Reusing the Task-specific Classifier as a Discriminator: Discriminator-free Adversarial Domain Adaptation

Supplementary Material

This supplementary material provides more details that are not presented in the main paper due to space limitations. In the following sections, we first provide the proof of the implicitly constructed discriminator $D = \|C\|_*$ satisfying the K-Lipschitz constraint. Then, we prove that the expected target risk can be bounded by the expected and empirical measures of the Nuclear-norm 1-Wasserstein discrepancy (NWD) on the source and target domains. Finally, more implementation details, experimental results, and insight analysis are presented, including the detailed comparisons on the VisDA-2017 dataset, the extra comparisons on the DomainNet [15] dataset, and analyses regarding toy experiments, Proxy $\mathcal{A}$-distance, self-correlation matrix, convergence, and trade-off parameters (λ and γ).

A. K-Lipschitz constraint

To prove that the implicitly constructed discriminator $D = \|C\|_*$ satisfies the K-Lipschitz constraint, where the classifier C consists of a fully connected layer $L_c(\cdot)$ and a softmax function $S_m(\cdot)$, we first analyze $L_c(\cdot)$ and $S_m(\cdot)$, respectively.

Definition 1. *Given two metric spaces (M, d_m) and (N, d_n), where d_m denotes the metric on the compact set $M \subseteq \mathbb{R}^m$ and d_n is the metric on the compact set $N \subseteq \mathbb{R}^n$, a function $h : M \to N$ is called K-Lipschitz continuous if there exists a real constant $K \geq 0$ (the minimum K called Lipschitz constant) such that, for $\forall m_1, m_2 \in M$, the following holds*

$$\|h\|_L = \sup_{m_1 \neq m_2} \frac{d_N(h(m_1), h(m_2))}{d_M(m_1, m_2)} \leq K. \quad (1)$$

Proposition 1. *Given two metric spaces $(F, |\cdot|)$ and $(O, |\cdot|)$, where $F \subseteq \mathbb{R}^d$ and $O \subseteq \mathbb{R}^k$ denote the compact input feature set and output set, respectively, $|\cdot|$ denotes the Frobenius norm in O or F. Then, for every input feature $f \in F$, the Lipschitz constant K of the fully connected layer $L_c(f) = Wf + b$, where $L_c(\cdot) : F \to O$ maps the feature $f \in F$ to the output $o \in O$, $W \in \mathbb{R}^{k \times d}$ denotes the weight matrix, and $b \in \mathbb{R}^k$ denotes the bias vector, has a upper bound $\|W\|_F$.*

Proof. Given features $f_1, f_2 \in F$, if $f_1 = f_2$, we have $|h(f_1) - h(f_2)| = K|f_1 - f_2| = 0$, and if $f_1 \neq f_2$, we have

$$\begin{aligned} |L_c(f_1) - L_c(f_2)| &= |(Wf_1 + b) - (Wf_2 + b)| \\ &= |W(f_1 - f_2)|. \end{aligned}$$

Meanwhile, the spectral norm of the matrix W induced by $|f|$ is defined as

$$\|W\|_2 = \max_{f \neq \mathbf{0}} \frac{|Wf|}{|f|} = \sigma_{\max},$$

where $\sigma_{\max}$ is the maximum singular value obtained by singular value decomposition (SVD) on the matrix W. Therefore, according to Definition 1, the Lipschitz constant K is $\|W\|_2$. Additionally, the Frobenius norm of the matrix W is defined as

$$\|W\|_F = \sqrt{\sum_{i=1}^{k}\sum_{j=1}^{d} W_{i,j}^2} = \sqrt{\sum_{i=1}^{r} \sigma_i^2},$$

where $r = \min\{k, d\}$, σ_i denotes the i-th singular value. Thus, for every $f_1, f_2 \in F$, we have

$$\begin{aligned} |L_c(f_1) - L_c(f_2)| &= |W(f_1 - f_2)| \\ &\leqslant \|W\|_2 |f_1 - f_2| \\ &= K|f_1 - f_2| \\ &\leqslant \|W\|_F |f_1 - f_2|. \end{aligned}$$

□

According to Proposition 1, $\|W\|_F$ can be an upper bound of the Lipschitz constant K. As a widely used strategy, the weight decay (implemented with Frobenius norm regularization), which improves the generalization performance of a DNN model through minimizing an additional term $\lambda_0 \|W\|_F^2$ (where λ_0 is a trade-off parameter), can simultaneously enforce the fully connected layer to satisfy the K-Lipschitz constraint.

Definition 2. *(Remark 4.6.10 in [19]) Given a function $h : M \to N$, where M and N denote the compact subsets of $\mathbb{R}^m$ and $\mathbb{R}^n$, h will satisfy K-Lipschitz continuous if there exists a real constant $K \geq 0$ (the minimum K called Lipschitz constant), such that, for any $m_1, m_2 \in M$, $m_1 \neq m_2$, all the first partial derivatives are bounded by K.*

Proposition 2. *Given the compact output set $O \subseteq \mathbb{R}^k$ and prediction set $P \subseteq \mathbb{R}^k$, the softmax function $S_m(\cdot) : O \to P$ mapping the output $o \in O$ to the prediction $p \in P$ satisfies the 1-Lipschitz constraint.*

Proof. Let $p = S_m(o)$, where p_i is in the range from 0 to 1,

for $\forall i \in 1 \ldots k$, then, we have

$$p_i = \frac{e^{o_i}}{\sum_{j=1}^{k} e^{o_j}} \quad \forall i \in 1 \ldots k$$
$$\sum_{i=1}^{k} p_i = 1 \quad \forall i \in 1 \ldots k.$$

For simplicity, we denote $\sum_{j=1}^{k} e^{o_j}$ as Σ. Then, the Jacobian matrix can be written as

$$J = \begin{bmatrix} \frac{\partial p_1}{\partial o_1} & \cdots & \frac{\partial p_1}{\partial o_k} \\ \vdots & \ddots & \vdots \\ \frac{\partial p_k}{\partial o_1} & \cdots & \frac{\partial p_k}{\partial o_k} \end{bmatrix}.$$

According to the quotient rule, when $i = j$, we have

$$\begin{aligned} \frac{\partial p_i}{\partial o_j} &= \frac{e^{o_i}\Sigma - e^{o_j}e^{o_i}}{\Sigma^2} \\ &= \frac{e^{o_i}}{\Sigma} \cdot \frac{\Sigma - e^{o_j}}{\Sigma} \\ &= p_i\left(1 - p_j\right). \end{aligned}$$

When $i \neq j$, we have

$$\begin{aligned} \frac{\partial p_i}{\partial o_j} &= \frac{0 - e^{o_j}e^{o_i}}{\Sigma^2} \\ &= -\frac{e^{o_i}}{\Sigma} \cdot \frac{e^{o_j}}{\Sigma} \\ &= -p_i p_j. \end{aligned}$$

Therefore, we have

$$J_{i,j} = \begin{cases} p_i\left(1 - p_j\right) & i = j \\ -p_i p_j & i \neq j. \end{cases}$$

Therefore, given p_i in the range from 0 to 1, $J_{i,j}$ can be bounded by 1. Thus, according to Definition 2, softmax function satisfies the 1-Lipschitz constraint. □

B. Generalization Bound

In this section, we first provide the proof for Lemma 1 and Theorem 1. Then, based on Theorem 1, we prove that the expected target risk can be bounded by the expected and empirical measures of NWD on the source and target domains.

Lemma 1. *(Lemma 1 of [16]; Lemma1 of [18]) Let $\nu_s, \nu_t \in \mathcal{P}(\mathcal{F})$ denote the probability measures of the source and target domain features, $\rho(f^s, f^t)$ be the cost of transporting a unit of material from location f^s satisfying $f^s \sim \nu_s$ to location f^t satisfying $f^t \sim \nu_t$, $W_1(\nu_s, \nu_t)$ represent the NWD, and K denote a Lipschitz constant. Given a family of classifiers $C \in \mathcal{H}_1$ and a ideal classifier $C^* \in \mathcal{H}_1$ satisfying the K-Lipschitz constraint, where $\mathcal{H}_1$ is a subspace of $\mathcal{H}$, the following holds for every $C, C^* \in \mathcal{H}_1$.*

$$\left|\varepsilon_s(C, C^*) - \varepsilon_t(C, C^*)\right| \leqslant 2KW_1(\nu_s, \nu_t). \quad (2)$$

Proof. For every $C, C^* \in \mathcal{H}_1$ satisfying the K-Lipschitz constraint, according to the triangle inequality, we have

$$\begin{aligned} \left|C(f^s) - C^*(f^s)\right| &\leqslant \left|C(f^s) - C\left(f^t\right)\right| \\ &\quad + \left|C\left(f^t\right) - C^*(f^s)\right| \\ &\leqslant \left|C(f^s) - C\left(f^t\right)\right| \\ &\quad + \left|C\left(f^t\right) - C^*\left(f^t\right)\right| \\ &\quad + \left|C^*(f^s) - C^*\left(f^t\right)\right|. \end{aligned}$$

Therefore, for every $f_1, f_2 \in \mathcal{F}$, the following holds,

$$\frac{\left|\left|C(f^s) - C^*(f^s)\right| - \left|C(f^t) - C^*(f^t)\right|\right|}{\rho(f^s, f^t)} \leqslant 2K.$$

For simplicity, we denote the discrepancy term $\left|\varepsilon_s(C, C^*) - \varepsilon_t(C, C^*)\right|$ as dis. Thus, given $f_1, f_2 \in \mathcal{F}$, and C' denotes the labeling function $C' : \mathcal{F} \to [0, 1]$, for every $C, C^* \in \mathcal{H}_1$, we have

$$\begin{aligned} dis &= \mathbb{E}_{\nu_t}\left[\left|C\left(f^t\right) - C^*\left(f^t\right)\right|\right] - \mathbb{E}_{\nu_s}\left[\left|C(f^s) - C^*(f^s)\right|\right] \\ &\leqslant \sup_{\|C'\|_L \leqslant K} \mathbb{E}_{\nu_s}\left[C'(f^s)\right] - \mathbb{E}_{\nu_t}\left[C'\left(f^t\right)\right] \\ &= 2KW_1(\nu_s, \nu_t). \end{aligned}$$

□

Theorem 1. *Based on Lemma 1, for every $C \in \mathcal{H}_1$, the following holds*

$$\varepsilon_t(C) \leqslant \varepsilon_s(C) + 2KW_1(\nu_s, \nu_t) + \eta^*, \quad (3)$$

where $\eta^* = \varepsilon_s(C^*) + \varepsilon_t(C^*)$ is the risk of ideal joint hypothesis, which is a sufficiently small constant.

Proof. Based on Lemma 1, we have

$$\begin{aligned} \varepsilon_t(C) &\leqslant \varepsilon_t(C^*) + \varepsilon_t(C^*, C) \\ &= \varepsilon_t(C^*) + \varepsilon_s(C, C^*) + \varepsilon_t(C^*, C) - \varepsilon_s(C, C^*) \\ &= \varepsilon_t(C^*) + \varepsilon_s(C, C^*) + \varepsilon_t(C, C^*) - \varepsilon_s(C, C^*) \\ &\leqslant \varepsilon_t(C^*) + \varepsilon_s(C, C^*) + 2KW_1(\nu_s, \nu_t) \\ &\leqslant \varepsilon_t(C^*) + \varepsilon_s(C) + \varepsilon_s(C^*) + 2KW_1(\nu_s, \nu_t) \\ &= \varepsilon_s(C) + 2KW_1(\nu_s, \nu_t) + \eta^*. \end{aligned}$$

□

Therefore, the expected target risk can be bounded by the expected measures of the NWD on the source and target domain distributions. Furthermore, we show the convergence of the empirical measures to the expected measures of the NWD on the source and target domain samples.

Definition 3. *Given $p \geqslant 1$ and $\eta > 0$, a probability measure ν on $\mathcal{F}$ satisfies $T_p(\eta)$ if the inequality*

$$W(\nu', \nu) \leqslant \sqrt{\frac{2}{\eta} H(\nu'|\nu)}, \quad (4)$$

where

$$H\left(\nu'|\nu\right)=\int\frac{d\nu'}{d\nu}\log\frac{d\nu'}{d\nu}d\nu, \tag{5}$$

holds for any probability measure ν'.

Lemma 2. *(Theorem 2.1 of [1]; Theorem 1 of [16]; Theorem 2 of [18]) Let $\nu\in\mathcal{P}\left(\mathcal{F}\right)$ be a probability measure in representation space $\mathcal{F}$, where $\mathcal{F}$ is a subspace of $\mathbb{R}^d$, satisfying $T_1\left(\eta^*\right)$ inequality. Let $\hat{\nu}=\frac{1}{N}\sum_{i=1}^{N}\delta_{f_i}$ be its associated empirical measure defined on a sample set $\{f_i\}_{i=1}^{N}$ of size N drawn i.i.d from ν. Then for any $d'>d$ and $\eta'<\eta^*$, there exists some constant N_0 depending on d' and some square exponential moment of ν such that for any $\epsilon>0$ and $N\geqslant N_0\max\left(\epsilon^{-(d+2)},1\right)$, the following holds*

$$\mathbb{P}\left[W_1\left(\nu,\hat{\nu}\right)>\epsilon\right]\leqslant\exp\left(-\frac{\eta'}{2}N\epsilon^2\right), \tag{6}$$

where d', η' can be calculated explicitly.

Theorem 2. *(Theorem 3 of [16]; Theorem 3 of [18]) Under the assumption of Lemma 1 and Lemma 2, let two probability measures $\nu_s,\nu_t\in\mathcal{P}\left(\mathcal{F}\right)$ of the source and target domain features satisfy the $T_1\left(\eta^*\right)$ inequality, F_s and F_t be two sample sets of size N_s and N_t drawn i.i.d from ν_s and ν_t, respectively. Let $\hat{\nu}_s=\frac{1}{N_s}\sum_{i=1}^{N_s}\delta_{f_i^s}$ and $\hat{\nu}_t=\frac{1}{N_t}\sum_{i=1}^{N_t}\delta_{f_i^t}$ be the associated empirical measures. Then for any $d'>d$ and $\eta'<\eta^*$, there exists some constant N_0 depending on d' such that for any $\delta>0$ and $\min\left(N_s,N_t\right)\geqslant N_0\max\left(\delta^{-(d'+2)},1\right)$ with probability at least $1-\delta$ for all C, the following holds*

$$\begin{aligned}\varepsilon_t\left(C\right)\leqslant&\varepsilon_s\left(C\right)+2KW_1\left(\hat{\nu}_s,\hat{\nu}_t\right)+\eta^*\\&+2K\sqrt{2\log\left(\frac{1}{\delta}\right)\Big/\eta'}\left(\sqrt{\frac{1}{N_s}}+\sqrt{\frac{1}{N_t}}\right),\end{aligned} \tag{7}$$

where $\eta^*=\varepsilon_s\left(C^*\right)+\varepsilon_t\left(C^*\right)$ is a sufficiently small constant representing the ideal combined risk.

Proof. Based on Lemma 1 and Lemma 2, we have

$$\begin{aligned}\varepsilon_t\left(C\right)\leqslant&\varepsilon_s\left(C\right)+2KW_1\left(\nu_s,\nu_t\right)+\eta^*\\\leqslant&\varepsilon_s\left(C\right)+2KW_1\left(\nu_s,\hat{\nu}_s\right)+2KW_1\left(\hat{\nu}_s,\nu_t\right)+\eta^*\\\leqslant&\varepsilon_s\left(C\right)+2KW_1\left(\hat{\nu}_s,\hat{\nu}_t\right)+2KW_1\left(\hat{\nu}_t,\nu_t\right)+\eta^*\\&+2K\sqrt{2\log\left(\frac{1}{\delta}\right)\Big/N_s\eta'}\\\leqslant&\varepsilon_s\left(C\right)+2KW_1\left(\hat{\nu}_s,\hat{\nu}_t\right)+\eta^*\\&+2K\sqrt{2\log\left(\frac{1}{\delta}\right)\Big/\eta'}\left(\sqrt{\frac{1}{N_s}}+\sqrt{\frac{1}{N_t}}\right).\end{aligned}$$

□

C. Implementation details

The proposed method is implemented based on the PyTorch framework running on a GPU (Tesla-V100 32 GB). Following the existing methods [6, 13], the ResNet50 or ResNet101 pretrained on the ImageNet is used as the feature extractor G, in which we use the bottleneck layer used in [8] to replace the last fully connected layer. Classifier C is a fully connected layer depending on the specific task . The setting of the gradient reverse layer follows that of [6]. The SGD optimizer is used to train the model with a moment of 0.9, a weight decay of 1e-3, a batch size of 36, and a cropped image size of 224×224. The initial learning rate of classifier C is set to 5e-3, which is 10 times larger than that of the feature extractor G. Additionally, to facilitate model training, we use the annealing strategy [5] for the decay of the learning rate. **One can refer to our provided code for more implementation details.**

D. Detailed results on VisDA-2017

The detailed results on VisDA-2017 are shown in Table 1. The proposed DALN achieves an average accuracy of 80.6 %, outperforming the existing SOTA methods. Combining the proposed NWD with other methods, the performances of these methods are substantially improved by 22.6%, 7.5%, 5.2%, and 4.9% for the DANN, CDAN, MDD, and MCC, respectively. In particular, the improvements are evidently exhibited on categories including bus, car, person, and truck. These results come from the proposed NWD helping these methods distinguish some confusing class pairs such as bus and car, and train and truck.

E. Extra experiments on DomainNet

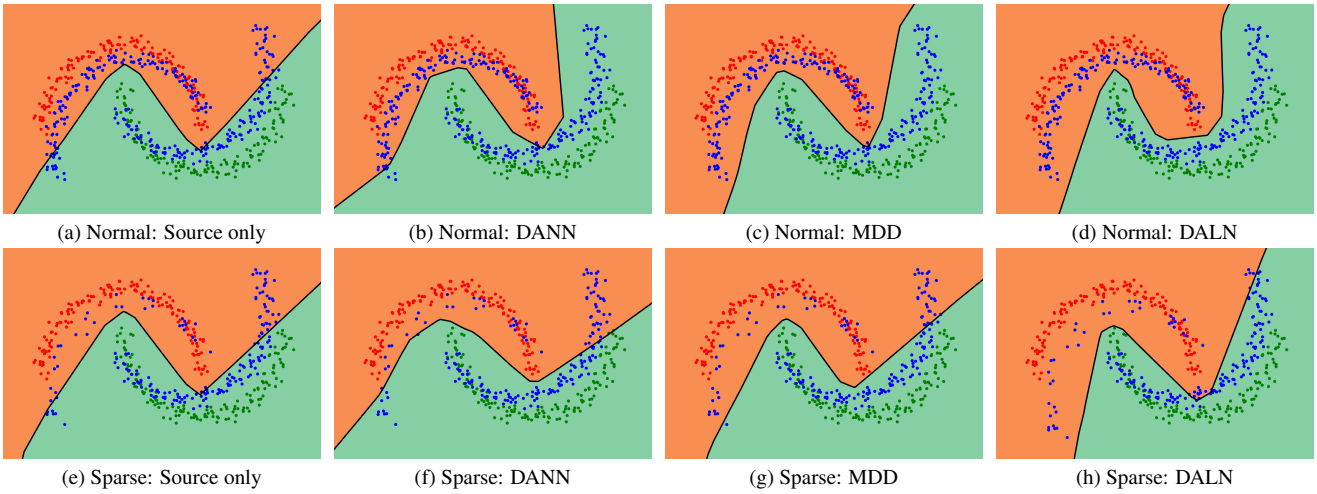
We further conduct an experiment on DomainNet [15] (containing 0.6 million images, 345 categories, and 6 subdomains), which consists of 30 sub-experiments. And the batch size is 64 for DomainNet. As the results shown in Table 2, DALN outperforms the previous SOTA methods impressively in terms of the average accuracy. Such encouraging results demonstrate the superiority of DALN for processing complex datasets.

F. Insight Analysis

Toy examples. We perform toy experiments to discuss the inter twinning moons 2D problem [14], helping analyze the learned decision boundary. The presented examples consider two cases. One case is generating balanced samples for both the source and target domains, and the other studies the class imbalance problem by reducing the samples of one class in the target domain. Specifically, for the first case, we generate 300 samples for two classes of the source samples labeled 0 and 1, and each class has 150 samples. As shown

Table 1. Classification accuracy (%) on VisDA-2017 for unsupervised domain adaptation (using ResNet-101 as the backbone). † denotes that the results are reproduced using the publicly released code. The best accuracy is indicated in **bold red** and the second best accuracy is indicated in undelined blue.

Method	plane	bcycl	bus	car	horse	knife	mcycl	person	plan	sktbrd	train	truck	Avg
ResNet-101 [7]	55.1	53.3	61.9	59.1	80.6	17.9	79.7	31.2	81.0	26.5	73.5	8.5	52.4
WDGRL† [18]	85.4	54.2	76.2	41.4	68.9	56.8	86.9	48.2	57.2	51.9	81.8	27.2	61.3
MCD [17]	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
BSP [2]	92.4	61.0	81.0	57.5	89.0	80.6	90.1	77.0	84.2	77.9	82.1	38.4	75.9
SWD [9]	90.8	82.5	81.7	70.5	91.7	69.5	86.3	77.5	87.4	63.6	85.6	29.2	76.4
BNM [3]	89.6	61.5	76.9	55.0	89.3	69.1	81.3	65.5	90.0	47.3	89.1	30.1	70.4
GVB-GD† [4]	90.1	68.7	81.9	61.7	91.2	67.3	**90.2**	76.5	90.2	77.8	**90.3**	41.0	77.2
DADA [20]	92.9	74.2	82.5	65.0	90.9	93.8	87.2	74.2	89.9	71.5	86.5	48.7	79.8
TSA [11]	-	-	-	-	-	-	-	-	-	-	-	-	78.6
SCDA† [12]	93.1	84.6	78.2	52.2	90.8	95.2	81.0	77.2	91.1	80.5	89.1	43.5	79.7
DALN(Ours)	96.0	**86.3**	74.3	50.0	92.4	94.7	83.5	76.4	91.0	87.2	88.4	47.4	80.6
DANN [6]	81.9	77.7	82.8	44.3	81.2	29.5	65.1	28.6	51.9	54.6	82.8	7.8	57.4
DANN+**NWD**	96.0	73.6	84.3	48.3	88.0	92.8	89.4	78.2	89.1	90.9	88.7	40.3	80.0
CDAN [13]	85.2	66.9	83.0	50.8	84.2	74.9	88.1	74.5	83.4	76.0	81.9	38.0	73.9
CDAN+**NWD**	94.8	80.0	84.2	56.0	92.3	91.5	90.1	78.7	88.0	**91.1**	88.9	41.4	81.4
MDD† [21]	80.1	61.3	83.7	51.8	90.7	83.8	89.7	77.3	90.2	86.6	82.2	44.5	76.8
MDD+**NWD**	94.0	81.0	**86.2**	63.5	90.5	**97.0**	87.5	76.3	88.6	86.5	85.2	48.2	82.0
MCC [8]	88.1	80.3	80.5	**71.5**	90.1	93.2	85.0	71.6	89.4	73.8	85.0	36.9	78.8
MCC+**NWD**	**96.1**	82.7	76.8	71.4	**92.5**	96.8	88.2	**81.3**	**92.2**	88.7	84.1	**53.7**	**83.7**



(a) Normal: Source only (b) Normal: DANN (c) Normal: MDD (d) Normal: DALN

(e) Sparse: Source only (f) Sparse: DANN (g) Sparse: MDD (h) Sparse: DALN

Figure 1. Comparisons of decision boundaries on a toy example dataset. Red points and green points denote classes 0 and 1 of source data, respectively. Blue points are target data generated by rotating the source data distribution by 30 degrees. In the second row, we reduce the number of samples to 1/4 for the upper moon of the target domain via a sampling strategy, which thus generates imbalanced class samples. The orange and green regions are classified as classes 0 and 1 by the final decision boundary, respectively.

in Fig. 1, samples corresponding to 0 are denoted by an upper moon, while samples corresponding to 1 are denoted by a lower moon. Then, the target samples are generated by rotating the data distribution of the source samples by 30 degrees, resulting in a domain shift for the target domain. In this case, the number of samples of each class in the two domains are equal. In contrast, for the second case, we reduce the number of samples to 38 for the upper moon of the target domain via a sampling strategy, which thus generates imbalanced class samples. As shown in Fig. 1, for the first case, the model trained on the source-only data can correctly classify the source samples, but cannot perform properly in the overall target samples. DANN improves the decision boundary, but some samples in the upper moon are misclassified. MDD and DALN successfully classify both the source and target samples, but the proposed DALN

achieves better classification performances compared with MDD. For the second case, except our DALN, both the DANN and MDD cannot learn a favorable decision boundary for the target samples. Some samples in the upper moon and lower moon are misclassified.

Table 2. Accuracy(%) on DomainNet for UDA. In each sub-table, the column-wise domains are selected as the source domain and the row-wise domains are selected as the target domain.

ResNet-101 [7]	clp	inf	pnt	qdr	rel	skt	Avg.
clp	-	19.3	37.5	11.1	52.2	41.0	32.2
inf	30.2	-	31.2	3.6	44.0	27.9	27.4
pnt	39.6	18.7	-	4.9	54.5	36.3	30.8
qdr	7.0	0.9	1.4	-	4.1	8.3	4.3
rel	48.4	22.2	49.4	6.4	-	38.8	33.0
skt	46.9	15.4	37.0	10.9	47.0	-	31.4
Avg.	34.4	15.3	31.3	7.4	40.4	30.5	26.6

SCDA(21) [12]	clp	inf	pnt	qdr	rel	skt	Avg.
clp	-	18.6	39.3	5.1	55.0	44.1	32.4
inf	29.6	-	34.0	1.4	46.3	25.4	27.3
pnt	44.1	19.0	-	2.6	56.2	42.0	32.8
qdr	30.0	4.9	15.0	-	25.4	19.8	19.0
rel	54.0	22.5	51.9	2.3	-	42.5	34.6
skt	55.6	18.5	44.7	6.4	53.2	-	35.7
Avg.	42.6	16.7	37.0	3.6	47.2	34.8	30.3

BCDM(21) [10]	clp	inf	pnt	qdr	rel	skt	Avg.
clp	-	19.9	38.5	15.1	53.2	43.9	34.1
inf	31.9	-	32.7	6.9	44.7	28.5	28.9
pnt	42.5	19.8	-	7.9	54.5	38.5	32.6
qdr	23.0	4.0	9.5	-	16.9	16.2	13.9
rel	51.9	24.9	51.2	8.7	-	40.6	35.5
skt	53.7	20.5	46.0	13.1	53.4	-	37.1
Avg.	40.6	17.8	35.6	10.3	44.3	33.5	30.4

DALN(Ours)	clp	inf	pnt	qdr	rel	skt	Avg.
clp	-	20.0	40.2	11.4	57.5	45.4	34.9
inf	35.2	-	34.7	4.7	47.9	29.0	30.3
pnt	45.3	19.2	-	3.2	57.4	40.0	33.0
qdr	27.5	4.2	13.2	-	21.8	16.6	16.7
rel	55.6	22.8	54.0	5.1	-	40.4	35.6
skt	59.0	19.9	46.0	8.3	56.3	-	37.9
Avg.	44.5	17.2	37.6	6.5	48.2	34.3	**31.4**

Proxy $\mathcal{A}$-distance. As shown in Fig. 2, we calculate the proxy $\mathcal{A}$-distance of the feature representations achieved by different methods based on task A→W of Office-31. Note that a smaller proxy $\mathcal{A}$-distance denotes better transferability. The proposed DALN achieves the lowest proxy $\mathcal{A}$-distance, demonstrating its superiority in learning transferable features. Moreover, by taking the NWD as a regularizer for DANN and MDD, their proxy $\mathcal{A}$-distances are considerably decreased, demonstrating the effectiveness of the proposed NWD in improving the transferability of the features.

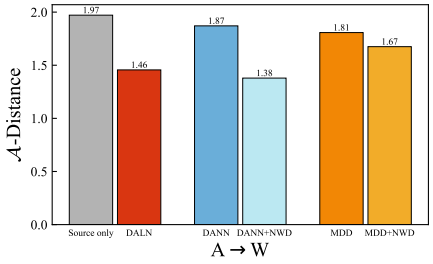
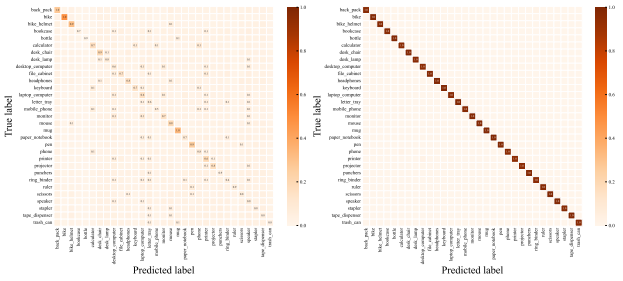


Figure 2. Visualization of the proxy $\mathcal{A}$-distance on task A→W of Office-31. Note that a smaller proxy $\mathcal{A}$-distance denotes better transferability.

Self-correlation matrix. As shown in Fig. 3, the model trained on the source-only data generates large values on the off-diagonal elements for the target domain samples. In contrast, with the adaptation of the proposed paradigm, the values of the self-correlation matrix generated from the target samples are highly concentrated on the main diagonal as shown in Fig. 3(b). Thus, the intra-class correlation I_a is increased and the inter-class correlation I_e is decreased, which demonstrates the effectiveness of the proposed method.

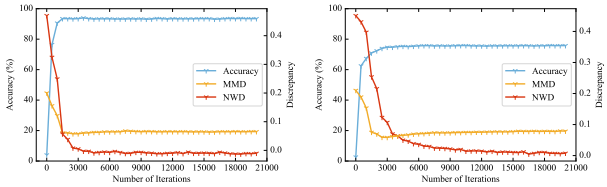
Convergence. We present the convergence curves of the test accuracy, NWD, and MMD with respect to the number of



(a) Source only (b) DALN

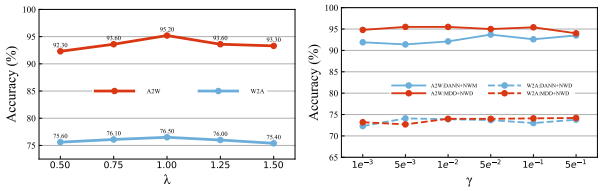
Figure 3. The self-correlation matrices of the predictions on the target domain on task A→W of Office-31. (Zoom in for a clear visualization.)

iterations on tasks A→W and W→A of Office-31, as shown in Fig. 4. Benefiting from the definite guidance meaning, DALN achieves rapid convergence with competitive accuracy. In particular, it can be observed that minimizing the NWD can also effectively decrease the widely-used maximum mean discrepancy (MMD), which also demonstrates the effectiveness of the NWD.



(a) A→W (b) W→A

Figure 4. The test accuracy, NWD and MMD convergence curves of the target domain on tasks A→W and W→A of Office-31.



(a) Influence of λ (b) Influence of γ

Figure 5. The influence of λ and γ on tasks A→W and W→A of Office-31.

Discussion of trade-off parameters λ and γ. λ is used to balance the losses $\mathcal{L}_{cls}$ and $\mathcal{L}_{nwd}$. γ is also a balance weight used for taking the proposed NWD as a regularizer. Here, we conduct influence analysis for these two parameters based on tasks A→W and W→A of Office-31. As shown in Fig. 5, DALN achieves the best performance when λ ranges from 0.75 to 1.25. For the parameter γ, pleasing results occur when γ is in the range of 0.005 to 0.01. Similar trends can also be observed in other datasets. For simplicity, in this work, we set λ to 1 and γ to 0.01 for all the experiments.

G. Limitations

Despite the simplicity and the impressive performance of our method, here comes two problems in the training process. One problem is that the SVD process takes some time to compute the Nuclear norm, and the other problem is that the performance always reaches the highest value early in the training process and then decreases slowly. These two problems will be explored in our future work.

References

[1] François Bolley, Arnaud Guillin, and Cédric Villani. Quantitative concentration inequalities for empirical measures on non-compact spaces. *Probability Theory and Related Fields*, 137(3-4):541–593, 2007. 3

[2] Xinyang Chen, Sinan Wang, Mingsheng Long, and Jianmin Wang. Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation. In *ICML*, pages 1081–1090, 2019. 4

[3] Shuhao Cui, Shuhui Wang, Junbao Zhuo, Liang Li, Qingming Huang, and Qi Tian. Towards discriminability and diversity: Batch nuclear-norm maximization under label insufficient situations. In *CVPR*, pages 3941–3950, 2020. 4

[4] Shuhao Cui, Shuhui Wang, Junbao Zhuo, Chi Su, Qingming Huang, and Qi Tian. Gradually vanishing bridge for adversarial domain adaptation. In *CVPR*, pages 12455–12464, 2020. 4

[5] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *ICML*, pages 1180–1189, 2015. 3

[6] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *JMLR*, pages 2096–2030, 2016. 3, 4

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 4, 5

[8] Ying Jin, Ximei Wang, Mingsheng Long, and Jianmin Wang. Minimum class confusion for versatile domain adaptation. In *ECCV*, pages 464–480, 2020. 3, 4

[9] Chen-Yu Lee, Tanmay Batra, Mohammad Haris Baig, and Daniel Ulbricht. Sliced wasserstein discrepancy for unsupervised domain adaptation. In *CVPR*, pages 10285–10295, 2019. 4

[10] Shuang Li, Fangrui Lv, Binhui Xie, Chi Harold Liu, Jian Liang, and Chen Qin. Bi-classifier determinacy maximization for unsupervised domain adaptation. In *AAAI*, 2021. 5

[11] Shuang Li, Mixue Xie, Kaixiong Gong, Chi Harold Liu, Yulin Wang, and Wei Li. Transferable semantic augmentation for domain adaptation. In *CVPR*, pages 11516–11525, 2021. 4

[12] Shuang Li, Mixue Xie, Fangrui Lv, Chi Harold Liu, Jian Liang, Chen Qin, and Wei Li. Semantic concentration for domain adaptation. In *ICCV*, pages 9102–9111, 2021. 4, 5

[13] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. In *NeurIPS*, volume 31, 2018. 3, 4

[14] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *JMLR*, 12:2825–2830, 2011. 3

[15] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *ICCV*, pages 1406–1415, 2019. 1, 3

[16] Ievgen Redko, Amaury Habrard, and Marc Sebban. Theoretical analysis of domain adaptation with optimal transport. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 737–753. Springer, 2017. 2, 3

[17] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *CVPR*, pages 3723–3732, 2018. 4

[18] Jian Shen, Yanru Qu, Weinan Zhang, and Yong Yu. Wasserstein distance guided representation learning for domain adaptation. In *AAAI*, 2018. 2, 3, 4

[19] Houshang H Sohrab. *Basic real analysis*, volume 231. Springer, 2003. 1

[20] Hui Tang and Kui Jia. Discriminative adversarial domain adaptation. In *AAAI*, pages 5940–5947, 2020. 4

[21] Yuchen Zhang, Tianle Liu, Mingsheng Long, and Michael Jordan. Bridging theory and algorithm for domain adaptation. In *ICML*, pages 7404–7413. PMLR, 2019. 4