Homeomorphism Alignment for Unsupervised Domain Adaptation

Lihua Zhou[1], Mao Ye[1,*], Xiatian Zhu[3], Siying Xiao[1], Xu-Qian Fan[2], Ferrante Neri[3]
[1]University of Electronic Science and Technology of China
[2]Jinan University
[3]University of Surrey
lihua.zhou@std.uestc.edu.cn, cvlab.uestc@gmail.com, xiatian.zhu@surrey.ac.uk
2018270101016@std.uestc.edu.cn, txqfan@jnu.edu.cn, f.neri@surrey.ac.uk

Abstract

Existing unsupervised domain adaptation (UDA) methods rely on aligning the features from the source and target domains explicitly or implicitly in a common space (i.e., the domain invariant space). Explicit distribution matching ignores the discriminability of learned features, while the implicit counterpart such as self-supervised learning suffers from pseudo-label noises. With distribution alignment, it is challenging to acquire a common space which maintains fully the discriminative structure of both domains. In this work, we propose a novel HomeomorphisM Alignment *(HMA) approach characterized by aligning the source and target data in two separate spaces. Specifically, an invertible neural network based homeomorphism is constructed. Distribution matching is then used as a sewing up tool for connecting this homeomorphism mapping between the source and target feature spaces. Theoretically, we show that this mapping can preserve the data topological structure (e.g., the cluster/group structure). This property allows for more discriminative model adaptation by leveraging both the original and transformed features of source data in a supervised manner, and those of target domain in an unsupervised manner (e.g., prediction consistency). Extensive experiments demonstrate that our method can achieve the state-of-the-art results. Code is released at* https://github.com/buerzlh/HMA.

1. Introduction

Deep learning methods rely mostly on a large quantity of manually labeled data [13, 37, 21, 55] which however could be prohibitively expensive or impossible to collect in many scenarios. One effective strategy is to exploit pre-existing labeled data (*i.e.*, the source domain) for a target domain without need for manual labeling. Due to the domain shift challenge [35], a model pretrained on a source domain often suffers from drastic performance degradation when directly applied on a target domain. This gives rise to the research attention of Unsupervised Domain Adaptation (UDA).

Existing UDA methods can be roughly divided into two categories. One is based on distribution alignment [31, 18, 10, 32] which minimizes domain discrepancy by aligning the distributions between two domains. They usually align two different distributions to a single distribution. However, this strategy could distort the original structural information, potentially hurting the final model generalization [5, 11, 44]. The other category is based on self-supervised learning [9, 26, 43] which also learns *a single common feature space* using pseudo labels or other self-supervision information. Given inevitable noise with self-supervision, it is difficult to obtain a common feature space with discriminative structure well kept. As shown in Fig. 1(a), adapting a model in a common space cannot guarantee better classification performance.

A natural solution for the above problem is to learn a latent distribution transformation function without destroying the original distribution of both domains. There are a few works in this line. For example, CyCADA [14] uses ordinary bijection implemented by two different networks to transform the images of the source domain to the target domain and vice versa. However, its learned two networks are not strictly inverse mappings, making the transformed images not necessarily semantically consistent through the transformation. That is, an image cannot be reconstructed after going through the two learned mappings. In addition, CyCADA learns two networks which is expensive.

To address the above limitations, we leverage a stricter bijection, namely Homeomorphism mapping, a concept borrowed from the topology field [34]. If a bijection satisfies the definition of homeomorphism (*i.e.*, one-to-one correspondence and continuous), theoretically we prove that the data topological structure can be well preserved in the projected space (*i.e.*, the samples in the same cluster are still

*corresponding author.

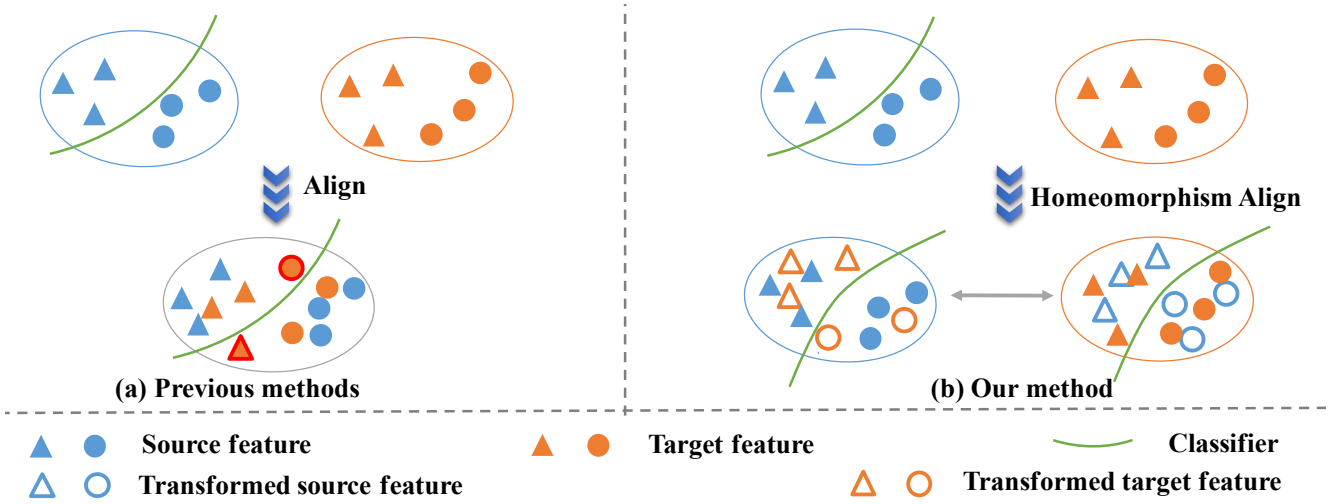

Figure 1. **Illustration of previous UDA methods and our homeomorphism alignment**. (a) Previous methods align the distributions between two domains in ***a single common feature space***, leading to that per-domain data discriminative structure is not well preserved. To overcome this problem, (b) our method leverages a homeomorphism mapping as a bridge for alignment ***across the source and target feature spaces***, since the homeomorphism mapping can preserve the original data topological structure.

in the same projected cluster). By this property, we further introduce a noise-free self-supervised learning task. With such a reversible mapping, semantic consistency is guaranteed. As shown in Fig.1(b), with our homeomorphism mapping, the adapted model works better along with both source and target feature spaces.

Motivated by the above analysis, we propose a novel unsupervised domain adaptation method, called ***HomeomorphisM Alignment*** (HMA). Our method consists of three components. The ***first*** is to construct a homeomorphism mapping for connecting the source and target feature spaces. We show that an Invertible Neural Network (INN) [20] can be used to derive a pair of mutually invertible functions (*i.e.*, homeomorphism mapping) through its forward and invertible processes. The ***second*** is sewing up across the two feature spaces with using distribution matching, as it is required that the cross-space transformed features are aligned with the original features. For discriminative learning, class semantic information is also considered during homeomorphic mapping. The ***third*** is to train the model in a self-supervised manner with the labels of source domain and the predictive consistency of unlabeled target domain as supervision. This is noise-free due to the preserved topological structure.

Our ***contributions*** are summarized as follows: (1) We theoretically prove that homeomorphism mapping can guarantee the topological structure of the mapped data. This is an important property yet ignored in the existing researches. We show that an INN can implement a homeomorphism. (2) We propose a novel UDA method with homeomorphism, the first attempt to consider the UDA problem from the viewpoint of topology by conducting domain alignment across two feature spaces. This design differs from the previous alternative methods typically learning a single common feature space for domain alignment. (3) Extensive experiments demonstrate the superiority of our HMA over prior art alternatives, along with in-depth ablation studies.

2. Related work

Unsupervised Domain Adaptation. UDA aims to improve the generalization ability of a model on an unlabeled target domain by leveraging the labeled source domain. Existing methods can be roughly divided into two categories. The first category adopts the idea of *distribution alignment* that minimizes the source error and the discrepancy between source and target domains concurrently. There exist five main strategies: *statistic moment matching*, *adversarial learning*, *optimal transport*, *bi-classifier adversarial learning* and *adversarial generation*.

Statistic moment matching based methods minimize the statistic discrepancy to align the distributions between two domains (*e.g.*, DAN [31], CORAL [42] and CAN [18]). *Adversarial learning* based methods are inspired by GAN [12], which plays a minimax game between feature extractor and discriminator to learn domain invariant features (*e.g.*, DANN [10], CDAN [32] and ADDA [46]). *Optimal transport* based methods generally consist of two steps: Finding a coupling matrix for connecting each source sample and target sample, followed by minimizing the cost of these pair-wise connections (*e.g.*, DeepJDOT [7], RWOT [50] and ETD [23]). *Bi-classifier adversarial learning* based methods play a minimax game with a single feature extractor and two distinct classifiers during adaptation.

Commonly, they maximize the prediction discrepancy when training the classifiers and minimize the prediction discrepancy when training the feature extractor (*e.g.*, MCD [39], SWD [22], BCDM [24] and CDAL [56]). *Adversarial generation* methods combine the domain discriminator and a generator, and generate fake data to align the distributions across domains at the pixel level (*e.g.*, CoGAN [29], SimGAN [41] and CycleGAN [57]). Overall, the first four strategies do feature-level distribution alignment while the last one considers pixel-level alignment. And in this work, we focus on feature-level distribution alignment.

The second category of UDA methods consider domain adaptation as a *self-supervised learning* problem. The key of this strategy is to obtain accurate pseudo-labels or self-supervision information. Similarly, such methods also aim to find a common space to implicitly align the source and target features so that the source and target domain features projected by the feature extractor have better discriminability. For example, SE [9] uses the mean teacher framework with a student and a teacher. For the update of the student, it uses the cross-entropy of source samples and the consistency constraints of target samples. While the teacher is updated by exponential moving average of the student. ATDOC [26] assigns a pseudo-label for each target sample by employing a memory mechanism or neighborhood aggregation. ssUDA [43] performs self-supervised tasks (*e.g.*, rotation, flip and patch location predictions) to improve the model generalization. SHOT [27] adopts the information maximization to solve the source-data free UDA problem. BNM [6] proposes Batch Nuclear-norm Maximization on the output matrix to improve both discriminability and diversity. CDTRANS [51] designs a two-way center-aware labeling algorithm to produce pseudo labels for target samples.

Invertible Neural Network (INN) is a flow-based model that transforms a probability distribution to another by a sequence of invertible and differentiable mappings. This model has been applied in a variety of problems. For example, HCFlow [28] utilizes the hierarchical conditional flow as a unified framework for image super-resolution and image rescaling. NCSR [19] proposes noise conditional flow model for super-resolution for increasing the visual quality and diversity of images. Derived from general volume preserving flows, iVPF [54] solve lossless compression by an exact bijective mapping without any numerical error. iFlow [53] also solves lossless compression by a modular scale transform with numerically invertible flow transformations. DIST [2] a diverse image style transfer framework by enforcing invertible cross-space mapping. Additionally, invertible networks play an important role in protecting privacy, such as invertible de-identification and image hiding [1, 17]. To the best of our knowledge, this is first attempt that leverages INN for unsupervised domain adaptation.

3. Analysis of Distribution Alignment

Problem statements. In UDA, we have a source domain $D_s = \{(\boldsymbol{x}_i^s, \boldsymbol{y}_i^s)\}_{i=1}^{n_s}$ with n_s labeled samples and a target domain $D_t = \{(\boldsymbol{x}_i^t)\}_{i=1}^{n_t}$ with n_t unlabeled samples. The two domains share the same label space $\{1, 2, \cdots, K\}$, but in different data distributions. The source model Γ_s, pretrained on labeled source data, is composed of a feature extractor F and a classifier C. The goal of UDA is to adapt the source model for the unlabeled target domain.

Ideally, given ground-truth labels, the trained model can work well in both the source and target domains. To verify this, we conduct experiments on the Office-31 dataset [38]. Using ground-truth labels of both source and target domains, supervised training is applied. It is found that 100% accuracy on both domains can be achieved. Next, we will investigate the adapted models trained by varying distribution alignment methods. See Appendix for more algorithm details.

Can previous distribution alignment methods really achieve 100% accuracy given ground-truth target labels? As mentioned earlier, existing UDA methods usually adapt the source domain model to the target domain through two strategies: distribution alignment and self-supervised learning. Commonly, both tackle domain adaptation by projecting source and target samples into a common space (*i.e.*, a domain invariant space). We conduct an experiment on the top-4 most challenging tasks from Office-31. We use the ground-truth target labels for domain alignment. The results are shown in Table 1. The first row shows the results of CAN [18] which aligns feature distributions based on the ground-truth target labels. The second row shows the results with an adversarial learning method CDAN [32]. The third row gives the performance of an optimal transport method DeepJDOT [7]. Note, the *bi-classifier adversarial learning* methods are excluded as they are designed to use the predictions of the classifiers but not pseudo-labels, and thus ground-truth labels can not be used. It is observed that these explicit distribution alignment methods cannot achieve 100% accuracy, meaning that the discriminative data structure is not well preserved during alignment. The fourth row gives the results of a self-supervised learning approach using the ground-truth target sample labels. It works well which means that the feature extractor can find the domain invariant space while keeping data discriminative structure. However, in practice the label noise cannot be eliminated fully.

Can bijection achieve 100% accuracy? It is difficult to find a feature extractor to obtain a common space while keeping discriminative data structure. We then investigate whether source and target samples can be aligned in two feature spaces in a way of bijection for superior domain

Table 1. Up-bound performance probing: Comparing different distribution alignment strategies on *Office-31* using the ground-truth target sample labels. SMM: Statistic moment matching; AL: Adversarial learning; OP: Optimal transport; SL: Self-supervised learning; BA:Bijection alignment.

Component	A→D	A→W	D→A	W→A
SMM	99.9±0.1	99.9±0.0	92.6±0.2	93.8±0.1
AL	99.2±0.1	99.8±0.1	90.9±0.2	92.3±0.1
OP	96.2±0.1	98.8±0.1	89.9±0.2	90.9±0.1
SL	**100.0±0.0**	**100.0±0.0**	**100.0±0.0**	**100.0±0.0**
BA	97.8±0.2	98.9±0.1	90.7±0.1	93.2±0.2
Ours	**100.0±0.0**	**100.0±0.0**	**100.0±0.0**	**100.0±0.0**

adaptation. The fifth row in Table 1 shows the performance of using two different networks to map the source features to the target feature space and vice versa. In this test, we use ground-truth target labels to train these two networks and align the conditional distributions between the transformed features and original features. This bijection made by the two networks is *not* a homeomorphism. Thus, the data topological structure can not be preserved in the process of bidirectional projections. It is shown that despite access to all ground-truth labels of target domain, this method still cannot achieve 100% accuracy.

Homeomorphism alignment can achieve 100% accuracy. A homeomorphism (also known as a continuous transformation) is a one-to-one correspondence mapping between the points in two topological spaces. It is continuous in both directions (refer to [34] for more details). Let M and N be two topological spaces, and $g : M \to N$ be a bijection. If both the function g and its inverse function $g^{-1} : N \to M$ are continuous, then g is called a **homeomorphism**. That being said, a homeomorphism is a bijective correspondence $g : M \to N$ such that $g(U)$ is open if and only if U is open. By the definition above, g is a homeomorphism if and only if g^{-1} is a homeomorphism. Based on this definition, the following theorem can be derived.

Theorem 1. The set boundary corresponds to the set boundary by homeomorphism. More precisely, let (M, d_M) and (N, d_N) be two metric spaces where d_M and d_N are the metrics on M, N respectively. Suppose there is a homeomorphism $g : M \to N$, and A is an open subset in (M, d_M), we have that its image $B := g(A)$ is an open subset in (N, d_N), and

$$g(\partial A) = \partial B = \partial g(A).$$

where ∂ means the boundary (see more details in Sec. 6).

As shown in Theorem 1, data topological structure is preserved by homeomorphism mapping, *i.e.*, the samples in the same cluster are still in the same projected cluster. In the literature, we find that there exists a network satisfying homeomorphism definition – Invertible Neural Network (INN) [20]. We validate that an INN satisfies the following theorem.

Theorem 2. Invertible Neural Network is a homeomorphism (see more details in Sec. 6).

As shown in the last row of Table 1, we use an INN to connect two domains at the feature level. With the ground-truth target sample labels, the homeomorphism alignment can achieve 100% accuracy thanks to the topological structure preserving property.

4. Method of Homeomorphism Alignment

Overview: Based on the analysis of distribution alignment, as shown in Fig. 2, we propose a HomeomorphisM Alignment (HMA) method with three components. The first is homeomorphism mapping construction using an INN for connecting the source and target feature spaces. The second is sewing up across the two feature spaces with using distribution matching. The third is training the source model in the source and target feature spaces with a noise-free self-supervised task by using the property of homeomorphism.

4.1. Homeomorphism implemented by invertible neural network (INN)

In each iteration, we randomly sample a batch of source and target samples. We use ResNet [13] as the feature extractor F to map a sample x to the feature space: $\boldsymbol{f}^{s/t} = F(\boldsymbol{x}^{s/t})$ where s/t represents the source and target domain respectively. Due to the distribution discrepancy between the source and target domains, we consider that the source and target features reside on two different spaces (manifolds) respectively.

A homeomorphism g consists of m blocks of INN. m is a hyperparameter discussed in Appendix. In each block, we use an affine network to implement the INN [8], as shown in Fig. 2. In the *forward* process, we transform from the source feature space $\boldsymbol{f}^s$ to the target feature space $\boldsymbol{f}^t$. Specifically, for the i-th block, we denote the input $\mu^i_{1:2d}$ with $2d$ dimension. And we split evenly $\mu^i_{1:2d}$ to two parts $[\mu^i_{1:d}, \mu^i_{d+1:2d}]$, and further transform them with two respective linear neural networks $s(\cdot)$, $t(\cdot)$. The output of i-th block $\mu^{i+1}_{1:2d}$ is then obtained with residual as follows:

$$\begin{aligned} \mu^{i+1}_{1:d} &= \mu^i_{1:d} + s(\mu^i_{d+1:2d}), \\ \mu^{i+1}_{d+1:2d} &= \mu^i_{d+1:2d} + t(\mu^{i+1}_{1:d}). \end{aligned} \tag{1}$$

The output $\mu^{i+1}_{1:2d}$ (a concatenation of $\mu^{i+1}_{1:d}$ and $\mu^{i+1}_{d+1:2d}$) will be set as the input of the next block.

In the *inverse projection* process, we transform from the target feature space $\boldsymbol{f}^t$ to the source feature space $\boldsymbol{f}^s$. For the i-th block, we map $\mu^{i+1}_{1:2d}$ to $\mu^i_{1:2d}$ in the opposite away around. According to Eq.(1), we can get the following equation:

$$\begin{aligned} \mu^i_{d+1:2d} &= \mu^{i+1}_{d+1:2d} - t(\mu^{i+1}_{1:d}), \\ \mu^i_{1:d} &= \mu^{i+1}_{1:d} - s(\mu^i_{d+1:2d}). \end{aligned} \tag{2}$$

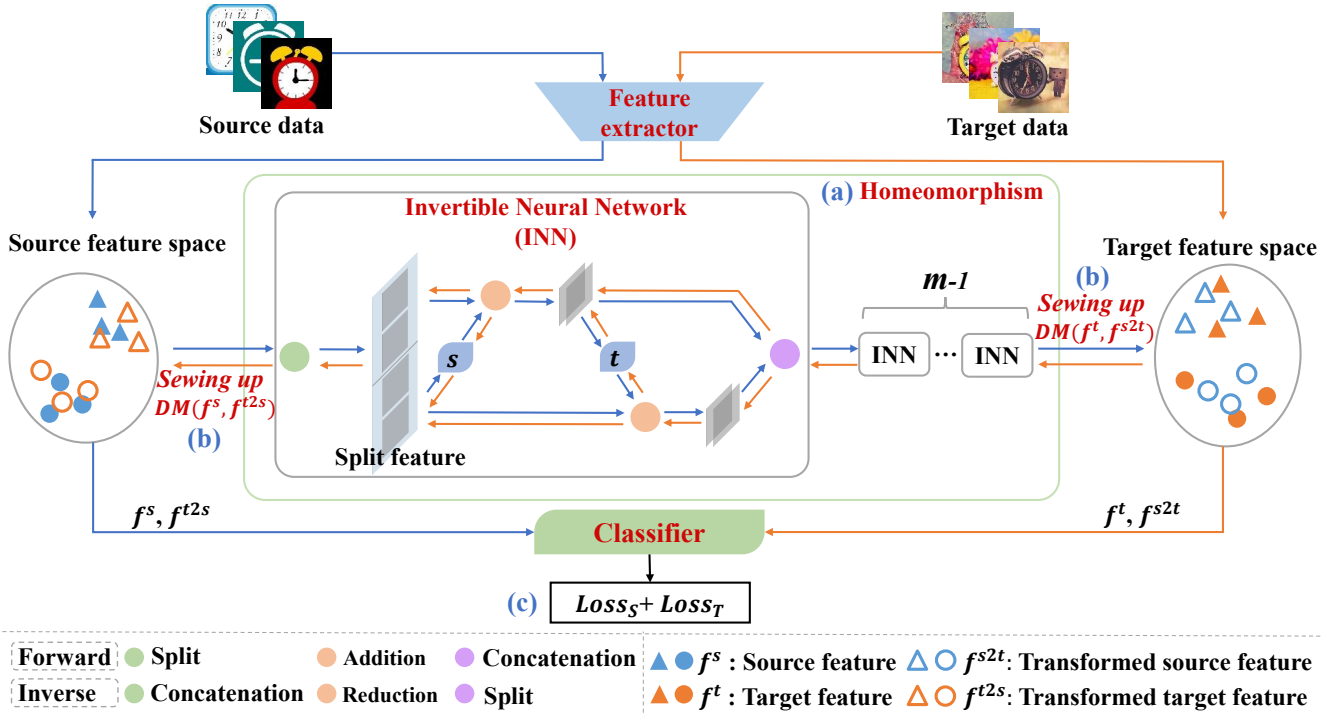

Figure 2. The framework of the proposed HomeomorphisM Alignment (HMA). (a) We cascade m invertible neural networks to implement a homeomorphism. (b) The transformed features are sewed up with the corresponding feature spaces by category. (c) The source model is iteratively trained in the two spaces concurrently.

We similarly split $\mu^{i+1}_{1:2d}$ into two parts $[\mu^{i+1}_{1:d}, \mu^{i+1}_{d+1:2d}]$ and follow Eq.(2) to get the original input $\mu^{i}_{1:2d}$ ((concatenation of $\mu^{i}_{1:d}$ and $\mu^{i}_{d+1:2d}$). Obviously Eqs. (1) and (2) are inverse functions of each other. We denote the forward process of the m INNs as the function g, while the inverse process as g^{-1}. Hence, the function g is a bijection. Since the functions $s(\cdot)$, $t(\cdot)$ are implemented by two linear connected neural networks, they are continuous; This means both g and g^{-1} are continuous, too. According to the definition of homeomorphism, this INN is a homeomorphism.

4.2. Sewing up

Next we will sew up homeomorphism mapping g to the source and target feature spaces such that the corresponding classes are aligned. Suppose the transformed feature $\boldsymbol{f}^{s2t} = g(\boldsymbol{f}^s)$ according to the source feature $\boldsymbol{f}^s$ and the transformed feature $\boldsymbol{f}^{t2s} = g^{-1}(\boldsymbol{f}^t)$ according to target feature $\boldsymbol{f}^t$. According to the homeomorphism mapping g, $\boldsymbol{f}^s = g^{-1}(\boldsymbol{f}^{s2t})$ and $\boldsymbol{f}^t = g(\boldsymbol{f}^{t2s})$. To guarantee that the transformed features are located discriminatively (*i.e.*, aligned with the corresponding classes), distribution matching is used for sewing up. The loss function is defined as follows,

$$\min_g \text{Loss}_{Sew} = DM(\boldsymbol{f}^{s2t}, \boldsymbol{f}^t) + DM(\boldsymbol{f}^{t2s}, \boldsymbol{f}^s), \quad (3)$$

where $DM(\cdot,\cdot)$ refer to any existing distribution matching method (*e.g.*, DAN [31], CAN [18], DANN [10], CDAN [32], DeepJDOT [7] and MCD [39]). Note, in case of using the adversarial learning method or bi-classifier adversarial learning method, an additional discriminator network or two auxiliary classifiers are needed.

Recall that only marginal distribution matching methods (*e.g.*, DAN and DANN) cannot achieve satisfactory results, because they focus on overall distribution alignment instead of class-wise alignment. Although our homeomorphism mapping g can ensure that the transformed features preserve the topological structure, without discriminative sewing up, it is still hard to achieve lossless transformation across domains. Thus class conditional distribution matching is a better choice as validated in ours experiment (Table 2-4).

4.3. Model Training

A model often suffers performance degradation from the domain shift. To address this problem, we leverage the homeomorphic property (preserved topological structure) to perform a noise-free self-supervised training. Specifically, as proved by Theorem 1, $\boldsymbol{f}^s$ has the same structural information as $\boldsymbol{f}^{s2t}$. Concretely, for a specific labeled source sample x, the corresponding feature $\boldsymbol{f}^s$ and $\boldsymbol{f}^{s2t}$ share the same label. The following loss function is applied to the

supervised training of feature extractor F and classifier C,

$$Loss_S = \mathcal{L}^{ce}(C(\boldsymbol{f}^s), \boldsymbol{y}^s) + \mathcal{L}^{ce}(C(\boldsymbol{f}^{s2t}), \boldsymbol{y}^s), \quad (4)$$

where $\boldsymbol{y}^s$ is the corresponding label of the source sample x^s, and $\mathcal{L}^{ce}(\cdot,\cdot)$ denotes the cross entropy function. In particular, the term $\mathcal{L}^{ce}(C(\boldsymbol{f}^s), \boldsymbol{y}^s)$ focuses on the classification of the source domain, whilst $\mathcal{L}^{ce}(C(\boldsymbol{f}^{s2t}), \boldsymbol{y}^s)$ is concerned with the classification of the target domain since $\boldsymbol{f}^{s2t}$ and $\boldsymbol{f}^t$ have been aligned.

Considering that our homeomorphism preserves the structure across the mapping and no label information in the target domain, unsupervised consistency constraint is a natural strategy for optimization. Formally, for an unlabeled target sample x^t, we formulate the consistency constraint on the predictions between $\boldsymbol{f}^t$ and $\boldsymbol{f}^{t2s}$ as:

$$Loss_T = L_C(C(\boldsymbol{f}^t), C(\boldsymbol{f}^{t2s})), \quad (5)$$

where $L_C(\cdot,\cdot)$ is a consistency constraint such as L_1-Norm and L_2-Norm. In practice, we found L_2-Norm suffices. By combining (4) and (5), the overall loss is defined as follows,

$$\min_{F,C} Loss_S + Loss_T. \quad (6)$$

Summary. At the training phase, in each iteration, we first train an INN based homeomorphism mapping, followed by model training in two spaces concurrently. At the inference phase, both the target features $\boldsymbol{f}^t$ and the transformed target features $\boldsymbol{f}^{t2s}$ can be used to make the prediction. Also, average based ensemble can be used to obtain the final prediction.

Remarks. Our model is trained in the source and target feature spaces concurrently. Compared with the existing alignment based UDA methods in a common space, this design naturally overcomes the intrinsic challenges of projecting the source and target domain samples into a single shared feature space using a feature extraction network while keeping their respective discriminative structures. When the distributions between the two feature spaces are not originally aligned typical in practice (*e.g.*, due to domain-specific characteristics such as different background, viewing conditions, etc.), the homeomorphism provides a flexible non-invasive means for cross-domain relating via transforming their individual features from each other *externally*. Critically, this alignment in two spaces allows to fully keep the original per-domain characteristics including some discriminative information. Compared with the self-supervised learning methods suffering the noises of pseudo-labeling, our transformed source features in the target domain can directly use the ground-truth source labels, in addition to additionally exploiting the topological structure of the source domain. Further, our consistency constraint can exploit the unlabeled target training data (*i.e.*, the original and transformed target features with shared topological structure) in an unsupervised manner, without the notorious pseudo-label noise issue.

5. Experiments

5.1. Experimental Setup

Datasets: In our experiments, three standard datasets are used. ***Office-31*** [38] is a popular benchmark. It contains a total of 4110 images of 31 office environment objects from 3 domains: Amazon (A), Webcam (W), Dslr(D). ***Office-Home*** [47] is a more challenging dataset which contains 15588 images within 65 classes from 4 domains: Artistic images (A), Clip-Art images (C), Product images (P) and RealWorld images (R). ***Visda-17*** [36] is a widely used benchmark for domain adaptation with focus on a 12-class synthesis-to-real object classification task. The source domain contains 152,397 synthetic images and the target domain has 55,388 real object images.

Implementation details: Our experiment is performed in Pytorch. Each task is run 5 times to enhance the robustness of the results. The same backbone network is selected as other compared methods for fair comparison. Specifically, Resnet-50 is selected as the backbone on *Office-31* and *Office-Home*, and Resnet-101 is selected on *Visda-17*. It is worth noting that the output dimension of the classifier in the original backbone is replaced by the number of categories to fit each task. The SGD optimizer is chosen to update the network and the CosineAnnealingLR [33] is used to update the learning rate of the SGD optimizer.

Competitors: For extensive evaluation, we compare our model with five groups of state-of-the-art methods. The first is based on distribution alignment, such as statistic moment matching methods DAN [31], CAN [18], and TSA [25]. The second uses adversarial learning including DANN [10], CDAN [32], MDD+IA [16], DADA [45], CLS [30], ILA [40], MetaAlign [48], DWL [49], and DALN [3]. The third exploits optimal transport: DeepJDOT [7]. The fourth is based on bi-classifier adversarial learning, MCD [39]. The fifth adopts self-supervised learning, including ALDA [4], ATDOC [26], CaCo [15], and SUDA [52].

5.2. Comparisons to State-of-the-Art

The performance comparison with the state-of-the-art methods on Office-31, Office-home and Visda-17 are shown in Table 2, Table 3 and Table 4 respectively. The methods HMA(DANN) and HMA(CDAN) mean the sewing up by distribution matching method based on adversarial learning, with DANN and CDAN focusing on marginal distribution alignment and conditional distribution alignment respectively. HMA(DeepHDOT) integrates the optimal transport method DeepJDOT, and HMA(MCD) the bi-classifier adversarial learning method MCD, as the sewing up tool.

Table 2. Comparison with the state-of-the-art methods on *Office-31* dataset. Metric: classification accuracy (%); Backbone: ResNet-50.

Method	Venue	A→D	A→W	D→A	D→W	W→A	W→D	*avg*
ResNet-50 [13]	CVPR16	68.9	68.4	62.5	96.7	60.7	99.3	76.1
DAN [31]	ICML15	78.6	80.5	63.6	97.1	62.8	99.6	80.4
CAN [18]	CVPR19	95.0	94.5	78.0	99.1	77.0	99.8	90.6
TSA [25]	CVPR21	92.6	94.8	74.9	99.1	74.4	**100.0**	89.3
DANN [10]	JMLR16	79.7	82.0	68.2	96.9	67.4	99.1	82.2
CDAN [32]	NIPS18	89.8	93.1	70.1	98.2	68.0	**100.0**	86.5
DADA [45]	AAAI20	93.9	92.3	74.4	99.2	74.2	**100.0**	89.0
MDD+IA [16]	ICML20	92.1	90.3	75.3	98.7	74.9	99.8	88.8
ILA [40]	CVPR21	93.4	**95.7**	72.1	**99.3**	75.4	**100.0**	89.3
MetaAlign [48]	CVPR21	94.5	93.0	75.0	98.6	73.6	**100.0**	89.2
DWL [49]	CVPR21	91.2	89.2	73.1	99.2	69.8	**100.0**	87.1
DALN [3]	CVPR22	95.4	95.2	76.4	99.1	76.5	**100.0**	90.4
DeepJDOT [7]	ECCV18	88.2	88.9	72.1	98.5	70.1	99.6	86.2
MCD [39]	CVPR18	92.2	88.6	69.5	98.5	69.7	100.0	86.5
ALDA [4]	AAAI20	94.0	95.6	72.2	97.7	72.5	**100.0**	88.7
ATDOC [26]	CVPR21	94.4	94.5	75.6	98.9	75.2	99.6	89.7
CaCo [15]	CVPR22	91.7	89.7	73.1	98.4	72.8	**100.0**	87.6
SUDA [52]	CVPR22	91.2	90.8	72.2	98.7	71.4	**100.0**	87.4
HMA(DANN)	Ours	83.9±0.1	83.5±0.2	70.5±0.1	98.2±0.1	70.1±0.2	**100.0±0.0**	84.4
HMA(CDAN)	Ours	92.4±0.2	95.1±0.2	73.7±0.1	99.2±0.1	72.8±0.2	**100.0±0.0**	88.9
HMA(DeepJDOT)	Ours	90.9±0.1	91.6±0.2	74.6±0.2	99.0±0.1	73.8±0.1	99.8±0.0	88.3
HMA(MCD)	Ours	93.5±0.1	91.3±0.1	73.1±0.1	99.2±0.1	73.5±0.1	**100.0±0.0**	88.4
HMA(DAN)	Ours	85.1±0.2	84.5±0.2	67.9±0.3	98.9±0.2	66.7±0.3	**100.0±0.0**	83.9
HMA(CAN)	Ours	**95.8±0.3**	95.1±0.1	**79.3±0.3**	**99.3±0.1**	**77.6±0.2**	**100.0±0.0**	**91.2**

Table 3. Comparisons with the state-of-the-art methods on *Office-Home* dataset. Metric: classification accuracy (%); Backbone: ResNet-50.

Method	Venue	A→C	A→P	A→R	C→A	C→P	C→R	P→A	P→C	P→R	R→A	R→C	R→P	*avg*
ResNet-50 [13]	CVPR16	34.9	50.0	58.0	37.4	41.9	46.2	38.5	31.2	60.4	53.9	41.2	59.9	46.1
DAN [31]	ICML15	43.6	57.0	67.9	45.8	56.5	60.4	44.0	43.6	67.7	63.1	51.5	74.3	56.3
CAN [18]	CVPR19	58.7	78.1	82.1	67.4	75.7	78.1	67.2	54.2	82.5	73.4	60.9	83.5	71.8
TSA [25]	CVPR21	53.6	75.1	78.3	64.4	73.7	72.5	62.3	49.4	77.5	72.2	58.8	82.1	68.3
DANN [10]	JMLR16	45.6	59.3	70.1	47.0	58.5	60.9	46.1	43.7	68.5	63.2	51.8	76.8	57.6
CDAN [32]	NIPS18	49.0	69.3	74.5	54.4	66.0	68.4	55.6	48.3	75.9	68.4	55.4	80.5	63.8
MDD+IA [16]	ICML20	56.2	77.9	79.2	64.4	73.1	74.4	64.2	54.2	79.9	71.2	58.1	83.1	69.5
MetaAlign [48]	CVPR21	59.3	76.0	80.2	65.7	74.7	75.1	65.7	**56.5**	81.6	74.1	61.1	85.2	71.3
DALN [3]	CVPR22	57.8	**79.9**	82.0	66.3	76.2	77.2	66.7	55.5	81.3	73.5	60.4	85.2	71.8
DeepJDOT [7]	ECCV18	50.7	68.6	74.4	59.9	65.8	68.1	55.2	46.3	73.8	66.0	54.9	78.3	63.5
MCD [39]	CVPR18	48.9	68.3	74.6	61.3	67.6	68.8	57.0	47.1	75.1	69.1	52.2	79.6	64.1
ALDA	AAAI20	53.7	70.1	76.4	60.2	72.6	71.5	56.8	51.9	77.1	70.2	56.3	82.1	66.6
ATDOC [26]	CVPR21	58.3	78.8	82.3	**69.4**	**78.2**	78.2	67.1	56.0	82.7	72.0	58.2	**85.5**	72.2
HMA(DANN)	Ours	48.2	65.1	75.4	57.0	65.0	68.3	55.6	45.2	73.5	66.6	54.3	78.4	62.7
HMA(CDAN)	Ours	58.7	78.1	81.6	67.4	75.8	78.1	66.8	54.2	82.5	73.4	59.7	83.5	71.7
HMA(DeepJDOT)	Ours	57.5	74.7	80.7	66.4	72.1	74.2	61.2	52.8	80.6	72.8	61.7	84.8	70.0
HMA(MCD)	Ours	55.5	74.5	81.3	67.6	74.1	75.5	63.6	53.9	82.2	75.3	58.7	82.4	70.4
HMA(DAN)	Ours	46.2	63.5	73.9	58.1	65.3	68.3	55.3	43.9	74.8	67.2	53.4	78.4	62.4
HMA(CAN)	Ours	**60.6**	79.1	**82.9**	68.9	77.5	**79.3**	**69.1**	55.9	**83.5**	**74.6**	**62.3**	84.4	**73.2**

While HMA(DAN) and HMA(CAN) apply the statistic moment matching methods as the sewing up tool, where DAN and CAN focus on marginal distribution alignment and conditional distribution alignment respectively.

It can be observed that HMA(CAN) yields the best average performance on both three datasets. This also confirms our previous analysis. Different sewing up methods will affect the final performance. In general, conditional distribution alignment is better than marginal distribution alignment on the two kinds of methods based on adversar-

Table 4. Comparison with the state-of-the-art methods on *Visda-17* dataset. Metric: per-class classification accuracy (%); Backbone: ResNet-101.

Method	Venue	plane	bcycl	bus	car	horse	knife	mcycl	person	plant	sktbrd	train	truck	*avg*
ResNet-101 [13]	CVPR16	55.1	53.3	61.9	59.1	80.6	17.9	79.7	31.2	81.0	26.5	73.5	8.5	52.4
DAN [31]	ICML15	84.8	42.1	75.4	53.0	77.9	62.6	86.6	50.7	59.7	52.9	82.5	26.0	62.9
CAN [18]	CVPR19	97.0	87.2	82.5	74.3	97.8	96.2	90.8	80.7	96.6	96.3	87.5	59.9	87.2
TSA [25]	CVPR21	-	-	-	-	-	-	-	-	-	-	-	-	78.6
DANN [10]	JMLR16	81.9	77.7	82.8	44.3	81.2	29.5	65.1	28.6	51.9	54.6	82.8	7.8	57.4
CDAN [32]	NIPS18	85.2	66.9	83.0	50.8	84.2	74.9	88.1	74.5	83.4	76.0	81.9	38.0	73.9
DWL [49]	CVPR21	90.7	80.2	**86.1**	67.6	92.4	81.5	86.8	78.0	90.6	57.1	85.6	28.7	77.1
CLS [30]	ICCV21	92.6	84.5	73.7	72.7	88.5	83.3	89.1	77.6	89.5	89.2	85.8	**72.7**	81.6
DALN [3]	CVPR22	-	-	-	-	-	-	-	-	-	-	-	-	80.6
DeepJDOT [7]	ECCV18	85.4	73.4	77.3	87.3	84.1	64.7	91.5	79.3	91.9	44.4	88.5	61.8	77.4
MCD [39]	CVPR18	87.0	60.9	83.7	64.0	88.9	79.6	84.7	76.9	88.6	40.3	83.0	25.8	71.9
ALDA [4]	AAAI20	93.8	74.1	82.4	69.4	90.6	87.2	89.0	67.6	93.4	76.1	87.7	22.2	77.8
ATDOC [26]	CVPR21	93.7	83.0	76.9	58.7	89.7	95.1	84.4	71.4	89.4	80.0	86.7	55.1	80.3
CaCo [15]	CVPR22	90.4	80.7	78.8	57.0	88.9	87.0	81.3	79.4	88.7	88.1	86.8	63.9	80.9
SUDA [52]	CVPR22	88.3	79.3	66.2	64.7	87.4	80.1	85.9	78.3	86.3	87.5	78.8	74.5	79.8
HMA(DANN)	Ours	86.9	79.1	83.5	50.5	86.7	47.3	86.1	55.1	64.6	59.8	84.6	36.2	68.4
HMA(CDAN)	Ours	88.3	71.2	85.1	66.4	86.3	79.3	88.8	**87.6**	83.9	79.3	83.4	46.2	78.8
HMA(DeepJDOT)	Ours	89.3	76.8	80.6	68.3	87.9	75.6	90.6	86.2	89.6	73.5	82.2	49.8	79.2
HMA(MCD)	Ours	89.0	72.2	85.5	70.3	90.4	86.6	87.6	83.8	92.2	55.5	89.1	42.1	78.7
HMA(DAN)	Ours	87.5	49.2	80.2	53.8	81.8	71.8	87.8	57.6	60.9	57.0	85.3	32.8	67.1
HMA(CAN)	Ours	**97.6**	**88.4**	84.3	**76.0**	**98.4**	**97.1**	**91.3**	81.4	**97.0**	**96.7**	**88.8**	60.7	**88.1**

Table 5. Homeomorphism mapping vs ordinary bijection on *Office-31*. OB(CAN) means ordinary bijection sewed by the distribution matching method CAN.

Component	A→D	A→W	D→A	W→A	Parameters
HMA(CAN)	**95.8±0.3**	**95.1±0.1**	**79.3±0.3**	**77.6±0.2**	20992000
OB(CAN)	90.3±0.3	91.7±0.2	76.6±0.1	75.8±0.2	33603584

Table 6. Ablation study on *Office-31*.

Component	A→D	A→W	D→A	W→A
CAN	95.0±0.3	94.5±0.3	78.0±0.3	77.0±0.3
INN(CAN)	94.8±0.3	94.1±0.2	77.3±0.4	76.7±0.2
INN(CAN)+S2T	95.6±0.2	94.9±0.3	78.9±0.2	77.4±0.2
HMA(CAN)	**95.8±0.3**	**95.1±0.1**	**79.3±0.3**	**77.6±0.2**
DAN	78.6±0.2	80.5±0.4	63.6±0.3	62.8±0.2
INN(DAN)	78.3±0.3	79.9±0.2	62.9±0.2	62.6±0.1
INN(DAN)+S2T	84.1±0.2	83.8±0.3	66.5±0.2	66.1±0.2
INN(DAN)+S2T+T2S	**85.1±0.2**	**84.5±0.2**	**67.9±0.3**	**66.7±0.3**

ial learning strategy and statistic moment matching strategy because conditional distribution alignment can stitch homeomorphism mapping with two spaces by category. Furthermore, for condition distribution matching method, statistic moment matching strategy is better than adversarial learning strategy. The reason is that statistic moment matching strategy stitches homeomorphism mapping with two feature spaces explicitly by category. Interestingly, the alignment method CAN published in 2019 still achieves SOTA results. It can be seen that alignment by category is very important for extracting domain invariant features. Our method HMA(CAN) further boosts the CAN performance, because we realize the difficulty of extracting a domain invariant space and we do alignment in the two spaces by homeomorphism mapping.

5.3. Ablation Analysis and Discussion

Homeomorphism map is better than ordinary bijection. As mentioned in Section 3, there does not exist many methods which do alignments in two feature spaces based on bijection. In this experiment, we will apply the ordinary bijection method to top-4 challenging tasks on Office-31 dataset, the difference between homeomorphism implemented by INN and ordinary bijection is mainly reflected in topological structure maintenance. The results are shown in Table 5. It is obvious that homeomorphism mapping is superior to the ordinary bijection in both accuracy and model parameters. For the model size, the INN based homeomorphism mapping consumes around half of the parameters compared to using two neural networks.

Ablation study. To show the effectiveness of alignment in two spaces, we conduct an experiment on top-4 challenging tasks of Office-31 dataset. The results are shown in Table 6. The method CAN is considered as the baseline. INN(CAN) means just sewing up homeomorphism mapping to the two feature spaces and the source model is retrained based on

the source labels. INN(CAN)+S2T means the transformed source features are used to learn the model in the target feature space compared with INN(CAN). HMA(CAN) uses all features in two spaces. From Table 6, we can find that simply using INN can achieve similar performance as the feature distribution alignment method CAN. As shown in the third row in Table 6, by transferring the source features to the target feature space, the performance is greatly improved. For this case, the data topological structure and label information from source domain can be correctly transformed to the target domain, which allows that the learned model works well in the target domain. The fourth row in Table 6 shows that the performance can be further improved if both transformed features are used. Because our homeomorphism mapping keeps the corresponding relationship by category, with the help of supervision information in the source feature spaces, the generalization performance of the model in the two domains is improved. To further verify our ideas, we test DAN as a sewing tool, giving the consistent conclusion.

6. Proof of Theorem

6.1. Proof of Theorem 1

Theorem 1. Let (M, d_M) and (N, d_N) be two metric spaces with a homeomorphism

$$g : M \to N,$$

and A is an open subset in (M, d_M), we have that its image $B := g(A)$ is an open subset in (N, d_N), and

$$g(\partial A) = \partial B = \partial g(A),$$

where ∂ means the boundary.

Proof. It is sufficient to show that

$$g(\partial A) \subset \partial B. \tag{7}$$

Assume this is true, then we can imply Eq. 7 to g^{-1}, and obtain

$$g^{-1}(\partial B) \subset \partial A.$$

Hence we have $\partial B \subset g(\partial A)$. Combing this with Eq. 7, we have $g(\partial A) = \partial B$.

Now we want to show Eq. 7, that is, for any $x \in \partial A$, we have $g(x) \in \partial B$. Since $x \in \partial A$, but $x \notin A$, then $g(x) \notin B$, and there is a sequence $\{x_i\} \subset A$ such that $\lim_{i\to\infty} x_i = x$. By the continuity of the function g, we have

$$g(x) = g(\lim_{i\to\infty} x_i) = \lim_{i\to\infty} g(x_i).$$

Noting that $g(x_i) \in B$, we get $g(x) \in \partial B$. This completes the proof. □

6.2. Proof of Theorem 2

Theorem 2. Invertible Neural Network is a homeomorphism.

Proof. The definition of homeomorphism is that, a function $g : M \to N$ between two topological spaces is a homeomorphism if it has the following properties: (1) g is a bijection; (2) g is continuous; (3) The inverse function g^{-1} is continuous.

For any invertible neural network, assuming that its forward process is g, then its invertible process can be represent as g^{-1}, so g is a bijection. Because the function of each part of the invertible neural network is continuous, as $s(\cdot)$ and $t(\cdot)$ in our method, so both g and g^{-1} are continuous. To sum up, the invertible network is a homeomorphism. □

7. Conclusion

In this paper, we have proposed a new unsupervised domain adaptation method, termed as *HomeomorphisM Alignment in two spaces* (HMA). By analyzing previous alignment based methods, we argue that it is difficult to find a common space or domain invariant space to adapt the pretrained source model. So the alignment is performed in two spaces. The extracted source and target features can be further transformed respectively by a homeomorphism mapping so that they can be aligned semantically. Our method consists of three steps, i.e., constructing an INN based homeomorphism mapping, sewing up by category and retraining iteratively training the model in two spaces. In this way, the source labels can be fully used even in the target feature space for improving the model generalization for the target domain. Extensive experimental results demonstrate the effectiveness of our method.

8. Acknowledgment

This work was supported in part by the National Key R&D Program of China (2018YFE0203900), National Natural Science Foundation of China (62276048).

References

[1] Jingyi Cao, Bo Liu, Yunqian Wen, Rong Xie, and Li Song. Personalized and invertible face de-identification by disentangled identity information manipulation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3334–3342, October 2021. 3

[2] Haibo Chen, Lei Zhao, Huiming Zhang, Zhizhong Wang, Zhiwen Zuo, Ailin Li, Wei Xing, and Dongming Lu. Diverse image style transfer via invertible cross-space mapping. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 14880–14889, October 2021. 3

[3] Lin Chen, Huaian Chen, Zhixiang Wei, Xin Jin, Xiao Tan, Yi Jin, and Enhong Chen. Reusing the task-specific clas-

sifier as a discriminator: Discriminator-free adversarial domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7181–7190, 2022. 6, 7, 8

[4] Minghao Chen, Shuai Zhao, Haifeng Liu, and Deng Cai. Adversarial-learned loss for domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 3521–3528, 2020. 6, 7, 8

[5] Xinyang Chen, Sinan Wang, Mingsheng Long, and Jianmin Wang. Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation. In *International conference on machine learning*, pages 1081–1090. PMLR, 2019. 1

[6] Shuhao Cui, Shuhui Wang, Junbao Zhuo, Liang Li, Qingming Huang, and Qi Tian. Towards discriminability and diversity: Batch nuclear-norm maximization under label insufficient situations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3941–3950, 2020. 3

[7] Bharath Bhushan Damodaran, Benjamin Kellenberger, Rémi Flamary, Devis Tuia, and Nicolas Courty. Deepjdot: Deep joint distribution optimal transport for unsupervised domain adaptation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 447–463, 2018. 2, 3, 5, 6, 7, 8

[8] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. In *International Conference on Learning Representations*, 2017. 4

[9] Geoff French, Michal Mackiewicz, and Mark Fisher. Self-ensembling for visual domain adaptation. In *International Conference on Learning Representations*, 2018. 1, 3

[10] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016. 1, 2, 5, 6, 7, 8

[11] Chunjiang Ge, Rui Huang, Mixue Xie, Zihang Lai, Shiji Song, Shuang Li, and Gao Huang. Domain adaptation via prompt learning. *arXiv preprint arXiv:2202.06687*, 2022. 1

[12] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, 2014. 2

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 1, 4, 7, 8

[14] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *International conference on machine learning*, pages 1989–1998. Pmlr, 2018. 1

[15] Jiaxing Huang, Dayan Guan, Aoran Xiao, Shijian Lu, and Ling Shao. Category contrast for unsupervised domain adaptation in visual tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1203–1214, 2022. 6, 7, 8

[16] Xiang Jiang, Qicheng Lao, Stan Matwin, and Mohammad Havaei. Implicit class-conditioned domain alignment for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 4816–4827. PMLR, 2020. 6, 7

[17] Junpeng Jing, Xin Deng, Mai Xu, Jianyi Wang, and Zhenyu Guan. Hinet: Deep image hiding by invertible network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4733–4742, October 2021. 3

[18] Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. Contrastive adaptation network for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4893–4902, 2019. 1, 2, 3, 5, 6, 7, 8

[19] Younggeun Kim and Donghee Son. Noise conditional flow model for learning the super-resolution space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 424–432, 2021. 3

[20] Diederik P Kingma and Prafulla Dhariwal. Glow: generative flow with invertible $1\times$ 1 convolutions. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 10236–10245, 2018. 2, 4

[21] Sajja Tulasi Krishna and Hemantha Kumar Kalluri. Deep learning and transfer learning approaches for image classification. *International Journal of Recent Technology and Engineering (IJRTE)*, 7(5S4):427–432, 2019. 1

[22] Chen-Yu Lee, Tanmay Batra, Mohammad Haris Baig, and Daniel Ulbricht. Sliced wasserstein discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10285–10295, 2019. 3

[23] Mengxue Li, Yi-Ming Zhai, You-Wei Luo, Peng-Fei Ge, and Chuan-Xian Ren. Enhanced transport distance for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13936–13944, 2020. 2

[24] Shuang Li, Fangrui Lv, Binhui Xie, Chi Harold Liu, Jian Liang, and Chen Qin. Bi-classifier determinacy maximization for unsupervised domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 8455–8464, 2021. 3

[25] Shuang Li, Mixue Xie, Kaixiong Gong, Chi Harold Liu, Yulin Wang, and Wei Li. Transferable semantic augmentation for domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11516–11525, 2021. 6, 7, 8

[26] Jian Liang, Dapeng Hu, and Jiashi Feng. Domain adaptation with auxiliary target domain-oriented classifier. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16632–16642, 2021. 1, 3, 6, 7, 8

[27] Jian Liang, Dapeng Hu, Yunbo Wang, Ran He, and Jiashi Feng. Source data-absent unsupervised domain adaptation through hypothesis transfer and labeling transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):8602–8617, 2021. 3

[28] Jingyun Liang, Andreas Lugmayr, Kai Zhang, Martin Danelljan, Luc Van Gool, and Radu Timofte. Hierarchi-

cal conditional flow: A unified framework for image super-resolution and image rescaling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4076–4085, 2021. 3

[29] Ming-Yu Liu and Oncel Tuzel. Coupled generative adversarial networks. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pages 469–477, 2016. 3

[30] Xiaofeng Liu, Zhenhua Guo, Site Li, Fangxu Xing, Jane You, C.-C. Jay Kuo, Georges El Fakhri, and Jonghye Woo. Adversarial unsupervised domain adaptation with conditional and label shift: Infer, align and iterate. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10367–10376, October 2021. 6, 8

[31] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In *International conference on machine learning*, pages 97–105. PMLR, 2015. 1, 2, 5, 6, 7, 8

[32] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. In *Advances in neural information processing systems*, 2018. 1, 2, 3, 5, 6, 7, 8

[33] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 6

[34] James R. Munkres. Topology (2nd edition). In *Prentice Hall*, 2000. 1, 4

[35] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, pages 1345–1359, 2009. 1

[36] Xingchao Peng, Ben Usman, Neela Kaushik, Judy Hoffman, Dequan Wang, and Kate Saenko. Visda: The visual domain adaptation challenge. *arXiv preprint arXiv:1710.visda06924*, 2017. 6

[37] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 1

[38] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. European conference on computer vision, pages 213–226. Springer, 2010. 3, 6

[39] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3723–3732, 2018. 3, 5, 6, 7, 8

[40] Astuti Sharma, Tarun Kalluri, and Manmohan Chandraker. Instance level affinity-based transfer for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5361–5371, 2021. 6, 7

[41] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Joshua Susskind, Wenda Wang, and Russell Webb. Learning from simulated and unsupervised images through adversarial training. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2107–2116, 2017. 3

[42] Baochen Sun, Jiashi Feng, and Kate Saenko. Correlation alignment for unsupervised domain adaptation. In *Domain Adaptation in Computer Vision Applications*, pages 153–171. Springer, 2017. 2

[43] Yu Sun, Eric Tzeng, Trevor Darrell, and Alexei A Efros. Unsupervised domain adaptation through self-supervision. *arXiv preprint arXiv:1909.11825*, 2019. 1, 3

[44] Hui Tang, Ke Chen, and Kui Jia. Unsupervised domain adaptation via structurally regularized deep clustering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8725–8735, 2020. 1

[45] Hui Tang and Kui Jia. Discriminative adversarial domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 5940–5947, 2020. 6, 7

[46] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7167–7176, 2017. 2

[47] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5018–5027, 2017. 6

[48] Guoqiang Wei, Cuiling Lan, Wenjun Zeng, and Zhibo Chen. Metaalign: Coordinating domain alignment and classification for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16643–16653, 2021. 6, 7

[49] Ni Xiao and Lei Zhang. Dynamic weighted learning for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15242–15251, 2021. 6, 7, 8

[50] Renjun Xu, Pelen Liu, Liyan Wang, Chao Chen, and Jindong Wang. Reliable weighted optimal transport for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4394–4403, 2020. 2

[51] Tongkun Xu, Weihua Chen, Pichao Wang, Fan Wang, Hao Li, and Rong Jin. Cdtrans: Cross-domain transformer for unsupervised domain adaptation. *arXiv preprint arXiv:2109.06165*, 2021. 3

[52] Jingyi Zhang, Jiaxing Huang, Zichen Tian, and Shijian Lu. Spectral unsupervised domain adaptation for visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9829–9840, 2022. 6, 7, 8

[53] Shifeng Zhang, Ning Kang, Tom Ryder, and Zhenguo Li. iflow: Numerically invertible flows for efficient lossless compression via a uniform coder. *Advances in Neural Information Processing Systems*, 34:5822–5833, 2021. 3

[54] Shifeng Zhang, Chen Zhang, Ning Kang, and Zhenguo Li. ivpf: Numerical invertible volume preserving flow for efficient lossless compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 620–629, 2021. 3

[55] Bo Zhao, Jiashi Feng, Xiao Wu, and Shuicheng Yan. A survey on deep learning-based fine-grained object classification and semantic segmentation. *International Journal of Automation and Computing*, 14(2):119–135, 2017. 1

[56] Lihua Zhou, Mao Ye, Xiatian Zhu, Shuaifeng Li, and Yiguang Liu. Class discriminative adversarial learning for unsupervised domain adaptation. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 4318–4326, 2022. 3

[57] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017. 3