Semi-supervised Domain Adaptive Retrieval via Discriminative Hashing Learning

Haifeng Xia†, Taotao Jing†, Chen Chen‡, Zhengming Ding†
†Department of Computer Science, Tulane University, USA
‡Center for Research in Computer Vision, University of Central Florida, USA
{hxia, tjing, zding1}@tulane.edu
chen.chen@crcv.ucf.edu

ABSTRACT
Domain adaptive image retrieval (DAR) aims to train the model with well-labeled source domain and target images in order to retrieve source instances given query target samples from the identical category space. However, the practical scenario hinders to manually annotate all retrieved images due to huge labeling cost. Motivated by the realistic demand, we firstly define the semi-supervised domain adaptive retrieval (SDAR) problem, assuming the database includes a small proportion annotated source images and abundant unlabeled ones. To overcome the challenging SDAR, this paper propose a novel method named Discriminative Hashing learning (DHLing) which mainly includes two modules, i.e., domain-specific optimization and domain-invariant memory bank. Specifically, the first component explores the structural knowledge of samples to predict the unlabeled images with pseudo labels to achieve hash coding consistency. While, the second one attempts to construct the domain-invariant memory bank to guide the feature generation and achieve cross-domain alignment. Experimental results on several popular cross-domain retrieval benchmarks illustrate the effectiveness of our proposed DHLing on both conventional DAR and new SDAR scenarios by comparing with the state-of-the-art retrieval methods.

CCS CONCEPTS
• Information systems → Image search; • Theory of computation → Semi-supervised learning.

KEYWORDS
Semi-supervised Learning, Cross-domain Retrieval

ACM Reference Format:

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.

MM ’21, October 20–24, 2021, Virtual Event, China
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-8651-7/21/10 ...$15.00
https://doi.org/10.1145/3474085.3475526

1 INTRODUCTION
Image retrieval recently attracts more attention in multimedia community [11, 47]. For the online shopping scenario, customers always upload their interested images and query the corresponding product from platform by matching it with ones in the retrieval database. Existing researches [7, 52] typically assume the query images from customer belong to the identical distribution with the retrieved images. However, abundant environmental factors such as illumination and background easily result in the significant difference (domain shift) among images with the same semantic information [19, 44]. Under such a practical application, image retrieval problem becomes challenging for strategies [10, 36] that rely on the identical or similar distribution assumption to accurately provide clients the product image of interest. To overcome the challenge, [13] attempts to eliminate the domain discrepancy by combining the attribute information with visual semantic similarity. Although this scheme effectively improves the retrieval precision, the adoption of attribution introduces additional computational cost.

To overcome the bottleneck, [12] introduces the domain adaptive retrieval (DAR) by considering the well-labeled retrieval database as source domain. In addition, DAR has access to additional unlabeled images of target domain in the training stage, as Figure 1 (a) shows. It is worth noting that target images and the unseen queries are sampled from the same distribution. Under this condition, DAR exploits the supervision of source annotation to train a
model with more discriminative ability and then adapts the model to target domain by mitigating the cross-domain discrepancy. Finally, DAR matches the given queries with the available source images, resulting in state-of-the-art performance in cross-domain image retrieval. However, annotating the whole retrieval database becomes time-consuming and expensive for realistic retrieval applications.

Moreover, to accelerate the retrieval speed and reduce the storage space, many methods [8, 23, 26, 31] explore the hash coding to transform the continuous real-value feature into the binary code and calculate the Hamming distance of arbitrary two instances as their similarity [5, 24, 33, 40, 49]. Since the hash operation removes abundant semantic information from the original representation, the binary code mechanism slightly reduces the negative influence of domain shift and promotes the retrieval ability of model.

In this paper, we consider a novel application scenario named as Semi-supervised Domain Adaptive Retrieval (SDAR) illustrated as Figure 1 (b), where insufficient source annotation heavily increases the difficulty of learning discriminative feature and achieving domain adaptation. Consequently, to address SDAR challenge, we develop the discriminative hashing learning (DHLing) consisting of two important modules, i.e., domain-specific optimization and domain-invariant memory bank. The first component of DHLing mainly remedies the effect of absent annotation and facilitates the model to learn discriminative representations. Concretely, we employ the labeled source images to learn the initial class centers and adaptively update them to form domain-specific centers for unlabeled source and target instances. According to the spherical k-means [2], the unlabeled training samples are automatically assigned their labels by measuring the distance to the class centers. Benefiting from the pseudo label, DHLing constructs the semantic similarity and learns the consistent binary codes for semantically similar samples by achieving the agreement between semantic similarity and the inner product of hash codes. However, hash code easily neglects the meaningful semantic so that it becomes sensitive to the distortion over the original hidden feature. Thus, the second module aims to early achieve cross-domain feature alignment before the hashing operation to improve the robustness of model. With the latent features, we firstly adopt dictionary learning [34, 50] to construct the domain-specific memory banks and then fuse them to seek domain-invariant memory bank which guides next feature generation to eliminate the cross-domain discrepancy. The main contributions of our work are summarized as three folds:

- From the practical application demand, we introduce a novel image retrieval problem named Semi-supervised Domain Adaptive Retrieval (SDAR), and propose a novel discriminative hashing learning (DHLing) with two modules for effective cross-domain image retrieval.
- To mitigate the label insufficiency, the first domain-specific optimization module explores the structural knowledge of samples to automatically annotate the unlabeled instances and further achieve coding consistency. To reduce the domain shift, the second one based on dictionary learning seeks the domain-invariant memory bank to align cross-domain feature distributions.
- Extensive experimental results on several cross-domain benchmarks fully verify the effectiveness of our DHLing on solving DAR and SDAR problems. The quantitative and qualitative analysis explicitly illustrates the contribution of each module on improving the retrieval ability of our DHLing model.

2 RELATED WORK

2.1 Single-domain Retrieval

The single-domain image retrieval refers to the content-based visual information matching and assumes all images belonging to the identical distribution [1, 43, 47]. Given the query images, the system searches the semantically similar ones from the database by calculating the similarity between the query and each retrieved image [25, 32, 35, 39]. To improve the retrieval efficacy, methods [15, 31, 46] typically adopt hash algorithm to obtain the binary code from the continuous real-value feature. Although they have achieved the comparable performance on standard benchmarks, their application easily suffers from retrieval precision degradation in the practical scenario where there exists significant difference between the query images and the retrieved ones due to various capturing conditions (illumination, background). The realistic demand naturally stimulates the exploration of domain adaptive retrieval.

2.2 Domain Adaptive Retrieval

The earlier explorations on cross-domain retrieval [13, 15, 17, 22, 51] constrain the application scenario on online shopping and organize the training set with paired user-product images. In addition, they also need the additional attribute information of instances to promote the retrieval precision. However, it is expensive to collect the cross-domain paired images in reality so that such an assumption heavily impedes the application of these methods. Thus, [12] further modifies the cross-domain retrieval as domain adaptive retrieval (DAR), where one can train the model with well-labeled source domain and unlabeled target images in order to retrieve source instances given the query target samples. These two domains share the identical category space but belong to different distributions [45, 48]. Although [12] has achieved the state-of-the-art performance, it manually annotates abundant source images to train the model, which becomes time-consuming and not affordable.

Furthermore, we are first to consider the semi-supervised domain adaptive retrieval (SDAR), where the source domain includes a small proportion of annotated images and other unlabeled ones. Due to the absence of many source annotation, SDAR significantly increases the difficulty of learning discriminative feature and eliminating domain shift.

3 THE PROPOSED METHOD

3.1 Motivation and Problem Definition

The typical domain adaptive retrieval (DAR) [12] aims to train the system with well-labeled retrieval database (source domain) and unlabeled target domain. Due to the domain shift caused by varying light conditions, occlusions, and devices, the mentioned two domains belong to different distributions. For the application, the trained model selects instances from source database according to the unseen queries following the same distributions as the target domain. However, the collection of annotated source samples generally becomes time-consuming and expensive. To meet
the more realistic demand, we propose a novel problem named Semi-supervised Domain Adaptive Retrieval (SDAR) where only a small portion of source samples are provided with the corresponding category information. The main challenges of SDAR mainly include two folds. Parallel to conventional DAR, the first important challenge of SDAR is how to mitigate the negative influence of cross-domain discrepancy on learning semantic-wise representations. On the other hand, how to learn discriminative features with insufficient source label knowledge is another difficulty of the new problem setting, through the comparison between DAR and SDAR.

Formally, SDAR assumes that the retrieved database consists of two parts: a small portion of well-labeled source instances \( \mathcal{D}_t^l = \{ (x_i^l, y_i) \mid x_i^l \in \mathbb{R}^d, y_i \in \mathbb{R}^c \}^{n_l}_{i=1} \) and a large amount of unlabeled ones \( \mathcal{D}_u^l = \{ x_i^u \mid x_i^u \in \mathbb{R}^d \}^{n_u}_{i=1} \) where \( n \) denotes the number of categories and \( y_i \) is the annotation corresponding to \( x_i \). In addition, the training stage has access to the target data samples \( \mathcal{D}_t = \{ x_i, y_i \mid x_i \in \mathbb{R}^d \}^{n_t}_{i=1} \) from the same distribution with the unseen queries. In terms of the final application, we explore the retrieval system to search samples from \( \mathcal{D}_t^l \) and \( \mathcal{D}_u^l \) with the similar semantic information as the given queries.

Moreover, inspired by the success of memory bank, DHLing adopts dictionary structure to construct the memory bank which facilitates the domain-invariant feature learning. The specific operations for each component of DHLing are elaborated as follows.

### 3.3 Domain-Specific Optimization

Before introducing the details of each module, we briefly illustrate the network architecture of DHLing. For simplicity, the symbols of domain and class are neglected. Given the instances \( x_i \) flowing in the network, it is straightforward to obtain the hidden features of each layer as follows:

\[
\begin{align*}
    f_i &= \sigma(W_f^T x_i + b), \\
    g_i &= \sigma(W_g^T f_i), \\
    h_i &= W_h^T g_i, \\
    \hat{y}_i &= W_c^T h_i,
\end{align*}
\]

where \( W_f, W_g, W_h, W_c \) and \( b \) are the learnable parameters of network, \( \sigma(\cdot) \) denotes the activation function and \( \hat{y}_i \) is the prediction only used by labeled source instances. Moreover, the binary hash representation is directly transformed from \( h_i \), i.e., \( b_i = \text{sign}(h_i) \in \mathbb{R}^K \) (\( K \) is the length of the hash code).

**Consistent Coding Optimization.** The primary goal of this module is to achieve coding consistency for semantically similar samples via the exploration of sample-wise association. To explicitly describe the relationship of samples, we firstly define the semantic similarity \( s_{ij} \) between \( x_i \) and \( x_j \), where \( s_{ij} = 1 \) means these two samples have the similar semantic knowledge, otherwise \( s_{ij} = 0 \). Intuitively, two instances are annotated with the same label, which is also equivalent to \( s_{ij} = 1 \).

From another perspective, SDAR mainly attempts to learn the unified binary code for source and target samples. For the hash coding, the Hamming distance defined as \( d(x_i, x_j) = \frac{1}{2} (K - (b_i \cdot b_j)) \) is usually used to measure the distance between two instances, where \( (b_i, b_j) \) is the inner product. Based on the above definition, the similarity of two samples can also be evaluated by their inner product over the hash codes, i.e., \( (b_i, b_j) \). Therefore, it is clear that
the larger inner product of two binary codes indicates they have the similar semantic information with a higher probability, formulated as Gaussian distributions:

\[ p(s_{ij}|b_i, b_j) = \begin{cases} 
\delta(\phi_{ij}), & s_{ij} = 1 \\
1 - \delta(\phi_{ij}), & s_{ij} = 0 
\end{cases} \tag{2} \]

where \( \phi_{ij} = \frac{1}{2}(b_i \cdot b_j) \) and \( \delta(\phi_{ij}) = \frac{1}{1 + \exp(-\phi_{ij})} \). With the supervision of semantic similarity \( s_{ij} \), we control the consistency of binary codes by adjusting their inner product. Thus, the loss function of coding consistency is written as:

\[ \min L_{ce} = -s_{ij} \log(\delta(\phi_{ij})) - (1 - s_{ij}) \log(1 - \delta(\phi_{ij})). \tag{3} \]

Along with the optimization of Eq. (3), the system gradually promotes the consistency between \( \delta(\phi_{ij}) \) and \( s_{ij} \). However, the insufficient labeled source instances in SDAR heavily affect the employment of coding consistency loss. Thereby, our model further develops domain-specific clustering to mitigate the contradiction.

**Domain-Specific Clustering.** The intuitive strategy is generating the pseudo label from the hidden features for unlabeled source/target samples to effectively avoid the absence of annotation. However, the derived annotations from the latent layer are also employed to optimize the feature learning. Considering the coupled relationship, we conduct these two operations in an iterative fashion. For each epoch, the fixed network takes all available samples as input to synthesize the hidden features \( h \) used to infer the label with the spherical k-means [2]. Then the instances with ground-truth or pseudo labels take part in the optimization of model.

In details, the generation of annotation consists of three steps. First, the features \( h^s \) of labeled source instances are used to compute each class center with:

\[ O^c_s = \frac{1}{N_c} \sum_{n=1}^{n^s} I(y_{s_n} = c) \frac{h^s_{s_n}}{||h^s_{s_n}||}. \tag{4} \]

where \( O^c_s \) and \( N_c \) denote the feature center and the number of samples of the \( c \)-th category respectively, and \( I(\cdot) \) is the indicator function. Second, we assign annotation to unlabeled source samples

\[ \hat{s}_{ij} = \arg \min_c \delta(h^u_{ij}, O^c_s) = \frac{1}{2} \left(1 - \frac{\langle h^u_{ij}, O^c_s \rangle}{||h^u_{ij}|| \cdot ||O^c_s||} \right). \tag{5} \]

To improve the annotation accuracy, we update the initial class centers via the following formulation and reassign the label to \( x_{ij}^t \):

\[ O^c_t = \frac{1}{N_t} \sum_{n=1}^{n^t} I(y_{t_n} = c) \frac{h^t_{t_n}}{||h^t_{t_n}||} + \frac{1}{N_c} \sum_{n=1}^{n^s} I(\hat{s}_{ij} = c) \frac{h^s_{s_n}}{||h^s_{s_n}||}. \tag{6} \]

Similarly, in the third step, the target centers are initialized with \( O^c_s \) and then we iteratively allocate labels to target samples and update the class centers with:

\[ O^c_s = \frac{1}{N_c} \sum_{n=1}^{n^s} I(\hat{s}_{ij} = c) \frac{h^s_{s_n}}{||h^s_{s_n}||}. \tag{7} \]

To this end, we easily determine the semantic similarity according to their assigned category information. Thanks to the pseudo label, the cross-domain coding consistency is easily formulated as:

\[ \min L_{cc} = -\hat{s}_{ij} \log(\delta(\phi_{ij}^t)) - (1 - \hat{s}_{ij}) \log((1 - \delta(\phi_{ij}^t))). \tag{8} \]

where \( \hat{s}_{ij} = 1 \) when \( x_{ij}^t \) and \( x_t \) belong to the same category, otherwise \( \hat{s}_{ij} = 0 \), and \( \delta^t_{ij} = \frac{1}{2}(h^t_{ij} \cdot h_t) \). On the other hand, Eq. (3) is also applied in the exploration of intra-domain relationship. Thus the loss function of target domain is rewritten as:

\[ \min L_{cc} = -s_{ij} \log(\delta(\phi_{ij}^t)) - (1 - s_{ij}) \log((1 - \delta(\phi_{ij}^t))). \tag{9} \]

This constraint further improves the discriminative ability of binary codes by learning more compact category subspace. Thus, we combine these constraints to form the final coding consistency objective function:

\[ \min L_{cc} = L_{cc}^s + L_{cc}^t + L_{cc}^f. \tag{10} \]

### 3.4 Domain-Invariant Memory Bank

Considering the hash retrieval, our method DHLing alternately aims to learn domain-invariant representations over the hidden features before the hash codes. Focusing on the network design, the second fully-connection layer \( g_i = \sigma(W_g f_i) \) omits the bias term. When regarding each column vector of \( W_g \) as the individual atom, another way of understanding it is that \( g_i \) is the projection of \( f_i \) over the basis vector of space. Thus, we also define \( W_g \) as the memory bank storing the essential attribution of all instances.

However, the cross-domain discrepancy easily results in the significant difference across source and target feature distributions. Learning the projection with the same memory bank becomes unreasonable and unfair for \( f^{su}_i \) and \( f_u \). To avoid the problem, source and target domains independently learn their individual memory bank and transmit them into \( W_g \) to form the domain-invariant atoms. Concretely, the first important point is to discover the basis vector from source and target domains, respectively. Motivated by the dictionary learning [34, 50] where each data point can be represented by the sparse combination of atoms, DHLing gradually learns the domain-specific memory bank from the original hidden features \( f_t \) and their target subspace as an example, the learning process is to minimize the following formulation:

\[ L_d = \sum_{i=1}^{n^s} ||f_t - D_t \alpha^s_t||_2^2 + ||\alpha^s_t||_1. \tag{11} \]

where \( D_t \) is set as the same size with \( W_g \) and \( \| \cdot \|_1 \) represents the L1-norm regularization which uses as few atoms as possible to reconstruct the original feature representation \( f_t \). Similarly, the source memory bank is induced from minimizing the following objective with labeled and unlabeled source instances:

\[ L_d = \sum_{i=1}^{n^s} ||f_s - D_s \alpha^s_t||_2^2 + ||\alpha^s_t||_1 + \sum_{i=1}^{n^u} ||f_s - D_s \alpha^u_t||_2^2 + ||\alpha^u_t||_1. \tag{12} \]

With the fixed source and target dictionaries, the domain-invariant memory bank derives from their fusion by minimizing:

\[ L_m = ||W_g - D_s||_F^2 + ||W_g - D_t||_F^2, \tag{13} \]
where \( \| \cdot \|_F \) is the Frobenius Norm. Since \( W_q \) seeks for the balance between source and target memory bank, the domain shift is effectively eliminated via such transformation from \( g_i \) to \( h_i \).

### 3.5 Overall Objective Function

Although the retrieval system based on hash coding accelerates (3,847 images), ImageNet (4,000 images) and SUN (2,626 images), the Cross-dataset TestBed solves the cross-domain retrieval by mitigating the significant domain discrepancy. In addition, the considerable domain discrepancy heavily obstructs the adaptive retrieval. For example, the artistic painting of A is significantly different from the real images of R captured by the regular cameras. According to [12], we convert each original image into a 4096-dimensional feature by using VGG-16 network [38].

#### Baselines.

Due to the application of the binary code in our DHLing, we evaluate the performance of DHLing by comparing with the state-of-the-art hashing methods including PWCF [12], KSH [29], SDH [37], ITQ [9], LapITQ+ [53], SGH [16], DSH [18], ITQ+ [53], GTH [28], OCH [27], LSH [3], SH [42], NoTL [12]. Specifically, for the two supervised methods (i.e., SDH, KSH), only the labeled source instances of \( D_s^t \) are used as training samples. While for other left unsupervised methods and PWCF, all available source and target instances are used in the training stage.

#### Implementation details.

To verify the effectiveness of DHLing for cross-domain retrieval, this paper designs two experimental manners. First, for the conventional DAR setting, we provide all source samples with its corresponding annotation for model training. Second, for the semi-supervised DAR, instances of source domain are divided into two subsets. One consists of samples with its category information and the other one is without any label. In addition, we set three different label ratios (10%, 30% and 60%), which means the proportion of the labeled source samples over the whole retrieved database.

With respect to the organization of training and test sets, we follow the protocol of [12] to randomly select 500 images from target domain as queries (test set) and consider the remaining ones together with all source instances as training set. From the quantitative evaluation, we adopt the widely-used mean average precision (MAP) to make fair comparisons. For the qualitative measurement, the precision and recall curves and feature visualization are used to illustrate the performance of approaches. In terms of the trade-off parameters of our method, we set them as \( \lambda = 0.1, \beta = 0.01, \gamma = 0.1 \) by default for all experiments. In fact, our model performance is not sensitive to those parameters.

### 4 EXPERIMENTS

#### 4.1 Experimental Setting

**Datasets.** The MNIST [21] and USPS [14] are the popular benchmarks for handwritten digits recognition including ten categories from 0 to 9. We follow [30] to resize each original image into 16×16 and separately consider MNIST and USPS as the source and target domains to perform the cross-domain retrieval tasks.

The VLCS [41] collects images from four sets: Caltech101, LabelMe, Pascal VOC2007 and SUN909, sharing the identical label space with five classes. According to the protocol [12], we also adopt the classical network architecture [6] to extract the features with 4,096 dimensions per raw image. In the experiment, the VOC2007 with 3,376 images is regarded as the retrieved database, while the Caltech101 serves as the target domain, which contains 1,415 images.

The Cross-dataset TestBed [4] includes three domains: Caltech256 (3,847 images), ImageNet (4,000 images) and SUN (2,626 images), which are captured from different environments and the images of each domain cover 40 categories. Similar with the operation in VLCS, each image is transformed into a 4096-dimensional feature vector by using the CNN-based network [6]. In the retrieval experiments, the Caltech256 dataset is considered as the source domain, while the ImageNet is adopted as target domain.

The Office-Home dataset is a more challenging benchmark for cross-domain retrieval problem and consists of 15,500 images from four domains: Art (A), Clipart (C), Product (P) and RealWorld (R). There are 65 categories across each domain, increasing the difficulty of image classification. In addition, the considerable domain discrepancy heavily obstructs the adaptive retrieval. For example, the artistic painting of A is significantly different from the real images of R captured by the regular cameras. According to [12], we convert each original image into a 4096-dimensional feature by using VGG-16 network [38].

#### 4.2 Quantitative Results

For the traditional DAR setting, tables 1 and 2 report the comparisons of performance between our DHLing and other hashing methods. We easily achieve four conclusions by analysing them. First, our DHLing outperforms other baselines by a large margin for most cases. Specifically, with 64-bit binary code, DHLing surpasses the second best PWCF by 6.5% in terms of MAP for the MNIST&USPS task, which illustrates that our method effectively solves the cross-domain retrieval by mitigating the significant domain discrepancy. Moreover, although PWCF adopts the triplet loss constraint to capture the relationship of cross-domain instances,
Table 1: Comparisons of MAP under DAR for MNIST&USPS, VOC2007&Caltech, and Caltech256&ImageNet benchmarks with the varying coding bit from 16 to 128. The best MAP with the identical code length is highlighted with bold type.

<table>
<thead>
<tr>
<th></th>
<th>MAP 16</th>
<th>MAP 32</th>
<th>MAP 48</th>
<th>MAP 64</th>
<th>MAP 96</th>
<th>MAP 128</th>
</tr>
</thead>
<tbody>
<tr>
<td>SH</td>
<td>15.71</td>
<td>13.85</td>
<td>12.05</td>
<td>11.78</td>
<td>11.38</td>
<td>11.78</td>
</tr>
<tr>
<td>OCH</td>
<td>18.94</td>
<td>25.73</td>
<td>26.73</td>
<td>26.34</td>
<td>27.88</td>
<td>29.22</td>
</tr>
<tr>
<td>GTH</td>
<td>19.10</td>
<td>24.17</td>
<td>24.27</td>
<td>24.38</td>
<td>23.64</td>
<td>29.36</td>
</tr>
<tr>
<td>ITQ+</td>
<td>20.27</td>
<td>20.53</td>
<td>16.77</td>
<td>15.87</td>
<td>17.79</td>
<td>14.90</td>
</tr>
<tr>
<td>DSH</td>
<td>21.15</td>
<td>27.53</td>
<td>29.71</td>
<td>26.13</td>
<td>26.6</td>
<td>28.94</td>
</tr>
<tr>
<td>SGH</td>
<td>24.83</td>
<td>24.78</td>
<td>25.85</td>
<td>27.78</td>
<td>28.26</td>
<td>29.35</td>
</tr>
<tr>
<td>ITQ</td>
<td>27.38</td>
<td>30.92</td>
<td>31.44</td>
<td>32.25</td>
<td>33.12</td>
<td>33.44</td>
</tr>
<tr>
<td>NoTL</td>
<td>28.13</td>
<td>30.05</td>
<td>28.24</td>
<td>30.34</td>
<td>31.76</td>
<td>31.25</td>
</tr>
<tr>
<td>SDH</td>
<td>29.98</td>
<td>43.02</td>
<td>42.57</td>
<td>46.56</td>
<td>42.4</td>
<td>48.12</td>
</tr>
<tr>
<td>KSH</td>
<td>43.75</td>
<td>46.91</td>
<td>50.02</td>
<td>47.84</td>
<td>45.25</td>
<td>46.81</td>
</tr>
<tr>
<td>PWCF</td>
<td>47.47</td>
<td>51.99</td>
<td>51.44</td>
<td>51.75</td>
<td>50.89</td>
<td>53.95</td>
</tr>
<tr>
<td>Ours</td>
<td>49.24</td>
<td>54.90</td>
<td>56.30</td>
<td>58.28</td>
<td>59.14</td>
<td>58.68</td>
</tr>
</tbody>
</table>

(a) PWCF (P→R)  (b) Ours (P→R)  (c) PWCF (MNIST&USPS)  (d) Ours (MNIST&USPS)

Table 2: Comparisons of mean average precision (MAP) of domain adaptive retrieval (DAR) on Office-Home with 64-bit binary code. The best MAP is highlighted with bold type.

<table>
<thead>
<tr>
<th></th>
<th>MAP P→R</th>
<th>MAP R→P</th>
<th>MAP C→R</th>
<th>MAP R→C</th>
<th>MAP A→R</th>
<th>MAP R→A</th>
</tr>
</thead>
<tbody>
<tr>
<td>NoTL</td>
<td>25.02</td>
<td>29.09</td>
<td>16.98</td>
<td>14.73</td>
<td>23.78</td>
<td>21.05</td>
</tr>
<tr>
<td>SH</td>
<td>15.01</td>
<td>16.82</td>
<td>10.38</td>
<td>8.32</td>
<td>14.52</td>
<td>12.41</td>
</tr>
<tr>
<td>LSH</td>
<td>10.74</td>
<td>14.78</td>
<td>9.85</td>
<td>8.37</td>
<td>12.24</td>
<td>10.02</td>
</tr>
<tr>
<td>OCH</td>
<td>20.06</td>
<td>20.23</td>
<td>10.97</td>
<td>10.61</td>
<td>19.35</td>
<td>15.32</td>
</tr>
<tr>
<td>GTH</td>
<td>19.4</td>
<td>22.8</td>
<td>12.27</td>
<td>12.03</td>
<td>20.98</td>
<td>16.47</td>
</tr>
<tr>
<td>ITQ+</td>
<td>17.94</td>
<td>-</td>
<td>10.61</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>DSH</td>
<td>8.86</td>
<td>7.72</td>
<td>6.44</td>
<td>6.15</td>
<td>10.07</td>
<td>9.11</td>
</tr>
<tr>
<td>LapITQ+</td>
<td>15.94</td>
<td>-</td>
<td>11.72</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>SDH</td>
<td>25.75</td>
<td>27.90</td>
<td>15.97</td>
<td>16.72</td>
<td>32.06</td>
<td>22.79</td>
</tr>
<tr>
<td>KSH</td>
<td>32.02</td>
<td>34.42</td>
<td>21.56</td>
<td>18.51</td>
<td>25.87</td>
<td>20.04</td>
</tr>
<tr>
<td>PWCF</td>
<td>34.03</td>
<td>34.44</td>
<td>24.22</td>
<td>18.42</td>
<td>34.57</td>
<td>29.91</td>
</tr>
<tr>
<td>Ours</td>
<td>48.47</td>
<td>45.24</td>
<td>30.81</td>
<td>25.15</td>
<td>43.30</td>
<td>38.68</td>
</tr>
</tbody>
</table>

DHLing further explores the intra-domain knowledge to intensify feature representation with powerful discriminative ability. Second, with the varying coding bit from 16 to 64, DHLing extracts more meaningful knowledge from longer hash code to improve the precision and obtains the stable retrieval ability. For example, with the increasing length of hash code, LapITQ+ did not obtain higher precision for MNIST&USPS, our method, however, stably improves its performance from 49.24% (16 bits) to 59.14% (128 bits). The main reason lies in that our DHLing focuses on learning the consistent hidden features before the binary code to improve the robustness of system, which helps our method avoid abandoning abundant semantic information when transforming the continuous real-value representation into the hash code. Third, when compared with others, DHLing becomes more suitable for the practical retrieval task on large-scale dataset. As the aforementioned discussion, Office-Home dataset is a challenging benchmark for cross-domain retrieval since it involves abundant images and categories. Interestingly, our DHLing still achieves the highest MAP on six tasks. In terms of the average result, DHLing beats PWCF with the second best performance by 9.5%, which demonstrates our method employs the consistent coding optimization and domain-invariant memory bank to eliminate the negative influence of domain shift on cross-domain retrieval problem. Finally, the supervision of annotation significantly enhances the identify ability of model. Concretely, the supervised approaches such as PWCF and SDH both achieve the comparable even better precision when contrasting them with other unsupervised methods (OCH, GTH). It explicitly emphasises the importance of annotation on learning discriminative features. However, the...
Table 3 shows the results of baselines and our method under SDAR setting with two various retrieval tasks and these three competitors are supervised methods. According to the observation of tables 1 and 3, it is simple to observe that all supervised methods suffer from the considerable performance degradation. Although our method also encounters with such a drop, DHLing effectively mitigates the effect of insufficient annotation in several cases. On task Caltech256&ImageNet, when the proportion of labeled source samples is changed from 100% to 10%, our method always fights off others. The success of DHLing mainly results from the design of domain-specific clustering. This module explores the structural knowledge of samples to annotate the pseudo label, making the consistent coding optimization more effective for aligning source and target domains. From the above comparison and analysis about the mean average precision, our method both achieves the better performance than others under DAR and SDAR setting.

4.3 Empirical Analysis
To provide the intuitive comparison between our DHLing and others, we additionally report the feature embedding, the relationship between recall/precision and the number of retrieved instances and the analysis of retrieved images.

Feature visualization. Under DAR scenario, we firstly carry out the experiment on task P→R of Office-Home dataset with 64 bits and extract the hidden feature $h_i$ before the hash process to draw the feature embedding of the training instances from source and target domains shown in Figure 3 (b). Follow the same operation, Figure 3 (a) shows the feature embedding obtained by PWCF. Through their comparison, we notice that our DHLing achieves better alignment across these two domains than PWCF, which is beneficial from the consistent coding optimization narrowing the distance between
### Table 3: Comparisons of MAP under SDAR for MNIST&USPS and Caltech256&ImageNet benchmarks with the varying coding bit from 16 to 128 and various ratios (10%, 30% and 60%) of labeled source samples over the size of the whole retrieved database.

<table>
<thead>
<tr>
<th>Ratio</th>
<th>Bit</th>
<th>MNIST&amp;USPS</th>
<th>Caltech256&amp;ImageNet</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>16</td>
<td>32</td>
<td>48</td>
</tr>
<tr>
<td>10%</td>
<td>SDH</td>
<td>8.84</td>
<td>10.38</td>
</tr>
<tr>
<td></td>
<td>KSH</td>
<td>12.93</td>
<td>13.56</td>
</tr>
<tr>
<td></td>
<td>PWCF</td>
<td>16.38</td>
<td>18.88</td>
</tr>
<tr>
<td></td>
<td>Ours</td>
<td>35.29</td>
<td>45.67</td>
</tr>
<tr>
<td>30%</td>
<td>SDH</td>
<td>15.55</td>
<td>24.64</td>
</tr>
<tr>
<td></td>
<td>KSH</td>
<td>25.67</td>
<td>29.75</td>
</tr>
<tr>
<td></td>
<td>PWCF</td>
<td>29.91</td>
<td>36.10</td>
</tr>
<tr>
<td></td>
<td>Ours</td>
<td>38.64</td>
<td>47.06</td>
</tr>
<tr>
<td>60%</td>
<td>SDH</td>
<td>19.05</td>
<td>29.64</td>
</tr>
<tr>
<td></td>
<td>KSH</td>
<td>28.67</td>
<td>33.95</td>
</tr>
<tr>
<td></td>
<td>PWCF</td>
<td>33.31</td>
<td>39.93</td>
</tr>
<tr>
<td></td>
<td>Ours</td>
<td>41.58</td>
<td>48.46</td>
</tr>
</tbody>
</table>

![Figure 5](https://example.com/fig5.png)

**Figure 5:** Comparison of retrieval ability in terms of (a) Recall and (b) Precision under different size of retrieved samples for task P→R of Office-Home dataset with 64 bits.

To clearly understand the influence of each module on improving the model performance, we design two model variants (i.e., DHLing-CC and DHLing-M) by removing the consistent coding constraint $L_{cc}$ and domain-invariant memory bank $L_{in}$, respectively. Under DAR scenario, we perform experiments on MNIST&USPS and report the corresponding results in Figure 6. Accordingly, we easily know that the consistent coding constraint makes more contribution to the model training. Moreover, without the design of domain-invariant memory bank, the performance of model suffers from obvious reduction. Thus, both modules lead to the improvements of DHLing.

### 5 CONCLUSIONS

Conventional domain adaptive retrieval assumes that the model is trained with well-labeled source domain and target images and retrieves source images given the query target image. Differently, in this paper we introduce the challenging yet more practical setting –SDAR– assuming the source domain includes a small portion of annotated source images and abundant unlabeled ones. We propose a novel Discriminative Hashing Learning (DHLing) model by simultaneously exploring structural knowledge to assign annotation for unlabeled samples for more discriminative hash code, and seeking the domain-invariant memory bank to supervise the effective cross-domain feature learning. The comprehensive experimental results on several benchmarks demonstrate the superiority of our approach over state-of-the-art retrieval methods under both conventional and semi-supervised cross-domain retrieval settings.