Supplementary Materials — SPA: A Graph Spectral Alignment Perspective for Domain Adaptation

In the appendix, we firstly provide more details of experiment setup in Section A, including experimental environment and implementation details. In the Section B, we offer more details of experiments to unsupervised domain adaptation in Section B.1, and then show the extension on semi-supervised domain adaptation in Section B.2. Finally, we further give a detailed model analysis of our SPA model in Section B.3, including robust analysis, parameter sensitivity, the analysis of transferability and discriminability, and finally more feature visualization.

A More Setup

Hardware and Software Configurations. All experiments are conducted on a server with the following configurations:

- Operating System: Ubuntu 20.04.4 LTS
- CPU: Intel(R) Xeon(R) Platinum 8358P CPU @ 2.60GHz, 32 cores, 128 processors
- GPU: NVIDIA GeForce RTX 3090

More Implementation Details. We use PyTorch and tllib toolbox [12] to implement our method and fine-tune ResNet pre-trained on ImageNet [10, 11]. Following the standard protocols for unsupervised domain adaptation in previous methods [20, 22], we use the same backbone networks for fair comparisons. For Office31 and OfficeHome dataset, we use ResNet-50 as the backbone network. For VisDA2017 and DomainNet dataset, we use ResNet-101 as the backbone network. Following previous work [20], we adopt mini-batch stochastic gradient descent (SGD) to learn the feature encoder by fine-tuning from the ImageNet pre-trained model with the learning rate 0.001, and new layers, as bottleneck layer and classification layer. The learning rates of the layers trained from scratch are set to be 0.01. We use the the same learning rate schedule in [20, 21], including a learning rate scheduler with a momentum of 0.9, a weight decay of 0.005, the bottleneck size of 256, and batch size of 32.

We report main experimental results with the average accuracy over 5 random trials with the initial seed 0. For transductive unsupervised domain adaptation, the reported accuracy is computed on the complete unlabeled target data, following established protocol for UDA [7, 21, 14, 3, 20]. For inductive unsupervised domain adaptation on DomainNet, the reported accuracy is computed on the provided test dataset.. We use a standard batch size of 32 for both source and target in all experiments and for all variants of our method. The reverse validation [19, 30] is conducted to select hyper-parameters. For both unsupervised domain adaptation (UDA) and semi-supervised domain adaptation (SSDA) scenarios, we fix the coefficient of $\mathcal{L}_{nas}$ as 0.2 and the coefficient of $\mathcal{L}_{gsa}$ as 1.0, while we will offer a sensitivity analysis for this two coefficients in the following section. More details refer to our code in the supplemental materials.

B More Experiments

B.1 Unsupervised Domain Adaptation

In the main paper, we present the classification accuracy results on VisDA2017 dataset for unsupervised domain adaptation and leave out per-category accuracy details. In the appendix, Table 1, we give the full table on VisDA2017, using ResNet101 as backbone. Looking into this table, we can find that our SPA model consistently outperforms most of domain adaptation methods. For classic baselines, we improve DANN [7] by 30.3%, and CDAN [21] by 14 %. For recent and state-of-the-art baselines, our results is 4% higher than NWD [2] and 0.5 % better than FixBi [22]. Our SPA model ranks top in 6 out of 12 categories, ranks top 2 in 9 out of 12 categories, and SPA also achieves the best classification accuracy in total.

Furthermore, in the main paper, we present the classification accuracy results on original DomainNet with 365 categories. While the original DomainNet dataset has noisy labels, the previous work [25] use a subset of it that contains 126 categories from C, P, R and S, 4 domains in total, which we refer to as DomainNet126. In the appendix, Table 2, we show the results on DomainNet126. Our SPA model consistently ranks top among 12 tasks across 4 domains and achieves the best accuracy of 77.1 %, which is 12.3 % better than the second one. For classic baselines, we improve DANN [7] by 20.2 %, and CDAN [21] by 16 %.

Table 1: Per-category Accuracy (%) on VisDA2017 for unsupervised domain adaptation, using ResNet101 as backbone. All the results are based on ResNet101 except those with mark †, which are based on ResNet50. The best accuracy is indicated in **bold** and the second best one is underlined.

Method	aero	bike	bus	car	horse	knife	motor	person	plant	skate	train	truck	Avg.
Source Only [10]	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
DANN [7]	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
CDAN [21]	85.2	66.9	83.0	50.8	84.2	74.9	88.1	74.5	83.4	76.0	81.9	38.0	73.7
MixMatch [1]	93.9	71.8	93.5	82.1	95.3	0.7	90.8	38.1	94.5	96.0	86.3	2.2	70.4
BSP [3]	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
NPL [17]	90.9	74.6	73.2	55.8	89.6	64.6	86.8	68.7	90.7	64.8	89.5	47.7	74.7
GVB [5]	-	-	-	-	-	-	-	-	-	-	-	-	75.3†
MCC [14]	92.2	82.9	76.8	66.6	90.9	78.5	87.9	73.8	90.1	76.1	87.1	41.0	78.8
BNM [4]	93.6	68.3	78.9	70.3	91.1	82.8	93.0	78.7	90.9	76.5	89.1	40.9	79.5
ATDOC [20]	95.3	84.7	82.4	75.6	95.8	**97.7**	88.7	76.6	94.0	91.7	**91.5**	**61.9**	86.3
FixBi [22]	96.1	87.8	**90.5**	**90.3**	96.8	95.3	92.8	**88.7**	**97.2**	94.2	90.9	25.7	87.2
SDAT [24]	94.8	77.1	82.8	60.9	92.3	95.2	91.7	79.9	89.9	91.2	88.5	41.2	82.1
NWD [2]	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
SPA (Ours)	**98.5**	**92.2**	86.3	63.0	**97.5**	95.4	**93.5**	80.7	**97.2**	**95.2**	91.1	61.4	**87.7**

Table 2: Classification Accuracy (%) on DomainNet126 for unsupervised domain adaptation, using ResNet101 as backbone. The best accuracy is indicated in **bold** and the second best one is underlined.

Method	C→P	C→R	C→S	P→C	P→R	P→S	R→C	R→P	R→S	S→C	S→P	S→R	Avg.
Source Only [10]	38.4	50.9	43.9	50.3	66.7	39.9	54.6	57.9	43.7	52.5	43.5	48.3	49.2
DANN [7]	46.5	58.2	51.6	52.7	64.2	52.9	61.7	60.3	53.9	62.7	56.7	61.6	56.9
MCD [26]	43.7	55.7	47.6	51.9	67.8	45.0	52.9	57.3	40.4	56.3	50.8	56.8	52.3
BSP [3]	45.7	58.7	55.5	48.6	65.2	48.6	55.2	60.8	48.6	56.8	55.8	61.4	55.1
CDAN [21]	50.9	61.6	54.8	59.4	68.5	55.5	70.4	66.9	57.7	64.2	59.1	64.3	61.1
SAFN [29]	50.0	58.7	52.4	56.3	73.7	53.5	55.8	64.8	48.5	60.7	59.5	64.3	58.2
RSDA [9]	45.5	56.6	46.6	45.7	60.4	48.6	54.6	61.5	50.9	56.1	54.0	58.6	53.4
PAN [28]	58.8	65.2	54.6	57.5	70.5	53.1	67.6	66.7	55.9	64.4	60.2	66.6	61.8
MemSAC [15]	53.6	66.5	58.8	63.2	71.2	58.1	73.2	70.5	61.5	68.8	64.1	67.6	64.8
SPA (Ours)	**73.5**	**84.0**	**70.6**	**76.5**	**85.9**	**71.9**	**76.6**	**77.0**	**69.8**	**78.3**	**76.8**	**83.9**	**77.1**

Table 3: Classification Accuracy (%) on DomainNet126 for 1-shot and 3-shot semi-supervised domain adaptation, using ResNet34 as backbone. The best accuracy is indicated in **bold** and the second best one is underlined.

Method	C→S		P→C		P→R		R→C		R→P		R→S		S→P		Avg.	
	1-	3-	1-	3-	1-	3-	1-	3-	1-	3-	1-	3-	1-	3-	1-	3-
S + T [10]	54.8	57.9	59.2	63.0	73.7	75.6	61.2	63.9	64.5	66.3	52.0	56.0	60.4	62.2	60.8	63.6
DANN [7]	52.8	55.4	70.3	72.2	56.3	59.6	58.2	59.8	61.4	62.8	52.2	54.9	57.4	59.9	58.4	60.7
ENT [8]	54.6	60.0	65.4	71.1	75.0	78.6	65.2	71.0	65.9	69.2	52.1	61.1	59.7	62.1	62.6	67.6
MME [25]	56.3	61.8	69.0	71.7	76.1	78.5	70.0	72.2	67.7	69.7	61.0	61.9	64.8	66.8	66.4	68.9
APE [16]	56.7	63.1	72.9	**76.7**	76.6	79.4	70.4	76.6	70.8	72.1	63.0	67.8	64.5	66.1	67.6	71.7
BiAT [13]	57.9	61.5	71.6	74.6	77.0	78.6	73.0	74.9	68.0	68.8	58.5	62.1	63.9	67.5	67.1	69.7
MixMatch [1]	59.3	62.7	66.7	68.7	74.8	78.8	69.4	72.6	67.8	68.8	62.5	65.6	66.3	67.1	66.7	69.2
NPL [17]	62.5	64.5	67.6	70.7	78.3	79.3	70.9	72.9	69.2	70.7	62.0	64.8	67.0	68.6	68.2	70.2
BNM [4]	58.4	62.6	69.4	72.7	77.0	79.5	69.8	73.7	69.8	71.2	61.4	65.1	64.1	67.6	67.1	70.3
MCC [14]	56.8	60.5	62.8	66.5	75.3	76.5	65.5	67.2	66.9	68.1	57.6	59.8	63.4	65.0	64.0	66.2
ATDOC [20]	65.6	66.7	72.8	74.2	**81.2**	81.2	74.9	76.9	71.3	72.5	65.2	64.6	68.7	70.8	71.4	72.4
AESL [23]	59.5	65.2	74.6	76.2	72.4	74.1	74.8	**77.3**	**79.4**	**80.5**	**66.2**	**69.5**	66.6	69.6	70.5	**73.2**
SPA (Ours)	**65.9**	**67.0**	**74.8**	76.5	81.1	**82.3**	**75.3**	76.0	71.8	72.2	65.8	67.2	**69.8**	**71.1**	**72.1**	**73.2**

Table 4: Classification Accuracy (%) on OfficeHome for 1-shot and 3-shot semi-supervised domain adaptation, using ResNet34 as backbone. The best accuracy is indicated in **bold** and the second best one is underlined.

Method (1-shot)	A→C	A→P	A→R	C→A	C→P	C→R	P→A	P→C	P→R	R→A	R→C	R→P	Avg.
$S + T$ [10]	52.1	78.6	66.2	74.4	48.3	57.2	69.8	50.9	73.8	70.0	56.3	68.1	63.8
DANN [7]	53.1	74.8	64.5	68.4	51.9	55.7	67.9	52.3	73.9	69.2	54.1	66.8	62.7
ENT [8]	53.6	81.9	70.4	79.9	51.9	63.0	75.0	52.9	76.7	73.2	63.2	73.6	67.9
MME [25]	61.9	82.8	71.2	79.2	57.4	64.7	75.5	59.6	77.8	74.8	**65.7**	74.5	70.4
APE [16]	60.7	81.6	72.5	78.6	58.3	63.6	**76.1**	53.9	75.2	72.3	63.6	69.8	68.9
CDAC [18]	61.9	**83.1**	72.7	**80.0**	59.3	64.6	75.9	**61.2**	78.5	**75.3**	64.5	75.1	71.0
SPA (Ours)	**62.3**	76.7	**79.0**	66.6	**77.3**	**76.4**	65.7	59.1	**80.7**	71.4	65.2	**84.1**	**72.0**

Method (3-shot)	A→C	A→P	A→R	C→A	C→P	C→R	P→A	P→C	P→R	R→A	R→C	R→P	Avg.
$S + T$ [10]	55.7	80.8	67.8	73.1	53.8	63.5	73.1	54.0	74.2	68.3	57.6	72.3	66.2
DANN [7]	57.3	75.5	65.2	69.2	51.8	56.6	68.3	54.7	73.8	67.1	55.1	67.5	63.5
ENT [8]	62.6	85.7	70.2	79.9	60.5	63.9	79.5	61.3	79.1	76.4	64.7	79.1	71.9
MME [25]	64.6	85.5	71.3	80.1	64.6	65.5	79.0	63.6	79.7	76.6	67.2	79.3	73.1
APE [16]	66.4	**86.2**	73.4	**82.0**	65.2	66.1	**81.1**	63.9	80.2	76.8	66.6	79.9	74.0
CDAC [18]	**67.8**	85.6	72.2	81.9	67.0	67.5	80.3	**65.9**	80.6	**80.2**	**67.4**	81.4	74.2
SPA (Ours)	63.1	81.0	**80.2**	68.5	**81.7**	**77.5**	69.5	65.2	**82.0**	73.9	67.2	**87.0**	**74.7**

B.2 Semi-supervised Domain Adaptation

We also extend our SPA model to semi-superivsed domain adaptation (SSDA) scenario and conduct experiments on 1-shot and 3-shot setting, and $S + T$ in SSDA task means the model trained only by the labeled source and target data.

We present the classificaition accuracy results on DomainNet126 and OfficeHome datasets for SSDA scenario in the Table 3 and Table 4 respectively. Looking at the details, Table 3 shows the classification results for 1-shot and 3-shot SSDA setting on DomainNet126 dataset. For the 1-shot setting, our SPA model can improve DANN [7] by 13.7 % and ENT [8] by 9.5 %. our SPA consistently ranks top among 4 out of 7 tasks and ranks top 2 among all tasks, achieving the best accuracy of 72.1 %, which is better 1.6 % then the second one. For the 3-shot setting, our SPA model can improve DANN [7] by 12.5 % and ENT [8] by 5.6 %. SPA achieves the best accuracy of 73.2 %, comparable with recent work AESL [23].

Furthermore, Table 4 shows the classification results for 1-shot and 3-shot SSDA setting on OfficeHome dataset. To verify that our SPA model can also generalize to SSDA scenario, we compare SPA with several classic and recent baselines. The first section of the table shows our SPA model can improve DANN [7] by 9.3 % and ENT [8] by 4.1 % in the 1-shot setting. The second section shows our SPA model can improve DANN [7] by 11.2 % and ENT [8] by 2.8 % in the 3-shot setting. This shows that our SPA model can greatly improve the classic baselines and achieve comparable results with CDAC [18].

B.3 Model Analysis

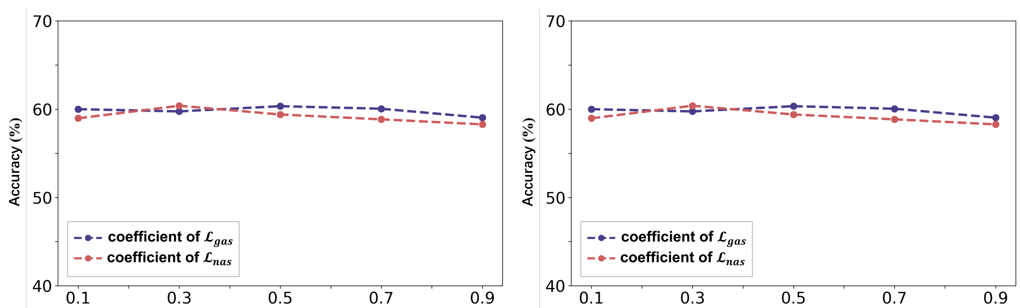
Robustness Analysis. In the main paper, we have already verified the robustness of SPA to different graph structures on OfficeHome dataset. In the appendix, we further show the experimental results on Office31 dataset in Table 5. Similarly, we conduct experiments on different types of Laplacian matrix, similarity metric of graph relations, and k number of nearest neighbors of KNN classification methods. The Laplacian matrices are chosen from the random walk laplacian matrix $\mathbf{L}_{rwk} = \mathbf{D}^{-1}\mathbf{A}$, and the symmetrically normalized Laplacian matrix $\mathbf{L}_{sym} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$, where $\mathbf{D}$ denotes the degree matrix based on the adjacency matrix $\mathbf{A}$. In addition, the similarity metrics are chosen from cosine similarity and Gaussian similarity, and different $k = 3, 5$ when applying KNN classification algorithm. From Table 5, We can find that different types of Laplacian matrix still lead to comparable results. As for the similarity metric, the Gaussian similarity brings better performance than cosine similarity. On Office31 dataset, 5-NN graphs is superior than 3-NN graphs when combining with $\mathbf{L}_{rwk}$, and comparable when combining with $\mathbf{L}_{sym}$. For all these aforementioned experiments results, the differences between them are within 1% around, confirming the robustness of SPA.

Table 5: Classification Accuracy (%) on OfficeHome for unsupervised domain adaptation. The table is divided into three sections corresponding to robustness analysis, and parameter sensitivity, each separated by a double horizontal line.

Method	A→D	A→W	D→A	D→W	W→A	W→D	Avg.
$\beta = 0.1$	94.2	95.1	76.6	98.9	78.6	99.6	90.5
$\beta = 0.3$	94.0	96.2	76.9	98.9	79.3	99.8	90.8
$\beta = 0.5$	94.0	96.4	77.9	99.0	78.0	99.8	90.8
$\beta = 0.7$	94.4	96.0	79.0	98.6	80.3	100.	91.4
$\beta = 0.8$	94.2	95.7	76.1	98.6	78.3	99.8	90.5
$\mathbf{L}_{rwk}$ w/ cos	93.8	95.0	76.3	98.6	79.7	100.	90.6
$\mathbf{L}_{rwk}$ w/ $gauss$	95.0	95.6	78.2	98.6	80.0	99.8	91.2
$\mathbf{L}_{sym}$ w/ cos	93.8	93.8	78.5	98.6	79.9	100.	90.8
$\mathbf{L}_{sym}$ w/ $gauss$	93.8	95.6	79.4	98.6	78.8	99.8	91.0
$\mathbf{L}_{rwk}$ w/ $k = 3$	93.8	95.2	78.3	98.9	80.4	99.8	91.1
$\mathbf{L}_{rwk}$ w/ $k = 5$	93.8	95.0	76.3	98.6	79.7	100.	90.6
$\mathbf{L}_{sym}$ w/ $k = 3$	94.0	95.8	75.2	99.0	80.5	100.	90.7
$\mathbf{L}_{sym}$ w/ $k = 5$	93.8	93.8	78.5	98.6	79.9	100.	90.8

Parameter Sensitivity. In the main paper, we have already analyzed the experiments on the hyperparameter β of exponential moving averaging strategy for memory updates and verify that SPA is insensitive to this hyperparameter. Here, we show the experimental results on Office31 dataset in Table 5. Similarly, we design experiments on the hyperparameter β of exponential moving averaging strategy for memory updates, choosing $\beta = 0.1, 0.3, 0.5, 0.7, 0.9$ respectively. These results are based on DANN [7]. From the series of results, we can find that in Office31 dataset, the choice of $\beta = 0.7$ outperforms than others. In addition, the differences between these results are within 1.0%, which means that SPA is relatively stable to this hyperparameter.

Furthermore, we design experiments for the coefficient of $\mathcal{L}_{nas}$ and the coefficient of $\mathcal{L}_{gsa}$ to analyze the stability of SPA. The experimental results are shown in the Figure 2. These results are based on DANN [7]. Fixing the coefficient of $\mathcal{L}_{nas} = 0.2$, the coefficient of $\mathcal{L}_{gsa}$ changes from 0.1 to 0.9. Fixing the coefficient of $\mathcal{L}_{gsa} = 1.0$, the coefficient of $\mathcal{L}_{nas}$ changes from 0.1 to 0.9. From the series of results, we can find that in OfficeHome dataset, the choice of different coefficients result in similar results, which means that SPA is insensitive to these coefficients.

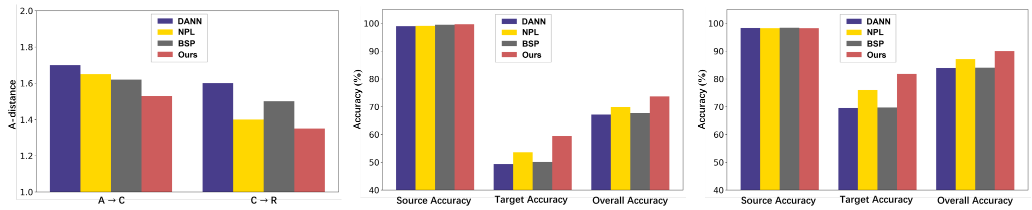


(a) A → C

(b) C → R

Figure 1: Parameter Sensitivity. The line plot of the coefficient of $\mathcal{L}_{nas}$ and the coefficient of $\mathcal{L}_{gsa}$ change from 0.1 to 0.9, leading to different accuracy results. The experiments are conducted on OfficeHome dataset, where (a) is A → C setting and (b) is C → R setting.

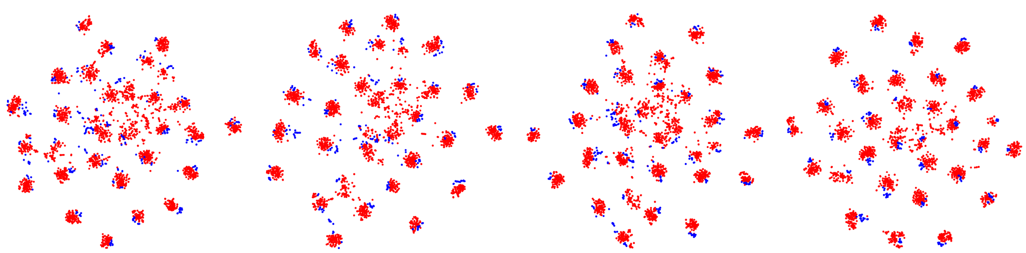
Transferability and Discriminability. The $\mathcal{A}$-distance [27] measures the distribution discrepancy that is defined as $d_{\mathcal{A}} = 2(1 - 2\epsilon)$, where ϵ is the classifier loss to discriminate the souce and target domains. Smaller $\mathcal{A}$-distance indicates better domain-invariant features. Figure 2a shows that SPA can achieve a lower $d_{\mathcal{A}}$, implying a lower generalization error. Furthermore, following previous work [3], we further offer the source accuracy and target accuracy specifically, in Figure 2b and Figure 2c. We can find that various methods achieve similar results of source accuracy, and SPA can always achieve higher target accuracy. Combined with the experimental results in Figure 2a, this reveals that SPA enhances transferability while still keep a strong discriminability. Back to our Introduction, this means that SPA can find a more suitable utilization of intra-domain information and inter-domain information to properly align target samples.



(a) A-distance (b) A $\rightarrow$ C (c) C $\rightarrow$ R

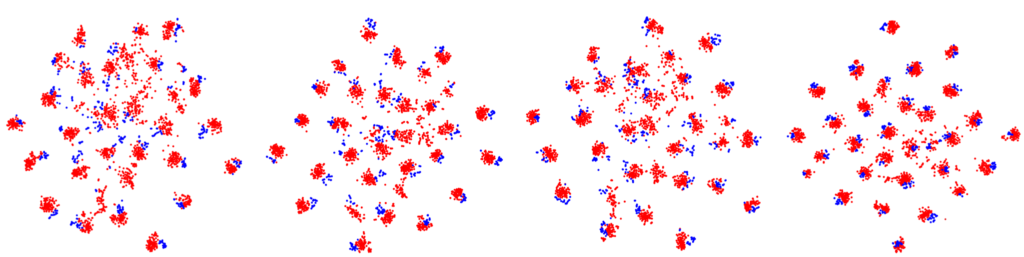
Figure 2: Transferability and Discriminability. We compare SPA with DANN [7], BSP [3] and NPL [17] on OfficeHome dataset, where (a) is $\mathcal{A}$-distance in A $\rightarrow$ C and C $\rightarrow$ R setting, (b) is accuracy results in A $\rightarrow$ C setting and (c) is accuracy results in C $\rightarrow$ R setting.

Feature Visualization. To demonstrate the learning ability of SPA, we visualize the features of DANN [7], BSP [4], NPL [17] and SPA with the t-SNE embedding [6] under the C $\rightarrow$ R setting of OfficeHome Dataset in the main paper, referred as Figure 6. In the appendix, we offer more visualization figures, including Figure 3 in A $\rightarrow$ D setting and Figure 4 in A $\rightarrow$ W setting on Office31 dataset, Figure 5 in A $\rightarrow$ C setting on OfficeHome dataset. According to all the figures, the observations are consistent that the source features and target features learned by SPA are transferred better. These observations imply the superiority of SPA over discriminability and transferability in unsupervised domain adaptation scenario.



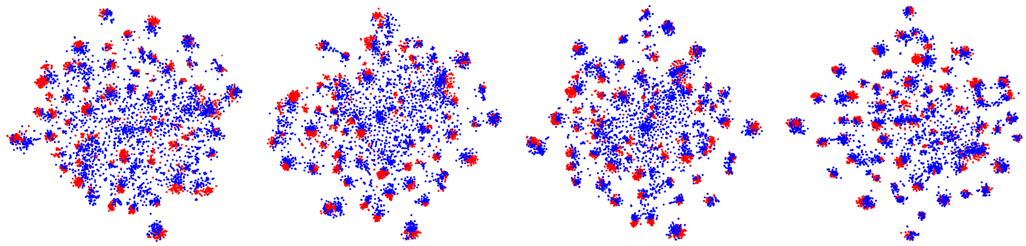
(a) DANN (b) BSP (c) NPL (d) SPA

Figure 3: Feature Visualization. the t-SNE plot of DANN [7], BSP [4], NPL [17], and SPA features on office31 dataset in the A $\rightarrow$ D setting. We use red markers for source domain features and blue markers for target domain features.



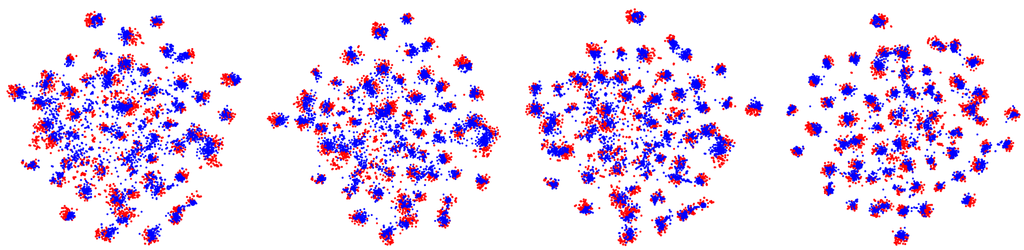
(a) DANN (b) BSP (c) NPL (d) SPA

Figure 4: Feature Visualization. the t-SNE plot of DANN [7], BSP [4], NPL [17], and SPA features on Office31 dataset in the A $\rightarrow$ W setting. We use red markers for source domain features and blue markers for target domain features.



(a) DANN (b) BSP (c) NPL (d) SPA

Figure 5: Feature Visualization. the t-SNE plot of DANN [7], BSP [4], NPL [17], and SPA features on OfficeHome dataset in the A $\rightarrow$ C setting. We use red markers for source domain features and blue markers for target domain features.



(a) DANN (b) BSP (c) NPL (d) SPA

Figure 6: Feature Visualization. the t-SNE plot of DANN [7], BSP [4], NPL [17], and SPA features on OfficeHome dataset in the C $\rightarrow$ R setting. We use red markers for source domain features and blue markers for target domain features.

References

[1] David Berthelot, Nicholas Carlini, Ian J. Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, pages 5050–5060, 2019.

[2] Lin Chen, Huaian Chen, Zhixiang Wei, Xin Jin, Xiao Tan, Yi Jin, and Enhong Chen. Reusing the task-specific classifier as a discriminator: Discriminator-free adversarial domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7181–7190. IEEE, 2022.

[3] Xinyang Chen, Sinan Wang, Mingsheng Long, and Jianmin Wang. Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pages 1081–1090. PMLR, 2019.

[4] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3941–3950. IEEE, 2020.

[5] Shuhao Cui, Shuhui Wang, Junbao Zhuo, Chi Su, Qingming Huang, and Qi Tian. Gradually vanishing bridge for adversarial domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12452–12461. IEEE, 2020.

[6] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *The Journal of Machine Learning Research (JMLR)*, 9(11), 2008.

[7] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by back propagation. In *Proceedings of the 32th International Conference on Machine Learning (ICML)*, pages 1180–1189. PMLR, 2015.

[8] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2004.

[9] Xiang Gu, Jian Sun, and Zongben Xu. Spherical space domain adaptation with robust pseudo-label loss. pages 9101–9110. IEEE, 2020.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778. IEEE, 2016.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *the 14th European Conference on Computer Vision (ECCV)*, volume 9908, pages 630–645. Springer, 2016.

[12] Junguang Jiang, Baixu Chen, Bo Fu, and Mingsheng Long. Transfer-learning-library. `https://github.com/thuml/Transfer-Learning-Library`, 2020.

[13] Pin Jiang, Aming Wu, Yahong Han, Yunfeng Shao, Meiyu Qi, and Bingshuai Li. Bidirectional adversarial training for semi-supervised domain adaptation. In *Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI)*, page 934–940, 2020.

[14] Ying Jin, Ximei Wang, Mingsheng Long, and Jianmin Wang. Minimum class confusion for versatile domain adaptation. In *the 16th European Conference on Computer Vision (ECCV)*, volume 12366, pages 464–480. Springer, 2020.

[15] Tarun Kalluri, Astuti Sharma, and Manmohan Chandraker. Memsac: Memory augmented sample consistency for large scale domain adaptation. In *the 17th European Conference on Computer Vision (ECCV)*, pages 550–568. Springer, 2022.

[16] Taekyung Kim and Changick Kim. Attract, perturb, and explore: Learning a feature alignment network for semisupervised domain adaptation. In *the 16th European Conference on Computer Vision (ECCV)*, pages 591–607. Springer, 2020.

[17] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, page 896, 2013.

[18] Jichang Li, Guanbin Li, Yemin Shi, and Yizhou Yu. Cross-domain adaptive clustering for semi-supervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2505–2514, 2021.

[19] Jian Liang, Ran He, Zhenan Sun, and Tieniu Tan. Exploring uncertainty in pseudo-label guided unsupervised domain adaptation. *Pattern Recognition*, 96, 2019.

[20] Jian Liang, Dapeng Hu, and Jiashi Feng. Domain adaptation with auxiliary target domain-oriented classifier. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16632–16642. IEEE, 2021.

[21] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I. Jordan. Conditional adversarial domain adaptation. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, pages 1647–1657, 2018.

[22] Jaemin Na, Heechul Jung, Hyung Jin Chang, and Wonjun Hwang. Fixbi: Bridging domain spaces for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1094–1103. IEEE, 2021.

[23] Md Mahmudur Rahman, Rameswar Panda, and Mohammad Arif Ul Alam. Semi-supervised domain adaptation with auto-encoder via simultaneous learning. *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, 2023.

[24] Harsh Rangwani, Sumukh K Aithal, Mayank Mishra, Arihant Jain, and R. Venkatesh Babu. A closer look at smoothness in domain adversarial training. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*. PMLR, 2022.

[25] Kuniaki Saito, Donghyun Kim, Stan Sclaroff, Trevor Darrell, and Kate Saenko. Semi-supervised domain adaptation via minimax entropy. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 8050–8058, 2019.

[26] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3723–3732. IEEE, 2018.

[27] Koby Crammer Shai Ben-David, John Blitzer and Fernando Pereira. Analysis of representa-tionsfor domain adaptation. In *Advances in Neural Information Processing Systems (NeurIPS)*, page 137–144, 2007.

[28] Sinan Wang, Xinyang Chen, Yunbo Wang, Mingsheng Long, and Jianmin Wang. Progressive adversarial networks for fine-grained domain adaptation. pages 9213–9222, 2020.

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

[30] Erheng Zhong, Wei Fan, Qiang Yang, Olivier Verscheure, and Jiangtao Ren. Cross validation framework to choose amongst models and datasets for transfer learning. In *Machine Learning and Knowledge Discovery in Databases, European Conference, ECML PKDD*, volume 6323 of *Lecture Notes in Computer Science*, pages 547–562. Springer, 2010.