Leveraging Frozen Batch Normalization for Co-Training in Source-Free Domain Adaptation

Xianwen Deng **Yijun Wang*** **Zhi Xue***
Shanghai Jiao Tong University

Abstract

Source-free domain adaptation (SFDA) aims to adapt a source model, initially trained on a fully-labeled source domain, to an unlabeled target domain. Previous works assume that the statistics of Batch Normalization layers in the source model capture domain-specific knowledge and directly replace them with target domain-related statistics during training. However, our observations indicate that *source-like* samples in target data exhibit less deviation in the feature space of the source model when preserving the source domain-relevant statistics. In this paper, we propose co-training the source model with frozen Batch Normalization layers as part of the domain adaptation process. Specifically, we combine the source model and the target model to produce more robust pseudo-labels for *global* class clustering and to identify more precise neighbor samples for *local* neighbor clustering. Extensive experiments validate the effectiveness of our approach, showcasing its superiority over current state-of-the-art methods on three standard benchmarks. Our codes are available on https://github.com/SJTU-dxw/BN-SFDA.

1 INTRODUCTION

Leveraging extensive labeled training data, deep Convolutional Neural Networks (CNN) have exhibited remarkable advancements across diverse computer vision tasks, including classification, object detection, and semantic segmentation, among others. However, the effectiveness of CNNs is strongly dependent on the availability of sufficient labeled data and the hypothesis that the training and test data share identical distributions. When the test environment deviates from the source domain, models experience a substantial decline in performance, commonly referred to as the domain shift problem (Pan and Yang, 2009).

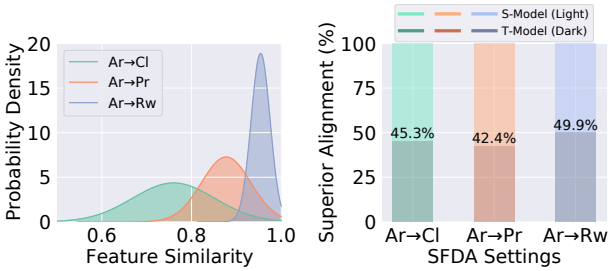

Figure 1: (Left) The similarity of the features extracted by S-Model and T-Model. (Right) Superior alignment between the features and the ground-truth class prototype in the fixed classifier.

In the pursuit to mitigate annotation costs and solve the domain shift problem, Unsupervised Domain Adaptation (UDA) has emerged as a strategic approach, transferring knowledge from a label-rich source domain to a label-scarce target domain. However, despite the proposal of numerous UDA methods (Long et al., 2018; Cui et al., 2020; Jin et al., 2020; Na et al., 2021) leveraging joint learning on source and target data, their applicability diminishes in specific real-world scenarios, particularly those involving privacy concerns (e.g., medical images, surveillance videos), where access to source-domain data is restricted. Consequently, recent advancements have redirected attention toward Source-Free Unsupervised Domain Adaptation (SFUDA) (Liang et al., 2020; Xia et al., 2021; Yang et al., 2021b; Kundu et al., 2022a,b). In this paradigm, labeled source data becomes inaccessible during the training of the target model, leaving only the pre-trained model from the source domain available. This prompts a fundamental question: What knowledge should be transferred to facilitate the learning of unlabeled target-domain data?

Some approaches (Liang et al., 2020; Ding et al., 2022) assume that the source model possesses enough knowledge of the target domain. Then they employ self-supervised

*Corresponding authors.

Proceedings of the 28th International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258.

pseudo-labels to achieve class-wise *global* alignment. Another line of works (Yang et al., 2021b, 2022) assume the source model has already developed some semantic structure, and then utilize the *local* structure of feature space (*i.e.*, neighborhood) to ensure consistency in the logit space. Despite these advances, previous works hypothesize that the statistics of Batch Normalization (Ioffe and Szegedy, 2015) layers in the source model represent the source domain-related knowledge and directly replace them with target domain-relevant statistics during the training phase. To investigate the role of statistics in BN layers in domain adaptation, we replace the statistics in the source model (denoted as S-Model) with the target domain-relevant statistics through forward propagation to obtain the target model (denoted as T-Model). We then visualize the similarity between features extracted by S-Model and T-Model and compare the alignment of features with the ground-truth class prototype in the fixed linear classifier. As shown in Figure 1, feature similarity is closely related to the degree of domain shift. For example, the domain adaptation task Ar→Cl has the lowest feature similarity and the lowest accuracy (55.4% on SHOT-IM (Liang et al., 2020)), indicating the largest domain shift; while the task Ar→Rw has the highest feature similarity and the highest accuracy (80.4% on SHOT-IM), indicating the smallest domain shift. Moreover, regardless of the degree of domain shift, nearly half of the features extracted by S-model are better aligned with the ground-truth class prototype than those extracted by T-Model, proving that the target data can be divided into *source-like* and *target-specific* samples (Zhang et al., 2022). Therefore, preserving statistics in the source model helps correct the shift of *source-like* samples in the feature space of T-Model. Furthermore, we explore combining S-Model and T-Model to jointly make predictions. As shown in Figure 2, when both models make the same prediction (*i.e.*, consistency), the accuracy improves significantly, though fewer are recognized. It can be concluded that combining the source model and the target model contributes to the generation of robust pseudo-labels for target data.

However, existing works do not fully exploit the statistics from both domains simultaneously. Most studies discard the statistics from the source domain. SVDP (Yang et al., 2024) employs the Mean-Teacher framework to update the source model, preserving the source domain statistics while discarding those from the target domain. Other studies (Kim et al., 2021; Liang et al., 2022) that retain the relevant statistics from both domains rely on the fixed-parameter source model, which hinders the full exploitation of the source domain statistics' potential. In this work, we propose a co-training framework that jointly optimizes the source model (with frozen Batch Normalization layers) and the target model, to better exploit both *global* and *local* structures. Specifically, we design a two-branch co-learning strategy where the source model and the target model are updated simultaneously. Samples predicted

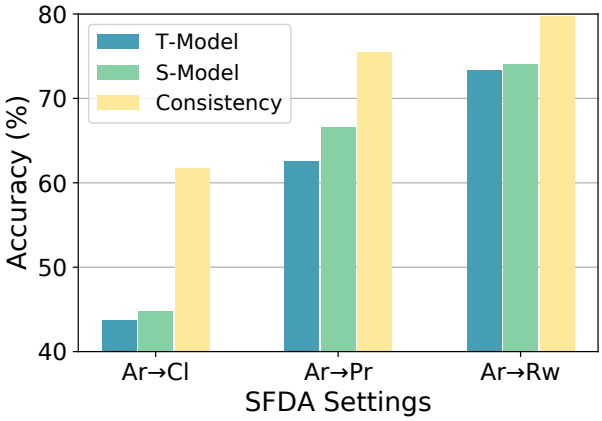

Figure 2: The joint predictions of S-Model and T-Model achieve higher accuracy when both models predict consistently.

with high confidence by both models are utilized as robust pseudo-labels for learning *global* class clustering. To explore the *local* structure, we design a new neighbor discovery strategy. The two models each calculate their own neighborhoods and the corresponding core distance (*i.e.*, the distance to the farthest neighborhood), and neighborhoods with a smaller core distance will be adopted to perform neighbor clustering. Our contributions are summarized as follows:

- We observe that *source-like* samples are less shifted in the feature space of the source model preserving the source domain-relevant statistics.
- Motivated by the above observation, we propose a co-training framework that jointly optimizes the source model (with frozen Batch Normalization layers) and the target model.
- We combine the source model and the target model to generate robust pseudo-labels for learning *global* class clustering and design a new neighbor discovery strategy to exploit *local* structure.
- We perform comprehensive experiments to assess our proposed method, and the results affirm its effectiveness. Our approach attains state-of-the-art performance across various benchmarks for Source-Free Unsupervised Domain Adaptation (SFUDA).

2 RELATED WORK

2.1 Unsupervised Domain Adaptation

In traditional unsupervised domain adaptation (UDA), models are trained concurrently on a labeled source dataset and an unlabeled target dataset to enhance task performance in the target domain (Wilson and Cook, 2020). Conventional UDA methods primarily concentrate on acquir-

ing domain-invariant representations, which can be broadly categorized into two main types. The first type aims to minimize specific distribution discrepancy metrics (Long et al., 2017; Peng et al., 2019; Tzeng et al., 2014; Yan et al., 2017), while the second type focuses on aligning feature distributions across different domains through adversarial training (Cicek and Soatto, 2019; Cui et al., 2020; Liu et al., 2019; Long et al., 2018; Saito et al., 2018; Wang et al., 2019; Zhang and Wang, 2020). Certain techniques (Na et al., 2021; Cui et al., 2020) facilitate alignment across distant domains by introducing intermediate domains, and others (Sugiyama et al., 2007; Long et al., 2017, 2016, 2015) work towards aligning feature distributions by minimizing the Maximum Mean Discrepancy (MMD). Additionally, some methods (French et al., 2018; Kim et al., 2019; Kumar et al., 2020; Liu et al., 2021; Mei et al., 2020; Pan et al., 2020) select high-confidence predictions as positive pseudo-labels to regularize model training, while others construct prototypes (Chen et al., 2019; Li et al., 2022; Pan et al., 2019; Xie et al., 2018; Zhang et al., 2021a,b; Zheng et al., 2020) or cluster centers (Deng et al., 2019; Kang et al., 2019; Liu and Wang, 2022; Tang et al., 2020) across domains for class-wise alignment. These conventional UDA methods assume access to both source and target data, making them unsuitable when the source data is unavailable for joint training.

2.2 Source-Free Unsupervised Domain Adaptation

In SFUDA, the adaptation of the source model to unseen domains occurs without utilizing the original training data. For example, SHOT (Liang et al., 2020) proposes freezing the source classifier and employing information maximization, entropy minimization, and clustering-based pseudo-labeling. BMD (Qu et al., 2022) introduces a class-balanced multi-centric dynamic prototype strategy for more accurate pseudo-labels. Other methods, such as NRC (Yang et al., 2021b), G-SFDA (Yang et al., 2021c), and AaD (Yang et al., 2022), focus on neighborhood clustering to enforce prediction consistency among local neighbors. Furthermore, alternative approaches address SFUDA from different perspectives. For instance, 3C-GAN (Li et al., 2020) uses generative models to enhance model performance on the target domain by enriching target data. A^2Net (Xia et al., 2021) introduces a new target classifier for domain alignment using adversarial training. SFDA-DE (Ding et al., 2022) estimates source feature distribution to align source and target domains. DIPE (Wang et al., 2022) explores the transferability of source model parameters. CRCo (Zhang et al., 2023) explicitly transfers class relationships which is more domain-invariant. SF(DA)^2 (Hwang et al., 2024) constructs an efficient augmentation graph and identify partitions. Recent studies (Yang et al., 2021a; Sanyal et al., 2023) have also explored SFDA approaches based on Transformers.

To exploit the statistics in the BN layers from the source domain, SVDP (Yang et al., 2024) employs the Mean-Teacher framework to update the source model, preserving the source domain statistics while discarding those from the target domain. Other studies (Kim et al., 2021; Liang et al., 2022) that retain the relevant statistics from both domains rely on the fixed-parameter source model, which hinders the full exploitation of the source domain statistics' potential. Notably, none of these methods in the existing literature fully exploit the statistics from both domains simultaneously.

3 METHOD

3.1 Preliminaries and Notations

For the scenario of source-free unsupervised domain adaptation (SFUDA), we work with a source pre-trained model and an unlabeled target domain dataset comprising N_t samples denoted as $\mathcal{D}_t = \{x_i^t\}_{i=1}^{N_t}$. The target domain shares the same C classes as the source domain (in the closed-set setting). The objective of SFUDA is to adapt the source model to the target domain without relying on source data and the model can be divided into two components: the feature extractor f and the classifier g. First, the output of the feature extractor is represented as a feature vector $\mathbf{z} = f(x) \in \mathbb{R}^d$, where d is the dimension of the feature space. Then, the output of the classifier is denoted as logits $\mathbf{h} = g(f(x)) \in \mathbb{R}^C$. Finally, the softmax function is applied to compute the probability output $\mathbf{p} = \sigma(g(f(x))) \in \mathbb{R}^C$ and we use $\hat{\mathbf{p}} = argmax(\mathbf{p})$ to represent the predicted category.

To effectively discern *source-like* samples within the target data, we choose to freeze the statistics in Batch Normalization layers within the pre-trained feature extractor. This feature extractor is denoted as the source feature extractor f_s, and while its other parameters can be updated to align the source and target domains, the statistics remain fixed. The source model comprises the source feature extractor f_s and the shared and frozen classifier g as in SHOT (Liang et al., 2020). In contrast, all parameters, including statistics, are updatable in the target feature extractor f_t. We jointly train the source model and the target model to explore both *global* and *local* structures. During inference, we directly use the target model. In the following, we propose a co-learning strategy to better achieve both *global* class clustering (Section 3.2) and *local* neighbor clustering (Section 3.3).

3.2 *Global* Class Clustering

To facilitate class-aware learning, the proper selection of samples and the acquisition of accurate pseudo-labels are crucial. Drawing inspiration from prior work in semi-supervised learning (SSL) (Sohn et al., 2020), we adopt a

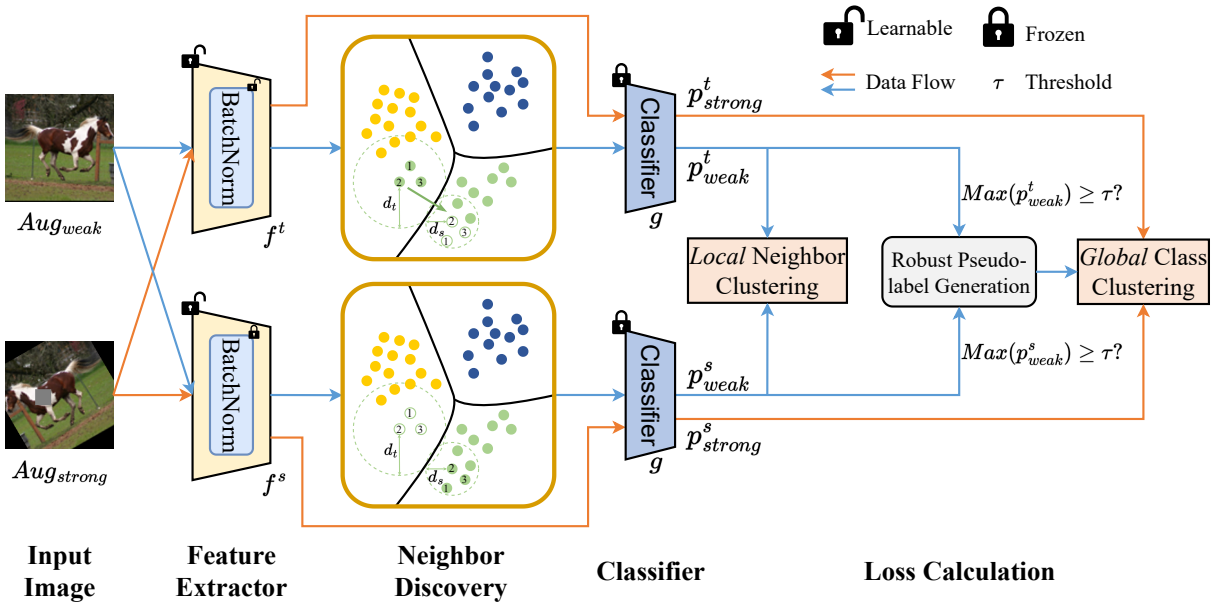

Figure 3: The framework of our proposed method. We propose a co-learning strategy to better achieve *global* class clustering by generating more robust pseudo-labels (see Section 3.2) and *local* neighbor clustering by identifying more accurate neighbor samples (see Section 3.3).

strategy of generating pseudo-labels for weakly augmented samples with high confidence.

Specifically, when provided with probabilities $\mathbf{p}_{strong}$ and $\mathbf{p}_{weak}$ of strongly and weakly augmented views, we initially compare $Max(\mathbf{p}_{weak})$ against a predefined threshold τ to identify high-confidence samples. Subsequently, $\hat{\mathbf{p}}_{weak} = argmax(\mathbf{p}_{weak})$ is employed as pseudo-labels for these selected samples. In the context of our two-branch co-learning strategy, only predictions exhibiting high confidence from both models are utilized, contributing to more robust pseudo-labeling. Formally, we employ the cross-entropy loss to establish self-supervised learning:

$$\mathcal{L}_{global} = \frac{1}{N^t}\sum_{i=1}^{N^t}\mathbb{1}(Max(\mathbf{p}^s_{weak}) \geq \tau) \times \mathbb{1}(Max(\mathbf{p}^t_{weak}) \geq \tau) \times \mathbb{1}(\hat{\mathbf{p}}^s_{weak} = \hat{\mathbf{p}}^t_{weak}) \times H(\mathbf{p}_{strong}, \hat{\mathbf{p}}_{weak}), \tag{1}$$

where H represents the cross-entropy function and the superscripts (s and t) denote the outputs of the source model and the target model. The two models employ $\mathbf{p}^s_{strong}$ and $\mathbf{p}^t_{strong}$ to compute the loss for *global* class clustering, respectively. Following CRCo (Zhang et al., 2023), we embed the source-domain class relationship into the pseudo-label loss. The combination of the source model and the target model contributes to the generation of more accurate pseudo-labels (see Figure 5 for more details).

3.3 *Local* Neighbor Clustering

NRC (Yang et al., 2021b) and AaD (Yang et al., 2022) utilize *local* neighbor clustering methods to address the SFUDA problem. For each feature $\mathbf{z}_i$, they define a neighbor set $\mathcal{C}_i$ containing K-nearest neighbors of $\mathbf{z}_i$ (measured by cosine similarity), and then encourage the predictions of nearest neighbors to be consistent. Formally, the loss for *local* neighbor clustering can be written as:

$$\mathcal{L}_{local} = -\frac{1}{N^t}\sum_{i=1}^{N^t}\sum_{j\in\mathcal{C}_i}\mathbf{p}_i^T\mathbf{p}_j. \tag{2}$$

To facilitate *local* structure learning, it is important to select correct neighbor samples (*i.e.* samples from the same class). Intuitively, areas characterized by higher density typically imply a more robust *local* structure. In our two-branch co-learning strategy, we utilize the core distance (Ankerst et al., 1999) (*i.e.*, the distance to the farthest neighbor) as a criterion for neighborhood selection. Specifically, the source model and the target model individually propose neighbor sets $\mathcal{C}^s_i$ and $\mathcal{C}^t_i$ within their respective feature spaces. Then the corresponding core distances can be defined as:

$$\begin{aligned} d^s_i &= \underset{j\in\mathcal{C}^s_i}{Max}(CosDistance(\mathbf{z}^s_i, \mathbf{z}^s_j)), \\ d^t_i &= \underset{j\in\mathcal{C}^t_i}{Max}(CosDistance(\mathbf{z}^t_i, \mathbf{z}^t_j)), \end{aligned} \tag{3}$$

where the superscripts s and t denote the source model and the target model, respectively. The neighbor set with a small core distance will be adopted to perform neighbor clustering:

$$\mathcal{C}_i = \begin{cases} \mathcal{C}^s_i, & d^s_i \leq d^t_i, \\ \mathcal{C}^t_i, & d^s_i > d^t_i. \end{cases} \tag{4}$$

While the mentioned neighbor discovery strategy contributes to identifying more accurate neighbor samples (see Figure 5), the refined neighbor sets still contain some noise due to domain shift. Inspired by the pseudo-label generation strategy, we visualize the relationship between the accuracy of K-nearest neighbor samples and the confidence of model predictions (*i.e.,* $Max(\mathbf{p})$) for both models on task Ar→Cl of Office-Home, as shown in Figure 4. The results indicate that the accuracy of K-nearest neighbor samples increases linearly with the confidence of model predictions. Therefore, we propose a novel way to re-weight the loss for *local* neighbor clustering as follows:

$$\mathcal{L}_{local} = -\frac{1}{N^t}\sum_{i=1}^{N^t}\frac{Max(\mathbf{p}_i^s + \mathbf{p}_i^t)}{2}\sum_{j\in\mathcal{C}_i}\mathbf{p}_i^T\mathbf{p}_j. \quad (5)$$

The proposed re-weighting strategy enables both models to pay more attention to high-confidence samples, thereby achieving more accurate *local* neighbor clustering. Nevertheless, if a sample and its neighbors are shifted to incorrect categories within the feature space of a specific model, performing *local* neighbor clustering within its individual feature space cannot correct this misalignment. Thus, in cases where the predicted categories from the two models diverge, we employ the probability output of the model that proposed the selected neighbor set to rectify this misalignment directly. For example, for the i-th sample, if the neighbor set proposed by the source model (*i.e.,* $\mathcal{C}_i^s$) is selected and the predicted categories of the two models differ (*i.e.,* $d_i^s \leq d_i^t$ and $\hat{\mathbf{p}}_i^s \neq \hat{\mathbf{p}}_i^t$), the target model will directly use the probability output of the source model for the selected neighbor set (*i.e.,* $\{p_j^s | j \in \mathcal{C}_i^s\}$) to calculate the loss: $\mathcal{L}_{local}^{i(t)} = \frac{Max(\mathbf{p}_i^s+\mathbf{p}_i^t)}{2}\sum_{j\in\mathcal{C}_i^s}\mathbf{p}_i^{tT}\mathbf{p}_j^s$. Formally, the *local* neighbor clustering loss on the i-th sample for the target model can be expressed as:

$$\mathcal{L}_{local}^{i(t)} = \begin{cases} W_i\sum_{j\in\mathcal{C}_i}\mathbf{p}_i^{tT}\mathbf{p}_j^t, & d_i^s > d_i^t \| \hat{\mathbf{p}}_i^s = \hat{\mathbf{p}}_i^t, \\ W_i\sum_{j\in\mathcal{C}_i}\mathbf{p}_i^{tT}\mathbf{p}_j^s, & d_i^s \leq d_i^t \& \hat{\mathbf{p}}_i^s \neq \hat{\mathbf{p}}_i^t, \end{cases} \quad (6)$$

where $W_i = \frac{Max(\mathbf{p}_i^s+\mathbf{p}_i^t)}{2}$ and $\mathcal{C}_i$ is the neighbor set selected according to Equation 4. The probability exchange strategy across models provides the opportunity to achieve correct *local* neighbor clustering even when the sample and its neighbors are misaligned with incorrect categories within the feature space of one model, while the other model maintains correct predictions. In Section 4.3.2, we discuss the benefits of the re-weighting strategy and the probability exchange strategy.

3.4 Overall Training Objective

The proposed two losses can effectively exploit *global* and *local* structures, respectively. To further mitigate the risk of model collapse, we minimize the fair diversity-promoting objective following SHOT (Liang et al., 2020):

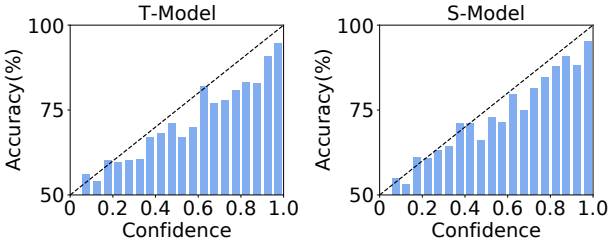

Figure 4: The relationship between the accuracy of K-nearest neighbor samples and the confidence of model predictions for both models on task Ar→Cl of Office-Home. K is set to 4.

$$\mathcal{L}_{div} = \mathbb{E}_{x_t\in\mathcal{D}_t}\sum_{c=1}^{C}\overline{\mathbf{p}}^c log\overline{\mathbf{p}}^c, \overline{\mathbf{p}} = \mathbb{E}_{x_t\in\mathcal{D}_t}\sigma(g(f(x_t))). \quad (7)$$

As a result, the overall training objective is as follows:

$$\mathcal{L}_{total} = \gamma_1\mathcal{L}_{global} + \gamma_2\mathcal{L}_{local} + \gamma_3\mathcal{L}_{div}, \quad (8)$$

where $\gamma_1 = \gamma_2 = \gamma_3 = 1$ are fixed hyper-parameters.

4 EXPERIMENTS

4.1 Experimental Setup

4.1.1 Datasets

Our method's performance is assessed on three standard benchmarks for image classification. The **Office-31** benchmark (Saenko et al., 2010) comprises three domains within office environments: Amazon (**A**), DSLR (**D**), and Webcam (**W**), each containing 31 categories. The **Office-Home** dataset (Venkateswara et al., 2017) poses a more challenging task, involving images of everyday objects distributed across four domains: Artistic (**Ar**), Clipart (**Cl**), Product (**Pr**), and Real-World (**Rw**), each with 65 classes. The **VisDA-C** dataset (Peng et al., 2017) serves as a large-scale dataset for synthetic-to-real domain adaptation, with 152,397 synthetic images in the source domain and 55,388 real-world images in the target domain. Our reported metrics include classification accuracy on the target domain for all domain pairs in Office-31 and Office-Home, as well as per-class accuracy on the real domain in VisDA-C.

4.1.2 Implementation Details

To ensure a fair comparison with related methodologies, we utilize the ResNet-50 backbone for Office-31 and Office-Home datasets and ResNet-101 for VisDA-C. Following the approach of SHOT (Liang et al., 2020), we introduce an additional 256-dimensional fully-connected+BatchNorm bottleneck after the feature extractor and apply Weight-Norm (Salimans and Kingma, 2016) on the classifier,

Table 1: Accuracies (%) on Office-Home for ResNet50-based methods.

Method	SF	Office-Home												
		Ar→Cl	Ar→Pr	Ar→Rw	Cl→Ar	Cl→Pr	Cl→Rw	Pr→Ar	Pr→Cl	Pr→Rw	Rw→Ar	Rw→Cl	Rw→Pr	Avg
SENTRY (ICCV'21)	✗	61.8	77.4	80.1	66.3	71.6	74.7	66.8	63.0	80.9	74.0	66.3	84.1	72.2
FixBi (CVPR'21)	✗	58.1	77.3	80.4	67.7	79.5	78.1	65.8	57.9	81.7	76.4	62.9	86.7	72.7
SCDA (ICCV'21)	✗	60.7	76.4	82.8	69.8	77.5	78.4	68.9	59.0	82.7	74.9	61.8	84.5	73.1
SHOT-IM (ICML'20)	✓	55.4	76.6	80.4	66.9	74.3	75.4	65.6	54.8	80.7	73.7	58.4	83.4	70.5
SHOT (ICML'20)	✓	57.1	78.1	81.5	68.0	78.2	78.1	67.4	54.9	82.2	73.3	58.8	84.3	71.8
A^2Net (ICCV'21)	✓	58.4	79.0	82.4	67.5	79.3	78.9	68.0	56.2	82.9	74.1	60.5	85.0	72.8
G-SFDA (ICCV'21)	✓	57.9	78.6	81.0	66.7	77.2	77.2	65.6	56.0	82.2	72.0	57.8	83.4	71.3
NRC (NeurIPS'21)	✓	57.7	80.3	82.0	68.1	79.8	78.6	65.3	56.4	83.0	71.0	58.6	85.6	72.2
DE (CVPR'22)	✓	59.7	79.5	82.4	69.7	78.6	79.2	66.1	57.2	82.6	73.9	60.8	85.5	72.9
DIPE (CVPR'22)	✓	56.5	79.2	80.7	70.1	79.8	78.8	67.9	55.1	83.5	74.1	59.3	84.8	72.5
BMD (ECCV'22)	✓	58.1	79.7	82.6	69.3	81.0	80.7	**70.8**	57.6	**83.6**	74.0	60.0	85.9	73.6
AaD (NeurIPS'22)	✓	59.3	79.3	82.1	68.9	79.8	79.5	67.2	57.4	83.1	72.1	58.5	85.4	72.7
CRCo* (CVPR'23)	✓	64.2	82.4	85.0	**71.8**	**82.0**	82.2	70.0	61.8	82.6	**74.4**	64.4	**87.7**	75.7
I-SFDA (CVPR'24)	✓	60.7	78.9	82.0	69.9	79.5	79.7	67.1	58.8	82.3	74.2	61.3	86.4	73.4
DPC (CVPR'24)	✓	59.5	80.6	82.9	69.4	79.3	80.1	67.3	57.2	83.7	73.1	58.9	84.9	73.1
Ours	✓	**66.5**	**83.4**	**85.4**	70.6	81.4	**82.7**	70.0	**63.8**	83.3	73.3	**66.7**	87.6	**76.2**

Table 2: Accuracies (%) on Office-31 for ResNet50-based methods.

Method	SF	Office-31						
		A→D	A→W	D→W	W→D	D→A	W→A	Avg
FixBi (CVPR'21)	✗	95.0	96.1	99.3	100.0	78.7	79.4	91.4
SCDA (ICCV'21)	✗	95.4	95.3	99.0	100.0	77.2	75.9	90.5
SHOT-IM (ICML'20)	✓	90.6	91.2	98.3	99.9	72.5	71.4	87.3
SHOT (ICML'20)	✓	94.0	90.1	98.4	99.9	74.7	74.3	88.6
A^2Net (ICCV'21)	✓	94.5	94.0	99.2	**100.0**	76.7	76.1	90.1
NRC (NeurIPS'21)	✓	96.0	90.8	99.0	**100.0**	75.3	75.0	89.4
DE (CVPR'22)	✓	96.0	94.2	98.5	99.8	76.6	75.5	90.1
DIPE (CVPR'22)	✓	**96.6**	93.1	98.4	99.6	75.5	77.2	90.1
BMD (ECCV'22)	✓	96.2	94.2	98.0	**100.0**	76.0	76.0	90.1
AaD (NeurIPS'22)	✓	96.4	92.1	**99.1**	**100.0**	75.0	76.5	89.9
CRCo* (CVPR'23)	✓	**96.6**	**95.4**	98.5	99.8	77.1	77.8	90.9
I-SFDA (CVPR'24)	✓	95.3	94.2	98.3	99.9	76.4	77.5	90.3
DPC (CVPR'24)	✓	95.8	94.5	98.9	**100.0**	76.5	76.8	90.5
Ours	✓	95.8	94.5	98.2	99.8	**78.8**	**80.0**	**91.2**

which is fixed in all experiments. During target adaptation, the model is trained using the SGD optimizer for 15,000 iterations with a batch size of 64 and a learning rate of 0.0001. Regarding hyper-parameters, we empirically set the threshold (τ) to 0.95, in line with FixMatch (Sohn et al., 2020). The trade-off parameters ($\gamma_1, \gamma_2, \gamma_3$) are directly set to 1.0. The number of nearest neighbors (K) is chosen as 4 for Office-31 and Office-Home and 5 for VisDA-C. All the experiments are implemented using Pytorch 1.9.0 and trained on PC with two NVIDIA GeForce RTX3090 GPUs.

4.2 Results

We compare our approach with state-of-the-art methods: (1) Unsupervised domain adaptation: Minimum Class Confusion (**MCC**) (Jin et al., 2020), Selective Entropy Optimization via Committee Consistency (**SENTRY**) (Prabhu et al., 2021), Bridging Domain Spaces (**FixBi**) (Na et al., 2021), Semantic Concentration for Domain Adaptation (**SCDA**) (Li et al., 2021); (2) Source-free domain adaptation: Source Hypothesis Transfer (**SHOT-IM** and **SHOT**) (Liang et al., 2020), Adaptive Adversarial Network (**A^2Net**) (Xia et al., 2021), Generalized Source-free Domain Adaptation (**G-SFDA**) (Yang et al., 2021c), Neighborhood Reciprocity Clustering (**NRC**) (Yang et al., 2021b), Distribution Estimation (**DE**) (Ding et al., 2022), Domain-Invariant Parameter Exploring (**DIPE**) (Wang et al., 2022), class-Balanced Multicentric Dynamic prototype (**BMD**) (Qu et al., 2022), Attracting and Dispersing (**AaD**) (Yang et al., 2022), Class Relationship Embedded Learning (**CRCo**) (Zhang et al., 2023), Lens of Data Augmentation (**SF(DA)^2**) (Hwang et al., 2024), Improving SFDA (**I-SFDA**) (Mitsuzumi et al., 2024), Discriminative Pattern Calibration (**DPC**) (Xia et al., 2024).

Table 1, Table 2 and Table 3 report the classification accuracy on three standard benchmarks: Office-Home, Office-31, and VisDA-C, respectively. In each table, SF denotes *source-free*, and * denotes results reproduced from the provided code. In Table 1, experimental results show that our approach markedly surpasses all state-of-the-art methods on the Office-Home dataset. We enhance the average accuracy from 75.7% to 76.2% without accessing the source data. Furthermore, among all *source-free* methods, our method exhibits superior performance in 6 out of 12 tasks. Specifically, our approach achieves a performance boost of over 2% in three tasks (*i.e.,* 2.3%↑ on task Ar→Cl, 2.0%↑ on task Pr→Cl, and 2.3%↑ on task Rw→Cl). This highlights that preserving statistics in the source model effectively aids in correcting the shift of *source-like* samples. As shown in Table 2, our method also outperforms others on the Office-31 dataset, elevating the average ac-

Table 3: Accuracies (%) on VisDA-C for ResNet101-based methods.

Method	SF	VisDA-C												
		plane	bcycl	bus	car	horse	knife	mcycl	person	plant	sktbrd	train	truck	Avg
MCC (ECCV'20)	✗	88.1	80.3	80.5	71.5	90.1	93.2	85.0	71.6	89.4	73.8	85.0	36.9	78.8
FixBi (CVPR'21)	✗	96.1	87.8	90.5	90.3	96.8	95.3	92.8	88.7	97.2	94.2	90.9	25.7	87.2
SHOT-IM (ICML'20)	✓	93.7	86.4	78.7	50.7	91.0	93.5	79.0	78.3	89.2	85.4	87.9	51.1	80.4
SHOT (ICML'20)	✓	94.3	88.5	80.1	57.3	93.1	94.9	80.7	80.3	91.5	89.1	86.3	58.2	82.9
A^{2}Net (ICCV'21)	✓	94.0	87.8	85.6	66.8	93.7	95.1	85.8	81.2	91.6	88.2	86.5	56.0	84.3
NRC (NeurIPS'21)	✓	96.8	91.3	82.4	62.4	96.2	95.9	86.1	80.6	94.8	94.1	90.4	59.7	85.9
DE (CVPR'22)	✓	95.3	91.2	77.5	72.1	95.7	**97.8**	85.5	86.1	95.5	93.0	86.3	61.6	86.5
DIPE (CVPR'22)	✓	95.2	87.6	78.8	55.9	93.9	95.0	84.1	81.7	92.1	88.9	85.4	58.0	83.1
BMD (ECCV'22)	✓	96.9	87.8	**90.1**	**91.3**	97.8	**97.8**	90.6	84.4	96.9	94.3	90.9	45.9	88.7
AaD (NeurIPS'22)	✓	97.4	90.5	80.8	76.2	97.3	96.1	89.8	82.9	95.5	93.0	92.0	64.7	88.0
CRCo* (CVPR'23)	✓	98.1	92.4	87.8	89.4	**98.0**	97.4	**94.8**	**88.4**	97.2	**95.9**	93.2	52.9	90.4
SFDA^2 (ICLR'24)	✓	96.8	89.3	82.9	81.4	96.8	95.7	90.4	81.3	95.5	93.7	88.5	**64.7**	88.1
I-SFDA (CVPR'24)	✓	97.5	91.4	87.9	79.4	97.2	97.2	92.2	83.0	96.4	94.2	91.1	53.0	88.4
DPC (CVPR'24)	✓	96.5	89.3	86.5	83.2	97.4	97.3	91.8	83.7	96.4	94.8	92.1	56.2	88.8
Ours	✓	**98.3**	**92.5**	88.1	88.4	97.6	96.5	94.1	86.9	**97.9**	95.6	**94.1**	59.4	**90.8**

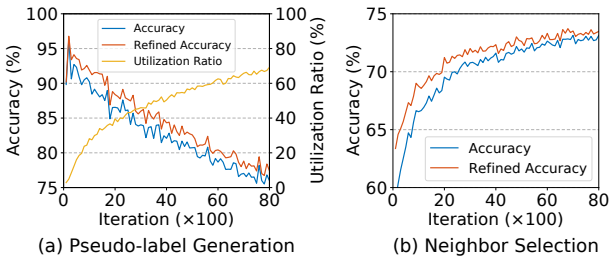

Figure 5: The accuracy trend of the refined pseudo-labels (left) and refined neighbor samples (right). The co-learning strategy helps generate more robust pseudo-labels and select more precise neighbor samples.

curacy from 90.9% to 91.2%. Notably, in two of the tasks, co-learning with the source model yields substantial improvements (*i.e.,* 1.7%↑ on task D→A and 2.2%↑ on task W→A). Concerning the large-scale and challenging synthetic-to-real VisDA-C dataset, our method attains the highest average per-class accuracy. These outcomes affirm that by incorporating the source model into the domain adaptation process, our approach achieves a more accurate exploration of both *global* and *local* structures.

4.3 Analysis

4.3.1 Effect of Joint Learning with S-Model

Here, we illustrate how the integration of the source model into the domain adaptation process facilitates the exploitation of both *global* and *local* structures. As outlined in Section 3, we introduce the co-learning strategy to generate more robust pseudo-labels and to select more precise neighbor samples. Figure 5 plots the accuracy trend of the refined pseudo-labels and refined neighbor samples for the Ar→Cl task in the Office-Home dataset. As domain adaptation progresses, the utilization ratio of pseudo-labels gradually rises, resulting in a decline in pseudo-label accuracy and an enhancement in neighbor sample accuracy. Nevertheless, whether it is pseudo-labels or neighbor samples, the joint learning strategy proves effective in bolstering their robustness by improving accuracy.

Table 4: Ablation study (%) on Office-Home (ResNet-50).

Co-learning		Neighbor cluster		Ar→Cl	Ar→Pr	Ar→Rw	Avg
Pseudo-label generation	Neighbor discovery	Re-weight strategy	Probability exchange				
✗	✓	✓	✓	65.7	83.4	85.4	78.2
✓	✗	✓	✓	64.4	81.7	84.6	76.9
✓	✓	✗	✓	65.3	83.2	84.7	77.7
✓	✓	✓	✗	65.9	82.5	84.9	77.8
✓	✓	✓	✓	**66.5**	**83.4**	**85.4**	**78.4**

4.3.2 Ablation Study

In Table 4, we present the outcomes of ablation studies conducted on various strategies within our approach across three tasks in the Office-Home dataset. As outlined in Section 3, we propose a co-learning strategy with the source

Table 5: Impact of the pseudo-label generation strategy on Ar→Cl, Ar→Pr, and Ar→Rw tasks in Office-Home.

$\hat{\mathbf{p}}^s = \hat{\mathbf{p}}^t$	$Max(\mathbf{p}^s) \geq \tau$	$Max(\mathbf{p}^t) \geq \tau$	Pseudo-label Strategy		
			Equal	OR	AND
✓	✓	✓	$\hat{\mathbf{p}}^t$	$\hat{\mathbf{p}}^t$	$\hat{\mathbf{p}}^t$
✓	✗	✓	$\hat{\mathbf{p}}^t$	$\hat{\mathbf{p}}^t$	-
✓	✓	✗	$\hat{\mathbf{p}}^t$	$\hat{\mathbf{p}}^t$	-
✓	✗	✗	$\hat{\mathbf{p}}^t$	-	-
✗	✓/✗	✓/✗	-	-	-
Average Accuracy			77.8	78.0	78.4

model to enhance both the robustness of pseudo-label generation and the accuracy of neighbor discovery. Ablation experiments specifically targeting the co-learning strategy were carried out, and the results are detailed in the initial two rows of Table 4. Within the co-learning strategy, it is noteworthy that achieving more accurate neighbor discovery resulted in a more substantial performance improvement (1.5%↑) compared to enhancing the robustness of pseudo-label generation (0.2%↑). The improvement observed with co-learning demonstrates that preserving statistics in the source model helps correct the shift of *source-like* samples in the feature space. Furthermore, the last two rows in Table 4 highlight that the re-weighting strategy and probability exchange strategy can individually boost accuracy by 0.7% and 0.6%, respectively. This demonstrates their efficacy in reducing neighbor noise and fostering improved *local* neighbor clustering.

4.3.3 Pseudo-label Generation Strategy

In this section, we investigate the impact of the pseudo-label generation strategy in Equation 1. We compare the logical AND strategy with two other approaches: logical OR and Equal. These strategies, along with the corresponding experimental results, are summarized in Table 5. The logical OR and Equal strategies generated more pseudo-labels, but also introduced more noise, resulting in accuracy drops of 0.4% and 0.6%, respectively. The experimental findings suggest that, in the presence of local neighbor clustering, the quality of the pseudo-labels is more important than their quantity.

4.3.4 Parameter Sensitivity

We conducted a sensitivity analysis of hyper-parameters on the Ar→Cl and Ar→Pr tasks in the Office-Home dataset. Two hyper-parameters, the probability threshold τ and the number of nearest neighbors K, are considered in Figure 6 and Figure 7, respectively. The threshold τ regulates the generation of pseudo-labels, where a too-low value introduces significant noise, and a too-high value reduces pseudo-label quantity (with $\tau = 1$ implying no pseudo-label generation). Setting τ to 0.95 strikes a balance between pseudo-label quantity and quality. Regarding the parameter K, our method demonstrates stable accuracy within specific ranges, showcasing its robustness to the choice of K.

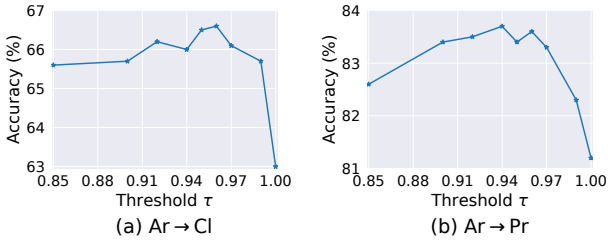

Figure 6: Sensitivity analysis of the probability threshold τ on the task of Ar→Cl and Ar→Pr in Office-Home.

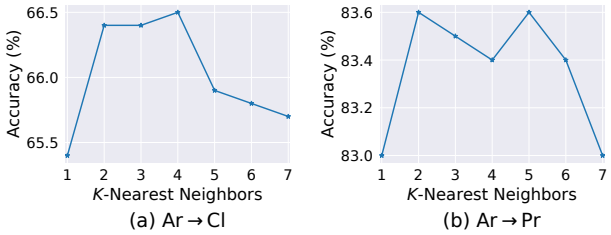

Figure 7: Sensitivity analysis of the number of nearest neighbors K on the task of Ar→Cl and Ar→Pr in Office-Home.

4.3.5 Feature Visualization

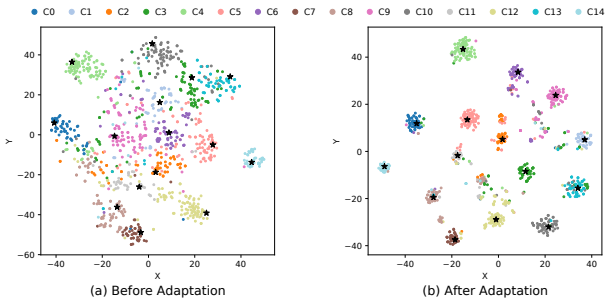

Figure 8: The t-SNE visualization of classifier weights (represented by stars) and target domain features (represented by dots) before (left) and after (right) adaptation on the task Ar→Cl in Office-Home. We directly utilize the first 15 classes, with distinct colors assigned to each class for differentiation.

Figure 8 (a) and (b) demonstrate the t-SNE visualization of target domain features for the first 15 classes in the challenging task Ar→Cl within the Office-Home dataset, both before and after adaptation. In Figure 8 (a), a significant amount of target data (depicted by dots) is widely dispersed in the feature space before adaptation, highlighting the presence of a substantial domain shift problem. After SFDA training, as depicted in Figure 8 (b), target domain features are drawn towards corresponding anchors

Table 6: Impact of the data augmentations on Ar→Cl, Ar→Pr, and Ar→Rw tasks in Office-Home..

Weak Augmentation	Strong Augmentation		Avg
Flip and Crop	RandAugment	Cutout	
✗	✗	✗	73.9
✓	✗	✗	77.4
✓	✓	✗	78.2
✓	✓	✓	78.4

(*i.e.,* the weight vectors in the fixed classifier) and amalgamated into dense clusters. This indicates that our approach can learn discriminative features for the unlabeled target domain without relying on source data.

4.3.6 Data Augmentation

In this section, we examine the effect of weak-strong augmentations, which have proven highly effective in semi-supervised learning (Sohn et al., 2020) and have been integrated into several SFDA methods (e.g., CRCo (Zhang et al., 2023)). Weak augmentation typically involves random image flipping and cropping, while strong augmentation, introduced by RandAugment (Cubuk et al., 2020), applies 14 random transformations followed by Cutout (DeVries and Taylor, 2017). The results of the ablation study on weak-strong augmentations are summarized in Table 6. The findings show that strong augmentations boost accuracy by 1%, while the absence of any data augmentation causes a significant accuracy drop of 4.5%.

4.3.7 Limitation

Recent studies (Yang et al., 2021a; Sanyal et al., 2023) have explored SFDA approaches based on vision transformers. The proposed co-training strategy cannot be directly applied to transformer-based models, as they utilize layer normalization rather than batch normalization. Despite this, the co-training framework can still serve as a plug-in when two models are available.

5 CONCLUSION

In this paper, we propose to preserve statistics in the source model to correct the shift of *source-like* samples in the feature space. By incorporating the source model into the domain adaptation process, we devise a two-branch co-learning strategy to enhance *global* class clustering by generating more robust pseudo-labels and to improve *local* neighbor clustering through the identification of more accurate neighbor samples. Additionally, we introduce the re-weighting strategy and the probability exchange strategy to mitigate neighbor noise resulting from domain shift. Extensive experiments demonstrate the effectiveness of our proposed method, showcasing state-of-the-art performance.

Acknowledgements

This work is supported by SJTU-QI'ANXIN Joint Lab of Information System Security. We are grateful to anonymous reviews for their constructive comments to improve this paper.

References

Mihael Ankerst, Markus M Breunig, Hans-Peter Kriegel, and Jörg Sander. Optics: Ordering points to identify the clustering structure. *ACM Sigmod record*, 28(2):49–60, 1999.

Chaoqi Chen, Weiping Xie, Wenbing Huang, Yu Rong, Xinghao Ding, Yue Huang, Tingyang Xu, and Junzhou Huang. Progressive feature alignment for unsupervised domain adaptation. In *CVPR*, pages 627–636, 2019.

Safa Cicek and Stefano Soatto. Unsupervised domain adaptation via regularized conditional alignment. In *ICCV*, pages 1416–1425, 2019.

Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *CVPR*, pages 702–703, 2020.

Shuhao Cui, Shuhui Wang, Junbao Zhuo, Chi Su, Qingming Huang, and Qi Tian. Gradually vanishing bridge for adversarial domain adaptation. In *CVPR*, pages 12455–12464, 2020.

Zhijie Deng, Yucen Luo, and Jun Zhu. Cluster alignment with a teacher for unsupervised domain adaptation. In *ICCV*, pages 9944–9953, 2019.

Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

Ning Ding, Yixing Xu, Yehui Tang, Chao Xu, Yunhe Wang, and Dacheng Tao. Source-free domain adaptation via distribution estimation. In *CVPR*, pages 7212–7222, 2022.

Geoff French, Michal Mackiewicz, and Mark Fisher. Self-ensembling for visual domain adaptation. In *ICLR*, 2018.

Uiwon Hwang, Jonghyun Lee, Juhyeon Shin, and Sungroh Yoon. SF(DA)2: source-free domain adaptation through the lens of data augmentation. In *ICLR*, 2024.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, pages 448–456, 2015.

Ying Jin, Ximei Wang, Mingsheng Long, and Jianmin Wang. Minimum class confusion for versatile domain adaptation. In *ECCV*, pages 464–480, 2020.

Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. Contrastive adaptation network for unsupervised domain adaptation. In *CVPR*, pages 4893–4902, 2019.

Seunghyeon Kim, Jaehoon Choi, Taekyung Kim, and Changick Kim. Self-training and adversarial background regularization for unsupervised domain adaptive one-stage object detection. In *ICCV*, pages 6092–6101, 2019.

Youngeun Kim, Donghyeon Cho, Kyeongtak Han, Priyadarshini Panda, and Sungeun Hong. Domain adaptation without source data. *IEEE Transactions on Artificial Intelligence*, 2(6):508–518, 2021.

Ananya Kumar, Tengyu Ma, and Percy Liang. Understanding self-training for gradual domain adaptation. In *ICML*, pages 5468–5479, 2020.

Jogendra Nath Kundu, Suvaansh Bhambri, Akshay Kulkarni, Hiran Sarkar, Varun Jampani, and R Venkatesh Babu. Concurrent subsidiary supervision for unsupervised source-free domain adaptation. In *ECCV*, pages 177–194, 2022a.

Jogendra Nath Kundu, Akshay R Kulkarni, Suvaansh Bhambri, Deepesh Mehta, Shreyas Anand Kulkarni, Varun Jampani, and Venkatesh Babu Radhakrishnan. Balancing discriminability and transferability for source-free domain adaptation. In *ICML*, pages 11710–11728, 2022b.

Junjie Li, Zilei Wang, Yuan Gao, and Xiaoming Hu. Exploring high-quality target domain information for unsupervised domain adaptive semantic segmentation. In *ACM MM*, pages 5237–5245, 2022.

Rui Li, Qianfen Jiao, Wenming Cao, Hau-San Wong, and Si Wu. Model adaptation: Unsupervised domain adaptation without source data. In *CVPR*, pages 9641–9650, 2020.

Shuang Li, Mixue Xie, Fangrui Lv, Chi Harold Liu, Jian Liang, Chen Qin, and Wei Li. Semantic concentration for domain adaptation. In *ICCV*, pages 9102–9111, 2021.

Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *ICML*, pages 6028–6039, 2020.

Jian Liang, Dapeng Hu, Jiashi Feng, and Ran He. Dine: Domain adaptation from single and multiple black-box predictors. In *CVPR*, pages 8003–8013, 2022.

Hong Liu, Mingsheng Long, Jianmin Wang, and Michael Jordan. Transferable adversarial training: A general approach to adapting deep classifiers. In *ICML*, pages 4013–4022, 2019.

Hong Liu, Jianmin Wang, and Mingsheng Long. Cycle self-training for domain adaptation. *NeurIPS*, 34:22968–22981, 2021.

Qinying Liu and Zilei Wang. Collaborating domain-shared and target-specific feature clustering for cross-domain 3d action recognition. In *ECCV*, pages 137–155, 2022.

Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In *ICML*, pages 97–105, 2015.

Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Unsupervised domain adaptation with residual transfer networks. *NeurIPS*, 29, 2016.

Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Deep transfer learning with joint adaptation networks. In *ICML*, pages 2208–2217, 2017.

Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. *NeurIPS*, 31, 2018.

Ke Mei, Chuang Zhu, Jiaqi Zou, and Shanghang Zhang. Instance adaptive self-training for unsupervised domain adaptation. In *ECCV*, pages 415–430, 2020.

Yu Mitsuzumi, Akisato Kimura, and Hisashi Kashima. Understanding and improving source-free domain adaptation from a theoretical perspective. In *CVPR*, pages 28515–28524, 2024.

Jaemin Na, Heechul Jung, Hyung Jin Chang, and Wonjun Hwang. Fixbi: Bridging domain spaces for unsupervised domain adaptation. In *CVPR*, pages 1094–1103, 2021.

Fei Pan, Inkyu Shin, Francois Rameau, Seokju Lee, and In So Kweon. Unsupervised intra-domain adaptation for semantic segmentation through self-supervision. In *CVPR*, pages 3764–3773, 2020.

Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.

Yingwei Pan, Ting Yao, Yehao Li, Yu Wang, Chong-Wah Ngo, and Tao Mei. Transferrable prototypical networks for unsupervised domain adaptation. In *CVPR*, pages 2239–2247, 2019.

Xingchao Peng, Ben Usman, Neela Kaushik, Judy Hoffman, Dequan Wang, and Kate Saenko. Visda: The visual domain adaptation challenge. *arXiv preprint arXiv:1710.06924*, 2017.

Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *ICCV*, pages 1406–1415, 2019.

Viraj Prabhu, Shivam Khare, Deeksha Kartik, and Judy Hoffman. Sentry: Selective entropy optimization via committee consistency for unsupervised domain adaptation. In *ICCV*, pages 8558–8567, 2021.

Sanqing Qu, Guang Chen, Jing Zhang, Zhijun Li, Wei He, and Dacheng Tao. Bmd: A general class-balanced multicentric dynamic prototype strategy for source-free domain adaptation. In *ECCV*, pages 165–182, 2022.

Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *ECCV*, pages 213–226, 2010.

Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *CVPR*, pages 3723–3732, 2018.

Tim Salimans and Durk P Kingma. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. *NeurIPS*, 29, 2016.

Sunandini Sanyal, Ashish Ramayee Asokan, Suvaansh Bhambri, Akshay Kulkarni, Jogendra Nath Kundu, and R Venkatesh Babu. Domain-specificity inducing transformers for source-free domain adaptation. In *ICCV*, pages 18928–18937, 2023.

Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *NeurIPS*, 33:596–608, 2020.

Masashi Sugiyama, Shinichi Nakajima, Hisashi Kashima, Paul Buenau, and Motoaki Kawanabe. Direct importance estimation with model selection and its application to covariate shift adaptation. *NeurIPS*, 20, 2007.

Hui Tang, Ke Chen, and Kui Jia. Unsupervised domain adaptation via structurally regularized deep clustering. In *CVPR*, pages 8725–8735, 2020.

Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474*, 2014.

Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *CVPR*, pages 5018–5027, 2017.

Fan Wang, Zhongyi Han, Yongshun Gong, and Yilong Yin. Exploring domain-invariant parameters for source free domain adaptation. In *CVPR*, pages 7151–7160, 2022.

Ximei Wang, Liang Li, Weirui Ye, Mingsheng Long, and Jianmin Wang. Transferable attention for domain adaptation. In *AAAI*, volume 33, pages 5345–5352, 2019.

Garrett Wilson and Diane J Cook. A survey of unsupervised deep domain adaptation. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11(5):1–46, 2020.

Haifeng Xia, Handong Zhao, and Zhengming Ding. Adaptive adversarial network for source-free domain adaptation. In *ICCV*, pages 9010–9019, 2021.

Haifeng Xia, Siyu Xia, and Zhengming Ding. Discriminative pattern calibration mechanism for source-free domain adaptation. In *CVPR*, pages 23648–23658, 2024.

Shaoan Xie, Zibin Zheng, Liang Chen, and Chuan Chen. Learning semantic representations for unsupervised domain adaptation. In *ICML*, pages 5423–5432, 2018.

Hongliang Yan, Yukang Ding, Peihua Li, Qilong Wang, Yong Xu, and Wangmeng Zuo. Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation. In *CVPR*, pages 2272–2281, 2017.

Guanglei Yang, Hao Tang, Zhun Zhong, Mingli Ding, Ling Shao, Nicu Sebe, and Elisa Ricci. Transformer-based source-free domain adaptation. *arXiv preprint arXiv:2105.14138*, 2021a.

Senqiao Yang, Jiarui Wu, Jiaming Liu, Xiaoqi Li, Qizhe Zhang, Mingjie Pan, Yulu Gan, Zehui Chen, and Shanghang Zhang. Exploring sparse visual prompt for domain adaptive dense prediction. In *AAAI*, volume 38, pages 16334–16342, 2024.

Shiqi Yang, Joost van de Weijer, Luis Herranz, Shangling Jui, et al. Exploiting the intrinsic neighborhood structure for source-free domain adaptation. *NeurIPS*, 34:29393–29405, 2021b.

Shiqi Yang, Yaxing Wang, Joost Van De Weijer, Luis Herranz, and Shangling Jui. Generalized source-free domain adaptation. In *ICCV*, pages 8978–8987, 2021c.

Shiqi Yang, Shangling Jui, Joost van de Weijer, et al. Attracting and dispersing: A simple approach for source-free domain adaptation. *NeurIPS*, 35:5802–5815, 2022.

Pan Zhang, Bo Zhang, Ting Zhang, Dong Chen, Yong Wang, and Fang Wen. Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation. In *CVPR*, pages 12414–12424, 2021a.

Yixin Zhang and Zilei Wang. Joint adversarial learning for domain adaptation in semantic segmentation. In *AAAI*, volume 34, pages 6877–6884, 2020.

Yixin Zhang, Zilei Wang, and Yushi Mao. Rpn prototype alignment for domain adaptive object detector. In *CVPR*, pages 12425–12434, 2021b.

Yixin Zhang, Zilei Wang, and Weinan He. Class relationship embedded learning for source-free unsupervised domain adaptation. In *CVPR*, pages 7619–7629, 2023.

Ziyi Zhang, Weikai Chen, Hui Cheng, Zhen Li, Siyuan Li, Liang Lin, and Guanbin Li. Divide and contrast: Source-free domain adaptation via adaptive contrastive learning. *NeurIPS*, 35:5137–5149, 2022.

Yangtao Zheng, Di Huang, Songtao Liu, and Yunhong Wang. Cross-domain object detection through coarse-to-fine feature adaptation. In *CVPR*, pages 13766–13775, 2020.

Checklist

The checklist follows the references. For each question, choose your answer from the three possible options: Yes, No, Not Applicable. You are encouraged to include a justification to your answer, either by referencing the appropriate section of your paper or providing a brief inline description (1-2 sentences). Please do not modify the questions. Note that the Checklist section does not count towards the page limit. Not including the checklist in the first submission won't result in desk rejection, although in such case we will ask you to upload it during the author response period and include it in camera ready (if accepted).

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   We provide clear descriptions in section 3.
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Not Applicable]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Not Applicable]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Yes]
   (c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
   See Abstract.
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   See section 4.1.2.
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [No]
   Experimental results in SFDA are very stable.
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]
   See section 4.1.2.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]
   All relevant papers have been cited.
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Yes]
   See Abstract.
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   We didn't use crowdsourcing or conduct research with human subjects.
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   We didn't use crowdsourcing or conduct research with human subjects.
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]
   We didn't use crowdsourcing or conduct research with human subjects.