Semi-supervised Domain Adaptive Structure Learning

Can Qin, Lichen Wang, Qianqian Ma, Yu Yin, Huan Wang, and Yun Fu, *Fellow, IEEE*

Abstract—Semi-supervised domain adaptation (SSDA) is quite a challenging problem requiring methods to overcome both 1) overfitting towards poorly annotated data and 2) distribution shift across domains. Unfortunately, a simple combination of domain adaptation (DA) and semi-supervised learning (SSL) methods often fail to address such two objects because of training data bias towards labeled samples. In this paper, we introduce an adaptive structure learning method to regularize the cooperation of SSL and DA. Inspired by the multi-views learning, our proposed framework is composed of a shared feature encoder network and two classifier networks, trained for contradictory purposes. Among them, one of the classifiers is applied to group target features to improve intra-class density, enlarging the gap of categorical clusters for robust representation learning. Meanwhile, the other classifier, serviced as a regularizer, attempts to scatter the source features to enhance the smoothness of the decision boundary. The iterations of target clustering and source expansion make the target features being well-enclosed inside the dilated boundary of the corresponding source points. For the joint address of cross-domain features alignment and partially labeled data learning, we apply the maximum mean discrepancy (MMD) distance minimization and self-training (ST) to project the contradictory structures into a shared view to make the reliable final decision. The experimental results over the standard SSDA benchmarks, including DomainNet and Office-home, demonstrate both the accuracy and robustness of our method over the state-of-the-art approaches.

Index Terms—Semi-supervised Domain Adaptation, Multi-views, Self-training, Adaptive Structure Learning

I. INTRODUCTION

RECENTLY, Deep Neural Networks (DNNs) have been extensively utilized in various tasks, e.g., image classification, text translation, and news recommendation [1], [2], [3], [4], ranging from computer vision to natural language understanding. However, despite the remarkable success, the DNNs heavily rely on huge amounts of annotated data for training, which are costly to obtain in practice.

Domain Adaptation (DA) [5], [6], [7], [8], defined as transferring/adapting the model trained from the rich annotated dataset (i.e., source domain) to the label-scarce target domain, is one of the promising solutions to address such a data-deficiency challenge. In general, domain adaptation can be

Can Qin, Lichen Wang, Yu Yin, Huan Wang are with the Department of Electrical and Computer Engineering, Northeastern University, Boston, USA (Email: qin.ca@northeastern.edu, wanglichenxj@gmail.com, {yin.yu1, wang.huan}@northeastern.edu).

Qianqian Ma is with the Department of Electrical and Computer Engineering, Boston University, Boston, USA (Email: maqq@bu.edu).

Yun Fu is with the Department of Electrical and Computer Engineering, and Khoury College of Computer Science, Northeastern University, Boston, USA (Email: yunfu@ece.neu.edu).
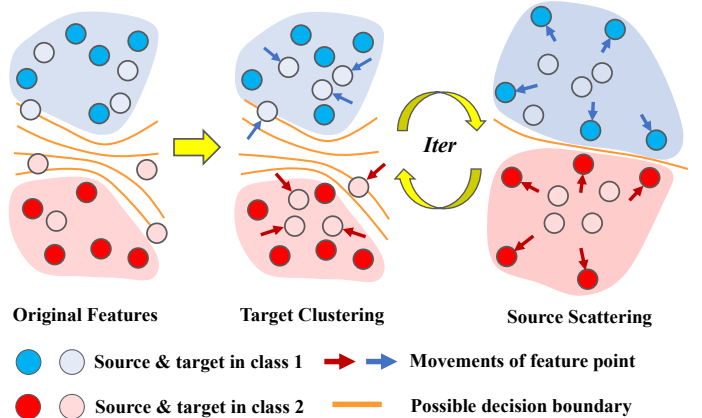

Fig. 1. The illustration of our motivation. In Adaptive Structure Learning (ASL), there are alternative optimizations between source scattering and target clustering to explicitly align the cross-domain features. In the end, the target features are expected to be well enclosed within the expanded boundary of corresponding source features.

classified as unsupervised domain adaptation (UDA) and semi-supervised domain adaptation (SSDA) according to the access to target labels during training. This paper focuses on SSDA, with significantly better performance than UDA, given a tiny amount of labeled target samples.

However, considerable distribution mismatches (i.e., domain gaps) between cross-domain data/features degenerate most conventional machine learning models built upon the distribution-sharing assumption. Feature alignment approaches attempt to match the cross-domain features explicitly by minimizing certain divergence or distances over the features projected in a shared feature space. How to define such a distance/divergence is the key to success. Several methods, such as Correlation Alignment (CORAL) [9], Geodesic Flow Kernel (GFK) [10], [11], Maximum Mean Discrepancy (MMD) [12], have been developed. In recent years, implicit feature alignment methods have gained increasing popularity with some typical methods including DIRT-T [13], MCD [6], MME [14], UODA [15], Contrastive Domain Adaptation [16], [17] and other adversarial alignment methods such as DANN [18] or ADDA [19]. Specifically, MCD [6] deploys two classifiers to learn two different boundaries, where the difference knowledge between the two boundaries are used in adversarial learning mechanisms for implicit alignment. MME [14] and DIRT-T [13] employ entropy loss minimization strategy to group target domain features as the implicit conditional alignment by enforcing the well-clustered features. UODA [15] extends the MME [14] and MCD [6] by employing two

classifiers with opposite objects for learning a well-structural latent space where the cross-domain features are implicitly aligned. Moreover, some other DA methods utilize contrastive learningn [16], [17] without any explicit alignment loss, which is also the representative method of this type. Inspired by the development of Generate Adversarial Network (GAN) [20], the typical adversarial training solutions employ a zero-sum game between a domain classifier (i.e., discriminator) and a feature extractor to enforce the confusion of cross-domain features until the discriminator cannot tell any difference [18], [19]. However, the conditional distributions alignment still remains challenging due to the lack of target labels.

MME [6] and DIRT-T [13] employed entropy loss minimization to group target domain features for conditional alignment by the assumption that well-clustered features are more discriminative. However, the classifier is also trained by the cross-entropy loss over the labeled data from the source domain. Despite such a structure-wise regularization, the model is still strongly biased towards the source domain due to the inequality of labeled data across domains, inevitably leading to two negative impacts: 1) the decision boundary easily crosses the region of high-density target features mistakenly, and 2) the target domain features close to the decision boundary are likely driven to the wrong areas during alignment. The low-density region of target domain features cannot be well learned if the cross-domain features are not firmly aligned, as the decision boundary may easily cross the region of high-density target features mistakenly.

Inspired by the noisy/soft labels commonly applied in weakly/semi-supervised learning, it is logical to assume that the decision boundary can be de-biased by the regularizer with slightly disturbed and scattered patterns. With this regard, UODA [15] summarized that the ideal representation for SSDA should include conflicting aspects: 1) well-clustered target features and 2) scattered source features. Accordingly, this paper further extends this work as a more powerful adaptive structure learning framework for the semi-supervised domain adaptation problem, with the help of a generator and two classifier networks for different purposes.

However, UODA [15] has two main limitations. Firstly, it overlooks the explicit alignment of cross-domain features, which provides a strong constraint to avoid mis-alignments. Secondly, it has not fully exploited the consistency between different data transformations, which is commonly adopted in SSL and has the great potential to learn more representative features. This paper attempts to extend UODA [15] to address the both challenges. The minimization of MMD loss over the deep features helps to solve the first weakness. Such explicit alignment in the reproducing kernel Hilbert space (RKHS) has further supported the structure learning for precise conditional alignment. To solve the problem two, we utilize self-training to automatically achieve pseudo labels by enforcing the consistency between weakly and strongly transformed data. Such data transformations naturally bring the self-supervision for instance-wise regularization. The final scores on unlabeled data are jointly inferred by the two views/classifiers that provide complementary information for robust decisions. The whole pipeline can be trained in the end-to-end manner to ensure excellent cooperation between different modules, expected to overcome both domain gaps and data bias.

In summary, this paper has three major contributions as follows:

- The implicit feature alignment based on contradictory structure learning might bring negative transfer due to conditional mismatch. This paper introduces the explicit alignment loss to assist the implicit one for the more precise alignment.
- Instead of focusing on opposite structure learning, this paper pays more attention to reunite such contradictions, which are also crucial. To fulfill it, we take the self-training to enforce the consistency among the predictions of differently augmented data along with the opposite structure learning.
- Extensive experiments on the popular benchmarks, such as Office-home [21] and DomainNet [22], demonstrate the advantage of our proposed approach over the direct baseline [15] and other latest methods.[1]

II. RELATED WORKS

A. Unsupervised Domain Adaptation

In conventional machine learning scenarios, the training and testing sets are supposed to share the same feature distributions. It is the essential assumption for most of the algorithms. However, as machine learning technologies extend to real-world applications, such as pandemic prediction [23], and explore diverse data formats, the assumption is not always the truth anymore. Unsupervised Domain Adaptation (UDA) aims to adopt a model from the source domain to the target domain without supervision. Specifically, any labels from the target samples are inaccessible. Therefore, it is a more challenging task compared with the semi-supervised scenario. There are multiple UDA methods have been proposed [24], [18], [25], [26], [27], [28], [29] in recent years which achieved promising results. These methods could be grouped into three categories, which are 1) source-target divergence metric-based methods, 2) generalization extension-based methods, and 3) constant term-based methods [30]. The main strategy of the first category is projecting both source and target samples into a latent common space, where these samples could be well aligned by minimizing the divergence.

Metric learning [31], [32] used to be a popular solution for UDA. As Deep Neural Networks have achieved considerable progress in recent decent, they are naturally explored in UDA to obtain representations. The straightforward strategy is projecting the source and target features into a common subspace where the divergence/shift across different domains would be minimized. [33] designed a correlation learning strategy that effectively aligns different domains. Maximum Mean Discrepancy is set as the metric for UDA [12]. Although these methods achieved promising results, however, the distance metrics are fixed, which cannot well handle the various representations from different domains. Adversarial learning provides another

---

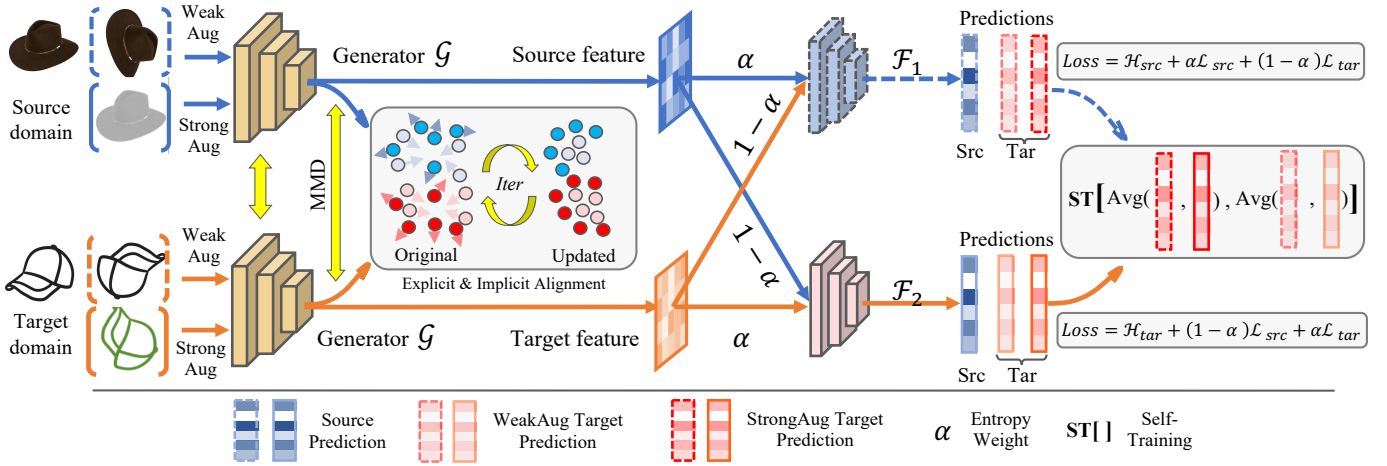[1] Our code can be seen at: https://github.com/canqin001/ASDA

Fig. 2. Illustration of the proposed Adaptive Structure Learning for Semi-supervised Domain Adaptation (ASDA) including three modules: 1) a deep feature encoder network $\mathcal{G}$, 2) a source-scattering classifier network $\mathcal{F}_1$, and 3) a target-clustering classifier network $\mathcal{F}_2$. This paper takes a Convolutional Neural Network as $\mathcal{G}$ and a two-layer Feed-forward Neural Network as either $\mathcal{F}_1$ or $\mathcal{F}_2$. The raw data will be transformed into different formats as inputs according to the WeakAug and StrongAug operations. In this figure, both generators (in yellow) share the parameters for feature extraction. The two classifiers will take the features from the generator for classification. More details about the loss functions can be referred in Sec. III.

strategy to evaluate the domain shift where a discriminator, as well as encoders, are deployed to competently obtain the domain-invariant representations (i.e., Gradient Reversal [18], ADDA [19]). Specifically, CyCADA [34] is an extension of the ADDA with the cycle consistency/translation [35] to generate synthetic images consisting of both target styles and source content.

Although adversarial learning is effective in reducing the domain shift, the conditional distribution mismatch is still an unsolved problem. The multi-classifier framework is proposed in Tri-training [36], which derives pseudo labels from multiple views. DIRT-T [13] fully explores the target features in both the feature space and label space. The density of the intra-class knowledge and the divergence of the inter-class knowledge is explored for tuning the classification boundaries. MCD [14] proposes another solution that deploys two classifiers to learn two different boundaries, where the differences of the boundaries are used in adversarial learning mechanisms to obtain the domain-invariant representations.

Graph matching methods are also popular in domain adaptation [37], [38], [39]. [37] embeds the cross-domain features into a shared latent space with the maximization of similarity measured by the spectrum distance of the graph Laplacians. [38] proposes the graph-based domain adaptation method based on the transferring of frequency content where the cross-domain features between domains are aligned by the Fourier bases of the two graphs. [39] applies the hyper-graph matching over multiple orders to address the unsupervised domain adaptation. For efficient computation, [39] also proposes a customized optimization routine for graph-matching.

However, due to the inaccessibility of target labels, it is tough to describe the conditional distribution of target features precisely. Thus, UDA methods have limited potentials for real-world applications.

B. Semi-supervised Domain Adaptation

Semi-supervised Domain Adaptation (SSDA) assumes that a few samples in the target domain are labeled, and the corresponding labels are accessible in the training stage. Due to this, it is also named Few-shot Domain Adaptation (FSDA) scenario. SSDA is a more practical setting in real-world applications since labeling a small number of samples is cost-efficient, and it could lead to considerable performance improvement. Furthermore, by exploring the supervision/label information, the distribution knowledge of the target domain could be fully explored in a more fine-grained way, which potentially improves the domain-adaptation performance. However, there are still drawbacks. Specifically, it is difficult to automatically balance the learning contribution between the source and target domains, which could cause the overfitting issue in the target domain and reduce the generalization ability and overall test performance.

There are a few works proposed for solving the challenges. [40] explores the label correlations and transfers the learned semantic knowledge across the source and target domains. Soft label scores, as well as the matching loss, are utilized in this method. CCSA [41] effectively explores the latent label correlation knowledge from the target samples for conditional distribution matching. The feature and label consistency from source and target domains are jointly optimized via separation loss in a maximization manner. FADA [42] is an extension of the CCSA method. Specifically, an extra adversarial learning module is designed, which includes multiple discriminators as well as a generator. The goal is to further stabilize and tune the learned domain invariant representations in different classes. MME [6] jointly considers the source and target samples' predictions and maximizes the entropy of the predictions across domains. Moreover, the clustering strategy obtains the prediction confidence and gets more fine-grain entropy loss for encoder training. However, due to the considerable unbalanced sample numbers between source and target domains, it is

still difficult to control the learned decision boundaries in the target domain. UODA [15] is proposed to address the above challenge by the opposite structure learning. However, considering the relatively small number of the labeled target samples, the unpredictable sample distributions (e.g., density and divergence) could further confuse the classifiers in the target domain. CDAC [43] is the latest technique that employs both adaptive adversarial clustering and pseudo labeling with a set of data augmentations. The adversarial clustering performs minimax training on unlabeled target domain samples to drive features of similar samples in the target domain compact but enlarge the distance of the dissimilar features. Moreover, it proposes a consistency loss and uses the pseudo labeling loss with two groups of data augmentation. Self-training is a well-explored technique widely applied in semi-supervised learning [44]. [45] has found that the data distribution has connectivity within the manifold of a ground-truth class and provides ground-truth accuracy guarantees for self-training algorithms which regularize input consistency on unlabeled data. This paper further improves conventional self-training methods based on the two classifiers/views, which provide complementary information for making the robust decision. Furthermore, the confusion of the classifiers in UODA can also be corrected by the explicit alignment cooperated with the implicit structure learning.

C. Transferability and Discriminability

The tradeoff of domain-invariance and feature discriminability has a long history [46], [47], [48], [49]. More specifically, [46] claims that the eigenvectors with the largest singular values dominate the feature transferability and proposes the Batch Spectral Penalization of the largest singular values to keep the discriminability where the transferability is enhanced at the expense of over-penalization of other eigenvectors. According to this statement, we can infer that the degeneration of discrimination is caused by over-alignment instead of misalignment. [47] has observed that adversarial feature adaptation may potentially deteriorate the adaptability and proposes Transferable Adversarial Training of classifiers to avoid the over-alignment. Similarly, [48] provides a detailed theoretical study to analyze the tradeoff between learning invariant representations and achieving small joint errors in both domains. [49] proposes the domain-adaptive extreme learning machine (ELM) where the source features are transferred to train an ELM classifier to predict transferred target features under a unified latent space. We have a similar observation that excessive alignment will degenerate the discrimination of target features. Moreover, we think cross-domain misalignment and target-domain discrimination are both critical.

III. PROPOSED APPROACH

A. Problem Setting and Motivation.

In SSDA, the model is expected to be well generalized on the target domain with fully labeled source samples and partially labeled target samples. To formulate this, suppose $\mathcal{S} = \{\boldsymbol{x}_i^s, y_i^s\}_{i=1}^{N_s}$ is the set of the source domain data, $\mathcal{T} = \{\boldsymbol{x}_i^t, y_i^t\}_{i=1}^{N_t}$ is the set of labeled target data, where

TABLE I
NOTATIONS OF SYMBOLS

Symbol	Descriptions
x_i^s, y_i^s	the i-th source image data and groundtruth label
x_i^t, y_i^t	the i-th labeled target image data and groundtruth label
x_i^u	the i-th unlabeled target image data
f_i	the deep feature of the i-th input data
$\mathcal{S}, \mathcal{T}, \mathcal{U}$	the sets of source, labeled and unlabeled target domain
$\mathcal{G}(\cdot), \mathcal{F}_1(\cdot), \mathcal{F}_2(\cdot)$	encoder, source-scattering and target-clustering classifier
d, K	dimensions of feature space and label space
N_s, N_t, N_u	source, labeled and unlabeled target data point numbers
α, β, λ	loss weights
x', x''	weakly augmented and strongly augmented data
$\kappa(\cdot)$	kernel function for MMD computation
$\sigma(\cdot)$	softmax function
$\mathbb{1}_{[\cdot]}$	identity function
$\Theta_{\mathcal{G}}$	parameters of feature encoder network $\mathcal{G}$
$\Theta_{\mathcal{F}_1}$	parameters of source scattering classifier $\mathcal{F}_1$
$\Theta_{\mathcal{F}_2}$	parameters of target clustering classifier $\mathcal{F}_2$
$p_1(\cdot), p_2(\cdot)$	predictions of classifier $\mathcal{F}_1$ and $\mathcal{F}_2$
$\Gamma(\cdot)$	average of two classifiers' predictions

$N_s = |S|$ and $N_t = |T|$ is the size of the dataset $\mathcal{S}$ and $\mathcal{T}$, respectively. $\boldsymbol{x}_i^s$ and $\boldsymbol{x}_i^t$ represents the labeled source and target images, respectively, and y^s, y^t denote the corresponding labels. Apart from the labeled data, the unlabeled target image set $\mathcal{U} = \{\boldsymbol{x}_i^u\}_{i=1}^{N_u}$, on which the model is expected to be well generalized, is composed of N_u images where $N_u = |\mathcal{U}|$ and $N_u \gg N_t$. Due to the distributions mismatch across domains, i.e., $p(\boldsymbol{x}^s) \neq p(\boldsymbol{x}^u)$, and imbalance between labeled and unlabeled data, i.e., $N_u \gg N_t$, the model usually performs poorly if trained only by the supervised learning method.

As shown in Fig. 2, there are three modules for our proposed model: 1) the siamese feature extractor networks $\mathcal{G}(\cdot)$, 2) the source-scattering classifier $\mathcal{F}_1(\cdot)$, and 3) the target-clustering classifier $\mathcal{F}_2(\cdot)$. Specifically, the cross-domain data would be projected into a shared latent space with the help of siamese feature extractor networks. The source-scattering is designed to disturb the source features as the structure regularization. The target-clustering classifier implicitly groups the target features expected to be enclosed by the corresponding stretched source features. Next, We will explain in details of the proposed framework as well as the training procedure.

B. Supervised Training.

Our model requires minimizing the empirical risk on both the labeled source and target sets as conventional DA methods. Therefore, we take a Convolutional Neural Network (CNN) as the feature extractor network $\mathcal{G}(\cdot)$ to map the image $\boldsymbol{x}$ into the d-dimensional feature $\boldsymbol{f} \in \mathbb{R}^d$ as:

$$\boldsymbol{f} = \mathcal{G}(\boldsymbol{x}, \Theta_{\mathcal{G}}), \quad (1)$$

where $\boldsymbol{x}$ can come from either domains, and $\Theta_{\mathcal{G}}$ represents the parameter set of $\mathcal{G}(\cdot)$. Then, the classifiers $\mathcal{F}_1(\cdot)$ and $\mathcal{F}_2(\cdot)$ would classify the input feature $\boldsymbol{f}$ into K categories as following:

$$\begin{aligned} \boldsymbol{p}_1(y|\boldsymbol{x}) &= \sigma\left(\mathcal{F}_1(\mathcal{G}(\boldsymbol{x}), \Theta_{\mathcal{F}_1})\right), \\ \boldsymbol{p}_2(y|\boldsymbol{x}) &= \sigma(\mathcal{F}_2(\mathcal{G}(\boldsymbol{x}), \Theta_{\mathcal{F}_2})), \end{aligned} \quad (2)$$

where $\sigma(\cdot)$ is the softmax function and $\boldsymbol{p}_1(y|\boldsymbol{x})$ and $\boldsymbol{p}_2(y|\boldsymbol{x}) \in \mathbb{R}^K$ are the K-dimensional softmax scores of classifiers $\mathcal{F}_1(\cdot)$

and $\mathcal{F}_2(\cdot)$, which are parameterized by the $\Theta_{\mathcal{F}_1}$ and $\Theta_{\mathcal{F}_2}$. The two classifiers, which are Feed-forward Neural Networks (FFN), are set as the same architecture with different parameters.

For model training, the empirical loss is composed of two items: source-domain supervision $\mathcal{L}_{src}$ and the target-domain supervision $\mathcal{L}_{tar}$ formulated as:

$$\mathcal{L}_{src} = -\mathbb{E}_{(\boldsymbol{x}^s, y^s) \sim \mathcal{S}} \left[\sum_{k=1}^{K} \mathbb{1}_{k=y^s} \log(p_1(y = y^s | \boldsymbol{x}^s)) \right], \quad (3)$$

$$\mathcal{L}_{tar} = -\mathbb{E}_{(\boldsymbol{x}^t, y^t) \sim \mathcal{T}} \left[\sum_{k=1}^{K} \mathbb{1}_{k=y^t} \log(p_2(y = y^t | \boldsymbol{x}^t)) \right]. \quad (4)$$

The discrepancy between the two classifiers plays an important role for model training. Therefore, to enforce $\Theta_{\mathcal{F}_1} \neq \Theta_{\mathcal{F}_2}$, we apply different weights on both supervision losses $\mathcal{L}_{src}$ and $\mathcal{L}_{tar}$ where $\mathcal{F}_1(\cdot)$ takes $\alpha \mathcal{L}_{src} + (1 - \alpha)\mathcal{L}_{tar}$, and $(1 - \alpha)\mathcal{L}_{src} + \alpha \mathcal{L}_{tar}$ is for $\mathcal{F}_2(\cdot)$ with α assigned as 0.75.

C. Features Alignment.

To bridge the domain gap, this paper proposes the alignment of the features in both explicit and implicit ways. For explicit alignment, we directly minimize the MMD loss over the cross-domain features, and learning the opposite feature structures can be regarded as the implicit alignment.

Implicit Alignment: As separate views, the two classifiers are assigned with different purposes to learn the easy-to-adapt structure expected to have the dense target features enclosed by the sparse source features. The minimization of conditional entropy on output softmax scores enforces the high-confident predictions for features clustering. To fulfill such an object, this paper applies the entropy loss to measure features' closeness. Therefore, the conditional entropy over the unlabeled target domain can be denoted as:

$$H_{tar} = -\mathbb{E}_{\boldsymbol{x}^u \sim \mathcal{U}} \left[\sum_{k=1}^{K} p_2(y = k | \boldsymbol{x}^u) \log p_2(y = k | \boldsymbol{x}^u) \right], \quad (5)$$

where $p_2(y = k | \boldsymbol{x}^u)$ represents the possibility of data $\boldsymbol{x}^u$ as the class k.

Feature scattering can be simply seen as the reverse of feature clustering. To this end, we will also take the conditional entropy loss H_{src} to implement the source feature expansion:

$$H_{src} = -\mathbb{E}_{\boldsymbol{x}^s \sim \mathcal{S}} \left[\sum_{k=1}^{K} p_1(y = k | \boldsymbol{x}^s) \log p_1(y = k | \boldsymbol{x}^s) \right], \quad (6)$$

where H_{src} would be optimized in the reverse way as H_{tar} for learning the opposite structure.

Explicit Alignment: It is hard to say that the cross-domain features are well aligned only with the implicit alignment adopted in UODA [15] and MME [6]. Instead, such a structure-wise constraint cannot precisely ensure the match between the corresponding feature groups across domains. Therefore, narrowing certain divergence/distance of cross-domain features plays a crucial role in closing the relaxing gap inherited from implicit alignment. In this paper, we minimize

the MMD [50], [12] loss to align cross-domain features explicitly as:

$$\mathcal{L}_{align} = \frac{1}{N_s N_s} \sum_{i,j=1}^{N_s} \kappa(\boldsymbol{f}_i^s, \boldsymbol{f}_j^s) + \frac{1}{N_s N_u} \sum_{i,j=1}^{N_s, N_u} \kappa(\boldsymbol{f}_i^s, \boldsymbol{f}_j^t)$$
$$+ \frac{1}{N_u N_u} \sum_{i,j=1}^{N_u} \kappa(\boldsymbol{f}_i^t, \boldsymbol{f}_j^t), \quad (7)$$

where $\kappa(\cdot)$ is a kernel function and we apply the combination of Linear and Radial Basis Function (RBF) kernel in our model. $\boldsymbol{f}^s$ and $\boldsymbol{f}^t$ are the deep features of source domain and target domain, respectively.

D. Views-consistent Self-training.

Although the domain alignment is proposed to address the distribution shift, the over-fitting problem is still unsolved yet. Inspired by the techniques from semi-supervised learning [44], the variant augmented data are crucial to enriching the training set, which can be naturally applied to regularize the decision boundary towards smoothing. This paper also employs the self-training strategy where both weakly augmented data $\mathcal{U}' = \{\boldsymbol{x}_i^{u'}\}_{i=1}^{N_u'}$ and strongly augmented data $\mathcal{U}'' = \{\boldsymbol{x}_i^{u''}\}_{i=1}^{N_u''}$ can be automatically obtained as $\mathcal{U}' = WeakAug(\mathcal{U})$ and $\mathcal{U}'' = StrongAug(\mathcal{U})$. The weakly augmenting includes random horizontal flip and random crop, and the strong augmentation randomly selects two transformations from the ten choices [44]. The hard pseudo labels are obtained by the weakly augmented data with the highest confidence among all the classes, i.e., $\hat{y}' = \mathrm{argmax}(\Gamma(x^{u'}))$ where $\Gamma(\cdot) = \frac{p_1(\cdot) + p_2(\cdot)}{2}$ denotes the average predictions of two classifiers. Such pseudo labels are applied as the ground truth for self-training:

$$\mathcal{L}_{st} = -\mathbb{E}_{\boldsymbol{x}^{u'} \sim \mathcal{U}', \boldsymbol{x}^{u''} \sim \mathcal{U}''} \left[\sum_{k=1}^{K} \mathbb{1}_{\Gamma(y=k|x^{u'})>\tau} \hat{y}' \right.$$
$$\left. \log \left(\Gamma(y = k | x^{u''}) \right) \right], \quad (8)$$

where a threshold τ is applied to filter the high-confident pseudo labels and $\mathbb{1}_{[\cdot]}$ is the identity function. For convenient denotation, the weakly augmented unlabeled target set $\mathcal{U}'$ is equal to the original unlabeled target set $\mathcal{U}$ in the practice.

E. Overall Training.

In this paper, there are three groups of parameters, i.e, $\Theta_{\mathcal{F}_1}$, $\Theta_{\mathcal{F}_2}$ and $\Theta_{\mathcal{G}}$, to be optimized based on multiple loss functions. The back-propagation of overall gradients starts from the two classifiers, and the cross-entropy (Eq. 3 and Eq. 4) is an important target of all the objectiveness. We firstly apply the different weights on the task losses to enforce the difference between $\Theta_{\mathcal{F}_1}$ and $\Theta_{\mathcal{F}_2}$. Then, we maximize the target-domain entropy loss (Eq. 5) to update the class-wise prototypes and minimize the source-domain entropy loss (Eq. 6) to make source features slightly gathered. Moreover, the self-training

Algorithm 1: Overall Training

---

1 Input: Labeled source set $\mathcal{S}$, labeled target set $\mathcal{T}$ and unlabeled target set $\mathcal{U}$ (i.e., $\mathcal{U}'$). Strongly augmented unlabeled target set $\mathcal{U}''$. The number of training epochs T. The hyper-parameters α, β, τ and λ..

2 Initialize: The sets of parameters of encoder network, source scattering classifier and target clustering classifier: $\Theta_{\mathcal{G}}^0$, $\Theta_{\mathcal{F}_1}^0$ and $\Theta_{\mathcal{F}_2}^0$.

3 for $t = 1 \sim T$ **do**

4   *# Loss Computation*

5   Get the cross-domain features from encoder network with Eq. 1;

6   Compute the MMD loss based on the kernel function of Eq. 7;

7   Compute the cross-entropy loss of all the labeled data on both classifiers with Eq. 2.

8   Compute the source entropy and target entropy based on Eq. 6 and Eq. 5.

9   Compute the self-training loss based on Eq. 8.

10   *# Parameters Updating*

11   Update $\Theta_{\mathcal{F}_1}^{t-1}$ to $\Theta_{\mathcal{F}_1}^t$ by Eq. 9.

12   Update $\Theta_{\mathcal{F}_2}^{t-1}$ to $\Theta_{\mathcal{F}_2}^t$ by Eq. 10.

13   Update $\Theta_{\mathcal{G}}^{t-1}$ to $\Theta_{\mathcal{G}}^t$ by Eq. 11.

14 end

15 Output: $\Theta_{\mathcal{G}}^* \leftarrow \Theta_{\mathcal{G}}^T$, $\Theta_{\mathcal{F}_1}^* \leftarrow \Theta_{\mathcal{F}_1}^T$ and $\Theta_{\mathcal{F}_2}^* \leftarrow \Theta_{\mathcal{F}_2}^T$.

---

loss (Eq. 8) is also necessary for model training. Therefore, the overall training objectiveness for the two classifiers can be written as:

$$\Theta_{\mathcal{F}_1}^* = \underset{\Theta_{\mathcal{F}_1}}{\arg\min}(1-\alpha)\mathcal{L}_{tar} + \alpha\mathcal{L}_{src} + \beta H_{src} + \mathcal{L}_{st}, \quad (9)$$

$$\Theta_{\mathcal{F}_2}^* = \underset{\Theta_{\mathcal{F}_2}}{\arg\min}(1-\alpha)\mathcal{L}_{src} + \alpha\mathcal{L}_{tar} - \lambda H_{tar} + \mathcal{L}_{st}, \quad (10)$$

where λ and β are loss weights to balance the influence of conditional entropy loss and supervision loss. The minimization of conditional entropy loss drives the features away from the decision boundary to implicitly cluster features. In turn, maximizing the conditional entropy loss is the reverse operation which leads to the features scattering. The self-training loss helps reunite the opposites to learn the consistent predictions on the augmented data, which serves as the views-consistent regularization for robust decisions.

To progressively disperse source features and cluster target features, the feature encoder network $\mathcal{G}(\cdot)$ is trained by the reversal of conditional entropy loss, where we attempt to minimize the target entropy loss and maximize the source entropy loss. Apart from the task loss and self-training loss inherited from the classifiers, we also take the MMD loss for the explicit alignment (Eq. 7). The overall loss for training the feature encoder network is formalized as below:

$$\Theta_{\mathcal{G}}^* = \underset{\Theta_{\mathcal{G}}}{\arg\min}\mathcal{L}_{src} + \mathcal{L}_{tar} - \beta H_{src} + \lambda H_{tar} + \mathcal{L}_{st} + \mathcal{L}_{align}. \quad (11)$$

The whole framework is trained in an end-to-end manner with the help of gradient reversal [51] for adversarial training and would continue to loop until reaching the ending epochs. The ablation studies on different components can be referred to Sec. IV-D.

F. Theoretical Insights

Next, we provide some theoretical insights behind our proposed framework, especially for the design of adaptive structure learning. According to the conclusion in [30], the risk for the target domain can be bounded by the risk for the source domain and the domain divergence as follows,

$$\forall h \in H, \quad \mathcal{R}_{\mathcal{T}}(h) \leq \mathcal{R}_{\mathcal{S}}(h) + \frac{1}{2}d_{\mathcal{H}}(\mathcal{S}, \mathcal{T}) + \delta, \quad (12)$$

where $\mathcal{S}$, $\mathcal{T}$ represents the source domain and target domain, respectively, $\mathcal{R}_{\mathcal{T}}(h)$, $\mathcal{R}_{\mathcal{S}}(h)$ represents the expected risk on domain $\mathcal{T}$ and $\mathcal{S}$, respectively, $d_{\mathcal{H}}(p, q)$ is the $\mathcal{H}$-distance between distribution p and q, δ is a constant which is related to the error of a perfect hypothesis on both domains. In this case, if we can decrease the value of $d_{\mathcal{H}}(\mathcal{S}, \mathcal{T})$ via training the domain classifiers and the feature generators, we can decrease the risk on the target domain. Now, we will show in details how we achieved this. As $d_{\mathcal{H}}(\mathcal{S}, \mathcal{T})$ is defined as:

$$d_{\mathcal{H}}(\mathcal{S}, \mathcal{T}) = 2 \sup_{h \in \mathcal{H}} \left| \Pr_{\boldsymbol{f}^s \sim p}[h(\boldsymbol{f}^s) = 1] - \Pr_{\boldsymbol{f}^t \sim q}[h(\boldsymbol{f}^t) = 1] \right|, \quad (13)$$

where $\boldsymbol{f}^s$ and $\boldsymbol{f}^t$ represents the features extracted from domain $\mathcal{S}$ and domain $\mathcal{T}$ respectively. In our framework, we use the entropy function $H(\cdot)$ to train the parameters of $\mathcal{F}_1(\cdot)$, $\mathcal{F}_2(\cdot)$ and $\mathcal{G}(\cdot)$. Though the entropy function is not the usual classification loss, our framework can be seen as minimizing divergence (13) via adversarial training strategy on target domain and source domain. Suppose h is a binary classifier as follows:

$$h(f) = \begin{cases} 1 & \text{if} \quad H(\mathcal{F}_i(\boldsymbol{f})) \geq \gamma \\ 0 & \text{otherwise} \end{cases}, \quad (14)$$

where $i = 1, 2$, γ is a threshold. To facilitate the analysis, we assume the output of the classifiers $\mathcal{F}_1(\cdot)$ and $\mathcal{F}_2(\cdot)$ are the conditional probabilities. Then (13) can be rewritten as:

$$\begin{aligned} d_{\mathcal{H}}(\mathcal{S}, \mathcal{T}) \approx \; & 2 \sup_{\mathcal{F}_1, \mathcal{F}_2} \left| \Pr_{\boldsymbol{f}^s \sim p}[H(\mathcal{F}_1(\boldsymbol{f}^s)) \geq \gamma] \right. \\ & \left. - \Pr_{\boldsymbol{f}^u \sim q}[H(\mathcal{F}_2(\boldsymbol{f}^u)) \geq \gamma] \right| \\ = \; & 2 \sup_{\mathcal{F}_1, \mathcal{F}_2} \left(\Pr_{\boldsymbol{f}^u \sim q}[H(\mathcal{F}_2(\boldsymbol{f}^u)) \geq \gamma] \right. \\ & \left. - \Pr_{\boldsymbol{f}^s \sim p}[H(\mathcal{F}_1(\boldsymbol{f}^s)) \geq \gamma] \right), \quad (15) \end{aligned}$$

where the approximate equality is because we are using unlabeled data here. As the number of the unlabeled samples on target domain is much larger than the labeled ones, we can use the probability on unlabeled samples to replace the probability on whole target domain. The equality above is due to the assumption that $\Pr_{\boldsymbol{f}^u \sim q}[H(\mathcal{F}_2(\boldsymbol{f}^u)) \geq \gamma] \geq$

$\Pr_{\boldsymbol{f}^s \sim p}[H(\mathcal{F}_1(\boldsymbol{f}^s)) \geq \gamma]$. This can be easily achieved as the labels of all the samples in the source domain are available, which implies that we can make the corresponding entropy be 0. Next, we replace $\sup$ with $\max$ in (15), then we can get

$$
\begin{aligned}
d_{\mathcal{H}}(\mathcal{S}, \mathcal{T}) &\approx 2 \max_{\mathcal{F}_1, \mathcal{F}_2} \left(\Pr_{\boldsymbol{f}^u \sim q}[H(\mathcal{F}_2(\boldsymbol{f}^u)) \geq \gamma] \right. \\
&\quad \left. - \Pr_{\boldsymbol{f}^s \sim p}[H(\mathcal{F}_1(\boldsymbol{f}^s)) \geq \gamma] \right) \\
&= 2 \min_{\mathcal{F}_1, \mathcal{F}_2} \left(- \Pr_{\boldsymbol{f}^u \sim q}[H(\mathcal{F}_2(\boldsymbol{f}^u)) \geq \gamma] \right. \\
&\quad \left. + \Pr_{\boldsymbol{f}^s \sim p}[H(\mathcal{F}_1(\boldsymbol{f}^s)) \geq \gamma] \right) \\
&= -2 \min_{\mathcal{F}_2} \Pr_{\boldsymbol{f}^u \sim q}[H(\mathcal{F}_2(\boldsymbol{f}^u)) \geq \gamma] \\
&\quad + 2 \min_{\mathcal{F}_1} \Pr_{\boldsymbol{f}^s \sim p}[H(\mathcal{F}_1(\boldsymbol{f}^s)) \geq \gamma],
\end{aligned}
$$

which matches with the update rules (9), (10) in our framework. Intuitively, the training of classifers $\mathcal{F}_1$ and $\mathcal{F}_2$ can be seen as approximating the divergence $d_{\mathcal{H}}(\mathcal{S}, \mathcal{T})$, which defines an upper bound of the target domain risk. Moreover, we aim to minimize the divergence with respect to the features $\boldsymbol{f}^s$ and $\boldsymbol{f}^u$ to bound the risk on $\mathcal{T}$:

$$
\min_{\boldsymbol{f}^s, \boldsymbol{f}^u} \left\{ \max_{\mathcal{F}_1, \mathcal{F}_2} \left(2 \Pr_{\boldsymbol{f}^u \sim q}[H(\mathcal{F}_2(\boldsymbol{f}^u)) \geq \gamma] \right. \right.
$$
$$
\left. \left. - 2 \Pr_{\boldsymbol{f}^s \sim p}[H(\mathcal{F}_1(\boldsymbol{f}^s)) \geq \gamma] \right) \right\}, \tag{16}
$$

where finding optimal $\boldsymbol{f}^s$ and $\boldsymbol{f}^u$ is equivalent to finding the optimal feature extractor $\mathcal{G}(\cdot)$, which corresponds to the update of (11) in our model. To summarize, our maximum training process with classifier $\mathcal{F}_1$ and $\mathcal{F}_2$ can be seen as measuring the domain divergence, while minimum training process with generator $\mathcal{G}$ can be seen as minimizing the divergence. In this case, we can effectively reduce the risk on the target domain $\mathcal{T}$. The discussions of MMD for DA can be referred to in many previous works [52], [53], and we will leave the theoretical analysis of ST loss in future works.

IV. EXPERIMENTS

A. Experiments Setup.

Implementation. For the evaluation of different architectures, we choose both the VGG16 [2] and ResNet34 [1] as the backbones of the feature encoder network $\mathcal{G}(\cdot)$. The two classifiers $\mathcal{F}_1(\cdot)$ and $\mathcal{F}_2(\cdot)$ utilize the same architecture Feedforward Network (FFN) with two layers randomly initialed. For model optimization, we take the momentum Stochastic Gradient Descent (SGDM) as the optimizer on PyTorch [54]. The learning rate is assigned as 0.01 with the momentum as 0.9 and weight decay as 0.0005. The loss weights α, β and λ are assigned as 0.75, 0.1, and 0.1 respectively. The strong data augmentation strategy consists of a sequence of two operations randomly picked from the ten following [44].

Datasets. We select the latest DA benchmarks including DomainNet [22] and Office-home [21], for a thorough evaluation of our proposed approach. To fairly compare with the baseline methods, we use the same protocol as [6], [15]. The Office-home benchmark is slightly smaller than DomainNet, which consists of 4 domains including *Real* (**R**), *Clipart* (**C**), *Art* (**A**) and *Product* (**P**) with 65 classes shared over all the domains. We have applied all the 12 adaptation scenarios to fairly compare the proposed method with previous approaches. On the DomainNet benchmark, there are 4 domains including *Real* (**R**), *Painting* (**P**), *Clipart* (**C**) and *Sketch* (**S**) with 126 classes for evaluation. And there are 7 adaptation scenarios with different scales/types of domain gap to overcome.

Baselines. S+T denotes the model trained by labeled source and target data only. **ENT** [57] is an SSL method that minimizes the conditional entropy on unlabeled target samples without explicit alignment. **DANN** [51] adversarially applies a discriminator to confuse the cross-domain features. **CDAN** [56] employs entropy minimization to control the uncertainty of predictions for transferability. **ADR** [55] is a GAN-based method designed to learn domain-invariant and discriminative features. **MME** [6] is an SSDA method where the conditional entropy of the unlabeled target samples is adversarially minimized. **BNM** [58] is a UDA/SSDA method attempting to maximize nuclear-norm to learn transferable features. **UODA** [15] considered the source domain regularization and extended the MME based on the opposite structure learning. **CDAC** [43] is a recent paper that utilizes both adaptive adversarial clustering and unsupervised data augmentation for SSDA.

Evaluation. Following the protocol of [6], all the baseline and proposed models will use the transductive setting where the unlabeled target samples are seen as the final target for evaluation, given the source and target labeled data and unlabeled target samples for training. All the methods will be evaluated under the one-shot and three-shot settings where there are one or three labeled samples per class in the target domain. The top-1 accuracy over all the samples has been reported as the evaluation matrix.

Data Augmentation. We follow the data augmentation strategies of RandAugment [59] where two of ten augments are randomly picked for strong data transformation. The total set of the data augmentation strategies consist of $AutoContrast$, $Brightness$, $Color$, $Contrast$, $Equalize$, $Identity$, $Posterize$, $Rotate$, $Sharpness$, $ShearX$, $ShearY$, $Solarize$, $TranslateX$ and $TranslateY$. The weak augmentation is a common practice which involves $RandomHorizontalFlip$ and $RandomCrop$ from 256×256 to 224×224 with paddings as 28.

B. Results on DomainNet.

Table II summarizes the quantitative comparison of our proposed approach with baseline methods on the DomainNet dataset. We can see that ours have outperformed all the previous methods in the average on both 1-shot and 3-shot settings. Compared with the direct baseline UODA, the current method shows a huge superiority in most of the adaptation scenarios. Especially on the ResNet-34, there is a 5% improvement for both 1-shot and 3-shot settings. According to our survey, the most challenging domain gap to overcome is Real to Sketch,

TABLE II
QUANTITATIVE RESULTS (%) ON THE DOMAINNET [22] UNDER RESNET-34 [1].

Methods	R→C		R→P		P→C		C→S		S→P		R→S		P→R		Avg	
	1_{shot}	3_{shot}	1_{shot}	3_{shot}	1_{shot}	3_{shot}	1_{shot}	3_{shot}	1_{shot}	3_{shot}	1_{shot}	3-shot	1_{shot}	3_{shot}	1_{shot}	3_{shot}
							ResNet-34									
S+T	55.6	60.0	60.6	62.2	56.8	59.4	50.8	55.0	56.0	59.5	46.3	50.1	71.8	73.9	56.9	60.0
DANN [51]	58.2	59.8	61.4	62.8	56.3	59.6	52.8	55.4	57.4	59.9	52.2	54.9	70.3	72.2	58.4	60.7
ADR [55]	57.1	60.7	61.3	61.9	57.0	60.7	51.0	54.4	56.0	59.9	49.0	51.1	72.0	74.2	57.6	60.4
CDAN [56]	65.0	69.0	64.9	67.3	63.7	68.4	53.1	57.8	63.4	65.3	54.5	59.0	73.2	78.5	62.5	66.5
ENT [57]	65.2	71.0	65.9	69.2	65.4	71.1	54.6	60.0	59.7	62.1	52.1	61.1	75.0	78.6	62.6	67.6
MME [6]	70.0	72.2	67.7	69.7	69.0	71.7	56.3	61.8	64.8	66.8	61.0	61.9	76.1	78.5	66.4	68.9
BNM [58]	66.8	68.7	67.3	68.6	66.7	69.3	58.2	58.3	63.9	65.6	59.1	60.5	76.4	78.1	65.5	67.0
UODA [15]	72.7	75.4	70.3	71.5	69.8	73.2	60.5	64.1	66.4	69.4	62.7	64.2	77.3	80.8	68.5	71.2
CDAC [43]	**77.4**	**79.6**	74.2	75.1	**75.5**	**79.3**	**67.9**	69.9	71.0	73.4	69.2	**72.5**	**80.4**	81.9	73.6	76.0
Ours	77.0	79.4	**75.4**	**76.7**	75.5	78.3	66.5	**70.2**	**72.1**	**74.2**	70.9	72.1	79.7	**82.3**	**73.9**	**76.2**
							VGG-16									
S+T	49.0	52.3	55.4	56.7	47.7	51.0	43.9	48.5	50.8	55.1	37.9	45.0	69.0	71.7	50.5	54.3
DANN [51]	43.9	56.8	42.0	57.5	37.3	49.2	46.7	48.2	51.9	55.6	30.2	45.6	65.8	70.1	45.4	54.7
ADR [55]	48.3	50.2	54.6	56.1	47.3	51.5	44.0	49.0	50.7	53.5	38.6	44.7	67.6	70.9	50.2	53.7
CDAN [56]	57.8	58.1	57.8	59.1	51.0	57.4	42.5	47.2	51.2	54.5	42.6	49.3	71.7	74.6	53.5	57.2
ENT [57]	39.6	50.3	43.9	54.6	26.4	47.4	27.0	41.9	29.1	51.0	19.3	39.7	68.2	72.5	36.2	51.1
MME [6]	60.6	64.1	63.3	63.5	57.0	60.7	50.9	55.4	60.5	60.9	50.2	54.8	72.2	75.3	59.2	62.1
BNM [58]	57.3	60.3	59.1	60.3	54.6	59.1	48.6	53.3	56.1	58.4	44.1	50.2	71.1	73.8	55.9	59.3
UODA [15]	62.2	66.2	63.6	65.7	59.4	65.1	52.3	57.6	59.2	63.2	49.6	55.9	**74.1**	76.3	60.1	64.3
Ours	**64.9**	**67.3**	**66.8**	**68.2**	**61.6**	**67.2**	**55.3**	**61.1**	**63.4**	**65.5**	**52.8**	**59.4**	73.6	**76.6**	**62.6**	**66.5**

where ours have beaten the latest paper on the 1-shot setting. Since CDAC has not tested its method on the VGG-16, the strongest baseline for comparison is UODA in this case. The improvements on 1-shot SSDA are slightly inferior to those of the 3-shot on both backbones in the comparison of the performance of both tasks. This means that our methods need more supervision to better exploit its potential since more labeled target examples are helpful to indicate the conditional distributions of the target features. Our proposed approach is stably higher than UODA with the backbone of VGG-16 except for the Painting to Real scenario, where the proposed self-training and explicit brings the negative transfer. Such instability can be seen as a weakness of our method, which, however, is robust in most of the cases.

C. Results on Office-Home.

Table III shows the quantitative results and comparison on the benchmark Office-home. We can notice that the proposed approach has outperformed the direct baseline, i.e., UODA, on most adaptation scenarios. However, on the 1-shot setting, ours is largely beaten by the UODA on the case of adapting from Clipart to Art. This likely happens when the domain gap is large with the diversified data that self-training will cause more substantial confirmation bias towards the labeled target set. Moreover, on the scenarios as Art to Real and Clipart to Real, the UODA and ours have similar results indicating that the proposed self-training and explicit alignment cause no extra boost or damage for the transfer. On the 3-shot setting, ours is also inferior to UODA on the Art to Real scenario, which shows the similar phenomenon as the 1-shot setting. Compared with the results on DomainNet, the improvements on Office-home are significantly weaker in both 1-shot and 3-shot settings despite the fact that the Office-home dataset seems more straightforward to solve due to fewer classes. This may be explained as the more effective self-training in

DomainNet due to the larger quantity of unlabeled data for the regularizing more smoothing decision boundaries.

D. Ablation Study.

Since the proposed system relies on multiple components for fulfilling the final objective, it is necessary to fully investigate their effects. As shown in the Table IV, there is a detailed ablation study for the analysis of each component in the four transfer scenarios, including Real to Sketch and Real to Clipart on both 1 or 3-shot settings. Six modules are chosen for analysis including the model with one classifier (1-C) or two classifiers (2-C) composed of target-clustering classifier and source-scattering classifier, the target entropy (H_{tar}), and the combination of the source and target entropy ($H_{tar}+H_{src}$), and the explicit alignment loss (MMD) and self-training (ST).

In the top three rows, which compare the results of one classifier structure, the explicit alignment and self-training bring significant improvements on all the adaptation cases. Furthermore, the contradictory structure learning on one classifier will, however, cause a performance drop, indicating the necessity of two classifiers. The bottom five rows summarize the results with two classifiers as utilized in our method. The effectiveness of contradictory structure learning can be verified by comparing row 4 and row 5, with or without the source entropy loss. Over all the scenarios, the latter is obviously inferior to the former with the contradictory structure learning. Finally, compared with the MMD only (row 6) and ST (row7) only cases, it is easy to notice that the self-training contributes more to the performance boost. And the joint applying of MMD and ST will achieve the best as expected.

E. Analyses.

Convergence Analysis. The Fig. 3 (a) shows the accuracy along with the training of different methods, including both baselines and ours for the 1-shot R→S problem. It is clear

TABLE III
QUANTITATIVE RESULTS (%) ON OFFICE-HOME [21] UNDER THE BACKBONE OF VGG-16 [2].

| Methods	R→C	R→P	R→A	P→R	P→C	P→A	A→P	A→C	A→R	C→R	C→A	C→P	Avg
					ONE-SHOT								
S+T	39.5	75.3	61.2	71.6	37.0	52.0	63.6	37.5	69.5	64.5	51.4	65.9	57.4
DANN [51]	**52.0**	75.7	62.7	72.7	45.9	51.3	64.3	44.4	68.9	64.2	52.3	65.3	60.0
ADR [55]	39.7	76.2	60.2	71.8	37.2	51.4	63.9	39.0	68.7	64.8	50.0	65.2	57.4
CDAN [56]	43.3	75.7	60.9	69.6	37.4	44.5	67.7	39.8	64.8	58.7	41.6	66.2	55.8
ENT [57]	23.7	77.5	64.0	74.6	21.3	44.6	66.0	22.4	70.6	62.1	25.1	67.7	51.6
MME [6]	49.1	78.7	65.1	74.4	46.2	56.0	68.6	**45.8**	72.2	68.0	57.5	71.3	62.7
BNM [58]	51.0	79.5	62.8	72.3	44.0	51.8	67.1	45.7	68.4	65.3	52.7	69.1	60.8
UODA [15]	49.6	79.8	66.1	75.4	45.5	58.8	**72.5**	43.3	**73.3**	70.5	**59.3**	72.1	63.9
Ours	51.6	**80.9**	**66.9**	**75.9**	**49.7**	**60.5**	71.0	44.9	73.2	**70.6**	58.7	**72.8**	**64.7**
					THREE-SHOT								
S+T	49.6	78.6	63.6	72.7	47.2	55.9	69.4	47.5	73.4	69.7	56.2	70.4	62.9
DANN [51]	56.1	77.9	63.7	73.6	52.4	56.3	69.5	50.0	72.3	68.7	56.4	69.8	63.9
ADR [55]	49.0	78.1	62.8	73.6	47.8	55.8	69.9	49.3	73.3	69.3	56.3	71.4	63.0
CDAN [56]	50.2	80.9	62.1	70.8	45.1	50.3	74.7	46.0	71.4	65.9	52.9	71.2	61.8
ENT [57]	48.3	81.6	65.5	76.6	46.8	56.9	73.0	44.8	75.3	72.9	59.1	77.0	64.8
MME [6]	56.9	82.9	65.7	76.7	53.6	59.2	75.7	54.9	75.3	72.9	61.1	76.3	67.6
BNM [58]	56.0	81.2	64.7	73.8	52.7	55.8	72.1	51.7	73.2	70.3	57.0	73.3	65.2
UODA [15]	57.6	83.6	67.5	77.7	54.9	61.0	77.7	55.4	**76.7**	73.8	61.9	78.4	68.9
Ours	**59.3**	**83.6**	**68.0**	**78.3**	**56.8**	**61.8**	**78.6**	**55.7**	75.3	**74.0**	**63.3**	**78.9**	**69.5**
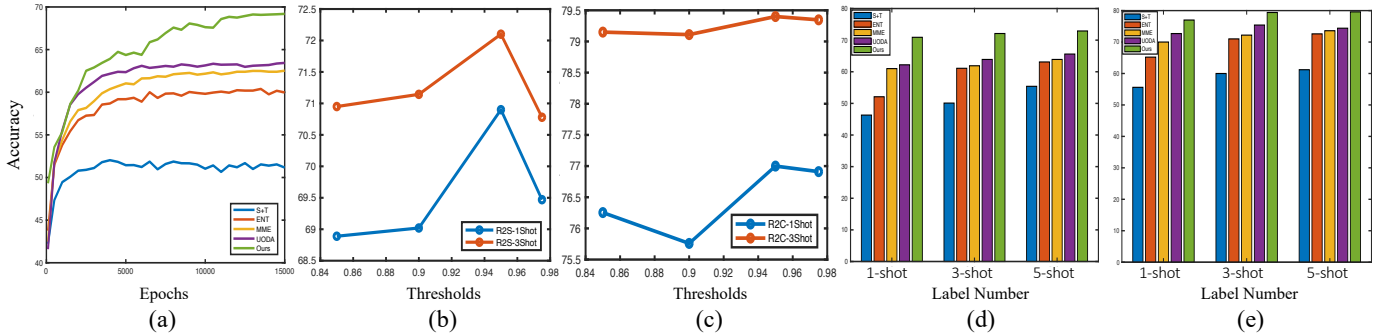
Fig. 3. (a): Convergence analysis of four baselines and ours on R→S. (b) - (c): The performance of our method in different values of threshold τ (with $\tau=0.85$, $\tau=0.9$, $\tau=0.95$, $\tau=0.975$) on R→S, i.e., sub-figure (b) and R→C, i.e., sub-figure (c). (d) - (e): Histogram of quantitative comparisons under 1-shot, 3-shot, and 5-shot settings on R→S, i.e., sub-figure (d) and R→C, i.e., sub-figure (e).

to notice that the accuracy of the proposed method has continuously improving whereas the others are converged in the early stages. The performance gap keeps growing through the iterations, which is mostly contributed from the self-training and explicit alignment. More high-confident pseudo labels are generated due to the increase of softmax scores during training.

Sensitivity of Threshold τ. As shown in Fig. 3 (b) - (c), we have analyzed the sensitivity of threshold τ on both Real to Sketch and Real to Clipart settings with the options as $\{\tau=0.85, \tau=0.9, \tau=0.95, \tau=0.975\}$. We can clearly observe that the proposed approach will achieve the best when setting the threshold as 0.95, which is consistent to the SSL methods [44].

Sensitivity of labeled samples. In the sub-figure (d) - (e) of Fig. 3, we have shown the histogram for comparisons with different labeled samples. We can draw similar conclusions from both sub-figures that our proposed method has achieved the top results despite the number of labeled data. However, the improvements from 3-shot to 5-shot are slightly narrow than those of 1-shot to 3-shot. Such a phenomenon indicates the diminishing gains from more labeling, which will eventually

converge to the fully supervised one.

F. Feature Visualization.

As shown in Fig. 4, we take the t-SNE [60] to reduce the dimensions of raw deep features into two for visualization. Compared with the feature map of S+T, the other three adaptive methods have more dense and separate features, which obviously are better aligned. Although the difference between the MME, UODA, and ours is not significant, we can still notice that the clusters are more uniformly distributed and more separable, leading to more accurate transfer.

G. Split Analysis.

To verify the robustness of the proposed method over the baselines against the data selections/bias, Table V shows the multiple results on three different splits of labeled/unlabeled target domain data. Ours has achieved the best on all the splits with a relatively low variance to show good stability against the bias of selected labeled data.
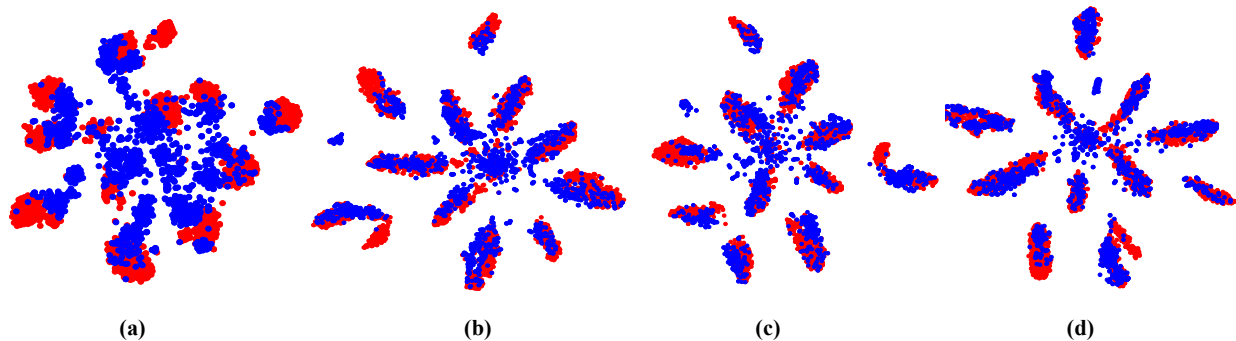
Fig. 4. The t-SNE [60] visualization results of shared top ten-class features in the 3-shot R→S problem obtained by: (a) S+T, (b) MME, (c) UODA, (d) Ours. The red and blue spots indicate the source and target feature points, respectively.

TABLE IV
QUANTITATIVE RESULTS (%) OF ABLATION STUDY BY THE BACKBONE RESNET34 [1].

COMPONENTS						REAL ⟶ SKETCH		REAL ⟶ CLIPART	
1-C	2-C	H_{tar}	$H_{tar}+H_{src}$	MMD	ST	ONE-SHOT	THREE-SHOT	ONE-SHOT	THREE-SHOT
✓		✓				61.03	61.93	70.04	72.19
✓			✓			60.40	61.16	69.24	71.47
✓		✓		✓	✓	**69.39**	**71.03**	**73.94**	**77.88**
	✓	✓				61.11	62.81	70.53	72.42
	✓		✓			62.17	63.90	71.57	74.02
	✓		✓		✓	69.16	71.46	75.68	78.57
	✓		✓	✓		65.58	67.18	74.11	75.34
	✓		✓	✓	✓	**70.9**1	**72.12**	**76.98**	**79.40**

TABLE V
SPLIT ANALYSIS OF REAL TO SKETCH ON THREE SHOT SETTING

Methods	Split-1	Split-2	Split-3	Avg $\pm$ Var
S+T	50.1	52.7	51.5	51.4 $\pm$ 1.7
MME [6]	61.9	61.2	63.8	62.3 $\pm$ 1.8
UODA [15]	64.2	64.1	63.0	63.8 $\pm$ 0.4
Ours	**72.1**	**72.1**	**70.9**	**71.7 $\pm$ 0.5**

TABLE VI
AVERAGE QUANTITATIVE RESULTS ON DOMAINNET WITH DIFFERENT
CLASSIFIERS

Backbones	Settings	Classifier-F1	Classifier-F2	Ensemble
ResNet-34	3-shot	76.2 (0.0)	76.1 (-0.1)	76.2
ResNet-34	1-shot	73.9 (0.0)	73.9 (0.0)	73.9
VGG-16	3-shot	66.4 (-0.1)	66.5 (0.0)	66.5
VGG-16	1-shot	62.6 (0.0)	62.5 (-0.1)	62.6

H. Classifier Analysis

We have recorded the performance of our model with one classifier for inference. The results on DomainNet and Office-home are summarized in Tab. VI. From the table, we can see that the performance of the classifiers ensemble is quite similar to those of one classifier with the variance less or equal to 0.1. Such similarity is caused by the self-training loss, i.e., L_{st}, since both classifiers have the same pseudo labels. At the beginning of model training, there are only a few pseudo labels due to the high thresholding where the two classifiers are more dominated by the opposite entropy loss with different weights. With the model training and increased score of target predictions, the self-training loss has become dominating and made the two classifiers towards matching.

I. Complexity Analysis

In our framework, we need three types of data: source domain data, labeled target domain data, and unlabeled target domain data. Our framework is specifically designed for the situation where there exists very limited number of labeled target domain data. This means the performance of our method

will not be affected when the number of labeled target data varies. From the experimental results, it can also be seen that our method works well even in three-shot or one-shot cases.

For the source domain data and unlabeled target domain data, the performance of our framework can be affected if the numbers of the available samples are very limited. Our method is a semi-supervised learning method, which means we need the information of unlabeled target data to do the training. From the theoretical perspective, our framework is trying to minimize the domain divergence $d_H(S, T)$ (i.e., Eq. (16)) between the distribution of source data and target data, where the distribution of target data is estimated from the available unlabeled target data. If numbers of samples of source data and target domain data are too small, the distributions cannot be accurately sampled, which may affect the performance. However, from empirical perspective, our method still works well when there are limited number of available source domain data and unlabeled target domain data.

V. CONCLUSION

Semi-supervised domain adaptation (SSDA) is crucial in transfer learning which has a significant performance boost compared to the unsupervised domain adaptation with only marginal annotations. This paper attempts to address the SSDA by learning the adaptive structures for feature transfer. The previous implicit feature alignment for learning well-clustered target features and scattered source features may result in the categorical mismatch across domains. To solve this, we have applied explicit alignment by minimizing the distance (i.e., MMD loss) between pairs of cross-domain features in the reproducing kernel Hilbert space. It helped to project the contradictory structures into a shared view for the robust final decision. Moreover, pseudo-labeling is employed to regularize the decision boundary towards smoothness in a self-training manner. Extensive experiments on the multiple benchmarks, including Office-home [21] and DomainNet [22], have shown the advantages of our proposed approach over our direct baseline [15] and other latest methods. Our work may further inspire the community to investigate the better adaptive structures for transfer learning.

REFERENCES

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[2] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[3] J. Zhu, Y. Xia, L. Wu, D. He, T. Qin, W. Zhou, H. Li, and T.-Y. Liu, "Incorporating bert into neural machine translation," *arXiv preprint arXiv:2002.06823*, 2020.

[4] G. Zheng, F. Zhang, Z. Zheng, Y. Xiang, N. J. Yuan, X. Xie, and Z. Li, "Drn: A deep reinforcement learning framework for news recommendation," in *World Wide Web Conference (WWW)*, 2018.

[5] 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.

[6] K. Saito, D. Kim, S. Sclaroff, T. Darrell, and K. Saenko, "Semi-supervised domain adaptation via minimax entropy," *arXiv preprint arXiv:1904.06487*, 2019.

[7] L. Cheng, R. Guo, K. S. Candan, and H. Liu, "Representation learning for imbalanced cross-domain classification," in *SIAM International Conference on Data Mining (SDM)*, 2020.

[8] Y. Zhang, Y. Zhang, Y. Wei, K. Bai, Y. Song, and Q. Yang, "Fisher deep domain adaptation," in *SIAM International Conference on Data Mining (SDM)*, 2020.

[9] B. Sun and K. Saenko, "Subspace distribution alignment for unsupervised domain adaptation." in *British Machine Vision Conference (BMVC)*, 2015.

[10] B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[11] R. Gopalan, R. Li, and R. Chellappa, "Domain adaptation for object recognition: An unsupervised approach," in *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2011.

[12] M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu, "Transfer feature learning with joint distribution adaptation," in *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2013.

[13] R. Shu, H. H. Bui, H. Narui, and S. Ermon, "A dirt-t approach to unsupervised domain adaptation," *arXiv preprint arXiv:1802.08735*, 2018.

[14] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada, "Maximum classifier discrepancy for unsupervised domain adaptation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[15] C. Qin, L. Wang, Q. Ma, Y. Yin, H. Wang, and Y. Fu, "Contradictory structure learning for semi-supervised domain adaptation," in *SIAM International Conference on Data Mining (SDM)*, 2021.

[16] M. Thota and G. Leontidis, "Contrastive domain adaptation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 2209–2218.

[17] A. Singh, "Clda: Contrastive learning for semi-supervised domain adaptation," *Advances in Neural Information Processing Systems*, vol. 34, pp. 5089–5101, 2021.

[18] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," *arXiv preprint arXiv:1409.7495*, 2014.

[19] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[20] 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 (NeurIPS)*, 2014.

[21] H. Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan, "Deep hashing network for unsupervised domain adaptation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[22] X. Peng, Q. Bai, X. Xia, Z. Huang, K. Saenko, and B. Wang, "Moment matching for multi-source domain adaptation," in *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[23] Q. Ma, Y.-Y. Liu, and A. Olshevsky, "Optimal lockdown for pandemic control," *arXiv preprint arXiv:2010.12923*, 2020.

[24] Y. Cao, M. Long, and J. Wang, "Unsupervised domain adaptation with distribution matching machines," in *AAAI Conference on Artificial Intelligence (AAAI)*, 2018.

[25] M. N. A. Khan and D. R. Heisterkamp, "Adapting instance weights for unsupervised domain adaptation using quadratic mutual information and subspace learning," in *International Conference on Pattern Recognition (ICPR)*, 2016.

[26] J. Wang, W. Feng, Y. Chen, H. Yu, M. Huang, and P. S. Yu, "Visual domain adaptation with manifold embedded distribution alignment," in *ACM Multimedia (MM)*, 2018.

[27] C. Qin, H. You, L. Wang, C.-C. J. Kuo, and Y. Fu, "Pointdan: A multi-scale 3d domain adaption network for point cloud representation," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[28] C. Qin, L. Wang, Y. Zhang, and Y. Fu, "Generatively inferential co-training for unsupervised domain adaptation," in *IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, 2019.

[29] J. Dong, Y. Cong, G. Sun, Y. Liu, and X. Xu, "Cscl: Critical semantic-consistent learning for unsupervised domain adaptation," in *European Conference on Computer Vision (ECCV)*, 2020.

[30] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan, "A theory of learning from different domains," *Machine learning*, vol. 79, no. 1-2, pp. 151–175, 2010.

[31] I.-H. Jhuo, D. Liu, D. Lee, and S.-F. Chang, "Robust visual domain adaptation with low-rank reconstruction," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[32] Q. Ma and A. Olshevsky, "Adversarial crowdsourcing through robust rank-one matrix completion," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[33] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *European Conference on Computer Vision (ECCV)*, 2016.

[34] J. Hoffman, E. Tzeng, T. Park, J.-Y. Zhu, P. Isola, K. Saenko, A. A. Efros, and T. Darrell, "Cycada: Cycle-consistent adversarial domain adaptation," *arXiv preprint arXiv:1711.03213*, 2017.

[35] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2017.

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

[37] Y. Zhang, S. Miao, and R. Liao, "Structural domain adaptation with latent graph alignment," in *2018 25th IEEE International Conference on Image Processing (ICIP)*. IEEE, 2018, pp. 3753–3757.

[38] M. Pilanci and E. Vural, "Domain adaptation on graphs by learning aligned graph bases," *IEEE Transactions on Knowledge and Data Engineering*, 2020.

[39] D. Das and C. G. Lee, "Unsupervised domain adaptation using regularized hyper-graph matching," in *2018 25th IEEE International Conference on Image Processing (ICIP)*. IEEE, 2018, pp. 3758–3762.

[40] E. Tzeng, J. Hoffman, T. Darrell, and K. Saenko, "Simultaneous deep transfer across domains and tasks," in *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2015.

[41] S. Motiian, M. Piccirilli, D. A. Adjeroh, and G. Doretto, "Unified deep supervised domain adaptation and generalization," in *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2017.

[42] S. Motiian, Q. Jones, S. Iranmanesh, and G. Doretto, "Few-shot adversarial domain adaptation," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[43] J. Li, G. Li, Y. Shi, and Y. Yu, "Cross-domain adaptive clustering for semi-supervised domain adaptation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[44] K. Sohn, D. Berthelot, N. Carlini, Z. Zhang, H. Zhang, C. A. Raffel, E. D. Cubuk, A. Kurakin, and C.-L. Li, "Fixmatch: Simplifying semi-supervised learning with consistency and confidence," *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[45] C. Wei, K. Shen, Y. Chen, and T. Ma, "Theoretical analysis of self-training with deep networks on unlabeled data," *arXiv preprint arXiv:2010.03622*, 2020.

[46] X. Chen, S. Wang, M. Long, and J. Wang, "Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation," in *International conference on machine learning*. PMLR, 2019, pp. 1081–1090.

[47] H. Liu, M. Long, J. Wang, and M. Jordan, "Transferable adversarial training: A general approach to adapting deep classifiers," in *International Conference on Machine Learning*. PMLR, 2019, pp. 4013–4022.

[48] H. Zhao, R. T. Des Combes, K. Zhang, and G. Gordon, "On learning invariant representations for domain adaptation," in *International Conference on Machine Learning*. PMLR, 2019, pp. 7523–7532.

[49] Y. Chen, S. Song, S. Li, L. Yang, and C. Wu, "Domain space transfer extreme learning machine for domain adaptation," *IEEE transactions on cybernetics*, vol. 49, no. 5, pp. 1909–1922, 2018.

[50] K. M. Borgwardt, A. Gretton, M. J. Rasch, H.-P. Kriegel, B. Schölkopf, and A. J. Smola, "Integrating structured biological data by kernel maximum mean discrepancy," *Bioinformatics*, 2006.

[51] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *Journal of Machine Learning Research (JMLR)*, 2016.

[52] M. Long, Y. Cao, J. Wang, and M. I. Jordan, "Learning transferable features with deep adaptation networks," *arXiv preprint arXiv:1502.02791*, 2015.

[53] W. Wang, H. Li, Z. Ding, and Z. Wang, "Rethink maximum mean discrepancy for domain adaptation," *arXiv preprint arXiv:2007.00689*, 2020.

[54] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[55] K. Saito, Y. Ushiku, T. Harada, and K. Saenko, "Adversarial dropout regularization," *arXiv preprint arXiv:1711.01575*, 2017.

[56] M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Conditional adversarial domain adaptation," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[57] Y. Grandvalet and Y. Bengio, "Semi-supervised learning by entropy minimization," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2005.

[58] S. Cui, S. Wang, J. Zhuo, L. Li, Q. Huang, and Q. Tian, "Towards discriminability and diversity: Batch nuclear-norm maximization under label insufficient situations," *arXiv preprint arXiv:2003.12237*, 2020.

[59] E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Le, "Randaugment: Practical automated data augmentation with a reduced search space," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2020.

[60] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research (JMLR)*, 2008.
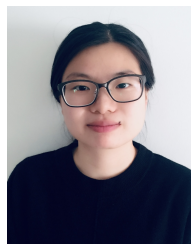
Can Qin receive the B.E. from the School of Microelectronics, Xidian University, China in 2018. He is pursuing the Ph.D. at the Department of Electrical and Computer Engineering, Northeastern University, under the supervision of Prof. Yun Raymond Fu. His research focus spans transfer learning, semi-supervised learning, and deep learning in broad. He received awards for the Best Paper Award of the Real-World Recognition from Low-Quality Images and Videos workshop at ICCV, 2019. Additionally, he has published some papers at top-tier conferences, including NeurIPS, ICLR, AAAI, ECCV, SDM, etc.

Lichen Wang received the B.Eng. degree in automation from Harbin Institute of Technology, Harbin, China in 2013. And the M.Eng. degree in control science and engineering from Xi'an Jiaotong University, Xi'an, China in 2016. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, Northeastern University, Boston, MA, USA. His research interests include computer vision, machine learning and deep learning. Mr. Wang is an AAAI Student Member. He was a recipient of the Student Travel Grant of AAAI 18. He has served as the Reviewer for CVPR, IJCAI, NN and PC member for AAAI.
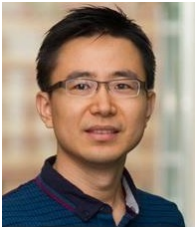

Qianqian Ma received the B.Eng. degree and the M.Eng. degree in control science and engineering from Harbin Institute of Technology, Harbin, China in 2014 and 2017, respectively. She is currently pursuing the Ph.D. degree at the Department of Electrical and Computer Engineering, Boston University, Boston, USA. Her research interests include distributed optimization, machine learning, and reinforcement learning. She has served as the reviewer for ICLR, NeurIPS, IJCAI, AAAI, etc.

Yu Yin received B.S. in Electrical and Information Engineering (2016) from Wuhan University of Technology, China, M.S. in Electrical and Computer Engineering (2018) at Northeastern University (NEU), Boston, MA, and is currently pursuing a Ph.D. in Computer Engineering at NEU under Dr. Yun Fu. Her research spans image processing (i.e.super-resolution, face generation), visual recognition (i.e.face recognition, pose estimation, face alignment, and emotion recognition), and biosignal processing. She served as organizing chair of RFIW Workshop @ FG'20 and AMFG @ CVPR21, conference PC (AAAI & IJCAI & FG), and journal reviewer (IEEE Trans. on TIP & TNNLS).

Huan Wang received the M.S. and B.E. degrees in information and communication engineering from Zhejiang University, Hangzhou, China, in 2019 and 2016, respectively. He is currently working toward the Ph.D. degree with the Department of Electrical and Computer Engineering, Northeastern University, Boston, MA, USA. His research interests include compute-efficient and data-efficient deep learning methods, mainly regarding computer vision tasks.

Yun Fu (S'07-M'08-SM'11-F'19) received the B.Eng. degree in information engineering and the M.Eng. degree in pattern recognition and intelligence systems from Xi'an Jiaotong University, China, respectively, and the M.S. degree in statistics and the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana-Champaign, respectively. He is an interdisciplinary faculty member affiliated with College of Engineering and the College of Computer and Information Science at Northeastern University since 2012. His research interests are Machine Learning, Computational Intelligence, Big Data Mining, Computer Vision, Pattern Recognition, and Cyber-Physical Systems. He has extensive publications in leading journals, books/book chapters and international conferences/workshops. He serves as associate editor, chairs, PC member and reviewer of many top journals and international conferences/workshops. He received seven Prestigious Young Investigator Awards from NAE, ONR, ARO, IEEE, INNS, UIUC, Grainger Foundation; nine Best Paper Awards from IEEE, IAPR, SPIE, SIAM; many major Industrial Research Awards from Google, Samsung, and Adobe, etc. He is currently an Associate Editor of the IEEE Transactions on Neural Networks and Leaning Systems (TNNLS). He is fellow of IAPR, OSA and SPIE, a Lifetime Senior Member of ACM, Lifetime Member of AAAI and Institute of Mathematical Statistics, member of ACM Future of Computing Academy, Global Young Academy, AAAS, INNS and Beckman Graduate Fellow during 2007-2008.