Progressive Dual-Space Discovering of Unknowns for Source-Free Open-Set Domain Adaptation

Qianshan Zhan[1], Qian Wang[2], and Xiao-Jun Zeng[1] (✉)

[1] the Department of Computer Science, University of Manchester, Manchester, M13 9PL, United Kingdom {qianshan.zhan, x.zeng}@manchester.ac.uk

[2] Luca Healthcare, Shanghai, 200000, China qian.wang173@hotmail.com

Abstract. Open-set domain adaptation (OSDA) transfers knowledge to an unlabeled target domain under both distribution shift and unknown classes absent in the source domain. Most OSDA methods require access to both source and target data and rely on either feature-space or logit-space information for known-unknown separation. However, source data is often restricted due to storage or privacy constraints, and single-space reliance can weaken separation, as unknown samples may be distinguishable in one space but not the other. To address these limitations, we propose Progressive Dual-Space Discovering (PDD), a source-free OSDA method that progressively adapts a pre-trained model for improved domain alignment and known-unknown separation. PDD iteratively builds a credible domain by selecting target samples close to the known-class distribution through dual-space selection: energy-based filtering in logit space followed by prototype-based refinement in feature space. Besides, PDD performs clustering using feature-space information from the credible domain and logit-space information from previously trained models, forming known and unknown domains. With these established domains, cross-entropy loss optimizes learning within the credible domain, while HSIC loss aligns the credible and known domains. Additionally, dual-space uncertainty losses enhance the separation between known and unknown classes. Extensive experiments on three OSDA benchmarks demonstrate the effectiveness of dual-space discovering, known-unknown separation, and progressive updates, facilitating PDD to achieve state-of-the-art performance. Code is available at **https://github.com/qszhan/PDD**.

Keywords: Source-free domain adaptation · Unknown classes · Known-unknown separation · Progressive dual-space discovering.

1 Introduction

Unsupervised domain adaptation (UDA) improves performance in an unlabeled target domain by transferring knowledge from a labeled source domain. A key challenge in UDA is handling distribution shifts, often addressed by domain-invariant features learning [20], [35] or adversarial learning [5], [21]. Despite their advances, these methods assume a closed-set setting [15] with identical label spaces in both domains, limiting applicability when the target domain includes unknown classes.

Open-set domain adaptation (OSDA) [13] addresses this scenario where the target domain includes unknown classes absent from source data. OSDA aims to align data from known classes across domains (domain alignment) and separate data from known and unknown classes (known-unknown separation). Existing OSDA methods employ strategies such as adversarial feature alignment [28], [18], domain similarity measures [12], or learning a shared subspace [32]. Despite their advances, most methods require source data during training, raising storage and privacy concerns, especially in sensitive fields like healthcare [14]. This highlights the need for source-free OSDA [17], [24], which operates without access to source data. However, source-free OSDA is still under-explored, with only a few methods like SHOT [17], AaD [34], USD [9], UPUK [30], and LEAD [24]. Without source data, source-free OSDA faces greater challenges in both domain alignment and known-unknown separation due to absent source features and known classes.

In addition to being source-dependent, current OSDA methods usually discover unknown samples using either **(1)** logit space information, such as the data similarity based on maximum class probabilities [18], prediction variances [28], entropy values [33], [17], confidence scores [10], [2], the Jensen-Shannon distance between logits and pseudo labels [9]; or **(2)** feature space information, including factorized representations [1], distances between source and target features from unknown data [12], a common feature space [32], or decomposed feature components [24]. Relying solely on logit-space or feature-space information is often insufficient for known-unknown separation, as unknown samples may be easier to identify in one space but harder in the other. Virtual-logit Matching (ViM) [31] first introduced the use of dual-space information for Out-of-Distribution (OOD) detection by constructing a virtual logit to represent an OOD class. However, ViM cannot be directly applied to source-free OSDA since it relies on in-distribution data, which is unavailable in source-free settings. Moreover, ViM requires a user-specified threshold for separation, limiting its practicality. Therefore, a new known-unknown separation method is needed by leveraging dual-space information without depending on source data or threshold tuning.

To overcome these limitations, we propose Progressive Dual-space Discovering (PDD), which leverages information from both logit and feature spaces to achieve domain alignment and known-unknown separation in source-free OSDA. Without source feature and known-class information, PDD progressively selects credible target samples that closely align with the known-class distribution. This selection is guided by a progressive dual-space discovering strategy, which integrates energy-based selection in logit space with prototype-based refinement in feature space, utilizing logits and features obtained from previously trained models. Through this iterative process, PDD constructs a series of credible domains with gradually shifting distributions across multiple stages (Fig. 1), progressively enhancing the domain alignment for known classes.

With the constructed credible domain, PDD performs clustering by leveraging both feature-space information from the credible domain and logit-space information from the pre-trained models. This clustering process divides the target samples into known and unknown parts, forming known and unknown domains. Given these established domains, we employ cross-entropy loss to optimize learning within the credible domain and HSIC loss to align the known domain with the credible domain, reinforcing the learning of known classes. Furthermore, dual-space uncertainty losses facilitate effective separation between known and unknown classes. Extensive experiments on three benchmarks demonstrate that our PDD method achieves new state-of-the-art performance. Contributions are summarized below:

(1) We propose Progressive Dual-space Discovering (PDD) for source-free OSDA, leveraging feature and logit space information to enhance domain alignment and known-unknown separation. The framework of PDD is provided in Fig. 2.

(2) To address the absence of source data and the resulting lack of source feature and known-class information, PDD constructs credible domains that are close to the known-class distribution using the previously trained models, progressively enhancing the domain alignment for known classes over multiple stages.

(3) To facilitate known-unknown separation, PDD performs clustering by leveraging both feature-space information from the credible domain and logit-space information from previously trained models.

(4) Extensive experiments on three OSDA benchmarks confirm that PDD achieves state-of-the-art performance, validating the effectiveness of dual-space discovering, known-unknown separation, and progressive updates.

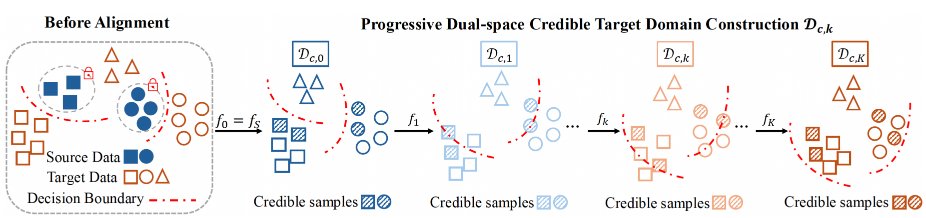

Fig. 1: **Progressive dual-space credible target domain construction**.

2 Method

2.1 Problem formulation

Domains, datasets, and source model. Suppose that there is a source domain $\mathcal{D}_s$ and a target domain $\mathcal{D}_t$. For $\mathcal{D}_s$, the input space and discrete label space are respectively denoted by $\mathcal{X}_s$ and $\mathcal{Y}_s \in \{1, 2, \cdots, L\}$. For $\mathcal{D}_t$, the input space and discrete label space are respectively denoted by $\mathcal{X}_t$ and $\mathcal{Y}_t \in \{1, 2, \cdots, L'\}$. The target dataset for $\mathcal{D}_t$ is denoted by $D_t = \{(\mathbf{x}_i^t, y_i^t)\}_{i=1}^{n_t}$. The source dataset for $\mathcal{D}_s$ is given by $D_s = \{(\mathbf{x}_i^s, y_i^s)\}_{i=1}^{n_s}$. With the source dataset D_s, a source

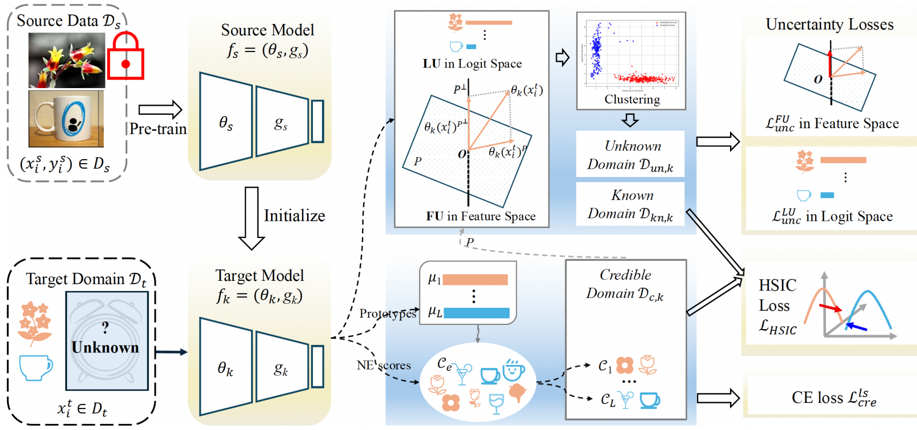

Fig. 2: **PDD framework for constructing target model** f_k. Before constructing f_k, a credible domain $\mathcal{D}_{c,k} \subset \mathcal{D}_t$ is constructed by selecting samples close to known-class distribution. Unknown classes are identified by discovering in both feature and logit spaces, resulting in unknown domain $\mathcal{D}_{un,k}$ and known domain $\mathcal{D}_{kn,k}$ with $\mathcal{D}_{un,k} \cup \mathcal{D}_{kn,k} = \mathcal{D}_t$ and $\mathcal{D}_{un,k} \cap \mathcal{D}_{kn,k} = \varnothing$. For the discovered unknowns in feature space (**FU**), each representation $\theta_k(\mathbf{x}_i^t)$ in $\mathcal{D}_t$ decomposes into $\theta_k(\mathbf{x}_i^t)^P$ (principal space component) and $\theta_k(\mathbf{x}_i^t)^{P^\perp}$ (deviation component). In logit space, the discovered unknowns **LU** are obtained from negative energy scores. Cross-entropy loss $\mathcal{L}_{cre}^{ls}$ trains on credible samples in $\mathcal{D}_{c,k}$. HSIC loss $\mathcal{L}_{HSIC}$ aligns $\mathcal{D}_{kn,k}$ with $\mathcal{D}_{c,k}$ to improve the learning of known classes. Dual-space uncertainty losses $\mathcal{L}_{unc}^{FU}$ and $\mathcal{L}_{unc}^{LU}$ facilitate known-unknown separation.

model $f_s : (\theta_s, g_s)$ is pre-trained with two canonical stages: representation followed by classification. The feature extractor, denoted as $\theta_s : \mathcal{X}_s \to \mathbb{R}^d$, maps the input data to a d-dimensional representation in the feature space $\mathcal{Z}$. This representation is then passed through the classifier $g_s : \mathbb{R}^d \to \mathbb{R}^L$, which transforms the representation into a logit vector $\delta \in \mathbb{R}^L$. The transformation is achieved through a fully connected layer with weight matrix $\mathbf{W} \in \mathbb{R}^{d \times L}$ and bias vector $\mathbf{b} \in \mathbb{R}^L$, formally expressed as $\delta_i = \mathbf{W}^T \mathbf{x}_i^s + \mathbf{b}$. The final soft predictions, yielding the probability distributions $\mathcal{P} = \{\mathbf{p}_i\}_{i=1}^{n_s} = \{[p_{i,1}, p_{i,2}, \ldots, p_{i,L}]\}_{i=1}^{n_s}$, are obtained by applying the softmax function to the logits.

Source-free open-set unsupervised domain adaptation. Due to the source-free and unsupervised constraints, source data D_s and true target labels $\{y_i^t\}_{i=1}^{n_t}$ are unavailable during adaptation. Besides, the source label set $C_s \subset \mathcal{Y}_s$ and target label set $C_t \subset \mathcal{Y}_t$ satisfy $C_s \subset C_t$, with the target-private label set $\overline{C}_t = C_t \backslash C_s$. Our goal is to adapt the source pre-trained model to obtain a target model f_t that can classify target samples into L known classes in C_s and identify them as "unknown" if they belong to $\overline{C}_t$.

Adaptative training process. To obtain the target model $f_t : (\theta_t, g_t)$, we freeze the classifier ($g_t = g_s$) and train only a target-specific feature extractor θ_t

to ensure that target features from known classes align with the source features through same decision boundaries [17]. The adaptation process is divided into K successive stages, initializing θ_0 as the source feature extractor θ_s and performing a training step to obtain f_1. This process continues iteratively, updating model from f_{k-1} at stage $k-1$ to f_k at stage k.

2.2 Progressive credible domain construction

This section progressively constructs a credible domain, $\mathcal{D}_{c,k}$, at each stage k to select target samples that are more likely to belong to, or close to, the distribution captured by the previous model, f_{k-1}, initialized from a source model pre-trained on known-class data. Consequently, the selected credible samples inherently reflect the characteristics of known-class distribution. Through this iterative process, PDD constructs a series of credible domains with gradually shifting distributions across multiple stages (Fig. 1), enhancing domain alignment for known classes. This selection is performed in dual-space: energy-based selection in the logit space, followed by prototype-based refinement in the feature space, which together enhance the reliability of the credible domain.

Energy-based selection in logit space. We begin by operating in logit space, selecting samples with high negative energy (NE) scores to form a group $\mathcal{C}_e$. This method leverages the established correlation between NE scores and sample likelihood within the distribution learned by the previous model f_{k-1} [19]. A higher NE score $NE(\mathbf{x}_i^t)$ indicates that the sample $\mathbf{x}_i^t$ is more likely to belong to or closely match the distribution captured by f_{k-1}. Since f_{k-1} was initialized from a source model pre-trained on known-class data, a high NE score $NE(\mathbf{x}_i^t)$ suggests that $\mathbf{x}_i^t$ is more likely to be within the known-class distribution. Therefore, we use NE scores, $NE(\mathbf{x}_i^t)$, to identify credible samples.

For a target sample $\mathbf{x}_i^t$, the negative energy score, based on the logits δ_i from the model f_k, is computed as $NE(\mathbf{x}_i^t) = \log \sum_l^L e^{\delta_{i,l}}$, where $\delta_{i,l}$ denotes the logit of $\mathbf{x}_i^t$ for class l. A higher $NE(\mathbf{x}_i^t)$ indicates that $\mathbf{x}_i^t$ is more likely to belong to, or closely match known-class distribution and is thus a credible known-class sample, while lower values suggest reduced credibility. Accordingly, the group $\mathcal{C}_e$ is defined by selecting target samples whose negative energy values rank within the top $\sigma_e n_t$ values: $\mathcal{C}_e = \{\mathbf{x}_i^t \mid \mathbf{x}_i^t \in \mathcal{X}_t,\, i \in \mathrm{top}_{\sigma_e n_t}(\{NE(\mathbf{x}_i^t)\}_{i=1}^{n_t})\}$, where $\{NE(\mathbf{x}_i^t)\}_{i=1}^{n_t}$ denotes the set of negative energy scores for all target samples, and σ_e is a scaling parameter.

Prototype-based refinement in feature space. To enhance the credibility of $\mathcal{C}_e$, we further refine $\mathcal{C}_e$ by leveraging geometric prototypes for L classes in feature space. Specifically, the prototype μ_l for class l is computed as $\mu_l = \frac{\sum_{i=1}^{n_t} p_{i,l}\, \theta_k(\mathbf{x}_i^t)}{\sum_{i=1}^{n_t} p_{i,l}}, \quad 1 \leq l \leq L$, where $p_{i,l}$ represents the probability of sample $\mathbf{x}_i^t$ belonging to class l, and $\theta_k(\mathbf{x}_i^t)$ denotes its representation.

For each class l, the distance between the representation $\theta_k(\mathbf{x}_i^t)$ and the prototype μ_l is given by $d(\theta_k(\mathbf{x}_i^t), \mu_l)$, where $d(\cdot,\cdot)$ denotes the cosine distance. Using μ_l, the credible group for class l, i.e., $\mathcal{C}_l$, is defined by selecting samples from $\mathcal{C}_e$ that have the small distances to μ_l:

$$\mathcal{C}_l = \left\{(\mathbf{x}_i^t, l) \mid \mathbf{x}_i^t \in \mathcal{X}_t, d(\theta_k(\mathbf{x}_i^t), \mu_l) \in \text{top}_m \left(\{d(\theta_k(\mathbf{x}_j^t), \mu_l) \mid \mathbf{x}_j^t \in \mathcal{C}_e\}_{i=1}^{n_t}\right)\right\},$$

where l represents the pseudo-label for $\mathbf{x}_i^t$, and top_m denotes the selection of the m samples from $\mathcal{C}_e$ with the smallest distances to μ_l. This iterative selection continues until $\mathcal{C}_l$ reaches m samples. The number of selected target samples m per class l is defined as $m = \lfloor m_0 + (k\eta) \cdot m_0 \rfloor$, where m_0 is the initial number of selected samples, k is the current epoch, and η is an enlarging factor that progressively increases credible domain size.

The final credible domain $\mathcal{D}_{c,k}$ for stage k is obtained by combining the credible groups across all classes: $\mathcal{D}_{c,k} = \bigcup_{l=1}^{L} \mathcal{C}_l = \bigcup_{l=1}^{L} \left\{(\mathbf{x}_i^t, l) \mid \mathbf{x}_i^t \in \mathcal{C}_l\right\}$. For analysis, we denote the credible domain at stage k as $\mathcal{D}_{c,k} = \left\{(\mathbf{x}_i^{t,c}, \tilde{y}_i^t)\right\}_{i=1}^{mL}$.

2.3 Progressive construction of known and unknown domains

Discovered unknowns in feature space. To achieve known-unknown separation, we leverage the principal subspace P, obtained from the eigen-decomposition of representations in credible domain $\mathcal{D}_{c,k}$. Since the credible domain closely reflects the characteristics of known-class data as analyzed in Section 2.2, the principal subspace P captures the core structure of known samples. Consequently, target samples lie within or near P are likely from known classes, while those deviating significantly suggest an unknown distribution. By measuring the distance of sample representations from P, we can distinguish between samples from known and unknown classes.

Inspired by the principal subspace definition in residual score [31], we define the principal subspace P by first offsetting the feature space with the vector $\mathbf{o} = -(\mathbf{W}^T)^{+}\mathbf{b}$, removing the influence the bias term $\mathbf{b}$ on sample positioning. Here $\mathbf{W}$ and $\mathbf{b}$ represent the weight matrix and bias vector from the fully connected layer. The subspace P is then constructed using the adjusted representations $\tilde{\mathbf{Z}}$ from the credible domain $\mathcal{D}_{c,k}$, where each element $\tilde{\mathbf{z}}_i = \theta_k(\mathbf{x}_i^{t,c}) - \mathbf{o}$ denotes the representation of credible sample $\mathbf{x}_i^{t,c}$ in the new coordinate system with origin $\mathbf{o}$. Performing eigen-decomposition on the matrix $\tilde{\mathbf{Z}}^T\tilde{\mathbf{Z}}$ yields $\tilde{\mathbf{Z}}^T\tilde{\mathbf{Z}} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^{-1}$, where the eigenvalues in $\mathbf{\Lambda}$ are sorted decreasingly. The span of the first d' columns of $\mathbf{Q}$ forms the d'-dimensional principal subspace P. In this study, d' set to $d' = \lfloor \frac{d}{2} \rfloor$, capturing the main structures of the credible domain. Let $\mathbf{R} \in \mathbb{R}^{d \times (d-d')}$ be the matrix formed by the $(d'+1)$-th column to the last column of $\mathbf{Q}$. For any target sample $\mathbf{x}_i^t$ with representation $\theta_k(\mathbf{x}_i^t)$, the orthogonal projection of $\theta_k(\mathbf{x}_i^t)$ outside the principal subspace P is given by $\mathbf{R}\mathbf{R}^T\theta_k(\mathbf{x}_i^t)$. The unknowns discovered in feature space is defined as the norm of the component of $\theta_k(\mathbf{x}_i^t)$ that lies outside P, i.e., $fu_i = \sqrt{\theta_k(\mathbf{x}_i^t)\mathbf{R}\mathbf{R}^T\theta_k(\mathbf{x}_i^t)}$. A larger fu_i indicates that $\mathbf{x}_i^t$ is farther away from P, suggesting it may belong to the unknown distribution. For all target samples, the discovered unknowns in feature space are denoted by $\mathbf{FU} = [fu_i, ..., fu_{n_t}]^T \in \mathbb{R}^{n_t}$.

Discovered unknowns in logit space. Given the logit δ from model f_k, the negative energy scores for target domain are represented as $\mathbf{LU} =$

$\left[\mathrm{NE}(\mathbf{x}_1^t), \mathrm{NE}(\mathbf{x}_2^t), \ldots, \mathrm{NE}(\mathbf{x}_{n_t}^t)\right]^T \in \mathbb{R}^{n_t}$, where $NE(\mathbf{x}_i^t) = \log \sum_l^L e^{\delta_{i,l}}$. As discussed on Section 2.2, the negative energy scores is linearly correlated with the likelihood of samples being known. Higher values of $NE(\mathbf{x}_i^t)$ indicate a higher likelihood, making these samples credible for the previous target model f_{k-1} on the previous stage, thus more likely being the data from known classes. Conversely, samples with low negative energy values are less credible for model f_{k-1}, thus more likely being the data from unknown classes.

Discovery of unknowns. The obtained **FU** and **LU** are combined into a clustering feature matrix $\mathbf{U} = [\mathbf{FU}, \mathbf{LU}] \in \mathbb{R}^{n_t \times 2}$. Using $\mathbf{U}$, a clustering algorithm, such as K-means [6] or Gaussian Mixture Model [25], is applied to divide samples into two clusters C_a and C_b, which are assigned as either the known cluster C_{kn} or unknown cluster C_{un}. The unknown cluster C_{un} is identified as the one with the lower mean negative energy score, i.e., $C_{un} = \arg\min_{C \in \{C_a, C_b\}} \left\{ \frac{1}{|C|} \sum_{\mathbf{x}_i^t \in C} NE(\mathbf{x}_i^t) \right\}$, while the known cluster C_{kn} has higher mean negative energy, i.e., $C_{kn} = \arg\max_{C \in \{C_a, C_b\}} \left\{ \frac{1}{|C|} \sum_{\mathbf{x}_i^t \in C} NE(\mathbf{x}_i^t) \right\}$.

The pseudo labels for samples from unknown classes can be assigned with $\tilde{y}_i^t = L + 1$ as they are considered belonging to the class $L + 1$. For identified samples from known classes, it can be further specified using the distance of each known sample to the class prototypes μ_l, which are

$$\tilde{y}_i^t = \begin{cases} L + 1, & \text{if } \mathbf{x}_i^t \in C_{un} \text{ (unknown)}, \\ \arg\min_l d(\theta_k(\mathbf{x}_i^t), \mu_l), & \text{if } \mathbf{x}_i^t \in C_{kn} \text{ (known)}. \end{cases}$$

This results in the unknown domain $\mathcal{D}_{un,k}$ for stage k defined as $\mathcal{D}_{un,k} = \{(\mathbf{x}_i^t, L+1) \mid \mathbf{x}_i^t \in C_{un}\}$, and the known domain $\mathcal{D}_{kn,k}$ as $\mathcal{D}_{kn,k} = \{(\mathbf{x}_i^t, \arg\min_l d(\theta_k(\mathbf{x}_i^t), \mu_l)) \mid \mathbf{x}_i^t \in C_{kn}\}$. The designed clustering for known-unknown separation improves reliability by combining feature and logit space insights. When a sample is identified as unknown in both spaces, it strengthens our confidence in classifying it as unknown. If it aligns in only one space, the dual-space method captures this inconsistency, reducing errors. Further analysis of the separation mechanism is in Section 3.3. For analysis, we denote the known domain $\mathcal{D}_{kn,k}$ at stage k as $\mathcal{D}_{kn,k} = \{(\mathbf{x}_i^t, \tilde{y}_i^t)\}_{i=1}^{m_{kn}} = \{(\mathbf{x}_i^t, \arg\min_l d(\theta_k(\mathbf{x}_i^t), \mu_l))\}_{i=1}^{m_{kn}}$, unknown domain $\mathcal{D}_{un,k}$ at stage k as $\mathcal{D}_{un,k} = \{(\mathbf{x}_i^t, \tilde{y}_i^t)\}_{i=1}^{n_t - m_{kn}} = \{(\mathbf{x}_i^t, L+1)\}_{i=1}^{n_t - m_{kn}}$, where m_{kn} is the number of known class samples selected from clustering.

2.4 Target domain adaptation

This section introduce the designed loss function to progressively update the model f_k for target domain adaptation.

Credible domain learning. To ensure effective learning on the credible domain $\mathcal{D}_{c,k}$, a cross-entropy loss with label smoothing is employed [22]: $\mathcal{L}_{cre}^{ls}(f_k) = -\mathbb{E}_{(\mathbf{x}_i^t, \tilde{y}_i^t) \in \mathcal{D}_{c,k}} \sum_{l=1}^{L} \delta_l^t \log f_k(\mathbf{x}_t)$, where $\delta_l^t = (1 - \alpha) q_l^t + \alpha / L$, with q^t as the one-hot encoding of the pseudo label $\tilde{y}_i^t$ and $\alpha = 0.1$ as the smoothing parameter. Here, $f_k(\mathbf{x}_i^t)$ is the softmax logit of model f_k.

Dual-space learning of unknowns. With the defined known domain $\mathcal{D}_{kn,k}$ and unknown domain $\mathcal{D}_{un,k}$ at stage k, we define an dual-space uncertainty losses in both feature and logit spaces to enhance the differentiation of the model between the samples from known and unknown classes. Specifically, the uncertainty loss in feature space is defined as

$$\mathcal{L}_{unc}^{FU}(f_k) = \exp\left[\mathbb{E}_{(\mathbf{x}_i^t, \tilde{y}_i^t)\in\mathcal{D}_{kn,k}} fu_i\right] - \exp\left[\mathbb{E}_{(\mathbf{x}_i^t, \tilde{y}_i^t)\in\mathcal{D}_{un,k}} fu_i\right], \tag{1}$$

where fu_i, calculated using Sec. (2.3), represents the discovered unknowns in feature space of each target sample $\mathbf{x}_i^t$. This loss encourages smaller values of fu_i for samples from known classes and larger values for samples from unknown classes, thereby reinforcing separation in the feature space. Similarly, the uncertainty loss in logit space is defined as

$$\begin{aligned}\mathcal{L}_{unc}^{LU}(f_k) = &\exp\left[\mathbb{E}_{(\mathbf{x}_i^t, \tilde{y}_i^t)\in\mathcal{D}_{kn,k}} \frac{-\sum_{i=1}^{L} p_{i,l}\log p_{i,l}}{\log L}\right] \\ &- \exp\left[\mathbb{E}_{(\mathbf{x}_i^t, \tilde{y}_i^t)\in\mathcal{D}_{un,k}} \frac{-\beta\sum_{i=1}^{L} p_{i,l}\log p_{i,l}}{\log L}\right],\end{aligned} \tag{2}$$

where $p_{i,l}$ represents the softmax probability of class l for sample $\mathbf{x}_i^t$. The scaling parameter β adjusts the emphasis on minimizing uncertainty for unknown samples. A larger β encourages faster learning for classifying uncertain samples. This loss facilitates lower entropy for samples from known classes and higher entropy for those from unknown classes, reinforcing separation in the logit space.

Domain alignment for knowns. As analyzed in Section 2.2, the distribution within credible domain $\mathcal{D}_{c,k}$ is close to the known-class distribution. Therefore, aligning $\mathcal{D}_{c,k}$ with the known domain $\mathcal{D}_{kn,k}$ facilitates the adaptation of samples from known classes. To achieve this alignment, we use the Hilbert-Schmidt Independence Criterion (HSIC), a kernel-based method that measures statistical dependence between distributions without requiring density estimation [4]. By capturing dependencies between credible and known domains, HSIC supports effective domain adaptation for samples from known classes.

To calculate the HSIC value, each class prototype $\tilde{\mu}_l$ within the credible domain $\mathcal{D}_{c,k}$ is refined as $\tilde{\mu}_l = \frac{\sum_{i=1}^{mL} p_{i,l}\,\theta_k(\mathbf{x}_i^t)}{\sum_{i=1}^{mL} p_{i,l}}$, $\quad 1 \le l \le L, \mathbf{x}_i^t \in \mathcal{D}_{c,k}$, where $p_{i,l}$ represents the probability for sample $\mathbf{x}_i^t$ belonging to class l. Using these refined class prototypes, each target sample $\mathbf{x}_i^t$ in the known domain is assigned a prototype $\tilde{\mu}_{l_i}$ based on the nearest distance: $l_i = \arg\min_l d(\theta_k(\mathbf{x}_i^t), \tilde{\mu}_l)$. Let $\mathbf{N} = \{\theta_k(\mathbf{x}_i^t)\}_{i=1}^{m_{kn}}$ represent the target representations in the known domain, and $\mathbf{M} = \{\tilde{\mu}_{l_i}\}_{i=1}^{m_{kn}}$ represent the corresponding prototypes from the credible domain. Using $\mathbf{N}$ and $\mathbf{M}$, we compute the HSIC value to measure dependence between the two distributions and define the HSIC loss for domain alignment as: $\mathcal{L}_{HSIC} = \mathrm{HSIC}(\mathbf{N}, \mathbf{M})$. Due to space constraints, the detailed calculation of $\mathrm{HSIC}(\mathbf{N}, \mathbf{M})$ is provided in the Supplementary Material.

2.5 Overall objective

Combining the designed losses, the overall objective function is formulated as

$$\mathcal{L}_{obj} = \mathcal{L}_{cre}^{ls} + \mathcal{L}_{HSIC} + \lambda_1 \mathcal{L}_{unc}^{FU} + \lambda_2 \mathcal{L}_{unc}^{LU}, \tag{3}$$

where λ_1 and λ_2 are regularization parameters.The pseudo code of PDD is presented in the Supplementary Material.

3 Experiments

3.1 Setup

Datasets. PDD is evaluated on three image classification benchmarks: Office-31, Office-Home, and Digits. (1) *Office-31*[26] includes three object domains: AMAZON (A), DSLR (D), and WEBCAM (W), each with 31 classes. The first 10 classes (alphabetically) are used as known classes, and the others as unknown, forming 6 OSDA tasks: A $\rightarrow$ D, $\cdots$, W$\rightarrow$D. (2) *Office-Home* [29] includes four domains: Artistic (Ar), Clipart (Cl), Product (Pr), and Real-World (R), each with 65 classes. Classes 1–25 are labeled as known and 26–65 as unknown, creating 12 OSDA tasks: Ar$\rightarrow$Cl, $\cdots$, R$\rightarrow$Pr. (3) *Digits* [8] include three datasets: MNIST (M), SVHN (S), and USPS (U). Classes 0–4 are labeled as known, and classes 5–9 as unknown, forming 3 OSDA tasks: S$\rightarrow$M, $\cdots$, U$\rightarrow$M.
Baseline. This study compares the performance of our PDD with the standard OSDA methods such as OSBP [28], STA [18], ROS [2], OSLPP [32], DANN [5], ANNA [15], BCL [7], and source-free OSDA methods such as SHOT [17], AaD [34], USD [9], UPUK [30], and LEAD [24].
Evaluation Metrics. Three widely-used metrics are adopted for evaluation [28], [18]: (1) **OS***: normalized accuracy for the known classes only; (2) **UNK**: accuracy of the unknown class; and (3) **HOS**: harmonic mean accuracy of OS* and UNK. Detailed calculations of these metrics and implementation details are provided in the Supplementary Material.

3.2 Main Results

Tables 1, 2, and 3 present HOS score comparisons on Office-31, Office-Home, and Digits datasets. ✓ denotes source-free OSDA methods, while ✗ denotes standard OSDA methods. The best performance among source-free methods is in bold blue. While standard OSDA methods often outperform source-free methods due to direct access to source data, PDD achieves HOS scores competitive with top-performing standard OSDA methods on tasks such as A $\rightarrow$ D in Office-31, Ar $\rightarrow$ Pr in Office-Home, and U $\rightarrow$ M in Digits. Among source-free OSDA methods, PDD consistently outperforms all other methods across every task in Office-Home and most tasks in Office-31 and Digits. Notably, on tasks like A $\rightarrow$ D in Office-31 and R $\rightarrow$ Ar in Office-Home, PDD shows marked improvements, validating the effectiveness of PDD in source-free OSDA.

Table 1: HOS (%) on 6 tasks from Office-31. Additional results for OS* and UNK metrics are available in Supplementary Material.

Method	SF?	A-D	A-W	D-A	D-W	W-A	W-D	AVG
OSBP [28]	✗	82.4	82.7	75.1	97.2	73.7	91.1	83.7
STA [18]	✗	75	75.9	73.2	69.8	66.1	75.2	72.6
ROS [2]	✗	82.4	82.1	77.9	96	77.2	**99.7**	85.9
DANCE [27]	✗	66.9	70.7	80	84.8	65.8	70.2	73.1
cUADAL [11]	✗	90.1	87.9	**98.2**	99.4	80.5	75.1	88.5
OSLPP [32]	✗	91.5	**89**	79.3	92.3	78.7	93.6	87.4
ANNA [15]	✗	83.8	85.5	82.5	**99.5**	81.6	98.4	88.6
BCL [7]	✗	**92.4**	88.3	84.8	**99.5**	**86.9**	**99.7**	**92.1**
SHOT [17]	✓	62	58.7	53.3	86.1	59.6	82.1	67
AaD [34]	✓	78.3	74.3	74.2	87	73	95.7	80.4
USD [9]	✓	81.2	77.9	73.9	97.3	74	95.2	83.3
UPUK [30]	✓	76.7	80.0	75.5	83.2	78.2	84.3	79.7
LEAD [24]	✓	84.9	85.1	**90.2**	94.8	**90.3**	96.5	90.3
PDD	✓	**93.4**	**86.5**	84	**97.8**	81.5	**99.6**	**90.5**

Table 2: HOS (%) on 12 tasks from Office-Home. Additional results for OS* and UNK metrics are available in Supplementary Material.

Method	SF?	Ar-Cl	Ar-Pr	Ar-R	Cl-Ar	Cl-Pr	Cl-R	Pr-Ar	Pr-Cl	Pr-R	R-Ar	R-Cl	R-Pr	AVG
OSBP[28]	✗	55.1	65.2	72.9	64.3	64.7	70.6	63.2	53.2	73.9	66.7	54.5	72.3	64.7
STA[18]	✗	56.3	63.7	62.1	57.9	62.5	66.3	61.9	53.2	69.5	67.1	54.5	64.5	61.1
DAOD[3]	✗	60.5	56.6	69.5	60.4	60.4	65.8	59.1	49.4	62.5	52.5	45.5	49.1	57.6
DANCE[27]	✗	53.1	49.8	39.4	40.9	45.9	30.2	54.2	55.7	41.2	27.5	48.3	44	44.2
ROS[2]	✗	60.1	69.3	76.5	58.9	65.2	68.6	60.6	56.3	74.4	68.8	60.4	75.7	66.2
cUADAL[11]	✗	63.6	71.6	77.5	**65**	68.3	72.6	62.9	54.6	76.8	**72.6**	59.9	76.7	68.5
OSLPP[32]	✗	61	72.8	74.3	60.9	66.9	70.4	63.6	59.3	74	67.2	59	74.4	67
ANNA[15]	✗	**69**	73.7	76.8	64.7	68.6	73	**66.5**	**63.1**	76.6	71.3	**65.7**	78.7	70.7
BCL[7]	✗	64.3	**75.4**	**79.0**	63.1	**70.0**	**73.4**	66.2	62.5	**77.3**	69.7	64.7	**82.1**	**70.8**
SHOT[17]	✓	39.5	39.8	47	54.6	40.2	39.1	57.7	40.8	46.2	59.9	40.1	42.3	45.6
AaD[34]	✓	57.6	66.9	69.9	60.5	61.4	67.8	60.1	55.9	70.6	64.6	57.5	70.1	63.6
USD[9]	✓	61.1	70	76.3	60.1	65.2	68.9	62.6	56.3	72.2	67.8	59.1	71.1	65.9
UPUK[30]	✓	55.8	76.7	78.4	66.4	73.1	77.6	67.6	55.1	78.6	67.8	59.4	74.4	69.2
LEAD[24]	✓	60.7	70.8	76.5	61.0	68.6	70.8	65.5	59.8	74.2	64.8	57.7	75.8	67.2
PDD	✓	**67.3**	**79.5**	**82.1**	**67.3**	**76.5**	**77.8**	**71.5**	**59.9**	**81.0**	**75.6**	**64.2**	**80.4**	**73.6**

3.3 Further Analyses

(1) Mechanism of PDD for Known-Unknown Separation. PDD clusters known and unknown samples by discovering unknowns in both feature (FU) space and logit (LU) space, as illustrated in Fig. 3 for the W $\rightarrow$ D task on Office-31. Initially (Fig. 3a), there is obvious overlap between known and unknown

Table 3: HOS (%) on tasks from Digits. Additional results for OS* and UNK metrics are available in Supplementary Material.

SF?	OSVM	DANN	ATI-λ	OSBP	STA	KASE	BADA	PDD
	[10] ✗	[5] ✗	[23] ✗	[28] ✗	[18] ✗	[16] ✗	[36] ✗	✓
S-M	18	65.5	<u>69.6</u>	58.4	63.7	65.3	**83.5**	64.9
U-M	48.5	89.8	84	**95.8**	95.1	93.9	86.3	**95.8**
M-U	83	86.8	80.4	<u>91.7</u>	91.6	<u>91.7</u>	90.8	**92.8**
AVG	61.4	80.8	78.4	82	83.6	83.7	<u>86.9</u>	**94.3**

samples, hindering separation accuracy. As training progresses, PDD increases LU and reduces FU for known samples while doing the opposite for unknowns. By Epoch 6 (Fig. 3b), this strategy reduces overlap, clustering known samples (high LU, low FU) and unknown samples (low LU, high FU) more effectively. By Epoch 15 (Fig. 3c), clusters for known and unknown samples are well-established, achieving clear separation in both spaces. This progressive dual-space adaptation enables reliable identification of known and unknown samples.

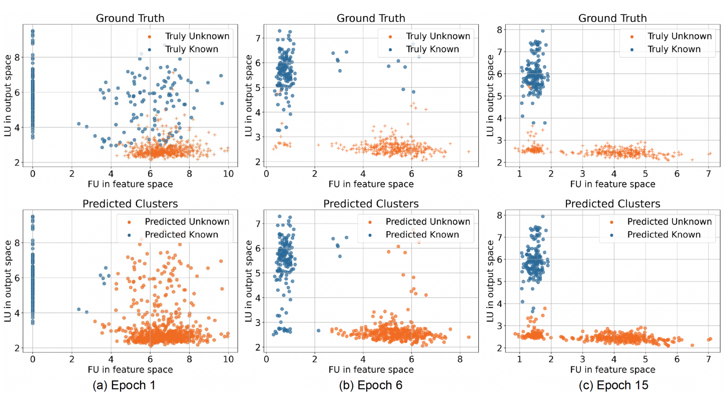

Fig. 3: Progressive clustering of known and unknown samples in dual spaces (feature and logit) across training epochs.

(2) Ablation on Dual-Space Discovering and Progressive Updates. We evaluate PDD against three variants on HOS scores in the Office-31 dataset, as shown in Fig. 4. (1) **PDD w/o P**. This variant skips progressive construction of credible domain $\mathcal{D}_{c,k}$, known domain $\mathcal{D}_{kn,k}$, and unknown domain $\mathcal{D}_{un,k}$ at each epoch. PDD significantly outperforms PDD w/o P, indicating the importance of updating these domains progressively for domain alignment and known/unknown separation. (2) **PDD w/o OSD** and **PDD w/o FSD**: PDD w/o OSD excludes logit space discovering when constructing known/unknown domains and the logit space uncertainty loss $\mathcal{L}_{unc}^{LU}$, while PDD w/o FSD removes feature space discovering and feature space uncertainty loss $\mathcal{L}_{unc}^{FU}$. PDD demonstrates superior performance over both variants, confirming the effectiveness of dual-space discovering and adaptation for accurately identifying unknowns.

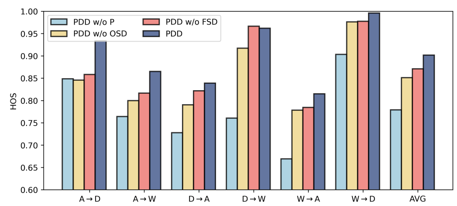

Fig. 4: HOS comparison of PDD and its three variants on Office-31.

(3) Ablation Analysis on Loss Component Contributions. We evaluate PDD with different combinations of loss components. Table 4 presents the HOS scores on the Office-31 dataset for various configurations. The results in the last row show that PDD incorporating all four loss components yields the highest overall HOS scores across most tasks. This highlights the complementary roles of each component: $\mathcal{L}_{cre}^{ls}$ improves credible data learning, $\mathcal{L}_{HSIC}$ enhances domain alignment, and $\mathcal{L}_{unc}^{LU}$ and $\mathcal{L}_{unc}^{FU}$ facilitate effective known-unknown separation.

Table 4: HOS (%) from ablation analysis of individual loss components on various tasks in the Office-31 dataset.

	A→D	A→W	D→A	D→W	W→A	W→D
PDD w/o $\mathcal{L}_{cre}^{ls}$	89.9	84.8	**84.3**	95.7	80.9	95.2
PDD w/o $\mathcal{L}_{HSIC}$	92.6	84.8	<u>84.1</u>	96.1	80.6	96.6
PDD w/o $\mathcal{L}_{unc}^{FU}$	91	85.6	83.5	93.7	81.4	95
PDD w/o $\mathcal{L}_{unc}^{LU}$	91.6	74.5	74.5	72.4	74.2	95.6
PDD	**93.4**	**86.5**	84	**97.8**	**81.5**	**99.6**

(4) HOS during Training. As shown in Fig. 5, we present HOS curves of AaD, SHOT, and PDD on Office-31 tasks D $\rightarrow$ A, D $\rightarrow$ W, as well as Office-Home tasks Ar $\rightarrow$ Cl and Ar $\rightarrow$ Pr. HOS scores for AaD and SHOT either remain stable or decline during training. This is because SHOT, designed for closed-set settings, lacks mechanisms to explicitly separate unknown samples from known ones in OSDA. In contrast, our method consistently achieves the highest HOS scores and steadily improves over epochs, validating its training stability.

(5) Impact of Parameters σ_e, m_0, and η. Fig. 6 analyzes the impact of σ_e, m_0, and η on HOS scores, which control the sample numbers in credible domain. For m_0 (first subplot), a lower value like 5 results in fewer samples selected, restricting learning due to limited data. Conversely, a higher m_0 (e.g.,

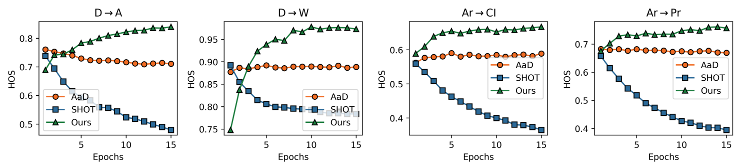

Fig. 5: HOS variation curve for methods AaD, SHOT, and PDD during training on Office-31 and Office-Home tasks.

20) allows more samples but introduces noise from non-credible data, reducing HOS stability. Moderate values like 10 or 15 achieve a balance, selecting enough samples and yielding high, stable HOS scores. This trade-off also applies to η and σ_e: larger values enhance learning by increasing sample count but reduce credibility, while smaller values improve credibility but limit learning data. Therefore, optimal and stable performance is achieved with moderate parameter settings: m_0 between 10 and 15, σ_e between 0.4 and 0.8, and η between 0.1 and 0.5. These ranges provide a balanced trade-off, maximizing learning potential while preserving sample credibility.

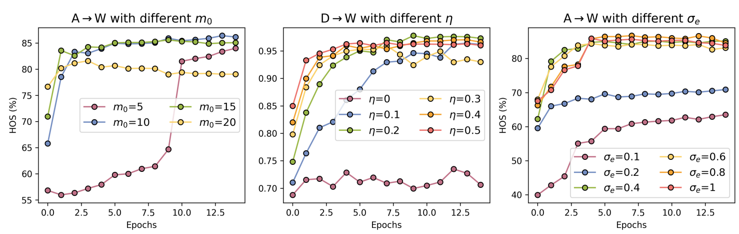

Fig. 6: Effect of σ_e, m_0, and η on HOS in Office-31 dataset.

(6) Sensitivity to Loss Coefficients β, λ_1, and λ_2. We evaluate the sensitivity of PDD to variations in loss coefficients β, λ_1, and λ_2, which influence uncertainty losses across feature and logit spaces. Fig. 7 shows that HOS remains stable across varying λ_1 and λ_2 values (first and second plot), indicating robustness to these parameters. Although minor variations may occur on specific tasks, the overall impact across multiple tasks remains minimal. For β (third plot), a higher value (e.g., 3) overemphasizes uncertainty, reducing HOS and affecting performance, while $\beta = 0$ neglects logit space uncertainty, leading to suboptimal performance. Moderate values (1 or 2) provide better balance, effectively managing uncertainty without overfitting.

(7) Impact of Openness. To verify the robustness of PDD across varying openness levels, we conducted experiments on Office-31 dataset with openness O set to $\{0.25, 0.5, 0.75, 0.93\}$, where $O = 1 - \frac{|C_s|}{|C_t|}$, following [18]. Higher O values indicate greater challenges due to more unknown classes in target domain.

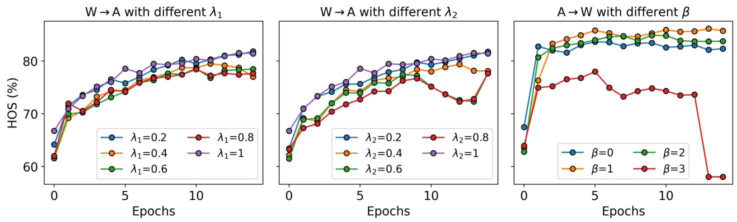

Fig. 7: HOS variation on Tasks W $\rightarrow$ A and A $\rightarrow$ W in the Office-31 dataset as the λ_1, λ_2 and β varying.

Fig. 8 presents HOS performance for PDD, AaD, and SHOT on tasks A$\rightarrow$W and W$\rightarrow$D. While AaD and SHOT degrade as openness increases, likely due to reliance on source-like samples, PDD remains stable or improves, consistently outperforming both methods across all openness levels. This stability is due to its dual-space discovering of unknowns, improving known-unknown separation. Besides, as openness increases and known classes decrease, the classification of known data is simplified, further supports the domain alignment. All these highlights its effectiveness across varying openness levels in OSDA.

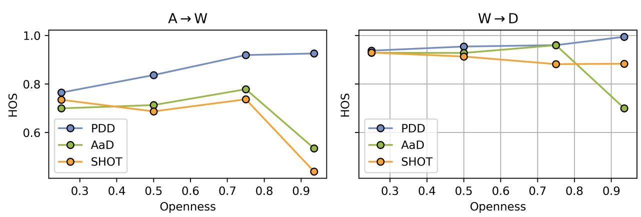

Fig. 8: HOS performance comparison of PDD, AaD, and SHOT on Office-31 dataset across different openness levels.

(8) Feature Visualization. We visualize the last-layer features of ResNet-50, AaD, SHOT, and PDD on W $\rightarrow$ D task using t-SNE embeddings, as shown in Fig. 9. For ResNet-50, AaD, and SHOT methods, we observe overlapping features between known and unknown classes, along with the misalignment between source and target features within the same known classes. In contrast, PDD achieves an obvious separation between known and unknown classes. Besides, within each known class, PDD closely aligns source and target samples, demonstrating effective domain alignment. These results indicate that PDD successfully identifies unknown samples and achieves consistent alignment of source and target features for known classes, validating its effectiveness.

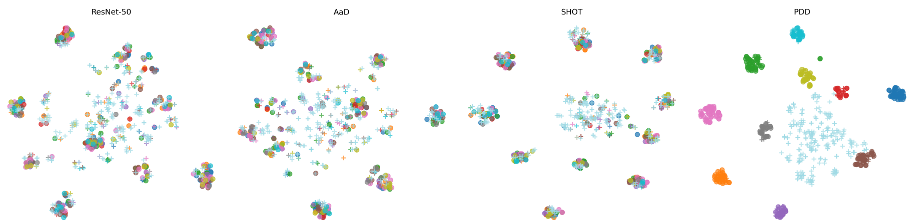

Fig. 9: Visualization of the features extracted by ResNet-50, AaD, SHOT, and PDD on Office-31 task W $\rightarrow$ D using t-SNE embeddings, respectively. $\circ$ and $+$ represent the source and target data. Different classes are distinguished by unique colors, with unknown samples marked in light blue.

4 Conclusion

This study introduces PDD, a source-free OSDA method that leverages dual-space, i.e., both feature and logit spaces information to improve alignment and known-unknown separation. By progressively constructing a credible domain through dual-space selection and establishing known and unknown domains via dual-space discovering, PDD achieves effective alignment for known classes across domains and known-unknown separation. Extensive experiments validate the effectiveness of our dual-space consideration and progressive updates, with PDD achieving state-of-the-art performance across three OSDA benchmarks.

References

1. Baktashmotlagh, M., Faraki, M., Drummond, T., Salzmann, M.: Learning factorized representations for open-set domain adaptation. In: ICLR (2019)
2. Bucci, S., Loghmani, M.R., Tommasi, T.: On the effectiveness of image rotation for open set domain adaptation. In: ECCV. pp. 422–438. Springer (2020)
3. Fang, Z., Lu, J., Liu, F., Xuan, J., Zhang, G.: Open set domain adaptation: Theoretical bound and algorithm. IEEE TNNLS **32**(10), 4309–4322 (2020)
4. Fukumizu, K., Gretton, A., Sun, X., Schölkopf, B.: Kernel measures of conditional dependence. NIPS **20** (2007)
5. Ganin, Y., Lempitsky, V.: Unsupervised domain adaptation by backpropagation. In: International conference on machine learning. pp. 1180–1189. PMLR (2015)
6. Hartigan, J.A., Wong, M.A.: Algorithm as 136: A k-means clustering algorithm. Journal of the Royal Statistical Society: Series C **28**(1), 100–108 (1979)
7. Huang, Z.X., Ren, C.X.: Rethinking correlation learning via label prior for open set domain adaptation. In: IJCAI (2024)
8. Hull, J.J.: A database for handwritten text recognition research. IEEE TPAMI **16**(5), 550–554 (1994)
9. Jahan, C.S., Savakis, A.: Unknown sample discovery for source free open set domain adaptation. In: CVPR. pp. 1067–1076 (2024)
10. Jain, L.P., Scheirer, W.J., Boult, T.E.: Multi-class open set recognition using probability of inclusion. In: ECCV. pp. 393–409. Springer (2014)

11. Jang, J., Na, B., Shin, D.H., Ji, M., Song, K., Moon, I.C.: Unknown-aware domain adversarial learning for open-set domain adaptation. NIPS **35**, 16755–16767 (2022)
12. Jing, M., Li, J., Zhu, L., Ding, Z., Lu, K., Yang, Y.: Balanced open set domain adaptation via centroid alignment. In: AAAI. vol. 35, pp. 8013–8020 (2021)
13. Jing, T., Liu, H., Ding, Z.: Towards novel target discovery through open-set domain adaptation. In: ICCV. pp. 9322–9331 (2021)
14. Kim, Y., Cho, D., Han, K., Panda, P., Hong, S.: Domain adaptation without source data. IEEE TAI **2**(6), 508–518 (2021)
15. Li, W., Liu, J., Han, B., Yuan, Y.: Adjustment and alignment for unbiased open set domain adaptation. In: CVPR. pp. 24110–24119 (2023)
16. Lian, Q., Li, W., Chen, L., Duan, L.: Known-class aware self-ensemble for open set domain adaptation. arXiv preprint arXiv:1905.01068 (2019)
17. Liang, J., Hu, D., Feng, J.: Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In: ICML. pp. 6028–6039. PMLR (2020)
18. Liu, H., Cao, Z., Long, M., Wang, J., Yang, Q.: Separate to adapt: Open set domain adaptation via progressive separation. In: CVPR. pp. 2927–2936 (2019)
19. Liu, W., Wang, X., Owens, J., Li, Y.: Energy-based out-of-distribution detection. NIPS **33**, 21464–21475 (2020)
20. Long, M., Cao, Y., Wang, J., Jordan, M.: Learning transferable features with deep adaptation networks. In: ICML. pp. 97–105. PMLR (2015)
21. Mei, Z., Ye, P., Ye, H., Li, B., Guo, J., Chen, T., Ouyang, W.: Automatic loss function search for adversarial unsupervised domain adaptation. IEEE TCSVT **33**(10), 5868–5881 (2023)
22. Müller, R., Kornblith, S., Hinton, G.E.: When does label smoothing help? NIPS **32** (2019)
23. Panareda Busto, P., Gall, J.: Open set domain adaptation. In: ICCV. pp. 754–763 (2017)
24. Qu, S., Zou, T., He, L., Röhrbein, F., Knoll, A., Chen, G., Jiang, C.: Lead: Learning decomposition for source-free universal domain adaptation. In: CVPR. pp. 23334–23343 (2024)
25. Reynolds, D.A.: Gaussian mixture models. In: Encyclopedia of Biometrics, pp. 741–745. Springer, Boston, MA (2009)
26. Saenko, K., Kulis, B., Fritz, M., Darrell, T.: Adapting visual category models to new domains. In: ECCV. pp. 213–226 (2010)
27. Saito, K., Kim, D., Sclaroff, S., Saenko, K.: Universal domain adaptation through self supervision. NIPS **33**, 16282–16292 (2020)
28. Saito, K., Yamamoto, S., Ushiku, Y., Harada, T.: Open set domain adaptation by backpropagation. In: ECCV. pp. 153–168 (2018)
29. Venkateswara, H., Eusebio, J., Chakraborty, S., Panchanathan, S.: Deep hashing network for unsupervised domain adaptation. In: CVPR. pp. 5018–5027 (2017)
30. Wan, F., Zhao, H., Yang, X., Deng, C.: Unveiling the unknown: Unleashing the power of unknown to known in open-set source-free domain adaptation. In: CVPR. pp. 24015–24024 (2024)
31. Wang, H., Li, Z., Feng, L., Zhang, W.: Vim: Out-of-distribution with virtual-logit matching. In: CVPR (2022)
32. Wang, Q., Meng, F., Breckon, T.P.: Progressively select and reject pseudo-labelled samples for open-set domain adaptation. IEEE TAI (2024)
33. Wang, Y., Zhu, R., Ji, P., Li, S.: Open-set graph domain adaptation via separate domain alignment. In: AAAI. vol. 38, pp. 9142–9150 (2024)

34. Yang, S., Jui, S., van de Weijer, J., et al.: Attracting and dispersing: A simple approach for source-free domain adaptation. NIPS **35**, 5802–5815 (2022)
35. Zhan, Q., Zeng, X.J., Wang, Q.: Reducing bias in source-free unsupervised domain adaptation for regression. Neural Networks p. 107161 (2025)
36. Zhong, L., Fang, Z., Liu, F., Yuan, B., Zhang, G., Lu, J.: Bridging the theoretical bound and deep algorithms for open set domain adaptation. IEEE TNNLS **34**(8), 3859–3873 (2021)

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.