PROXY DENOISING FOR SOURCE-FREE DOMAIN ADAPTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Source-Free Domain Adaptation (SFDA) aims to adapt a pre-trained source model to an unlabeled target domain with no access to the source data. Inspired by the success of large Vision-Language (ViL) models in many applications, the latest research has validated ViL's benefit for SFDA by using their predictions as pseudo supervision. However, we observe that ViL's supervision could be noisy and inaccurate at an unknown rate, introducing additional negative effects during adaption. To address this thus-far ignored challenge, we introduce a novel ***Pro**xy **De**noising* (**ProDe**) approach. The key idea is to leverage the ViL model as a proxy to facilitate the adaptation process towards the latent domain-invariant space. Concretely, we design a proxy denoising mechanism to correct ViL's predictions. This is grounded on a proxy confidence theory that models the dynamic effect of proxy's divergence against the domain-invariant space during adaptation. To capitalize the corrected proxy, we further derive a mutual knowledge distilling regularization. Extensive experiments show that ProDe significantly outperforms the current state-of-the-art alternatives under both conventional closed-set setting and the more challenging open-set, partial-set, generalized SFDA, multi-target, multi-source, and test-time settings. Our code will be released.

1 INTRODUCTION

Conventional Unsupervised Domain Adaptation (UDA) uses well-annotated source data and unannotated target data to achieve cross-domain transfer. Its data access requirement however raises the increasing concerns around safety and privacy. There is thus a call for restricted access to source domain training data, leading to a more practical but challenging transfer learning setting – Source-Free Domain Adaptation (SFDA) (Li et al., 2020a; Xia et al., 2021; Roy et al., 2022).

At the absence of source samples, applying traditional cross-domain distribution matching approaches is no longer feasible (Ganin & Lempitsky, 2015; Kang et al., 2019). Instead, self-supervised learning comes into play by aiming to generate/mine auxiliary information for unsupervised adaptation. There are two main routes. *The first* makes SFDA as a special case of UDA by explicitly creating a pseudo-source domain, making previous UDA methods such as adversarial learning (Xia et al., 2021; Kurmi et al., 2021) or minimizing domain shift (Ding et al., 2022; Tian et al., 2021; Kundu et al., 2022) applicable. *The second* further refines generated supervision from the source model (Lao et al., 2021; Wang et al., 2022a; Huang et al., 2021) or target data (Yang et al., 2022; Tang et al., 2022; Yang et al., 2021a), as the constructed pseudo source domain may be noisy. These existing methods all perform a free alignment without external guidance from the target feature space to the unknown domain-invariant feature space.

There has been growing interest in leveraging pre-trained large Vision-Language (ViL) models, e.g., CLIP (Radford et al., 2021), for transfer learning challenges. This is because ViL models have been trained with a massive amount of diverse vision-language data, encompassing rich knowledge potentially useful for many downstream tasks. For instance, Ge et al. (2022); Lai et al. (2023); Singha et al. (2023) disentangle domain and category information within ViL model's visual features by learning domain-specific textual or visual prompts. Recently, ViL models have also been used to tackle the SFDA problem (Tang et al., 2024c; Xiao et al., 2024). However, they simply treat the ViL model's predictions as ground truth, which would be not true in many unknown cases and finally harming their performance.

To address the limitation mentioned above, in this paper, we propose a new **Pro**xy **De**noising (**ProDe**) approach for SFDA. In contrast to (Tang et al., 2024c; Xiao et al., 2024), we consider the ViL model/space as a *noisy* proxy of the latent domain-invariant space[1], with a need to be denoised. At the absence of any good reference models for measuring the noisy degree with the already strong ViL model's predictions, we exploit the dynamics of domain adaptation process, starting at the source model space and terminating presumably in the latent domain-invariant space. In particular, this takes into account the proxy's divergence against the domain-invariant space (Fig. 1). Specifically, we model approximately the effect of ViL model's prediction error on domain adaption by formulating a proxy confidence theory, in relation to the discrepancy between the source domain and the current under-adaptation model. This leads to a novel proxy denoising mechanism for ViL prediction correction. To capitalize the corrected ViL predictions more effectively, a mutual knowledge distilling regularization is further designed.

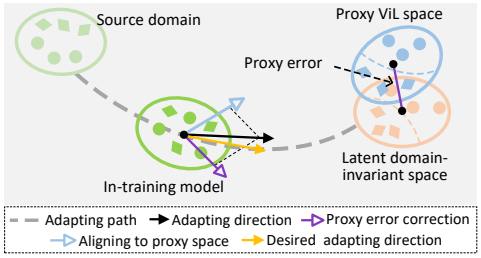


Figure 1: Conceptual illustration of ProDe. We align the adapting direction with the desired trajectory by leveraging a proxy space that approximates the latent domain-invariant space. This process incorporates direction adjustments based on proxy error correction, implementing proxy denoising, and finally achieves enhanced model adaptation.

Our **contributions** are summarized as follows: **(1)** We for the first time investigate the inaccurate predictions of ViL models in the context of SFDA. **(2)** We formulate a novel ProDe method that reliably corrects the ViL model's predictions under the guidance of a proxy confidence theory. A mutual knowledge distilling regularization is also introduced for capitalizing the refined proxy predictions more effectively. **(3)** Extensive experiments on four benchmarks show that our ProDe significantly outperforms previous art alternatives in closed-set settings, as well as the more challenging partial-set, open-set, and generalized SFDA, multi-target, multi-source and test-time settings .

2 Related Work

Source-Free Domain Adaptation The main issue with SFDA is the lack of supervision during model adaptation. To overcome this challenge, current methods are broadly divided into three categories. The first category involves converting SFDA to conventional UDA by introducing a pseudo-source domain. This can be achieved by building the pseudo-source domain through generative models (Tian et al., 2022; Li et al., 2020b) or by splitting a source-distribution-like subset from the target domain (Du et al., 2023). The second category involves mining auxiliary information from the pre-trained source model to assist in aligning the feature distribution from the target domain to the source domain. Commonly used auxiliary factors include multi-hypothesis (Lao et al., 2021), prototypes (Zhou et al., 2024), source distribution estimation (Ding et al., 2022), or hard samples (Li et al., 2021). The third category focuses on the target domain and creates additional constraints to correct the semantic noise in model transferring. In practice, domain-aware gradient control (Yang et al., 2021b), data geometry such as the intrinsic neighborhood structure (Tang et al., 2021) and target data manifold (Tang et al., 2022; Tang et al., 2024a), are exploited to generate high-quality pseudo-labels (Liang et al., 2020; Chen et al., 2022b) or inject assistance in an unsupervised fashion (Yang et al., 2021a). The existing solutions refine auxiliary information from domain-specific knowledge, such as the source model and unlabeled target data, while neglecting the extensive general knowledge encoded in off-the-shelf pre-trained multimodal models.

Vision-Language Models ViL models, such as CLIP (Radford et al., 2021) and GLIP (Li et al., 2022), have shown promise in various tasks (Liang et al., 2023; Wang et al., 2022c) due to their ability to capture modality invariant features. There are two main lines of research related to these models. The first line aims to improve their performance. For instance, text-prompt learning (Zhou et al., 2022; Ge et al., 2022) and visual-prompt learning (Wang et al., 2023; Jia et al., 2022) were

[1]The issue of noisy predictions is evidenced by the inferior zero-shot performance of the ViL model, e.g., CLIP, on the target domains (see Tab. 4). Also, "domain invariant space" refers to an ideal latent embedding space where the mapped features from different domains align with the same probability distribution.

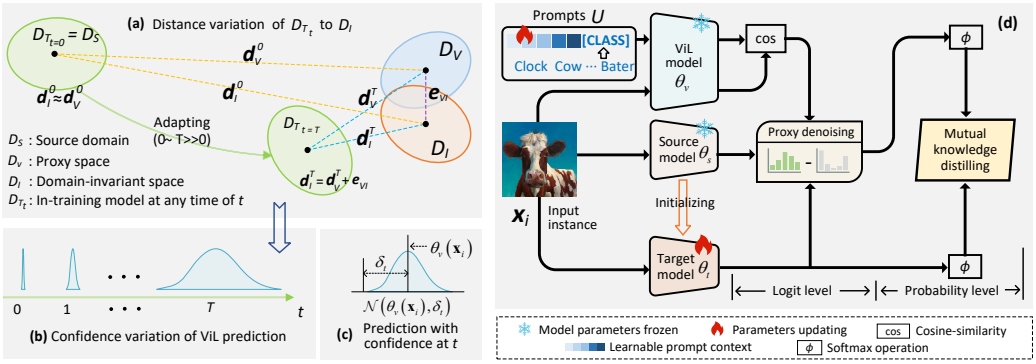


Figure 2: **Left:** Dynamics of effect of ViL model's prediction error (proxy error) during proxy alignment. (a) In the initial adaptation phase, it is possible to overlook the proxy errors. However, as the in-training model approaches the proxy space, these errors become more noticeable, leading to a continuous decline in the reliability of ViL predictions as shown in (b) and (c). **Right:** Our ProDe capitalizes the corrected proxy, involving a mutual knowledge distilling regularization and a proxy denoising mechanism imposing adjustment on the ViL logits for more reliable ViL prediction.

adopted to optimize the text encoder and image encoder, respectively, using learnable prompts related to application scenarios. Some researchers have also improved the data efficiency of these models by re-purposing (Andonian et al., 2022) or removing noisy data (Wang et al., 2021). The second line of research focuses on using ViL models as external knowledge to boost downstream tasks. Related work in this area mainly follows three frameworks: Plain fusion (Liu et al., 2024), knowledge distillation (Pei et al., 2023) and information entropy regulating (Cha et al., 2022). Moving further from recent ViL based SFDA models (Tang et al., 2024c; Xiao et al., 2024), we tackle the challenge of mitigating the noise of ViL's supervision.

3 METHODOLOGY

3.1 PROBLEM FORMULATION

We start with a labeled source domain and an unlabeled target domain, sharing the same C categories. Let $\mathcal{X}_s$ and $\mathcal{Y}_s$ be the source data samples and labels. Similarly, the target samples and the truth target labels are denoted as $\mathcal{X}_t = \{\boldsymbol{x}_i\}_{i=1}^n$ and $\mathcal{Y}_t = \{y_i\}_{i=1}^n$, respectively, where n is the sample number. SFDA aims to learn a target model $\theta_t : \mathcal{X}_t \to \mathcal{Y}_t$ given (1) the pre-trained source model $\theta_s : \mathcal{X}_s \to \mathcal{Y}_s$, (2) the unlabeled target data $\mathcal{X}_t$. In this context, we further leverage a ViL model θ_v that produces noise supervision.

To overcome this issue, we exploit the dynamics of domain adaptation process. As shown in Fig. 2 (a), we consider three spaces: source domain D_S (i.e., source image embedding space), domain-invariant space D_I, and ViL space D_V (our best possible proxy approximating D_I). In this context, D_I typically refers to an ideal, unknown latent space that is domain generalized in image embedding format. We want to align the in-training model D_{T_t} from D_S to D_I as $t \in [0 \sim T] \gg 0$. Without access to D_I, we instead perform ***proxy alignment*** (aligning D_{T_t} to proxy D_V) with adjustment. The discrepancy between D_I and D_V is referred to as ***proxy error*** $\boldsymbol{e}_{VI}$, underpinning intuitively ViL's prediction errors. We further transform the task of minimizing the errors of ViL predictions to controlling the proxy error by establishing a proxy confidence theory.

3.2 PROXY CONFIDENCE THEORY

This theory is grounded on understanding the impact of the proxy error on the domain adaptation process. This can be achieved by examining the dynamics of the proxy alignment, which is outlined in `Section 3.1`.

To account for the continuity of movement, as demonstrated in Fig. 2 (a), we consider two typical situations in the proxy alignment process, in which the distance of D_{T_t} to D_V and D_I are denoted as

$\boldsymbol{d}_V^t$ and $\boldsymbol{d}_I^t$, respectively; the distinction between D_V and D_I, i.e., proxy error $\boldsymbol{e}_{VI}$, is a space-to-space distance in the vector form.

- **Case1:** When D_{T_t} is significantly far from D_V, e.g., the beginning of adaptation ($t = 0$), it is hold that $\boldsymbol{d}_I^0 \approx \boldsymbol{d}_V^0 \gg \boldsymbol{e}_{VI}$. This implies that aligning to D_I or D_V is equivalent. Consequently, the proxy errors $\boldsymbol{e}_{VI}$ can be ignored, thereby the ViL prediction can be deemed trustworthy.
- **Case2:** When D_{T_t} approaches D_V, e.g., the later phase in the adaptation ($t = T \gg 0$), it becomes crucial to consider the proxy error and the distance relationship changes to $\boldsymbol{d}_I^T = \boldsymbol{d}_V^T + \boldsymbol{e}_{VI}$ (this equation is established based on the vector geometric property that $\boldsymbol{u}$, $\boldsymbol{v}$, and $\boldsymbol{u} + \boldsymbol{v}$ form a triangle, where $\boldsymbol{u}$ and $\boldsymbol{v}$ are two sides, the $\boldsymbol{u} + \boldsymbol{v}$ is the left one). This is when the ViL predictions become less reliable.

It is seen that the proxy errors dynamically impact on this proxy alignment process, reflected in the relative relationship between $\boldsymbol{d}_V^t$ and $\boldsymbol{d}_I^t$ as:

$$\eta_t = \frac{|\boldsymbol{d}_I^t|}{|\boldsymbol{d}_V^t|} = \frac{|\boldsymbol{d}_V^t + \boldsymbol{e}_{VI}|}{|\boldsymbol{d}_V^t|} \leq \frac{|\boldsymbol{d}_V^t| + |\boldsymbol{e}_{VI}|}{|\boldsymbol{d}_V^t|} = 1 + \frac{|\boldsymbol{e}_{VI}|}{|\boldsymbol{d}_V^t|}, \tag{1}$$

where η_t means the impact degree, $|\boldsymbol{a}|$ means the absolute value (length) of distance vector $\boldsymbol{a}$. During the proxy alignment, the ratio of $|\boldsymbol{e}_{VI}|/|\boldsymbol{d}_V^t|$ in Eq. (1) gradually increases from 0 (e.g., Case 1) to a non-zero value (e.g., Case 2), leading to a gradual increase in η_t from 1. In other words, the impact of errors gradually increases.

Corresponding to this dynamics mentioned above, as shown in Fig. 2 (b), the ViL prediction variance gradually increases, which implies a progressive decrease in the reliability of the ViL prediction. At any time t, we treat the ViL prediction as a Gaussian distribution $\mathcal{N}\left(\theta_v\left(x_i\right), \delta_t\right)$ with the mean of ViL model's prediction $\theta_v\left(x_i\right)$ and prediction variance $\delta_t \propto \eta_t$ (Fig. 2 (c)). Here, we consider the VLM's predictions to be influenced by various sources of noise and uncertainty, which justifies the Gaussian approximation according to Central Limit Theorem.

Since the $\boldsymbol{e}_{VI}$ is unknown, we cannot formulate these dynamics explicitly. We consider this problem approximately: Quantifying the prediction variance with the varying confidence of the ViL model predictions. This conversion can be expressed in the form of a probability distribution with proxy confidence as:

$$\mathcal{N}\left(\theta_v\left(x_i\right), \delta_t\right) \Longrightarrow P\left(G_{P(V)} = True, t\right) P\left(V\right), \tag{2}$$

where $P(V)$ is the probability distribution of the proxy space D_V; $G_{P(V)}$ stands for a random event that the sampling results (i.e., ViL model's prediction) from $P(V)$ is confident; $P\left(G_{P(V)} = True, t\right)$ is proxy confidence, indicating the probability of the event $G_{P(V)}$ being true at time t, and it decreases progressively, matching the reduction of the ViL prediction reliability. By framing the prediction as a probabilistic event, we can leverage the concept of proxy confidence, $P\left(G_{P(V)} = True, t\right)$, to quantify how reliable we consider the VLM's predictions to be at any point in the adaptation process. This conversion allows us to connect the notion of prediction reliability with the underlying distributions, making it easier to reason about the impact of proxy errors. Within this probability context, we can formulate the *proxy confidence theory* for $P\left(G_{P(V)} = True, t\right)$ as detailed in **Theorem 1** with proof in `Appendix-A`.

Theorem 1 *Given a proxy alignment formulated in* `Section 3.1`*. The source domain (D_S), the domain-invariant space (D_I), the proxy space (D_V) and the in-training model (D_{T_t}) satisfy the probability distributions $P(S)$, $P(I)$, $P(V)$ and $P(T_t)$, respectively, where S, I, V and T_t are corresponding random variables. The factor describing the credibility of $P(V)$ has a p below.*

$$P\left(G_{P(V)} = True, t\right) \propto \frac{P(T_t)}{P(S)}. \tag{3}$$

Given that the effect of proxy error causes the varying of the confidence factor $P\left(G_{P(V)} = True, t\right)$, as mentioned earlier, **Theorem** 1 provides us an insight: *The effect of ViL model's prediction error on domain adaption is approximately reflected by the discrepancy between the source domain and the current in-training model.*

3.3 CAPITALIZING THE CORRECTED PROXY

Overview To capitalize the corrected proxy, we propose a novel ProDe method involving two designs: A proxy denoising mechanism and a mutual knowledge distilling regularization, as shown in Fig. 2 (d). In this method, the proxy denoising converts the original ViL predictions to reliable ones by imposing correction on the logit level. The mutual distilling regularization encourages knowledge synchronization between the ViL model θ_v (teacher) and the in-training target model θ_t (student), coupled with a refinement for useful ingredients. In practice, this knowledge synchronization is jointly encouraged by learning target-specific prompt context (for the teacher model) and encoding the reliable proxy knowledge (for the student model). Additionally, unlike all previous SFDA approaches, the source model θ_s in ProDe not only initiates the target model at the beginning of adaptation, but also continues to serve the proxy denoising operation. ProDe's details are presented below.

Proxy denoising This module aims to filter out the noisy ViL prediction in an individual correction fashion, serving the target domain-specific task. Based on the results from **Theorem** 1 (Eq. (3)), we further convert the ViL space's probability distribution with proxy confidence, i.e., Eq. (2), into

$$\log\left(\frac{P(T_t)}{P(S)}P(V)\right) = \log P(V) - \left[\log P(S) - \log P(T_t)\right]. \tag{4}$$

In Eq. (4), the latter two items form an adjustment to correct for the first item, essentially providing a strategy to obtain reliable ViL prediction. Inspired by Eq. (4), we design denoising mechanism as:

$$\boldsymbol{p}_i' = \phi\left(\boldsymbol{l}_i'\right),\ \boldsymbol{l}_i' = \theta_v\left(\boldsymbol{x}_i, \boldsymbol{v}\right) - \omega\Delta_t,\ \Delta_t = \theta_s\left(\boldsymbol{x}_i\right) - \theta_t\left(\boldsymbol{x}_i\right), \tag{5}$$

where $\boldsymbol{l}_i'$ and $\boldsymbol{p}_i'$ are the denoised ViL logit and prediction of input instance $\boldsymbol{x}_i$, respectively, $\boldsymbol{v}$ is the learnable prompt context and ϕ means softmax operation; Δ_t refers to the adaptive adjustment for correction, and the hyper-parameter ω specifies the correction strength.

Mutual knowledge distilling The regularization consists of two components L_{Syn} and L_{Ref}. First of all, L_{Syn} synchronizes knowledge from both sides by maximizing the unbiased mutual information between the denoised ViL prediction $\boldsymbol{p}_i'$ and the target prediction $\boldsymbol{p}_i = \phi(\theta_t(\boldsymbol{x}_i))$. This design is motivated by that despite massive (often noisy) data, ViL models (e.g., CLIP) don't always outperform source domain supervised models focused on the target task. There are three reasons: (1) ViL models are generalists, while source domain models are specialized. (2) ViL models may include irrelevant data, whereas source domain models use curated, relevant data. (3) ViL models might overlook task-specific features that are captured by source domain models. Meanwhile, to avoid the solution collapse (Ghasedi Dizaji et al., 2017), we introduce a widely used category balance constraint (Yang et al., 2021a). Importantly, L_{Ref} distills a useful fraction of knowledge obtained by the interaction learning as it is still likely noisy due to the lack of ground-truth labels in SFDA setting. We use classification with the denoised ViL predictions as the labels.

Formally, we can summarize the designs mentioned above with the following objective.

$$L_{ProDe} = \min_{\theta_t, \boldsymbol{v}} \alpha \overbrace{\left(-\mathbb{E}_{\boldsymbol{x}_i \in \mathcal{X}_t} \mathbf{MI}\left(\boldsymbol{p}_i', \boldsymbol{p}_i\right) + \gamma \sum_{c=1}^{C} \bar{q}_c \log \bar{q}_c\right)}^{L_{Syn}} - \beta \overbrace{\mathbb{E}_{\boldsymbol{x}_i \in \mathcal{X}_t} \sum_{c=1}^{C} \mathbb{1}\left[c = y_i'\right] \log p_{i,c}}^{L_{Ref}}, \tag{6}$$

where $\mathbf{MI}(\cdot,\cdot)$ computes the mutual information (Ji et al., 2019); the second item in L_{Syn} is the balance loss, C is the category number, $\bar{q}_c = \frac{1}{n}\sum_{i=1}^{n} q_{i,c}$ is c-th element of $\bar{\boldsymbol{q}}$, in which $\bar{\boldsymbol{q}}$ is are empirical label distribution over the C categories, $q_{i,c}$ is the probability value of target prediction $\theta_t\left(\boldsymbol{x}_i\right)$ in the c-th category; in L_{Ref}, $p_{i,c}$ is the c-th element of $\boldsymbol{p}_i$, $\mathbb{1}[c = y_i']$ is a one-hot encoding of hard category label y_i' predicted by the denoised ViL prediction $\boldsymbol{p}_i'$. With regulating of Eq. (6), we accomplish the model training whose concrete procedure is summarized to the algorithm provided in `Appendix B`.

4 EXPERIMENTS

Datasets We evaluate four widely used domain adaptation benchmarks. Among them, **Office-31** (Saenko et al., 2010) and **Office-Home** (Venkateswara et al., 2017) are small-scaled and medium-scale datasets, respectively, whilst **VisDA** (Peng et al., 2017) and **DomainNet-126** (Saito et al., 2019) are both challenging large-scale datasets. Their details are provided in `Appendix-C`.

SFDA settings We consider five distinct SFDA settings: (1) closed-set, (2) partial-set, open-set (initialized in SHOT (Liang et al., 2020)), (3) generalized SFDA, which is detailed in GDA (Yang et al., 2021b) initially, (4) multi-target (SF-MTDA), multi-source (SF-MSDA) that are detailed in (Kumar et al., 2023) and (Ahmed et al., 2021), respectively, and (5) test-time adaptation (TTA) detailed in (Wang et al., 2020). `Appendix-D` elaborates other experiment implementations.

4.1 COMPETITORS

To evaluate ProDe, we select 30 related comparisons divided into four groups. *(1) The first* includes 2 base models involved in the SFDA problem: The source model (termed Source) and CLIP zero-shot (termed CLIP) (Radford et al., 2021). *(2) The second* includes 7 current state-of-the-art domain adaptation methods with ViL model (adopting CLIP in practice), covering UDA and SFDA settings: DAPL-R (Ge et al., 2022), PADCLIP-R (Lai et al., 2023), ADCLIP-R (Singha et al., 2023), PDA-R (Bai et al., 2024), DAMP-R (Du et al., 2024), DIFO-R (Tang et al., 2024c) and DIFO-V (Tang et al., 2024c). Among them, DIFO-R and DIFO-V are the SFDA methods, while others are UDA methods. The suffix of "-R" and "-V" means that the image-encoder in CLIP uses the backbone of ResNet and ViT, respectively. Specifically, DIFO-V employs the backbone of ViT-B/32 across all datasets, whilst the rest methods with "-R" use ResNet101 on VisDA and ResNet50 on the other three datasets. *(3) The third* comprises 16 state-of-the-art SFDA models without using ViL model: SHOT (Liang et al., 2020), NRC (Yang et al., 2021a), GKD (Tang et al., 2021), HCL (Huang et al., 2021), AaD (Yang et al., 2022), AdaCon (Chen et al., 2022a), CoWA (Lee et al., 2022), ELR (Yi et al., 2023), PLUE (Litrico et al., 2023), CRS (Zhang et al., 2023), CPD (Zhou et al., 2024), TPDS (Tang et al., 2024a), GDA (Yang et al., 2021b), PSAT-ViT (Tang et al., 2024b) CoNMix (Kumar et al., 2023) and DECISION (Ahmed et al., 2021). Among them, GDA and PSAT-ViT are specific for the generalized SFDA setting, while CoNMix and DECISION are SF-MTDA and SF-MSDA methods, respectively. *(4) The fourth* comprises 5 state-of-the-art TTA models: Tent (Wang et al., 2020), T3A (Iwasawa & Matsuo, 2021), CoTTA (Wang et al., 2022b), EATA (Niu et al., 2022) and SAR (Niu et al., 2023). Additionally, for a fair comparison with DIFO, the previous best SFDA method with ViL model, we have initiated ProDe into the same versions mentioned above: A strong version ProDe-V and a weak version ProDe-R.

Table 1: Closed-set SFDA results (%) on **Office-31**. **SF** means source-free.

Method	Venue	SF	A→D	A→W	D→A	D→W	W→A	W→D	Avg.
Source	–	–	79.1	76.6	59.9	95.5	61.4	98.8	78.6
SHOT	ICML20	✓	93.7	91.1	74.2	98.2	74.6	**100.**	88.6
NRC	NIPS21	✓	96.0	90.8	75.3	99.0	75.0	**100.**	89.4
GKD	IROS21	✓	94.6	91.6	75.1	98.7	75.1	**100.**	89.2
HCL	NIPS21	✓	94.7	92.5	75.9	98.2	77.7	**100.**	89.8
AaD	NIPS22	✓	96.4	92.1	75.0	**99.1**	76.5	**100.**	89.9
AdaCon	CVPR22	✓	87.7	83.1	73.7	91.3	77.6	72.8	81.0
CoWA	ICML22	✓	94.4	95.2	76.2	98.5	77.6	99.8	90.3
ELR	ICLR23	✓	93.8	93.3	76.2	98.0	76.9	**100.**	89.6
PLUE	CVPR23	✓	89.2	88.4	72.8	97.1	69.6	97.9	85.8
CPD	PR24	✓	96.6	94.2	77.3	98.2	78.3	**100.**	90.8
TPDS	IJCV24	✓	97.1	94.5	75.7	98.7	75.5	99.8	90.2
DIFO-R	CVPR24	✓	93.6	92.1	78.5	95.7	78.8	97.0	89.3
DIFO-V	CVPR24	✓	**97.2**	95.5	83.0	**97.2**	**83.2**	98.8	92.5
ProDe-R	–	✓	92.6	93.2	80.9	94.6	81.0	98.0	90.0
ProDe-V	–	✓	96.6	**96.4**	**83.1**	96.9	82.9	99.8	**92.6**

4.2 COMPARISON RESULTS ON MULTI-SFDA SETTINGS

Comparisons on closed-set SFDA. Tab. 1∼3 lists the quantitative comparisons on the four evaluation datasets. Both ProDe-R and ProDe-V beat all non-multimodal SFDA methods by a large margin. Compared with the second-best method CPD (Office-31), TPDS (Office-Home), PLUE (VisDA) and GKD (DomainNet-126), ProDe-V improves by **1.8%**, **12.7%** **3.3%** and **16.3%** in average accuracy, respectively. As for those methods with CLIP, ProDe also beat them in the same backbone setting. In particular, compared with the multimodal SFDA method DIFO, ProDe improves by **4.8%** and **5.0%** (DomainNet-126) at most using ResNet and ViT-B/32, respectively. Actually, the weak version of our method, ProDe-R, is competitive with the strong version of DIFO, DIFO-V. All of these results indicate that ProDe can significantly boost the cross-domain adaptation under the SFDA setting.

Comparison to CLIP prediction results. It only makes sense for ProDe to outperform CLIP. To assess this, we conducted a quantitative comparison between our model's adaptation performance and CLIP's zero-shot performance. The results of our model are reported with average accuracy. As reported in Tab. 4, ProDe-R and ProDe-V improve at least by **6.2** % (on VisDA) and **8.7**% (on VisDA and DomainNet-126), respectively, compared with CLIP's results on the four datasets. This result

Table 2: Closed-set SFDA results (%) on **Office-Home** and **VisDA**. **SF** means source-free. The full results on **VisDA** are provided in `Appendix E.1`.

Method	Venue	SF	Office-Home													VisDA
			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.	Sy→Re
Source	–	–	43.7	67.0	73.9	49.9	60.1	62.5	51.7	40.9	72.6	64.2	46.3	78.1	59.2	49.2
SHOT	ICML20	✓	56.7	77.9	80.6	68.0	78.0	79.4	67.9	54.5	82.3	74.2	58.6	84.5	71.9	82.7
NRC	NIPS21	✓	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	85.9
GKD	IROS21	✓	56.5	78.2	81.8	68.7	78.9	79.1	67.6	54.8	82.6	74.4	58.5	84.8	72.2	83.0
AaD	NIPS22	✓	59.3	79.3	82.1	68.9	79.8	79.5	67.2	57.4	83.1	72.1	58.5	85.4	72.7	88.0
AdaCon	CVPR22	✓	47.2	75.1	75.5	60.7	73.3	73.2	60.2	45.2	76.6	65.6	48.3	79.1	65.0	86.8
CoWA	ICML22	✓	56.9	78.4	81.0	69.1	80.0	79.9	67.7	57.2	82.4	72.8	60.5	84.5	72.5	86.9
ELR	ICLR23	✓	58.4	78.7	81.5	69.2	79.5	79.3	66.3	58.0	82.6	73.4	59.8	85.1	72.6	85.8
PLUE	CVPR23	✓	49.1	73.5	78.2	62.9	73.5	74.5	62.2	48.3	78.6	68.6	51.8	81.5	66.9	88.3
CPD	PR24	✓	59.1	79.0	82.4	68.5	79.7	79.5	67.9	57.9	82.8	73.8	61.2	84.6	73.0	85.8
TPDS	IJCV24	✓	59.3	80.3	82.1	70.6	79.4	80.9	69.8	56.8	82.1	74.5	61.2	85.3	73.5	87.6
DAPL-R	TNNLS23	✗	54.1	84.3	84.8	74.4	83.7	85.0	74.5	54.6	84.8	75.2	54.7	83.8	74.5	86.9
PADCLIP-R	ICCV23	✗	57.5	84.0	83.8	77.8	85.5	84.7	76.3	59.2	85.4	78.1	60.2	86.7	76.6	88.5
ADCLIP-R	ICCVW23	✗	55.4	85.2	85.6	76.1	85.8	86.2	76.7	56.1	85.4	76.8	56.1	85.5	75.9	87.7
PDA-R	AAAI24	✗	55.4	85.1	85.8	75.2	85.2	85.2	74.2	55.2	85.8	74.7	55.8	86.3	75.3	86.4
DAMP-R	CVPR24	✗	59.7	88.5	86.8	76.6	88.9	87.0	76.3	59.6	87.1	77.0	61.0	89.9	78.2	88.4
DIFO-R	CVPR24	✓	62.6	87.5	87.1	79.5	87.9	87.4	78.3	63.4	88.1	80.0	63.3	87.7	79.4	88.8
DIFO-V	CVPR24	✓	70.6	90.6	88.8	82.5	90.6	88.8	80.9	70.1	88.9	83.4	70.5	91.2	83.1	90.3
ProDe-R	–	✓	66.0	91.2	90.8	81.4	91.4	90.5	82.2	67.3	90.8	83.6	67.7	91.6	82.9	89.9
ProDe-V	–	✓	**74.6**	**92.9**	**92.4**	**84.4**	**93.0**	**92.2**	**83.8**	**74.8**	**92.4**	**84.9**	**75.2**	**93.7**	**86.2**	**91.6**

Table 3: Closed-set SFDA results (%) on **DomainNet-126**. **SF** means source-free.

Method	Venue	SF	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	–	–	44.6	59.8	47.5	53.3	75.3	46.2	55.3	62.7	46.4	55.1	50.7	59.5	54.7
SHOT	ICML20	✓	63.5	78.2	59.5	67.9	81.3	61.7	67.7	67.6	57.8	70.2	64.0	78.0	68.1
GKD	IROS21	✓	61.4	77.4	60.3	69.6	81.4	63.2	68.3	68.4	59.5	71.5	65.2	77.6	68.7
NRC	NIPS21	✓	62.6	77.1	58.3	62.9	81.3	60.7	64.7	69.4	58.7	69.4	65.8	78.7	67.5
AdaCon	CVPR22	✓	60.8	74.8	55.9	62.2	78.3	58.2	63.1	68.1	55.6	67.1	66.0	75.4	65.4
CoWA	ICML22	✓	64.6	80.6	60.6	66.2	79.8	60.8	69.0	67.2	60.0	69.0	65.8	79.9	68.6
PLUE	CVPR23	✓	59.8	74.0	56.0	61.6	78.5	57.9	61.6	65.9	53.8	67.5	64.3	76.0	64.7
TPDS	IJCV24	✓	62.9	77.1	59.8	65.6	79.0	61.5	66.4	67.0	58.2	68.6	64.3	75.3	67.1
DAPL-R	TNNLS23	✗	72.4	87.6	65.9	72.7	87.6	65.6	73.2	72.4	66.2	73.8	72.9	87.8	74.8
ADCLIP-R	ICCVW23	✗	71.7	88.1	66.0	73.2	86.9	65.2	73.6	73.0	68.4	72.3	74.2	89.3	75.2
DAMP-R	CVPR24	✗	76.7	88.5	71.7	74.2	88.7	70.8	74.4	75.7	70.5	74.9	76.1	88.2	77.5
DIFO-R	CVPR24	✓	73.8	89.0	69.4	74.0	88.7	70.1	74.8	74.6	69.6	74.7	74.3	88.0	76.7
DIFO-V	CVPR24	✓	76.6	87.2	74.9	80.0	87.4	75.6	80.8	77.3	75.5	80.5	76.7	87.3	80.0
ProDe-R	–	✓	79.3	91.0	75.3	80.0	90.9	75.6	80.4	78.9	75.4	80.4	79.2	91.0	81.5
ProDe-V	–	✓	**83.2**	**92.4**	**79.0**	**85.0**	**92.3**	**79.3**	**85.5**	**83.1**	**79.1**	**85.5**	**83.4**	**92.4**	**85.0**

shows that *the multimodal CLIP space only approximates the domain-invariant space, suggesting the need for denoising that this paper focuses on*.

Comparison on partial-set and open-set settings. For a complete evaluation, we also evaluate ProDe on two variation scenarios: Partial-set and open-set settings. As reported in Tab. 5, ProDe-V achieves a gain of **0.6**% (partial-set) and **6.7**% (open-set) compared with the best competitor DIFO-V.

Comparison on generalized SFDA settings. The generalized SFDA is an extended problem of closed-set SFDA, highlighting the anti-forgetting ability on the seen source domain. The same as (Yang et al., 2021b), we adopt the harmonic mean accuracy as evaluation protocol, which is computed by $H = (2 * Acc_s * Acc_t)/(Acc_s + Acc_t)$ where Acc_s and Acc_t are the accuracies of the adapted target model on the source domain and the target domain, respectively. Note that the Acc_s is computed based on the source-testing set. The same to (Yang et al., 2021b; Tang et al., 2024b), on the source domain, the ratio of training and testing sets is 9:1. To evaluate effectiveness, two generalized SFDA methods, GDA and PSAT-ViT, are chosen as additional comparisons. Based on Tab. 6, it is seen that ProDe-V outperforms all comparisons in terms of H-accuracy, even those designed to imitate forgetting. Meanwhile, both ProDe-R and ProDe-V deliver balanced results across the source and target domains. This is due to the correction in the proxy denoising, which incorporates information from the source model, thereby mitigating forgetting of the source domain.

Comparison on SF-MTDA, SF-MSDA and TTA settings. This part evaluates ProDe in broader SF-MTDA, SF-MSDA and TTA settings. For SF-MTDA, we treat multiple target domains as a single integrated domain and adapt the source model accordingly. For SF-MSDA, we follow the ensembling

approach from (Ahmed et al., 2021), passing the target data through each adapted source model and averaging the soft predictions to derive the test labels. The results, as shown in the left side of Tab. 7, demonstrate that ProDe substantially outperforms state-of-the-art alternatives in both settings.

The right side of Fig. 7 reports the results on the online SFDA setting of TTA, where all comparison methods maintain a fixed batch size of 64, similar to ours. It is seen that ProDe demonstrates advantages over previous state-of-the-art methods.

Table 4: Comparison results with CLIP (%). `Appendix E.1` presents the full results.

Method	Office-31	Office-Home	VisDA	DomainNet-126
CLIP-R	71.4	72.1	83.7	72.7
ProDe-R	**90.0**	**82.9**	**89.9**	**81.5**
CLIP-V	79.8	76.1	82.9	76.3
ProDe-V	**92.6**	**86.2**	**91.6**	**85.0**

Table 5: Partial-set and open-set results (%) on **Office-Home**. `Appendix E.1` presents the full results.

Partial-set	Venue	Avg.	Open-set	Venue	Avg.
Source	–	62.8	Source	–	46.6
SHOT	ICML20	79.3	SHOT	ICML20	72.8
HCL	NIPS21	79.6	HCL	NIPS21	72.6
CoWA	ICML22	83.2	CoWA	ICML22	73.2
AaD	NIPS22	79.7	AaD	NIPS22	71.8
CRS	CVPR23	80.6	CRS	CVPR23	73.2
DIFO-V	CVPR24	85.6	DIFO-V	CVPR24	75.9
ProDe-V	–	**86.2**	**ProDe**-V	–	**82.6**

Table 6: Generalized SFDA results (%) on **Office-Home**. S, T are the results of the adapted target model on the source and target domains, i.e., Acc_s, Acc_t, respectively; **WAD** means With Anti-forgetting Design. `Appendix E.1` presents the full results.

Method	Venue	WAD	Avg. S (98.1-S)	T	H
Source	–	✗	98.1	59.2	73.1
SHOT	ICML20	✗	84.2 (13.9)	71.8	77.5
GKD	IROS21	✗	86.8 (11.3)	72.5	79.0
NRC	NIPS21	✗	91.3 (6.8)	72.3	80.7
AdaCon	CVPR22	✗	88.2 (9.9)	65.0	74.8
CoWA	ICML22	✗	91.8 (6.3)	72.4	81.0
PLUE	CVPR23	✗	**96.3** (1.8)	66.9	79.0
TPDS	IJCV24	✗	83.8 (14.3)	73.5	78.3
GDA	ICCV21	✓	80.0 (18.1)	70.2	74.4
PSAT-ViT	TMM24	✓	86.4 (11.7)	83.6	85.0
DIFO-R	CVPR24	✗	78.3 (19.8)	79.4	78.8
DIFO-V	CVPR24	✗	78.0 (20.1)	83.1	80.5
ProDe-R	–	✗	83.3 (14.8)	82.9	83.1
ProDe-V	–	✗	84.1 (14.0)	**86.2**	**85.1**

Table 7: SF-MTDA, SF-MSDA and TTA results (%) on **Office-Home**. The full results of TTA are provided in `Appendix E.1`.

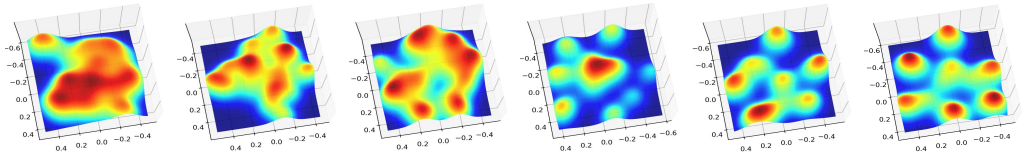
	Model	Venue	Ar→	Cl→	Pr→	Rw→	Avg.
SF-MTDA	CoNMix	WACV23	75.6	81.4	71.4	73.4	75.4
	ProDe-V	–	**84.4**	**89.4**	**80.9**	**81.9**	**84.2**
SF-MSDA	Method	Venue	→Rw	→Pr	→Cl	→Ar	Avg.
	SHOT-Ens	ICML20	82.9	82.8	59.3	72.2	74.3
	DECISION	CVPR21	83.6	84.4	59.4	74.5	75.5
	ProDe-V-Ens	–	**84.4**	**89.4**	**80.9**	**81.9**	**84.2**

	Method	Venue	Avg.
TTA	Tent	ICLR20	61.7
	T3A	NeurIPS21	63.8
	CoTTA	CVPR22	60.5
	EATA	ICML22	60.7
	SAR	ICLR23	60.3
	ProDe-V	–	**78.0**

4.3 Model Analysis

Feature distribution visualization. Based on the task Cl→Ar in Office-Home, we conducted a toy experiment to visualize the feature distribution of ProDe using the t-SNE tool. Meanwhile, five comparisons are considered, including CLIP-V, SHOT, TPDS, DIFO-V and Oracle. Among them, CLIP-V is the zero-shot result, and Oracle is trained on target domain Ar with the ground truth . For a clear view, all results are presented in 3D density charts. As shown in Fig. 3, from CLIP-V to Oracle, category clustering becomes increasingly apparent. The distribution shape of DIFO-V and ProDe-V is closer to the expert model than that of non-multimodal methods, SHOT and TPDS. Furthermore, although DIFO-V and ProDe-V have a similar pattern, ProDe-V's shape is more detailed with Oracle.

Ablation studies. This part isolates the effect of (1) the objective components in Eq. (6) and (2) proxy denoising (PD). Tab. 8 presents the ablation study results, with the baseline being the results of the source model (1 row). When $\mathcal{L}_{Syn}$ or $\mathcal{L}_{Ref}$ is used alone (2, 3 row), their performances show similar average accuracy. However, when they work together, the best results are achieved (4 row). This comparison indicates that the proposed two losses jointly contribute to the final performance. Additionally, we further evaluate the mutual information item $\mathrm{MI}(\cdot,\cdot)$ in L_{Syn} with a variant of ProDe, denoted ProDe w KL, where $\mathrm{MI}(\cdot,\cdot)$ is replaced by the KL divergence loss. A significant average gap of **4.8%** (compared with the results in 4 row) confirms the advantage of the mutual information optimization (5 row).



(a) CLIP-V (b) SHOT (c) TPDS (d) DIFO-V (e) ProDe-V (f) Oracle

Figure 3: Feature visualization comparison in 3D density charts.

Furthermore, removing proxy denoising from the model (ProDe-V w/o PD in 6 row) leads to a decrease in average accuracy by **2.0**%, which confirms its effectiveness. To evaluate the effect of components in the proxy denoising design, we respectively remove the source and target models' logits (see Eq. (5)) to obtain two ProDe variation methods, ProDe-V w/o PD-source and ProDe-V w/o PD-target. As listed in 7 and 8 rows, using either adjustment alone led to a significant decrease in performance. Also, we perform the correction at the probability level, instead of the logit level, in another comparison ProDe-V w/o PD-logits. The average **3.0%** decrease (compared with ProDe-V's results in 4 row) confirms the rationality of correction based on logits (9 row).

Table 8: Ablation study results (%) on **Office-31**, **Office-Home** and **VisDA**.

#	L_{Syn}	L_{Ref}	**Office-31**	**Office-Home**	**VisDA**	Avg.
1	✗	✗	78.6	59.2	49.2	62.3
2	✓	✗	91.8	78.8	90.2	86.9
3	✗	✓	86.5	83.2	90.7	86.8
4	✓	✓	**92.6**	**86.2**	**91.6**	**90.1**
5	**ProDe**-V w KL		82.6	83.7	89.6	85.3
6	**ProDe**-V w/o PD		90.5	83.9	89.9	88.1
7	**ProDe**-V w/o PD-source		91.2	84.6	90.2	88.7
8	**ProDe**-V w/o PD-target		80.1	83.5	90.8	84.8
9	**ProDe**-V w/o PD-logits		86.8	83.3	91.3	87.1

Table 9: Comparison results (%) on **Office-31**, **Office-Home** and **VisDA** as image encoder backbone in CLIP adopts architecture ViT-B/16. **SF** means source-free.

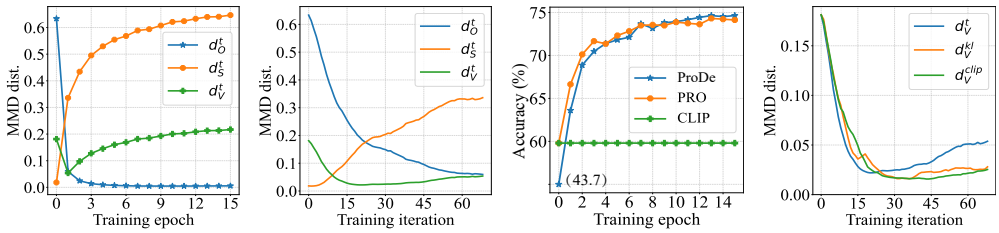
Method	Venue	**SF**	**Office-31**	**Office-Home**	**VisDA**
CLIP-V16	ICML21	✗	77.6	80.1	85.6
DAPL-V16	TNNLS23	✗	–	85.8	89.8
ADCLIP-V16	ICCVW23	✗	–	86.1	90.7
PAD-V16	AAAI24	✗	91.2	85.7	89.7
DAMP-V16	CVPR24	✗	–	87.1	90.9
DIFO-V16	CVPR24	✓	**92.5**	85.5	91.0
ProDe-V16	–	✓	**92.5**	**88.0**	**92.0**

Impact of image encoder backbone in CLIP. In addition to the ResNet and ViT-B/32 architectures aforementioned, we also implement ProDe using another well-known architecture, ViT-B/16, which we refer to as ProDe-V16. Furthermore, we compare the performance of CLIP-V16, DAPL-V16, ADCLIP-V16, PAD-V16, DAMP-V16 and DIFO-V16, which also use ViT-B/16 as their image encoder. As listed in Tab. 9, ProDe-V16 still surpasses all comparisons. Combining with the ResNet and ViT-B/32 results reported in Tab. 1∼Tab. 2, it is concluded that the advantage of ProDe is robust to the selection of the image-encoder backbone.

4.4 Quantitative Analysis of Proxy Denoising in Proxy Alignment View

In this part, we make a feature space shift analysis using the measure of MMD (Maximum Mean Discrepancy) distance to verify whether our ProDe method ensures the proxy alignment. In this experiment, we initially train a domain-invariant Oracle model over all Office-Home data with real labels, and use the logits to express the domain-invariant space O. Sequentially, we perform a transfer experiment of Ar→Cl. During this adaptation, there are K (epoch number) intermediate adapting target models. We feedforward the target data through each intermediate model and take the logits as a space. Thus, we obtain K intermediate target feature spaces $\{U_k\}_{k=1}^{K}$. These intermediate spaces can lead to three different kinds of distances corresponding to these frozen spaces, termed $\boldsymbol{d}_S^t$ (to the source domain), $\boldsymbol{d}_O^t$ (to the Oracle space) and $\boldsymbol{d}_V^t$ (to the proxy CLIP space). In practice, the CLIP image encoder's backbone is set to ViT-B/32.

Fig. 4 (a) displays the varying curves (epoch view) of $\boldsymbol{d}_S^t$, $\boldsymbol{d}_O^t$ and $\boldsymbol{d}_V^t$. As expected, $\boldsymbol{d}_S^t$ increases, along with a decreasing on $\boldsymbol{d}_O^t$. Meanwhile, $\boldsymbol{d}_V^t$ exhibits a V-shaped trend. For a clear view, we zoom into the first epoch and observe its variation details, as shown in Fig. 4 (b). In particular, there is a smooth transition from decrease to increase on the curve of $\boldsymbol{d}_V^t$. This phenomenon indicates that the in-training model indeed approaches the proxy space and then moves away from it to close the domain-invariant space as our proxy error control gradually comes into play.



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

Figure 4: Analysis for proxy denoising on the Al→Cl task in Office-Home. (a) The MMD-distance varying curves (epoch view) between the intermediate spaces to the source, oracle and proxy CLIP spaces, respectively, i.e., $\boldsymbol{d}_S^t$, $\boldsymbol{d}_O^t$ and $\boldsymbol{d}_V^t$. (b) The details of $\boldsymbol{d}_V^t$ (iteration view) during the first epoch. (c) The accuracy curves of typical signals during the adaptation. (d) The MMD-distance varying curves of ProDe ($\boldsymbol{d}_V^t$), ProDe-KL ($\boldsymbol{d}_V^{kl}$), ProDe-CLIP ($\boldsymbol{d}_V^{clip}$) during the first epoch (iteration view).

Correspondingly, we also provide the accuracy varying curves of two typical signals in Fig. 4 (c), including the target prediction (termed ProDe) and the denoised CLIP prediction (termed PRO). In this experiment, CLIP zero-shot result (termed CLIP) is the baseline. It is seen that PRO is better than ProDe in the early phase (0∼7 epoch) and surpassed by ProDe in the rest epochs. The results indicate that the guidance of reliable ViL predictions can boost the adaptation performance. Meanwhile, the PRO and ProDe curves closely resemble each other. It is understandable that the current prediction of the in-training target model, $\theta_t(\boldsymbol{x}_i)$, is utilized to adjust the raw ViL prediction (see Eq. (5)).

To better understand the impact of proxy denoising, we also conduct a comparison using two variations of ProDe. In ProDe-KL, the loss L_{Syn} is changed to conventional KL-Divergence, whilst in ProDe-CLIP, the training is based on the raw ViL prediction without proxy denoising. Employing the same MMD-distance quantification method mentioned above, we can plot two distance curves to the proxy space, termed $\boldsymbol{d}_V^{kl}$, $\boldsymbol{d}_V^{clip}$. In Fig. 4 (d), it is evident that ProDe moves away from the proxy space more quickly than the other two comparisons. This result suggests that ProDe is more responsive to proxy errors, resulting in agile error correction to match desired adapting direction. Additionally, the three curves at the early iterations are similar, indicating the impact of denoising e_{VI} is negligible during this stage. This observation provides empirical evidence supporting Case1 in our assumption.

5 Conclusion

The success of multimodal foundation models has sparked interest in transferring general multimodal knowledge to assist with domain-specific tasks, particularly in the field of transfer learning. However, for label-free scene scenarios such as SFDA discussed in this paper, the issue of filtering out noise from multimodal foundation models has been largely overlooked. To address this fundamental issue, this paper introduces a new ProDe approach. We first introduce a new approach called proxy denoising, which corrects the raw ViL predictions and provides reliable ViL guidance. This approach is based on a novel proxy confidence theory that we developed by modeling the impact of the proxy error between the proxy ViL space and the latent domain-invariant space, using the adaptation dynamics in the proxy alignment. Additionally, we propose a mutual distilling method to make use of the reliable proxy. Extensive experiment results indicate that our ProDe can achieve state-of-the-art results with significant improvements on four challenging datasets, confirming its effectiveness.

Limitation and future work. ProDe has shown impressive performance in multi-SFDA settings, highlighting its efficacy. However, it is important to note that it is specifically designed for a white-box and offline scenario, which may not be applicable in certain real-world contexts. For the kind of black-box application, such as models in the cloud, our proxy denoising may not work well since all details of the model, including the required logits features, are transparent to us. Additionally, the training supervised by mutual knowledge distilling regularization relies on the dataset prepared in advance, which limits the data flow over time. In the future, finding ways to extend our method to these new scenarios will be an interesting direction.

REPRODUCIBILITY STATEMENT

The code and data will be made available after the publication of this paper.

REFERENCES

Sk Miraj Ahmed, Dripta S Raychaudhuri, Sujoy Paul, Samet Oymak, and Amit K Roy-Chowdhury. Unsupervised multi-source domain adaptation without access to source data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10103–10112, 2021.

Alex Andonian, Shixing Chen, and Raffay Hamid. Robust cross-modal representation learning with progressive self-distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16430–16441, 2022.

Shuanghao Bai, Min Zhang, Wanqi Zhou, Siteng Huang, Zhirong Luan, Donglin Wang, and Badong Chen. Prompt-based distribution alignment for unsupervised domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 729–737, 2024.

Junbum Cha, Kyungjae Lee, Sungrae Park, and Sanghyuk Chun. Domain generalization by mutual-information regularization with pre-trained models. In *European Conference on Computer Vision*, pp. 440–457. Springer, 2022.

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*, pp. 295–305, 2022a.

Weijie Chen, Luojun Lin, Shicai Yang, Di Xie, Shiliang Pu, and Yueting Zhuang. Self-supervised noisy label learning for source-free unsupervised domain adaptation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 10185–10192. IEEE, 2022b.

M. Cherti, R. Beaumont, R. Wightman, M. Wortsman, G. Ilharco, C. Gordon, et al. Reproducible scaling laws for contrastive language-image learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2818–2829, 2023.

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*, pp. 7212–7222, 2022.

Yuntao Du, Haiyang Yang, Mingcai Chen, Hongtao Luo, Juan Jiang, Yi Xin, and Chongjun Wang. Generation, augmentation, and alignment: A pseudo-source domain based method for source-free domain adaptation. *Machine Learning*, pp. 1–21, 2023.

Zhekai Du, Xinyao Li, Fengling Li, Ke Lu, Lei Zhu, and Jingjing Li. Domain-agnostic mutual prompting for unsupervised domain adaptation. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2024.

Yaroslav Ganin and Victor S. Lempitsky. Unsupervised domain adaptation by backpropagation. In *International Conference on Machine Learning*, pp. 1180–1189, 2015.

Chunjiang Ge, Rui Huang, Mixue Xie, Zihang Lai, Shiji Song, Shuang Li, and Gao Huang. Domain adaptation via prompt learning. *arXiv:2202.06687*, 2022.

Kamran Ghasedi Dizaji, Amirhossein Herandi, Cheng Deng, Weidong Cai, and Heng Huang. Deep clustering via joint convolutional autoencoder embedding and relative entropy minimization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5736–5745, 2017.

Jiaxing Huang, Dayan Guan, Aoran Xiao, and Shijian Lu. Model adaptation: Historical contrastive learning for unsupervised domain adaptation without source data. *Advances in Neural Information Processing Systems*, 34:3635–3649, 2021.

Y. Iwasawa and Y. Matsuo. Test-time classifier adjustment module for model-agnostic domain generalization. In *Advances in Neural Information Processing Systems*, volume 34, pp. 2427–2440, 2021.

Xu Ji, Joao F Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9865–9874, 2019.

Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *Proceedings of the European Conference on Computer Vision*, pp. 709–727. Springer, 2022.

Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. Contrastive adaptation network for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4893–4902, 2019.

Vikash Kumar, Rohit Lal, Himanshu Patil, and Anirban Chakraborty. Conmix for source-free single and multi-target domain adaptation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 4178–4188, 2023.

Jogendra Nath Kundu, Akshay R Kulkarni, Suvaansh Bhambri, Deepesh Mehta, Shreyas Anand Kulkarni, Varun Jampani, and Venkatesh Babu Radhakrishnan. Balancing discriminability and transferability for source-free domain adaptation. In *International Conference on Machine Learning*, pp. 11710–11728. PMLR, 2022.

Vinod K Kurmi, Venkatesh K Subramanian, and Vinay P Namboodiri. Domain impression: A source data free domain adaptation method. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 615–625, 2021.

Zhengfeng Lai, Noranart Vesdapunt, Ning Zhou, Jun Wu, Cong Phuoc Huynh, Xuelu Li, Kah Kuen Fu, and Chen-Nee Chuah. Padclip: Pseudo-labeling with adaptive debiasing in clip for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16155–16165, 2023.

Qicheng Lao, Xiang Jiang, and Mohammad Havaei. Hypothesis disparity regularized mutual information maximization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 8243–8251, 2021.

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

Jingjing Li, Zhekai Du, Lei Zhu, Zhengming Ding, Ke Lu, and Heng Tao Shen. Divergence-agnostic unsupervised domain adaptation by adversarial attacks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):8196–8211, 2021.

Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10965–10975, 2022.

Rui Li, Qianfen Jiao, Wenming Cao, Hau-San Wong, and Si Wu. Model adaptation: Unsupervised domain adaptation without ource data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9638–9647, 2020a.

Rui Li, Qianfen Jiao, Wenming Cao, Hau-San Wong, and Si Wu. Model adaptation: Unsupervised domain adaptation without source data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9641–9650, 2020b.

Feng Liang, Bichen Wu, Xiaoliang Dai, Kunpeng Li, Yinan Zhao, Hang Zhang, Peizhao Zhang, Peter Vajda, and Diana Marculescu. Open-vocabulary semantic segmentation with mask-adapted clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7061–7070, 2023.

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

Mattia Litrico, Alessio Del Bue, and Pietro Morerio. Guiding pseudo-labels with uncertainty estimation for source-free unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7640–7650, 2023.

Jie Liu, Jinzong Cui, Mao Ye, Xiatian Zhu, and Song Tang. Shooting condition insensitive unmanned aerial vehicle object detection. *Expert Systems with Applications*, 246:123221, 2024.

R. Müller, S. Kornblith, and G. E Hinton. When does label smoothing help? In *Advances in Neural Information Processing Systems*, pp. 4696–4705, 2019.

S. Niu, J. Wu, Y. Zhang, Y. Chen, S. Zheng, P. Zhao, and M. Tan. Efficient test-time model adaptation without forgetting. In *International Conference on Machine Learning*, pp. 16888–16905. PMLR, 2022.

S. Niu, J. Wu, Y. Zhang, Z. Wen, Y. Chen, P. Zhao, and M. Tan. Towards stable test-time adaptation in dynamic wild world. In *International Conference on Learning Representations*, 2023.

Renjing Pei, Jianzhuang Liu, Weimian Li, Bin Shao, Songcen Xu, Peng Dai, Juwei Lu, and Youliang Yan. Clipping: Distilling clip-based models with a student base for video-language retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18983–18992, 2023.

Xingchao Peng, Ben Usman, Neela Kaushik, Judy Hoffman, Dequan Wang, and Kate Saenko. Visda: The visual domain adaptation challenge. *arXiv:1710.06924*, 2017.

Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1406–1415, 2019.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *Proceedings of the International Conference on Machine Learning*, pp. 8748–8763. PMLR, 2021.

Subhankar Roy, Martin Trapp, Andrea Pilzer, Juho Kannala, Nicu Sebe, Elisa Ricci, and Arno Solin. Uncertainty-guided source-free domain adaptation. In *Proceedings of the European Conference on Computer Vision*, pp. 537–555. Springer, 2022.

Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *Proceedings of the European Conference on Computer Vision*, pp. 213–226. Springer, 2010.

Kuniaki Saito, Donghyun Kim, Stan Sclaroff, Trevor Darrell, and Kate Saenko. Semi-supervised domain adaptation via minimax entropy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8050–8058, 2019.

S. Sanyal, A. R. Asokan, S. Bhambri, A. Kulkarni, J. N. Kundu, and R. V. Babu. Domain-specificity inducing transformers for source-free domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 18928–18937, 2023.

Mainak Singha, Harsh Pal, Ankit Jha, and Biplab Banerjee. AD-CLIP: Adapting domains in prompt space using CLIP. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshop*, pp. 4355–4364, 2023.

S Tang, Yuji Shi, Zhiyuan Ma, Jian Li, Jianzhi Lyu, Qingdu Li, and Jianwei Zhang. Model adaptation through hypothesis transfer with gradual knowledge distillation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5679–5685. IEEE, 2021.

Song Tang, Yan Yang, Zhiyuan Ma, Norman Hendrich, Fanyu Zeng, Shuzhi Sam Ge, Changshui Zhang, and Jianwei Zhang. Nearest neighborhood-based deep clustering for source data-absent unsupervised domain adaptation. *arXiv:2107.12585*, 2021.

Song Tang, Yan Zou, Zihao Song, Jianzhi Lyu, Lijuan Chen, Mao Ye, Shouming Zhong, and Jianwei Zhang. Semantic consistency learning on manifold for source data-free unsupervised domain adaptation. *Neural Networks*, 152:467–478, 2022.

Song Tang, An Chang, Fabian Zhang, Xiatian Zhu, Mao Ye, and Changshui Zhang. Source-free domain adaptation via target prediction distribution searching. *International Journal of Computer Vision*, 132(3):654–672, 2024a.

Song Tang, Yuji Shi, Zihao Song, Mao Ye, Changshui Zhang, and Jianwei Zhang. Progressive source-aware transformer for generalized source-free domain adaptation. *IEEE Transactions on Multimedia*, 26:4138–4152, 2024b. doi: 10.1109/TMM.2023.3321421.

Song Tang, Wenxin Su, Mao Ye, and Xiatian Zhu. Source-free domain adaptation with frozen multimodal foundation model. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024c.

Jiayi Tian, Jing Zhang, Wen Li, and Dong Xu. Vdm-da: Virtual domain modeling for source data-free domain adaptation. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(6): 3749–3760, 2021.

Jiayi Tian, Jing Zhang, Wen Li, and Dong Xu. Vdm-da: virtual domain modeling for source data-free domain adaptation. *IEEE Trans. Circuits Syst. Video Technol.*, 32(6):3749–3760, 2022.

Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5385–5394, 2017.

D. Wang, E. Shelhamer, S. Liu, B. Olshausen, and T. Darrell. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726*, 2020.

F. Wang, Z. Han, Y. Gong, and Y. Yin. Exploring domain-invariant parameters for source free domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7151–7160, 2022a.

Jue Wang, Haofan Wang, Jincan Deng, Weijia Wu, and Debing Zhang. Efficientclip: Efficient cross-modal pre-training by ensemble confident learning and language modeling. *arXiv preprint arXiv:2109.04699*, 2021.

Q. Wang, O. Fink, L. Van Gool, and D. Dai. Continual test-time domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7201–7211, 2022b.

Yan Wang, Jian Cheng, Yixin Chen, Shuai Shao, Lanyun Zhu, Zhenzhou Wu, Tao Liu, and Haogang Zhu. Fvp: Fourier visual prompting for source-free unsupervised domain adaptation of medical image segmentation. *IEEE Transactions on Medical Imaging*, 2023.

Zhaoqing Wang, Yu Lu, Qiang Li, Xunqiang Tao, Yandong Guo, Mingming Gong, and Tongliang Liu. CRIS: Clip-driven referring image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11686–11695, 2022c.

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*, pp. 9010–9019, 2021.

Siying Xiao, Mao Ye, Qichen He, Shuaifeng Li, Song Tang, and Xiatian Zhu. Adversarial experts model for black-box domain adaptation. In *ACM Multimedia*, 2024.

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/CVF International Conference on Computer Vision*, pp. 1426–1435, 2019.

Shiqi Yang, Joost van de Weijer, Luis Herranz, Shangling Jui, et al. Exploiting the intrinsic neighborhood structure for source-free domain adaptation. In *Advances in Neural Information Processing Systems*, volume 34, pp. 29393–29405, 2021a.

Shiqi Yang, Yaxing Wang, Joost Van De Weijer, Luis Herranz, and Shangling Jui. Generalized source-free domain adaptation. In *Proceedings of IEEE International Conference on Computer Vision*, pp. 8978–8987, 2021b.

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.

Li Yi, Gezheng Xu, Pengcheng Xu, Jiaqi Li, Ruizhi Pu, Charles Ling, A Ian McLeod, and Boyu Wang. When source-free domain adaptation meets learning with noisy labels. *International Conference on Learning Representations*, 2023.

M. Zhan, Z. Wu, R. Hu, P. Hu, H. T. Shen, and X. Zhu. Towards dynamic-prompting collaboration for source-free domain adaptation. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 182–188, 2024.

Yixin Zhang, Zilei Wang, and Weinan He. Class relationship embedded learning for source-free unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7619–7629, 2023.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022.

Lihua Zhou, Nianxin Li, Mao Ye, Xiatian Zhu, and Song Tang. Source-free domain adaptation with class prototype discovery. *Pattern recognition*, 145:109974, 2024.

A PROOF OF THEOREM 1

Restatement of Theorem 1 *Given a proxy alignment formulated in* `Section 3.1`*. The source domain (D_S), the domain-invariant space (D_I), the proxy space (D_V) and the in-training model (D_{T_t}) satisfy the probability distributions $P(S)$, $P(I)$, $P(V)$ and $P(T_t)$, respectively, where S, I, V and T_t are corresponding random variables. The factor describing the credibility of $P(V)$ has a relation below.*

$$P\left(G_{P(V)} = True, t\right) \propto \frac{P(T_t)}{P(S)}.$$

Proof 1 *We use the spatial distance relation to represent the variation in confidence of ViL prediction, which is causally linked to the variation in distance to D_I, as demonstrated in Fig. 2 (a). At any given time t, the correction factor can be expressed as*

$$P\left(G_{P(V)} = True, t\right) \propto \frac{|Distance(D_{T_t}, D_I)|}{|Distance(D_S, D_I)|} = \frac{|\boldsymbol{d}_I^t|}{|\boldsymbol{d}_S|}. \tag{7}$$

where $\boldsymbol{d}_I^t$ and $\boldsymbol{d}_S$ refers to the distance from D_{T_t} and D_S to D_I, respectively. Easily finding, Eq. (7) satisfies the reliability feature of gradually decreasing from 1 to 0 as D_{T_t} evolves from D_S to D_I.

To account for the fact that spaces are defined by probability distributions, we instantiate the space distance using the widely used measurement of KL-divergence. This gives us:

$$\begin{aligned}
\frac{|\boldsymbol{d}_I^t|}{|\boldsymbol{d}_S|} &= \frac{KL\left(P(T_t)||P(I)\right)}{KL\left(P(S)||P(I)\right)} = \frac{\int_{T_t} P(T_t) \log \frac{P(T_t)}{P(I)} dT_t}{\int_S P(S) \log \frac{P(S)}{P(I)} dS} \\
&= \frac{-\int_{T_t} P(T_t) \log P(T_t) dT_t + \int_{T_t} P(T_t) \log P(I) dT_t}{-\int_S P(S) \log P(S) dS + \int_S P(S) \log P(I) dS} \\
&= \frac{H(T_t) + \log P(I)}{H(S) + \log P(I)}
\end{aligned} \tag{8}$$

where $H(\cdot)$ stands for the information entropy. Since D_I is an domain-invariant space, $P(I)$ always outputs 1 for the category of interesting, such that $\log P(I) = 0$. Eq. (8) can be further converted to

$$\frac{H(T_t) + \log P(I)}{H(S) + \log P(I)} = \frac{H(T_t)}{H(S)} \propto \frac{P(T_t)}{P(S)} \tag{9}$$

B PSEUDO TRAINING CODE OF PRODE

Based on the proposed objective presented in Eq. (6), we achieve the model training iteration-wise. The training process are summarized as Alg. 1.

Algorithm 1 Training of ProDe

Input: Source model θ_s, ViL model θ_v, target dataset $\mathcal{X}_t$, C prompts with context $\boldsymbol{v}$, #iteration M.
Procedure:

1: **Initialisation**: Set target model $\theta_t = \theta_s$, prompt context $\boldsymbol{v}$ ="a photo of a".
2: **for** m = 1:M **do**
3: Sample a batch $\mathcal{X}_t^b$ from $\mathcal{X}_t$.
4: Forward updated prompts and $\mathcal{X}_t^b$ through θ_v.
5: Forward $\mathcal{X}_t^b$ through θ_t.
6: Conduct proxy denoising for the ViL predictions of $\mathcal{X}_t^b$ (Eq. (5)).
7: Update model θ_t and prompt context $\boldsymbol{v}$ by optimizing objective L_{ProDe} (Eq. (6)).
8: **end for**
9: **return** Adapted target model θ_t.

C EVALUATION DATASETS

In this paper, the ProDe method is evaluated on four widely used benchmarks for domain adaptation problems as follows.

- **Office-31** (Saenko et al., 2010) is a small-scaled dataset including three domains, i.e., Amazon (A), Webcam (W), and Dslr (D), all of which are taken of real-world objects in various office environments. The dataset has 4,652 images of 31 categories in total.

- **Office-Home** (Venkateswara et al., 2017) is a medium-scale dataset that is mainly used for domain adaptation, all of which contains 15k images belonging to 65 categories from working or family environments. The dataset has four distinct domains, i.e., Artistic images (Ar), Clip Art (Cl), Product images (Pr), and Real-word images (Rw).

- **VisDA** (Peng et al., 2017) is a large-scale dataset with 12 types of synthetic to real transfer recognition tasks. The source domain contains 152k synthetic images (Sy), whilst the target domain has 55k real object images (Re) from the famous Microsoft COCO dataset.

- **DomainNet-126** (Saito et al., 2019) is another challenging large-scale dataset. It has been created by removing severe noisy labels from the original DomainNet dataset (Peng et al., 2019) containing 600k images of 345 classes from 6 domains of varying image styles. The dataset is further divided into four domains: Clipart (C), Painting (P), Real (R), and Sketch (S), and contains 145k images from 126 classes.

D IMPLEMENTATION DETAILS

Souce model pre-training. For all transfer tasks on the four evaluation datasets, we train the source model θ_s on the source domain in a supervised manner using the following objective of the classic cross-entropy loss with smooth label, totally the same as other methods (Liang et al., 2020; Yang et al., 2021a; Tang et al., 2022).

$$L_{\mathrm{s}}\left(\mathcal{X}_s, \mathcal{Y}_s; \theta_s\right) = -\mathbb{E}_{\boldsymbol{x}_i^s \in \mathcal{X}_s} \sum_{c=1}^{C} \tilde{\mathbb{1}}\left[c = y_i^s\right] \log p_{i,c}^s,$$

where $p_{i,c}^s$ is the c-th element of $\boldsymbol{p}_i^s = \phi(\theta_s(\boldsymbol{x}_i^s))$ that is the category probability vector of input instance $\boldsymbol{x}_i^s$ after θ_s conversion with ending softmax operation ϕ; $\tilde{\mathbb{1}}\left[c = y_i^s\right] = (1-\sigma)\,\mathbb{1}\left[c = y_i^s\right] + \sigma/C$ is the smooth label (Müller et al., 2019), in which $\mathbb{1}\left[c = y_i^s\right]$ is a one-hot encoding of hard label y_i^s and $\sigma = 0.1$. The source dataset is divided into the training set and testing set in a 0.9:0.1 ratio.

Network setting. The ProDe framework involves two networks, namely the target model and the ViL model. In practice, the target model comprises a deep architecture-based feature extractor and a classifier that consists of a fully connected layer and a weight normalization layer. As seen in previous work (Xu et al., 2019; Liang et al., 2020; Roy et al., 2022), the deep architecture is transferred from the deep models pre-trained on ImageNet. Specifically, ResNet-50 is used on Office-31 and Office-Home, whilst ResNet-101 is employed on VisDA and Domain-Net. As for the ViL model, we choose CLIP to instantiate it where the text encoder adopts Transformer structure and the image encoder takes ResNet or ViT-B/32 according to the specific implementations, which are marked by suffix of "-R" or "-V".

Hyper-parameter setting. The ProDe model involves four parameters: The correction strength factor ω in Eq. (5) and two trade-off parameters α, β and γ in objective L_{ProDe} (Eq. (6)). On all four datasets, we set $(\omega, \alpha, \beta) = (1, 1, 0.4)$. Parameter γ is sensitive to the dataset scale, also noted in the TPDS method (Tang et al., 2024a). In practice, the setting of $\gamma = 1.0/1.0/0.1/0.5$ is employed on Office-31, Office-Home, VisDA and DomainNet-126, respectively.

Training setting. We chose a batch size of 64 and utilized the SGD optimizer with a momentum of 0.9 and 15 training epochs on all datasets. The learnable prompt context is initiated by the template of 'a photo of a [CLASS].', as suggested by (Radford et al., 2021), where the [CLASS] term is replaced with the name of the class being trained. All experiments are conducted with PyTorch on a single GPU of NVIDIA RTX. Each transfer task is repeated five times, and the final result is calculated as the average of the five attempts.

Table 10: Full results (%) of closed-set SFDA on **VisDA**. **SF** means source-free.

Method	Venue	**SF**	plane	bcycl	bus	car	horse	knife	mcycl	person	plant	sktbrd	train	truck	Perclass
Source	-	-	60.7	21.7	50.8	68.5	71.8	5.4	86.4	20.2	67.1	43.3	83.3	10.6	49.2
SHOT	ICML20	✓	95.0	87.4	80.9	57.6	93.9	94.1	79.4	80.4	90.9	89.8	85.8	57.5	82.7
NRC	NIPS21	✓	96.8	91.3	82.4	62.4	96.2	95.9	86.1	**90.7**	94.8	94.1	90.4	59.7	85.9
GKD	IROS21	✓	95.3	87.6	81.7	58.1	93.9	94.0	80.0	80.0	91.2	91.0	86.9	56.1	83.0
AaD	NIPS22	✓	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
AdaCon	CVPR22	✓	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
CoWA	ICML22	✓	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
ELR	ICLR23	✓	97.1	89.7	82.7	62.0	96.2	97.0	87.6	81.2	93.7	94.1	90.2	58.6	85.8
PLUE	CVPR23	✓	94.4	91.7	89.0	70.5	96.6	94.9	92.2	88.8	92.9	95.3	91.4	61.6	88.3
CPD	PR24	✓	96.7	88.5	79.6	69.0	95.9	96.3	87.3	83.3	94.4	92.9	87.0	58.7	85.5
TPDS	IJCV24	✓	97.6	91.5	89.7	83.4	97.5	96.3	92.2	82.4	**96.0**	94.1	90.9	40.4	87.6
DAPL-R	TNNLS23	✗	97.8	83.1	88.8	77.9	97.4	91.5	94.2	79.7	88.6	89.3	92.5	62.0	86.9
PADCLIP-R	ICCV23	✗	96.7	88.8	87.0	82.8	97.1	93.0	91.3	83.0	95.5	91.8	91.5	63.0	88.5
ADCLIP-R	ICCVW23	✗	98.1	83.6	**91.2**	76.6	98.1	93.4	**96.0**	81.4	86.4	91.5	92.1	64.2	87.7
PDA-R	AAAI24	✗	97.2	82.3	89.4	76.0	97.4	87.5	95.8	79.6	87.2	89.0	93.3	62.1	86.4
DAMP-R	CVPR24	✗	97.3	91.6	89.1	76.4	97.5	94.0	92.3	84.5	91.2	88.1	91.2	67.0	88.4
DIFO-R	CVPR24	✓	97.7	87.6	90.5	83.6	96.7	95.8	94.8	74.1	92.4	93.8	92.9	65.5	88.8
DIFO-V	CVPR24	✓	97.5	89.0	90.8	83.5	97.8	97.3	93.2	83.5	95.2	**96.8**	93.7	65.9	90.3
ProDe-R	–	✓	97.3	89.6	84.5	**86.1**	96.4	95.9	92.1	88.6	94.1	93.8	93.9	66.6	89.9
ProDe-V	–	✓	**98.3**	**92.0**	87.3	84.4	**98.5**	**97.5**	94.0	86.4	95.0	96.1	**94.2**	**75.6**	**91.6**

Table 11: Full results (%) of partial-set SFDA and open-set SFDA on **Office-Home**.

Partial-set	Venue	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.
Source	–	45.2	70.4	81.0	56.2	60.8	66.2	60.9	40.1	76.2	70.8	48.5	77.3	62.8
SHOT	ICML20	64.8	85.2	92.7	76.3	77.6	88.8	79.7	64.3	89.5	80.6	66.4	85.8	79.3
HCL	NIPS21	65.6	85.2	92.7	77.3	76.2	87.2	78.2	66.0	89.1	81.5	68.4	87.3	79.6
CoWA	ICML22	69.6	93.2	92.3	78.9	81.3	92.1	79.8	71.7	90.0	83.8	**72.2**	**93.7**	83.2
AaD	NIPS22	67.0	83.5	93.1	80.5	76.0	87.6	78.1	65.6	90.2	83.5	64.3	87.3	79.7
CRS	CVPR23	68.6	85.1	90.9	80.1	79.4	86.3	79.2	66.1	90.5	82.2	69.5	89.3	80.6
DIFO-V	CVPR24	70.2	**91.7**	91.5	**87.8**	**92.6**	**92.9**	87.3	70.7	92.9	88.5	69.6	91.5	85.6
ProDe-V	–	**71.4**	90.4	**94.5**	86.9	89.3	92.8	**89.4**	**74.2**	**93.7**	**89.5**	71.8	90.8	**86.2**
Open-set	Venue	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.
Source	–	36.3	54.8	69.1	33.8	44.4	49.2	36.8	29.2	56.8	51.4	35.1	62.3	46.6
SHOT	ICML20	64.5	80.4	84.7	63.1	75.4	81.2	65.3	59.3	83.3	69.6	64.6	82.3	72.8
HCL	NIPS21	64.0	78.6	82.4	64.5	73.1	80.1	64.8	59.8	75.3	78.1	69.3	81.5	72.6
CoWA	ICML22	63.3	79.2	85.4	67.6	83.6	82.0	66.9	56.9	81.1	68.5	57.9	85.9	73.2
AaD	NIPS22	63.7	77.3	80.4	66.0	72.6	77.6	69.1	62.5	79.8	71.8	62.3	78.6	71.8
CRS	CVPR23	65.2	76.6	80.2	66.2	75.3	77.8	70.4	61.8	79.3	71.1	61.1	78.3	73.2
DIFO-V	CVPR24	64.5	**86.2**	**87.9**	68.2	79.3	**86.1**	67.2	62.1	**88.3**	71.9	65.3	84.4	75.9
ProDe-V	–	**75.4**	85.8	86.5	**83.2**	**86.3**	**86.1**	**83.6**	**74.5**	86.8	**81.9**	**74.6**	**86.5**	**82.6**

Table 12: Results (%) of CLIP on the four evaluation datasets. The backbone of CLIP image-encoder in CLP-R and CLP-V are the same as ProDe-R and ProDe-V, respectively.

Method	Venue	**Office-31**				**Office-Home**					**VisDA**	**DomainNet-126**				
		→A	→D	→W	→Avg.	→Ar	→Cl	→Pr	→Rw	→Avg.	Sy→Re	→C	→P	→R	→S	→Avg.
CLIP-R	ICML21	73.1	73.9	67.0	71.4	72.5	51.9	81.5	82.5	72.1	83.7	67.9	70.2	87.1	65.4	72.7
ProDe-R	–	**80.9**	**95.3**	**93.9**	**90.0**	**82.4**	**67.0**	**91.4**	**90.7**	**82.9**	**89.9**	**80.3**	**79.2**	**91.0**	**75.4**	**81.5**
CLIP-V	ICML21	76.0	82.7	80.6	79.8	74.6	59.8	84.3	85.5	76.1	82.9	74.7	73.5	85.7	71.2	76.3
ProDe-V	–	**83.0**	**98.2**	**96.6**	**92.6**	**84.3**	**74.9**	**93.2**	**92.3**	**86.2**	**91.6**	**85.3**	**83.2**	**92.4**	**79.1**	**85.0**

E Supplemental Experiments

E.1 Supplementation of Full Experiment Results

Full results on VisDA. Tab. 10 is the supplement of average results on the VisDA dataet (reported in Tab. 2), displaying the full classification results over the 12 categories. Specifically, the ProDe-R and ProDe-V totally obtain best results on 7/12 categories, leading to the advantage on average accuracy. On some cases, such as bcycl, car and truck, ProDe has presents significant advantages over the previous methods.

Table 13: Generalized SFDA results (%) on **Office-Home**. S, T are the results of the adapted target domain on the source and target domains, respectively; H means the harmonic mean accuracy; **WAD** is short for With Anti-forgetting Design.

Method	Venue	**WAD**	Ar→Cl			Ar→Pr			Ar→Rw			Cl→Ar			Cl→Pr			Cl→Rw		
			S	T	H	S	T	H	S	T	H	S	T	H	S	T	H	S	T	H
Source	–	✗	97.9	43.7	60.4	97.9	67.0	79.5	97.9	73.9	84.2	97.1	49.9	65.9	97.1	60.1	74.2	97.1	62.5	76.0
SHOT	ICML20	✗	78.6	55.0	64.7	83.8	78.7	81.2	88.6	81.3	84.8	78.0	69.1	73.2	76.6	78.9	77.7	77.1	79.1	78.1
GKD	IROS21	✗	81.9	56.5	66.9	87.0	78.3	82.4	91.4	82.2	86.6	80.3	69.2	74.3	80.9	80.4	80.6	81.4	78.7	80.1
NRC	NIPS21	✗	86.9	57.2	69.0	92.9	79.3	85.6	95.3	81.3	87.7	81.7	68.9	74.8	89.1	80.6	84.6	88.8	80.2	84.3
AdaCon	CVPR22	✗	75.2	47.2	57.9	91.0	75.1	82.3	93.9	75.5	83.7	79.4	60.7	68.8	88.2	73.3	80.0	83.4	73.2	78.0
CoWA	ICML22	✗	89.0	57.3	69.7	93.0	79.3	85.6	94.6	81.0	87.3	86.6	69.3	77.0	86.3	77.9	81.9	83.4	79.6	81.5
PLUE	CVPR23	✗	**91.8**	49.1	63.9	**96.3**	73.5	83.4	**97.2**	78.2	86.6	**93.9**	63.0	75.3	**95.6**	73.5	83.1	**94.3**	74.5	83.2
TPDS	IJCV24	✗	78.0	59.3	67.4	83.6	80.3	81.9	88.1	82.1	85.0	75.4	70.6	72.9	77.3	79.4	78.3	76.2	80.9	78.5
GDA	ICCV21	✓	68.8	54.7	60.9	72.0	75.6	73.8	74.5	78.5	76.4	77.2	66.6	71.5	79.7	74.0	76.7	78.5	78.4	78.4
PSAT-ViT	TMM24	✓	81.6	73.1	77.1	87.0	88.1	87.6	88.1	89.2	88.7	82.7	82.1	**82.6**	82.7	88.8	**85.7**	83.5	88.9	**86.1**
DIFO-R	CVPR24	✗	73.8	62.6	67.8	76.3	87.5	81.5	79.7	87.1	83.2	73.1	79.5	76.2	64.8	87.9	74.6	66.3	87.4	75.4
DIFO-V	CVPR24	✗	73.8	70.6	72.2	75.0	90.6	82.1	80.7	88.8	84.6	70.4	82.5	75.9	64.3	90.6	75.2	65.9	88.8	75.7
ProDe-R	–	✗	77.5	66.0	71.3	82.9	91.2	86.9	86.8	90.8	88.8	76.0	81.4	78.6	73.5	91.4	81.4	72.5	90.5	80.5
ProDe-V	–	✗	79.7	**74.6**	**77.1**	84.9	**92.9**	**88.7**	89.0	**92.4**	**90.7**	76.1	**84.4**	80.0	74.3	**93.0**	82.6	73.5	**92.2**	81.8

Method	Venue	**WAD**	Pr→Ar			Pr→Cl			Pr→Rw			Rw→Ar			Rw→Cl			Rw→Pr			Avg.		
			S	T	H	S	T	H	S	T	H	S	T	H	S	T	H	S	T	H	S	T	H
Source	–	✗	99.2	51.7	68.0	99.2	40.9	57.9	99.2	72.6	83.8	98.1	64.2	77.6	98.1	46.3	62.9	98.1	78.1	87.0	98.1	59.2	73.1
SHOT	ICML20	✗	88.2	68.2	76.9	80.7	53.6	64.4	90.1	81.6	85.6	91.7	73.5	81.6	84.8	59.4	69.8	92.2	83.5	87.6	84.2	71.8	77.5
GKD	IROS21	✗	89.4	67.4	76.8	84.1	55.4	66.8	92.0	82.6	87.0	93.7	74.3	82.9	86.2	60.3	70.9	93.5	84.2	88.6	86.8	72.5	79.0
NRC	NIPS21	✗	89.1	66.6	76.2	90.1	57.3	70.1	96.6	82.0	88.7	97.8	71.0	82.3	90.7	57.9	70.7	97.1	84.9	90.6	91.3	72.3	80.7
AdaCon	CVPR22	✗	93.4	60.2	73.2	88.4	45.2	59.8	94.3	76.6	84.5	93.3	65.6	77.0	84.1	48.3	61.3	94.5	79.1	86.1	88.2	65.0	74.8
CoWA	ICML22	✗	94.6	68.1	79.2	93.2	56.4	70.3	95.0	82.6	88.3	96.3	72.9	83.0	93.7	61.3	74.1	95.6	83.7	89.3	91.8	72.4	81.0
PLUE	CVPR23	✗	**98.7**	62.2	76.3	**98.5**	48.3	64.8	**98.9**	78.6	87.6	**98.1**	68.6	80.7	**95.1**	51.8	67.1	**97.8**	81.5	88.9	**96.3**	66.9	79.0
TPDS	IJCV24	✗	87.7	69.8	77.7	81.4	56.8	66.9	90.4	82.1	86.0	92.3	74.5	82.5	83.2	61.2	70.5	92.0	85.3	88.5	83.8	73.5	78.3
GDA	ICCV21	✓	87.8	65.1	74.8	86.3	53.2	66.1	90.3	81.6	85.7	83.2	72.0	77.2	78.3	60.2	68.1	83.4	82.8	83.1	80.0	70.2	74.4
PSAT-ViT	TMM24	✓	89.6	83.0	**86.2**	87.4	72.0	**79.0**	92.5	89.6	91.0	87.4	83.3	85.3	84.2	73.7	78.6	89.6	91.3	90.5	86.4	83.6	85.0
DIFO-R	CVPR24	✗	85.6	78.3	81.8	76.6	63.4	69.4	86.0	88.1	87.0	89.4	80.0	84.4	80.7	63.3	70.9	87.2	87.7	87.4	78.3	79.4	78.8
DIFO-V	CVPR24	✗	84.3	80.9	82.5	77.4	70.1	73.6	87.2	88.9	88.0	88.5	83.4	85.9	80.9	70.5	75.3	87.4	91.2	89.3	78.0	83.1	80.5
ProDe-R	–	✗	90.0	82.2	85.9	82.3	67.3	74.1	91.1	90.8	90.9	92.4	83.6	87.7	82.7	67.7	74.4	91.4	91.6	91.5	83.3	82.9	83.1
ProDe-V	–	✗	88.7	**83.8**	**86.2**	81.8	**74.8**	78.1	91.6	**92.4**	**92.0**	92.6	**84.9**	**88.6**	84.4	**75.2**	**79.5**	92.4	**93.7**	**93.0**	84.1	**86.2**	**85.1**

Table 14: Full results (%) of the TTA setting on **Office-Home**.

Method	Venue	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.
Tent	ICLR20	47.6	63.2	72.3	57.1	63.7	65.9	55.9	46.6	72.7	67.7	51.8	77.1	61.7
T3A	NeurIPS21	49.7	73.2	77.0	55.5	67.7	68.5	55.8	46.1	75.7	67.0	49.6	78.0	63.8
CoTTA	CVPR22	44.5	62.5	72.3	55.4	63.0	65.3	54.9	46.0	76.7	66.0	49.5	76.7	60.5
EATA	ICML22	46.4	62.5	72.2	55.3	65.8	65.8	53.8	43.4	76.4	66.5	50.5	76.4	60.7
SAR	ICLR23	45.3	61.9	71.9	55.4	66.4	65.7	53.7	42.7	72.5	66.4	49.3	76.2	60.3
ProDe-V	–	**64.5**	**84.9**	**84.7**	**76.1**	**85.1**	**83.7**	**75.5**	**64.0**	**85.1**	**77.4**	**67.3**	**87.1**	**78.0**

Table 15: Reliance analysis results (%) on **Office-31** in the Closed-set SFDA setting.

Method	Venue	A→D	A→W	D→A	D→W	W→A	W→D	Avg.
DIFO w/ CLIP	CVPR24	**97.2**	95.5	83.0	**97.2**	**83.2**	98.8	92.5
ProDe w/ CLIP	–	96.6	**96.4**	**83.1**	96.9	82.9	**99.8**	**92.6**
DIFO w/ OpenCLIP	CVPR24	**96.8**	**98.1**	82.9	**98.7**	82.7	**100.**	93.2
ProDe w/ OpenCLIP	–	95.2	97.1	**87.0**	96.0	**87.3**	97.0	**93.3**

Full results of partial-set and open-set SFDA. Tab. 11 is the supplementation of these average accuracy in Tab. 5, reporting the full classification accuracy over 12 transfer tasks in Office-Home. In the partial-set setting (the top in the table), ProDe-V beats other methods on half of the tasks, whilst DIFO-V and CoWA dominate the rest of the tasks. As taking the open-set setting (the bottom in the table), ProDe-V gets the top results on 8/12 tasks. Moreover, besides the Rw→Pr task, the rest of the best eight tasks have **8.0**% increase at least, compared with the best-second methods. So, the ProDe gains substantial improvement in average performance.

Full results of the comparison to CLIP's zero-shot. As the supplement of average results in the comparison to CLIP (reported in Tab. 4), Tab. 12 presents the full quantitative results categorized by the target domain name. For instance, for domain A in Office-31, we averaged the adapting accuracy of other domains to A, such as D→A, W→A, notated by →A. As reported in Tab. 12, both ProDe-R and ProDe-V obtain the best results across all groups, compared to the respective CLIP version.

Table 16: Reliance analysis results (%) on **Office-Home** and **VisDA** in the Closed-set SFDA setting.

Method	Venue	Office-Home													VisDA
		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.	Sy→Re
DIFO w/ CLIP	CVPR24	70.6	90.6	88.8	82.5	90.6	88.8	80.9	70.1	88.9	83.4	70.5	91.2	83.1	90.3
ProDe w/ CLIP	–	**74.6**	**92.9**	**92.4**	**84.4**	**93.0**	**92.2**	**83.8**	**74.8**	**92.4**	**84.9**	**75.2**	**93.7**	**86.2**	**91.6**
DIFO w/ OpenCLIP	CVPR24	80.2	94.2	91.7	85.4	93.7	91.6	82.7	79.2	91.7	85.3	80.4	94.8	87.6	90.7
ProDe w/ OpenCLIP	–	**82.5**	**96.0**	**94.5**	**87.9**	**95.8**	**94.4**	**87.8**	**82.8**	**94.2**	**88.6**	**83.3**	**96.3**	**90.3**	**93.0**

Table 17: Reliance analysis results (%) on **DomainNet-126** in the Closed-set SFDA setting.

Method	Venue	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.
DIFO w/ CLIP	CVPR24	76.6	87.2	74.9	80.0	87.4	75.6	80.8	77.3	75.5	80.5	76.7	87.3	80.0
ProDe w/ CLIP	–	**83.2**	**92.4**	**79.0**	**85.0**	**92.3**	**79.3**	**85.5**	**83.1**	**79.1**	**85.5**	**83.4**	**92.4**	**85.0**
DIFO w/ OpenCLIP	CVPR24	91.2	91.5	79.4	85.2	91.2	79.7	85.7	82.7	80.5	85.9	81.3	91.4	84.6
ProDe w/ OpenCLIP	–	**86.7**	**93.7**	**84.4**	**89.2**	**93.7**	**84.5**	**89.6**	**86.6**	**84.4**	**89.5**	**86.7**	**93.7**	**88.6**

Table 18: Results (%) of OpenCLIP on the four evaluation datasets.

Method	Venue	Office-31				Office-Home					VisDA	DomainNet-126				
		→A	→D	→W	→Avg.	→Ar	→Cl	→Pr	→Rw	→Avg.	Sy→Re	→C	→P	→R	→S	→Avg.
OpenCLIP	CVPR23	85.7	91.2	91.8	89.6	83.8	76.1	93.5	92.3	86.4	86.7	86.4	82.0	92.3	80.8	85.4
ProDe w/ OpenCLIP	–	**87.2**	**96.1**	**96.5**	**93.3**	**88.1**	**82.9**	**96.0**	**94.4**	**90.3**	**93.0**	**89.4**	**86.7**	**93.7**	**84.4**	**88.6**

Full results of generalized SFDA. As a supplement to the average results of the generalized SFDA results (reported in Tab. 6), Tab. 13 presents the full results on 12 transfer tasks, including S-, T- and H-accuracy. In terms of H-accuracy, ProDe-V achieves the best results on 8 out of 12 tasks. These results are not only due to significant improvements in the target domain (see T-accuracy), but also derive from a balanced drop in the source domain (see S-accuracy).

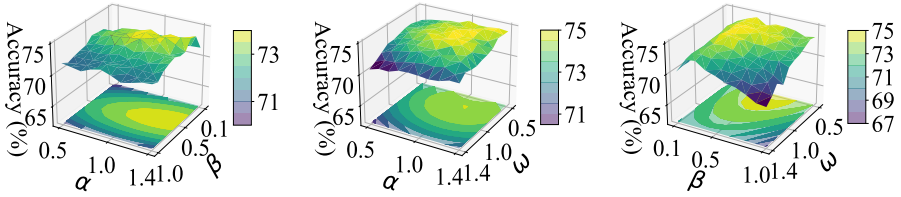
Full results of TTA. As a supplement to the average results of the TTA results (reported in Tab. **??**), Tab. 14 presents the full results on the Office-Home dataset. On all 12 transfer tasks, ProDe-V achieves substantial increase, leading to 14.2% gains on top of the second-best method T3A.

E.2 Expanded Model Analysis

Reliance analysis on ViL models. As illustrated in the right of Fig. 2, our proxy denoising is executed at the logit level, which means that the proposed method does not depend on a specific ViL model, such as CLIP, since it does not utilize the internal structure of these models. To validate this claim, we conduct an extensive test with OpenCLIP (Cherti et al., 2023). Meanwhile, we selected DIFO, the previous best ViL-based method, for comparison. Tab. 15∼Tab. 17 present comparison results across all four datasets. Regardless of whether we use CLIP or OpenCLIP as the ViL model, ProDe beats DIFO in average accuracy. Furthermore, the relative gains are consistent. In comparison to DIFO, ProDe improves approximately by **0.1**%, **3.0**%, **2.0**% and **4.5**% on Office-31, Office-Home, VisDA and DomainNet-126, respectively. This trend suggests that our method is generic with the ViL model, and can readily benefit from the advancement in ViL models.

In addition, Tab. 18 displays a comparison of the zero-shot results from OpenCLIP. In all tasks (which are detailed in the "Full results of the comparison to CLIP's zero-shot" section of `Section E.1`), ProDe w/ OpenCLIP surpasses OpenCLIP. This suggests that the task-specific target model effectively incorporates generic knowledge in ViL models.

Effect of prompt learning. In ProDe, prompt learning contributes to knowledge synchronization. To isolate its effectiveness, we propose a variation method ProDe-V w/o prompt that removes prompt learning. As shown in Tab. 19, the absence of prompt learning results in **0.9**% decrease in average accuracy. These results indicate that this prompt learning might reduce the proxy error by tuning space D_V close to the domain-invariant space D_I, meeting our expectations.



(a) α–β (b) α–ω (c) β–ω

Figure 5: Sensitivity analysis of hyper-parameters α, β and ω.

Table 19: Ablation results (%) of prompt learning on **Office-31**, **Office-Home** and **VisDA**.

#	Method	**Office-31**	**Office-Home**	**VisDA**	Avg.
1	**ProDe**-V w/o prompt	92.3	84.4	90.9	89.2
2	**ProDe**-V	**92.6**	**86.2**	**91.6**	**90.1**

Table 20: Comparison of training resource demands (per iter.) on Ar→Cl in **Office-Home**.

#	Item / Method	SHOT	AaD	**ProDe**
1	GPU memory consumption↓ (G)	**7.868**	9.622	9.851
2	Training times↓ (s)	**0.407**	0.547	0.491

Training resource demands. To evaluate the training resource demands, we select two typical methods without using ViL model, SHOT and AaD, as comparisons. We conducted the test using the transfer task Ar → Cl from Office-Home, under the same testing conditions, including mini-batch size. The results, as shown in Tab. 20, indicate that despite using a large ViL model, our approach does not incur significant additional training costs and requires a similar amount of computational resources. This is because: (1) The ViL model is frozen in our method, making its use efficient, and (2) Our ProDe approach does not require a feature bank with periodic updates for deep clustering like SHOT, nor does it involve identifying neighborhoods as in AaD.

Comparison with SFDA methods with ViT backbone. To achieve a comprehensive evaluation, in this part, we present comparisons with typical SFDA methods using ViT backbones (cited from DPC (Zhan et al., 2024)), employing ViT-B/16. Specifically, the comparison methods include SHOT-ViT (Liang et al., 2020), DIPE-ViT (Wang et al., 2022a), DSiT-ViT (Sanyal et al., 2023), AaD-ViT (Yang et al., 2022) and DPC. The results in the Tab. 21 show that ProDe-V16 consistently outperforms DPC in most cases. An exception is that ProDe-V16 is only 0.8% behind on Office-31, which may be attributed to potential overfitting on this relatively small dataset. Notably, even with a ResNet backbone for the target model, ProDe-V16 still surpasses DPC, which utilizes a ViT. Generally, using a ViT for such a small training dataset is unnecessary due to the tendency for overfitting.

Table 21: Comparison with SFDA methods with ViT backbone on closed-set SFDA setting (%). **ViL** means whether using the ViL model.

Method	Venue	**ViL**	**Office-31**	**Office-Home**	**VisDA**	**DomainNet-126**
SHOT-ViT	ICML20	✗	91.4	78.1	–	71.4
DIPE-ViT	CVPR22	✗	90.5	78.2	–	–
DSiT-ViT	ICCV23	✗	93.0	80.5	–	–
AaD-ViT	NeurIPS22	✗	–	–	–	72.7
DPC	IJCAI24	✓	93.3	85.4	–	85.6
ProDe-V16	–	✓	**92.5**	**88.0**	**92.0**	**88.1**

Parameter sensitivity. In this part, we discuss the parameter sensitivity of parameters α, β in L_{ProDe} (see Eq. (6)) and correction strength parameter ω in proxy denoising (see Eq. (5)). All experiments are conducted based on the transfer tasks Cl→Ar in the Office-Home dataset. The varying range are set to $0.5 \leq \alpha \leq 1.4$, $0.1 \leq \beta \leq 1.0$ and $0.5 \leq \omega \leq 1.4$ in 0.1 step size. Fig. 5 (a) depicts the results as α–β vary. When the two parameters changes, there are no evident drops in the accuracy variation curves, except for two boundary situations: (1) $\alpha = 0.5$ and (2) $\beta = 1.0$. The results indicate that ProDe is insensitive to parameters α and β. Meanwhile, when we select parameters, α's value should be lager than β. Besides, in Fig. 5 (b) and (c), we display the results when $\alpha \times \omega$ and $\beta \times \omega$ vary, respectively. Thus, we present the relation between the correction strength and regularization elements in L_{ProDe}. From the two sub-figures, it is seen that the performance has a significant drop as we adopt $\omega = 1.4$. This show that the correction strength in the proxy denoising block should not be too strong.