Supplementary Material for Revisiting Source-Free Domain Adaation: Insights into Representativeness, Generalization, and Variety

The supplementary material contains in-depth information regarding the theoretical analysis, implementation specifics, and supplementary experimental results.

1. Theoretical Details

A hypothesis is a function represented as $h : \mathcal{X} \to \{0, 1\}$. The probability according to the distribution D that a hypothesis h disagrees with the ground truth function f (which can also be a hypothesis) is defined as

$$\ell(h, f, D) = \mathbb{E}_{x \in D}\big[|h(x) - f(x)|\big]. \tag{1}$$

When we intend to denote the source error of a hypothesis associated with source domain D_s, we use the shorthand $\ell_s(h) = \ell(h, D_s) = \ell(h, f_s, D_s)$. Similarly, for the target domain $D_t \triangleq \{x_t^i\}_{i=1}^{n_t}$, we employ the notations $\ell(h, f_t, D_t)$, $\ell(h, D_t)$, and $\ell_t(h)$. Suppose that we can select some unlabeled target data $x_t^i \in D_t$ that the model h can produce highly accurate pseudo-labels $\tilde{y}_t^i$ to them. Thus, the target domain can be divided into two subsets: the selected subset denoted as $D_{t,l} \triangleq \{(x_{t,l}^i, \tilde{y}_{t,l}^i)\}_{i=1}^{n_t^l}$ and the remaining subset indicated as $D_{t,u} \triangleq \{x_{t,u}^i\}_{i=1}^{n_t^u}$.

Definition 1.1 (Based on [1])**.** Given a domain $\mathcal{X}$ with D and D' probability distribution over $\mathcal{X}$, let $\mathcal{H}$ be a hypothesis class on $\mathcal{X}$ and denote by $I(h)$ the set for which $h \in \mathcal{H}$ is the characteristic function; that is, $x \in I(h)$,. The $\mathcal{H}$-divergence between D and D' is

$$d_{\mathcal{H}}(D, D') = 2 \sup_{h \in \mathcal{H}} \big|Pr_D(I(h)) - Pr_{D'}[I(h)]\big| \tag{2}$$

Lemma 1.2 (Based on [1])**.** *Let $\mathcal{H}$ be a hypothesis space on $\mathcal{X}$ with VC dimension d, If $\mathcal{U}$ and $\mathcal{U}'$ are samples of size m from D and D' respectively and $\hat{d}_{\mathcal{H}}(\mathcal{U}, \mathcal{U}')$ is the empirical $\mathcal{H}$-divergence between samples, then for any $\delta \in (0, 1)$, with probability at least $1 - \delta$,*

$$d_{\mathcal{H}}(D, D') \leq \hat{d}_{\mathcal{H}}(\mathcal{U}, \mathcal{U}') + 4\sqrt{\frac{d \log(2m) + \log(\frac{2}{\delta})}{m}} \tag{3}$$

Lemma 1.3. *For any hypothesis $h, h' \in \mathcal{H}$,*

$$\begin{aligned}
\big|\ell(h, h', D_s) - \ell(h, h', D_t)\big| &\leq \sup_{h, h' \in \mathcal{H}} \big|\ell(h, h', D_s) - \ell(h, h', D_t)\big| \\
&= \sup_{h, h' \in \mathcal{H}} \big|Pr_{x \in D_s}[h(x) \neq h'(x)] - Pr_{x \in D_t}[h(x) \neq h'(x)]\big| \\
&= \frac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(D_s, D_t)
\end{aligned}$$

Theorem 1.4. *Given an unlabeled target domain D_t, we can assign the ground truth label y_t^i to some unlabeled target data x_t^i. Thus, the target domain can be divided into two subsets: the selected subset denoted as $D_{t,l}$ and the remaining subset indicated as $D_{t,u}$. We assume that $\mathcal{U}_{t,l}$ and $\mathcal{U}_{t,u}$ are i.i.d. induced from the $D_{t,l}$ and $D_{t,u}$ with size of m, respectively. Let $\ell(\cdot, \cdot)$ be a loss function on a hypothesis and a dataset (for empirical error) or a distribution (for generalization error). If h*

is governed by the parameter θ trained on D_t and belongs to a hypothesis space $\mathcal{H}$ of VC-dimension d, then with probability at least $1-p$ over the choice of samples, the inequality holds,

$$\ell_t(h, D_t) \leq 2\ell(h, D_{t,l}) + \frac{1}{2}\hat{d}_{\mathcal{H}\Delta\mathcal{H}}(D_{t,l}, D_{t,u}) + 2\sqrt{\frac{d\log(2m) + \log(\frac{2}{p})}{m}} + \lambda \tag{4}$$

where $d_{\mathcal{H}\delta\mathcal{H}}(D_l, D_u)$ denotes the distribution divergence, and $\lambda = \min\{\ell_{t,l}(h, f_t), \ell_{t,u}(h, f_t)\}$.

Proof. Recall that $\ell_t(h) = \ell(h, D_t) = \ell(h, f_t, D_t)$, and $D_t = \{D_{t,l}, D_{t,u}\}$. Similarly, we have $\ell_{t,l}(h) = \ell(h, D_{t,l}) = \ell(h, f_{t,l}, D_{t,l})$ and $\ell_{t,u}(h) = \ell(h, D_{t,u}) = \ell(h, f_{t,u}, D_{t,u})$.

$$\begin{aligned}
\ell(h, D_t) &= \mathbb{E}_{x_t \in D_t}\Big[\big|h(x_t) - f_t(x_t)\big|\Big] = \mathbb{E}_{x_t \in \{D_{t,l}+D_{t,u}\}}\Big[\big|h(x_t) - f_t(x_t)\big|\Big] \\
&\leq \mathbb{E}_{x_{t,l} \in D_{t,l}}\Big[\big|h(x_{t,l}) - f_{t,l}(x_{t,l})\big|\Big] + \mathbb{E}_{x_{t,u} \in D_{t,u}}\Big[\big|h(x_{t,u}) - f_{t,u}(x_{t,u})\big|\Big] = \ell(h, D_{t,l}) + \ell(h, D_{t,u}) \\
&= \ell(h, D_{t,l}) + \ell(h, D_{t,u}) + \ell(h, D_{t,l}) - \ell(h, D_{t,l}) + \ell(h, f_{t,u}, D_{t,l}) - \ell(h, f_{t,u}, D_{t,l}) \\
&= 2\ell(h, D_{t,l}) + \big(\ell(h, D_{t,u}) - \ell(h, f_{t,u}, D_{t,l})\big) + \big(\ell(h, f_{t,u}, D_{t,l}) - \ell(h, D_{t,l})\big) \\
&\leq 2\ell(h, D_{t,l}) + \big|\ell(h, f_{t,u}, D_{t,u}) - \ell(h, f_{t,u}, D_{t,l})\big| + \big|\ell(h, f_{t,u}, D_{t,l}) - \ell(h, f_{t,l}, D_{t,l})\big| \\
&\leq 2\ell_{t,l}(h) + \sup_{h, f_{t,u} \in \mathcal{H}} \big|\ell(h, f_{t,u}, D_{t,u}) - \ell(h, f_{t,u}, D_{t,l})\big| + \big|\ell(h, f_{t,u}, D_{t,l}) - \ell(h, f_{t,l}, D_{t,l})\big| \\
&= 2\ell_{t,l}(h) + \frac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(D_{t,l}, D_{t,u}) + \big|\ell(h, f_{t,u}, D_{t,l}) - \ell(h, f_{t,l}, D_{t,l})\big| \\
&\leq 2\ell_{t,l}(h) + \frac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(D_{t,l}, D_{t,u}) + \ell(h, f_{t,u}, D_{t,l}) + \ell(h, f_{t,l}, D_{t,l}) \\
&\leq 2\ell_{t,l}(h) + \frac{1}{2}\hat{d}_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{U}_{t,l}, \mathcal{U}_{t,u}) + 2\sqrt{\frac{d\log(2m) + \log(\frac{2}{\delta})}{m}} + \lambda
\end{aligned}$$

The last step is an application of Lemma 1.2 and 1.3. λ comes from the classification error on $\mathcal{D}_{t,l}$ with classifiers $f_{t,u}$ and $f_{t,l}$. □

2. Details of The Datasets and Implementation

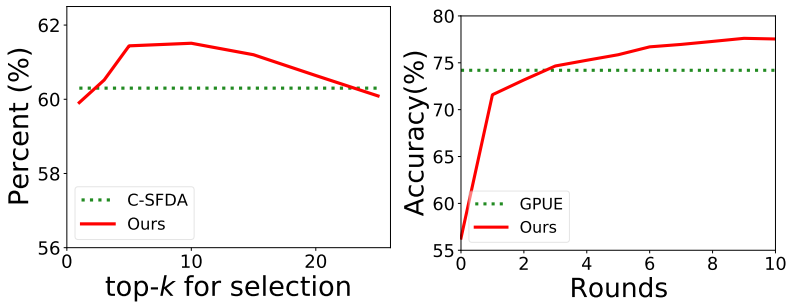
Office-Home is a challenging dataset, which includes 15,500 images from 65 categories in office and home circumstances, consisting of four particularly dissimilar domains: Artistic images (**A**), Clip Art (**C**), Product images (**P**), and Real-World images (**R**). We establish a total of 12 transfer tasks by incorporating all available domains. we configure the top-k sample parameter in Equation 7 to be 5, set the σ in Equation 5 at 50%, setup $\mathcal{M}$ nearest neighbors for target data refinement in Equation 9 to 5, and adjust the β in Equation 11 to 0.9. The training process contains 5 rounds, with each round consisting of 5 epochs.

DomainNet is a substantial domain adaptation dataset, notable for its extensive scale encompassing 6 domains and 345 classes. However, due to the presence of noisy labels in some domains and classes, we follow a specific protocol mentioned in [10]. In line with this protocol, 4 domains (Real, Clipart, Painting, Sketch) and 125 classes are selected. We focus on the adaptation scenarios where the target domain is not real images, and construct 7 scenarios from the 4 domains. we configure the top-k sample parameter in Equation 7 to be 15, set the σ in Equation 5 at 50%, setup $\mathcal{M}$ nearest neighbors for target data refinement in Equation 9 to 5, and adjust the β in Equation 11 to 0.9. The training process contains 10 rounds, with each round consisting of 5 epochs.

VisDA-C is a challenging large-scale synthesis-to-real object recognition dataset that contains 12 classes. The source domain includes 152k synthetic images and the target domain contains 55k real images. we configure the top-k sample parameter in Equation 7 to be 300, set the σ in Equation 5 at 50%, setup $\mathcal{M}$ nearest neighbors for target data refinement in Equation 9 to 5, and adjust the β in Equation 11 to 0.9. The training process contains 10 rounds, with each round consisting of 5 epochs.

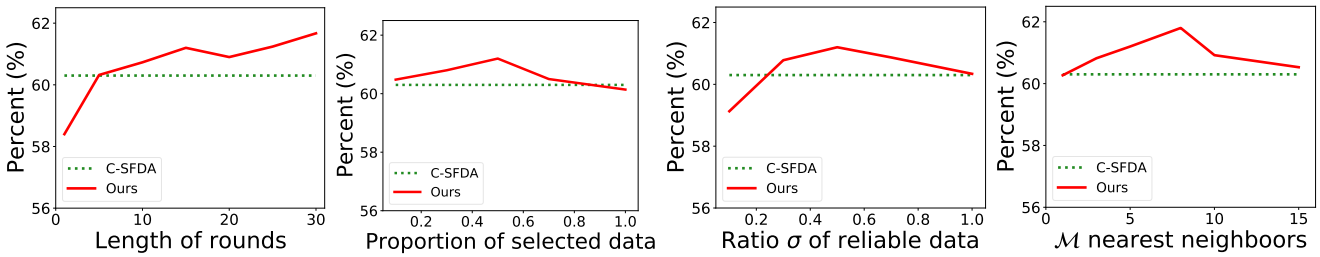
3. Sensitivity to top-k in Equation 7

To verify the impact of the top-k in Equation 7, we conduct experiments on Office-Home with the adaptation task A→C. The value of the top-k varies from 1 to 25. As shown in Figure 1 (a), We have observed that both a small and a large value



(a) (b)

Figure 1. (a) Visualizing the influence of top-k in Equation 7 on the selected pseudo labeled target data with domain task A→C on Office-Home. (b) Visualizing the training behavior of our method on adaptation task Re→Cl on DomainNet.



(a) (b) (c) (d)

Figure 2. As shown from left to right, four figures provide insights into the effect of hyperparameters on our method when applied to the domain task A→C on Office-Home dataset. (a) illustrates the impact of the length of rounds which range from 1 epoch to 30 epochs per round. (b) delves into the impact of the proportion of the selected target subset $D_{t,l}$ within the overall target dataset D_t. (c) describes the effect of the ratio σ of reliable data selected from the entire target data in Equation 5. (d) plots the influence of the $\mathcal{M}$ nearest neighbors selected for target data refinement in Equation 9.

for the top-k lead to decreased performance in our study. In the case of a small top-k value, the performance suffers due to the limited selection of pseudo-labeled data.This limitation negatively impacts the alignment between the pseudo-labeled and unlabeled target data, ultimately affecting the overall performance. Conversely, when employing a larger top-k value, our method tends to select more data, including those with noisy label information. This abundance of noisy data adversely influences the performance, resulting in a decrease in overall effectiveness.

4. Accuracy vs. Round Number Curve for DomainNet

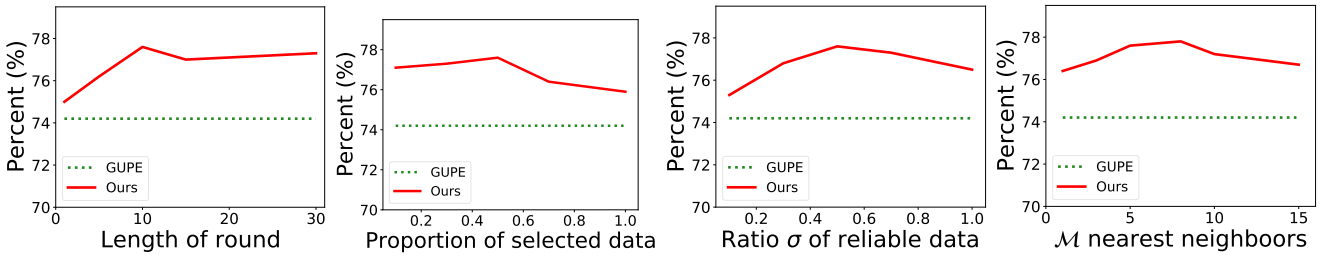
We delve deeper into understanding the training behavior of our approach on DomainNet. As depicted in Figure 1 (b), the accuracy trend of our method shows a gradual improvement. Notably, after four rounds, our approach demonstrates a significant performance boost, surpassing the state-of-the-art GPUE method [10].

5. Hyper-parameter Analysis

We evaluate the sensitivity of hyper-parameters in our method. Namely, the length of the round, the proportion of the selected target subset $D_{t,l}$ within the overall target dataset D_t, the ratio σ of the selected reliable data, and the number of $\mathcal{M}$ nearest neighbors used for pseudo-label refining.

As illustrated in Figure 2 (a) and Figure 3 (a), our performance exhibits continuous improvement with increasing round length. A longer round allows for more comprehensive model training, resulting in enhanced model performance.

In Figure 2 (b) and Figure 3 (b), we observe the performance of our method across various proportions of selected target data with pseudo-label assignment. The performance consistently improves within the range of $[0.1, 0.5]$. However, beyond a ratio of 0.5, there is a slight performance degradation. This decline is attributed to the inclusion of more target data as selected data, which results in lower-quality pseudo-labels and consequently, a deterioration in model performance.



(a) (b) (c) (d)

Figure 3. As shown from left to right, four figures provide insights into the effect of hyperparameters on our method when applied to the domain task Re→Cl on DomainNet dataset. (a) illustrates the impact of the length of rounds which range from 1 epoch to 30 epochs per round. (b) delves into the impact of the proportion of the selected target subset $D_{t,l}$ within the overall target dataset D_t. (c) describes the effect of the ratio σ of reliable data selected from the entire target data in Equation 5. (d) plots the influence of the $\mathcal{M}$ nearest neighbors selected for target data refinement in Equation 9.

Table 1. Sample Selection and Alignment strategies study.

Method	A→C	A→P	A→R
"SHOT++[Selection]" + "Our[Alignment]"	58.7	79.2	81.9
"Our[Selection]" + "AaD[Alignment]"	60.1	79.7	82.3
"Our[Selection]" + "Our[Alignment]" (Ours)	61.2	80.9	82.7
SHOT++ [9]	57.9	79.7	82.5
AaD [15]	59.3	79.3	82.1

As shown in Figure 2 (c) and Figure 3 (c), we observe that a small ratio σ of selected reliable data results in lower performance. This is because a lower ratio σ of selected reliable data leads to a biased estimation of target class centers through the assigned pseudo-labels.

In Figure 2(d) and Figure 3(d), we observe that both a small and a large number of nearest neighbors negatively impact performance. A smaller number of nearest neighbors introduces bias due to the limited contribution of label information from the surrounding data points. On the other hand, a larger number of nearest neighbors includes semantically dissimilar data points, which degrades the quality of the refined labels and ultimately reduces performance.

6. Sample Selection and Alignment Strategies Study

We integrated sample selection and alignment strategies from two state-of-the-art methods, namely SHOT++ [9] and AaD [15]. We denote our methods as "Our[Selection]" + "Our[Alignment]". Specifically, we utilize the sample selection strategy from SHOT++, referred to as "SHOT++[Selection]," and the alignment strategy from AaD, denoted as "AaD[Alignment]". We evaluate the performance of "SHOT++[Selection]" + "Our[Alignment]" and "Our[Selection]" + "AaD[Alignment]" on adaptation tasks, namely A→C, A→P, and A→R, using the Office-Home dataset.

In Table 1 We observed that adopting the sample selection strategy from SHOT++, which uses entropy as the metric to select target data with entropy values larger than the average entropy values over the entire dataset as selected data, results in "SHOT++[Selection]" + "Our[Alignment]" significantly underperforming our results. This phenomenon can be attributed to two reasons. Firstly, "SHOT++[Selection]" heavily relies on the entropy value, which may struggle to distinguish between confident and extremely sharp predictions. Secondly, "SHOT++[Selection]" selects target data only once, and the large number of selected data leads to a low quality of pseudo labels. Therefore, it demonstrates the advantage of our sample selection strategy compared to the sample selection strategy from SHOT++. Combining "Our[Selection]" with "AaD[Alignment]," we observe that it also underperforms compared to our approach. This is mainly due to "AaD[Alignment]" blindly trusting the predicted semantic information of the neighbors, which can lead to negative clustering when these predictions are not very accurate. Additionally, the reliance on inaccurate local clusters can result in suboptimal discriminative representation learning. Therefore, it proves the advantage of our adaptation strategy over the adaptation strategy from AaD.

Table 2. classification Accuracy (%) on VisDA-C (ResNet-101). The best results under SFDA setting are highlighted in bold. Note that "**SF**" means whether the method belongs to SFDA method.

Method	**SF**	plane	bcycl	bus	car	horse	knife	mcycl	person	plant	sktbrd	train	truck	Avg.
ERM [6]	×	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
CDAN [11]	×	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
MCC [4]	×	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
SHOT [8]	✓	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
A^2Net [13]	✓	94.0	87.8	85.6	66.8	93.7	95.1	85.8	81.2	91.6	88.2	86.5	56.0	84.3
AaD [15]	✓	97.4	90.5	80.8	76.2	**97.3**	96.1	89.8	82.9	95.5	93.0	92.0	64.7	88.0
CoWA [7]	✓	96.2	89.7	83.9	73.8	96.4	97.4	89.3	**86.8**	94.6	92.1	88.7	53.8	86.9
SDE [3]	✓	95.3	91.2	77.5	72.1	95.7	**97.8**	85.5	86.1	95.5	93.0	86.3	61.6	86.5
AdaCon [2]	✓	97.0	84.7	84.0	77.3	96.7	93.8	91.9	84.8	94.3	93.1	**94.1**	49.7	86.8
DaC [16]	✓	96.6	86.8	86.4	78.4	96.4	96.2	93.6	83.8	**96.8**	**95.1**	89.6	50.0	87.3
C-SFDA [5]	✓	**97.6**	88.8	86.1	72.2	97.2	94.4	92.1	84.7	93.0	90.7	93.1	63.5	87.8
I-SFDA [12]	✓	97.5	**91.4**	**87.9**	79.4	97.2	97.2	92.2	83.0	96.4	94.2	91.1	53.0	88.4
SHOT+DPC [14]	✓	95.6	88.2	82.8	59.4	92.5	95.7	85.6	81.7	91.6	90.9	87.6	60.1	84.3
Ours	✓	94.6	86.4	85.4	**96.8**	96.7	92.2	**96.1**	82.6	88.2	88.4	89.8	**72.4**	**89.1**

7. Results on VisDA-C

Table 2 compares our method against state-of-the-art UDA and SFDA approaches on the VisDA-C dataset, addressing the synthetic-to-real domain shift. The table is organized into three sections: ERM, UDA, and SFDA methods, with ERM serving as the baseline lower bound. Our approach consistently demonstrates significant improvements over the majority of the compared methods. Notably, our method outperforms the best SFDA baseline, I-SFDA, while achieving substantial gains over other state-of-the-art SFDA methods. Specifically, it delivers a 1.1% improvement over AaD, a 2.3% advantage over AC, and a 1.3% enhancement over C-SFDA in terms of average accuracy (Avg.).

References

[1] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79:151–175, 2010. 1

[2] Dian Chen, Dequan Wang, Trevor Darrell, and Sayna Ebrahimi. Contrastive test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 295–305, 2022. 5

[3] Ning Ding, Yixing Xu, Yehui Tang, Chao Xu, Yunhe Wang, and Dacheng Tao. Source-free domain adaptation via distribution estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7212–7222, 2022. 5

[4] Ying Jin, Ximei Wang, Mingsheng Long, and Jianmin Wang. Minimum class confusion for versatile domain adaptation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16*, pages 464–480. Springer, 2020. 5

[5] Nazmul Karim, Niluthpol Chowdhury Mithun, Abhinav Rajvanshi, Han-pang Chiu, Supun Samarasekera, and Nazanin Rahnavard. C-sfda: A curriculum learning aided self-training framework for efficient source free domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24120–24131, 2023. 5

[6] Vladimir Koltchinskii. *Oracle Inequalities in Empirical Risk Minimization and Sparse Recovery Problems: Ecole d'Eté de Probabilités de Saint-Flour XXXVIII-2008*. Springer Science & Business Media, 2011. 5

[7] Jonghyun Lee, Dahuin Jung, Junho Yim, and Sungroh Yoon. Confidence score for source-free unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 12365–12377. PMLR, 2022. 5

[8] Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 6028–6039. PMLR, 2020. 5

[9] Jian Liang, Dapeng Hu, Yunbo Wang, Ran He, and Jiashi Feng. Source data-absent unsupervised domain adaptation through hypothesis transfer and labeling transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):8602–8617, 2021. 4

[10] Mattia Litrico, Alessio Del Bue, and Pietro Morerio. Guiding pseudo-labels with uncertainty estimation for test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 2, 3

[11] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. *Advances in neural information processing systems*, 31, 2018. 5

[12] Yu Mitsuzumi, Akisato Kimura, and Hisashi Kashima. Understanding and improving source-free domain adaptation from a theoretical perspective. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28515–28524, 2024. 5

[13] Haifeng Xia, Handong Zhao, and Zhengming Ding. Adaptive adversarial network for source-free domain adaptation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9010–9019, 2021. 5

[14] Haifeng Xia, Siyu Xia, and Zhengming Ding. Discriminative pattern calibration mechanism for source-free domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23648–23658, 2024. 5

[15] Shiqi Yang, Yaxing Wang, Kai Wang, Shangling Jui, et al. Attracting and dispersing: A simple approach for source-free domain adaptation. In *Advances in Neural Information Processing Systems*, 2022. 4, 5

[16] Ziyi Zhang, Weikai Chen, Hui Cheng, Zhen Li, Siyuan Li, Liang Lin, and Guanbin Li. Divide and contrast: Source-free domain adaptation via adaptive contrastive learning. *Advances in Neural Information Processing Systems*, 35:5137–5149, 2022. 5