This ACCV 2022 paper, provided here by the Computer Vision Foundation, is the author-created version.
The content of this paper is identical to the content of the officially published ACCV 2022
LNCS version of the paper as available on SpringerLink: https://link.springer.com/conference/accv

Revisiting Unsupervised Domain Adaptation Models: a Smoothness Perspective

Xiaodong Wang[1], Junbao Zhuo[2⋆], Mengru Zhang[3], Shuhui Wang[2], and Yuejian Fang[1⋆]

[1] School of Software & Microelectronics, Peking University
[2] Key Lab of Intell. Info. Process., Inst. of Comput. Tech., CAS
[3] MEGVII Technology
{wangxiaodong21s@stu,fangyj@ss}.pku.edu.cn,
{junbao.zhuo@vipl.,wangshuhui@}ict.ac.cn, zhangmengru@megvii.com

Abstract. Unsupervised Domain Adaptation (UDA) aims to leverage the labeled source data and unlabeled target data to generalize better in the target domain. UDA methods utilize better domain alignment or carefully-designed regularizations to increase the discriminability of target features. However, most methods focus on directly increasing the distance between cluster centers of target features, i.e., enlarging inter-class variance, which intuitively increases the discriminability of target features and is easy to implement. However, due to intra-class variance optimization being under-explored, there are still some samples of the same class are prone to be classified into several classes. To handle this problem, we aim to equip UDA methods with the high smoothness constraint. We first define the model's smoothness as the predictions similarity within each class, and propose a simple yet effective technique LeCo (imp**L**icit smooth**n**ess **Co**nstraint) to promote the smoothness. We construct the weak and strong "views" of each target sample and enforce the model predictions of these two views to be consistent. Besides, a new uncertainty measure named Instance Class Confusion conditions the consistency is proposed to guarantee the transferability. LeCo implicitly reduces the model sensitivity to perturbations for target samples and guarantees smaller intra-class variance. Extensive experiments show that the proposed technique improves various baseline approaches by a large margin, and helps yield comparable results to the state-of-the-arts on four public datasets. Our codes are publicly available at https://github.com/Wang-Xiaodong1899/LeCo_UDA.

1 Introduction

Deep learning methods have achieved great success on a wide variety of tasks, and show surprising performances even without labels [1,2,3]. However, when the training set and test set are drawn from different data distributions, the deep learning models would usually have poor generalization performance on the test set. To handle this problem, the Unsupervised Domain Adaptation (UDA) [5,6,7]

⋆ Corresponding author

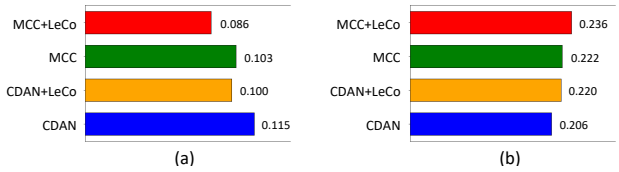


Fig. 1: Illustration of (a) the intra-class variance, and (b) the inter-class variance on VisDA-C [4]. Different colors indicate the results of different models.

technique was proposed. In the UDA scenario, the model is trained on a labeled source domain (training set) and an unlabeled target domain (test set) and required to perform well in the target domain, where the source and target distributions follow the Covariate Shift [8].

Recently, deep learning based UDA methods have almost dominated this field with promising results [9,10,11,12,13,14,15,16]. One direction is to learn domain-invariant feature. Methods [12,14] imposed adversarial training for better domain distribution alignment, and [15,16] employed the bidirectional alignment. The other direction is to add specific regularization items on the target data, which could obtain a striking performance [9,10,11] as the properties of target data are well exploited. For example, [9,10] added regularizations on classification responses for insuring larger prediction diversity or small prediction confusion. However, existing methods did not alleviate error accumulation well, and sometimes, the prediction confidence of wrongly classified examples also increases during training. Such accumulation of misclassification harms the learning process and affects those correctly classified examples with low prediction confidence. The above issue is very common in UDA methods, and recent methods like [15,16] expect to alleviate the accumulation problem by the better feature matching, or by using a target domain-oriented classifier [17] to generate more accurate pseudo labels. However, they relied on better models and did not explicitly reduce the sensitivity of models to sample disturbances, and are not generic.

In this paper, we revisit the UDA models in a new perspective: *smoothness*. Given some samples within a class or the augmented data of a sample, the model predictions should be similar. We define smoothness as the intra-class variance, which is a measure of how far a set of features (predictions) within a class is spread out from their average value. Obviously, the intra-class variance will be smaller if the predictions of the same class are more consistent. As shown in Fig 1, compared with the CDAN [14], MCC [9] shows smaller intra-class variance and larger inter-class variance. Besides, when equipped with high smoothness, they both show smaller intra-class and larger inter-class variances.

However, smoothness is hard to insured in UDA, as the labels of the target samples are not available. Inspired by semi-supervised learning methods [18,19] or self-supervised methods [1,2], we proposed a technique LeCo (imp**L**icit smoothn**e**ss **Co**nstraint). The key is to create new strongly perturbed samples which come

from the same instance (of course the same class) as original samples of the target domain, so construct a large number of such pairs to estimate and reduce the intra-class variance, insuring the high smoothness. Specifically, two "views" of the same target sample under the weak and strong augmentations pass the same network to generate the two predictions, and we minimize the $L2$-distance of them on all class predictions, which is regarded as *naïve constraint*.

Furthermore, avoiding the hard-to-transfer samples to deteriorate the optimization, we quantify the uncertainty of samples using a novel *Instance Class Confusion* to condition the constraint. Instance class confusion of a certain sample is defined as the sum of all cross-class confusion by the class predictions of it. We consider the cross-class information and the probabilities of all classes, which is better to measure the transferability than only considering the probabilities of all classes [20]. We utilize the instance class confusion to condition the naïve constraint so that we can achieve a better training convergence for domain adaptation. Additionally, we provide theoretical analysis which reveals that our technique could approximate the more expected target risk.

We finally summarize our contributions as follows:

- We analyze the existing UDA methods with a novel perspective: **smoothness**. We introduce the reciprocal of intra-class variance as the indicator of smoothness. To promote the smoothness of models, we propose a simple yet effective technique named LeCo.
- LeCo could implicitly encourage the model to generate consistent predictions on the target domain. It is generic and can be applied to various UDA methods, reducing the intra-class variance effectively, also increasing the inter-class variance.
- We validate the effectiveness of LeCo on the image classification task of UDA. Extensive experiments demonstrate the effectiveness and we achieve results comparable to the state-of-the-arts on four public datasets.

2 Related Work

Domain adaptation aims to transfer source domain knowledge to the related but different target domain, and there are various settings in this field, such as Unsupervised Domain Adaptation [12,21,22], Semi-Supervised Domain Adaptation [23,24,25,26], Model Adaptation [27,28,29], Noisy Domain Adaptation [30,31] etc. Most of the works focus on UDA which is adopted in this paper. We also review regularization based methods and consistency learning based methods.

Unsupervised Domain Adaptation: The deep unsupervised domain adaptation methods have made a success without any labels in the target domain. These methods can be mainly divided into domain alignment methods and regularization-based methods. For the domain alignment methods, the early methods [21,22,32,33,34] are based on feature distribution matching. MMD [35] is often used [21,22] in the deep neural network to deal with the domain adaptation by aligning the distribution. For better aligning the distributions, JAN [33] considered the joint feature distributions, and CMD [34] proposed the new domain

discrepancy metric. Due to the potential of GANs [36], various works [12,14,37,38,39] performed better domain alignment by using adversarial learning. DANN [37] firstly designed a novel adversarial pipeline. It imposed a domain classifier, and used adversarial training to learn domain-invariant representations. CDAN [14] conducted adversarial learning on the covariance of feature representations and classifier predictions. MCD [38] explicitly utilized two task-specific classifiers to measure the domain discrepancy and minimized it according to the $\mathcal{H}\Delta\mathcal{H}$ theory [6]. MDD [12] proposed a novel margin disparity discrepancy that firstly leveraged the scoring function and margin loss to bound the gap caused by domain shift. Recently, the bidirectional domain matching methods [15,16] exploited a novel and effective domain alignment strategy. Method [15] utilized mixup to augment intermediate domains for the bidirectional matching, while method [16] constructed virtual mirrors for both source and target domains.

Regularization Based Methods: Inspired by Semi-Supervised Learning (SSL), researchers [9,10,11,13,40] are more interested in exploring target data properties in UDA. Early, Entropy Minimization (EntMin) [20] is widely used in UDA and semi-supervised domain adaptation [14,28]. In recent years, some regularization based methods are proposed to hold the discriminability of the target domain. AFN [11] investigated that those task-specific features with larger norms are more transferable. BNM [10] proved that the batch nuclear-norm maximization can lead to the improvement on both the prediction discriminability and diversity, which works well in domain adaptation, SSL, and open domain recognition [41]. In [42], the authors proposed domain conditioned adaptation network, with a designed domain conditioned channel attention module to excite channel activation separately for each domain. In [43], a transferable semantic augmentation approach was proposed to enhance the classifier adaptation ability via implicitly generating source features toward target semantics. MCC [9] can be regarded as a regularization-based method that restrains the inter-class confusion of unlabeled data. These techniques can be considered as the self-training of target data, cooperating well with source supervised training. The regularization can also be the consistency regularization in self-ensembling [13,44] and achieved superior performance on VisDA-C [4].

Consistency Learning: We review related consistency learning methods in SSL and self-supervised learning. Consistency learning [44,45,46] or data augmentation [18,19] methods that have achieved good performance in SSL, as well as regularization methods [20,47]. Temporal ensembling [47] by constraining the consistency of different training epochs enabled the model to better learn unlabeled data. In [48], authors conceptually explored the regularization methods by comparing the gradient norms between regularization loss and cross-entropy loss in SSL. Our constraint is similar to FixMatch [18] using different augmentations [1,2], but we exploits all class predictions even the low confidence information which is shown to be effective for domain adaptation [15,49]. Our method utilizes all class predictions to maintain the consistency of two augmentations' predictions, rather than using the confidence thresholding. Nowadays, self-supervised learning makes a remarkable success due to large-scale data and

instance discrimination learning. Contrastive learning methods [1,2] utilized two random augmentations for images and encouraged predictions to be consistent, which aim to learn the instance discrimination. Domain adaptation can be considered as a special case of SSL, where labeled and unlabeled data are drawn from different distributions. Methods in SSL aim to learn the consistency in a single domain, and the sparse labels or pseudo labels share the same domain with unlabeled data, whereas UDA is a cross-domain task. We focus on the intra-class consistency, which is related to the intra-domain consistency adopted in [50]. [50] attempted to use primary and auxiliary classifiers that share the same feature extractor to force the model to generate more similar predictions. We first use different augmentations to construct the inconsistency for each instance and reduce it, and then we use instance class confusion to guide the model to focus on more reliable instances which will be helpful for cross-domain training.

3 method

3.1 Preliminaries

In UDA, we are given source domain data $\mathcal{D}_s = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$ of n_s labeled samples from $\mathcal{X}^s \times \mathcal{Y}^s$ and target domain data $\mathcal{D}_t = \{(x_i^t)\}_{i=1}^{n_t}$ of n_t unlabeled samples from $\mathcal{X}^t$. The two domains share the same K categories and their distributions follow the Covariate Shift [8]. Specifically, the input marginal distribution $P(\mathcal{X})$ changes ($P(\mathcal{X}^s) \neq P(\mathcal{X}^t)$) but the conditional $P(\mathcal{Y}|\mathcal{X})$ remains the same.

The model in our method is equipped with a feature extractor and a classifier. Here, the feature extractor ψ consists of the deep convolution networks and a bottleneck layer that is introduced to reduce the dimension of features, and the features are passed through the classifier f to generate predictions.

3.2 Recap of UDA Baselines

Domain-alignment Methods. Explicit domain alignment methods [22,33] use the discrepancy metrics such as [34,35]. To better align the source and target domains, methods [12,14,37] utilize the adversarial training. Recently, bidirectional alignment shows great performances [15,16] in this field. These methods try to simultaneously optimize the source classification loss $\mathcal{L}_s$ and domain alignment loss $\mathcal{L}_{dom}$. First, using the randomly sampled batch labeled examples $\{X^s, Y^s\}$ from the source domain $\mathcal{D}_s$ of size B, we can obtain the supervised classification objective:

$$\mathcal{L}_s = \frac{1}{B} \sum_{i=1}^{B} CE(Y_i^s, f(\psi(X_i^s))) \tag{1}$$

where the $CE(\cdot, \cdot)$ is the cross-entropy loss. As for the domain alignment methods using the discrepancy metrics such as the MMD [35], given the randomly sampled a batch of labeled examples $\{X^t\}$ from the target domain $\mathcal{D}_t$, the domain alignment loss $\mathcal{L}_{dom}$ is defined as below:

$$\mathcal{L}_{dom} = MMD(\psi(X^s), \psi(X^t)) \tag{2}$$

where $MMD(\cdot,\cdot)$ matches the feature distributions of $\mathcal{D}_s$ and $\mathcal{D}_t$. And the general optimization objective can be formulated as:

$$\min_{\psi,f} \mathcal{L}_s + \mathcal{L}_{dom} \tag{3}$$

As for the adversarial methods that introduce a domain discriminator D, if the inputs of D are the features, the domain alignment loss could be defined as below:

$$\mathcal{L}_{dom} = \frac{1}{B}\sum_{i=1}^{B} log(D(\psi(X_i^s))) + \frac{1}{B}\sum_{i=1}^{B} log(1 - D(\psi(X_i^t))) \tag{4}$$

And the optimization objective is formulated as:

$$\begin{aligned} &\min_{\psi,f} \mathcal{L}_s + \mathcal{L}_{dom} \\ &\max_{D} \mathcal{L}_{dom} \end{aligned} \tag{5}$$

Regularization-based Methods. The properties of the target domain can be modeled by some regularization methods [9,10,11,20], which aim to enhance the discriminability of target samples. We denote the $\mathcal{L}_{reg}$ as the regularization loss in these methods. And $\mathcal{L}_{reg}$ often only depends on target features, which is defined as below:

$$\mathcal{L}_{reg} = \frac{1}{B}\sum_{i=1}^{B} \left\|\psi(X_i^t)\right\|_* \tag{6}$$

where the symbol $*$ can be replaced by the specific norm such as adaptive feature norm [11], nuclear norm [10], entropy [20], and class confusion [9]. And the general optimization objective in regularization based methods is formulated as:

$$\min_{\psi,f} \mathcal{L}_s + \mathcal{L}_{reg} \tag{7}$$

In this paper, we aim to equip both domain-alignment and regularization-based UDA methods with an effective technique about smoothness constraint. Our method can bring remarkable improvements to these methods.

3.3 Implicit Smoothness Constraint

We noticed that the prediction intra-class variance is relatively large, i.e. lower smoothness. The core problem is that the model is sensitive to image perturbations in the target domain, so it is prone to misclassify images of the same class to different classes. Therefore, to reduce the sensitivity of the model, we directly impose the model to generate more consistent predictions from original and perturbed images. Specifically, we impose the constraint that the predictions of weak and strong augmentations should be close enough. Although the idea is simple, it can effectively contribute to more consistent predictions.

We aim to enhance the smoothness of the domain alignment and regularization based models. Given the randomly sampled batch of unlabeled examples

$\{X^t\}$ from target domain $\mathcal{D}_t$, we take two "views" of these examples by the weak augmentation (random flipping and cropping) and strong augmentation (RandAug [51]). We denote the weak and strong views as X_w^t and X_{str}^t, respectively.

Then, both the two views are passed through the feature extractor ψ and classifier f to generate the classification responses, respectively, as follows:

$$\hat{Y}_w^t = \sigma\left(f(\psi(X_w^t))\right), \hat{Y}_{str}^t = \sigma\left(f(\psi(X_{str}^t))\right) \tag{8}$$

where $\sigma(\cdot)$ is the softmax function. For each $x^t \in X^t$, we have the corresponding classification responses $\hat{y}_w^t \in \hat{Y}_w^t$, and $\hat{y}_{str}^t \in \hat{Y}_{str}^t$. The difference between $\hat{y}_w^t$ and $\hat{y}_{str}^t$ can be measured as below:

$$d(x^t) = \frac{1}{K}\left\|\hat{y}_w^t - \hat{y}_{str}^t\right\|_2^2 \tag{9}$$

where the $\|\cdot\|_2$ denotes L2-distance under K classes. Here we make full use of the low confident probabilities of all classes, rather than the most confident probability [18] to teach the strong one. We claim that more information including low confidence information [15,49] could promote the model transferability. Although we can not access the target labels, this can be regarded as an implicit smoothness constraint. Then, we can define the *naïve constraint* loss:

$$\mathcal{L}_{nc} = \frac{1}{B}\sum_{x^t \in X^t} d(x^t). \tag{10}$$

If we impose equal importance on all samples, we obtain the average on the sample level. Then, we can introduce the naïve constraint loss to both domain-alignment and regularization-based methods as follows,

$$\begin{aligned} &\min_{\psi,f}\ \mathcal{L}_s + \mathcal{L}_{dom} + \lambda\mathcal{L}_{nc} \\ &\min_{\psi,f}\ \mathcal{L}_s + \mathcal{L}_{reg} + \lambda\mathcal{L}_{nc}, \end{aligned} \tag{11}$$

where λ denotes the tradeoff parameter.

Confusion Conditioning. The naïve constraint imposes equal importance for different samples. However, samples are not equally important, enforcing the optimization on all samples may harm the training convergence. Some methods [14,9] noticed this issue and utilized the entropy of samples to reweight the loss function. We should pay more attention to reliable samples, and noticed that the class confusion of a sample can be a good measure of the uncertainty of it. Different from the technique in [9], we model the class confusion on instance level, and then propose the *Instance Class Confusion*.

Given a target sample $x^t \in X^t$ and the corresponding classification response $\hat{y}^t \in \hat{Y}^t$, the class confusion matrix of x^t is defined as below:

$$M = \hat{y}^t \times \hat{y}^{t^\top}, \tag{12}$$

where $M \in \mathcal{R}^{K\times K}$. It is possible to walk from one class to another for each sample when it is easy to be misclassified, often asymmetrically, and we investigated

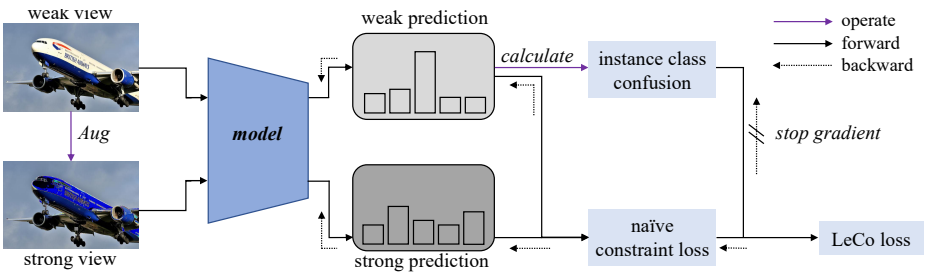


Fig. 2: Illustration of proposed LeCo framework.

that this category-normalization technique is the key to MCC [9]. Following MCC, the class confusion between class i and j is normalized as follows:

$$\tilde{M}_{i,j} = \frac{M_{i,j}}{\sum_{j'=1}^{K} M_{i,j'}}. \tag{13}$$

Since normalized, it contains comprehensive information of confidence and cross-class, better reflecting the uncertainty than only considering the confidence such as entropy [20]. The *Instance Class Confusion* of a sample x^t is as below:

$$I(x^t) = \sum_{i=1}^{K} \sum_{j \neq i}^{K} \tilde{M}_{i,j}. \tag{14}$$

The lower instance class confusion indicates the lower uncertainty of this sample. We assume that discarding unlabeled samples with too low confidence will not lead to information loss but reduce the learning pressure of the model. So we condition the loss in Eq (10) as below:

$$\mathcal{L}_{leco} = \frac{1}{B} \sum_{x^t \in X^t} \mathbb{1}(I(x^t) < \tau) d(x^t), \tag{15}$$

where $\mathbb{1}$ is the indicator function, and τ is the uncertainty threshold. We do not use the fixed threshold, because it can not properly reflect the changing uncertainty during training. We adopt an adaptive τ using the mean of the instance class confusion of mini-batch target samples. Finally, the proposed LeCo loss is pluggable to domain alignment and regularization based methods as follows:

$$\begin{aligned} &\min_{\psi,f} \ \mathcal{L}_s + \mathcal{L}_{dom} + \lambda \mathcal{L}_{leco} \\ &\min_{\psi,f} \ \mathcal{L}_s + \mathcal{L}_{reg} + \lambda \mathcal{L}_{leco}. \end{aligned} \tag{16}$$

The illustration of proposed LeCo framework is shown clearly in Fig 2.

4 Theoretical Guarantees

We present the theoretical analysis for the implicit smoothness constraint following the theory of [6]. Let $\mathcal{H}$ be the hypothesis class, given the source and target distribution S and T, and a ideal hypothesis $h^* = \underset{h\in\mathcal{H}}{argmin}\ \epsilon_S(h) + \epsilon_T(h)$. source risk ϵ_S and target risk ϵ_T are defined as follows:

$$\begin{aligned}\epsilon_S(h) = \epsilon_S(h, y) = \mathbb{E}_{(x,y)\sim S}|h(x) - y| \\ \epsilon_T(h) = \epsilon_T(h, y) = \mathbb{E}_{(x,y)\sim T}|h(x) - y|\end{aligned} \tag{17}$$

Given the two hypotheses h_1 and h_2, the disagreement under the data distribution D is defined as below:

$$\epsilon_D(h_1, h_2) = \mathbb{E}_{(x,y)\sim D}|h_1(x) - h_2(x)| \tag{18}$$

The target risk can be bounded following the theory of [6] as:

Theorem 1 *For any hypothesis $h \in \mathcal{H}$, we have the target risk*

$$\epsilon_T(h) \leq \epsilon_S(h) + \epsilon_T(h^*) + \epsilon_S(h^*) + |\epsilon_T(h, h^*) - \epsilon_S(h, h^*)| \tag{19}$$

Generally, the hypothesis h uses the normal transforms (same as the weak augmentation in this paper), so $h(x) = h(w(x))$, $w(\cdot)$ is the weak augmentation. We define the different hypotheses h_1 and h_2 as below:

$$h_1 = h(w(x)), h_2 = h(s(x)) \tag{20}$$

where the $s(\cdot)$ is the strong augmentation in this paper.

In the Theorem 1, for a hypothesis h which is well trained in source domain S, the first item $\epsilon_S(h)$ is small enough. And the second and third items $\epsilon_T(h^*)$, $\epsilon_S(h^*)$ are small enough for the ideal hypothesis h^*. We bound the last item using the two hypothesis h_1 and h_2 as below:

$$\begin{aligned}|\epsilon_T(h, h^*) - \epsilon_S(h, h^*)| \leq |\epsilon_T(h_1, h^*) - \epsilon_S(h_1, h^*)| \\ +|\epsilon_T(h_2, h^*) - \epsilon_S(h_2, h^*)|\end{aligned} \tag{21}$$

We using h_1 to approximate any hypothesis in $\mathcal{H}$, the proof of Eq (21) is obvious. We jointly optimize the hypotheses h_1 and h_2 in our method. Hence, the domain discrepancy $|\epsilon_T(h, h^*) - \epsilon_S(h, h^*)|$ can be bounded using our optimization. We denote $\triangle(h_1, h_2)$ as $|\epsilon_T(h_1, h^*) - \epsilon_S(h_1, h^*)| + |\epsilon_T(h_2, h^*) - \epsilon_S(h_2, h^*)|$.

Because of the perturbation of strong augmentation, the risk of h_2 in target is close but larger than the h_1, and they share the source classifier, i.e., $|\epsilon_T(h_1, h^*) - \epsilon_S(h_1, h^*)| \leq |\epsilon_T(h_2, h^*) - \epsilon_S(h_2, h^*)|$. So we have the follow:

$$\begin{aligned}\triangle(h_1, h_2) &\leq 2\ |\epsilon_T(h_2, h^*) - \epsilon_S(h_2, h^*)| \\ &\leq 2 \sup_{h,h'\in\mathcal{H}} |\epsilon_T(h, h') - \epsilon_S(h, h')| \\ &= d_{\mathcal{H}\triangle\mathcal{H}}(S, T)\end{aligned} \tag{22}$$

The goal in UDA is to approximate the target risk $\epsilon_T(h)$, and then optimize it to a low value. The objective in our technique is to minimize $\triangle(h_1, h_2)$, which is more close to the supremum of $\mathcal{H}\triangle\mathcal{H}$ divergence, and this approximation does not undermine the UDA theory according to our experiments.

5 Experiment

5.1 Setup

Dataset. We apply our technique to various baselines and compare with many state-of-the-art methods on four public datasets, i.e., Office-31 [52], Office-Home [53], VisDA-C [4] and DomainNet [54]. Details about datasets can be found in supplementary material.

Implementation details. Following the protocol for UDA in previous methods [10,9], we use the same backbone networks for fair comparisons. All baseline methods are reproduced in our codebase. All methods are trained with 10k iterations, and use same learning rate scheduler adopted in [10]. For DomainNet, we evaluated various methods following the settings in [55]. We fix the batch size as 36, and use the SGD optimizer. We set the same tradeoff as 1 for transfer loss both in domain-alignment and regularization-based methods. We choosed the best λ for LeCo loss by [56]. Each task was randomly repeated three times.

Table 1: Accuracy (%) on Office-Home for UDA using the ResNet-50 backbone.

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 [57]	34.9	50.0	58.0	37.4	41.9	46.2	38.5	31.2	60.4	53.9	41.2	59.9	46.1
DAN [22]	43.6	57.0	67.9	45.8	56.5	60.4	44.0	43.6	67.7	63.1	51.5	74.3	56.3
MCD [38]	48.9	68.3	74.6	61.3	67.6	68.8	57.0	47.1	75.1	69.1	52.2	79.6	64.1
EntMin [20]	51.0	71.9	77.1	61.2	69.1	70.1	59.3	48.7	77.0	70.4	53.0	81.0	65.8
AFN [11]	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
SRDC [58]	52.3	76.3	81.0	69.5	76.2	78.0	68.7	53.8	81.7	76.3	57.1	85.0	71.3
ATDOC [17]	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
FixBi [15]	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
Mirror [16]	57.6	77.6	81.6	**71.9**	77.8	78.7	**72.0**	56.3	82.5	**77.9**	61.3	85.3	**73.4**
DANN [37]	44.2	64.2	73.5	53.2	61.1	64.5	52.2	40.7	73.5	66.4	47.6	77.3	59.9
+LeCo	47.3↑	70.0↑	74.9↑	59.0↑	69.0↑	68.4↑	59.3↑	47.6↑	76.8↑	70.8↑	53.2↑	81.1↑	64.8↑
CDAN [14]	50.7	70.6	76.0	57.6	70.0	70.0	57.4	50.9	77.3	70.9	56.7	81.6	65.8
+LeCo	55.4↑	71.3↑	79.1↑	64.2↑	72.8↑	74.3↑	64.4↑	55.9↑	81.4↑	74.0↑	61.3↑	84.6↑	69.9↑
BNM [10]	57.3	74.1	80.6	66.2	76.1	76.8	65.8	51.9	81.1	73.0	59.3	83.4	70.5
+LeCo	59.3↑	74.8↑	80.9↑	66.8↑	77.0↑	76.7	67.0↑	53.8↑	81.1	74.2↑	60.6↑	83.8↑	71.3↑
MCC [9]	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
+LeCo	**59.4**↑	**79.2**↑	**82.7**↑	68.3↑	78.0↑	**79.1**↑	68.3↑	55.7↑	**83.7**↑	75.8↑	62.1↑	86.2↑	73.2↑

5.2 Results

We verify the effectiveness of applying our technique over various baselines, and compare with state-of-the-art methods on four public datasets. The results of Office-Home [53], VisDA-C [4], Office-31 [52], and DomainNet [54] are reported in Tables 1, 2, 3, and 4, respectively.

Office-Home. We evaluated various methods on the total of 12 tasks on Office-Home [53] shown in Tab 1. We apply our LeCo to various methods including DANN [37], CDAN [14], BNM [10], and MCC [9]. For regularization-based methods (BNM and MCC), our technique both improves the results, with average accuracy improvements of 0.8% and 2.2%, respectively. Compared with the SOTA Mirror [16], we help MCC to achieve the comparable average accuracy

of 73.2% and achieves the best accuracy on 5 out of 12 tasks, bringing improvements to all tasks. Meanwhile, the proposed LeCo brings large margins to domain alignment methods. Based on methods DANN and CDAN, our method surprisingly improves the accuracies of all tasks, with the large average accuracy improvements of 4.9% and 4.1%, respectively. In general, the improvements to all baselines are shown in almost all tasks, demonstrating the robustness of LeCo.

Table 2: Accuracy (%) on VisDA-C for UDA using the ResNet-101 backbone.

Method	plane	bcycl	bus	car	horse	knife	mcycl	person	plant	sktbrd	train	truck	Mean
ResNet-101 [57]	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
DAN [22]	87.1	63.0	76.5	42.0	90.3	42.9	85.9	53.1	49.7	36.3	85.8	20.7	61.1
MCD [38]	87.0	60.9	83.7	64.0	88.9	79.6	84.7	76.9	88.6	40.3	83.0	25.8	71.9
AFN [11]	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
ATDOC [17]	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
BCDM [59]	95.1	87.6	81.2	73.2	92.7	95.4	86.9	82.5	95.1	84.8	88.1	39.5	83.4
FixBi [15]	96.1	87.8	**90.5**	90.3	96.8	95.3	**92.8**	**88.7**	87.2	94.2	**90.9**	25.7	87.2
CAN [60]	97.0	87.2	82.5	74.3	**97.8**	96.2	90.8	80.7	**96.6**	96.3	87.5	59.9	87.2
CAN+Mirror [16]	**97.2**	88.2	84.9	76.0	97.2	95.8	89.2	86.4	96.1	**96.6**	85.9	**61.2**	**87.9**
DANN [37]	90.4	36.1	84.5	40.7	55.9	63.6	80.9	56.5	78.9	58.9	70.3	21.6	61.5
+LeCo	93.4↑	30.0	87.5↑	55.2↑	89.4↑	93.0↑	88.3↑	41.3	80.1↑	38.5	68.7	19.1	65.4↑
CDAN [14]	91.8	73.8	83.6	57.6	82.2	76.9	88.4	76.9	88.9	83.0	76.1	36.1	76.3
+LeCo	95.0↑	72.1	88.8↑	78.6↑	92.0↑	93.4↑	91.1↑	77.1↑	91.5↑	87.4↑	78.6↑	27.7	81.1↑
BNM [10]	94.8	84.1	74.3	52.7	89.2	93.0	81.3	82.5	88.7	65.9	81.9	49.0	78.1
+LeCo	95.8↑	**91.0**↑	75.5↑	75.9↑	96.6↑	**97.3**↑	80.4	80.0	95.3↑	87.0↑	80.3	51.0↑	83.8↑
MCC [9]	93.7	82.2	75.3	62.3	91.6	87.7	84.8	79.3	88.1	87.5	81.8	54.4	80.7
+LeCo	96.4↑	86.4↑	83.2↑	**90.6**↑	96.2↑	96.9↑	90.9↑	80.3↑	95.5↑	92.0↑	85.8↑	40.8	86.3↑

VisDA-C. We report classification accuracy in the synthetic-to-real transfer task as shown in Tab 2. We apply our LeCo to various methods including DANN [37], CDAN [14], BNM [10], and MCC [9]. It is noteworthy that the proposed LeCo achieved surprising improvements on these methods. We claim that due to the smaller number of categories, the intra-class variance can be effectively reduced by imposing our method. We can see that LeCo brings large improvements to these baselines in almost all tasks. Using our method, MCC can achieve a mean accuracy of 86.3%. It is worth noting that we get the highest accuracy of 90.6% on *car* class against the 90.3% of FixBi. However, the domain-alignment methods (FixBi, CAN and Mirror) beat our best result of 86.3%. Although our method achieved 4.5% improvement over MCC on average, it could lead to some class cluster overlap, resulting in a decrease in the accuracy of some classes. The reason is that our method does not explicitly reduce the inter-class variances, and results will be better if we cooperate with more powerful methods. Overall, our method can reasonably promote these baselines to reduce the intra-class variance, thus bringing considerable improvements.

Office-31. We compare various methods in this classic dataset. We apply our LeCo to methods DANN [37], BNM [10], CDAN [14], and MCC [9]. As shown in Tab 3, each method could obtain improvements by imposing our method. These methods do not directly consider the optimization for intra-class variance, and left room for improvements. Our method aims to lower the intra-class variance

by using implicit smoothness constraint. The results validate the effectiveness of our method. However, Mirror and FixBi got 91.7% and 91.4% accuracies, respectively, and both surpassed our best result of 90.0%. The reason may be that the images are fewer and the diversity of this dataset is relatively low, so it is hard to learn the instance discriminability, and the intra-class variances are lower naturally. We only achieve marginal improvements in this dataset.

DomainNet. Following the settings in [55], we compare various methods for 12 tasks among Clp, Pnt, Rel, and Skt domains on original DomainNet. The results are shown in Tab 4. Our method improves the average accuracy of MCC [9] by 2.9%. Specifically, our method brings improvements to CDAN on all tasks. Our method also improves MCC on 10 of 12 tasks. And we achieve the best average accuracy of 52.6% on this public benchmark.

Table 3: Accuracy (%) on Office-31 for UDA using the ResNet-50 backbone.

Method	A→D	A → W	D→A	D→W	W→A	W→D	Avg
ResNet-50 [57]	78.3	70.4	57.3	93.4	61.5	98.1	76.5
DAN [22]	78.6	80.5	63.6	97.1	62.8	99.6	80.4
AFN [11]	87.7	88.8	69.8	98.4	69.7	99.8	85.7
BCDM [59]	93.8	95.4	73.1	98.6	73.0	100.0	89.0
ATDOC [17]	94.4	94.3	75.6	98.9	75.2	99.6	89.7
FixBi [15]	95.0	96.1	**78.7**	99.3	**79.4**	100.0	91.4
Mirror [16]	**96.2**	**98.5**	77.0	**99.3**	78.9	**100.0**	**91.7**
DANN [37]	85.7	90.2	68.3	97.6	66.4	99.2	84.6
+LeCo	86.8↑	90.8↑	70.9↑	98.0↑	73.2↑	100.0↑	86.6↑
BNM [10]	90.3	91.4	71.4	97.9	71.7	100.0	87.1
+LeCo	92.8↑	92.8↑	73.7↑	98.6↑	73.2↑	100.0	88.5↑
CDAN [14]	92.9	93.1	71.0	98.6	69.3	100.0	87.5
+LeCo	91.6	91.1	75.6↑	98.4	74.9↑	100.0	88.6↑
MCC [9]	95.6	96.1	73.5	98.1	73.6	100.0	89.5
+LeCo	95.4	96.2↑	73.8↑	98.5↑	75.9↑	100.0	90.0↑

According to these extensive experiments, we believe that LeCo could exploit the underlying property of UDA about the intra-class variance, and then lead to promising performance from this perspective. Above all, our method is pluggable to various methods, and it brings remarkable improvements to these methods on different public datasets. Based on some widely-used baselines, we achieve comparable results against SOTA methods on four public datasets. These observations indicate that our technique is very effective for UDA setting.

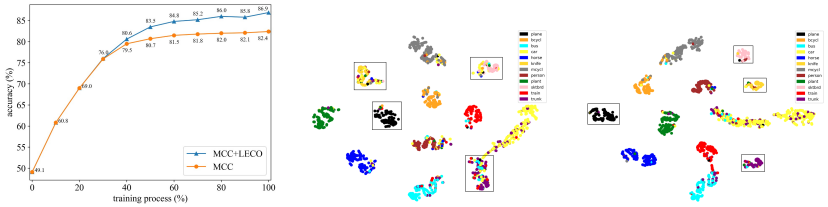
Table 4: Accuracy (%) on DomainNet for UDA using the ResNet-101 backbone.

Method	Clp→Pnt	Clp→Rel	Clp→Skt	Pnt→Clp	Pnt→Rel	Pnt→Skt	Rel→Clp	Rel→Pnt	Rel→Skt	Skt→Clp	Skt→Pnt	Skt→Rel	Avg
ResNet-101 [57]	32.7	50.6	39.4	41.1	56.8	35.0	48.6	48.8	36.1	49.0	34.8	46.1	43.3
DANN [37]	37.9	54.3	44.4	41.7	55.6	36.8	50.7	50.8	40.1	55.0	45.0	54.5	47.2
BCDM [59]	38.5	53.2	43.9	42.5	54.5	38.5	51.9	51.2	40.6	53.7	46.0	53.4	47.3
MCD [38]	37.5	52.9	44.0	44.6	54.5	41.6	52.0	51.5	39.7	55.5	44.6	52.0	47.5
ADDA [61]	38.4	54.1	44.1	43.5	56.7	39.2	52.8	51.3	40.9	55.0	45.4	54.5	48.0
DAN [22]	38.8	55.2	43.9	45.9	59.0	40.8	50.8	49.8	38.9	56.1	45.9	55.5	48.4
JAN [33]	40.5	56.7	45.1	47.2	**59.9**	43.0	54.2	52.6	41.9	56.6	46.2	55.5	50.0
MDD [12]	42.9	**59.5**	47.5	48.6	59.4	42.6	58.3	53.7	46.2	58.7	46.5	**57.7**	51.8
CDAN [14]	39.9	55.6	45.9	44.8	57.4	40.7	56.3	52.5	44.2	55.1	43.1	53.2	49.1
+LeCo	40.0↑	56.5↑	46.6↑	45.3↑	58.2↑	41.6↑	56.9↑	53.1↑	46.0↑	55.5↑	44.3↑	53.3↑	49.8↑
MCC [9]	40.1	56.5	44.9	46.9	57.7	41.4	56.0	53.7	40.6	58.2	45.1	55.9	49.7
+LeCo	**44.1**↑	55.3	**48.5**↑	**49.4**↑	57.5	**45.5**↑	**58.8**↑	**55.4**↑	**46.8**↑	**61.3**↑	**51.1**↑	**57.7**↑	**52.6**↑

5.3 Ablation Study

Training strategy. In the training process, the warm-up of supervised source training is important. For example, we set warm-up iteration to 3k on VisDA-C [4]. We show the classification accuracy of the synthetic-to-real task on VisDA-C during the training in Fig 3 (a). The accuracy at 0% denotes the accuracy of

the pre-trained ResNet-101 model, which has been finetuned with source labeled data. We compare the results of the MCC baseline and MCC+LeCo. In fact, during the warm-up phase, the differences only depend on the randomness of training. We can see that using our LeCo, the classification accuracy of the model is promoted stably. With our method, MCC outperforms the original baseline by a large margin.



(a) Training process. (b) t-SNE: MCC. (c) t-SNE: MCC+LeCo.

Fig. 3: (a): Training process. The start point denotes that the pre-trained ResNet-101 which has been source-only finetuned on VisDA-C, a random experiment. (b) and (c) correspond to the t-SNE embedding visualization of MCC and MCC+LeCo on VisDA-C.

Feature Visualization. To better illustrate that LeCo can reduce the intra-class variance of target features, we visualize the features learned by MCC and MCC+LeCo on VisDA-C. We employ the t-SNE method [62] for feature visualization. We randomly select 2000 samples across 12 categories from the real-world domain in VisDA-C, and extract the corresponding learned features. As shown in Fig 3 (b), (c), compared with the MCC, our method better separate the target samples in the feature space. These class centers become more compact by using our method, especially the centers marked by black rectangles. The results indicate small intra-class variances by using our method.

Effects of the components. We conduct the ablation study on the two components of LeCo. *NC* denotes the Naïve smoothness Constraint, and *Cond* denotes the naïve smoothness constraint conditioned by instance class confusion. We report the results on VisDA-C and Office-Home shown in Tab 5a. We analyze the effects of components of LeCo for CDAN [14] and MCC [9], and we observe stable improvements over the two datasets. Each component shows a positive effect, and proposed technique is verified to be very effective on various settings. In Tab 5b, We also compare the common technique entropy (Ent) with the instance class confusion (ICC) as the condition way. The results show the large improvements of the latter, and prove that it is more suitable for our technique.

Choice of constraint. The default choice of LeCo is using the L2-distance between the weak and strong predictions. We claimed that using the L2-distance considers the all class probabilities, and is better to transfer knowledge. We use

Table 5: Detailed ablations. (a) analyses the effectiveness of the components. (b) compares two condition ways. (c) analyses parameter sensitivity, and (d) evaluates different constraint types.

(a)

VisDA-C				Office-Home			
CDAN	*NC*	*Cond*	acc	CDAN	*NC*	*Cond*	acc
✓			76.3	✓			65.8
✓	✓		79.6	✓	✓		68.9
✓	✓	✓	81.1	✓	✓	✓	69.9
MCC	*NC*	*Cond*	acc	MCC	*NC*	*Cond*	acc
✓			80.7	✓			71.1
✓	✓		85.3	✓	✓		72.7
✓	✓	✓	**86.3**	✓	✓	✓	**73.2**

(b)

VisDA-C	
MCC	acc
+LeCo (Ent)	85.5
+LeCo (ICC)	**86.3**
Office-Home	
MCC	acc
+LeCo (Ent)	72.5
+LeCo (ICC)	**73.2**

(c)

VisDA-C		Office-Home	
method	acc	method	acc
MCC	80.7	MCC	71.1
+cos.	84.1	+cos.	71.2
+sup.	84.0	+sup.	69.0
+L1	85.0	+L1	72.4
+L2	**85.3**	+L2	**72.7**

(d)

VisDA-C		Office-Home	
λ	acc	λ	acc
1	85.8	1	73.1
2	85.6	2	**73.2**
3	**86.3**	3	72.8
4	85.6	4	72.7
5	85.1	5	71.8

other choices to construct the smoothness constraint, including L1-distance (L1), cosine distance (cos.), and weak supervising strong (sup.) [18]. The results are shown in Tab 5c. For fair comparisons, we select the proper λ to tradeoff the implicit smoothness constraint loss. As we can see, cosine distance and weak supervising show less improvement, and even worse on Office-Home. Selected L2-distance show stable and considerable performance, so we think the L2-distance is a good choice for implicit smoothness constraint.

Parameter sensitivity. We set the different values of the tradeoff for LeCo, i.e. the value of λ, and it is often sensitive in UDA scenario. In order to test the robustness, we simply change the values of $\lambda \in [1, 5]$ for MCC both on Office-Home and VisDA-C. The results are shown in Tab 5d. On VisDA-C, the mean accuracy of the synthetic-to-real task is more sensitive to the tradeoff. On Office-Home, we find the accuracy sensitivity with regard to λ is relatively small. In a word, the proper trade-off of LeCo could bring large improvements to baselines.

6 Conclusion

In this paper, we investigated the previous methods in Unsupervised Domain Adaptation (UDA) from a new perspective: *smoothness*. We dived into the domain-alignment and regularization-based methods. These methods aim to increase the distance between cluster centers, i.e., enlarge inter-class variance, but also cause cluster overlapping and error accumulation. We propose a simple yet effective technique named LeCo (imp**L**icit smoothn**e**ss **Co**nstraint), to implicitly increase the smoothness of baseline models, i.e., lower intra-class variance. The keys are consistency on all class probabilities over weak and strong augmentations and a novel uncertainty measure named Instance Class Confusion to condition the consistency. LeCo guarantees the lower sensitivity to perturbations of samples. Extensive experiments demonstrate that LeCo is applicable to various domain-alignment and regularization-based baseline approaches.

Acknowledgement. The paper is supported in part by the National Key Research and Development Project (Grant No.2020AAA0106600), in part by National Natural Science Foundation of China: 62022083.

References

1. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. (2020) 9729–9738
2. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. (2021) 9650–9660
3. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. arXiv preprint arXiv:2111.06377 (2021)
4. Peng, X., Usman, B., Kaushik, N., Hoffman, J., Wang, D., Saenko, K.: Visda: The visual domain adaptation challenge. arXiv preprint arXiv:1710.06924 (2017)
5. Ben-David, S., Blitzer, J., Crammer, K., Pereira, F.: Analysis of representations for domain adaptation. Advances in neural information processing systems **19** (2006)
6. Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., Vaughan, J.W.: A theory of learning from different domains. Machine learning **79** (2010) 151–175
7. Zhao, S., Yue, X., Zhang, S., Li, B., Zhao, H., Wu, B., Krishna, R., Gonzalez, J.E., Sangiovanni-Vincentelli, A.L., Seshia, S.A., et al.: A review of single-source deep unsupervised visual domain adaptation. IEEE Transactions on Neural Networks and Learning Systems (2020)
8. Shimodaira, H.: Improving predictive inference under covariate shift by weighting the log-likelihood function. Journal of statistical planning and inference **90** (2000) 227–244
9. Jin, Y., Wang, X., Long, M., Wang, J.: Minimum class confusion for versatile domain adaptation. In: European Conference on Computer Vision, Springer (2020) 464–480
10. Cui, S., Wang, S., Zhuo, J., Li, L., Huang, Q., Tian, Q.: Towards discriminability and diversity: Batch nuclear-norm maximization under label insufficient situations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. (2020) 3941–3950
11. Xu, R., Li, G., Yang, J., Lin, L.: Larger norm more transferable: An adaptive feature norm approach for unsupervised domain adaptation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. (2019) 1426–1435
12. Zhang, Y., Liu, T., Long, M., Jordan, M.: Bridging theory and algorithm for domain adaptation. In: International Conference on Machine Learning. (2019) 7404–7413
13. French, G., Mackiewicz, M., Fisher, M.H.: Self-ensembling for visual domain adaptation. In: 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings, OpenReview.net (2018)
14. Long, M., Cao, Z., Wang, J., Jordan, M.I.: Conditional adversarial domain adaptation. In: Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada. (2018) 1647–1657
15. Na, J., Jung, H., Chang, H.J., Hwang, W.: Fixbi: Bridging domain spaces for unsupervised domain adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. (2021) 1094–1103
16. Zhao, Y., Cai, L., et al.: Reducing the covariate shift by mirror samples in cross domain alignment. Advances in Neural Information Processing Systems **34** (2021)

17. Liang, J., Hu, D., Feng, J.: Domain adaptation with auxiliary target domain-oriented classifier. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. (2021) 16632–16642
18. Sohn, K., Berthelot, D., Carlini, N., Zhang, Z., Zhang, H., Raffel, C., Cubuk, E.D., Kurakin, A., Li, C.: Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In: Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual. (2020)
19. Berthelot, D., Carlini, N., Goodfellow, I.J., Papernot, N., Oliver, A., Raffel, C.: Mixmatch: A holistic approach to semi-supervised learning. In: Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada. (2019) 5050–5060
20. Grandvalet, Y., Bengio, Y., et al.: Semi-supervised learning by entropy minimization. CAP **367** (2005) 281–296
21. Tzeng, E., Hoffman, J., Zhang, N., Saenko, K., Darrell, T.: Deep domain confusion: Maximizing for domain invariance. arXiv preprint arXiv:1412.3474 (2014)
22. Long, M., Cao, Y., Wang, J., Jordan, M.: Learning transferable features with deep adaptation networks. In: International conference on machine learning. (2015) 97–105
23. Ma, N., Bu, J., Lu, L., Wen, J., Zhou, S., Zhang, Z., Gu, J., Li, H., Yan, X.: Context-guided entropy minimization for semi-supervised domain adaptation. Neural Networks **154** (2022) 270–282
24. Yao, T., Pan, Y., Ngo, C.W., Li, H., Mei, T.: Semi-supervised domain adaptation with subspace learning for visual recognition. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. (2015) 2142–2150
25. Saito, K., Kim, D., Sclaroff, S., Darrell, T., Saenko, K.: Semi-supervised domain adaptation via minimax entropy. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. (2019) 8050–8058
26. Kim, T., Kim, C.: Attract, perturb, and explore: Learning a feature alignment network for semi-supervised domain adaptation. In: European Conference on Computer Vision, Springer (2020) 591–607
27. Li, R., Jiao, Q., Cao, W., Wong, H.S., Wu, S.: Model adaptation: Unsupervised domain adaptation without source data. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. (2020) 9641–9650
28. Liang, J., Hu, D., Feng, J.: Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In: International Conference on Machine Learning. (2020) 6028–6039
29. Wang, X., Zhuo, J., Cui, S., Wang, S.: Learning invariant representation with consistency and diversity for semi-supervised source hypothesis transfer. arXiv preprint arXiv:2107.03008 (2021)
30. Shu, Y., Cao, Z., Long, M., Wang, J.: Transferable curriculum for weakly-supervised domain adaptation. In: Proceedings of the AAAI Conference on Artificial Intelligence. Volume 33. (2019) 4951–4958
31. Zhuo, J., Wang, S., Huang, Q.: Uncertainty modeling for robust domain adaptation under noisy environments. IEEE Transactions on Multimedia (2022)
32. Zhuo, J., Wang, S., Zhang, W., Huang, Q.: Deep unsupervised convolutional domain adaptation. In: Proceedings of the 25th ACM international conference on Multimedia, ACM (2017) 261–269

33. Long, M., Zhu, H., Wang, J., Jordan, M.I.: Deep transfer learning with joint adaptation networks. In: International conference on machine learning. (2017) 2208–2217
34. Zellinger, W., Grubinger, T., Lughofer, E., Natschläger, T., Saminger-Platz, S.: Central moment discrepancy (CMD) for domain-invariant representation learning. In: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings, OpenReview.net (2017)
35. Borgwardt, K.M., Gretton, A., Rasch, M.J., Kriegel, H.P., Schölkopf, B., Smola, A.J.: Integrating structured biological data by kernel maximum mean discrepancy. Bioinformatics **22** (2006) e49–e57
36. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. Advances in neural information processing systems **27** (2014)
37. Ganin, Y., Lempitsky, V.: Unsupervised domain adaptation by backpropagation. In: International conference on machine learning. (2015) 1180–1189
38. Saito, K., Watanabe, K., Ushiku, Y., Harada, T.: Maximum classifier discrepancy for unsupervised domain adaptation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2018) 3723–3732
39. Li, S., Lv, F., Xie, B., Liu, C.H., Liang, J., Qin, C.: Bi-classifier determinacy maximization for unsupervised domain adaptation. In: AAAI. Volume 2. (2021) 5
40. Cui, S., Wang, S., Zhuo, J., Li, L., Huang, Q., Tian, Q.: Fast batch nuclear-norm maximization and minimization for robust domain adaptation. arXiv preprint arXiv:2107.06154 (2021)
41. Zhuo, J., Wang, S., Cui, S., Huang, Q.: Unsupervised open domain recognition by semantic discrepancy minimization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. (2019) 750–759
42. Li, S., Xie, B., Lin, Q., Liu, C.H., Huang, G., Wang, G.: Generalized domain conditioned adaptation network. IEEE Transactions on Pattern Analysis and Machine Intelligence (2021)
43. Li, S., Xie, M., Gong, K., Liu, C.H., Wang, Y., Li, W.: Transferable semantic augmentation for domain adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. (2021) 11516–11525
44. Tarvainen, A., Valpola, H.: Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. Advances in neural information processing systems **30** (2017)
45. Verma, V., Kawaguchi, K., Lamb, A., Kannala, J., Bengio, Y., Lopez-Paz, D.: Interpolation consistency training for semi-supervised learning. arXiv preprint arXiv:1903.03825 (2019)
46. Chen, Y., Zhu, X., Gong, S.: Semi-supervised deep learning with memory. In: Proceedings of the European conference on computer vision (ECCV). (2018) 268–283
47. Laine, S., Aila, T.: Temporal ensembling for semi-supervised learning. In: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings, OpenReview.net (2017)
48. Athiwaratkun, B., Finzi, M., Izmailov, P., Wilson, A.G.: There are many consistent explanations of unlabeled data: Why you should average. In: 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019, OpenReview.net (2019)
49. Zhang, Y., Li, J., Wang, Z.: Low-confidence samples matter for domain adaptation. arXiv preprint arXiv:2202.02802 (2022)

50. Zheng, Z., Yang, Y.: Unsupervised scene adaptation with memory regularization in vivo. In: Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence. (2021) 1076–1082
51. Cubuk, E.D., Zoph, B., Shlens, J., Le, Q.V.: Randaugment: Practical automated data augmentation with a reduced search space. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. (2020) 702–703
52. Saenko, K., Kulis, B., Fritz, M., Darrell, T.: Adapting visual category models to new domains. In: European conference on computer vision, Springer (2010) 213–226
53. Venkateswara, H., Eusebio, J., Chakraborty, S., Panchanathan, S.: Deep hashing network for unsupervised domain adaptation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2017) 5018–5027
54. Peng, X., Bai, Q., Xia, X., Huang, Z., Saenko, K., Wang, B.: Moment matching for multi-source domain adaptation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. (2019) 1406–1415
55. Jiang, J., Chen, B., Fu, B., Long, M.: Transfer-learning-library. `https://github.com/thuml/Transfer-Learning-Library` (2020)
56. You, K., Wang, X., Long, M., Jordan, M.: Towards accurate model selection in deep unsupervised domain adaptation. In: International Conference on Machine Learning, PMLR (2019) 7124–7133
57. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2016) 770–778
58. Tang, H., Chen, K., Jia, K.: Unsupervised domain adaptation via structurally regularized deep clustering. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. (2020) 8725–8735
59. Li, S., Lv, F., Xie, B., Liu, C.H., Liang, J., Qin, C.: Bi-classifier determinacy maximization for unsupervised domain adaptation. In: Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021, AAAI Press (2021) 8455–8464
60. Kang, G., Jiang, L., Yang, Y., Hauptmann, A.G.: Contrastive adaptation network for unsupervised domain adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. (2019) 4893–4902
61. Tzeng, E., Hoffman, J., Saenko, K., Darrell, T.: Adversarial discriminative domain adaptation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2017) 7167–7176
62. Van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9** (2008)