Rethinking Neighborhood Consistency Learning on Unsupervised Domain Adaptation

Chang Liu
Northeastern University
Boston, United States
liu.chang6@northeastern.edu

Lichen Wang
Northeastern University
Boston, United States
wanglichenxj@gmail.com

Yun Fu
Northeastern University
Boston, United States
yunfu@ece.neu.edu

ABSTRACT

Unsupervised domain adaptation (UDA) involves predicting unlabeled data in a target domain by using labeled data from the source domain. However, recent advances in pseudo-labeling (PL) methods have been hampered by noisy pseudo-labels that diminish the local discriminativeness of the target structure. Although neighborhood-based PL can help preserve the local structure, it also risks assigning the whole local neighborhood to the wrong semantic category. To address this issue, we propose a novel framework called neighborhood consistency learning (NCL) that operates at both the semantic and instance levels and features a new consistency objective function. Specifically, our objective function aims to promote semantic consistency in the target neighborhood by computing the correlation matrix between the target samples and their neighborhood aggregation over a batch and matching the correlation matrix to an identity matrix. Importantly, our approach allows the target neighborhood to receive gradients from several potential positive categories instead of just one certain category. Our extensive experiments on UDA benchmarks demonstrate the effectiveness of NCL over other state-of-the-art PL-based methods.

CCS CONCEPTS

• **Computing methodologies** → **Object recognition**; **Image representations**.

KEYWORDS

Domain Adaptation, Transfer Learning, Pseudo-labeling

ACM Reference Format:
Chang Liu, Lichen Wang, and Yun Fu. 2023. Rethinking Neighborhood Consistency Learning on Unsupervised Domain Adaptation. In *Proceedings of the 31st ACM International Conference on Multimedia (MM '23), October 29-November 3, 2023, Ottawa, ON, Canada.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3581783.3612055

1 INTRODUCTION

Recent advances in deep neural networks have revolutionized many computer vision tasks, such as image recognition [6, 18, 22] and face recognition[19]. However, achieving these remarkable results heavily depends on costly and time-consuming human annotations. To tackle this challenge, researchers have explored semi-supervised learning (SSL) [23] and self-learning [5] to transfer knowledge from label-rich datasets to label-scarce ones. however, the domain shift problem between the source and target dataset usually exists in real-worlds scenario, leading to performance degradation. To mitigate this problem, domain adaptation (DA) has been exploited to transfer knowledge across datasets with domain discrepancy. Unsupervised domain adaptation (UDA) is a more challenging scenario where the target domain does not have labels.

One direction of UDA involves learning domain-invariant representations by simultaneously minimizing the source error and reducing the discrepancy between domains [20, 30]. Adversarial learning, which uses a domain discriminator, is effective for this purpose. However, it only aligns the global feature distribution of two domains and does not consider the categorical structure of target data, limiting the model's generalization on the target domain. Another UDA direction is to directly apply SSL techniques to UDA problems [10, 39] by considering the target domain as unlabeled data. Pseudo-labeling (PL) methods [33, 40] (Fig. 2(a)) have been widely used in UDA by generating pseudo-labels for unlabeled target samples and retraining the network on them in a supervised manner. However, PL is known for propagating errors and deteriorating the local structure of target data. Recent works [15, 37] propose generating pseudo-labels based on the aggregated neighborhood predictions obtained from the classifier, as

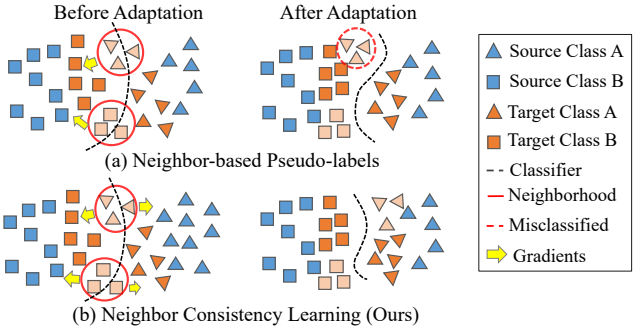

Figure 1: Comparison between neighborhood pseudo-labeling (NPL) [15] and our approach. (a) NPL carries the risk of misclassifying semantic-uncertain neighborhoods into a single wrong category due to unreliable supervision. (b) In contrast, our neighborhood consistency learning (NCL) approach prioritizes semantic consistency learning and enables the target neighborhood to receive gradients from multiple categories which might contain the positive one.

MM '23, October 29-November 3, 2023, Ottawa, ON, Canada

ACM ISBN 979-8-4007-0108-5/23/10…$15.00
https://doi.org/10.1145/3581783.3612055

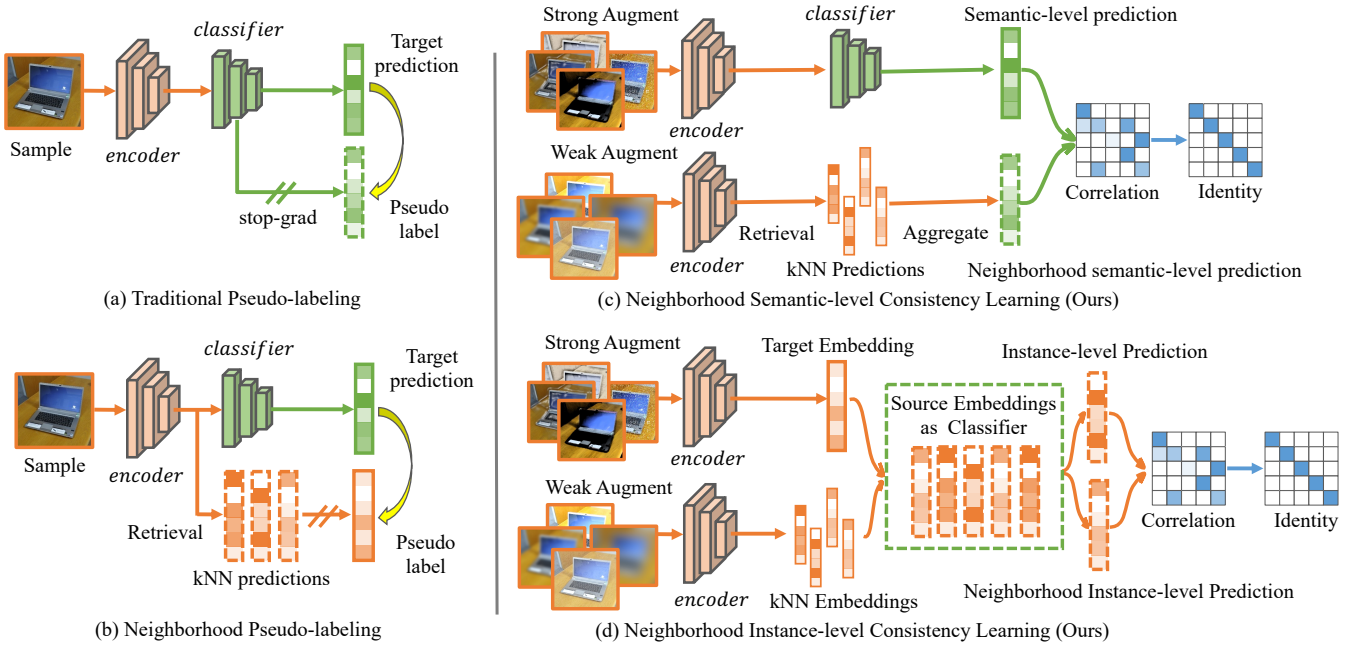

Figure 2: Framework comparison between Pseudo-labeling (PL), Neighborhood Pseudo-labeling (NPL), and Ours: (a) PL retrains the network based on pseudo-labels from one view. (b) NPL utilizes target neighborhood predictions to generate pseudo-labels. (c-d) NCL prioritizes the consistency learning between target samples and their neighborhood at semantic and instance levels.

shown in Fig. 2(b), to preserve the target's local structure. Nonetheless, neighbor-based pseudo-labels may also contain noise, which could push semantic-uncertain neighborhoods to wrong categories, as shown in Fig. 1(a).

In this paper, we present a novel neighborhood consistency learning (NCL) framework, illustrated in Figure 2(c)(d). To address the issue depicted in Fig. 1(a), we propose a new consistency training objective by matching the correlation matrix between target predictions and their aggregated neighbor predictions to an identity matrix, as shown in Figure 2(c). Instead of simply averaging neighbor predictions for aggregation, we design an uncertainty-aware weighted aggregation method to mitigate the adverse effects of potential negative neighbors (i.e., those who do not share the same class as the anchor). Our objective has two advantages: first, it allows the semantic-uncertain target neighborhood to receive gradients from multiple categories with high confidence, rather than from a single specific category. Consequently, it avoids the model having overconfident outputs and trusting on high-confident false positives with error accumulation. Second, it encourages category diversity over the target data.

As the domain shift problem can affect the reliability of semantic-level information from the classifier, we propose to leverage instance-level information for neighborhood consistency learning, as illustrated in Fig. 2(d). To achieve this, we use the embeddings of source data as an instance-level classifier and compute similarity scores between target embeddings and source embeddings (and likewise for target neighborhood embeddings). Similar to the semantic-level objective, we obtain the correlation matrix based on similarity scores and apply correlation matrix matching at the instance level for consistency learning.

Our contribution can be summarized as follows:

- We observe that the existing neighbor-based PL methods have the risk of pushing neighborhoods of target data to wrong categories even though the local discriminative structures are well-preserved.
- We propose a novel neighborhood consistency learning (NCL) framework at semantic and instance levels for unsupervised domain adaptation.
- We propose a new consistency objective in the form of matching the correlation matrix of two positive views to an identity matrix.
- We conduct extensive experiments and ablation studies to verify the effectiveness of NCL thoroughly. Our method could achieve competitive or better results than previous PL-based state-of-the-art methods across several UDA classification benchmarks.

2 RELATED WORK

2.1 Discrepancy-based DA

Existing methods have been explored to align the feature representations of the source and target images by minimizing the distribution discrepancy. For example, Maximum Mean Discrepancy (MMD) [31] is proposed to match the mean and covariance of source and target distributions. Alternatively, adversarial domain adaptation methods [3, 17, 30] solve this domain discrepancy by training a domain-invariant feature generator that produces the features to fool a discriminator that distinguishes the representations from source and target domains. However, since the domain discriminator aligns source and target features without considering the

class labels, merely aligning the global marginal distribution of the features in the two domains fails to align the class-wise distribution.

2.2 Pseudo-labeling in DA

Inspired by cluster assumption, pseudo-labeling can realize the class-wise alignment across domains. Specifically, it iteratively generates pseudo-labels for the target samples with high prediction probability and retrains the network based on those pseudo-labels along with labeled source data. This technique has been widely employed for UDA [12, 24, 33, 34, 40] tasks. However, due to the "overconfidence" issue on wrong pseudo-labels, PL deteriorates the local discriminative structure of target data.

Recent works [15, 28, 37] propose various approaches to preserve this local structure, e.g., Tang et al. [28] adopt deep clustering to assign local neighborhoods of target data to the same clusters. Liang et al. [15] generates pseudo-labels from the neighborhood classifier predictions of target samples. Contrastive learning [16] has been also integrated with pseudo-labels for better discriminative structure. Although category consistency is preserved for the local target neighborhoods, existing approaches have a risk of pushing the neighborhoods to the wrong category. In comparison, NCL introduces a new consistency objective that allows the target neighborhoods to be assigned to several potential positive categories. Further, NCL enforces semantic-level and instance-level consistency learning between target samples and their neighbors with more reliable matching targets.

2.3 Consistency Learning in SSL

Consistency Regularization stands as a popular technique in semi-supervised learning, aimed at ensuring the model generates consistent predictions across different views of the same instance. For example, VAT [23] generates different positive views with adversarial permutations. Mean Teacher [29] explores the exponential moving average (EMA) model and utilizes its output as another view. FixMatch [27] leverage the strong augmentations as another positive view with pseudo-labels retraining. Compared to these methods, NCL proposes a new positive view of unlabeled data with neighborhood information. Further, instead of simply enforcing the prediction consistency, our new objective based on the correlation matrix could avoid pushing the positive views to the wrong category with overconfidence.

3 PRELIMINARY

3.1 Problem Definition

In unsupervised domain adaptation (UDA) problem, we are given a source domain $\mathbb{D}_s = \{(\boldsymbol{x}_i^s, \boldsymbol{y}_i^s)|_{i=1}^{N_s}\}$ of N_s labeled source examples and a target domain $\mathbb{D}_t = \{(\boldsymbol{x}_i^t)|_{i=1}^{N_t}\}$ of N_t unlabeled target examples. Note that source and target domain share the same label space. The joint distributions of source and target domain are not identically and independently distributed, specifically $P(\boldsymbol{x}^s, \boldsymbol{y}^s) \neq Q(\boldsymbol{x}^t, \boldsymbol{y}^t)$. The objective of UDA is to train a deep neural network $G(\cdot|\boldsymbol{\theta})$ on labeled source data $(\boldsymbol{x}_i^s, \boldsymbol{y}_i^s)$ drawn from $\mathbb{D}_s$ and unlabeled target data $\boldsymbol{x}_i^t$ drawn from $\mathbb{D}_t$ such that the model $G(\cdot|\boldsymbol{\theta})$ can generalize well on target domain. In details, network $G(\cdot|\boldsymbol{\theta}) = C \circ F(\cdot|\boldsymbol{\theta})$ is comprised of a feature extractor $F(\cdot|\boldsymbol{\theta})$ and a classifier $C(\cdot|\boldsymbol{\theta})$ where $\boldsymbol{\theta}$ denotes network parameters.

In general, training a network $G(\cdot|\boldsymbol{\theta})$ on source domain only leads to sub-optimal performance as the domain gap issue is unsolved. The cross-entropy loss is applied to the source data in the form of:

$$\mathcal{L}_s(F, C) = \frac{1}{N_s} \sum_{i=1}^{N_s} \mathcal{L}_{ce}\left(C(F(x_i^s|\boldsymbol{\theta})), y_i^s\right). \tag{1}$$

As the consequence, domain alignment loss is incorporated with Eq. 1 to mitigate the domain shift problem.

3.2 Traditional Pseudo-labeling

PL-based methods [33, 40] are prone to make the network be confident on target predictions by retraining the network with pseudo labels which correspond to the largest prediction probability of target samples. The objective is defined as:

$$\mathcal{L}_{pl}^t(\boldsymbol{\theta}) = \frac{1}{N_t} \sum_{i=1}^{N_t} \mathcal{L}_{ce}\left(p_i^t, \tilde{y}_i^t\right), \tag{2}$$

$$y_i^t = \mathbb{1}[\max(p_i^t) > \tau], \quad p_i^t = C(F(x_i^t|\boldsymbol{\theta})), \tag{3}$$

where $\tilde{y}_i^t$ is the pseudo label and p_i^t is the output prediction of the i target sample, and τ is a confidence threshold. Nonetheless, the pseudo labels of target samples are noisy since the model is biased toward source data. Trusting those noisy pseudo-labels has a high risk of misleading the training, and deteriorates the intrinsic discriminative target structures.

3.3 Neighborhood Pseudo-labeling

Recent efforts [15, 37] have been made to preserve the local structure of target data with neighborhood PL. Specifically, they first save all the features and predictions of the target data in the memory bank $\mathbb{V}_t = \{(z_i^t, p_i^t)|_{i=1}^{N_t}\}$. Given a target sample x_i^t, they retrieve k nearest neighbors from the memory bank based on the cosine similarity. The predictions of retrieved neighbors can be aggregated as follows:

$$\hat{p}_i = \frac{1}{k} \sum\nolimits_{j \neq i, j \in \mathcal{N}_i} p_j, \tag{4}$$

$$\hat{y}_i^t = \mathbb{1}[\max(\hat{p}_i) > \tau], \tag{5}$$

where $\mathcal{N}_i$ denotes the index set of neighbors in the memory bank for the sample x_i^t and p_j is the prediction of neighbors. y_i^t is the pseudo label based on the aggregated predictions of the neighborhood. As the neighborhood aggregation on the semantic level (predictions) only is not always reliable, we claim that neighborhood PL methods still have a failure case where semantic-uncertain neighborhoods are pushed to the wrong categories with preserved local structures as Fig. 1(a) shows.

4 METHOD

In this section, motivated by the limitations of existing neighborhood PL methods above, we propose a neighborhood consistency learning (NCL) framework at semantic and instance levels with a new consistency training objective for better reliability as illustrated in Figure 2(c)(d).

4.1 Uncertainty-aware Neighborhood Aggregation

Existing neighborhood PL methods assume that the local neighborhood of target data share the same semantic class (positive), and thus, they treat each neighbor equally by averaging neighbor predictions as an aggregated target. We instead claim that not all the neighbors in the target feature space are positive to the anchor. Intuitively, the purity of the neighborhood in terms of semantic consistency is decreased with the increase of neighborhood size k. In other words, there is a trade-off between neighborhood diversity and purity.

To mitigate the adverse effect of potential negative neighbors, we propose an uncertainty-aware weighted aggregation for the target neighborhood. Following the notation from Eqn. 5, the aggregation can be conducted in both output predictions and feature embeddings as:

$$\hat{\boldsymbol{p}}_i = \frac{1}{k}\sum\nolimits_{j\in\mathcal{N}_i} w_{i,j}\boldsymbol{p}_j, \tag{6}$$

$$\hat{\boldsymbol{z}}_i = \frac{1}{k}\sum\nolimits_{j\in\mathcal{N}_i} w_{i,j}\boldsymbol{z}_j, \tag{7}$$

$$w_{i,j} = 1 - \frac{\mathcal{H}(p_i^t, p_j)}{\log M}, \tag{8}$$

Where $\hat{p}_i$ and $\hat{z}_i$ refer to the weighted aggregation of the prediction and embedding, respectively, for the target sample x_i^t. $w_{i,j}$ denotes the weight for the j-th neighbor of sample i. $\mathcal{H}(a,b) = -a\log(b)$ is the entropy function with M indicating the number of class. As the entropy value $\mathcal{H}(\cdot,\cdot)$ ranges from $(0, \log M]$, the weight $w_{i,j}$ will be constrained between $[0, 1)$. Intuitively, the neighbors with low entropy are more likely to be positive and will be upweighted in aggregation. In contrast, neighbors with high entropy are considered negative and down-weighted instead.

It is worth noting that, different from Eqn. 5, we use weak augmentation as one view for kNN retrieval and neighborhood aggregation while applying strong augmentation as another view as prediction (as shown in Figure 2(c)(d)). Secondly, we consider the sample itself as one of the neighbors because the strong augmentation could largely distort the spatial information while preserving the semantic clue unchanged. In this way, our modification increases the neighborhood diversity with strong augmentation and reduces pseudo-supervision noise by considering the samples themselves as neighbors.

4.2 Semantic-level Consistency Learning

Existing neighborhood PL methods [15] have the risk of pushing semantic-uncertain neighborhoods to the wrong categories even though the local structures are preserved, as Figure 1(a) shows.

Motivated by [38], we propose a new consistency training objective to match the correlation matrix between target predictions and their aggregated neighbor predictions to an identity matrix over a batch. Specifically, as illustrated in Figure 2(c), we construct two views of a target images batch with strong augmentation X_t^{sa} and weak augmentation X_t^{wa}, respectively. The weakly-augmented batch is then utilized to retrieve their k nearest neighbors (kNN) and perform uncertainty-aware neighborhood aggregation in network prediction level in Eqn. 6. After we obtain the output prediction for the strongly-augmented view p_t^{sa} and weighted neighborhood aggregated view $\hat{p}^{wa}$, our consistency objective over a target batch is formulated as:

$$\begin{aligned}\mathcal{L}_{NCL-S} &= \|C^p - \mathcal{I}\|^2, \\ &= \sum_i (1 - C_{ii}^p)^2 + \sum_i \sum_{j\neq i} {C_{ij}^p}^2\end{aligned} \tag{9}$$

where $\mathcal{I}$ is an identity matrix, and C^p is the cross-correlation matrix computed between the $p_{b,t}^{sa}$ and $\hat{p}_{b,t}^{wa}$ along the batch dimension:

$$\mathbf{C}^p = \frac{p_{b,t}^{sa}\cdot\hat{p}_{b,t}^{wa}}{\sqrt{(p_{b,t}^{sa})^2}\sqrt{(\hat{p}_{b,t}^{wa})^2}} \tag{10}$$

where b indexes batch samples. C^p is a square matrix with size the dimension M of the network's output.

Intuitively, the first term of the objective enforces the diagonal elements of the cross-correlation matrix to 1. In this way, the target samples are prone to be semantically consistent with their neighborhood aggregation with gradients from all potential positive categories rather than from a single specific category (shown in Figure 1(b)). Also, the first term implicitly encourages category diversity over a batch to be uniformly distributed. Further, it can reduce the network's bias towards the dominating categories.

The second term tries equating the cross-correlation matrix's off-diagonal elements to 0. It decorrelates each unit of output prediction, pushing the output units to contain non-redundant information about the sample.

4.3 Instance-level Consistency Learning

Due to the domain shift problem, the semantic-level information from the classifier may not always be dependable. We further explore the finer knowledge in the form of instance-level information for neighborhood consistency learning as shown in Figure 2(d).

After feeding the target batches to the network, we obtain the feature embeddings of weakly-augmented view z_t^{wa} and strongly-augmented view z_t^{sa}. Similar to section 4.2, the weakly-augmented view is utilized for kNN retrieval and then neighborhood aggregation in the embedding-level, resulting in the weighted aggregated embedding $\hat{z}^{wa}$ from Eqn. 7.

Instead of using semantic-level information from the classifier, we explore instance-level classifiers with the help of source feature embeddings. Specifically, we utilize a memory bank to store and update the weakly-augmented view of source embeddings in $\mathbb{V}_s = \{z_{s,i}|_{i=1}^{N_s}\}$ in each iteration. Then we calculate the similarities between the strongly-augmented view of the given target instance z_t^{sa} and i-th source instance $\mathbf{z}_{s,i}$ from $\mathbb{V}_s$ by using a similarity function $sim(\cdot)$, which represents the dot product between L_2 normalized vectors $sim(\mathbf{u},\mathbf{v}) = \mathbf{u}^T\mathbf{v}/\|\mathbf{u}\|\|\mathbf{v}\|$. The calculated similarities can be processed using a softmax layer, which generates a distribution.:

$$q_{t,i}^{sa} = \frac{\exp(sim(\mathbf{z}_t^{sa}, \mathbf{z}_{s,i})/t)}{\sum_{k=1}^{N_s}\exp(sim(\mathbf{z}_t^{sa}, \mathbf{z}_{s,k})/t)} \tag{11}$$

Alternatively, we can compute the similarities between the weighted aggregated target embedding $\hat{z}_t^{wa}$ in Eqn. 7 and source embedding

$\mathbf{z}_i$ as $sim(\hat{z}^{wa}, \mathbf{z}_i)$. The obtained similarity distribution can be expressed as follows:

$$\hat{q}^{wa}_{t,i} = \frac{\exp(sim(\hat{z}^{wa}_t, \mathbf{z}_{s,i})/t)}{\sum_{k=1}^{N_s} \exp(sim(\hat{z}^{wa}_t, \mathbf{z}_{s,k})/t)} \tag{12}$$

where the temperature parameter t regulates the sharpness of the distribution. and we empirically set it to 0.1. Finally, the instance-level consistency regularization as shown in Figure 2 (d) can be achieved by minimizing the correlation matrix C^f between q^{sa}_t and $\hat{q}^{wa}_t$ to an identity matrix:

$$\mathcal{L}_{NCL-I} = \|C^f - \mathcal{I}\|^2, \tag{13}$$

$$\mathbf{C}^f = \frac{q^{sa}_{b,t} \cdot \hat{\boldsymbol{q}}^{wa}_{b,t}}{\sqrt{(q^{sa}_{b,t})^2}\sqrt{(\hat{\boldsymbol{q}}^{wa}_{b,t})^2}}, \tag{14}$$

where C^f is a square matrix with size the dimension N_s. Intuitively, Eqn.13 shares a similar form as Eqn.9 but uses a different classifier (instance-level) to calculate samples' predictions.

Overall, we aim at regularizing both feature embedding and output prediction invariant to neighborhood variations by pushing the diagonal elements of the cross-correlation matrix fixed at 1. Meanwhile, we encourage the feature embedding of a sample having the least redundancy by setting the off-diagonal elements of the cross-correlation matrix to 0

4.4 Overall optimization for NCL

To summarize, our overall optimization objective of NCL can be formulated as,

$$\mathcal{L}_{NCL} = \mathcal{L}^s_{CE} + \lambda(\mathcal{L}^t_{NCL-S} + \mathcal{L}^t_{NCL-I}), \tag{15}$$

where $\mathcal{L}^s_{CE}$ is the cross-entropy loss for source labeled data, $\mathcal{L}^t_{NCL-S}$ and $\mathcal{L}^t_{NCL-I}$ are the neighborhood consistency learning loss for target unlabeled data in semantic-level and instance-level respectively. λ is a hyperparameter to balance the optimization between source and target loss. The detailed optimization is shown in Algorithm 1.

Algorithm 1: Optimization for NCL

Input: a labeled source batch $\mathbf{x}_s$ and an unlabeled target batch $\mathbf{x}_t$. $T_w(\cdot)$ and $T_s(\cdot)$: Weak and strong augmentation function. $\mathcal{F}(\cdot)$ and $C(\cdot)$: encoder and classifier. $\mathbb{V}_s$ and $\mathbb{V}_t$: source and target memory bank.
Output: The model $\mathcal{F}(\cdot)$ and $C(\cdot)$.
for $t = 1, 2, \cdots, t_{all}$ **do**
  $\mathbf{z}^{wa}_s = \mathcal{F}(T_{wa}(\mathbf{x}_s))$ $\mathbf{p}^{wa}_s = C(\mathbf{z}^{wa}_s)$
  $\mathbf{z}^{wa}_t = \mathcal{F}(T_{wa}(\mathbf{x}_t))$ $\mathbf{p}^{wa}_t = C(\mathbf{z}^{wa}_t)$
  $\mathbf{z}^{sa}_t = \mathcal{F}(T_{sa}(\mathbf{x}_t))$ $\mathbf{p}^{sa}_t = C(\mathbf{z}^{sa}_t)$
  Compute $\hat{\mathbf{p}}^{wa}$ by Eq.7
  $\mathcal{L}_{NCL-S}$ with $\mathbf{p}^{sa}_t$ and $\hat{\mathbf{p}}^{wa}$ by Eq.9
  Compute $\hat{\mathbf{z}}^{wa}$ by Eq.6
  Compute $\mathbf{q}^{sa}_t$ and $\hat{\mathbf{q}}^{wa}_t$ by Eq.11 and Eq.12
  $\mathcal{L}_{NCL-I}$ with $\mathbf{q}^{sa}_t$ and $\hat{\mathbf{q}}^{wa}_t$ by Eq.13
  $\mathcal{L}_{NCL}$ by Eq.15
  Optimize $\mathcal{F}(\cdot)$ and $C(\cdot)$
  Update $\mathbb{V}_s$ and $\mathbb{V}_t$ with $\mathbf{z}^{wa}_{s/t}$ and $\mathbf{p}^{wa}_{s/t}$
end

5 EXPERIMENT

5.1 Datasets

We conduct experiments on three widely used domain adaptation classification benchmarks: Office-31 [26], Office-Home [32] and VisDA17 [25]. **Office-31** is a commonly used dataset for unsupervised domain adaptation. It includes 4652 images of 31 classes from three domains: Amazon (A), Webcam (W) and DSLR (D). **Office-Home** presents a more demanding benchmark compared to Office-31. It encompasses images of everyday objects categorized into four domains: artistic images (Ar), clip art (Cl), product images (Pr), and real-world images (Rw). The dataset comprises 15,500 images across 65 classes. **VisDA17**, on the other hand, is a large-scale dataset that employs 152,409 2D synthetic images from 12 classes as the source training set and 55,400 real images from MS-COCO as the target set. These domains share 12 object categories, making it suitable for domain adaptation tasks.

5.2 Implementation details

We adhere to the standard protocol of UDA [3, 35]), which involves utilizing all labeled source samples and all unlabeled target samples as training data. The reported testing results are the average accuracy over three random repeats with center-crop images. We adopt ResNet-50 [6] on Office-31 and Office-Home dataset and ResNet101 on VisDA17 dataset. The models are initialized with the ImageNet pre-trained weights. We use Pytorch as implementation framework. We adopt Stochastic Gradient Descent (SGD) optimizer with learning rate of 1×10^{-3}, weight decay 5×10^{-4}, momentum 0.9 and batch size 32. For optimization, we first pre-train the model based on source data only in Eqn 1 and initialize the memory bank for source and target samples. Then, we train our NCL framework based on Eqn. 15 with the hyperparameter λ set to 0.1 for all datasets. Also, we set the number of neighbors $k = 5$ for Eqn.6. For strong augmentation, we adopt the strategy in Fixmatch [27]. The results of existing methods in Table 1, 2, 3 refer to their respective papers.

5.3 Comparison with State-of-the-Arts

We compare our approach with the following baselines with state-of-the-art performance. (1) source-only baseline that trains the network using labeled source data and unlabeled target data respectively, (2) existing UDA methods based on discrepancy minimization, including CADA-P [9], ATM [11], RWOT [36] and DALN [1] , (3) existing UDA based on pseudo-labeling and neighborhood closeness, including CRST [40], SAFN [35], DTA [10], SHOT [14], MCC [8], BNM [2], CaCo [7] and ATDOC [15]. Note that not all the comparing methods report their results on all three benchmarks.

5.3.1 Results on Office-31. Results based on ResNet-50 are shown in Table 1. Notably, 1) comparing to state-of-the-art neighborhood PL methods (e.g, ATDOC [15]), NCL significantly outperforms them by 1.4% and boosts the performance substantially on difficult transfer tasks such as A→ W, D→ A, and W→A. It demonstrated our hypothesis that (1) preserving neighborhood structure of target data is important in PL-based methods. (2) Adding our consistency learning regularization in semantic-level and instance-level is effective to mitigate the misclassified neighborhood issue. Comparing

Table 1: Experiment results on Office-31 classification using ResNet-50. Best (bold red), second best (*italic blue*).

Method	A→D	A→W	D→A	D→W	W→A	W→D	Avg
ResNet-50 [6]	78.3	70.4	57.3	93.4	61.5	98.1	76.5
SAFN+ENT [35]	90.7	90.1	73.0	98.6	70.2	99.8	87.1
CaCo [7]	92.4	90.3	73.2	98.6	72.8	**100.**	86.5
CRST [40]	88.7	89.4	72.6	98.9	70.9	**100.**	86.8
SHOT [14]	94.0	90.1	74.7	98.4	74.3	99.9	88.6
CADA-P [9]	95.6	**97.0**	71.5	**99.3**	73.1	**100.**	89.5
ATM [11]	**96.4**	95.7	74.1	**99.3**	73.5	**100.**	89.8
MCC [8]	92.1	94.0	74.9	98.5	75.3	**100.**	89.1
BNM [2]	92.2	94.0	74.9	98.5	75.3	**100.**	89.2
ATDOC [15]	94.4	94.3	75.6	98.9	75.2	99.6	89.7
DALN [1]	95.4	95.2	76.4	*99.1*	*76.5*	**100.**	*90.4*
NCL (ours)	*96.3*	*96.6*	**77.6**	98.7	**77.4**	**100.**	**91.1**

Table 2: Experiment results on Office-Home using ResNet-50.

Method	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
ResNet-50 [6]	44.9	66.3	74.3	51.8	61.9	63.6	52.4	39.1	71.2	63.8	45.9	77.2	59.4
SAFN [35]	52.0	71.7	76.3	64.2	69.9	71.9	63.7	51.4	77.1	70.9	57.1	81.5	67.3
CADA-P [9]	56.9	76.4	80.7	61.3	75.2	75.2	63.2	54.5	80.7	*73.9*	**61.5**	84.1	70.2
DCAN [13]	54.5	75.7	81.2	67.4	74.0	76.3	**67.4**	52.7	80.6	**74.1**	59.1	83.5	70.5
SHOT [14]	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
BNM [2]	56.7	77.5	81.0	67.3	76.3	77.1	65.3	55.1	82.0	73.6	57.0	84.3	71.1
MCC [8]	56.3	77.3	80.3	67.0	77.1	77.0	66.2	55.1	81.2	73.5	57.4	84.1	71.0
ATDOC [15]	*58.3*	*78.8*	*82.3*	**69.4**	*78.2*	*78.2*	67.1	*56.0*	**82.7**	72.0	58.2	*85.5*	*72.2*
DALN [1]	57.8	**79.9**	82.0	66.3	76.2	77.2	66.7	55.5	81.3	73.5	*60.4*	85.3	71.8
NCL (ours)	**58.9**	78.6	**82.6**	*69.2*	**79.4**	**78.6**	67.2	**57.1**	*82.3*	73.1	58.7	**85.6**	**72.6**

Table 3: Experimental results on VisDA17 classification using ResNet-101.

Method	Aero	Bike	Bus	Car	Horse	Knife	Motor	Person	Plant	Skateboard	Train	Truck	Mean
ResNet-101 [6]	67.7	27.4	50.0	61.7	69.5	13.7	85.9	11.5	64.4	34.4	84.2	19.2	49.1
MinEnt [4]	88.6	29.5	82.5	75.8	88.7	16.0	**93.2**	63.4	94.2	40.1	87.3	12.1	64.3
SAFN [35]	93.6	61.3	84.1	70.6	*94.1*	79.0	91.8	79.6	89.9	55.6	89.0	24.4	76.1
CRST [40]	88.0	79.2	61.0	60.0	87.5	81.4	86.3	78.8	85.6	86.6	73.9	**68.8**	78.1
DTA [10]	93.7	82.2	*85.6*	*83.8*	93.0	81.0	90.7	*82.1*	**95.1**	78.1	86.4	32.1	81.5
SHOT [14]	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
RWOT [36]	*95.1*	80.3	83.7	**90.0**	92.4	68.0	*92.5*	**82.2**	87.9	78.4	**90.4**	*68.2*	*84.0*
BNM [2]	91.1	69.0	76.7	64.3	89.8	61.2	90.8	74.8	90.9	66.6	88.1	46.1	75.8
MCC [8]	92.2	82.9	76.8	66.6	90.9	78.5	87.9	73.8	90.1	76.1	87.1	41.0	78.7
ATDOC-NA [15]	93.7	83.0	76.9	58.7	89.7	95.1	84.4	71.4	89.4	80.0	86.7	55.1	80.3
NCL (ours)	**97.1**	**88.5**	**90.0**	65.2	**96.7**	*92.9*	90.1	81.5	*94.6*	**89.5**	*89.0*	58.8	**86.2**

to the recent discrepancy-based methods such as DALN [1], NCL also achieves a performance boost by 0.7%.

5.3.2 Results on Office-Home. Result based on ResNet-50 are reported in Table 2. Similarly conclusion can be drawn that NCL shows consistent improvements on different discrepancy-based methods such as DALN [1] and PL-based methods such as SHOT [14], MCC [8], BNM [2], CaCo [7] and ATDOC [15].

5.3.3 Results on VisDA. Result based on ResNet-101 are reported in Table 3. In more challenging large scale benchmark, NCL still shows its consistent performance gains over previous state-of-the-art. Comparing to other consistency regularization method such as DTA [10], NCL shows a performance gain by 2.2%. Compared to the neighborhood PL methods, NCL significantly outperforms ATDOC [15] by 5.9 %. It is also worth noting that our NCL significantly boost the performance on several challenging categories such as Bus (+ 13.1 %), Car (+ 6.5 %) and Person (+ 10.1 %).

5.4 Analysis

5.4.1 Feature visualization. We visualize the target embeddings of (a) source model, (b) Neighborhood PL [15], and (c) NCL on Office-31 W→A via t-SNE [21] in Fig.4(a-c). We qualitatively observe that NCL could learn more discriminative and compact feature clusters than the source model and PL-based methods.

5.4.2 Visualization on the Retrieval Neighbors. We visualize the top 3 nearest neighbors given an anchor based on the target features from the source model on Office-31 A → W in Fig.3. We investigate both success and failure cases to get extra insights into our method.

For the first two-row, features from the source model could retrieve the correct neighbors for the mobile phone and backpack with a certain level of appearance and pose variations. However, in the last row where the pose of that laptop sample is unusual, our method might fail in those cases. The proposed uncertainty-aware weighting on neighbor class consistency is to alleviate the misleading learning by false neighborhood supervision.

5.4.3 How does uncertainty-aware weighting look like? To prove our hypothesis that negative neighbors are prone to have high entropy value (small weight), we compute the pairwise weight from Eqn. 8 for all the neighbors (k=2) on Office-31 A → W and W → A and take an average over weights on positive pairs and negative pairs respectively. From Fig.4 (f), we can see that positive neighbor pairs have much higher weights during neighborhood aggregation than negative pairs by 20% on average. *It demonstrates our hypothesis that uncertainty-aware weighting could alleviate the false positive neighbors via assigning smaller weight in neighborhood aggregation.*

5.4.4 Impact of neighbourhood size k. Neighborhood size k is an important parameter as it controls the amount of pairwise neighborhood supervision. However, there is a trade-off between increasing the neighboring diversity and increasing the risk of adding false neighborhood supervision. We evaluate k from {2, 5, 10} for our NCL objective in Eqn. 15 on Office-31 A → W as shown in Fig.4(d). We call our method with equally weighted neighborhood aggregation as NCL_{avg} and our method with uncertainty-aware neighborhood aggregation as NCL_{weight}. We observe that NCL_{weight} is more robust to larger neighborhood size than NCL_{avg}. This is also consistent with the intuition of uncertainty-aware weighting, which aims to alleviate the misleading effect of false positive neighbors. Based on the experiments, we empirically set the neighborhood size k to 5 for all the experiments, while our method is generally robust to the k.

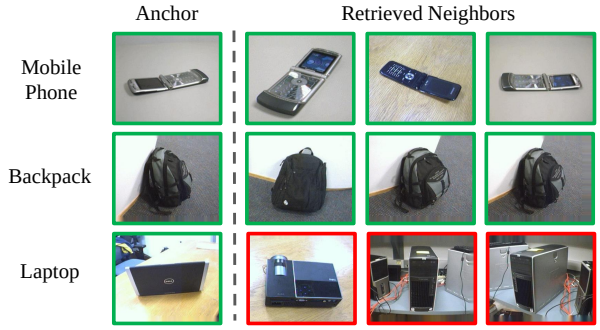

Figure 3: The top 3 nearest neighbors are given an anchor on Office-31. (Green: positive sample; Red: negative sample).

5.4.5 Hyper-parameter sensitivity. We conduct the hyper-parameter sensitivity analysis on the λ in Eqn. 15 on Office31 A → W. As Figure 4(e) shows, the performance is significantly improved when $\lambda > 0$. It illustrates that our method is robust to the wide range of λ. Note that we selected λ based on Office-31 and used these unified hyper-parameters for all other datasets (e.g office-home and VisDA17) without extra tuning. The consistent performance gains on other datasets prove that our method is robust to this hyperparameter.

Table 4: Ablation Study on office-31 for strong/weak augmentation.

$\mathcal{L}_{NCL-S}$	$\mathcal{L}_{NCL-I}$	w	Strong Aug.	**Office-31**
✗	✗	✗	✓	76.1
✓	✗	✗	✓	89.5
✓	✗	✓	✓	90.6
✗	✓	✗	✓	89.3
✗	✓	✓	✓	90.1
✓	✓	✓	✗	90.4
✓	✓	✓	✓	**91.1**

5.4.6 Ablation study on each component. We conduct ablation study to verify the effectiveness of each components of NCL on office-31 dataset in Table 4. First, we observe that neighborhood consistency learning significantly boosts the performance on the baseline in both semantic-level (+ 13.4 %) or instance-level (+ 13.2 %). Further, adding uncertainty-aware neighborhood aggregation benefits both two consistency regularization. Finally, incorporating all the components for optimization achieves the best performance with 91.1 %. Specifically, the weak/strong augmentations could improve our model with weak/weak augmentations from 90.4 % to 91.1 %. We claim that weak augmentation is more stable and thus is used for k-NN retrieval and neighborhood aggregation. Strong augmentation is to increase the sample variety for consistency learning. Note that we set the neighborhood size k to 3 in the ablation study.

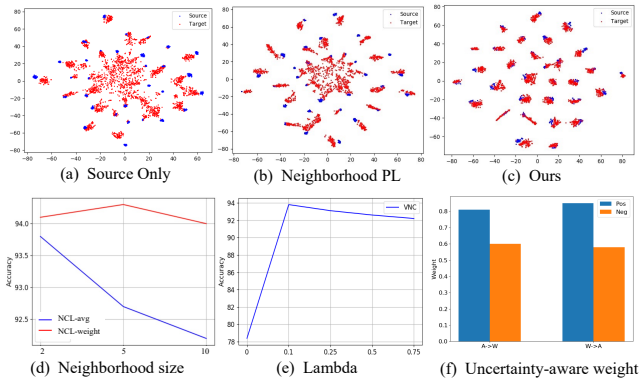

Figure 4: Analysis: (a-c) The t-SNE visualization of target feature (red) on Office-31 W→A. (d-e) Hyper-parameter sensitivity experiments on Office-31 A→W. (f) Uncertainty-aware weight visualization on positive and negative samples on Office-31 A→W and W →A.

6 CONCLUSION

In this work, we propose a novel neighborhood consistency learning framework at both the semantic-level and instance level with a new consistency objective function. Specifically, by computing the correlation matrix between the target samples and their neighborhood aggregation, and matching the correlation matrix to an identity matrix, our objective function encourages semantic consistency in the target neighborhood. Extensive experiments on three UDA benchmarks demonstrate the effectiveness of our method.

7 ACKNOWLEDGEMENT

This material is based upon work supported by the Air Force Office of Scientific Research under award number FA9550-23-1-0290.

REFERENCES

[1] Lin Chen, Huaian Chen, Zhixiang Wei, Xin Jin, Xiao Tan, Yi Jin, and Enhong Chen. 2022. Reusing the Task-specific Classifier as a Discriminator: Discriminator-free Adversarial Domain Adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7181–7190.

[2] Shuhao Cui, Shuhui Wang, Junbao Zhuo, Liang Li, Qingming Huang, and Qi Tian. 2020. Towards Discriminability and Diversity: Batch Nuclear-norm Maximization under Label Insufficient Situations. In *Proc. CVPR*. 3941–3950.

[3] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research* 17, 1 (2016), 2096–2030.

[4] Yves Grandvalet and Yoshua Bengio. 2005. Semi-supervised learning by entropy minimization. In *Advances in Neural Information Processing Systems*. 529–536.

[5] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition*. 9729–9738.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*. 770–778.

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

[8] Ying Jin, Ximei Wang, Mingsheng Long, and Jianmin Wang. 2020. Minimum Class Confusion for Versatile Domain Adaptation. In *Proc. ECCV*. 464–480.

[9] Vinod Kumar Kurmi, Shanu Kumar, and Vinay P Namboodiri. 2019. Attending to discriminative certainty for domain adaptation. In *Proc. CVPR*. 491–500.

[10] Seungmin Lee, Dongwan Kim, Namil Kim, and Seong-Gyun Jeong. 2019. Drop to adapt: Learning discriminative features for unsupervised domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*. 91–100.

[11] Jingjing Li, Erpeng Chen, Ding Zhengming, Lei Zhu, Ke Lu, and Heng Tao Shen. 2020. Maximum Density Divergence for Domain Adaptation. *IEEE Trans. Pattern Anal. Mach. Intell.* (2020), 1–1.

[12] Kai Li, Chang Liu, Handong Zhao, Yulun Zhang, and Yun Fu. 2021. ECACL: A Holistic Framework for Semi-Supervised Domain Adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

[13] Shuang Li, Chi Harold Liu, Qiuxia Lin, Binhui Xie, Zhengming Ding, Gao Huang, and Jian Tang. 2020. Domain Conditioned Adaptation Network. In *Proc. AAAI*. 11386–11393.

[14] Jian Liang, Dapeng Hu, and Jiashi Feng. 2020. Do We Really Need to Access the Source Data? Source Hypothesis Transfer for Unsupervised Domain Adaptation. In *Proc. ICML*. 6028–6039.

[15] Jian Liang, Dapeng Hu, and Jiashi Feng. 2021. Domain Adaptation with Auxiliary Target Domain-Oriented Classifier. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 16632–16642.

[16] Chang Liu, Kunpeng Li, Michael Stopa, Jun Amano, and Yun Fu. 2023. Discovering Informative and Robust Positives for Video Domain Adaptation. In *The Eleventh International Conference on Learning Representations*. https://openreview.net/forum?id=vk-j5pQY3Gv

[17] Chang Liu, Lichen Wang, and Yun Fu. [n. d.]. *Meta Adversarial Weight for Unsupervised Domain Adaptation*. 10–18. https://doi.org/10.1137/1.9781611977172.2 arXiv:https://epubs.siam.org/doi/pdf/10.1137/1.9781611977172.2

[18] Chang Liu, Lichen Wang, Kai Li, and Yun Fu. 2021. *Domain Generalization via Feature Variation Decorrelation*. Association for Computing Machinery, 1683–1691. https://doi.org/10.1145/3474085.3475311

[19] Chang Liu, Xiang Yu, Yi-Hsuan Tsai, Masoud Faraki, Ramin Moslemi, Manmohan Chandraker, and Yun Fu. 2022. Learning to learn across diverse data biases in deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4072–4082.

[20] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. 2018. Conditional adversarial domain adaptation. In *Advances in Neural Information Processing Systems*. 1640–1650.

[21] Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research* 9, Nov (2008), 2579–2605.

[22] Gaurav Mittal, Chang Liu, Nikolaos Karianakis, Victor Fragoso, Mei Chen, and Yun Fu. 2020. HyperSTAR: Task-Aware Hyperparameters for Deep Networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8736–8745.

[23] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. 2018. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41, 8 (2018), 1979–1993.

[24] Yingwei Pan, Ting Yao, Yehao Li, Yu Wang, Chong-Wah Ngo, and Tao Mei. 2019. Transferrable prototypical networks for unsupervised domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition*. 2239–2247.

[25] Xingchao Peng, Ben Usman, Neela Kaushik, Dequan Wang, Judy Hoffman, Kate Saenko, Xavier Roynard, Jean-Emmanuel Deschaud, Francois Goulette, Tyler L Hayes, et al. 2018. VisDA: A Synthetic-to-Real Benchmark for Visual Domain Adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*.

[26] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. 2010. Adapting visual category models to new domains. In *European Conference on Computer Vision*.

[27] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. 2020. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in neural information processing systems* 33 (2020), 596–608.

[28] Hui Tang, Ke Chen, and Kui Jia. 2020. Unsupervised domain adaptation via structurally regularized deep clustering. In *IEEE Conference on Computer Vision and Pattern Recognition*. 8725–8735.

[29] Antti Tarvainen and Harri Valpola. 2017. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems* 30 (2017).

[30] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. 2017. Adversarial discriminative domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition*. 7167–7176.

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

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

[33] Qian Wang and Toby Breckon. 2020. Unsupervised domain adaptation via structured prediction based selective pseudo-labeling. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 34. 6243–6250.

[34] Shaoan Xie, Zibin Zheng, Liang Chen, and Chuan Chen. 2018. Learning semantic representations for unsupervised domain adaptation. In *International Conference on Machine Learning*. 5423–5432.

[35] Ruijia Xu, Guanbin Li, Jihan Yang, and Liang Lin. 2019. Larger norm more transferable: An adaptive feature norm approach for unsupervised domain adaptation. In *IEEE International Conference on Computer Vision*. 1426–1435.

[36] Renjun Xu, Pelen Liu, Liyan Wang, Chao Chen, and Jindong Wang. 2020. Reliable Weighted Optimal Transport for Unsupervised Domain Adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition*. 4394–4403.

[37] Shiqi Yang, Yaxing Wang, Joost van de Weijer, Luis Herranz, and Shangling Jui. 2021. Generalized source-free domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 8978–8987.

[38] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. 2021. Barlow twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning*. PMLR, 12310–12320.

[39] Yabin Zhang, Bin Deng, Kui Jia, and Lei Zhang. 2020. Label Propagation with Augmented Anchors: A Simple Semi-Supervised Learning baseline for Unsupervised Domain Adaptation. In *European Conference on Computer Vision*. Springer, 781–797.

[40] Yang Zou, Zhiding Yu, Xiaofeng Liu, BVK Kumar, and Jinsong Wang. 2019. Confidence regularized self-training. In *IEEE International Conference on Computer Vision*. 5982–5991.