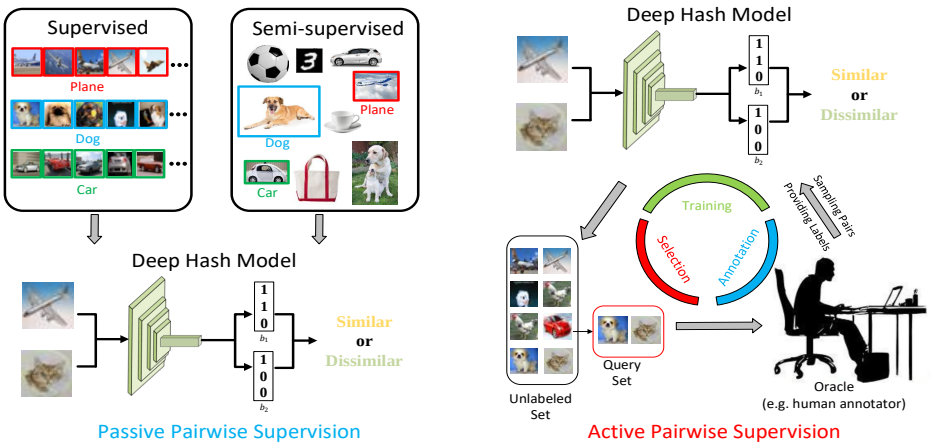

Fig. 1. Deep hashing methods with the passive and active pairwise supervision. For the former, supervised methods require exhaustive annotation with unbearable annotation cost to reach the full potential, and semi-supervised methods randomly select a few samples to label so that effective supervision is not provided. Our DH-APS selects samples providing effective pairwise supervision to label so that discriminative and generalizable binary codes are learned with limited annotation budget.

high-dimensional visual data due to the unbearable computational complexity and storage cost. Hence, it is desirable to extract compact features for the high-dimensional data in image search.

Recently, hashing-based approximating nearest neighbor search method have been presented to learn binary codes [41, 12, 53, 10, 11, 29]. The storage and the computational cost of retrieval is decreased significantly, as Hamming distance instead of Euclidean distance between different hash codes is calculated when comparing the similarity of various instances. The objective of hashing-based methods is to learn a set of hash functions that maps each visual sample into a compact binary feature vector, where conceptually similar samples are hashed into similar binary codes. As limited bitwidth degrades the representational capacity of the representations, deep neural networks are applied to learn informative hash codes. Because unsupervised deep hashing methods suffer from low discriminative power due to the lack of supervision, supervised deep hash models boost the performance of the learned binary codes. However, exhaustive labeling for learning supervised hash codes require large amount of cost, which is prohibited in realistic applications with limited annotation budget. Moreover, semi-supervised deep hashing methods randomly select partial instances for annotation and fail to provide effective pairwise supervision for hash code learning.

In this paper, we propose a Deep Hashing method with Active Pairwise Supervision (DH-APS) to learn effective binary codes for image search with limited annotation budget. Unlike methods applying passive pairwise supervision which require to label all training samples to reach the full potential, our method only

annotates a few samples which provide effective pairwise supervision, so that discriminative and generalizable binary codes are learned with limited annotation cost. More specifically, we extend the structural risk minimization principle to active deep hashing and obtain three annotation acquisition criteria: uncertainty, representativeness and diversity. As the goal of hashing is similarity preservation, our acquisition function is based on the pairwise relationship instead of individual samples that are usually considered in conventional active learning methods [54, 48, 28]. Accordingly, we label samples involved in the following pairs: pairs with highest uncertainty which is measured by Shannon Entropy [37], pairs which minimize the maximum mean discrepancy (MMD) between the labeled set and the unlabeled set, and pairs that have minimal similarity with the samples in the labeled set. Moreover, our method can also be employed as a plug-and-play module for semi-supervised deep hashing method to further enhance the performance. Fig. 1 shows deep hashing methods with passive and active pairwise supervision. Extensive experiments on CIFAR-10 [23], NUS-WIDE [8] and ImageNet [9] demonstrate that the proposed DH-APS obtains the competitive performance with supervised binary codes with only 30% annotated training samples and enhances the semi-supervised hash models by a large margin.

2 Related Work

Deep Hashing: Deep hashing has been widely studied in recent years due to strong discriminative power and the high efficiency in large scale visual search. Deep hashing obtains much better performance compared with hand-crafted and shallow binary codes due to the data-dependent hash functions and the employment of deep architectures. Existing deep hashing methods can be divided into three categories according to the type of supervision: unsupervised [12, 15, 42], supervised [24, 29, 50] and semi-supervised [46, 51] methods. For the first category, Liong *et al.* [12] utilized the deep neural networks with energy constraint objectives to enhance the discriminative ability of hash codes. Ghasedi *et al.* [15] applied the Generative Adversarial Networks (GANs) [16] to learn hash codes through which the reconstructed images were enforced to have minimum discrepancy with the real ones, so that the obtained binary representations acquired informativeness and independency. For supervised methods, relation among different samples or explicit class labels are usually employed as supervision for hash code learning. Liu *et al.* [29] enforced the similar samples to obtain closer binary codes and punished semantically dissimilar samples whose hash codes have short Hamming distance so that the learned binary representations could precisely preserve the topology of the semantic space. Yang *et al.* [50] used the category information to supervise the hash model, and the learned hash codes extracted the class-dependent information for image retrieval. For semi-supervised methods, Zhang *et al.* [51] mined the semantic topology between labeled and unlabeled samples and generated pseudo labels for unlabeled samples to leverage the limited supervision. Zhang *et al.* [52] utilized the knowledge distillation to train the student model for hash code generation, and the teacher network was

assembled by multiple students. Nevertheless, exhaustive labeling is not feasible in realistic application due to the large scale database and limited annotation budget, and randomly annotating part of samples in hash code learning fails to provide effective supervision.

Active Learning: Active learning aims to acquire better performance when learning with fewer labeled training samples by actively annotating part of the training data from a pool of unlabeled set. The criteria for active sample selection can be divided into two types: informativeness and representativeness. For the former category, the unlabeled data which the learner is most uncertain about is selected to annotate as effective supervision can enhance discriminality of the learner. The uncertainty can be defined as the entropy of the posterior probabilities [22, 30, 39], distance to the classification boundaries [27, 45], margin between the largest and the second largest posterior probabilities [2] and disagreement among independent classifiers [32, 44]. Gal *et al.* [14] utilized the neural networks to estimate the task-specific uncertainty through multiple forward passes in a data-driven manner. Beluch *et al.* [4] constructed a classifier committee comprising five deep neural networks to obtain accurate estimation of uncertainty disagreement. For the latter category, the samples that can represent the unlabeled pool are chosen to label as the learning over a representative subset is competitive over the whole pool. The representativeness can be measured by clustering [33], knowledge propagation [31, 19], expected model change [40, 13] and optimal experimental design which tries to query the representative samples directly [7]. Sener *et al.* [38] selected the core-set for annotation by minimizing the gap between an average loss over any given subset and the remaining data points. Meanwhile, as combining informativeness and representativeness can enhance the learner, a variety of hybrid strategies have been proposed for specific tasks [34, 28]. Active hashing has also been proposed in Zhen *et al.* [54] and Wang *et al.* [48], which only measured the uncertainty of individuals to select samples for annotation and failed to consider the representativeness and diversity. Meanwhile, pairwise relationship should be considered in the acquisition function because the goal of hashing is similarity preservation. However, existing methods just calculate the acquisition function according to individual samples, which leads to uninformative annotation. In this paper, we extend active learning to deep hashing by considering pairwise relationship so that samples providing effective pairwise supervision are labeled to learn discriminative and generalizable binary codes with limited annotation budget.

3 Approach

In this section, we first introduce the problem setting of deep hashing with active pairwise supervision and then build the link between the acquisition function for active annotation and the structural risk minimization principle. Finally, we design the acquisition function by considering pairwise relationship for active deep hashing.

3.1 Deep Hashing with Active Pairwise Supervision

The training data set $\mathcal{X}$ consists of an active seed set $\mathcal{L}$ including a few labeled samples $\{\boldsymbol{x}_L\}$, a large pool set $\mathcal{U}$ containing unlabeled data $\{\boldsymbol{x}_U\}$ and a query set $\mathcal{Q}$ comprising samples $\{\boldsymbol{x}_Q\}$ that are selected from $\mathcal{U}$ for the Oracle to label. For the initialization of active deep hashing, we randomly move only a few samples from $\mathcal{U}$ to $\mathcal{L}$, and $\mathcal{Q}$ is an empty set. Active deep hashing algorithms undergo three iterative steps listed as follows: (1) training the hash model $\mathcal{H}$ with the pairs sampled from $\mathcal{L}$; (2) selecting samples that can provide the most effective supervision from $\mathcal{U}$ based on the acquisition function $s(\mathcal{H}, \mathcal{U}, \mathcal{L})$ and adding them to $\mathcal{Q}$; (3) asking the Oracle to label the samples in $\mathcal{Q}$ and updating $\mathcal{L}$, $\mathcal{U}$ and $\mathcal{Q}$.

Let $\boldsymbol{f}^k$ be the k_{th} byte of the float feature for the input image, which is obtained after the projection of the hash model $\mathcal{H}$. The k_{th} bit of the hash code $\boldsymbol{b}^k$ is obtained as follows:

$$\boldsymbol{b}^k = sgn(\boldsymbol{f}^k) \tag{1}$$

where $sgn(x)$ means the sign function which equals to zero if x is negative and equals to one otherwise. Following the typical hinge loss in supervised hash model training [29], we optimize the following objective to learn the optimal binary codes:

$$\begin{aligned} J = & y||\boldsymbol{b}_a - \boldsymbol{b}_b||_2^2 + (1-y)\max(m - ||\boldsymbol{b}_a - \boldsymbol{b}_b||_2^2, 0) \\ & s.t. \quad \boldsymbol{b}_a, \boldsymbol{b}_b \in \{+1, -1\}^d \end{aligned} \tag{2}$$

where $\boldsymbol{b}_a$ and $\boldsymbol{b}_b$ are the learned binary codes of $\boldsymbol{x}_a$ and $\boldsymbol{x}_b$ in the labeled set, and sampling the labeled set constructs pairs for training. y is the label providing pairwise supervision, which equals to one if $\boldsymbol{x}_a$ and $\boldsymbol{x}_b$ are similar and zero otherwise. m is a margin threshold parameter assigned to be positive. The objective enforces the similar samples to be mapped into binary codes with short distance and punishes dissimilar sample pairs whose hash codes are close when their Hamming distance falls below m. As the sign function is non-differentiable and searching for the optimal solution of binary codes is NP-hard, we relax the optimization in (2) as the following problem:

$$\begin{aligned} J = & y||\boldsymbol{f}_a - \boldsymbol{f}_b||_2^2 + (1-y)\max(m - ||\boldsymbol{f}_a - \boldsymbol{f}_b||_2^2, 0) \\ & + \gamma(|||\boldsymbol{f}_a| - \mathbf{1}^d||_1 + |||\boldsymbol{f}_b| - \mathbf{1}^d||_1) \end{aligned} \tag{3}$$

where $\mathbf{1}^d$ is a all-one vector in d dimensions and $|\cdot|$ is the element-wise absolute value operation. $\boldsymbol{f}_a$ and $\boldsymbol{f}_b$ are the float feature for $\boldsymbol{b}_a$ and $\boldsymbol{b}_b$, and γ is an hyperparameter to balance the term for similarity preservation and quantization error minimization. In active deep hashing, pairs sampled from the labeled set are utilized to train the hash model via (3). Moreover, active deep hashing can also be integrated with semi-supervised methods [52] so that the performance of the semi-supervised binary codes can be further enhanced due to the effective supervision.

3.2 Structural Risk Minimization for Active Hashing

The target for supervised hashing is to learn a hash model that preserves similarity among all samples and generalizes well on unseen data. The structural risk minimization (SRM) principle illustrates the objective via minimizing the upper bound of the true risk under unknown data distribution, which holds for dataset containing n samples with the probability at least $1-\delta$ [3]:

$$\mathbb{E}(J(z)) \leqslant \hat{\mathbb{E}}(J(z)) + 2R_n(\Omega) + \sqrt{\frac{\ln 1/\delta}{n}} \tag{4}$$

where $J(z)$ means the loss over the data z. $\mathbb{E}(J(z))$ and $\hat{\mathbb{E}}(J(z))$ are the loss expectation over true distribution of z and the distribution sampled from the dataset, which are named true risk and empirical risk respectively. $R_n(\Omega)$ represents the Rademacher complexity of the loss function class Ω. The SRM principle requires the data to be i.i.d. sampled from the original data distribution. However, the pairs sampled from available labeled instances in active hashing follow different distribution compared with the whole training set as the chosen data is usually more informative and representative. In order to extend the SRM principle in active deep hashing, we reformulate the risk bound inequality with z omitted and the detailed formulation is in the supplementary material:

$$\mathbb{E}(J) \leqslant (\mathbb{E}(J) - \mathbb{E}_M(J)) + \hat{\mathbb{E}}_M(J) + \Phi \tag{5}$$

where $\mathbb{E}_M(J)$ and $\hat{\mathbb{E}}_M(J)$ are the true risk and empirical risk of available labeled data which includes the labeled set and query set. $\Phi = 2R_c(\Omega) + \sqrt{\frac{\ln 1/\delta}{c}}$ means the model complexity, and c is the size of the available labeled training pairs. In active hashing, the data z consists of sample pairs $\boldsymbol{x} = (\boldsymbol{x}_a, \boldsymbol{x}_b)$ and their labels y , we can rewrite the first term of (5) as follows:

$$\begin{aligned}
&\mathbb{E}(J) - \mathbb{E}_M(J) = \int p(\boldsymbol{x}|\boldsymbol{x} \in \mathcal{X}) \int J \cdot p(y|\boldsymbol{x}) d\boldsymbol{x} dy - \\
&\int p(\boldsymbol{x}|\boldsymbol{x} \in \mathcal{M}) \int J \cdot p(y|\boldsymbol{x}) d\boldsymbol{x} dy \\
&= \int g(\boldsymbol{x}) p(\boldsymbol{x}|\boldsymbol{x} \in \mathcal{X}) d\boldsymbol{x} - \int g(\boldsymbol{x}) p(\boldsymbol{x}|\boldsymbol{x} \in \mathcal{M}) d\boldsymbol{x}
\end{aligned} \tag{6}$$

where $\mathcal{M}$ means the labeled set $\mathcal{L}$ and the query set $\mathcal{Q}$. $p(\boldsymbol{x}|\boldsymbol{x} \in \mathcal{X})$ and $p(\boldsymbol{x}|\boldsymbol{x} \in \mathcal{M})$ are the distribution of all training pairs and available labeled pairs respectively, and $p(y|\boldsymbol{x})$ is the probability of the pair $\boldsymbol{x}$ to be similar. As $g(\boldsymbol{x}) = \int J \cdot p(y|\boldsymbol{x}) dy$ is bounded and measurable, a bounded and continuous function $\hat{g}(\boldsymbol{x})$ can guarantee the boundness of (6):

$$\begin{aligned}
&\mathbb{E}(J) - \mathbb{E}_M(J) \leqslant \sup_{\hat{g} \in S} [\int g(\boldsymbol{x}) p(\mathcal{X}) d\boldsymbol{x} - \int g(\boldsymbol{x}) p(\mathcal{M}) d\boldsymbol{x}] \\
&= MMD_S(p(\mathcal{X}), p(\mathcal{M}))
\end{aligned} \tag{7}$$

where we rewrite $p(\boldsymbol{x}|\boldsymbol{x} \in \mathcal{X})$ as $p(\mathcal{X})$ and $p(\boldsymbol{x}|\boldsymbol{x} \in \mathcal{M})$ as $p(\mathcal{M})$ for simplicity. $MMD_S(p_1, p_2)$ represents the maximum mean discrepancy between distribution

p_1 and p_2, which is measured by functions from class S. Finally, the SRM principle can be directly employed in deep hashing with active pairwise supervision and rewritten as follows:

$$\mathbb{E}(J) \leqslant \hat{\mathbb{E}}_M(J) + \Phi + MMD_S(p(\mathcal{X}), p(\mathcal{M})) \tag{8}$$

Minimizing the upper bound in (8) can actively distinguish the sample pairs that provide effective supervision.

3.3 Designing the Acquisition Function via Structural Risk Minimization

We propose a batch mode active deep hashing algorithm by minimizing the structural risk bound illustrated in (8) with pairwise relationship. The query set is selected via the following optimization objectives:

$$\min_{\mathcal{Q},\mathcal{H}} \quad \frac{1}{l+q} \sum_{\boldsymbol{x} \in \mathcal{L} \cup \mathcal{Q}} J + \lambda ||\mathcal{H}||_F^2 + MMD_S[p(\mathcal{X}), p(\mathcal{L} \cup \mathcal{Q})]$$

where l and q are the size of the labeled set and the query set. $||\mathcal{H}||_F$ is the Frobenius norm of the weight matrix in deep hash model, which demonstrates the model complexity Φ [3] equally. In the above objective function, we denote the first two terms as L_1 which corresponds to the regularized empirical risk for all labeled sample pairs. Minimizing L_1 enforces the learned hash codes to be discriminative to learn the topology of semantic space of images in visual retrieval according to the supervision. The last term is notated as L_2, which relates to the generalization ability of the active deep hash model. Optimizing L_2 requires the distribution difference between labeled pairs and all pairs in the training set to be small, which encourages the labeled set to capture the representative information of the whole training set for enhanced generalization ability.

According to the definition of J presented in (3), we should minimize the worst-case regularized empirical risk as labels for sample pairs in the query set is unknown. We can write the worst-case regularized empirical risk explicitly as follows:

$$\min_{\mathcal{Q},\mathcal{H}} L_1 = \frac{1}{l+q} \sum_{\boldsymbol{x} \in \mathcal{L}} J + \lambda ||\mathcal{H}||_F^2 + \frac{1}{l+q} \sup_y \sum_{\boldsymbol{x} \in \mathcal{Q}} J \tag{9}$$

The label of pairs sampled from the query set with the worst-case risk is $y = -sign(\frac{m}{2} - ||f_a - f_b||_2^2)$. The first two terms in (9) train the hash model with pairwise supervision. The last term measures the similarity uncertainty of pairs sampled from the query set and contributes to the acquisition function, as hard pairs leading to high training loss should acquire label information to provide effective supervision.

The distribution difference between pairs sampled from the labeled and training sets is measured by their mean maximum discrepancy (MMD). The MMD

objective ensures the labeled sample pairs are similar to the overall sample pairs so that representative information of the training data is captured. The hash model $\mathcal{H}$ yields two binary vectors to represent the sample pair, and the distance between binary code pairs is defined as follows:

$$\begin{aligned} &d(\mathcal{H}(\boldsymbol{x}), \mathcal{H}(\boldsymbol{t})) = \inf_{k} \ ||\mathcal{H}(\boldsymbol{x}) - \mathcal{T}_k(\mathcal{H}(\boldsymbol{t}))||_F \\ &= \min(||\mathcal{H}(\boldsymbol{x}_a) - \mathcal{H}(\boldsymbol{t}_a)||_F + ||\mathcal{H}(\boldsymbol{x}_b) - \mathcal{H}(\boldsymbol{t}_b)||_F, \\ &||\mathcal{H}(\boldsymbol{x}_b) - \mathcal{H}(\boldsymbol{t}_a)||_F + ||\mathcal{H}(\boldsymbol{x}_a) - \mathcal{H}(\boldsymbol{t}_b)||_F) \end{aligned} \tag{10}$$

where $\mathcal{T}_k$ and $k \in \{0, 1\}$ is the permutation operator and indicator respectively. $\mathcal{T}_1(\mathcal{H}(\boldsymbol{t}))$ means to permute the pair $(\mathcal{H}(\boldsymbol{t}_a), \mathcal{H}(\boldsymbol{t}_b))$ to $(\mathcal{H}(\boldsymbol{t}_b), \mathcal{H}(\boldsymbol{t}_a))$ when calculating the Hamming distance with other pairs, and $\mathcal{T}_0(\mathcal{H}(\boldsymbol{t}))$ remains $\mathcal{H}(\boldsymbol{t})$ unchanged. When large distance is caused by the sampling order for semantically similar pairs, we permute the instances in pairs to obtain the real semantic distance. As proved in the supplementary materials, the defined distance is non-negative, symmetric and follows the triangle inequality. According to the distance definition in (10), we write the MMD objectives for active deep hashing in the following [5, 18]:

$$\inf_{\boldsymbol{k}_1, \boldsymbol{k}_2} \left\| \frac{1}{l+q} \sum_{i=1}^{l+q} \mathcal{T}_{k_{1,i}}(\mathcal{H}(\boldsymbol{x}_{1,i})) - \frac{1}{u-q} \sum_{i=1}^{u-q} \mathcal{T}_{k_{2,i}}(\mathcal{H}(\boldsymbol{x}_{2,i})) \right\|_F^2$$

where $\boldsymbol{x}_{1,i} \in \mathcal{L} \bigcup \mathcal{Q}$ is the i_{th} pair sampled from the labeled and query sets, and $\boldsymbol{x}_{2,i} \in \mathcal{U} \setminus \mathcal{Q}$ is the i_{th} pair sampled from the unlabeled set excluding query instances. $k_{1,i}$ and $k_{2,i}$ is the i_{th} element of the permutation indicator $\boldsymbol{k}_1 \in \{0, 1\}^{l+q}$ and $\boldsymbol{k}_2 \in \{0, 1\}^{u-q}$. Similar to [7], we transfer the MMD objective for active deep hashing as follows during the optimization process:

$$\begin{aligned} \min_{\boldsymbol{\alpha}} L_2 = \frac{1}{2} \boldsymbol{\alpha}^T \boldsymbol{K}_{UU} \boldsymbol{\alpha} + \frac{u-q}{n} \mathbf{1}^l \boldsymbol{K}_{LU} \boldsymbol{\alpha} - \frac{l+q}{n} \mathbf{1}^u \boldsymbol{K}_{UU} \boldsymbol{\alpha} \\ s.t. \quad \boldsymbol{\alpha} \in \{0, 1\}^u, ||\boldsymbol{\alpha}||_1 = q \end{aligned} \tag{11}$$

where the k_{th} element of $\boldsymbol{\alpha}$ is one if the k_{th} pair sampled from the unlabeled set is require to obtain annotation and otherwise equals to zero. $\mathbf{1}^d$ is an all one vector in d dimensions. $\boldsymbol{K}_{UU}$ illustrates the self-correlation matrix of pairs sampled from the unlabeled set, and $\boldsymbol{K}_{LU}$ demonstrates the correlation between pairs sampled from the labeled set and the unlabeled pool. We denote the element in the i_{th} row and j_{th} column of $\boldsymbol{K}_{UU}$ and $\boldsymbol{K}_{LU}$ as $K_{UU,ij}$ and $K_{LU,ij}$ and represent them as $K_{UU,ij} = \inf_k \mathcal{H}(\boldsymbol{x}_{U,i})^T \mathcal{T}_k(\mathcal{H}(\boldsymbol{x}_{U,j}))$ and $K_{LU,ij} = \inf_k \mathcal{H}(\boldsymbol{x}_{L,i})^T \mathcal{T}_k(\mathcal{H}(\boldsymbol{x}_{U,j}))$ respectively, where $\boldsymbol{x}_{U,i}$ and $\boldsymbol{x}_{L,i}$ are the i_{th} pair sampled from the unlabeled and labeled sets. The MMD objective contributes to the acquisition function. In (11), the first term aims to minimize the self-correlation of pairs sampled from the query set in a batch so that the Oracle provides more information, and the second term purposes to encourage pairs sampled from the query set to be different from those sampled from the labeled set so that the provided

supervision is not redundant. The above two terms increase the diversity of information with redundancy elimination in batch mode deep active hashing. The goal of the last term is to ensure the pairs sampled from the query set are comprehensively similar to all unlabeled ones as they are representative for the whole dataset.

Finally, we obtain different terms of the acquisition function in the proposed DH-APS method with respect to uncertainty, representativeness and diversity:

$$\begin{aligned} &Uncertainty: && s_1 = \frac{1}{l+q} \sup_y \sum \boldsymbol{J}\boldsymbol{\alpha} \\ &Representativeness: && s_2 = -\frac{l+q}{n}\mathbf{1}^u \boldsymbol{K}_{UU}\boldsymbol{\alpha} \\ &Diversity: && s_3 = \frac{1}{2}\boldsymbol{\alpha}^T \boldsymbol{K}_{UU}\boldsymbol{\alpha} + \frac{u-q}{n}\mathbf{1}^l \boldsymbol{K}_{LU}\boldsymbol{\alpha} \end{aligned}$$

where the i_{th} element of $\boldsymbol{J} \in \mathcal{R}^{1\times u}$ represents the training loss of the i_{th} pair sampled from the unlabeled set. As searching for the optimal $\boldsymbol{\alpha}$ is an NP-hard problem, we employ the alternating direction method of multipliers (ADMM) algorithm [6] to solve the following problem in active deep hashing:

$$\min_{\boldsymbol{\alpha}} s = s_1 + s_2 + s_3 \tag{12}$$

Because $\boldsymbol{\alpha}$ indicates the selection of pairs instead of instances, we rank all unlabeled samples based on the number of selected pairs containing them in a descent order. Then we add the top q instances to the query set before the annotation process. Labeling the selected samples provides effective supervision and enforces the hash model to learn discriminative and generalizable binary codes.

4 Experiments

In this section, we evaluated our method on three datasets for image retrieval: CIFAR-10, NUS-WIDE and ImageNet. We first introduce the implementation details, and then investigate the influence of the acquisition function by ablation study. Meanwhile, we compare the proposed DH-APS with the state-of-the-art hash codes to show the benefits of effective supervision from actively selected samples. Finally, we visualize the query set to demonstrate our intuition.

4.1 Datasets and Implementation Details

We first introduce the datasets our DH-APS carried out experiments on and corresponding data preprocessing techniques:

CIFAR-10: The CIFAR-10 dataset consists of $60,000$ images of size 32×32 and is categorized into 10 classes. We randomly selected 1,000 images (100 images per class) as the query set, the rest 59,000 images as the training set and the retrieval database. We padded 4 pixels on each side of the image and cropped it into the size of 32×32 randomly with normalization.

Table 1. Effect of different components in the acquisition function and the ratio of labeled samples on mean average precision (%), where Unc., Rep. and Div. represent uncertainty, representativeness and diversity respectively. The proposed DH-APS was evaluated with the 32-bit hash codes on CIFAR-10.

Unc.	Rep.	Div.	Ratio of labeled samples									
			0%	1%	5%	10%	15%	20%	30%	50%	80%	100%
√	√	√	18.1	32.6	41.0	49.5	54.2	57.3	63.5	64.7	65.4	66.1
		×	18.1	31.8	39.5	47.6	51.9	55.9	62.9	64.1	65.1	66.1
	×	√	18.1	31.5	38.9	48.0	52.1	55.2	61.8	64.3	65.0	66.1
		×	18.1	28.7	36.6	46.0	49.3	53.1	59.5	63.8	64.8	66.1
×	√	√	18.1	29.0	36.4	44.3	49.2	52.5	58.8	63.6	64.7	66.1
		×	18.1	26.8	34.9	41.0	45.5	49.6	56.0	62.4	64.2	66.1
	×	√	18.1	26.4	35.4	41.2	45.3	48.9	54.7	62.3	64.4	66.1
		×	18.1	25.5	33.7	38.1	44.6	48.1	54.3	62.0	63.9	66.1

NUS-WIDE: The NUS-WIDE dataset contains 269,648 images collected from Flicker with 81 manually annotated concepts. Two images are regarded as positive if they share at least one positive label and are negative otherwise. Only the 21 most frequent concepts were used, resulting in a total of 166,047 images. We randomly chose 2,100 images (100 images per class) as the query set and regarded the rest as the training set and the retrieval database. The images were warped to 64×64 before feeding forward to the networks and normalized.

ImageNet-100: ImageNet (ILSVRC12) contains approximately 1.2 million training and 50K validation images from 1,000 categories. ImageNet is much more challenging because of its large scale and high diversity. Images of 100 randomly sampled categories were selected to construct the training set, and we applied all images in the selected classes from the validation set as queries. Followed by data augmentation of bias subtraction applied in CIFAR-10, a 224×224 region was randomly cropped for training from the resized image whose shorter side was 256. For inference, we employed the 224×224 center crop.

We trained our DH-APS with VGG16-like [43] architectures, where the softmax layer in the original VGG16 was replaced with a fully-connected layer to obtain the binary codes. In each iteration, we trained the hash model for 60 epochs, selected the samples for annotation and labeled the query samples by an Oracle until reaching the annotation cost limit. For hash model training, the SGD optimizer with the momentum of 0.9 and weight decay of 0 was leveraged. The learning rate started from 0.01 and changed to $1e^{-3}$ and $1e^{-4}$ at the 20_{th} and 40_{th} epoch. For sample selection, we randomly sampled $\frac{\eta n^2}{100}$ pairs from the unlabeled instances to construct the unlabeled pairs and then actively selected $\frac{\eta n^2}{1000}$ pairs by solving (12) via ADMM, where η is the assigned ratio of labeled data representing the annotation budget and n is the size of the dataset. We ranked all unlabeled samples based on the number of selected pairs containing them and added the top $\frac{\eta n}{1000}$ samples to the query set.

4.2 Ablation Study

As DH-APS selects samples that provide effective pairwise supervision for annotation, the learned binary codes are enhanced significantly with discriminative

information and strong generalization ability. To verify the importance of the proposed acquisition function and supervision, we implemented our DH-APS with different annotation budget and utilization of various terms in the acquisition function. Because the uncertainty, representativeness and diversity terms in the acquisition function contribute differently to sample selection, we conducted orthogonal ablation study w.r.t. them. We adopted VGG16-like architecture as the deep hash model, which was evaluated on the CIFAR-10 dataset. Mean Average Precision (MAP) in 32-bit binary codes was reported in Table 1. Based on the results, we observe the influence of different component in the proposed acquisition function and the ratio of labeled samples.

- Comparing the accuracy obtained with different ratio of labeled samples, we know that annotating a small quantity of informative samples improves the MAP of retrieval very significantly. The effective pairwise supervision benefits the hash code learning obviously especially when extremely little annotation budget is accessible for training. Although the ratio of labeled samples is positively related to the performance, the margin of the MAP enhancement caused by the extra annotation declines for the large labeled set, which means most samples fail to provide effective supervision and do not contribute to deep hash code learning. Our DH-APS achieves competitive accuracy compared with the fully supervised deep hashing methods by utilizing only 30% labeled data for training, which significantly saves the labeling cost.
- All components in the acquisition function including uncertainty, representativeness and diversity improve the MAP at various degrees, which implies the DH-APS method are universally suitable for various deep hash model. The uncertainty enhances the binary codes most significantly, because hard pairs with the most uncertain similarity provide large gradients in the back-propagation so that the deep hash model is supervised effectively. Because the representative samples capture the global topology of the semantic space and the diverse samples eliminate information redundancy in supervision, combining all components in the acquisition function further increases the MAP.

4.3 Comparison with the State-of-the-art Methods

In this section, we compare the performance of our DH-APS with existing unsupervised methods DH [12] and GraphBit [11], semi-supervised methods SSH [47], SSDH [51] and PTS^{3}H [52] and supervised methods DSH [29], DPSH [26] and SDSH [35] in image retrieval tasks on the CIFAR-10, NUS-WIDE and ImageNet datasets, and the applied backbone of the above methods were all VGG16 in our comparison. Table 2 illustrates the MAP of different methods in various code lengths, where SSH† means that we reimplemented the method with deep hash models. DH-APS (1%, 10%, 30%) represents the proposed active deep hashing method with corresponding ratio of labeled samples. We also implemented our DH-APS with the same annotation setting as semi-supervised methods [51, 52, 47] which is denoted as DH-APS (*). For DH-APS (*), we randomly selected

Table 2. Comparison of mean average precision (%)with state-of-the-art unsupervised, semi-supervised and supervised deep binary descriptors under varying code lengths. 12b, 24b, 32b and 48b means the hash codes in 12, 24, 36 and 48 bits. SSH† means that we integrate the method with deep hash models. DH-APS (1%, 10%, 30%) stands for our method with different ratio of labeled samples, and DH-APS ($*$) means that we adopt the same annotation setting as semi-supervised hashing methods. DH-APS+PTS^{3}H represents the presented DH-APS combined with PTS^{3}H.

Methods	CIFAR-10				NUS-WIDE				ImageNet-100			
	12b	24b	32b	48b	12b	24b	32b	48b	12b	24b	32b	48b
Unsupervised Hashing												
DH	22.3	23.0	23.6	23.7	22.5	23.1	23.4	23.3	12.5	13.8	14.0	14.2
GraphBit	26.9	27.2	27.0	27.3	26.7	27.0	27.2	27.4	12.9	14.5	14.7	15.1
Semi-supervised Hashing												
SSH†	35.3	37.0	38.1	38.2	30.0	31.6	35.8	32.6	19.9	21.0	21.6	23.1
SSDH	80.1	81.3	81.2	81.4	77.3	77.9	77.8	77.8	–	–	–	–
PTS^{3}H	79.8	82.8	83.5	84.3	75.2	77.4	78.3	78.9	66.1	67.5	68.0	69.7
Supervised Hashing												
DSH	61.6	65.6	66.1	67.3	54.5	55.3	55.9	56.0	47.9	50.3	50.7	51.4
DPSH	71.3	72.7	74.4	75.7	79.4	82.2	83.8	85.1	–	–	–	–
SDSH	93.9	93.9	93.9	93.4	–	81.7	82.1	82.1	–	–	–	–
Active Hashing												
DH-APS (1%)	30.5	31.9	32.6	32.8	30.1	30.6	31.2	31.8	17.9	18.1	19.5	19.6
DH-APS ($*$)	44.9	46.4	47.8	47.7	36.0	36.8	38.5	38.8	24.9	25.1	26.3	26.8
DH-APS (10%)	47.2	48.6	49.5	49.7	38.1	39.6	40.2	40.7	26.1	27.3	27.8	28.0
DH-APS (30%)	61.8	62.4	63.5	64.3	51.8	53.0	53.5	54.3	43.5	43.6	45.2	46.9
DH-APS+PTS^{3}H	82.1	85.3	86.7	86.9	79.1	81.1	82.2	82.3	68.9	70.0	70.3	71.8

500, 500 and 100 images of each class on CIFAR-10, NUS-WIDE and ImageNet-100 for labeling respectively. DH-APS+PTS^{3}H represent DH-APS deployed as a plug-and-play module in PTS^{3}H for labeled instance sampling. Table 2 illustrates the MAP of different hash methods. Fig. 2 depicts the precision within Hamming 2 for 12, 24, 32 and 48-bit hash codes and w.r.t. top-k in 48-bit binary representations on the three datasets. The performance of DH, GraphBit, SSH†, PTS^{3}H and DSH was obtained by rerunning the codes and the results of other baselines were copied from the referenced paper.

The results indicate DH-APS achieves the competitive accuracy with the supervised method DSH with only 30% labeled samples on both the CIFAR-10 and NUS-WIDE datasets, and the underperformance on ImageNet-100 is caused by the rich information of the dataset where the global structure of the semantic space is difficult to represent by few samples. Meanwhile, DH-APS enhances the semi-supervised method PTS^{3}H significantly due to the effective supervision from actively selected samples. Compared with unsupervised methods, DH-APS outperforms GraphBit across all datasets with only 1% data labeled. With better discriminality and generalization ability, DH-APS mines the global structure of the semantic space with few labeled samples.

4.4 Visualization

In order to demonstrate the intuition of the proposed method, we provide the visualization of DH-APS. We trained an active deep hash model with the LeNet5

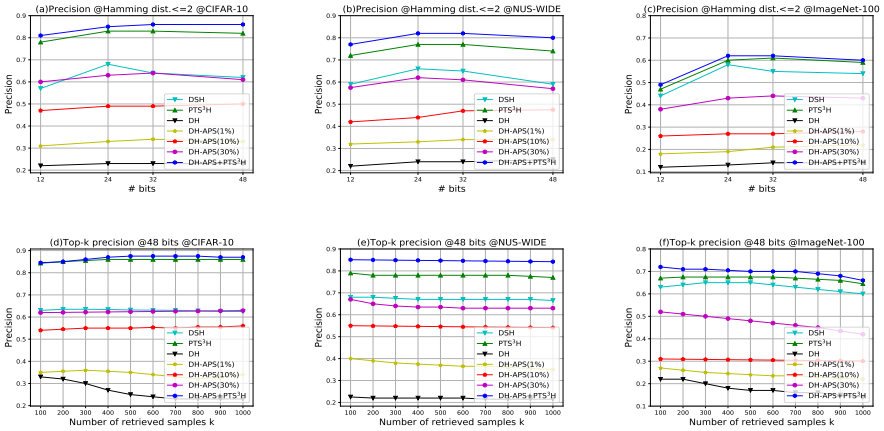

Fig. 2. The performance on image retrieval of different binary codes.

architecture on the MNIST dataset [25] through our method. The MNIST dataset consists of $60,000$ digit images with the size 28×28, which are divided into 10 categories. We scaled and biased all images into the range $[-0.5, 0.5]$. We randomly sampled $2,000$ images with 200 samples for each class to construct the training set and selected 10 images to annotate according to the acquisition function only consisting of uncertainty, representativeness and diversity terms respectively. Figure 3 (a) shows the 2-d projection of the samples via the t-SNE method, where dots in different colors represent various digits and dots with different borders stand for the selected instances based on various acquisition function. Fig. 3 (b) demonstrate the selected images according to acquisition function composed by uncertainty, representativeness and diversity terms respectively.

When only applying the uncertainty term, the ambiguous samples which remain far from the original distribution are selected. As these hard instances provide large gradient for the hash model learning, the discriminality of the learned hash codes is enhanced. Selecting samples for annotation based on the representativeness term encourages samples near the center of different classes to be labeled. The representative instances capture the semantic information of the whole dataset instead of local structure, so that the learned hash model can be generalized to image retrieval in large scale. When the diversity term is employed as the acquisition function, samples in different classes are chosen evenly for annotation. The diverse instances offer effective supervision without redundancy to fully utilize the representation capacity of the binary codes.

5 Conclusion

In this paper, we have proposed an deep hashing method with active pairwise supervision called DH-APS for large scale image search. The proposed DH-APS

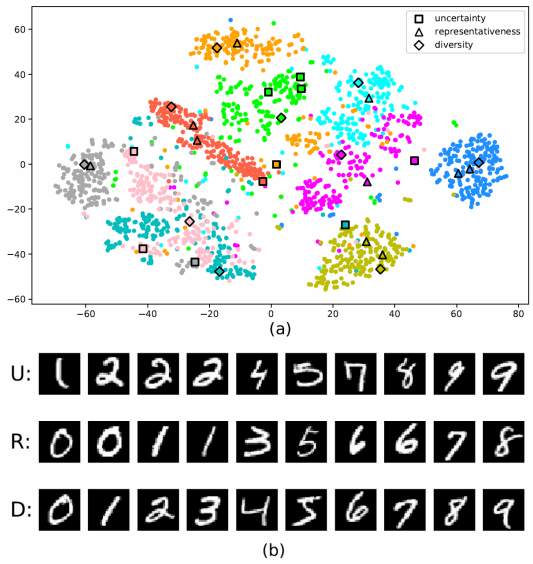

Fig. 3. Visualization of DH-APS. (a) The 2-d projection of the samples via the t-SNE method, where dots in different colors represent various digits. Dots with square, triangle and rhombus borders mean the selected instances based on acquisition function only containing uncertainty, representativeness and diversity terms respectively. (b) Selected images according to acquisition function only composed of uncertainty (U), representativeness (R) and diversity (D) respectively (best viewed in color).

actively selects a small quantity of samples for annotation via considering pairwise relationship and generalizing the structural risk minimization principle, so that uncertain, representative and diverse samples are labeled. The effective supervision significantly enhances the discriminality and generalization ability of the learned hash codes with limited annotation cost. Extensive experiments have demonstrated the effectiveness of the proposed approach.

Acknowledgement

This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFA0700802, in part by the National Natural Science Foundation of China under Grant 61822603, Grant U1813218, Grant U1713214, and Grant 61672306, in part by Beijing Natural Science Foundation under Grant No. L172051, in part by Beijing Academy of Artificial Intelligence (BAAI), in part by a grant from the Institute for Guo Qiang, Tsinghua University, in part by the Shenzhen Fundamental Research Fund (Subject Arrangement) under Grant JCYJ20170412170602564, and in part by Tsinghua University Initiative Scientific Research Program.

References

1. Babenko, A., Lempitsky, V.: Aggregating local deep features for image retrieval. In: ICCV. pp. 1269–1277 (2015)
2. Balcan, M.F., Broder, A., Zhang, T.: Margin based active learning. In: COLT. pp. 35–50 (2007)
3. Bartlett, P.L., Mendelson, S.: Rademacher and gaussian complexities: Risk bounds and structural results. JMLR **3**(Nov), 463–482 (2002)
4. Beluch, W.H., Genewein, T., Nürnberger, A., Köhler, J.M.: The power of ensembles for active learning in image classification. In: CVPR. pp. 9368–9377 (2018)
5. Borgwardt, K.M., Gretton, A., Rasch, M.J., Kriegel, H.P., Schölkopf, B., Smola, A.J.: Integrating structured biological data by kernel maximum mean discrepancy. Bioinformatics **22**(14), 49–57 (2006)
6. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J., et al.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends® in Machine learning **3**(1), 1–122 (2011)
7. Chattopadhyay, R., Wang, Z., Fan, W., Davidson, I., Panchanathan, S., Ye, J.: Batch mode active sampling based on marginal probability distribution matching. TKDD **7**(3), 13 (2013)
8. Chua, T.S., Tang, J., Hong, R., Li, H., Luo, Z., Zheng, Y.: 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. p. 48 (2009)
9. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR. pp. 248–255 (2009)
10. Duan, Y., Lu, J., Wang, Z., Feng, J., Zhou, J.: Learning deep binary descriptor with multi-quantization. In: CVPR. pp. 1183–1192 (2017)
11. Duan, Y., Wang, Z., Lu, J., Lin, X., Zhou, J.: Graphbit: Bitwise interaction mining via deep reinforcement learning. In: CVPR. pp. 8270–8279 (2018)
12. Erin Liong, V., Lu, J., Wang, G., Moulin, P., Zhou, J.: Deep hashing for compact binary codes learning. In: CVPR. pp. 2475–2483 (2015)
13. Freytag, A., Rodner, E., Denzler, J.: Selecting influential examples: Active learning with expected model output changes. In: ECCV. pp. 562–577 (2014)
14. Gal, Y., Islam, R., Ghahramani, Z.: Deep bayesian active learning with image data. In: ICML. pp. 1183–1192 (2017)
15. Ghasedi Dizaji, K., Zheng, F., Sadoughi, N., Yang, Y., Deng, C., Huang, H.: Unsupervised deep generative adversarial hashing network. In: CVPR. pp. 3664–3673 (2018)
16. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: NIPS. pp. 2672–2680 (2014)
17. Gordo, A., Almazán, J., Revaud, J., Larlus, D.: Deep image retrieval: Learning global representations for image search. In: ECCV. pp. 241–257 (2016)
18. Gretton, A., Borgwardt, K.M., Rasch, M.J., Schölkopf, B., Smola, A.: A kernel two-sample test. JMLR **13**(Mar), 723–773 (2012)
19. Hasan, M., Roy-Chowdhury, A.K.: Context aware active learning of activity recognition models. In: ICCV. pp. 4543–4551 (2015)
20. Jegou, H., Douze, M., Schmid, C.: Product quantization for nearest neighbor search. TPAMI **33**(1), 117–128 (2010)
21. Johnson, J., Krishna, R., Stark, M., Li, L.J., Shamma, D., Bernstein, M., Fei-Fei, L.: Image retrieval using scene graphs. In: CVPR. pp. 3668–3678 (2015)

22. Joshi, A.J., Porikli, F., Papanikolopoulos, N.: Multi-class active learning for image classification. In: CVPR. pp. 2372–2379 (2009)
23. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images. Tech. rep. (2009)
24. Lai, H., Pan, Y., Liu, Y., Yan, S.: Simultaneous feature learning and hash coding with deep neural networks. In: CVPR. pp. 3270–3278 (2015)
25. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., et al.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998)
26. Li, W.J., Wang, S., Kang, W.C.: Feature learning based deep supervised hashing with pairwise labels. In: IJCAI. pp. 1711–1717 (2016)
27. Li, X., Guo, Y.: Multi-level adaptive active learning for scene classification. In: ECCV. pp. 234–249 (2014)
28. Liu, B., Ferrari, V.: Active learning for human pose estimation. In: ICCV. pp. 4363–4372 (2017)
29. Liu, H., Wang, R., Shan, S., Chen, X.: Deep supervised hashing for fast image retrieval. In: CVPR. pp. 2064–2072 (2016)
30. Luo, W., Schwing, A., Urtasun, R.: Latent structured active learning. In: NIPS. pp. 728–736 (2013)
31. Mac Aodha, O., Campbell, N.D., Kautz, J., Brostow, G.J.: Hierarchical subquery evaluation for active learning on a graph. In: CVPR. pp. 564–571 (2014)
32. Melville, P., Mooney, R.J.: Diverse ensembles for active learning. In: ICML. p. 74 (2004)
33. Nguyen, H.T., Smeulders, A.: Active learning using pre-clustering. In: ICML. p. 79 (2004)
34. Paul, S., Bappy, J.H., Roy-Chowdhury, A.K.: Non-uniform subset selection for active learning in structured data. In: CVPR. pp. 6846–6855 (2017)
35. Pidhorskyi, S., Jones, Q., Motiian, S., Adjeroh, D., Doretto, G.: Deep supervised hashing with spherical embedding. In: ACCV. pp. 417–434 (2018)
36. Qin, D., Gammeter, S., Bossard, L., Quack, T., Van Gool, L.: Hello neighbor: Accurate object retrieval with k-reciprocal nearest neighbors. In: CVPR. pp. 777–784 (2011)
37. Rényi, A., et al.: On measures of entropy and information. In: Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics (1961)
38. Sener, O., Savarese, S.: Active learning for convolutional neural networks: A core-set approach. arXiv preprint arXiv:1708.00489 (2017)
39. Settles, B., Craven, M.: An analysis of active learning strategies for sequence labeling tasks. In: EMNLP. pp. 1070–1079 (2008)
40. Settles, B., Craven, M., Ray, S.: Multiple-instance active learning. In: NIPS. pp. 1289–1296 (2008)
41. Shen, F., Shen, C., Liu, W., Tao Shen, H.: Supervised discrete hashing. In: CVPR. pp. 37–45 (2015)
42. Shen, F., Xu, Y., Liu, L., Yang, Y., Huang, Z., Shen, H.T.: Unsupervised deep hashing with similarity-adaptive and discrete optimization. TPAMI **40**(12), 3034–3044 (2018)
43. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
44. Vasisht, D., Damianou, A., Varma, M., Kapoor, A.: Active learning for sparse bayesian multilabel classification. In: KDD. pp. 472–481 (2014)
45. Vijayanarasimhan, S., Grauman, K.: Large-scale live active learning: Training object detectors with crawled data and crowds. IJCV **108**(1-2), 97–114 (2014)

46. Wang, G., Hu, Q., Cheng, J., Hou, Z.: Semi-supervised generative adversarial hashing for image retrieval. In: ECCV. pp. 469–485 (2018)
47. Wang, J., Kumar, S., Chang, S.F.: Semi-supervised hashing for large-scale search. TPAMI **34**(12), 2393–2406 (2012)
48. Wang, Q., Si, L., Zhang, Z., Zhang, N.: Active hashing with joint data example and tag selection. In: SIGIR. pp. 405–414 (2014)
49. Wang, X., Yang, M., Cour, T., Zhu, S., Yu, K., Han, T.X.: Contextual weighting for vocabulary tree based image retrieval. In: ICCV. pp. 209–216 (2011)
50. Yang, H.F., Lin, K., Chen, C.S.: Supervised learning of semantics-preserving hash via deep convolutional neural networks. TPAMI **40**(2), 437–451 (2017)
51. Zhang, J., Peng, Y.: Ssdh: semi-supervised deep hashing for large scale image retrieval. TCSVT **29**(1), 212–225 (2017)
52. Zhang, S., Li, J., Zhang, B.: Pairwise teacher-student network for semi-supervised hashing. In: CVPR. pp. 0–0 (2019)
53. Zhao, F., Huang, Y., Wang, L., Tan, T.: Deep semantic ranking based hashing for multi-label image retrieval. In: CVPR. pp. 1556–1564 (2015)
54. Zhen, Y., Yeung, D.Y.: Active hashing and its application to image and text retrieval. Data Mining and Knowledge Discovery **26**(2), 255–274 (2013)