Exploiting the Intrinsic Neighborhood Structure for Source-free Domain Adaptation

Shiqi Yang[1], **Yaxing Wang**[1,2]*, **Joost van de Weijer**[1], **Luis Herranz**[1], **Shangling Jui**[3]
[1] Computer Vision Center, Universitat Autonoma de Barcelona, Barcelona, Spain
[2] PCALab, Nanjing University of Science and Technology, China
[3] Huawei Kirin Solution, Shanghai, China
{syang,yaxing,joost,lherranz}@cvc.uab.es, jui.shangling@huawei.com

NIPS-2021

Background

- Unsupervised Domain Adaptation(UDA)

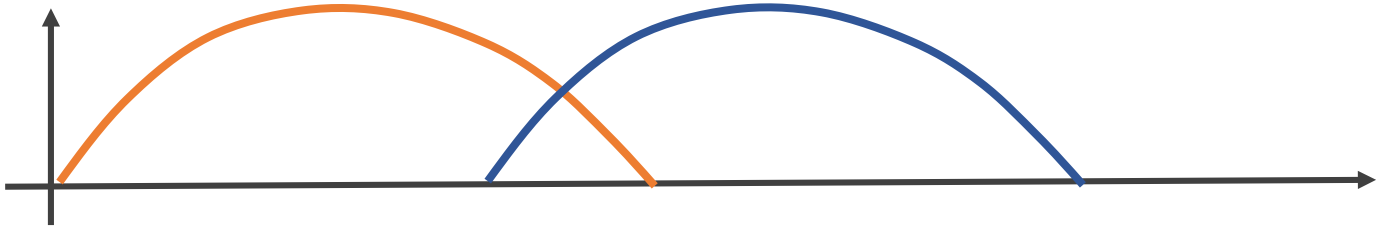
Bike:

Keyboard:

Source domain
with annotations

public

Target domain
without annotations



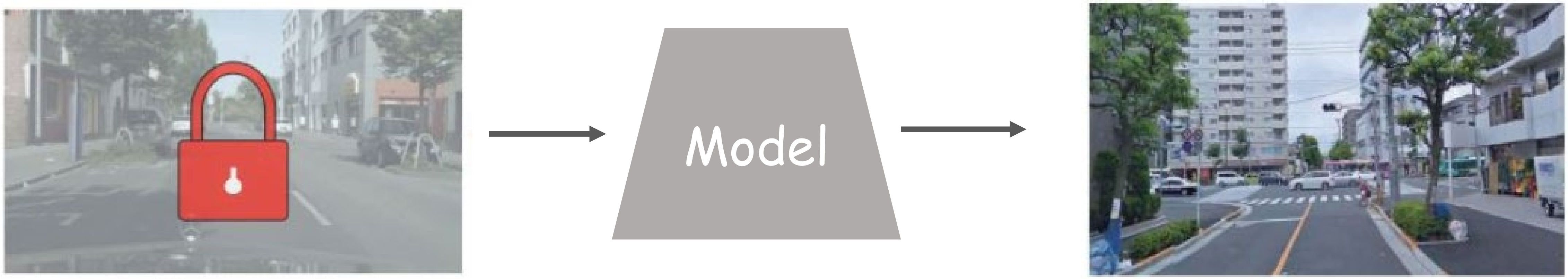
Solution: Domain Alignment

✓ **Goal**: Achieving good performance on the target domain.

Background

- Source-Free Domain Adaptation(SFDA)

Due to data privacy or intellectual property, we can solve the UDA problems by only utilizing the model trained by source domain.

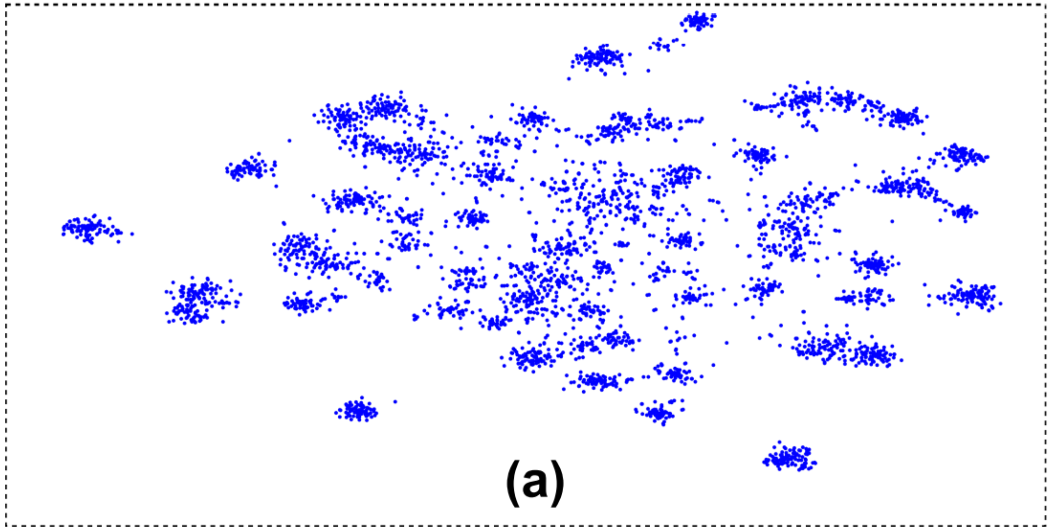


Private Source Domain

Public Target Domain

Motivation

- Previous methods ignore the intrinsic neighborhood structure of the target data in feature space which can be very valuable to tackle SFDA.



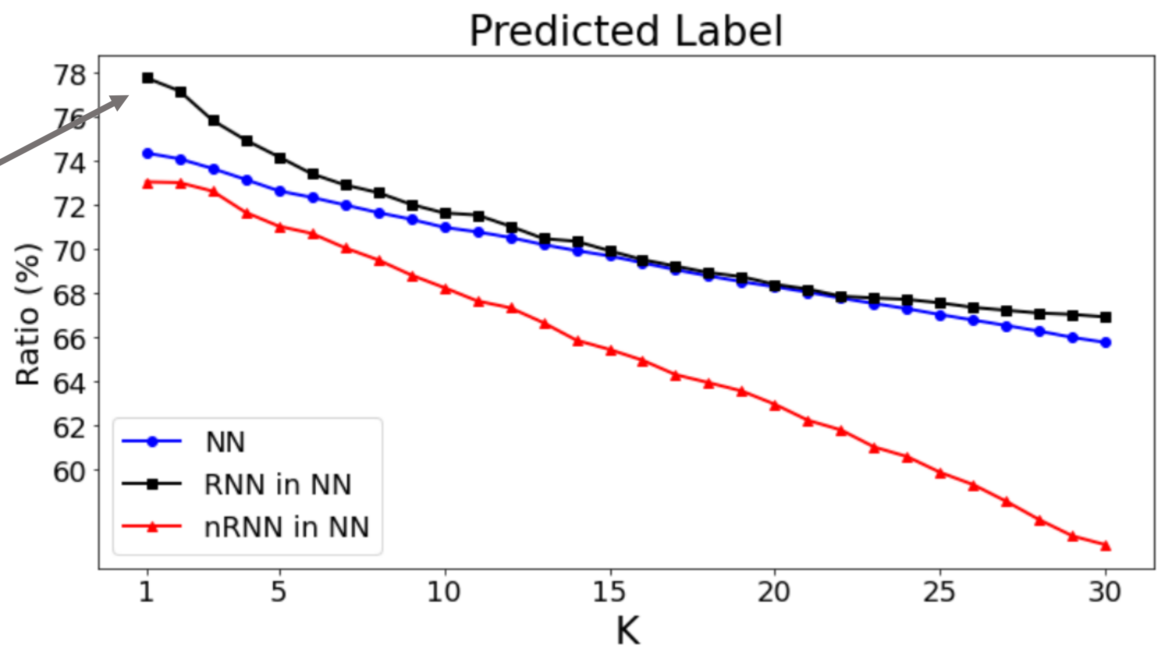
- Observation:

Even though the target data might have shifted in the feature space (due to the covariance shift), target data of the same class is still expected to form a cluster in the embedding space.

reciprocal nearest neighbors (RNN)

- A well-established way to assess the structure of points in high-dimensional spaces is by considering the nearest neighbors of points.

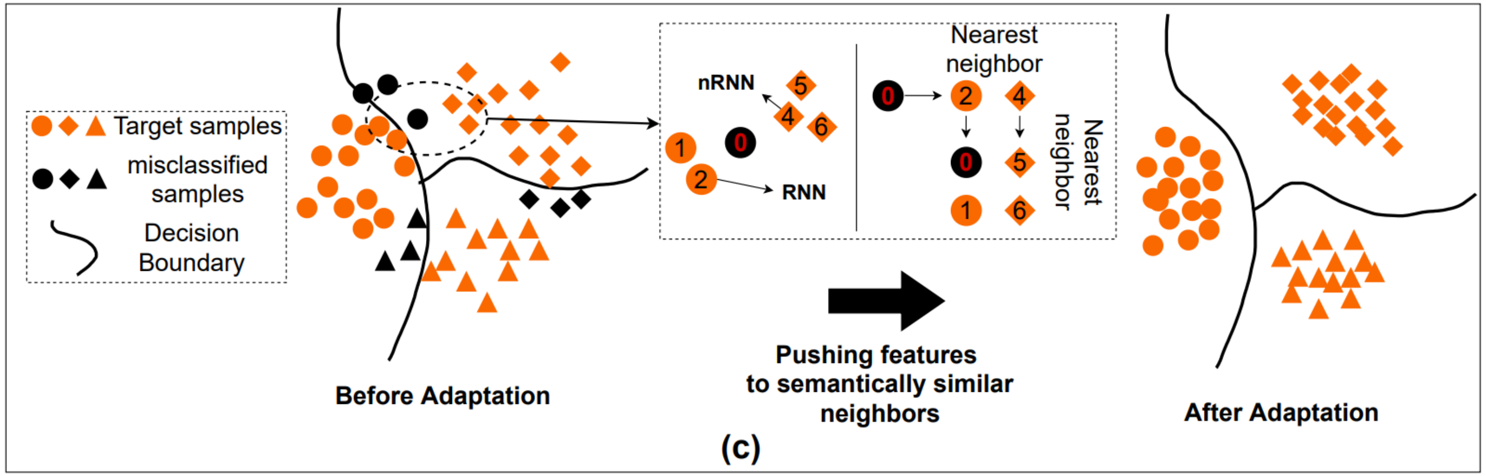
reciprocal nearest neighbors



(b)

Methods-NRC

- The author try to assign different weights to the supervision from nearest neighbors and aims to achieves source-free domain adaptation by encouraging reciprocal neighbors to concord in their label prediction.



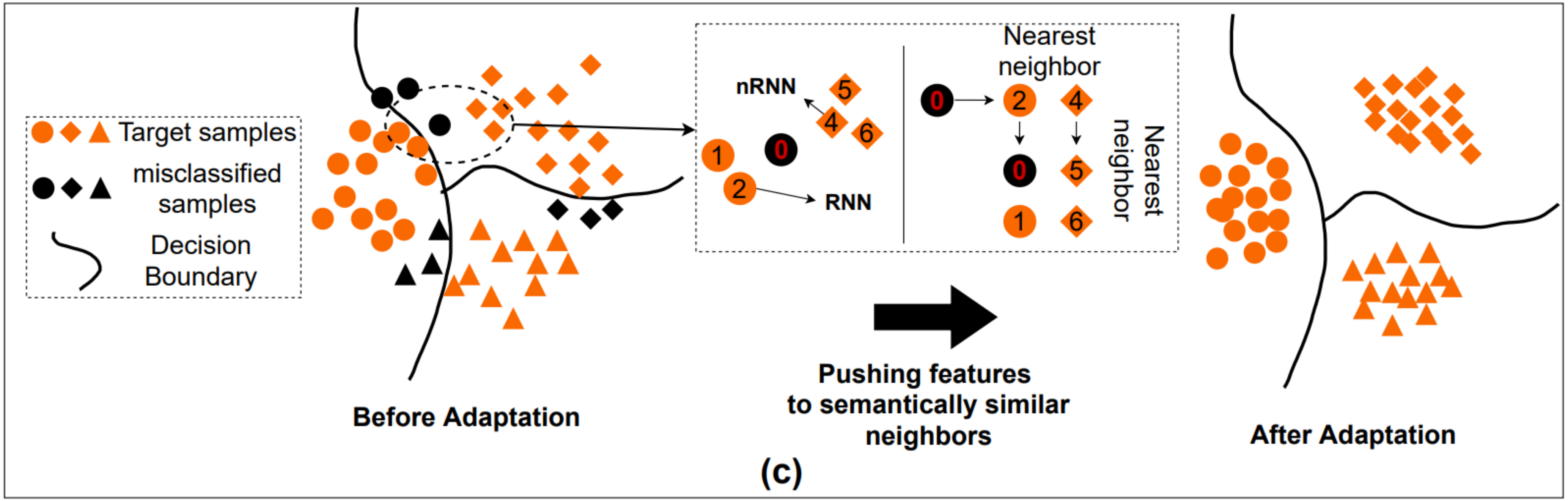
- Objective

$$\mathcal{L} = -\frac{1}{n_t} \sum_{x_i \in \mathcal{D}_t} \sum_{x_j \in \text{Neigh}(x_i)} \frac{D_{sim}(p_i, p_j)}{D_{dis}(x_i, x_j)}$$

- reciprocal nearest neighbors(RNN)

$$j \in \mathcal{N}_K^i \wedge i \in \mathcal{N}_M^j$$

Methods-NRC



- reciprocal nearest neighbors(RNN)

$$j \in \mathcal{N}_K^i \wedge i \in \mathcal{N}_M^j$$

- affinity value

$$A_{i,j} = \begin{cases} 1 & \text{if } j \in \mathcal{N}_K^i \wedge i \in \mathcal{N}_M^j \\ r & \text{otherwise. } 0.1 \end{cases} = \frac{1}{D_{dis}}$$

Methods-NRC

南京航空航天大學
NANJING UNIVERSITY OF AERONAUTICS AND ASTRONAUTICS

- Features memory bank and prediction scores memory bank

$$\mathcal{F} = [z_1, z_2, \ldots, z_{n_t}] \qquad \mathcal{S} = [p_1, p_2, \ldots, p_{n_t}]$$

- Consistent prediction

$$\mathcal{L}_{\mathcal{N}} = -\frac{1}{n_t}\sum_i \sum_{k\in\mathcal{N}_K^i} A_{ik}\mathcal{S}_k^\top p_i$$ k is the index of the k-th nearest neighbors of z_i

- self-regularization

  - To further reduce the potential impact of noisy neighbors in N_K, which belong to the different class but still are RNN

$$\mathcal{L}_{self} = -\frac{1}{n_t}\sum_i^{n_t} \mathcal{S}_i^\top p_i$$

- prediction diversity loss

$$\mathcal{L}_{div} = \sum_{c=1}^{C} \mathrm{KL}(\bar{p}_c || q_c), \text{with } \bar{p}_c = \frac{1}{n_t}\sum_i p_i^{(c)}, \text{and } q_{\{c=1,..,C\}} = \frac{1}{C}$$

Methods-NRC

- expanded neighbors of feature z_i

$$E_M(\boldsymbol{z}_i) = \mathcal{N}_M(\boldsymbol{z}_j)\ \forall j \in \mathcal{N}_K(\boldsymbol{z}_i)$$

- The author directly assign a small affinity value r to those expanded neighbors, since they are further than nearest neighbors

$$\mathcal{L}_E = -\frac{1}{n_t}\sum_i \sum_{k\in\mathcal{N}_K^i} \sum_{m\in E_M^k} r\mathcal{S}_m^\top p_i$$

- Final Loss

$$\mathcal{L} = \mathcal{L}_{div} + \mathcal{L}_{\mathcal{N}} + \mathcal{L}_E + \mathcal{L}_{self}$$

Algorithm

Algorithm 1 Neighborhood Reciprocity Clustering for Source-free Domain Adaptation

---

Require: $\mathcal{D}_s$ (only for source model training), $\mathcal{D}_t$
1: Pre-train model on $\mathcal{D}_s$
2: Build feature bank $\mathcal{F}$ and score bank $\mathcal{S}$ for $\mathcal{D}_t$
3: **while** Adaptation **do**
4:     Sample batch $\mathcal{T}$ from $\mathcal{D}_t$
5:     Update $\mathcal{F}$ and $\mathcal{S}$ corresponding to current batch $\mathcal{T}$
6:     Retrieve nearest neighbors $\mathcal{N}$ for each of $\mathcal{T}$
7:     Compute affinity value A ▷ Eq.5
8:     Retrieve expanded neighborhoods E for each of $\mathcal{N}$
9:     Compute loss and update the model ▷ Eq. 9
10: **end while**

Experiments

南京航空航天大學
NANJING UNIVERSITY OF AERONAUTICS AND ASTRONAUTICS

- Office-31

Table 1: Accuracies (%) on Office-31 for ResNet50-based methods.

Method	SF	A→D	A→W	D→W	W→D	D→A	W→A	Avg
MCD [35]	✗	92.2	88.6	98.5	**100.0**	69.5	69.7	86.5
CDAN [24]	✗	92.9	94.1	98.6	**100.0**	71.0	69.3	87.7
MDD [59]	✗	90.4	90.4	98.7	99.9	75.0	73.7	88.0
BNM [4]	✗	90.3	91.5	98.5	**100.0**	70.9	71.6	87.1
DMRL [49]	✗	93.4	90.8	99.0	**100.0**	73.0	71.2	87.9
BDG [53]	✗	93.6	93.6	99.0	**100.0**	73.2	72.0	88.5
MCC [15]	✗	95.6	95.4	98.6	100.0	72.6	73.9	89.4
SRDC [42]	✗	95.8	95.7	99.2	100.0	76.7	77.1	90.8
RWOT [51]	✗	94.5	95.1	**99.5**	100.0	**77.5**	77.9	90.8
RSDA-MSTN [10]	✗	95.8	**96.1**	99.3	**100.0**	77.4	**78.9**	**91.1**
SHOT [21]	✓	94.0	90.1	98.4	99.9	74.7	74.3	88.6
3C-GAN [20]	✓	92.7	93.7	98.5	99.8	75.3	77.8	89.6
NRC	✓	**96.0**	90.8	99.0	**100.0**	75.3	75.0	89.4

- Office-Home

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

Method	SF	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**
MCD [35]	✗	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
CDAN [24]	✗	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
SAFN [52]	✗	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
Symnets [58]	✗	47.7	72.9	78.5	64.2	71.3	74.2	64.2	48.8	79.5	74.5	52.6	82.7	67.6
MDD [59]	✗	54.9	73.7	77.8	60.0	71.4	71.8	61.2	53.6	78.1	72.5	**60.2**	82.3	68.1
TADA [47]	✗	53.1	72.3	77.2	59.1	71.2	72.1	59.7	53.1	78.4	72.4	60.0	82.9	67.6
BNM [4]	✗	52.3	73.9	80.0	63.3	72.9	74.9	61.7	49.5	79.7	70.5	53.6	82.2	67.9
BDG [53]	✗	51.5	73.4	78.7	65.3	71.5	73.7	65.1	49.7	81.1	74.6	55.1	84.8	68.7
SRDC [42]	✗	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
RSDA-MSTN [10]	✗	53.2	77.7	81.3	66.4	74.0	76.5	67.9	53.0	82.0	75.8	57.8	85.4	70.9
SHOT [21]	✓	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
NRC	✓	**57.7**	**80.3**	**82.0**	68.1	**79.8**	**78.6**	65.3	**56.4**	**83.0**	71.0	58.6	**85.6**	**72.2**

Experiments

□ VisDa

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

Method	SF	plane	bcycl	bus	car	horse	knife	mcycl	person	plant	sktbrd	train	truck	Per-class
ADR [34]	✗	94.2	48.5	84.0	72.9	90.1	74.2	92.6	72.5	80.8	61.8	82.2	28.8	73.5
CDAN [24]	✗	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+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
SAFN [52]	✗	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
SWD [19]	✗	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
MDD [59]	✗	-	-	-	-	-	-	-	-	-	-	-	-	74.6
DMRL [49]	✗	-	-	-	-	-	-	-	-	-	-	-	-	75.5
MCC [15]	✗	88.7	80.3	80.5	71.5	90.1	93.2	85.0	71.6	89.4	73.8	85.0	36.9	78.8
STAR [26]	✗	95.0	84.0	**84.6**	73.0	91.6	91.8	85.9	78.4	94.4	84.7	87.0	42.2	82.7
RWOT [51]	✗	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
3C-GAN [20]	✓	94.8	73.4	68.8	74.8	93.1	95.4	88.6	**84.7**	89.1	84.7	83.5	48.1	81.6
SHOT [21]	✓	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
NRC	✓	**96.8**	**91.3**	82.4	62.4	**96.2**	**95.9**	86.1	80.6	**94.8**	**94.1**	90.4	59.7	**85.9**

□ 3D point cloud dataset

Table 4: Accuracies (%) on PointDA-10. *The results except ours are from PointDAN [30].*

	SF	Model→Shape	Model→Scan	Shape→Model	Shape→Scan	Scan→Model	Scan→Shape	Avg
MMD [25]	✗	57.5	27.9	40.7	26.7	47.3	54.8	42.5
DANN [6]	✗	58.7	29.4	42.3	30.5	48.1	56.7	44.2
ADDA [44]	✗	61.0	30.5	40.4	29.3	48.9	51.1	43.5
MCD [35]	✗	62.0	31.0	41.4	31.3	46.8	59.3	45.3
PointDAN [30]	✗	64.2	**33.0**	47.6	**33.9**	49.1	64.1	48.7
Source-only		43.1	17.3	40.0	15.0	33.9	47.1	32.7
NRC	✓	**64.8**	25.8	**59.8**	26.9	**70.1**	**68.1**	**52.6**

Experiments-Analysis

Table 5: Ablation study of different modules on Office-Home (**left**) and VisDA (**middle**), comparison between using expanded neighbors and larger nearest neighbors (**right**).

$\mathcal{L}_{div}$	$\mathcal{L}_{\mathcal{N}}$	$\mathcal{L}_E$	$\mathcal{L}_{\hat{\mathbf{E}}}$	A	Avg
source model					59.5
✓					62.1
✓	✓				67.1
✓	✓			✓	69.1
✓	✓	✓			65.2
✓	✓	✓		✓	**72.2**
✓	✓		✓	✓	69.1

$\mathcal{L}_{div}$	$\mathcal{L}_{\mathcal{N}}$	$\mathcal{L}_E$	$\mathcal{L}_{\hat{\mathbf{E}}}$	A	Acc
source model					44.6
✓					47.8
✓	✓				74.6
✓	✓			✓	81.5
✓	✓	✓			61.2
✓	✓	✓		✓	**85.9**
✓	✓		✓	✓	82.0

Method&Dataset	Acc
VisDA (K=M=5)	**85.9**
VisDA w/o E (K=30)	84.0
OH (K=3,M=2)	**72.2**
OH w/o E (K=9)	69.5

Experiments-Analysis

Table 6: Runtime analysis on SHOT and our method. For SHOT, pseudo labels are computed at each epoch. 20%, 10% and 5% denote the percentage of target features which are stored in the memory bank.

VisDA	Runtime (s/epoch)	Per-class (%)
SHOT	618.82	82.9
NRC	540.89	85.9
NRC(20% for memory bank)	507.15	85.3
NRC(10% for memory bank)	499.49	85.2
NRC(5% for memory bank)	499.28	85.1

Experiments-Analysis

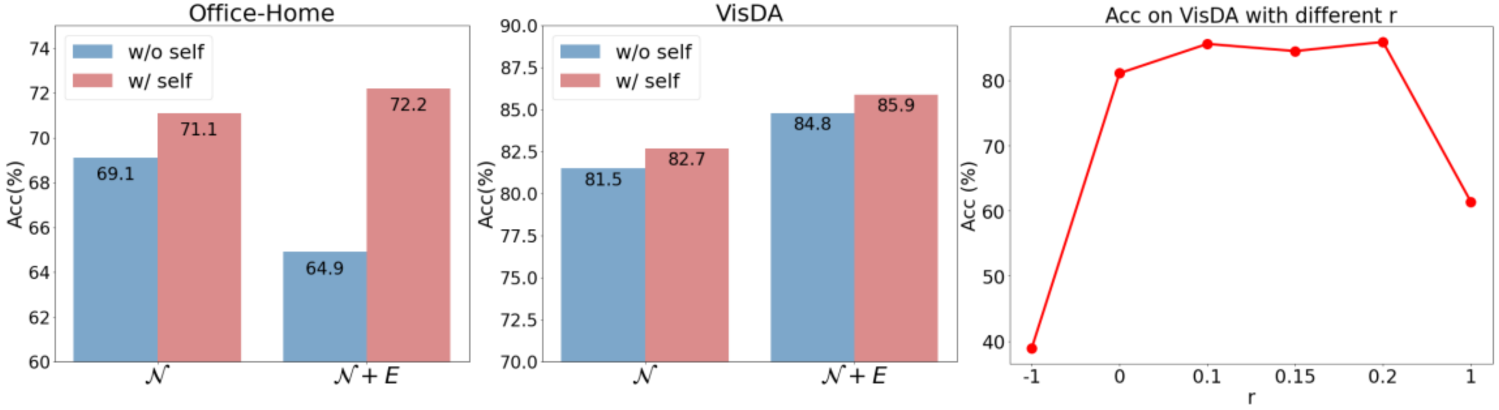


Figure 2: (**Left and middle**) Ablation study of $\mathcal{L}_{self}$ on Office-Home and VisDA respectivel (**Right**) Performance with different r on VisDA.

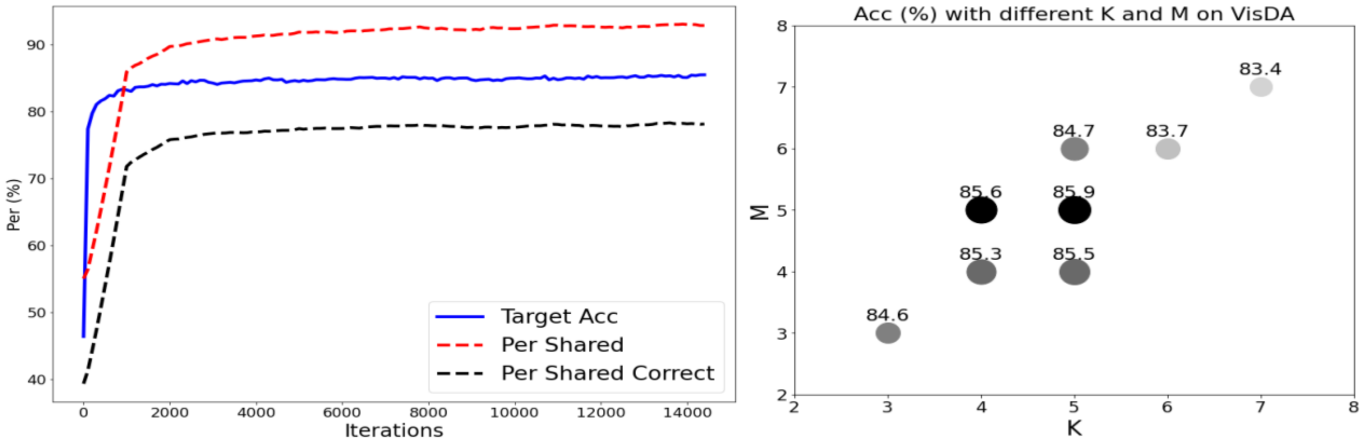
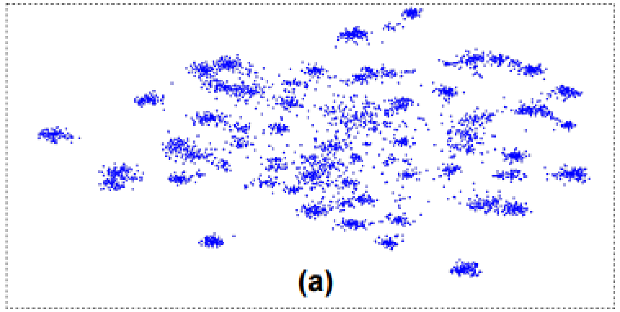


Figure 3: (**Left**) The three curves are (on VisDA): target accuracy (*Blue*), ratio of features which have 5-nearest neighbors all sharing the same predicted label (*dashed Red*), and ratio of features which have 5-nearest neighbors all sharing the same and *correct* predicted label (*dashed Black*). (**Right**) Ablation study on choice of K and M on VisDA.

Experiments-Analysis

NANJING UNIVERSITY OF AERONAUTICS AND ASTRONAUTICS



- Before

- Ar→Rw of Office-Home

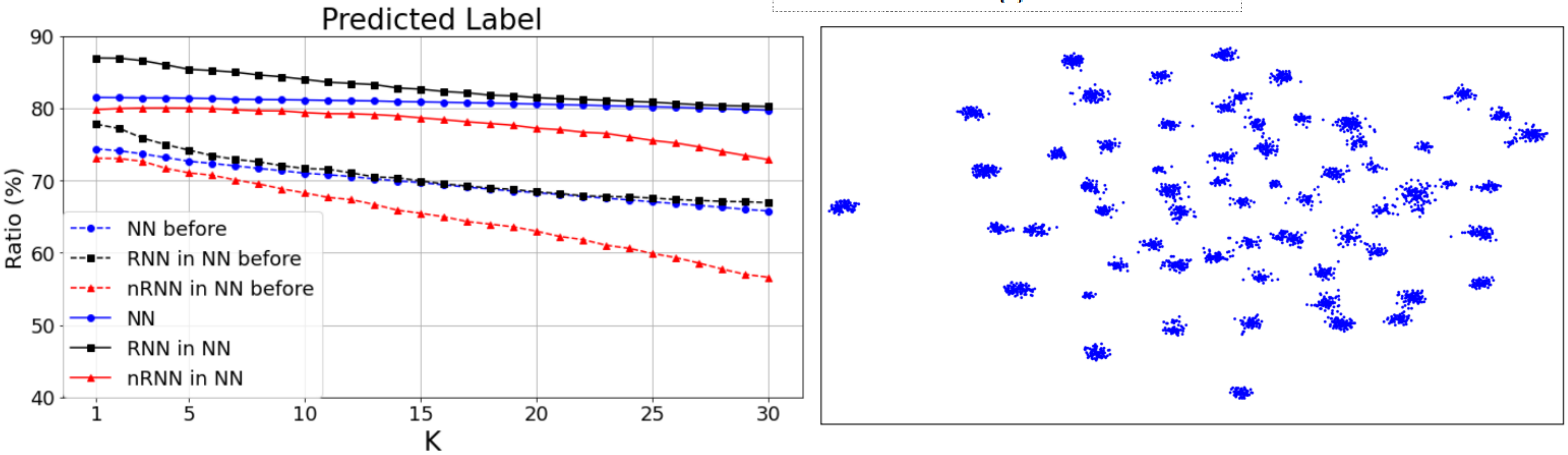


Figure 4: (**Left**) Ratio of different type of nearest neighbor features which have the correct predicted label, before and after adaptation. (**Right**) Visualization of target features after adaptation.

- It shows that after adaptation, the ratio of all types of neighbors having more correct predicted label.

Thanks