Task-Discriminative Domain Alignment for Unsupervised Domain Adaptation

Behnam Gholami [1], Pritish Sahu[1], Minyoung Kim[2], and Vladimir Pavlovic[1,2]

[1]Dept. of Computer Science, Rutgers University, NJ, USA
[2]Samsung AI Center, Cambridge, UK
{bb510,ps851,vladimir}@cs.rutgers.edu, v.pavlovic@samsung.com, mikim21@gmail.com

Abstract

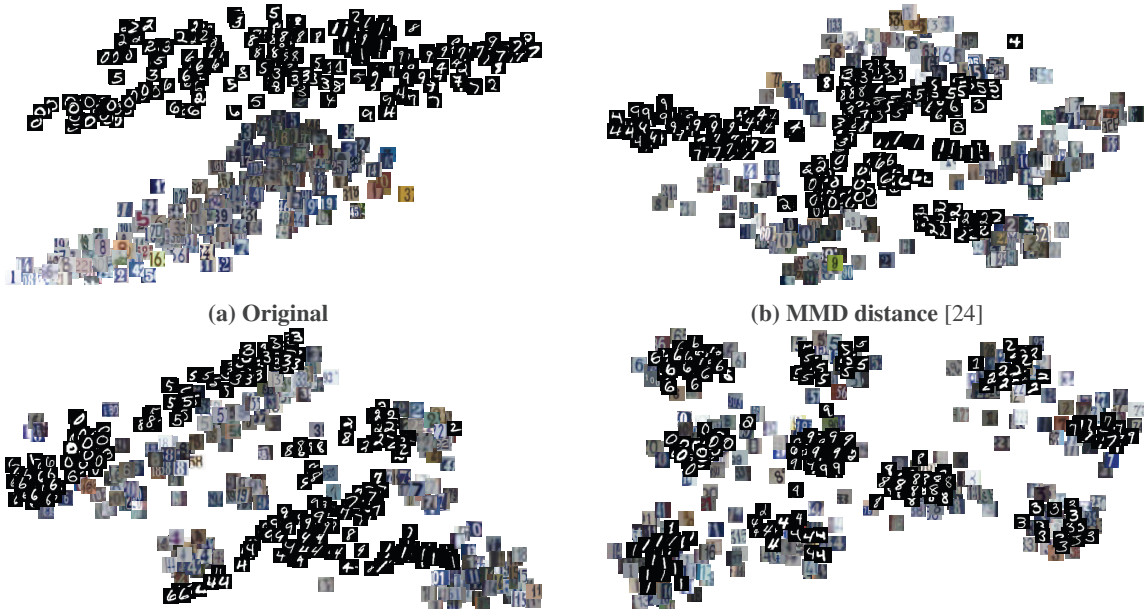
*Domain Adaptation (**DA**), the process of effectively adapting task models learned on one domain, the source, to other related but distinct domains, the targets, with no or minimal retraining, is typically accomplished using the process of source-to-target manifold alignment. However, this process often leads to unsatisfactory adaptation performance, in part because it ignores the task-specific structure of the data. In this paper, we improve the performance of **DA** by introducing a discriminative discrepancy measure which takes advantage of auxiliary information available in the source and the target domains to better align the source and target distributions. Specifically, we leverage the cohesive clustering structure within individual data manifolds, associated with different tasks, to improve the alignment. This structure is explicit in the source, where the task labels are available, but is implicit in the target, making the problem challenging. We address the challenge by devising a deep **DA** framework, which combines a new task-driven domain alignment discriminator with domain regularizers that encourage the shared features as task-specific and domain invariant, and prompt the task model to be data structure preserving, guiding its decision boundaries through the low density data regions. We validate our framework on standard benchmarks, including Digits (**MNIST**, **USPS**, **SVHN**, **MNIST-M**), **PACS**, and **VisDA**. Our results show that our proposal model consistently outperforms the state-of-the-art in unsupervised domain adaptation.*

1. Introduction

Domain adaptation refers to the problem of leveraging labeled task data in a source domain to learn an accurate model of the same tasks in a target domain where the labels are unavailable or very scarce [7]. The problem becomes challenging in the presence of strong data distribution shifts across the two domains [35, 11], which lead to high generalization error when using models trained on the source for predicting on target samples. Domain adaptation techniques seek to address the distribution shift problem. The key idea is to bridge the gap between the source and target in a joint feature space so that a task classifier trained on labeled source data can be effectively transferred to the target [29, 4, 2, 28]. In this regard, an important challenge is how to measure the discrepancy between the two domains. Many domain discrepancy measures have been proposed in previous **DA** studies, such as the moment matching-based methods [25, 4, 30, 42, 41], and adversarial methods [40, 3, 34, 43, 13]. Moment matching-based methods use Maximum Mean Discrepancy (**MMD**) [36] to align the distributions by matching all their statistics. Inspired by Generative Adversarial Networks (GAN) [14], adversarial divergences train a domain discriminator to discern the source from the target, while an encoder feature extractor is simultaneously learned to create features indistinguishable across the source and the target, confusing the discriminator.

Existing discrepancy approaches, reviewed in the next section, mainly focus on aligning domain-level feature distributions without considering category-level alignment. Thus, the alignment enforced by such discrepancy measures does not guarantee a good target performance as it ignores the cluster structure of the samples, aligned with their task labels. The assumption that the source features exhibit a well-defined cluster structure naturally transfers to the target: target features indicative of the same tasks as the source should manifest a similar cluster structure. In other words, when optimally aligned, the target features should amass around the source clusters such that the decision boundaries of the learned task classifiers do not induce partitioning of smooth clusters of target features. However, the aforementioned domain discrepancy measures only focus on global feature overlap, ignoring the finer task-aligned structure in the data. Consequently, they may inaccurately match the clusters and also cause the source features to form weakly separable clusters, as illustrated in Fig.1, b, c.

To alleviate the limitations of existing discrepancy measures for domain adaptation, we introduce a task (e.g., classification)-specific adversarial discrepancy measure that extends the discriminator output over the source classes, in order to additionally incorporate task knowledge into the adversarial domain alignment. The new discrepancy measure



(a) Original **(b) MMD distance** [24]

(c) Adversarial Divergence [13] **(d) Task-specific Divergence**

Figure 1: Feature visualization, via **t-SNE**, of Digit datasets when adapting **SVHN** (source) to **MNIST** (target). The target features in (b) and (c) are in close proximity of the source features but only weakly aligned to them. Source clusters are imperfectly delineated, demonstrating that the learned source features are insufficiently discriminative. Our new discrepancy measure in (d) leads to improved separation of source features and better alignment between the source and the target, which adheres to the structure of the data in both domains.

helps the feature extractor (encoder) make discriminative source/target features by considering the decision boundary information. Consequently, source-target alignment not only takes into account the domain-level feature densities but also the category-conditioned clusters-of-features information to produce an improved overlap, evident in Fig. 1, part d.

Motivated by the information-bottleneck principle [39], whose goal is to improve generalization by ignoring irrelevant (domain-variant) distractors present in the original data features, we also introduce a **source regularization** loss by minimizing the information between the source samples and their features by encouraging the marginal distribution of the source features to be similar to a prior distribution (the standard normal distribution) to enforce the model to focus only on the most discriminative (label-variant) features, less prone to overfitting. Moreover, an additional **target regularization** term is imposed on the classifier, trained on the shared features of the source samples, to encourage it not to pass through high-density regions of the target data. Previous **DA** methods did not explicitly consider these desiderata. Our ablation study in Sec. 4.4 empirically demonstrates the importance of the introduced objectives. We also empirically evaluate the advantages of our proposed method by demonstrating considerable improvements over the state-of-the-art methods on several standard domain adaptation benchmarks, including Digits, **PACS** and **VisDA** datasets.

2. Related Work

We summarize DA works most relevant to this paper. Several types of adversarial learning methods for unsupervised domain adaptation have been shown to match distributions of the features generated from source and target samples [9, 20, 37, 6, 10, 25].

The domain adversarial neural network (**DANN**) [13] first introduced a gradient reversal layer that reversed the gradients of the domain discriminator in order to encourage domain confusion. Other recent proposals [23, 3] have explored generative models such as GANs [14] to generate synthetic images for domain adaptation. These approaches typically train two GANs on the source and target input data with tied parameters with the goal of translating images between the domains. Despite being visually compelling, such image-space models have only been shown to work on small images and limited domain shifts.

In order to circumvent the need to generate images, **ADDA** [40] recently proposed an adversarial framework for directly minimizing the distance between the source and target encoded representations (shared features). A discriminator and (target) encoder are iteratively optimized in a two-player game, where the goal of the discriminator is to distinguish the target features from the source features, with the goal of the encoder being to confuse the discriminator.

The **DupGAN** [17] proposed a **GAN**-like model with duplex discriminators to restrict the latent representation to be domain invariant, with its category information preserved. Saito et al. [33] further introduce two classifiers as a discriminator to avoid ambiguous features near the class boundaries. By deploying two classifiers, the method therein employs the adversarial learning techniques to detect the disagreement across classifiers, such that the encoder is able to minimize this discrepancy on target samples.

In addition to the adversarial distribution matching oriented algorithms, pseudo-labels or conditional entropy regularization are also adopted in literature [32, 35, 43]. Sener et al. [35] construct a k-NN graph of target points based on a predefined similarity graph. Pseudo-labels are assigned to target samples via their nearest source neighbors, which allows end-to end joint training of the adaptation loss. Saito et al. [32] employ the asymmetric tri-training, which leverages target samples labeled by the source-trained classifier to learn target discriminative features. Zhang et al. [43] iteratively select pseudo-labeled target samples based on their proposed criterion and retrain the model with a training set including pseudo-labeled samples. However, these methods based on pseudo-labeled target samples have a critical bottleneck where false pseudo-labels can mislead learning of target discriminative features, leading to degraded performance.

3. Method

3.1. Problem Formulation

Without loss of generality, we consider a multi-class (K-class) classification problem as the running task example. *Consider the joint space of inputs and class labels, $\mathcal{X} \times \mathcal{Y}$ where $\mathcal{Y} = \{1, \dots, K\}$ for (K-way) classification. Suppose we have two domains on this joint space,* ***source (S)*** *and* ***target (T)****, defined by unknown distributions $P_S(\mathbf{x}, y)$ and $P_T(\mathbf{x}, y)$, respectively. We are given source-domain training examples with labels $\mathcal{D}_S = \{(\mathbf{x}_i^S, y_i^S)\}_{i=1}^{N_S}$ and target data $\mathcal{D}_T = \{\mathbf{x}_i^T\}_{i=1}^{N_T}$ with no labels. We assume the shared set of class labels between the two domains. The goal is to assign the correct class labels $\{y_i^T\}$ to target data points $\mathcal{D}_T$.*

To tackle the problem in the shared latent space framework, we introduce a shared encoder Q between the source and the target domains that maps a sample $\mathbf{x}$ into a stochastic embedding[1] $\mathbf{z} \sim Q(\mathbf{z}|\mathbf{x})$, and then apply a classifier h to map $\mathbf{z}$ into the label space $y \sim h(y|\mathbf{z})$ (h is trained to classify samples drawn from the encoder distribution). Although one can consider domain-wise different encoders, more recent **DA** approaches tend to adopt a shared encoder, which can prevent domain-specific nuisance features from being learned, reducing potential overfitting issues. We define the stochastic encoder E as a **conditional Gaussian** distribution with **diagonal** covariance that has the form $Q(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mathbf{z}|\mathbf{f}_\mu(\mathbf{x}), \mathbf{f}_{\mathbf{\Sigma}}(\mathbf{x}))$ where $\mathbf{f}_\cdot$ is a deep network mapping the data point $\mathbf{x}$ to the $2p$-dimensional latent code, with the first p outputs from f_e encoding $\mathbf{f}_\mu$, and the remaining p outputs encoding $\mathbf{f}_\Sigma$ (in this work, we set $p = 256$ for all the experiments). The classifier h outputs a K-dim probability vector of class memberships, modeled as a softmax form $\mathbf{h}(\mathbf{z}) = \mathrm{softmax}(\mathbf{f}_c(\mathbf{z}))$, where $\mathbf{f}_c(\mathbf{z})$ is a deep network mapping the latents $\mathbf{z}$ to the logits of K classes.

[1] Please see Remark 1 for the benefits of choosing a stochastic encoder over a deterministic one.

Remark 1. *The reason to choose a stochastic encoder over a deterministic one is two fold. First, it allows one to impose smoothness (local-Lipschtizness) constraint on the classifier h over target samples; see Sec. 3.1.5 for more details. Second, adding continuous noise to the inputs of the discriminators has been shown to improve instability and vanishing gradients in adversarial optimization problems through smoothening the distribution of features [1]. Our stochastic encoder equipped with the reparametarization approach inherently provides such mechanism to feature distribution smoothness; see Sec. 3.1.4 and 3.1.2 for more details.*

The proposed domain adaptation method can be summarized by the objective function consisting of six terms:

$$\mathcal{L}_{Class} + \mathcal{L}_{Disc} + \mathcal{L}_{Teach} + \mathcal{L}_{Smooth} + \mathcal{L}_{Entropic} + \mathcal{L}_{Adv}, \tag{1}$$

where $\mathcal{L}_{Class}$ is the classification loss applied to $\mathcal{D}_S$, $\mathcal{L}_{Disc}$ is the domain discrepancy loss measuring the discrepancy between the source and target distribution, $\mathcal{L}_{Teach}$ is the source-to-target teaching loss, which couples the source classifier with the target discriminator. The remaining losses, $\mathcal{L}_{Smooth}, \mathcal{L}_{Entropic}, \mathcal{L}_{Adv}$ will impose different regularization constraints on the model: $\mathcal{L}_{Smooth}$ will impose Lipschitz classifiers in the target space, $\mathcal{L}_{Entropic}$ will strive to drive the classifier towards regions of low density in the same target space, while $\mathcal{L}_{Adv}$ will impose regularization towards a reference density in the shared space $\mathcal{Z}$. We next discuss each of the above losses in more detail and then propose an algorithm to efficiently optimize the desired objective.

3.1.1 Source Classification Loss $\mathcal{L}_{Class}$

Having access to source labels, the stochastic mappings Q and h are trained on source samples to correctly predict the class label by minimizing the standard cross entropy loss,

$$\mathcal{L}_{Class}(Q, h) := -\mathbb{E}_{\mathbf{x},y\sim P_S(\mathbf{x},y)}\Big[\mathbb{E}_{\mathbf{z}\sim Q(\mathbf{z}|\mathbf{x})}\big[\mathbf{y}^\top \log h(\mathbf{z})\big]\Big], \tag{2}$$

where $\mathbf{y}$ is the K-D one-hot vector representing the label y.

3.1.2 Domain Discrepancy Loss $\mathcal{L}_{Disc}$

Since the stochastic encoder Q is shared between the source and target samples, to make sure the source and the target

features are well aligned in the shared space and respect the cluster structure of the original samples, we propose a novel domain alignment loss, which will be optimized in adversarial manner.

Rather than using the standard adversarial approach to minimizing the alignment loss between the source and the target densities in the shared space $\mathcal{Z}$, i.e., finding the encoder Q which "fools" the best binary discriminator D trying to discern source from target samples, our approach is inspired by semi-supervised GANs [8] where it has been found that incorporating task knowledge into the discriminator can jointly improve classification performance and quality of images produced by the generator. We incorporate task knowledge by replacing what would be a binary discriminator with a $(K+1)$-way multi-class discriminator $y' = D(\mathbf{z}) = \mathrm{softmax}(\mathbf{f}_d(\mathbf{z}))$. The first K classes indicate that a sample $\mathbf{z}$ belongs to the source domain *and* belongs to a specific classes in $\mathcal{Y}$, while the last $(K+1)$-th class "t" indicates $\mathbf{z}$ belongs to the target domain.

Since we have the class label for the source samples, the discriminator is trained to classify source features correctly, hence creating crisp source clusters in the feature space. On the other hand, the new discriminator seeks to distinguish the samples from the target domain from those of the source by assigning them to the $(K+1)$-th, "target" class.

$$\mathcal{L}_{Disc}(Q, D) := -\mathbb{E}_{\mathbf{x}\sim P_T(\mathbf{x})}\big[\mathbb{E}_{\mathbf{z}\sim Q(\mathbf{z}|\mathbf{x})}\big[[\mathbf{0}, 1]^\top \log D(\mathbf{z})\big]\big] - \mathbb{E}_{\mathbf{x},y\sim P_S(\mathbf{x},y)}\big[\mathbb{E}_{\mathbf{z}\sim Q(\mathbf{z}|\mathbf{x})}\big[[\mathbf{y}, 0]^\top \log D(\mathbf{z})\big]\big], \quad (3)$$

where $[\mathbf{0}, 1]$ is a one-hot vector indicating a point from the target domain and $[\mathbf{y}, 0]$ stands for a point from the source domain, labeled according to class label $\mathbf{y}$.

3.1.3 Teacher Target-Source Loss $\mathcal{L}_{Teach}$

Here, we seek the encoder Q to generate a feature representative of one of the first K task-specific classes for target samples preserving their cluster structure and aligning them to the source clusters in the feature space. However, the target data points are unlabeled, and the encoder will not have the chance to enforce the desired clustering structure of the target points, where points within a cluster would have the same predicted label. To "teach" the encoder, we ask the classifier $h(\cdot)$ to provide pseudo soft labels for the target points to our new discriminator using the following loss:

$$\mathcal{L}_{Teach}(Q, D, h) := -\mathbb{E}_{\mathbf{x}\sim P_T(\mathbf{x})}\Big[\mathbb{E}_{\mathbf{z}\sim Q(\mathbf{z}|\mathbf{x})}[[h(\mathbf{z}), 0]^\top \log D(\mathbf{z})]\Big]. \quad (4)$$

Intuitively, the encoder tries to fool the discriminator by assigning one of the first K classes to target features, leveraging on the output of the classifier h (augmented with 0 for the $K+1$-th dimension) as pseudo-labels for target features.

Remark 2. *The proposed task-specific domain discriminator can be used to improve any domain adaptation method that has an adversarial domain alignment component. Indeed, we observe (see Sec. 4.3) that the proposed discriminator significantly improves upon the standard binary discriminator.*

3.1.4 Source Domain Regularization Loss

One of the standard goals in representation learning is to find an encoding of the data point $\mathbf{x}$ that is maximally expressive about its label y while being maximally compressive about $\mathbf{x}$—finding a representation $\mathbf{z}$ which ignores as many details of $\mathbf{x}$ as possible. This is specifically useful for domain adaptation where we require a representation to be domain invariant. Essentially, we want $\mathbf{z}$ to act like a minimal sufficient statistic of $\mathbf{x}$ for predicting y in order to generalize better for samples form unseen domains. To do so, we introduce a regularizer that acts on the aggregated posterior of the shared features of the source samples $Q_z(\mathbf{z}) = \mathbb{E}_{\mathbf{x}\sim P_S(\mathbf{x})}[Q(\mathbf{z}|\mathbf{x})]$. The regularizer encourages $\mathbf{z}$ to be less informative about $\mathbf{x}$ in the form of mutual information by matching the aggregated posterior of the shared features with a factorized prior distribution $P_z(\mathbf{z})$[2], which in turn constrains the implicit capacity of $\mathbf{z}$ and encourages it be factorized:

$$\mathcal{D}\big[P_z(\mathbf{z})||Q_z(\mathbf{z})\big], \quad (5)$$

where $\mathcal{D}(\cdot||\cdot)$ is an arbitrary distribution divergence measure.

As the proxy for this divergence, we define an auxiliary loss which will be **adversarially** optimized. We introduce an a binary discriminator F in the latent space trying to separate ***true*** points sampled from P_z and ***fake*** ones sampled from Q_z. The encoder Q ensures the aggregated posterior distribution Q_z can fool the binary discriminator into thinking that the source features comes from the distribution P_z:

$$\mathcal{L}_{Adv}(Q, F) = -\mathbb{E}_{x\sim P_S(\mathbf{x})}\big[\mathbb{E}_{\mathbf{z}\sim Q(\mathbf{z}|\mathbf{x})}\big[\log F(\mathbf{z})\big]\big] - \mathbb{E}_{\mathbf{z}\sim P(\mathbf{z})}\big[\log(1 - F(\mathbf{z}))\big]. \quad (6)$$

Remark 3. *We empirically observed that imposing such regularization on target samples could be harmful to performance. We conjecture this is due to the lack of true class labels for the target samples, without which the encoder would not preserve the label information of the features, leading to unstructured target points in feature space.*

3.1.5 Target Domain Regularization Losses

In order to incorporate the target domain information into the model, we apply the cluster assumption, which states that the target data points $\mathcal{D}_T$ contains clusters and that points in the same cluster have homogeneous class labels. If the cluster assumption holds, the optimal decision boundaries

[2] In this work, we consider $P_z(\mathbf{z}) = \mathcal{N}(0, \mathbf{I})$

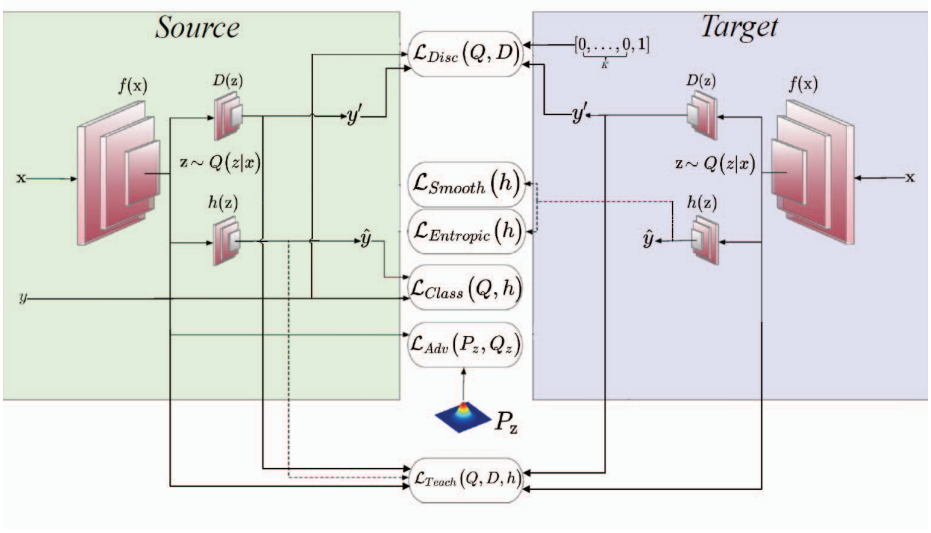


Figure 2: Proposed architecture includes a deep feature extractor $f(x)$ and a deep label predictor $h(z)$, which together form a standard feed-forward architecture. Unsupervised domain adaptation is achieved by adding a task-specific discriminator $D(z)$ connected to the feature extractor distinguishing the source from target features. The training proceeds standardly and minimizes the label prediction loss (for source examples) $\mathcal{L}_{Class}$, the domain discrepancy losses (for all samples) $\mathcal{L}_{Disc}$ and $\mathcal{L}_{Teach}$, the source domain regularization loss $\mathcal{L}_{Adv}$, and the target domain regularization losses $\mathcal{L}_{Smooth}$ and $\mathcal{L}_{Entropic}$.

should occur far away from data-dense regions in the feature space z. We achieve this by defining an entropic loss,

$$\mathcal{L}_{Entropic}(h, Q) := - \mathbb{E}_{\mathbf{x}\sim P_T(\mathbf{x})}\Big[\mathbb{E}_{\mathbf{z}\sim Q(\mathbf{z}|\mathbf{x})}\big[h(\mathbf{z})^\top \log h(\mathbf{z})\big]\Big]. \quad (7)$$

Intuitively, minimizing the conditional entropy forces the classifier to be confident on the unlabeled target data, thus driving the classifiers decision boundaries away from the target data. In practice, the conditional entropy must be empirically estimated using the available data.

However, Grandvale [15] suggested this approximation can be very poor if h is not locally-Lipschitz smooth. Without the smoothness constraint, the classifier could abruptly change its prediction in the neighborhood of training samples, allowing decision boundaries close to the training samples even when the empirical conditional entropy is minimized. To prevent this, we take advantage of our stochastic encoder and propose to explicitly incorporate the locally-Lipschitz constraint in the objective function,

$$\mathcal{L}_{Smooth}(h, Q) := \mathbb{E}_{\mathbf{x}\sim P_T(\mathbf{x})}\Big[\mathbb{E}_{\mathbf{z}_1,\mathbf{z}_2\sim Q(\mathbf{z}|\mathbf{x})}||h(\mathbf{z}_1) - h(\mathbf{z}_2)||_1\Big], \quad (8)$$

with $\|\cdot\|_1$ the L_1 norm. Intuitively, we enforce classifier consistency over proximal features of any target point $\mathbf{x}$.

Remark 4. *We empirically observed that having such constraints for source features would not improve performance. This is because access to the source labels and forcing the classifier to assign each source feature to its own class would already fulfill the smoothness and entropy constraints on the classifier for the source samples.*

3.2. Model Learning and Loss Optimization

Our goal, as outlined in Sec. 3.1, is to train the task-specific discriminator D, binary discriminator F, classifier h, and encoder Q to facilitate learning of the cross-domain classifier h. By approximating the expectations with the sample averages, using the stochastic gradient Descent (SGD), and the reparametarization approach [19], we solve the optimization task in the following four subtasks. The overall algorithm is available in the Supplementary Material (**SM**).

3.2.1 Optimizing the encoder Q

$$Q^* = \arg\min_Q \ \mathcal{L}_{Class}(Q, h^*) + \mathcal{L}_{Disc}(Q, D^*, h^*) + \lambda_Q\big[\mathcal{L}_{Adv}(Q, F^*)\big], \quad (9)$$

where λ_Q is a weighting factor. Intuitively, The first term in Eq. 9 encourages Q to produce discriminative features for the labeled source samples to be correctly classified by the classifier h. The second term simulates the adversarial training by encouraging Q to fool the task-specific discriminator D by pushing the target features toward the source features, leveraging the soft pseudo-labels provided by the classifier. Through the last term, the encoder seeks to fool the binary discriminator F into treating the source features as if they come from the fully-factorized $P(\mathbf{z})$ to produce domain-invariant source features.

3.2.2 Optimizing the classifier h

$$h^* = \arg\min_h \ \lambda_h[\mathcal{L}_{Class}(Q^*, h)] + \lambda'_h\big[\mathcal{L}_{Entropic}(Q^*, h) + \mathcal{L}_{Smooth}(Q^*, h)\big], \quad (10)$$

where λ_h and λ'_h are the trade-off factors. Intuitively, we enforce the classifier h to correctly predict the class labels of the source samples by the first term in Eq. 10. We use the second term to minimize the entropy of h for the target samples, reducing the effects of "confusing" labels of target samples. The last term guides the classifier to be locally consistent, shifting the decision boundaries away from target data-dense regions in the feature space.

3.2.3 Optimizing the task-specific discriminator D

$$D^* = \arg\min_D \ \mathcal{L}_{Disc}(Q^*, D). \tag{11}$$

The loss in Eq. 11 prompts D to shape its decision boundary to separate the source features (according to their class label) and target features from each other.

3.2.4 Optimizing the binary discriminator F

$$F^* = \arg\min_F \ \mathcal{L}_{Adv}(Q^*, F). \tag{12}$$

Intuitively, the loss in Eq. 12 encourages F to separate the source features from the features generated from the fully-factorized distribution $P_z(\mathbf{z})$ by assigning label 1 and 0 to the source feature samples and $P_z(\mathbf{z})$ samples, respectively.

3.3. Target Class Label Prediction

After model training, to determine the target class-label y_t of a given target domain instance $\mathbf{x}_t$, we first compute the distribution of y_t given $\mathbf{x}_t$ by integrating out the shared feature $\mathbf{z}_t$. Then, we select the most likely label as

$$\hat{y}_t = \arg\max_{y_t \in \{1,\dots,K\}} P(y_t|\mathbf{x}_t), \tag{13}$$

where $P(y_t|\mathbf{x}_t)$ can be computed as

$$P(y_t = k|\mathbf{z}_t) = \mathbb{E}_{\mathbf{z}\sim Q(\mathbf{z}|\mathbf{x}_t)=\mathcal{N}(f_e^\mu(\mathbf{x}_t), f_e^\Sigma(\mathbf{x}_t))}[h_k(\mathbf{z})], \tag{14}$$

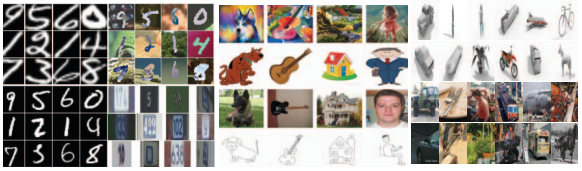
where $h_k(.)$ is the k-th entry of the classifier output. Since the above expression cannot be computed in a closed form, we approximate it with its mean value. Using this approximation, we compute y_t as:

$$\hat{y}_t = \arg\max_{k\in\{1,\dots,K\}} h_k(\mathbf{z}_t), \quad \mathbf{z}_t = f_e^\mu(\mathbf{x}_t). \tag{15}$$

Remark 5. *We empirically observed that estimating the expectation in Eq. 14 with Gibbs sampling from the posterior $Q(\mathbf{z}|\mathbf{x}_t)$ instead of its mean would not boost the performance. We conjecture this is due to the smoothness constraint we impose on the classifier through Eq. 8, enforcing consistency over proximal target samples drawn from $Q(\mathbf{z}|\mathbf{x})$.*

4. Experimental Results

We compare our proposed method with state-of-the-art on three benchmark tasks. The Digit datasets embody the digit cross-domain classification task across four datasets:



(a) Digits. **(b)** PACS **(c)** VisDA.

Figure 3: Example images from benchmark datasets.

MNIST, **MNIST-M**, **SVHN**, **USPS**, which consist of $K = 10$ digit classes (0-9). We also evaluated our method on **VisDA** object classification dataset [31] with more than 280K images across twelve categories. Finally, we report performance on **PACS** [21], a recently proposed benchmark which is especially interesting due to the significant domain shift between different domains. It contains images of seven categories extracted from four different domains:*Photo* (P), *Art paintings* (A), *Cartoon* (C), and *Sketch* (S). The details of the datasets are available in **SM**. Fig. 3 illustrates image samples from different datasets and domains. We evaluate the performance of all methods with the classification accuracy metric. We used ADAM [18] for training; the learning rate was set to 0.0002 and momentums to 0.5 and 0.999. Batch size was set to 16 for each domain, and the input images were mean-centered/rescaled to $[-1, 1]$. All the used architectures replicate those of state-of-the-art methods, detailed in **SM**. We followed the protocol of unsupervised domain adaptation and did not use validation set to tune the hyper-parameters $\lambda_Q, \lambda_h, \lambda'_h$. Full hyper-parameter details for each experiment can be found in **SM**. We compare the proposed method with several related methods, including **CORAL** [38], **DANN** [12], **ADDA** [40], **DTN** [44], **UNIT** [22], **PixelDA** [3], **DIAL** [5], **DLD** [27], **DSN** [4], and **MCDA** [33] on digit classification task (Digit datasets), and the object recognition task (VisDA and PACS datasets).

4.1. Results On Digits Recognition

In this evaluation, we follow the same protocol across all methods. Specifically, we use the network structure similar to **UNIT** [22]. See SM for more details.

We show the accuracy of different methods (averaged over five different runs) in Tab. 1. The proposed method outperformed the competing methods in five out of six settings, confirming consistently and significantly better generalization of our model over target data.

The higher performance of the proposed model compared to other methods is mainly attributed to the proposed task-specific alignment method, which not only encourages the source features to be well-separated, according to their class label, but also aligns the target to source features in a cluster-wise manner, "matching" the source and target clusters. This is in contrast to the standard domain-wise alignment, which ignores the source/target inherent cluster

Table 1: Mean classification accuracy on digit classification. M: MNIST; MM: MNIST-M, S: SVHN, U: USPS. The best is shown in red. The superscript shows the standard deviation. ***UNIT** trains with the extended **SVHN** (> 500K images vs ours 72K). ***PixelDA** uses ($\approx 1,000$) of labeled target domain data as a validation set for tuning the hyper-parameters.

method	S → M	M → MM	M → U	MM → M	MM → U	U → M
Source Only	62.10	55.98	78.30	84.46	80.43	50.64
1-NN	35.86	12.58	41.22	82.13	36.90	38.45
CORAL [38]	$63.10^{0.8}$	$57.70^{0.7}$	$81.05^{0.6}$	84.90	87.54	$85.01^{0.5}$
DANN[13]	$73.80^{0.6}$	77.40	$81.60^{0.4}$	61.05	85.34	$77.40^{0.4}$
ADDA[40]	$77.68^{1.5}$	$91.47^{0.6}$	$90.51^{0.3}$	$92.82^{0.6}$	$80.70^{0.6}$	$90.10^{0.8}$
DTN[44]	$81.40^{0.6}$	$85.70^{0.4}$	$85.80^{0.4}$	$88.80^{0.5}$	$90.68^{0.4}$	$89.04^{0.3}$
PixelDA[3]	–	**98.10***	94.10*	–	–	–
UNIT[22]	90.6*	–	92.90	–		90.60
DSN[4]	$82.70^{0.3}$	$83.20^{0.4}$	$91.65^{0.3}$	$90.20^{0.3}$	$89.95^{0.2}$	$91.40^{0.3}$
MCDA[4]	**96.20**$^{0.4}$	–	$96.50^{0.7}$	–	–	$94.10^{0.3}$
Ours	$94.67^{0.5}$	**98.01**$^{0.3}$	**99.05**$^{0.3}$	**99.11**$^{0.2}$	**99.16**$^{0.3}$	**97.85**$^{0.3}$

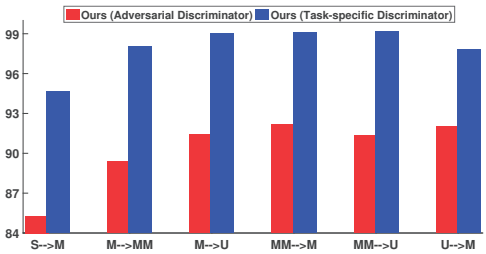


Figure 4: Comparison of proposed task-specific discriminator with the standard adversarial discriminator on Digit dataset.

structure. This superiority also benefits from the proposed source and target domain regularizers, which improve the source feature domain-invariance and the classifier's robustness respectively. See Sec. 4.4 for more details.

4.2. Results on Object Recognition

We also evaluate our method on two object recognition benchmark datasets **VisDA** [31] and **PACS** [21]. We follow **MCDA** [33], and use ResNet101 [16] as the backbone network which was pretrained on ImageNet dataset, and then finetune the parameters of ResNet101 with the source only **VisDA** dataset according to the procedure described in [33]. For the **PACS** dataset, we also follow the experimental protocol in [27], using ResNet18 [16] pretrained on ImageNet dataset, and training our model considering 3 domains as sources and the remaining as target, using all the images of each domain. For these experiments, we set the learning rate of resnets to 10^{-9}. We choose this small learning rate for ResNet parameters since the domain shift for both **VisDA** and **PACS** are significant, the training procedure benefits from a mild parameter updates back-propagated from the loss. Results for this experiment are summarized in Tab. 2 & Tab. 3. We observe that our model achieved, on average, the best performance compared to other competing methods for both datasets. The higher performance of our method is mainly attributed to incorporating the category-level information into the domain alignment through the proposed task-specific discriminator, which is beneficial to boost the discriminability of the source/target features.

4.3. Analysis of the task-specific discriminator

To measure how effective the new task-specific discriminator is, we conducted an experiment to compare the task-specific discriminator with the standard adversarial discriminator (training a logistic function on the discriminator by assigning labels 0 and 1 to the source and target domains respectively and training the encoder with inverted labels). The results are shown in Fig. 4. As is evident from the figure, there is a substantial increase in accuracy over all adaptation scenarios on switching from the standard adversarial discriminator to our task-specific discriminator. The superiority of the performance is mainly due to explicitly accounting for task knowledge in the proposed discriminator during adversarial training that encourages the discriminativity of the source/target samples in the feature space.

We further visualize the distribution of the learnt shared features to investigate the effect of task-specific discriminator (**Task-d**) and its comparison to adversarial discriminator (**Adv-d**). We use t-SNE [26] on **SVHN** to **MNIST** adaptation to visualize shared feature representations from two domains. Fig. 5 shows shared features from source (**SVHN**) and target (**MNIST**) before adaptation (a,d), after adaptation with **Adv-d** (b,e), and after adaptation with **Task-d** (c,f).

While a significant distribution gap is present between non-adapted features across domains (a), the domain discrepancy is significantly reduced in the feature space for both **Adv-d** (b) and **Task-d** (c). On the other hand, adaptation with **Task-d** led to pure and well-separated clusters in feature space compared to the adaptation with **Adv-d**, and leads to superior class separability. As supported by the quantitative results in Fig. 4, this implies that enforcing clustering in addition to domain-invariant embedding was essential for reducing the classification error. This is depicted in (f), where the points in the shared space are grouped into class-specific subgroups; color indicates the class label. This is in contrast to Fig. 5e, where the features show less class-specificity.

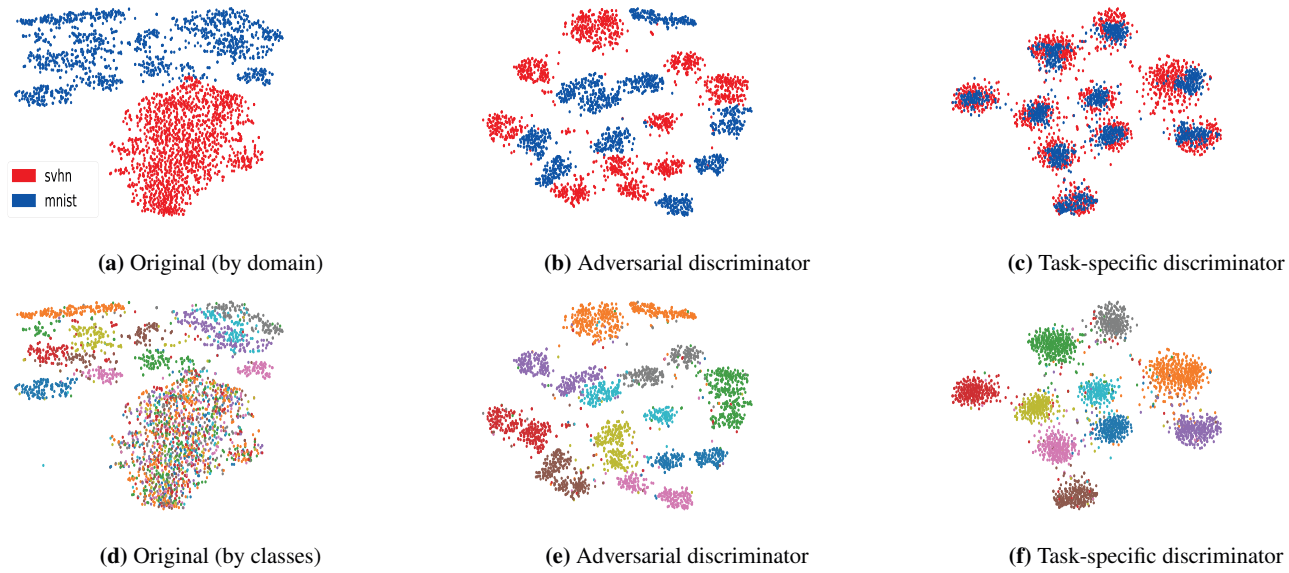
4.4. Ablation Studies

We performed an ablation study for our unsupervised domain adaptation approach on Digit dataset. Specifically, we considered training without source regularization, denoted as **Ours (w/o-s)**, training without target regularization, **Ours (w/0-t)**, and training by excluding both the source and the target regularization, **Ours (w/o-st)**.

The results are shown in Fig. 6. As can be seen, removing one or more of the objectives results in noticeable performance degradation. The more parts are removed, the worse the performance is. More precisely, disabling the source regularizer results in an average $\approx 3.5\%$ drop in performance. That demonstrates that the source regularizer can improve the generalization over target samples by encouraging the source

Table 2: Accuracy of ResNet101 model fine-tuned on the VisDA dataset. Last column shows the average rank of each method over all classes. The best in bold red, second best in red.

Method	plane	bcycl	bus	car	horse	knife	mcycl	person	plant	sktbrd	train	truck	mean	Ave. ranking
Source Only	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	4.91
MMD [24]	87.1	63.0	76.5	42.0	**90.3**	42.9	**85.9**	53.1	49.7	36.3	**85.8**	20.7	61.13	3.08
DANN [12]	81.9	77.7	82.8	44.3	81.2	29.5	65.1	28.6	51.9	**54.6**	82.8	7.8	57.42	3.00
MCDA [33]	87.0	60.9	**83.7**	64.0	88.9	79.6	84.7	**76.9**	88.6	40.3	83.0	**25.8**	71.90	2.41
Ours	**88.2**	**78.5**	79.7	**71.1**	90.0	**81.6**	84.9	72.3	**92.0**	52.6	82.9	18.4	**74.03**	**1.83**



(a) Original (by domain) **(b)** Adversarial discriminator **(c)** Task-specific discriminator

(d) Original (by classes) **(e)** Adversarial discriminator **(f)** Task-specific discriminator

Figure 5: Feature visualization for embedding of digit datasets for adapting **SVHN** to **MNIST** using t-SNE algorithm. The first and the second rows show the domains and classes, respectively, with color indicating domain and class membership. (a,d) Original features. (b,e) learned features for Ours with (binary) adversarial discriminator. (c,f) learned features for Ours with task-specific discriminator.

features to be domain-invariant, less informative about the identity of either of the domains. Immobilizing the target regularizer leads to $\approx 2.0\%$ average drop in performance. These results strongly indicate that it is beneficial to make use of the information from unlabeled target data the during classifier learning process, which further strengthens the feature discriminability in the target domain. Finally, the average performance drop that stems from disabling both the source and the target regularizer is $\approx 5.5\%$. This suggests that the two components operate in harmony with each other, forming an effective solution for domain adaptation.

Table 3: Mean classification accuracy on **PACS** dataset. The first row indicates the target domain, while all the others are considered as sources. The best (in bold red), the second best (in red).

method	Sketch	Photo	Art	Cartoon	Mean
Resnet18 (Source Only)	60.10	92.90	74.70	72.40	75.00
DIAL [5]	66.80	**97.00**	87.30	85.50	84.20
DLD [27]	69.60	**97.00**	87.70	86.90	85.30
Ours	**71.69**	96.81	**89.48**	**88.91**	**86.72**

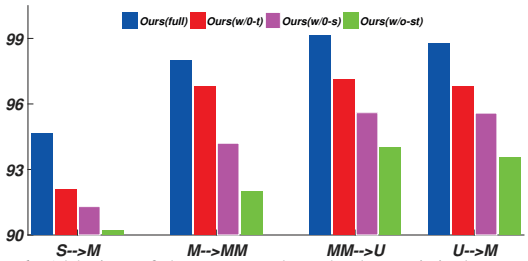


Figure 6: Ablation of the proposed method on Digit dataset. The regularization terms contribute to the overall performance.

5. Conclusion

We proposed a method to boosts the unsupervised domain adaptation by explicitly accounting for task knowledge in the cross-domain alignment discriminator, while simultaneously exploiting the agglomerate structure of the unlabeled target data using important regularization constraints. Our experiments demonstrate the proposed model achieves state-of-the-art performance across several domain adaptation benchmarks.

References

[1] M. Arjovsky and L. Bottou. Towards principled methods for training generative adversarial networks. *CoRR*, abs/1701.04862, 2017.

[2] J. Blitzer, S. Kakade, and D. P. Foster. Domain adaptation with coupled subspaces. In *International Conference onArtificial Intelligence and Statistics (AISTATS)*, pages 173–181, 2011.

[3] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, page 7, 2017.

[4] K. Bousmalis, G. Trigeorgis, N. Silberman, D. Krishnan, and D. Erhan. Domain separation networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 343–351, 2016.

[5] F. M. Carlucci, L. Porzi, B. Caputo, E. Ricci, and S. R. Bulò. Just dial: Domain alignment layers for unsupervised domain adaptation. In *International Conference on Image Analysis and Processing*, pages 357–369. Springer, 2017.

[6] M. Chen, K. Q. Weinberger, and J. Blitzer. Co-training for domain adaptation. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2456–2464, 2011.

[7] G. Csurka. A comprehensive survey on domain adaptation for visual applications. pages 1–35. Springer, 2017.

[8] Z. Dai, Z. Yang, F. Yang, W. W. Cohen, and R. R. Salakhutdinov. Good semi-supervised learning that requires a bad gan. In *Advances in Neural Information Processing Systems*, pages 6510–6520, 2017.

[9] J. Donahue, J. Hoffman, E. Rodner, K. Saenko, and T. Darrell. Semi-supervised domain adaptation with instance constraints. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 668–675, 2013.

[10] B. Fernando, A. Habrard, M. Sebban, and T. Tuytelaars. Unsupervised visual domain adaptation using subspace alignment. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2960–2967, 2013.

[11] G. French, M. Mackiewicz, and M. Fisher. Self-ensembling for visual domain adaptation. In *International Conference on Learning Representation (ICLR)*, 2018.

[12] Y. Ganin and V. Lempitsky. Unsupervised domain adaptation by backpropagation. *International Conference on Machine Learning (ICML)*, 2015.

[13] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016.

[14] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[15] Y. Grandvalet and Y. Bengio. Semi-supervised learning by entropy minimization. In *Advances in neural information processing systems*, pages 529–536, 2005.

[16] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[17] L. Hu, M. Kan, S. Shan, and X. Chen. Duplex generative adversarial network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1498–1507, 2018.

[18] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representation (ICLR)*, 2015.

[19] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[20] A. Kumar, A. Saha, and H. Daume. Co-regularization based semi-supervised domain adaptation. In *Advances in Neural Information Processing Systems (NIPS)*, pages 478–486, 2010.

[21] D. Li, Y. Yang, Y.-Z. Song, and T. M. Hospedales. Deeper, broader and artier domain generalization. In *IEEE International Conference on Computer Vision (ICCV)*, pages 5543–5551. IEEE, 2017.

[22] M.-Y. Liu, T. Breuel, and J. Kautz. Unsupervised image-to-image translation networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 700–708, 2017.

[23] M.-Y. Liu and O. Tuzel. Coupled generative adversarial networks. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems (NIPS)*, pages 469–477. Curran Associates, Inc., 2016.

[24] M. Long, Y. Cao, J. Wang, and M. I. Jordan. Learning transferable features with deep adaptation networks. *International Conference on Machine Learning (ICML)*, 2015.

[25] M. Long, H. Zhu, J. Wang, and M. I. Jordan. Unsupervised domain adaptation with residual transfer networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 136–144, 2016.

[26] L. v. d. Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.

[27] M. Mancini, L. Porzi, S. Rota Bulò, B. Caputo, and E. Ricci. Boosting domain adaptation by discovering latent domains. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3771–3780, 2018.

[28] T. Ming Harry Hsu, W. Yu Chen, C.-A. Hou, Y.-H. Hubert Tsai, Y.-R. Yeh, and Y.-C. Frank Wang. Unsupervised domain adaptation with imbalanced cross-domain data. In *IEEE International Conference on Computer Vision (ICCV)*, pages 4121–4129, 2015.

[29] P. Morerio, J. Cavazza, and V. Murino. Minimal-entropy correlation alignment for unsupervised deep domain adaptation. In *International Conference on Learning Representation (ICLR)*, 2018.

[30] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2):199–210, 2011.

[31] X. Peng, B. Usman, N. Kaushik, J. Hoffman, D. Wang, and K. Saenko. Visda: The visual domain adaptation challenge. *arXiv preprint arXiv:1710.06924*, 2017.

[32] K. Saito, Y. Ushiku, and T. Harada. Asymmetric tri-training for unsupervised domain adaptation. *International Conference on Machine Learning (ICML)*, 2017.

[33] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3723–3732, 2018.

[34] S. Sankaranarayanan, Y. Balaji, C. D. Castillo, and R. Chellappa. Generate to adapt: Aligning domains using generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8503–8512, 2018.

[35] O. Sener, H. O. Song, A. Saxena, and S. Savarese. Learning transferrable representations for unsupervised domain adaptation. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2110–2118, 2016.

[36] B. K. Sriperumbudur, A. Gretton, K. Fukumizu, B. Schölkopf, and G. R. Lanckriet. Hilbert space embeddings and metrics on probability measures. *Journal of Machine Learning Research*, 11(Apr):1517–1561, 2010.

[37] B. Sun, J. Feng, and K. Saenko. Return of frustratingly easy domain adaptation. *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[38] B. Sun and K. Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European Conference on Computer Vision (ECCV)*, pages 443–450. Springer, 2016.

[39] N. Tishby and N. Zaslavsky. Deep learning and the information bottleneck principle. In *IEEE Information Theory Workshop (ITW)*, pages 1–5. IEEE, 2015.

[40] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, page 4, 2017.

[41] H. Yan, Y. Ding, P. Li, Q. Wang, Y. Xu, and W. Zuo. Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2272–2281, 2017.

[42] W. Zellinger, T. Grubinger, E. Lughofer, T. Natschläger, and S. Saminger-Platz. Central moment discrepancy (cmd) for domain-invariant representation learning. *International Conference on Learning Representation (ICLR)*, 2017.

[43] W. Zhang, W. Ouyang, W. Li, and D. Xu. Collaborative and adversarial network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3801–3809, 2018.

[44] X. Zhang, F. X. Yu, S.-F. Chang, and S. Wang. Deep transfer network: Unsupervised domain adaptation. *arXiv preprint arXiv:1503.00591*, 2015.