Deep Hashing Network for Unsupervised Domain Adaptation

深度哈希网络用于无监督领域适应

Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, Sethuraman Panchanathan Center for Cognitive Ubiquitous Computing, Arizona State University, Tempe, AZ, USA

Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, Sethuraman Panchanathan 亚利桑那州立大学认知普适计算中心，美国亚利桑那州坦佩

{hemanthv, jeusebio, shayok.chakraborty, panch}@asu.edu

{hemanthv, jeusebio, shayok.chakraborty, panch}@asu.edu

Abstract

摘要

In recent years, deep neural networks have emerged as a dominant machine learning tool for a wide variety of application domains. However, training a deep neural network requires a large amount of labeled data, which is an expensive process in terms of time, labor and human expertise. Domain adaptation or transfer learning algorithms address this challenge by leveraging labeled data in a different, but related source domain, to develop a model for the target domain. Further, the explosive growth of digital data has posed a fundamental challenge concerning its storage and retrieval. Due to its storage and retrieval efficiency, recent years have witnessed a wide application of hashing in a variety of computer vision applications. In this paper, we first introduce a new dataset, Office-Home, to evaluate domain adaptation algorithms. The dataset contains images of a variety of everyday objects from multiple domains. We then propose a novel deep learning framework that can exploit labeled source data and unlabeled target data to learn informative hash codes, to accurately classify unseen target data. To the best of our knowledge, this is the first research effort to exploit the feature learning capabilities of deep neural networks to learn representative hash codes to address the domain adaptation problem. Our extensive empirical studies on multiple transfer tasks corroborate the usefulness of the framework in learning efficient hash codes which outperform existing competitive baselines for unsupervised domain adaptation.

近年来，深度神经网络已成为各种应用领域的主导机器学习工具。然而，训练深度神经网络需要大量标记数据，这在时间、劳动和人力专业知识方面都是一项昂贵的过程。领域适应或迁移学习算法通过利用不同但相关源领域中的标记数据来解决这一挑战，以为目标领域开发模型。此外，数字数据的爆炸性增长在存储和检索方面提出了根本性挑战。由于其存储和检索效率，近年来哈希在各种计算机视觉应用中得到了广泛应用。在本文中，我们首先介绍一个新的数据集，Office-Home，以评估领域适应算法。该数据集包含来自多个领域的各种日常物品的图像。然后，我们提出一个新颖的深度学习框架，可以利用标记的源数据和未标记的目标数据来学习信息丰富的哈希码，以准确分类未见过的目标数据。据我们所知，这是首次利用深度神经网络的特征学习能力来学习代表性哈希码以解决领域适应问题的研究工作。我们在多个迁移任务上的广泛实证研究证实了该框架在学习高效哈希码方面的有效性，这些哈希码优于现有的无监督领域适应竞争基准。

1. Introduction

1. 引言

Deep learning algorithms automatically learn a discriminating set of features and have depicted commendable performance in a variety of computer vision applications. Unfortunately, training a deep model necessitates a large volume of labeled data, which can be time consuming and expensive to acquire. However, labeled data from a different, but related domain is often available, which has motivated the development of algorithms which can leverage labeled data in a source domain to develop a machine learning model for the target domain. Learning a discriminative model in the presence of the shift between training and test distributions is known as transfer learning or domain adaptation [17]. Unsupervised domain adaptation is a challenging setting, where labeled data is available only in the source domain; no labeled data is available in the target domain. Conventional shallow transfer learning methods develop their models in two stages, feature extraction followed by domain adaptation. The features are fixed and then a model is trained to align the source and target domains [16,20,33,38,42,43,44]. On the other hand, deep transfer learning procedures exploit the feature learning capabilities of deep networks to learn transferable feature representations for domain adaptation and have demonstrated impressive empirical performance [17, 18, 31, 34, 46].

深度学习算法自动学习一组区分特征，并在各种计算机视觉应用中表现出令人赞赏的性能。不幸的是，训练深度模型需要大量标记数据，而获取这些数据可能既耗时又昂贵。然而，来自不同但相关领域的标记数据通常是可用的，这促使了能够利用源领域的标记数据来为目标领域开发机器学习模型的算法的发展。在训练和测试分布之间存在偏移的情况下学习一个区分模型被称为迁移学习或领域适应 [17]。无监督领域适应是一种具有挑战性的设置，其中标记数据仅在源领域可用；目标领域没有标记数据。传统的浅层迁移学习方法在两个阶段中开发其模型，特征提取后进行领域适应。特征是固定的，然后训练一个模型来对齐源领域和目标领域 [16,20,33,38,42,43,44]。另一方面，深度迁移学习程序利用深度网络的特征学习能力来学习可迁移的特征表示以进行领域适应，并展示了令人印象深刻的实证性能 [17, 18, 31, 34, 46]。

The explosive growth of digital data in the modern era has posed fundamental challenges regarding their storage, retrieval and computational requirements. Against this backdrop, hashing has emerged as one of the most popular and effective techniques due to its fast query speed and low memory cost [48]. Hashing techniques transform high dimensional data into compact binary codes and generate similar binary codes for similar data items. Motivated by this fact, we propose to train a deep neural network to output binary hash codes (instead of probability values), which can be used for classification. We see two advantages to estimating a hash value instead of a standard probability vector in the final layer of the network: (i) the hash values are used to develop a unique loss function for target data in the absence of labels and (ii) during prediction, the hash value of a test sample can be compared against the hash values of the training samples to arrive at a more robust category prediction.

在现代时代，数字数据的爆炸性增长对其存储、检索和计算要求提出了根本性的挑战。在这种背景下，哈希技术因其快速的查询速度和低内存成本而成为最受欢迎和有效的技术之一 [48]。哈希技术将高维数据转换为紧凑的二进制代码，并为相似的数据项生成相似的二进制代码。基于这一事实，我们提出训练一个深度神经网络以输出二进制哈希码 (而不是概率值)，可用于分类。我们认为在网络的最后一层估计哈希值而不是标准概率向量有两个优势:(i) 哈希值用于在缺乏标签的情况下为目标数据开发独特的损失函数；(ii) 在预测过程中，可以将测试样本的哈希值与训练样本的哈希值进行比较，从而得出更稳健的类别预测。

In this paper, we first introduce a new dataset, Office-Home, which we use to evaluate our algorithm. The Office-Home dataset is an object recognition dataset which contains images from 4 domains. It has around 15,500 images organized into 65 categories. We further propose a novel deep learning framework called Domain Adaptive Hashing (DAH) to learn informative hash codes to address the problem of unsupervised domain adaptation. We propose a unique loss function to train the deep network with the following components: (i) supervised hash loss for labeled source data, which ensures that source samples belonging to the same class have similar hash codes; (ii) unsupervised entropy loss for unlabeled target data, which imposes each target sample to align closely with exactly one of the source categories and be distinct from the other categories and (iii) a loss based on multi-kernel Maximum Mean Discrepancy (MK-MMD), which seeks to learn transferable features within the layers of the network to minimize the distribution difference between the source and target domains. Figure 1 illustrates the different layers of the DAH and the components of the loss function.

在本文中，我们首先介绍一个新的数据集，Office-Home，我们用它来评估我们的算法。Office-Home 数据集是一个对象识别数据集，包含来自 4 个领域的图像。它大约有 15,500 张图像，组织成 65 个类别。我们进一步提出了一种新颖的深度学习框架，称为领域自适应哈希 (DAH)，用于学习信息丰富的哈希码，以解决无监督领域适应的问题。我们提出了一种独特的损失函数来训练深度网络，包含以下几个组成部分:(i) 针对标记源数据的监督哈希损失，确保属于同一类别的源样本具有相似的哈希码；(ii) 针对未标记目标数据的无监督熵损失，强制每个目标样本与正好一个源类别紧密对齐，并与其他类别区分开；(iii) 基于多核最大均值差异 (MK-MMD) 的损失，旨在学习网络层内的可迁移特征，以最小化源域和目标域之间的分布差异。图 1 说明了 DAH 的不同层以及损失函数的组成部分。

2. Related Work

2. 相关工作

There have been many approaches to address the problem of domain-shift in unsupervised domain adaptation. One straightforward approach is, to modify a classifier trained for the source data by adapting it to classify target data [1, 4] or learn a transformation matrix to linearly transform the source data, so that it is aligned with the target [27, 42] . Some other procedures re-weight the data points in the source domain, to select source data that is similar to the target, when training a domain adaptive classifier, [9, 10, 19] . A standard procedure to reduce domain discrepancy is, to project the source and target data to a common subspace, thereby aligning their principal axes [16, 44]. Reducing domain disparity through nonlinear alignment of data has been possible with Maximum Mean Discrepancy (MMD) - a measure that provides the distribution difference between two datasets in a reproducing-kernel Hilbert space [13]. Kernel-PCA based methods apply the MMD to achieve nonlinear alignment of domains [32, 33, 38]. Manifold based approaches are also popular in domain adaptation for computer vision, where the subspace of a domain is treated as a point on the manifold and transformations are learned to align two domains [20,23]. A survey of popular domain adaptation techniques for computer vision is provided in [41] and a more generic survey of transfer learning approaches can be found in [39].

针对无监督领域适应中的领域偏移问题，已经提出了许多方法。一种直接的方法是，通过调整为源数据训练的分类器，使其能够对目标数据进行分类 [1, 4] ，或者学习一个变换矩阵，以线性变换源数据，使其与目标数据对齐 [27, 42] 。其他一些程序在训练领域自适应分类器时，对源领域中的数据点进行重加权，以选择与目标相似的源数据 [9, 10, 19] 。减少领域差异的标准程序是将源数据和目标数据投影到一个共同的子空间，从而对齐它们的主轴 [16, 44]。通过数据的非线性对齐来减少领域差异已经可以通过最大均值差异 (MMD) 实现——这是一种在重现核希尔伯特空间中提供两个数据集分布差异的度量 [13]。基于核主成分分析 (Kernel-PCA) 的方法应用 MMD 来实现领域的非线性对齐 [32, 33, 38]。基于流形的方法在计算机视觉的领域适应中也很流行，其中一个领域的子空间被视为流形上的一个点，并学习变换以对齐两个领域 [20, 23]。关于计算机视觉的流行领域适应技术的综述见于 [41]，而关于迁移学习方法的更通用的综述可以在 [39] 中找到。

All of the above techniques can be termed as shallow learning procedures, since the models are learned using predetermined features. In recent years deep learning has become very successful at learning highly discriminative features for computer vision applications [8]. Deep learning systems like deep CNNs learn representations of data that capture underlying factors of variation between different tasks in a multi-task transfer learning setting [3]. These representations also disentangle the factors of variation allowing for the transfer of knowledge between tasks [12, 18, 37]. Yosinski et al. [49] demonstrated how the lower layers of a network produce generic features and the upper layers output task specific features. Based on this, deep learning procedures for domain adaptation train networks to learn transferable features in the fully connected final layers of a network [31, 46] . In other approaches to deep domain adaptation, Ganin et al. [17] trained domain adversarial networks to learn features that make the source and target domain indistinguishable and Long et al. [34], trained a network to do both feature adaptation and classifier adaptation using residual transfer networks.

所有上述技术可以被称为浅层学习程序，因为模型是使用预定特征进行学习的。近年来，深度学习在计算机视觉应用中成功地学习到了高度区分性的特征 [8]。深度学习系统如深度卷积神经网络 (CNN) 学习数据的表示，这些表示捕捉了多任务迁移学习环境中不同任务之间的潜在变化因素 [3]。这些表示还解开了变化因素，使得知识能够在任务之间转移 [12, 18, 37]。Yosinski 等人 [49] 证明了网络的低层如何产生通用特征，而高层则输出特定任务的特征。基于此，深度学习程序用于领域适应，训练网络在网络的全连接最终层学习可转移特征 [31, 46] 。在深度领域适应的其他方法中，Ganin 等人 [17] 训练了领域对抗网络，以学习使源领域和目标领域不可区分的特征，而 Long 等人 [34] 则训练了一个网络，以使用残差转移网络进行特征适应和分类器适应。

Unsupervised hashing techniques have been developed to extract unique hash codes for efficient storage and retrieval of data [22, 25] . Neural network based hashing has led the way in state-of-the-art unsupervised hashing techniques [7, 11, 14] . The closest work incorporating hashing and adaptation appears in cross-modal hashing, where deep hashing techniques embed multi-modal data and learn hash codes for two related domains, like text and images [5, 6, 29] . However, these algorithms are not unsupervised and they are mainly applied to extract common hash codes for multi-modal data for retrieval purposes. To the best of our knowledge, there has been no work in unsupervised domain adaptation using deep hashing networks. We now present the Domain Adaptive Hashing (DAH) network for unsupervised domain adaptation through deep hashing.

无监督哈希技术已被开发出来，以提取独特的哈希码，以实现数据的高效存储和检索 [22, 25] 。基于神经网络的哈希技术在最先进的无监督哈希技术中处于领先地位 [7, 11, 14] 。结合哈希和适应的最接近的工作出现在跨模态哈希中，其中深度哈希技术嵌入多模态数据并为两个相关领域 (如文本和图像) 学习哈希码 [5, 6, 29] 。然而，这些算法并不是无监督的，主要用于提取多模态数据的共同哈希码以便于检索。根据我们所知，目前尚无使用深度哈希网络进行无监督领域适应的工作。我们现在提出了用于通过深度哈希进行无监督领域适应的领域自适应哈希 (DAH) 网络。

3. Domain Adaptive Hashing Networks

3. 领域自适应哈希网络

In unsupervised domain adaptation, we consider data from two domains; source and target. The source consists of labeled data, $\mathcal{D}_s = \{\mathbf{x}_i^s, y_i^s\}_{i=1}^{n_s}$ and the target has only unlabeled data $\mathcal{D}_t = \{\mathbf{x}_i^t\}_{i=1}^{n_t}$. The data points $\mathbf{x}_i^*$ belong to X , where X is some input space. The corresponding labels are represented by $y_i^* \in Y \ := \ \{1, \ldots, C\}$. The paradigm of domain adaptive learning attempts to address the problem of domain-shift in the data, where the data distributions of the source and target are different, i.e. $P_s(X, Y) \neq P_t(X, Y)$. The domain-shift notwithstanding, our goal is to train a deep neural network classifier $\psi(\ .\ ), thatcanpredictthela-$ bels $\{\widehat{y}_i^t\}_{i=1}^{n_t}$, for the target data.

在无监督领域适应中，我们考虑来自两个领域的数据；源领域和目标领域。源领域由标记数据组成，$\mathcal{D}_s = \{\mathbf{x}_i^s, y_i^s\}_{i=1}^{n_s}$ ，而目标领域只有未标记数据 $\mathcal{D}_t = \{\mathbf{x}_i^t\}_{i=1}^{n_t}$ 。数据点 $\mathbf{x}_i^*$ 属于 X ，其中 X 是某个输入空间。相应的标签由 $y_i^* \in Y \ := \ \{1, \ldots, C\}$ 表示。领域自适应学习的范式试图解决数据中的领域偏移问题，其中源领域和目标领域的数据分布是不同的，即 $P_s(X, Y) \neq P_t(X, Y)$ 。尽管存在领域偏移，我们的目标是为目标数据训练一个深度神经网络分类器 $\psi(\ .\ ), thatcanpredictthela-$ 标签 $\{\widehat{y}_i^t\}_{i=1}^{n_t}$ 。

We implement the neural network as a deep CNN which consists of 5 convolution layers conv1 - conv5 and 3 fully connected layers $fc6 - fc8$ followed by a loss layer. In our model, we introduce a hashing layer hash-fc8 in place of the standard $fc8$ layer to learn a binary code $\mathbf{h}_i$, for every data point $\mathbf{x}_i$, where $\mathbf{h}_i \in \{-1, +1\}^d$. The hash-fc8 layer is driven by two loss functions, (i) supervised hash loss for the source data, (ii) unsupervised entropy loss for the target data. The supervised hash loss ensures hash values that are distinct and discriminatory, i.e. if $\mathbf{x}_i$ and $\mathbf{x}_j$ belong to the same category, their hash values $\mathbf{h}_i$ and $\mathbf{h}_j$ are similar and different otherwise. The unsupervised entropy loss aligns the target hash values with source hash values based on the similarity of their feature representations. The output of the network is represented as $\psi(\mathbf{x})$, where $\psi(\mathbf{x}) \in \mathbb{R}^d$, which we convert to a hash code $\mathbf{h} = \mathrm{sgn}(\psi(\mathbf{x}))$, where $\mathrm{sgn}(\ .\ )$

我们将神经网络实现为一个深度卷积神经网络 (CNN)，它由 5 个卷积层 conv1 - conv5 和 3 个全连接层 $fc6 - fc8$ ，后面跟着一个损失层。在我们的模型中，我们引入了一个哈希层 hash-fc8，替代标准的 $fc8$ 层，以学习每个数据点 $\mathbf{x}_i$ 的二进制代码 $\mathbf{h}_i$ ，其中 $\mathbf{h}_i \in \{-1, +1\}^d$ 。hash-fc8 层由两个损失函数驱动，(i) 源数据的监督哈希损失，(ii) 目标数据的无监督熵损失。监督哈希损失确保哈希值是独特且具有区分性的，即如果 $\mathbf{x}_i$ 和 $\mathbf{x}_j$ 属于同一类别，则它们的哈希值 $\mathbf{h}_i$ 和 $\mathbf{h}_j$ 相似，否则不同。无监督熵损失根据特征表示的相似性将目标哈希值与源哈希值对齐。网络的输出表示为 $\psi(\mathbf{x})$ ，其中 $\psi(\mathbf{x}) \in \mathbb{R}^d$ ，我们将其转换为哈希代码 $\mathbf{h} = \mathrm{sgn}(\psi(\mathbf{x}))$ ，其中 $\mathrm{sgn}(\ .\ )$ 。

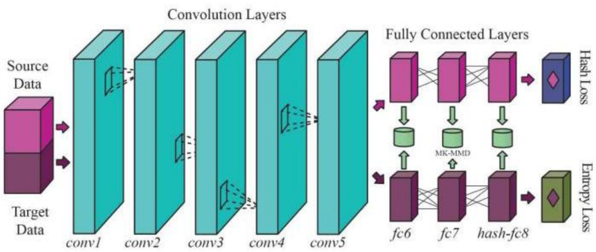


Figure 1: The Domain Adaptive Hash (DAH) network that outputs hash codes for the source and the target. The network is trained with a batch of source and target data. The convolution layers conv1 - conv5 and the fully connected layers $fc6$ and $fc7$ are fine tuned from the VGG-F network. The MK-MMD loss trains the DAH to learn feature representations which align the source and the target. The hash-fc8 layer is trained to output vectors of d dimensions. The supervised hash loss drives the DAH to estimate a unique hash value for each object category. The unsupervised entropy loss aligns the target hash values to their corresponding source categories. Best viewed in color.

图 1: 域自适应哈希 (DAH) 网络，为源和目标输出哈希码。该网络使用一批源和目标数据进行训练。卷积层 conv1 - conv5 和全连接层 $fc6$ 和 $fc7$ 从 VGG-F 网络进行微调。MK-MMD 损失使 DAH 学习特征表示，从而对齐源和目标。hash-fc8 层被训练以输出 d 维度的向量。监督哈希损失驱动 DAH 为每个对象类别估计一个唯一的哈希值。无监督熵损失将目标哈希值与其对应的源类别对齐。最佳效果请使用彩色显示。

is the sign function. Once the network has been trained, the probability of $\mathbf{x}$ being assigned a label y is given by $f(\mathbf{x}) = p(y \mid \mathbf{h})$. We train the network using $\mathcal{D}_s$ and $\mathcal{D}_t$ and predict the target data labels $\widehat{y}_*^t$ using $f(\ .\ )$.

是符号函数。一旦网络训练完成，分配给标签 y 的概率由 $f(\mathbf{x}) = p(y \mid \mathbf{h})$ 给出。我们使用 $\mathcal{D}_s$ 和 $\mathcal{D}_t$ 训练网络，并使用 $f(\ .\ )$. 预测目标数据标签 $\widehat{y}_*^t$ 。

In order to address the issue of domain-shift, we need to align the feature representations of the target and the source. We do that by reducing the domain discrepancy between the source and target feature representations at multiple layers of the network. In the following subsections, we discuss the design of the domain adaptive hash (DAH) network in detail.

为了解决领域转移的问题，我们需要对齐目标和源的特征表示。我们通过减少源和目标特征表示之间的领域差异，在网络的多个层次上实现这一点。在接下来的小节中，我们将详细讨论域自适应哈希 (DAH) 网络的设计。

3.1. Reducing Domain Disparity

3.1. 减少领域差异

Deep learning methods have been very successful in domain adaptation with state-of-the-art algorithms [17, 31, 34 , 46] in recent years. The feature representations transition from generic to task-specific as one goes up the layers of a deep CNN [49]. The convolution layers conv1 to conv5 have been shown to be generic and so, readily transferable, whereas the fully connected layers are more task-specific and need to be adapted before they can be transferred. In the DAH algorithm, we attempt to minimize the MK-MMD loss to reduce the domain difference between the source and target feature representations for fully connected layers, $\mathcal{F} = \{fc6, fc7, fc8\}$. Such a loss function has been used in previous research [31, 34]. The multi-layer MK-MMD loss is given by,

深度学习方法在领域适应方面取得了很大的成功，近年来出现了最先进的算法 [17, 31, 34 , 46]。随着深度卷积神经网络 (CNN) 层次的增加，特征表示从通用转变为特定任务 [49]。卷积层 conv1 到 conv5 被证明是通用的，因此可以轻松迁移，而全连接层则更具任务特异性，需要在迁移之前进行适应。在 DAH 算法中，我们试图最小化 MK-MMD 损失，以减少源特征表示和目标特征表示之间的领域差异，针对全连接层 $\mathcal{F} = \{fc6, fc7, fc8\}$ 。这样的损失函数在之前的研究中已被使用 [31, 34]。多层 MK-MMD 损失定义为，

$$\mathcal{M}(u_s, u_t) = \sum_{l \in \mathcal{F}} d_k^2\left(u_s^l, u_t^l\right) \tag{1}$$

where, $u_s^l = \left\{\mathbf{u}_i^{s,l}\right\}_{i=1}^{n_s}$ and $u_t^l = \left\{\mathbf{u}_i^{t,l}\right\}_{i=1}^{n_t}$ are the set of output representations for the source and target data at layer l , where $\mathbf{u}_i^{*,l}$ is the output representation of $\mathbf{x}_i^*$ for the l^{th} layer. The final layer outputs are denoted as $\mathcal{U}_s$ and $\mathcal{U}_t$. The MK-MMD measure $d_k^2(\ .\ ) isthemulti-kernelmaximum$ mean discrepancy between the source and target representations,[24]. For a nonlinear mapping $\phi(\ .\ ) associatedwitha$ reproducing kernel Hilbert space $\mathcal{H}_k$ and kernel $k(\ .\ ), where\ k(\mathbf{x}, \mathbf{y}) = \langle \phi(\mathbf{x}), \phi(\mathbf{y}) \rangle$, the MMD is defined as,

其中，$u_s^l = \left\{\mathbf{u}_i^{s,l}\right\}_{i=1}^{n_s}$ 和 $u_t^l = \left\{\mathbf{u}_i^{t,l}\right\}_{i=1}^{n_t}$ 是层 l 上源数据和目标数据的输出表示集合，其中 $\mathbf{u}_i^{*,l}$ 是 $\mathbf{x}_i^*$ 在 l^{th} 层的输出表示。最终层的输出表示为 $\mathcal{U}_s$ 和 $\mathcal{U}_t$ 。MK-MMD 测度 $d_k^2(\ .\ ) isthemulti-kernelmaximum$ 表示源表示和目标表示之间的均值差异 [24]。对于非线性映射 $\phi(\ .\ ) associatedwitha$ 重现核希尔伯特空间 $\mathcal{H}_k$ 和核 $k(\ .\ ), where\ k(\mathbf{x}, \mathbf{y}) = \langle \phi(\mathbf{x}), \phi(\mathbf{y}) \rangle$ ， MMD 定义为，

$$d_k^2\left(u_s^l, u_t^l\right) = \left\|\mathbb{E}\left[\phi\left(\mathbf{u}^{s,l}\right)\right] - \mathbb{E}\left[\phi\left(\mathbf{u}^{t,l}\right)\right]\right\|_{\mathcal{H}_k}^2. \tag{2}$$

The characteristic kernel $k(\ .\ ), isdeterminedasaconvex$ combination of κ PSD kernels, $\{k_m\}_{m=1}^{\kappa}, \mathcal{K} := \left\{k : k = \sum_{m=1}^{\kappa} \beta_m k_m, \sum_{m=1}^{\kappa} \beta_m = 1, \beta_m \geq 0, \forall m\right\}$. We set $\beta_m = 1/\kappa$ according to [34] and it works well in practice.

特征核 $k(\ .\ ), isdeterminedasaconvex$ 是 κ PSD 核的组合 $\{k_m\}_{m=1}^{\kappa}, \mathcal{K} := \{k : k = \sum_{m=1}^{\kappa} \beta_m k_m, \sum_{m=1}^{\kappa} \beta_m = 1, \beta_m \geq 0, \forall$ 。我们根据 [34] 设置 $\beta_m = 1/\kappa$ ，并且在实践中效果良好。

3.2. Supervised Hashing for Source Data

3.2. 源数据的监督哈希

The Hamming distance for a pair of hash values $\mathbf{h}_i$ and $\mathbf{h}_j$ has a unique relationship with the dot product $\langle \mathbf{h}_i, \mathbf{h}_j \rangle$, given by: $\operatorname{dist}_H(\mathbf{h}_i, \mathbf{h}_j) = \frac{1}{2}\left(d - \mathbf{h}_i^\top \mathbf{h}_j\right)$, where d is the hash length. The dot product $\langle \mathbf{h}_i, \mathbf{h}_j \rangle$ can be treated as a similarity measure for the hash codes. Larger the value of the dot product (high similarity), smaller is the distance dist_H and smaller the dot product (low similarity), larger is the distance dist_H . Let $s_{ij} \in \{0, 1\}$ be the similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$. If $\mathbf{x}_i$ and $\mathbf{x}_j$ belong to the same category, $s_{ij} = 1$ and 0, otherwise. The probability of similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$ given the corresponding hash values $\mathbf{h}_i$ and $\mathbf{h}_j$, can be expressed as a likelihood function, given by,

一对哈希值 $\mathbf{h}_i$ 和 $\mathbf{h}_j$ 的汉明距离与点积 $\langle \mathbf{h}_i, \mathbf{h}_j \rangle$ 之间存在独特的关系，表示为: $\operatorname{dist}_H(\mathbf{h}_i, \mathbf{h}_j) = \frac{1}{2}\left(d - \mathbf{h}_i^\top \mathbf{h}_j\right)$ ，其中 d 是哈希长度。点积 $\langle \mathbf{h}_i, \mathbf{h}_j \rangle$ 可以被视为哈希码的相似性度量。点积的值越大 (高相似性)，距离 dist_H 越小；点积的值越小 (低相似性)，距离 dist_H 越大。设 $s_{ij} \in \{0, 1\}$ 为 $\mathbf{x}_i$ 和 $\mathbf{x}_j$ 之间的相似性。如果 $\mathbf{x}_i$ 和 $\mathbf{x}_j$ 属于同一类别，则 $s_{ij} = 1$ ；否则为 0。给定相应的哈希值 $\mathbf{h}_i$ 和 $\mathbf{h}_j$ ，$\mathbf{x}_i$ 和 $\mathbf{x}_j$ 之间的相似性概率可以表示为一个似然函数，表示为，

$$p\left(s_{ij} \mid \mathbf{h}_i, \mathbf{h}_j\right) = \begin{cases} \sigma\left(\mathbf{h}_i^\top \mathbf{h}_j\right), & s_{ij} = 1 \\ 1 - \sigma\left(\mathbf{h}_i^\top \mathbf{h}_j\right), & s_{ij} = 0, \end{cases} \tag{3}$$

where, $\sigma(x) = \frac{1}{1+e^{-x}}$ is the sigmoid function. As the dot product $\langle \mathbf{h}_i, \mathbf{h}_j \rangle$ increases, the probability of $p\left(s_{ij} = 1 \left| \mathbf{h}_i, \mathbf{h}_j\right.\right)$ also increases, i.e., $\mathbf{x}_i$ and $\mathbf{x}_j$ belong to the same category. As the dot product decreases, the probability $p\left(s_{ij} = 1 \mid \mathbf{h}_i, \mathbf{h}_j\right)$ also decreases, i.e., $\mathbf{x}_i$ and $\mathbf{x}_j$ belong to different categories. We construct the $(n_s \times n_s)$ similarity matrix $\mathcal{S} = \{s_{ij}\}$, for the source data with the provided labels, where $s_{ij} = 1$ if $\mathbf{x}_i$ and $\mathbf{x}_j$ belong to the same category and 0, otherwise. Let $\mathbf{H} = \{\mathbf{h}_i\}_{i=1}^{n_s}$ be the set of source data hash values. If the elements of $\mathbf{H}$ are assumed to be i.i.d., the negative log likelihood of the similarity matrix $\mathcal{S}$ given $\mathbf{H}$ can be written as,

其中，$\sigma(x) = \frac{1}{1+e^{-x}}$ 是 sigmoid 函数。随着点积 $\langle \mathbf{h}_i, \mathbf{h}_j \rangle$ 的增加，概率 $p\left(s_{ij} = 1 \left| \mathbf{h}_i, \mathbf{h}_j\right.\right)$ 也随之增加，即 $\mathbf{x}_i$ 和 $\mathbf{x}_j$ 属于同一类别。随着点积的减少，概率 $p\left(s_{ij} = 1 \mid \mathbf{h}_i, \mathbf{h}_j\right)$ 也随之减少，即 $\mathbf{x}_i$ 和 $\mathbf{x}_j$ 属于不同类别。我们构建了源数据的相似性矩阵 $(n_s \times n_s)$ $\mathcal{S} = \{s_{ij}\}$ ，对于提供的标签，如果 $\mathbf{x}_i$ 和 $\mathbf{x}_j$ 属于同一类别，则为 $s_{ij} = 1$ ；否则为 0。设 $\mathbf{H} = \{\mathbf{h}_i\}_{i=1}^{n_s}$ 为源数据哈希值的集合。如果假设 $\mathbf{H}$ 的元素是独立同分布的，那么给定 $\mathbf{H}$ 的相似性矩阵 $\mathcal{S}$ 的负对数似然可以写为，

$$\begin{gathered} \min_{\mathbf{H}} \mathcal{L}(\mathbf{H}) = -\log p(\mathcal{S} \mid \mathbf{H}) \\ = -\sum_{s_{ij} \in \mathcal{S}} \left(s_{ij} \mathbf{h}_i^\top \mathbf{h}_j - \log\left(1 + \exp\left(\mathbf{h}_i^\top \mathbf{h}_j\right)\right)\right). \end{gathered}$$

(4)

By minimizing Equation (4), we can determine hash values $\mathbf{H}$ for the source data which are consistent with the similarity matrix $\mathcal{S}$. The hash loss has been used in previous research for supervised hashing [30, 50]. Equation (4) is a discrete optimization problem that is challenging to solve. We introduce a relaxation on the discrete constraint $\mathbf{h}_i \in \{-1, +1\}^d$ by instead solving for $\mathbf{u}_i \in \mathbb{R}^d$, where $u_s = \{\mathbf{u}_i\}_{i=1}^{n_s}$ is the output of the network and $\mathbf{u}_i = \psi(\mathbf{x}_i)$ (the superscript denoting the domain has been dropped for ease of representation). However, the continuous relaxation gives rise to (i) approximation error, when $\langle \mathbf{h}_i, \mathbf{h}_j \rangle$ is substituted with $\langle \mathbf{u}_i, \mathbf{u}_j \rangle$ and,(ii) quantization error, when the resulting real codes $\mathbf{u}_i$ are binarized [50]. We account for the approximation error by having a $\tanh(.)$ as the final activation layer of the neural network, so that the components of $\mathbf{u}_i$ are bounded between -1 and +1 . In addition, we also introduce a quantization loss $\left\|\mathbf{u}_i - \operatorname{sgn}(\mathbf{u}_i)\right\|_2^2$ along the lines of [22], where $\operatorname{sgn}(.)$ is the sign function. The continuous optimization problem for supervised hashing can now be outlined;

通过最小化方程 (4)，我们可以确定与相似性矩阵 $\mathcal{S}$ 一致的源数据的哈希值 $\mathbf{H}$ 。哈希损失在之前的研究中被用于监督哈希 [30, 50]。方程 (4) 是一个离散优化问题，解决起来具有挑战性。我们通过解决 $\mathbf{u}_i \in \mathbb{R}^d$ 来引入对离散约束 $\mathbf{h}_i \in \{-1, +1\}^d$ 的放松，其中 $u_s = \{\mathbf{u}_i\}_{i=1}^{n_s}$ 是网络的输出，$\mathbf{u}_i = \psi(\mathbf{x}_i)$ (上标表示域已被省略以便于表示)。然而，连续放松会导致 (i) 近似误差，当 $\langle \mathbf{h}_i, \mathbf{h}_j \rangle$ 被替换为 $\langle \mathbf{u}_i, \mathbf{u}_j \rangle$ 时，以及 (ii) 量化误差，当结果的实数代码 $\mathbf{u}_i$ 被二值化时 [50]。我们通过将 $\tanh(.)$ 作为神经网络的最终激活层来考虑近似误差，以便 $\mathbf{u}_i$ 的组件被限制在 -1 和 +1 之间。此外，我们还引入了量化损失 $\left\|\mathbf{u}_i - \operatorname{sgn}(\mathbf{u}_i)\right\|_2^2$ ，其思路与 [22] 类似，其中 $\operatorname{sgn}(.)$ 是符号函数。监督哈希的连续优化问题现在可以概述如下；

$$\min_{\mathcal{U}_s} \mathcal{L}(u_s) = -\sum_{s_{ij}\in\mathcal{S}} \left(s_{ij}\mathbf{u}_i^\top \mathbf{u}_j - \log\left(1+\exp\left(\mathbf{u}_i^\top \mathbf{u}_j\right)\right)\right) + \sum_{i=1}^{n_s} \left\|\mathbf{u}_i - \operatorname{sgn}(\mathbf{u}_i)\right\|_2^2 \tag{5}$$

3.3. Unsupervised Hashing for Target Data

3.3. 目标数据的无监督哈希

In the absence of target data labels, we use the similarity measure $\langle \mathbf{u}_i, \mathbf{u}_j \rangle$, to guide the network to learn discriminative hash values for the target data. An ideal target output $\mathbf{u}_i^t$, needs to be similar to many of the source outputs from the j^{th} category $\left(\left\{\mathbf{u}_k^{s_j}\right\}_{k=1}^K\right)$. We assume without loss of generality, K source data points for every category j where, $j \in \{1,\ldots,C\}$ and $\mathbf{u}_k^{s_j}$ is the k^{th} source output from category j . In addition, $\mathbf{u}_i^t$ must be dissimilar to most other source outputs $\mathbf{u}_k^{s_l}$ belonging to a different category $(j \neq l)$. Enforcing similarity with all the K data points makes for a more robust target data category assignment. We outline a probability measure to capture this intuition. Let p_{ij} be the probability that input target data point $\mathbf{x}_i$ is assigned to category j where,

在没有目标数据标签的情况下，我们使用相似性度量 $\langle \mathbf{u}_i, \mathbf{u}_j \rangle$ 来指导网络学习目标数据的区分哈希值。理想的目标输出 $\mathbf{u}_i^t$ 需要与来自 j^{th} 类别 $\left(\left\{\mathbf{u}_k^{s_j}\right\}_{k=1}^K\right)$ 的许多源输出相似。我们假设不失一般性，K 每个类别的源数据点 j ，其中 $j \in \{1,\ldots,C\}$ 和 $\mathbf{u}_k^{s_j}$ 是来自类别 j 的 k^{th} 源输出。此外，$\mathbf{u}_i^t$ 必须与大多数属于不同类别的其他源输出 $\mathbf{u}_k^{s_l}$ 不相似。强制与所有 K 数据点相似，使目标数据类别分配更加稳健。我们概述了一种概率度量来捕捉这一直觉。设 p_{ij} 为输入目标数据点 $\mathbf{x}_i$ 被分配到类别 j 的概率，其中，

$$p_{ij} = \frac{\sum\limits_{k=1}^{K} \exp\left(\mathbf{u}_i^{t\top} \mathbf{u}_k^{s_j}\right)}{\sum\limits_{l=1}^{C}\sum\limits_{k=1}^{K} \exp\left(\mathbf{u}_i^{t\top} \mathbf{u}_k^{s_l}\right)} \tag{6}$$

The exp(.) has been introduced for ease of differentiability and the denominator ensures $\sum_j p_{ij} = 1$. When the target data point output is similar to one category only and dissimilar to all the other categories, the probability vector $\mathbf{p}_i = [p_{i1},\ldots,p_{iC}]^T$ tends to be a one-hot vector. A one-hot vector can be viewed as a low entropy realization of $\mathbf{p}_i$. We can therefore envisage all the $\mathbf{p}_i$ to be one-hot vectors (low entropy probability vectors), where the target data point outputs are similar to source data point outputs in one and only one category. To this end we introduce a loss to capture the entropy of the target probability vectors. The entropy loss for the network outputs is given by,

为了便于求导，引入了 exp(.)，分母确保 $\sum_j p_{ij} = 1$ 。当目标数据点输出仅与一个类别相似而与所有其他类别不相似时，概率向量 $\mathbf{p}_i = [p_{i1},\ldots,p_{iC}]^T$ 倾向于成为一个独热向量。独热向量可以被视为 $\mathbf{p}_i$ 的低熵实现。因此，我们可以设想所有的 $\mathbf{p}_i$ 都是独热向量 (低熵概率向量)，其中目标数据点输出与一个且仅一个类别的源数据点输出相似。为此，我们引入了一种损失来捕捉目标概率向量的熵。网络输出的熵损失为，

$$\mathcal{H}(u_s, u_t) = -\frac{1}{n_t}\sum_{i=1}^{n_t}\sum_{j=1}^{C} p_{ij}\log(p_{ij}) \tag{7}$$

Minimizing the entropy loss gives us probability vectors $\mathbf{p}_i$ that tend to be one-hot vectors, i.e., the target data point outputs are similar to source data point outputs from any one category only. Enforcing similarity with K source data points from a category, guarantees that the hash values are determined based on a common similarity between multiple source category data points and the target data point.

最小化熵损失为我们提供了概率向量 $\mathbf{p}_i$ ，这些向量趋向于一热向量，即目标数据点的输出仅与某一类别的源数据点输出相似。通过强制与某一类别的 K 源数据点相似性，确保哈希值是基于多个源类别数据点与目标数据点之间的共同相似性来确定的。

3.4. Domain Adaptive Hash Network

3.4. 域自适应哈希网络

We propose a model for deep unsupervised domain adaptation based on hashing (DAH) that incorporates unsupervised domain adaptation between the source and the target (1), supervised hashing for the source (5) and unsupervised hashing for the target (7) in a deep convolutional neural network. The DAH network is trained to minimize

我们提出了一种基于哈希的深度无监督域自适应模型 (DAH)，该模型结合了源域与目标域之间的无监督域自适应 (1)、源域的监督哈希 (5) 以及目标域的无监督哈希 (7)，并在深度卷积神经网络中实现。DAH 网络的训练目标是最小化

$$\min_{\mathcal{U}} \mathcal{J} = \mathcal{L}(u_s) + \gamma \mathcal{M}(u_s, u_t) + \eta \mathcal{H}(u_s, u_t) \tag{8}$$

where, $u := \{u_s \cup u_t\}$ and (γ, η) control the importance of domain adaptation (1) and target entropy loss (7) respectively. The hash values $\mathbf{H}$ are obtained from the output of the network using $\mathbf{H} = \mathrm{sgn}(\mathcal{U})$. The loss terms (5) and (7) are determined in the final layer of the network with the network output $\mathcal{U}$. The MK-MMD loss (1) is determined between layer outputs $\{u_s^l, u_t^l\}$ at each of the fully connected layers $\mathcal{F} = \{fc6, fc7, fc8\}$, where we adopt the linear time estimate for the unbiased MK-MMD as described in [24] and [31]. The DAH is trained using standard back-propagation. The detailed derivation of the derivative of (8) w.r.t. $\mathcal{U}$ is provided in the supplementary material.

其中，$u := \{u_s \cup u_t\}$ 和 (γ, η) 分别控制域自适应的重要性 (1) 和目标熵损失 (7)。哈希值 $\mathbf{H}$ 是通过使用 $\mathbf{H} = \mathrm{sgn}(\mathcal{U})$ 从网络输出中获得的。损失项 (5) 和 (7) 是在网络的最终层中根据网络输出 $\mathcal{U}$ 确定的。MK-MMD 损失 (1) 是在每个全连接层的层输出 $\{u_s^l, u_t^l\}$ 之间确定的，我们采用了 [24] 和 [31] 中描述的无偏 MK-MMD 的线性时间估计。DAH 使用标准反向传播进行训练。关于 (8) 相对于 $\mathcal{U}$ 的导数的详细推导见补充材料。

Network Architecture: Owing to the paucity of images in a domain adaptation setting, we circumvent the need to train a deep CNN with millions of images by adapting the pre-trained VGG-F [8] network to the DAH. The VGG-F has been trained on the ImageNet 2012 dataset and it consists of 5 convolution layers (conv1 - conv5) and 3 fully connected layers(fc6, fc7, fc8). We introduce the hashing layer hash-fc8 that outputs vectors in $\mathbb{R}^d$ in the place of fc8. To account for the hashing approximation, we introduced a tanh( ) layer. However, we encounter the issue of vanishing gradients [26] when using tanh( ) as it saturates with large inputs. We therefore preface the tanh( ) with a batch normalization layer which prevents the tanh( ) from saturating. In effect, hash-fc8 $:= \{fc8 \to \text{batch-norm} \to \tanh()\}$. The hash-fc8 provides greater stability when fine-tuning the learning rates than the deep hashing networks [30, 50]. Figure 1 illustrates the proposed DAH network.

网络架构: 由于在领域适应设置中图像数量稀缺，我们通过将预训练的 VGG-F [8] 网络适应于 DAH，避免了使用数百万张图像训练深度 CNN 的需求。VGG-F 已在 ImageNet 2012 数据集上进行训练，包含 5 个卷积层 (conv1 - conv5) 和 3 个全连接层 (fc6, fc7, fc8)。我们引入了哈希层 hash-fc8，它在 fc8 的位置输出向量 $\mathbb{R}^d$ 。为了考虑哈希近似，我们引入了一个 tanh( ) 层。然而，当使用 tanh( ) 时，我们遇到了梯度消失的问题 [26]，因为它在大输入时会饱和。因此，我们在 tanh( ) 前加上了一个批量归一化层，以防止 tanh( ) 饱和。实际上，hash-fc8 $:= \{fc8 \to \text{batch-norm} \to \tanh()\}$ 。与深度哈希网络 [30, 50] 相比，hash-fc8 在微调学习率时提供了更大的稳定性。图 1 说明了所提出的 DAH 网络。

Figure 2: Sample images from the Office-Home dataset. The dataset consists of images of everyday objects organized into 4 domains; Art: paintings, sketches and/or artistic depictions, Clipart: clipart

images, Product: images without background and Real-World: regular images captured with a camera. The figure displays examples from 16 of the 65 categories.

图 2:Office-Home 数据集的样本图像。该数据集包含日常物品的图像，分为 4 个领域；艺术: 绘画、素描和/或艺术表现，剪贴画: 剪贴画图像，产品: 无背景的图像和现实世界: 用相机拍摄的常规图像。该图展示了 65 个类别中的 16 个类别的示例。

Table 1: Statistics for the Office-Home dataset. Min: # is the minimum number of images amongst all the categories, Min: Size and Max: Size are the minimum and maximum image sizes across all categories and Acc. is the classification accuracy.

表 1:Office-Home 数据集的统计数据。Min:# 是所有类别中图像的最小数量，Min:Size 和 Max:Size 是所有类别中图像的最小和最大尺寸，Acc. 是分类准确率。

Domain.	Min: #	Min: Size	Max: Size	Acc
Art	15	117×85 pix.	4384×2686 pix.	44.99±1.85
Clipart	39	18 × 18 pix.	2400×2400 pix.	53.95±1.45
Product	38	75 × 63 pix.	2560×2560 pix.	66.41 ± 1.18
Real-World	23	88 × 80 pix.	6500 × 4900 pix.	59.70±1.04

领域。	最小值:#	最小值: 大小	最大值: 大小	准确率
艺术	15	117×85 像素	4384×2686 像素	44.99±1.85
剪贴画	39	18 × 18 像素	2400×2400 像素	53.95±1.45
产品	38	75 × 63 像素	2560×2560 像素	66.41 ± 1.18
真实世界	23	88 × 80 像素。	6500 × 4900 像素。	59.70±1.04

4.The Office-Home Dataset

4. Office-Home 数据集

Supervised deep learning models require a large volume of labeled training data. Unfortunately, existing datasets for vision-based domain adaptation are limited in their size and are not suitable for validating deep learning algorithms. The standard datasets for vision based domain adaptation are, facial expression datasets CKPlus [35] and MMI [40], digit datasets SVHN [36], USPS and MNIST[28], head pose recognition datasets PIE [33], object recognition datasets COIL[33], Office [42] and Office-Caltech [20]. These datasets were created before deep-learning became popular and are insufficient for training and evaluating deep learning based domain adaptation approaches. For instance, the object-recognition dataset Office has 4110 images across 31 categories and Office-Caltech has 2533 images across 10 categories.

监督深度学习模型需要大量标记的训练数据。不幸的是，现有的基于视觉的领域适应数据集在规模上有限，并不适合验证深度学习算法。标准的基于视觉的领域适应数据集包括面部表情数据集 CKPlus [35] 和 MMI [40]，数字数据集 SVHN [36]、USPS 和 MNIST [28]，头部姿态识别数据集 PIE [33]，物体识别数据集 COIL [33]、Office [42] 和 Office-Caltech [20]。这些数据集是在深度学习流行之前创建的，无法满足训练和评估基于深度学习的领域适应方法的需求。例如，物体识别数据集 Office 包含 31 个类别的 4110 张图像，而 Office-Caltech 包含 10 个类别的 2533 张图像。

We release the Office-Home dataset for domain adaptation based object recognition, that can be used to evaluate deep learning algorithms for domain adaptation. The Office-Home dataset consists of 4 domains, with each domain containing images from 65 categories of everyday objects and a total of around 15,500 images. The domains include, Art: artistic depictions of objects in the form of sketches, paintings, ornamentation, etc.; Clipart: collection of clipart images; Product: images of objects without a background, akin to the Amazon category in Office dataset; Real-World: images of objects captured with a regular camera.

我们发布了 Office-Home 数据集，用于基于领域适应的物体识别，可以用于评估领域适应的深度学习算法。Office-Home 数据集由 4 个领域组成，每个领域包含 65 个日常物体类别的图像，总计约 15,500 张图像。这些领域包括: 艺术: 以素描、绘画、装饰等形式表现物体的艺术作品；剪贴画: 剪贴画图像的集合；产品: 没有背景的物体图像，类似于 Office 数据集中的亚马逊类别；现实世界: 用普通相机拍摄的物体图像。

Public domain images were downloaded from web-sites like www.deviantart.com and www.flickr.com to create the Art and Real-World domains. Clipart images were gathered from multiple clipart websites. The Product domain images were exclusively collected from www.amazon.com using web-crawlers. The collected images were manually filtered on the basis of quality, size and content. The dataset has an average of around 70 images per category and a maximum of 99 images in a category. The primary challenge in creating this dataset was acquiring sufficient number of public domain images across all the 4 domains. Figure 2 depicts a sampling of 16 categories from the Office-Home dataset and Table 1 outlines some meta data for the dataset. The Acc. column in the Table 1 refers to classification accuracies using the LIBLINEAR SVM [15] classifier (5-fold cross validation) with deep features extracted using the VGG-F network. The dataset is publicly available for research [1] .

从 www.deviantart.com 和 www.flickr.com 等网站下载了公共领域图像，以创建艺术和现实世界领域。剪贴画图像来自多个剪贴画网站。产品领域的图像则专门从 www.amazon.com 收集，使用网络爬虫进行获取。收集的图像根据质量、大小和内容进行了手动筛选。该数据集每个类别平均约有 70 张图像，最多可达 99 张图像。创建该数据集的主要挑战是获取足够数量的公共领域图像，涵盖所有四个领域。图 2 展示了 Office-Home 数据集中的 16 个类别的样本，表 1 列出了该数据集的一些元数据。表 1 中的 Acc. 列指的是使用 LIBLINEAR SVM [15] 分类器 (5 倍交叉验证) 和使用 VGG-F 网络提取的深度特征进行的分类准确率。该数据集可公开用于研究 [1] 。

5. Experiments

5. 实验

In this section we conduct extensive experiments to evaluate the DAH algorithm. Since we propose a domain adaptation technique based on hashing, we evaluate objection recognition accuracies for unsupervised domain adaptation and also study the discriminatory capability of the learned hash codes for unsupervised domain adaptive hashing. The implementation details are available at

在本节中，我们进行广泛的实验以评估 DAH 算法。由于我们提出了一种基于哈希的领域适应技术，因此我们评估无监督领域适应的对象识别准确率，并研究学习到的哈希码在无监督领域自适应哈希中的区分能力。实施细节可在此处获得。https://github.com/hemanthdv/da-hash

5.1. Datasets

5.1. 数据集

Office [42]: This is currently the most popular benchmark dataset for object recognition in the domain adaptation computer vision community. The dataset consists of images of everyday objects in an office environment. It has 3 domains; Amazon (A), Dslr (D) and Webcam (W). The dataset has around 4,100 images with a majority of the images (2816 images) in the Amazon domain. We adopt the common evaluation protocol of different pairs of transfer tasks for this dataset [31, 34] . We consider 6 transfer tasks for all combinations of source and target pairs for the 3 domains. Office-Home: We introduce this new dataset and evaluate it in a similar manner to the Office dataset. We consider 12 transfer tasks for the Art (Ar), Clipart (Cl), Product (Pr) and Real-World (Rw) domains for all combinations of source and target for the 4 domains. Considering all the different pairs of transfer enables us to evaluate the inherent bias between the domains in a comprehensive manner [45].

办公室 [42]: 这是目前在领域适应计算机视觉社区中最受欢迎的对象识别基准数据集。该数据集包含办公环境中日常物品的图像。它有 3 个领域: 亚马逊 (A)、单反相机 (D) 和网络摄像头 (W)。该数据集大约有 4100 张图像，其中大多数图像 (2816 张) 位于亚马逊领域。我们采用该数据集的不同传输任务对的常见评估协议 [31, 34] 。我们考虑 3 个领域的源和目标对的所有组合，共有 6 个传输任务。Office-Home: 我们引入这个新数据集，并以类似于 Office 数据集的方式进行评估。我们考虑艺术 (Ar)、剪贴画 (Cl)、产品 (Pr) 和真实世界 (Rw) 领域的所有源和目标组合，共有 12 个传输任务。考虑所有不同的传输对使我们能够以全面的方式评估领域之间的固有偏差 [45]。

[1] https://hemanthdv.github.io/officehome-dataset/
[1] https://hemanthdv.github.io/officehome-dataset/

5.2. Implementation Details

5.2. 实施细节

We implement the DAH using the MatConvnet framework [47]. Since we train a pre-trained VGG-F, we fine-tune the weights of conv1-conv5, fc6 and fc7. We set their learning rates to $1/10^{th}$ the learning rate of hash-fc8. We vary the learning rate between 10^{-4} to 10^{-5} over 300 epochs with a momentum 0.9 and weight decay 5×10^{-4} . We set $K = 5$ (number of samples from a category). Since we have 31 categories in the Office dataset, we get a source batch size of $31 \times 5 = 155$. For the target batch, we randomly select 155 samples. The total batch size turns out to be 310. For the Office-Home dataset, with $K = 5$ and 65 categories, we get a batch size of 650 . We set $d = 64$ (hash code length) for all our experiments. Since there is imbalance in the number of like and unlike pairs in $\mathcal{S}$, we set the values in similarity matrix $\mathcal{S}_{i,j} \in \{0, 10\}$. Increasing the similarity weight of like-pairs improves the performance of DAH. For the entropy loss, we set $\eta = 1$. For the MK-MMD loss, we follow the heuristics mentioned in [24], to determine the parameters. We estimate γ , by validating a binary domain classifier to distinguish between source and target data points and select γ which gives largest error on a validation set. For MMD, we use a Gaussian kernel with a bandwidth σ given by the median of the pairwise distances in the training data. To incorporate the multi-kernel, we vary the bandwidth $\sigma_m \in \left[2^{-8}\sigma, 2^{8}\sigma\right]$ with a multiplicative factor of 2 . We define the target classifier $f\left(\mathbf{x}_i^t\right) = p\left(y \mid \mathbf{h}_i^t\right)$ in terms of 6 . The target data point is assigned to the class with the largest probability, with $\widehat{y}_i = \max_j \left(p_{ij}\right)$ using the hash codes for the source and the target.

我们使用 MatConvnet 框架实现 DAH [47]。由于我们训练的是预训练的 VGG-F，因此我们微调 conv1-conv5、fc6 和 fc7 的权重。我们将它们的学习率设置为 $1/10^{th}$ ，即 hash-fc8 的学习率。我们在 300 个周期内将学习率在 10^{-4} 和 10^{-5} 之间变化，动量为 0.9，权重衰减为 5×10^{-4} 。我们设置 $K = 5$ (来自一个类别的样本数量)。由于 Office 数据集中有 31 个类别，我们得到的源批量大小为 $31 \times 5 = 155$ 。对于目标批量，我们随机选择 155 个样本。总批量大小为 310。对于 Office-Home 数据集，设置 $K = 5$ 和 65 个类别，我们得到的批量大小为 650。我们为所有实验设置 $d = 64$ (哈希码长度)。由于 $\mathcal{S}$ 中相似对和不相似对的数量不平衡，我们在相似性矩阵中设置值 $\mathcal{S}_{i,j} \in \{0, 10\}$ 。增加相似对的相似性权重可以提高 DAH 的性能。对于熵损失，我们设置 $\eta = 1$ 。对于 MK-MMD 损失，我们遵循 [24] 中提到的启发式方法来确定参数。我们通过验证一个二元领域分类器来区分源数据点和目标数据点，估计 γ ，并选择在验证集上产生最大误差的 γ 。对于 MMD，我们使用带宽为 σ 的高斯核，该带宽由训练数据中的成对距离的中位数给出。为了结合多核，我们将带宽 $\sigma_m \in \left[2^{-8}\sigma, 2^{8}\sigma\right]$ 乘以一个因子 2 进行变化。我们将目标分类器 $f\left(\mathbf{x}_i^t\right) = p\left(y \mid \mathbf{h}_i^t\right)$ 定义为 6。目标数据点被分配给具有最大概率的类别，使用 $\widehat{y}_i = \max_j \left(p_{ij}\right)$ 的哈希码来区分源和目标。

5.3. Unsupervised Domain Adaptation

5.3. 无监督领域适应

In this section, we study the performance of the DAH for unsupervised domain adaptation, where labeled data is available only in the source domain and no labeled data is available in the target domain. We compare the DAH with state-of-the-art domain adaptation methods: (i) Geodesic Flow Kernel (GFK) [20], (ii) Transfer Component Analysis (TCA) [38], (iii) Correlation Alignment (CORAL) [44] and (iv) Joint Distribution Adaptation (JDA) [33]. We also compare the DAH with state-of-the-art deep learning methods for domain adaptation: (v) Deep Adaptation Network (DAN) [31] and (vi) Domain Adversarial Neural Network (DANN) [17]. For all of the shallow learning methods, we extract and use deep features from the fc7 layer of the VGG-F network that was pre-trained on the ImageNet 2012 dataset. We also evaluate the effect of the entropy loss on hashing for the DAH. The DAH-e is the DAH algorithm where η is set to zero, which implies that the target hash values are not driven to align with the source categories. We follow the standard protocol for unsupervised domain adaptation, where all the labeled source data and all the unlabeled target data is used for training.

在本节中，我们研究了 DAH 在无监督领域适应中的性能，其中标记数据仅在源领域可用，而目标领域没有标记数据。我们将 DAH 与最先进的领域适应方法进行比较:(i) 流形流核 (GFK) [20]，(ii) 转移成分分析 (TCA) [38]，(iii) 相关性对齐 (CORAL) [44] 和 (iv) 联合分布适应 (JDA) [33]。我们还将 DAH 与最先进的深度学习领域适应方法进行比较:(v) 深度适应网络 (DAN) [31] 和 (vi) 领域对抗神经网络 (DANN) [17]。对于所有浅层学习方法，我们从在 ImageNet 2012 数据集上预训练的 VGG-F 网络的

fc7 层提取并使用深度特征。我们还评估了熵损失对 DAH 哈希的影响。DAH-e 是 DAH 算法，其中 η 设置为零，这意味着目标哈希值不会被驱动与源类别对齐。我们遵循无监督领域适应的标准协议，其中所有标记的源数据和所有未标记的目标数据用于训练。

Table 2: Recognition accuracies (%) for domain adaptation experiments on the Office dataset. {Amazon (A), Dslr (D), Webcam (W)}.A $\rightarrow$ W implies A is source and W is target.

表 2: 在 Office 数据集上进行领域适应实验的识别准确率 (%)。{亚马逊 (A)，单反相机 (D)，网络摄像头 (W)}.A $\rightarrow$ W 意味着 A 是源，W 是目标。

Expt.	**A → D**	**A → W**	**D → A**	**D → W**	**W → A**	**W → D**	Avg.
GFK	48.59	52.08	41.83	89.18	49.04	93.17	62.32
TCA	51.00	49.43	48.12	93.08	48.83	96.79	64.54
CORAL	54.42	51.70	48.26	95.97	47.27	98.59	66.04
JDA	59.24	58.62	51.35	96.86	52.34	97.79	69.37
DAN	67.04	67.80	50.36	95.85	52.33	99.40	72.13
DANN	72.89	72.70	56.25	96.48	53.20	99.40	75.15
DAH-e	66.27	66.16	55.97	94.59	53.91	96.99	72.31
DAH	66.47	68.30	55.54	96.10	53.02	98.80	73.04

实验。	**A → D**	**A → W**	**D → A**	**D → W**	**W → A**	**W → D**	平均。
GFK	48.59	52.08	41.83	89.18	49.04	93.17	62.32
TCA	51.00	49.43	48.12	93.08	48.83	96.79	64.54
CORAL	54.42	51.70	48.26	95.97	47.27	98.59	66.04
JDA	59.24	58.62	51.35	96.86	52.34	97.79	69.37
DAN	67.04	67.80	50.36	95.85	52.33	99.40	72.13
DANN	72.89	72.70	56.25	96.48	53.20	99.40	75.15
DAH-e	66.27	66.16	55.97	94.59	53.91	96.99	72.31
DAH	66.47	68.30	55.54	96.10	53.02	98.80	73.04

Results and Discussion: The results are reported for the target classification in each of the transfer tasks in Tables 2 and 3, where accuracies denote the percentage of correctly classified target data samples. We present results with hash length $d = 64$ bits. The DAH algorithm consistently outperforms the baselines across all the domains for the Office-Home dataset. However, DANN marginally surpasses DAH for the Office dataset, prompting us to reason that domain adversarial training is more effective than DAH when the categories are fewer in number. Since domain alignment is category agnostic, it is possible that the aligned domains are not classification friendly in the presence of large number of categories. When the number of categories is large, as in Office-Home, DAH does best at extracting transferable features to achieve higher accuracies. We also note that DAH delivers better performance than DAH-e; thus, minimizing the entropy on the target data through 7 aids in improved alignment of the source and target samples, which boosts the accuracy.

结果与讨论: 在表 2 和表 3 中报告了每个迁移任务中目标分类的结果，其中准确率表示正确分类的目标数据样本的百分比。我们展示了哈希长度为 $d = 64$ 位的结果。DAH 算法在 Office-Home 数据集的所有领域中始终优于基线。然而，DANN 在 Office 数据集上略微超越了 DAH，这使我们推测当类别数量较少时，领域对抗训练比 DAH 更有效。由于领域对齐与类别无关，因此在类别数量较多的情况下，对齐的领域可能不利于分类。当类别数量较多时，例如在 Office-Home 中，DAH 在提取可迁移特征以实现更高准确率方面表现最佳。我们还注意到，DAH 的性能优于 DAH-e；因此，通过 7 最小化目标数据的熵有助于改善源样本和目标样本的对齐，从而提高准确率。

Feature Analysis: We also study the feature representations of the penultimate layer (fc7) outputs using t-SNE em-beddings as in [12]. Figure 3a depicts the $\mathcal{A}$ -distance between domain pairs using Deep (VGG-F), DAN and DAH features. Ben-David et al. [2] defined $\mathcal{A}$ -distance as the distance between two domains that can be viewed as the discrepancy between two domains. Although it is difficult to estimate its exact value, an approximate distance measure is given by $2\left(1-2\epsilon\right)$, where ϵ is the generalization error for a binary classifier trained to distinguish between the two domains. We used a LIBLINEAR SVM [15] classifier with 5-fold cross-validation to estimate ϵ . Figure 3a indicates that the DAH features have the least discrepancy between the source and target compared to DAN and Deep features. This is also confirmed with the t-SNE embeddings in Figures 3b-3d. The Deep features show very little overlap between the domains and the categories depict minimal clustering. Domain overlap and clustering improves as we move to DAN and DAH features, with DAH providing the best visualizations. This corroborates the efficacy of the DAH algorithm to exploit the feature learning capabilities of deep neural networks to learn representative hash codes to address domain adaptation.

特征分析: 我们还使用 t-SNE 嵌入研究倒数第二层 (fc7) 输出的特征表示，如 [12] 所示。图 3a 描述了使用深度 (VGG-F)、DAN 和 DAH 特征的领域对之间的 $\mathcal{A}$ -距离。Ben-David 等人 [2] 定义了 $\mathcal{A}$ -距离为可以视为两个领域之间差异的两个领域之间的距离。虽然很难估计其确切值，但近似距离度量由 $2(1-2\epsilon)$ 给出，其中 ϵ 是训练用于区分两个领域的二元分类器的泛化误差。我们使用 LIBLINEAR SVM [15] 分类器进行 5 倍交叉验证来估计 ϵ 。图 3a 表明，与 DAN 和 Deep 特征相比，DAH 特征在源和目标之间的差异最小。图 3b-3d 中的 t-SNE 嵌入也证实了这一点。Deep 特征在领域之间几乎没有重叠，类别表现出最小的聚类。随着我们转向 DAN 和 DAH 特征，领域重叠和聚类有所改善，DAH 提供了最佳的可视化。这证实了 DAH 算法利用深度神经网络的特征学习能力来学习代表性哈希码以解决领域适应问题的有效性。

Table 3: Recognition accuracies (%) for domain adaptation experiments on the Office-Home dataset. {Art (Ar), Clipart (Cl), Product (Pr), Real-World (Rw)}. Ar $\rightarrow$ Cl implies Ar is source and Cl is target.

表 3:Office-Home 数据集上领域适应实验的识别准确率 (%)。{艺术 (Ar)、剪贴画 (Cl)、产品 (Pr)、现实世界 (Rw)}。Ar $\rightarrow$ Cl 表示 Ar 是源，Cl 是目标。

Expt.	Ar → Cl	**Ar → Pr**	Ar→Rw	**Cl → Ar**	Cl → Pr	**Cl → Rw**	Pr → Ar	Pr→Cl	Pr → Rw	**Rw → Ar**	Rw → Cl	**Rw → Pr**	Avg.
GFK	21.60	31.72	38.83	21.63	34.94	34.20	24.52	25.73	42.92	32.88	28.96	50.89	32.40
TCA	19.93	32.08	35.71	19.00	31.36	31.74	21.92	23.64	42.12	30.74	27.15	48.68	30.34
CORAL	27.10	36.16	44.32	26.08	40.03	40.33	27.77	30.54	50.61	38.48	36.36	57.11	37.91
JDA	25.34	35.98	42.94	24.52	40.19	40.90	25.96	32.72	49.25	35.10	35.35	55.35	36.97
DAN	30.66	42.17	54.13	32.83	47.59	49.78	29.07	34.05	56.70	43.58	38.25	62.73	43.46
DANN	33.33	42.96	54.42	32.26	49.13	49.76	30.49	38.14	56.76	44.71	42.66	64.65	44.94
DAH-e	29.23	35.71	48.29	33.79	48.23	47.49	29.87	38.76	55.63	41.16	44.99	59.07	42.69
DAH	31.64	40.75	51.73	34.69	51.93	52.79	29.91	39.63	60.71	44.99	45.13	62.54	45.54

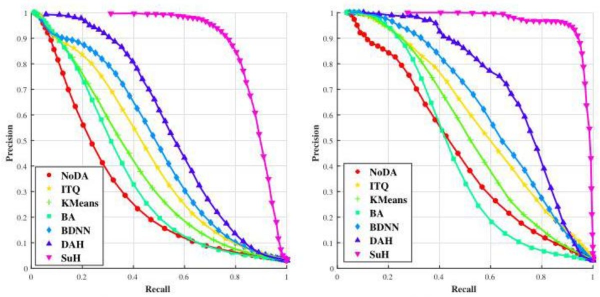
实验	Ar → Cl	**Ar → Pr**	Ar→Rw	**Cl → Ar**	Cl → Pr	**Cl → Rw**	Pr → Ar	Pr→Cl	Pr → Rw	**Rw → Ar**	Rw → Cl	**Rw → Pr**	平均值
GFK	21.60	31.72	38.83	21.63	34.94	34.20	24.52	25.73	42.92	32.88	28.96	50.89	32.40
TCA	19.93	32.08	35.71	19.00	31.36	31.74	21.92	23.64	42.12	30.74	27.15	48.68	30.34
CORAL	27.10	36.16	44.32	26.08	40.03	40.33	27.77	30.54	50.61	38.48	36.36	57.11	37.91
JDA	25.34	35.98	42.94	24.52	40.19	40.90	25.96	32.72	49.25	35.10	35.35	55.35	36.97
DAN	30.66	42.17	54.13	32.83	47.59	49.78	29.07	34.05	56.70	43.58	38.25	62.73	43.46
DANN	33.33	42.96	54.42	32.26	49.13	49.76	30.49	38.14	56.76	44.71	42.66	64.65	44.94
DAH-e	29.23	35.71	48.29	33.79	48.23	47.49	29.87	38.76	55.63	41.16	44.99	59.07	42.69
DAH	31.64	40.75	51.73	34.69	51.93	52.79	29.91	39.63	60.71	44.99	45.13	62.54	45.54

5.4. Unsupervised Domain Adaptive Hashing

5.4. 无监督领域自适应哈希

In this section, we study the performance of our algorithm to generate compact and efficient hash codes from the data for classifying unseen test instances, when no labels are available. This problem has been addressed in the literature, with promising empirical results [7, 11, 21] . However, in a real-world setting, labels may be available from a different, but related (source) domain; a strategy to utilize the labeled data from the source domain to learn representative hash codes for the target domain is therefore of immense practical importance. Our work is the first to identify and address this problem. We consider the following scenarios to address this real-world challenge: (i) No labels are available for a given dataset and the hash codes need to be learned in a completely unsupervised manner. We evaluate against baseline unsupervised hashing methods (ITQ) [22] and (KMeans) [25] and also state-of-the-art methods for unsupervised hashing (BA) [7] and (BDNN) [11]. (ii) Labeled data is available from a different, but related source domain. A hashing model is trained on the labeled source data and is used to learn hash codes for the target data. We refer to this method as NoDA, as no domain adaptation is performed. We used the deep pairwise-supervised hashing (DPSH) algorithm [30] to train a deep network with the source data and applied the network to generate hash codes for the target data. (iii) Labeled data is available from a different, but related source domain and we use our DAH formulation to learn hash codes for the target domain by

在本节中，我们研究了我们的算法在没有标签可用的情况下，从数据中生成紧凑且高效的哈希码以对未见测试实例进行分类的性能。文献中已经解决了这个问题，并取得了有希望的实证结果 [7, 11, 21] 。然而，在实际应用中，标签可能来自不同但相关的 (源) 领域；因此，利用源领域的标记数据来学习目标领域的代表性哈希码的策略具有重要的实际意义。我们的工作是首次识别并解决这个问题。我们考虑以下场景来应对这一现实挑战:(i) 给定数据集没有可用标签，哈希码需要以完全无监督的方式学习。我们与基线无监督哈希方法 (ITQ) [22] 和 (KMeans) [25] 进行评估，并与无监督哈希的最先进方法 (BA) [7] 和 (BDNN) [11] 进行比较。(ii) 来自不同但相关的源领域的标记数据可用。一个哈希模型在标记的源数据上进行训练，并用于学习目标数据的哈希码。我们将这种方法称为 NoDA，因为没有进行领域适应。我们使用深度成对监督哈希 (DPSH) 算法 [30] 在源数据上训练深度网络，并应用该网络生成目标数据的哈希码。(iii) 来自不同但相关的源领域的标记数据可用，我们使用我们的 DAH 公式通过



(a) Amazon **(b)** Webcam

Figure 5: Precision-Recall curves @ 64 bits for the Office dataset. Comparison of hashing without domain adaptation (NoDA), shallow unsupervised hashing (ITQ, KMeans), state-of-the-art deep unsupervised hashing (BA, BDNN), unsupervised domain adaptive hashing (DAH) and supervised hashing (SuH). Best viewed in color.

图 5:Office 数据集在 64 位下的精确率-召回曲线。比较不进行领域适应的哈希 (NoDA)、浅层无监督哈希 (ITQ, KMeans)、最先进的深度无监督哈希 (BA, BDNN)、无监督领域适应哈希 (DAH) 和监督哈希 (SuH)。最佳效果请在彩色下查看。

Table 4: Mean average precision @64 bits. For the NoDA and DAH results, Art is the source domain for Clipart, Product and Real-World and Clipart is the source domain for Art. Similarly, Amazon and Webcam are source target pairs. reducing domain disparity. (iv) Labeled data is available in the target domain. This method falls under supervised hashing (SuH) (as it uses labeled data in the target domain to learn hash codes in the same domain) and denotes the upper bound on the performance. It is included to compare the performance of unsupervised hashing algorithms relative to the supervised algorithm. We used the DPSH algorithm [30] to train a deep network on the target data and used it to generate hash codes on a validation subset.

表 4: 在 64 位下的平均精确率。对于 NoDA 和 DAH 结果，Art 是 Clipart、Product 和 Real-World 的源领域，而 Clipart 是 Art 的源领域。同样，Amazon 和 Webcam 是源目标对。减少领域差异。(iv) 目标领域中有标记数据可用。该方法属于监督哈希 (SuH)(因为它使用目标领域中的标记数据来学习同一领域的哈希码)，并表示性能的上限。它被纳入以比较无监督哈希算法相对于监督算法的性能。我们使用 DPSH 算法 [30] 在目标数据上训练深度网络，并用它生成验证子集上的哈希码。

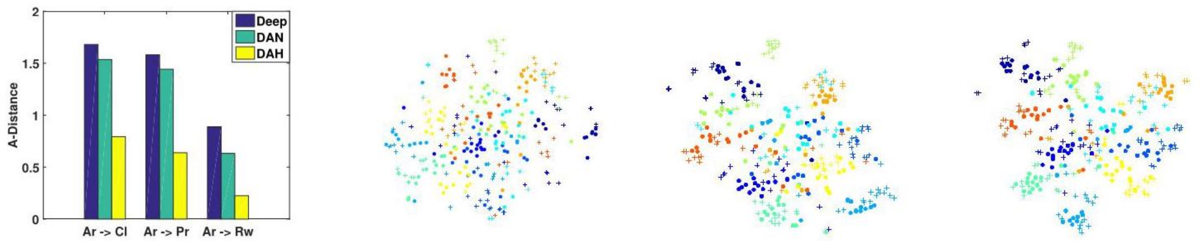
Expt.	NoDA	ITQ	KMeans	BA	BDNN	DAH	SuH
Amazon	0.324	0.465	0.403	0.367	0.491	0.582	0.830
Webcam	0.511	0.652	0.558	0.480	0.656	0.717	0.939
Art	0.155	0.191	0.170	0.156	0.193	0.302	0.492
Clipart	0.160	0.195	0.178	0.179	0.206	0.333	0.622
Product	0.239	0.393	0.341	0.349	0.407	0.414	0.774
Real-World	0.281	0.323	0.279	0.273	0.336	0.533	0.586
Avg.	0.278	0.370	0.322	0.301	0.382	0.480	0.707

实验	NoDA	ITQ	KMeans	BA	BDNN	DAH	SuH
亚马逊	0.324	0.465	0.403	0.367	0.491	0.582	0.830
网络摄像头	0.511	0.652	0.558	0.480	0.656	0.717	0.939
艺术	0.155	0.191	0.170	0.156	0.193	0.302	0.492
剪贴画	0.160	0.195	0.178	0.179	0.206	0.333	0.622
产品	0.239	0.393	0.341	0.349	0.407	0.414	0.774
现实世界	0.281	0.323	0.279	0.273	0.336	0.533	0.586
平均值	0.278	0.370	0.322	0.301	0.382	0.480	0.707

Results and Discussion: We applied the precision-recall curves and the mean average precision (mAP) measures to evaluate the efficacy of the hashing methods, similar to previous research [7, 11, 21] . The results are depicted in Figures 4 and 5 (precision-recall curves) and Table 4 (mAP values), where we present hashing with code length $d = 64$ bits. Hashing performance with $d = 16$ bits also follows a similar trend and is presented in the supplementary material. For the sake of brevity, we drop the results with Dslr as it is very similar to Webcam, with little domain difference. We note that the NoDA has the poorest performance due to domain mismatch. This demonstrates that domain disparity needs to be considered before deploying a hashing network to extract hash codes. The unsupervised hashing methods

ITQ, KMeans, BA and BDNN perform slightly better compared to NoDA. The proposed DAH algorithm encompasses hash code learning and domain adaptation in a single integrated framework. It is thus able to leverage the labeled data in the source domain in a meaningful manner to learn efficient hash codes for the target domain. This accounts for its improved performance, as is evident in Figures 4 and 5 and Table 4. The supervised hashing technique (SuH) uses labels from the target and therefore depicts the best performance. The proposed DAH framework consistently delivers the best performance relative to SuH when compared with the other hashing procedures. This demonstrates the merit of our framework in learning representative hash codes by utilizing labeled data from a different domain. Such a framework will be immensely useful in a real-world setting.

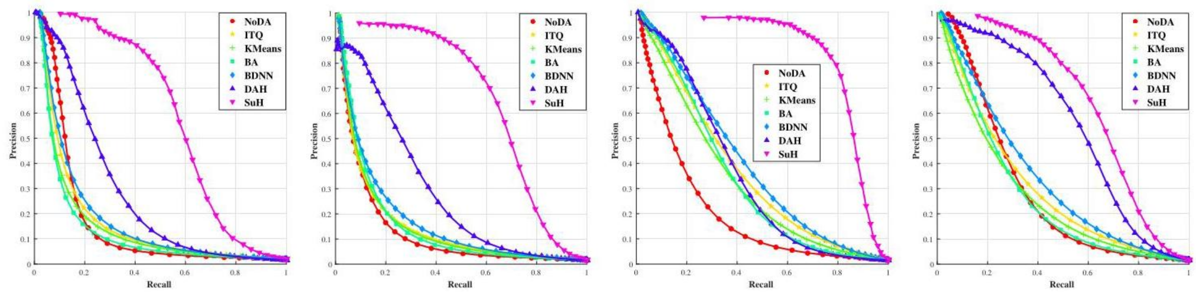
结果与讨论: 我们应用了精确率-召回率曲线和平均精确度均值 (mAP) 指标来评估哈希方法的有效性，类似于之前的研究 [7, 11, 21] 。结果如图 4 和图 5(精确率-召回率曲线) 以及表 4(mAP 值) 所示，其中我们展示了哈希代码长度为 $d = 64$ 位的情况。哈希性能在 $d = 16$ 位时也遵循类似的趋势，并在补充材料中呈现。为了简洁起见，我们省略了 Dslr 的结果，因为它与 Webcam 非常相似，领域差异很小。我们注意到，由于领域不匹配，NoDA 的性能最差。这表明在部署哈希网络以提取哈希代码之前，需要考虑领域差异。无监督哈希方法 ITQ、KMeans、BA 和 BDNN 的表现略优于 NoDA。所提出的 DAH 算法在一个集成框架中涵盖了哈希代码学习和领域适应。因此，它能够以有意义的方式利用源领域中的标记数据，为目标领域学习有效的哈希代码。这解释了其改进的性能，正如图 4、图 5 和表 4 所示。监督哈希技术 (SuH) 使用来自目标的标签，因此表现最佳。所提出的 DAH 框架在与其他哈希程序的比较中，始终相对 SuH 提供最佳性能。这证明了我们框架在利用来自不同领域的标记数据学习代表性哈希代码方面的优越性。这样的框架在实际应用中将极为有用。



(a) $\mathcal{A}$-Distance **(b)** Deep Features (**Ar**,**Cl**) **(c)** DAN Features (**Ar**,**Cl**) **(d)** DAH Features (**Ar**,**Cl**)

Figure 3: Feature analysis of fc7 layer. (a) $\mathcal{A}$ -distances for Deep, DAN and DAH,(b),(c) and (d) t-SNE embeddings for 10 categories from Art ( ) and Clipart(+) domains. Best viewed in color.

图 3:fc7 层的特征分析。(a) $\mathcal{A}$ -深度、DAN 和 DAH 的距离，(b)、(c) 和 (d) 来自艺术 ( ) 和剪贴画 (+) 领域的 10 个类别的 t-SNE 嵌入。最佳查看效果为彩色。



(a) Art **(b)** Clipart **(c)** Product **(d)** Real-World

Figure 4: Precision-Recall curves @64 bits for the Office-Home dataset. Comparison of hashing without domain adaptation (NoDA), shallow unsupervised hashing (ITQ, KMeans), state-of-the-art deep unsupervised hashing (BA, BDNN), unsupervised domain adaptive hashing (DAH) and supervised hashing (SuH). Best viewed in color.

图 4:Office-Home 数据集在 64 位下的精确度-召回曲线。比较没有领域适应 (NoDA)、浅层无监督哈希 (ITQ，KMeans)、最先进的深度无监督哈希 (BA，BDNN)、无监督领域自适应哈希 (DAH) 和监督哈希 (SuH)。最佳效果请在彩色下查看。

6. Conclusions

6. 结论

In this paper, we have proposed a novel domain adaptive hashing (DAH) framework which exploits the feature learning capabilities of deep neural networks to learn efficient hash codes for unsupervised domain adaptation. The DAH framework solves two important practical problems: category assignment with weak supervision or insufficient labels (through domain adaptation) and the estimation of hash codes in an unsupervised setting (hash codes for target data). Thus, two practical challenges are addressed through a single integrated framework. This research is the first of its kind to integrate hash code learning with unsupervised domain adaptation. We also introduced a new dataset, Office-Home, which can be used to further research in domain adaptation.

在本文中，我们提出了一种新颖的领域自适应哈希 (DAH) 框架，该框架利用深度神经网络的特征学习能力，为无监督领域适应学习有效的哈希码。DAH 框架解决了两个重要的实际问题: 通过领域适应进行弱监督或标签不足的类别分配，以及在无监督环境中估计哈希码 (针对目标数据的哈希码)。因此，通过一个集成框架解决了两个实际挑战。这项研究是首个将哈希码学习与无监督领域适应相结合的研究。我们还引入了一个新的数据集 Office-Home，可用于进一步研究领域适应。

Acknowledgements: This material is based upon work supported by the National Science Foundation (NSF) under Grant No:1116360. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

致谢: 本材料基于国家科学基金会 (NSF) 资助的工作，资助编号:1116360。本文中表达的任何观点、发现、结论或建议均为作者的观点，并不一定反映 NSF 的观点。

References

参考文献

[1] Y. Aytar and A. Zisserman. Tabula rasa: Model transfer for object category detection. In IEEE ICCV, 2011. 2

[2] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan. A theory of learning from different domains. Machine learning, 79(1-2):151-175, 2010. 6

[3] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. IEEE transactions on pattern analysis and machine intelligence, 35(8):1798- 1828, 2013. 2

[4] L. Bruzzone and M. Marconcini. Domain adaptation problems: A dasvm classification technique and a circular validation strategy. IEEE, PAMI, 32(5):770-787, 2010. 2

[5] Y. Cao, M. Long, J. Wang, Q. Yang, and P. S. Yu. Deep visual-semantic hashing for cross-modal retrieval. In ACM-SIGKDD, 2016. 2

[6] Z. Cao, M. Long, and Q. Yang. Transitive hashing network for heterogeneous multimedia retrieval. In AAAI, 2016. 2

[7] M. A. Carreira-Perpinán and R. Raziperchikolaei. Hashing with binary autoencoders. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 557-566, 2015. 2, 7

[8] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In BMVC, 2014. 2, 4

[9] R. Chattopadhyay, Q. Sun, W. Fan, I. Davidson, S. Pan-chanathan, and J. Ye. Multisource domain adaptation and its application to early detection of fatigue. ACM Transactions on Knowledge Discovery from Data (TKDD), 6(4):18, 2012. 2

[10] W.-S. Chu, F. De la Torre, and J. F. Cohn. Selective transfer machine for personalized facial action unit detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3515-3522, 2013. 2

[11] T.-T. Do, A.-D. Doan, and N.-M. Cheung. Learning to hash with binary deep neural network. In European Conference on Computer Vision, pages 219-234. Springer, 2016. 2, 7

[12] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In ICML, pages 647-655, 2014. 2, 6

[13] L. Duan, I. W. Tsang, and D. Xu. Domain transfer multiple kernel learning. IEEE PAMI, 34(3):465-479, 2012. 2

[14] V. Erin Liong, J. Lu, G. Wang, P. Moulin, and J. Zhou. Deep hashing for compact binary codes learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 2475-2483, 2015. 2

[15] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.- J. Lin. Liblinear: A library for large linear classification. Journal of machine learning research, 9(Aug):1871-1874, 2008. 5,7

[16] B. Fernando, A. Habrard, M. Sebban, and T. Tuytelaars. Unsupervised visual domain adaptation using subspace alignment. In CVPR, pages 2960-2967, 2013. 1, 2

[17] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky. Domain-

adversarial training of neural networks. Journal of Machine

神经网络的对抗训练。机器学习期刊

Learning Research, 17(59):1-35, 2016. 1, 2, 3, 6

[18] X. Glorot, A. Bordes, and Y. Bengio. Domain adaptation for large-scale sentiment classification: A deep learning approach. In Proceedings of the 28th International Conference on Machine Learning (ICML-11), pages 513-520, 2011. 1, 2

[19] B. Gong, K. Grauman, and F. Sha. Connecting the dots with landmarks: Discriminatively learning domain-invariant features for unsupervised domain adaptation. In ICML (1), pages 222-230, 2013. 2

[20] B. Gong, Y. Shi, F. Sha, and K. Grauman. Geodesic flow kernel for unsupervised domain adaptation. In IEEE CVPR, 2012. 1, 2, 5, 6

[21] Y. Gong and S. Lazebnik. Iterative quantization: A procrustean approach to learning binary codes. In Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, pages 817-824. IEEE, 2011. 7

[22] Y. Gong, S. Lazebnik, A. Gordo, and F. Perronnin. Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval. IEEE Transactions on Pattern Analysis and Machine Intelligence, 35(12):2916- 2929, 2013. 2, 4, 7

[23] R. Gopalan, R. Li, and R. Chellappa. Domain adaptation for object recognition: An unsupervised approach. In 2011 international conference on computer vision, pages 999-1006. IEEE, 2011. 2

[24] A. Gretton, D. Sejdinovic, H. Strathmann, S. Balakrishnan, M. Pontil, K. Fukumizu, and B. K. Sriperumbudur. Optimal kernel choice for large-scale two-sample tests. In Advances in neural information processing systems, pages 1205-1213, 2012. 3, 4, 6

[25] K. He, F. Wen, and J. Sun. K-means hashing: An affinity-preserving quantization method for learning binary compact codes. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2938-2945, 2013. 2, 7

[26] S. Hochreiter, Y. Bengio, P. Frasconi, and J. Schmidhuber. Gradient flow in recurrent nets: the difficulty of learning long-term dependencies, 2001. 4

[27] J. Hoffman, E. Rodner, J. Donahue, K. Saenko, and T. Darrell. Efficient learning of domain-invariant image representations. In ICLR, 2013. 2

[28] K. Jarrett, K. Kavukcuoglu, Y. Lecun, et al. What is the best multi-stage architecture for object recognition? In 2009 IEEE 12th International Conference on Computer Vision, pages 2146-2153. IEEE, 2009. 5

[29] Q.-Y. Jiang and W.-J. Li. Deep cross-modal hashing. arXiv preprint arXiv:1602.02255, 2016. 2

[30] W.-J. Li, S. Wang, and W.-C. Kang. Feature learning based deep supervised hashing with pairwise labels. In IJCAI, 2016, 2016. 3, 4, 7

[31] M. Long, Y. Cao, J. Wang, and M. Jordan. Learning transferable features with deep adaptation networks. In ICML, pages 97-105, 2015. 1, 2, 3, 4, 6

[32] M. Long, J. Wang, G. Ding, J. Sun, and P. Yu. Transfer joint matching for unsupervised domain adaptation. In CVPR, pages 1410-1417, 2014. 2

[33] M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu. Transfer feature learning with joint distribution adaptation. In Proceedings of the IEEE International Conference on Computer Vision, pages 2200-2207, 2013. 1, 2, 5, 6

[34] M. Long, H. Zhu, J. Wang, and M. I. Jordan. Unsupervised domain adaptation with residual transfer networks. In NIPS, 2016.1,2,3,6

[35] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In CVPR, pages 94-101. IEEE, 2010. 5

[36] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng . Reading digits in natural images with unsupervised feature learning. In NIPS Workshop on Deep Learning and Unsupervised

Feature Learning 2011, 2011. 5

[37] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1717-1724, 2014. 2

[38] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang. Domain adaptation via transfer component analysis. Neural Networks, IEEE Trans. on, 22(2):199-210, 2011. 1, 2, 6

[39] S. J. Pan and Q. Yang. A survey on transfer learning. IEEE TKDE, 22(10):1345-1359, 2010. 2

[40] M. Pantic, M. Valstar, R. Rademaker, and L. Maat. Web-based database for facial expression analysis. In ICME. IEEE, 2005. 5

[41] V. M. Patel, R. Gopalan, R. Li, and R. Chellappa. Visual domain adaptation: A survey of recent advances. IEEE signal processing magazine, 32(3):53-69, 2015. 2

[42] K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In ECCV, 2010. 1, 2, 5

[43] S. Shekhar, V. M. Patel, H. V. Nguyen, and R. Chellappa. Generalized domain-adaptive dictionaries. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 361-368, 2013. 1

[44] B. Sun, J. Feng, and K. Saenko. Return of frustratingly easy domain adaptation. In ICCV, TASK-CV, 2015. 1, 2, 6

[45] A. Torralba and A. A. Efros. Unbiased look at dataset bias. In Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, pages 1521-1528. IEEE, 2011. 6

[46] E. Tzeng, J. Hoffman, T. Darrell, and K. Saenko. Simultaneous deep transfer across domains and tasks. In Proceedings of the IEEE International Conference on Computer Vision, pages 4068-4076, 2015. 1, 2, 3

[47] A. Vedaldi and K. Lenc. Matconvnet - convolutional neural networks for matlab. In Proceeding of the ACM Int. Conf. on Multimedia, 2015. 6

[48] J. Wang, H. T. Shen, J. Song, and J. Ji. Hashing for similarity search: A survey. arXiv preprint arXiv:1408.2927, 2014. 1

[49] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? In Advances in neural information processing systems, pages 3320-3328, 2014. 2, 3

[50] H. Zhu, M. Long, J. Wang, and Y. Cao. Deep hashing network for efficient similarity retrieval. In Thirtieth AAAI Conference on Artificial Intelligence, 2016. 3, 4