Cross-Domain Attribute Alignment with CLIP: A Rehearsal-Free Approach for Class-Incremental Unsupervised Domain Adaptation

Kerun Mi
School of Computer Science and Engineering, Nanjing University of Science and Technology
Nanjing, China
kerun_mi@njust.edu.cn

Guoliang Kang*
School of Automation Science and Electrical Engineering, Beihang University
Beijing, China
kgl.prml@gmail.com

Guangyu Li
School of Computer Science and Engineering, Nanjing University of Science and Technology
Nanjing, China
guangyu.li2017@njust.edu.cn

Lin Zhao
School of Computer Science and Engineering, Nanjing University of Science and Technology
Nanjing, China
linzhao@njust.edu.cn

Tao Zhou
School of Computer Science and Engineering, Nanjing University of Science and Technology
Nanjing, China
taozhou.ai@gmail.com

Chen Gong*
School of Automation and Intelligent Sensing, Shanghai Jiao Tong University
Shanghai, China
chen.gong@sjtu.edu.cn

Abstract

Class-Incremental Unsupervised Domain Adaptation (CI-UDA) aims to adapt a model from a labeled source domain to an unlabeled target domain, where the sets of potential target classes appearing at different time steps are disjoint and are subsets of the source classes. The key to solving this problem lies in avoiding catastrophic forgetting of knowledge about previous target classes during continuously mitigating the domain shift. Most previous works cumbersomely combine two technical components. On one hand, they need to store and utilize rehearsal target sample from previous time steps to avoid catastrophic forgetting; on the other hand, they perform alignment only between classes shared across domains at each time step. Consequently, the memory will continuously increase and the asymmetric alignment may inevitably result in knowledge forgetting. In this paper, we propose to mine and preserve domain-invariant and class-agnostic knowledge to facilitate the CI-UDA task. Specifically, via using CLIP, we extract the class-agnostic properties which we name as "attribute". In our framework, we learn a "key-value" pair to represent an attribute, where the key corresponds to the visual prototype and the value is the textual prompt. We maintain two attribute dictionaries, each corresponding to a different domain. Then we perform attribute alignment across domains to mitigate the domain shift, via encouraging visual attention consistency and prediction consistency. Through attribute modeling and cross-domain alignment, we effectively reduce catastrophic knowledge forgetting while mitigating the domain shift, in a rehearsal-free way. Experiments on three CI-UDA benchmarks demonstrate that our method outperforms previous state-of-the-art methods and effectively alleviates catastrophic forgetting. Code is available at https://github.com/RyunMi/VisTA.

CCS Concepts

• **Computing methodologies** → **Transfer learning**; *Lifelong machine learning*.

Keywords

Unsupervised Domain Adaptation, Class-Incremental Learning, CLIP.

ACM Reference Format:
Kerun Mi, Guoliang Kang, Guangyu Li, Lin Zhao, Tao Zhou, and Chen Gong. 2025. Cross-Domain Attribute Alignment with CLIP: A Rehearsal-Free Approach for Class-Incremental Unsupervised Domain Adaptation. In *Proceedings of the 33rd ACM International Conference on Multimedia (MM '25), October 27–31, 2025, Dublin, Ireland*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3746027.3755184

1 Introduction

Class-Incremental Learning (CIL) [1, 18, 47] aims to handle sequentially arriving tasks, where at each time step new classes emerge. The model needs to classify all seen classes during testing without access to the task ID. CIL methods generally rely on labeled data, which is often limited due to the high cost of data annotation in real-world scenarios [28, 30, 44, 45]. A feasible approach is to leverage an off-the-shelf labeled dataset (*i.e.,* source domain) to transfer a model to a class-incremental unlabeled dataset (*i.e.,* target domain), with the source domain containing all classes.

However, the distribution shift between domains poses significant challenges to the transferability of a model. Conventional unsupervised domain adaptation (UDA) or partial domain adaptation (PDA) methods can be utilized to mitigate the distribution shift

*Corresponding authors.

MM '25, October 27–31, 2025, Dublin, Ireland.

ACM ISBN 979-8-4007-2035-2/2025/10
https://doi.org/10.1145/3746027.3755184

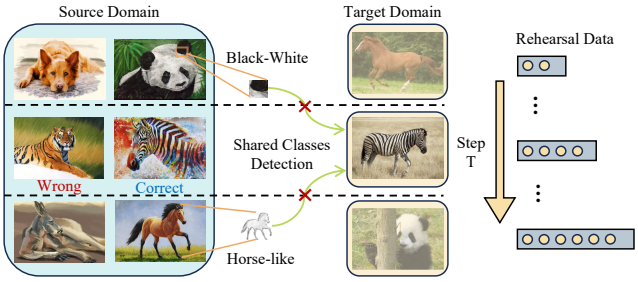


Figure 1: Existing CI-UDA methods retain knowledge by storing rehearsal target data (yellow circles), which will introduce additional computational overhead. These methods also detect the shared classes (*e.g.,* "zebra") between domains at each time step, which may result in errors (*e.g.,* "tiger") and ignore valuable attributes of source-private classes (*e.g.,* "Black-White" of "panda" and "Horse-like" of "horse").

by feature alignment [6, 10, 14, 17, 32, 35, 46] and domain-invariant knowledge transfer [23, 25, 29, 41, 42]. Nevertheless, existing domain adaptation methods may suffer from catastrophic knowledge forgetting [19] in class-incremental target domain, inspiring methods specifically designed for Class-Incremental Unsupervised Domain Adaptation (CI-UDA) [16, 37, 40].

Recently, several CI-UDA methods have been proposed [16, 37, 40]. They usually consist of two technical components. On one hand, they typically store and utilize rehearsal data from previous target classes (as illustrated by the yellow circles in Figure 1) to retain historical knowledge. However, rehearsal data may not be available due to constraints such as data privacy or memory limitations (*e.g.,* the memory will increase as the number of tasks increases). On the other hand, to avoid negative knowledge transfer [9, 23], CI-UDA methods perform alignment only between classes shared across domains. However, they may still suffer from knowledge forgetting. Specifically, as shown in Figure 1, suppose the target class is "zebra" (a shared class) at step T. On one hand, the shared-class discovery process is imperfect. It may mistakenly treat private classes in source domain (*i.e.,* source-private classes), such as "tiger," as a shared class, leading to misalignment. On the other hand, even in source-private classes, valuable knowledge exists but may be ignored during cross-domain alignment, such as "Black-White" of "panda," and "Horse-like" of "horse," which are also typical properties of "zebra" [8]. Recent works utilize CLIP-based [22] prompt learning [4, 7, 13, 21, 27, 49] to deal with domain adaptation problem. Technically, some methods can be directly applied to the CI-UDA setting, but as they typically do not consider the catastrophic forgetting issue, their performance is still far from satisfactory.

In this paper, we propose cross-domain **Vis**ion-**T**ext **A**ttribute **A**lignment (VisTA), a novel CI-UDA framework based on CLIP [22]. In our framework, we aim to mine and preserve domain-invariant and class-agnostic knowledge. Firstly, inspired by [34], we employ an **Attribute Modeling** module to utilize CLIP to extract class-agnostic properties which we refer to as "attribute". We freeze the encoders of CLIP and construct a dictionary for source domain and target domain, respectively. The dictionaries store attributes in the form of "key-value" pairs, where the key and value bridge the visual and textual modalities. For each input image, several textual attributes are selected from the dictionary based on its visual attributes. These textual attributes, serving as prompts, are sent into CLIP to compute the class probability of the image. Then we perform attribute alignment across domains to mitigate the domain shift, via encouraging visual attention consistency and prediction consistency. Specifically, for each image in both domains, prompts are selected from the two dictionaries to compute paired class probabilities (one from each domain). However, since the two dictionaries are learned independently, the attributes selected from the other domain are domain-specific and may not effectively contribute to the current prediction due to the domain shift. Therefore, we introduce a **Visual Attention Consistency** (VAC) module to ensure the semantically similar attributes across domains are selected for paired prediction. VisTA then encourages **Prediction Consistency** by minimizing the Jensen–Shannon divergence between these paired probability distributions, enabling the learning of domain-invariant attributes. Benefiting from the modeling of domain-invariant and class-agnostic attributes, we are able to deal with the CI-UDA task in a rehearsal-free manner.

In a nutshell, the contributions of our work are summarized as follows:

- We propose a CI-UDA framework named VisTA, which leverages CLIP to learn class-agnostic attributes that act as prompts, achieving rehearsal-free training.
- VisTA learns domain-invariant attributes through attribute alignment, guided by a Visual Attention Consistency module and a Prediction Consistency loss.
- Extensive experiments firmly demonstrate the effectiveness of VisTA, as it achieves state-of-the-art performance on Office-31, Office-Home, and Mini-DomainNet.

2 Related Work

In this section, we review some relevant works, including unsupervised domain adaptation and vision language models.

Unsupervised Domain Adaptation. To mitigate distribution shift, conventional UDA methods or PDA methods typically fall into two main directions. The first direction of work aims to align the feature distributions across different domains. Common techniques include minimizing the statistical distribution metrics in the feature space directly [10, 32, 46] and applying adversarial learning to obtain domain-agnostic features [6, 14, 17]. The other direction of work seeks to transfer domain-invariant knowledge between models in source domain and target domain. For example, the works of [23, 25, 41] propose to learn an invariant classifier with consistent predictions, while [29, 42] propose to improve the performance of the target domain by knowledge distillation.

However, in practice, UDA is often integrated with continual learning problems [5, 33, 38]. One scenario is where target data arrives in a streaming manner with different classes. In this scenario, conventional UDA methods suffer from the catastrophic forgetting problem [19]. Therefore, some recent CI-UDA methods [16, 37, 40] have been developed to mitigate the domain shift while learning class-incremental target classes. For instance, ProCA [16] detects the shared classes by computing cumulative prediction probabilities of target examples and achieves adaptation through prototype

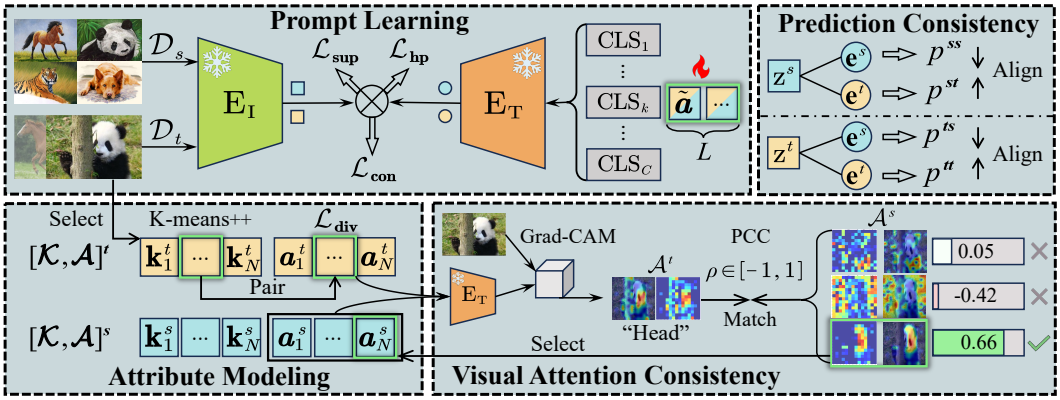


Figure 2: Framework of VisTA. Parameters of $\mathrm{E_I}$ and $\mathrm{E_T}$ in CLIP are frozen and only prompts are tunable during training. The classification loss $\mathcal{L}_{\text{sup}}$ is applied separately to $\mathcal{D}_s$ (visual features $\mathbf{z}^s$ and text embeddings $\mathbf{e}^s$ in blue) and $\mathcal{D}_t$ ($\mathbf{z}^t$ and $\mathbf{e}^t$ in yellow). VisTA integrates three core modules. Attribute Modeling module for constructing domain-specific dictionaries $[\mathcal{K}, \mathcal{A}]$ of visual attributes $\mathbf{k} \in \mathcal{K}$ and textual attributes $a \in \mathcal{A}$. Given an image from $\mathcal{D}_t$, VisTA selects target attributes $\tilde{a}^t$ based on the cosine similarity between $\mathbf{k}^t$ and $\mathbf{z}^t$, while selecting source attributes $\tilde{a}^s$ via a Visual Attention Consistency module. VisTA then aligns paired class probabilities (*e.g.*, p^{ts} and p^{tt}) via a Prediction Consistency loss $\mathcal{L}_{\text{con}}$. We use p^{tt} for inference on $\mathcal{D}_t$.

alignment. PLDCA [37] builds upon ProCA and further alleviates negative transfer through domain-level and instance-level contrastive alignment. Besides, GROTO [2] designs a multi-granularity class prototype self-organization module and a prototype topology distillation module to handle CI-UDA in a source-free scenario (*i.e.,* CI-SFUDA). It is worth noting that existing CI-UDA methods are suboptimal owing to continuously increasing memory and asymmetric alignment.

Vision Language Models. Recent Vision Language Models (VLMs) like CLIP [22] have demonstrated impressive performance on various downstream vision tasks by pretraining on large-scale image-text pairs [43]. VLMs typically use hand-crafted text like "a photo of a [class_name]" for zero-shot prediction on downstream tasks, which preserves generalization knowledge while maintaining low computational cost. However, hand-crafted text is not always effective, and thus prompt learning has gained increasing attention. For instance, CoOp [49] and CoCoOp [48] use learnable continuous prompts to improve the generalization performance of CLIP. MaPLe [11] proposes multi-modal prompt learning to align text and image representations. Moreover, since only the prompts should be stored, certain prompt learning methods serve as rehearsal-free learners, which can be effectively utilized to address class-incremental problem [34].

However, these prompt learning methods often suffer from performance degradation when encountering domain shift problems. To address this, DAPL [7] introduces domain-specific and domain-agnostic prompts to learn the label distribution of target domain. AD-CLIP [27] learns domain-invariant prompts by combining domain style information with image content information. DAMP [4] mutually aligns visual and textual embeddings to learn domain-agnostic prompts. PGA [21] frames UDA as a multi-objective optimization problem and promotes consensus among per-objective gradients. Although existing prompt learning methods have shown quite promising performance, they cannot deal with CI-UDA, as the historical knowledge encoded in prompts may be overwritten by new class information, leading to catastrophic forgetting. To this end, in this paper, we propose a rehearsal-free method based on prompt learning for CI-UDA, which effectively reduces catastrophic forgetting while mitigating distribution shift.

3 Preliminaries

3.1 CI-UDA Problem Formulation

In CI-UDA, we consider two domains, including the labeled source domain $\mathcal{D}_s = \{(\mathbf{x}_i^s, \mathbf{y}_i^s)\}^{n_s}$ (where $\mathbf{x}_i^s$ denotes the i-th source image, $\mathbf{y}_i^s \in \{1, 2 \cdots, C\}$ denotes the label of i-th image and n_s means the number of source examples), and unlabeled target domain $\mathcal{D}_t = \{\mathbf{x}_j^t\}^{n_t}$. The sample in $\mathcal{D}_s$ is available at all time steps and covers all the considered C classes, and the sample in $\mathcal{D}_t$ comes incrementally. For each time step, the underlying class set of $\mathcal{D}_t$ is only a subset of $\{1, \cdots, C\}$ and the class sets of different time steps are disjoint.

The goal of CI-UDA is to learn a model by leveraging data from $\mathcal{D}_s$ and class-incremental $\mathcal{D}_t$, such that it performs well on all seen classes of $\mathcal{D}_t$ during testing.

3.2 Prompt Learning in CLIP

CLIP [22], a prominent VLM, pretrains an image encoder $\mathrm{E_I}$ and a text encoder $\mathrm{E_T}$ on large-scale image-text pairs to learn well-aligned visual and textual representations. In downstream tasks, class-specific textual prompts $\mathbf{P}_k$ may be utilized for each class k (*e.g.,* "a photo of $[\mathrm{CLS}_k]$," where $[\mathrm{CLS}_k]$ is the k-th class name). CLIP predicts the probability that an input image $\mathbf{x}$ belongs to each class by computing the cosine similarity between the visual feature $\mathbf{z} = \mathrm{E_I}(\mathbf{x}) \in \mathbb{R}^D$ (where D is the feature dimension) and the class-wise text embeddings $\mathbf{w}_k = \mathrm{E_T}(\mathbf{P}_k) \in \mathbb{R}^D, k = 1, \cdots, C$:

$$p(\mathbf{y} = k|\mathbf{x}) = \frac{\exp\left(\cos\langle \mathbf{w}_k, \mathbf{z}\rangle/\tau\right)}{\sum_{c=1}^{C} \exp\left(\cos\langle \mathbf{w}_c, \mathbf{z}\rangle/\tau\right)}, \tag{1}$$

where $\cos\langle\cdot,\cdot\rangle$ is cosine similarity, and τ is temperature parameter.

However, such hand-crafted prompts may be suboptimal. To further enhance the performance of CLIP in downstream tasks, CoOp [49] introduces learnable continuous vectors $\boldsymbol{V}$ with length M to replace hand-crafted prompt templates. The learnable prompt for class k is then defined as $\mathbf{P}_k = [\boldsymbol{V}_1, \boldsymbol{V}_2, \cdots, \boldsymbol{V}_M, \mathrm{CLS}_k]$. During training, the prompt $\mathbf{P}_k$ is updated to minimize the cross-entropy loss on sample from downstream tasks. For the inference, they follow the same way which utilizes Eq. (1) to predict the label for each example. The only difference is that they use the learned prompts instead of manually designed prompts for prediction.

4 Method

As shown in Figure 2, at time step T, VisTA processes $\mathcal{D}_s$ and the current $\mathcal{D}_t$ through $\mathrm{E_I}$, generating visual features $\mathbf{z}^s$ and $\mathbf{z}^t$. Taking $\mathbf{z}^t$ as an example, we retrieve L textual attributes from target dictionary based on the cosine similarity between $\mathbf{z}^t$ and visual attributes (Section 4.1). These attributes serve as textual prompts for $\mathrm{E_T}$. Concurrently, we propose a Visual Attention Consistency module (Section 4.2), which applies a Grad-CAM-based attention heatmap matching mechanism to select L source attributes with similar semantic concepts as prompts. This process yields two class probability distributions for $\mathbf{z}^t$. The target attributes then are optimized through a Prediction Consistency loss (Section 4.2) to enable the learning of class-agnostic and domain-invariant knowledge. A similar procedure is adopted for $\mathbf{z}^s$. Finally, Section 4.3 discusses two regularization terms and the training objective of VisTA.

4.1 Attribute Modeling

In CI-UDA setting, the underlying class set of target domain at each time step is only a subset of that of source domain and is disjoint from that of previous time steps. If mitigating knowledge forgetting at the class-level, we may inevitably need to store previous target sample and discover shared classes between domains at each time step to perform alignment—a cumbersome process that may introduce too much noise during training. In this paper, we aim to mitigate the knowledge forgetting in CI-UDA at the "attribute" level. The "attribute" refers to the basic components which combine to support correct predictions.

Specifically, with CLIP extracting visual and textual features, we represent each attribute as a "key-value" pair, where the value refers to the textual representation of attribute and the key refers to the visual representation of attribute, in other words, the visual prototype. Formally, attributes are denoted as:

$$[\mathcal{K}, \mathcal{A}] \coloneqq [\{\mathbf{k}_1, \boldsymbol{a}_1\}, \{\mathbf{k}_2, \boldsymbol{a}_2\}, \cdots, \{\mathbf{k}_N, \boldsymbol{a}_N\}], \tag{2}$$

where each key $\mathbf{k}_i \in \mathbb{R}^D$ $(i = 1, 2, \ldots, N)$ is designed to capture the visual attributes of an image $\mathbf{x}$, and each value $\boldsymbol{a}_i \in \mathbb{R}^{M \times D}$ $(i = 1, 2, \ldots, N)$ encodes the textual description of a specific attribute.

VisTA maintains a source attribute dictionary $[\mathcal{K}^s, \mathcal{A}^s]$ and a target attribute dictionary $[\mathcal{K}^t, \mathcal{A}^t]$. We design specific update strategies for attributes to learn class-agnostic knowledge.

Visual Attributes. We perform K-means++ clustering on all source features to obtain source visual attributes $\mathcal{K}^s$ before training and keep $\mathcal{K}^s$ fixed during training. As the target classes appearing at different time steps are disjoint in CI-UDA, target visual attributes $\mathcal{K}^t$ are initialized via K-means++ clustering on the data from the first time step in the class-incremental training sequences. During each subsequent time step in class-incremental learning process, we apply a moving average strategy to update $\mathcal{K}^t$.

Textual Attributes. VisTA randomly initializes $\mathcal{A}^s$ and $\mathcal{A}^t$. These textual attributes are then modeled through supervised training or self-training to mine and preserve class-agnostic knowledge.

Given an image $\mathbf{x}$ from both domains, we select the top-L visual attributes $\widetilde{\mathcal{K}} \subseteq \mathcal{K}$ based on their cosine similarity to $\mathbf{x}$. The paired textual attributes are then indexed from the dictionary as $\widetilde{\mathcal{A}} = \widetilde{\boldsymbol{a}}_{1:L}$. These textual attributes are concatenated with the class name to form the prompt:

$$\mathbf{P}_k(\widetilde{\mathcal{A}}) = [\widetilde{\boldsymbol{a}}_1, \widetilde{\boldsymbol{a}}_2, \cdots, \widetilde{\boldsymbol{a}}_L, \mathrm{CLS}_k], k = 1, \cdots, C. \tag{3}$$

The source textual attributes $\mathcal{A}^s$ are optimized by minimizing cross-entropy loss on labeled $\mathcal{D}_s$:

$$\mathcal{L}^s_{\text{sup}} = -\log p\left(\mathbf{y} = \mathbf{y}^s | \mathbf{x}^s\right). \tag{4}$$

Moving to unlabeled $\mathcal{D}_t$, target textual attributes $\mathcal{A}^t$ are optimized by minimizing self-training loss:

$$\mathcal{L}^t_{\text{sup}} = -\mathbb{I}(\max(\hat{p}) \geq \gamma) \log p\left(\mathbf{y} = \hat{\mathbf{y}}^t | \mathbf{x}^t\right), \tag{5}$$

where $\hat{\mathbf{y}}^t$ represents the pseudo-label, $\mathbb{I}(\cdot)$ is the indicator function, γ is a threshold to select high-confidence pseudo-label of target example, and $\hat{p}$ denotes the debiased soft pseudo-label. This debiasing follows DebiasPL [36], which aims to enhance the reliability of pseudo-labels and has recently been adopted in several CLIP-based UDA methods [12, 15]. In detail, $\hat{p}$ is computed as:

$$\hat{p} = p - \tau \log q,\ q \leftarrow mq + (1 - m)\frac{1}{B}\sum_{j=1}^{B} p_j, \tag{6}$$

where m is a momentum coefficient, τ is a debias factor, B denotes the batch size, and q is initialized before training as a uniform probability vector over C classes.

4.2 Cross-Domain Attribute Alignment

To mitigate the domain shift in CI-UDA, VisTA performs cross-domain attribute alignment through a Visual Attention Consistency module and a Prediction Consistency loss. We want to mention that previous work AttriCLIP [34] can also extract class-agnostic attributes to benefit general continual learning, but cannot guarantee the domain-invariant property, which is crucial for CI-UDA. Hence, we design two novel modules to address this limitation.

Visual Attention Consistency (VAC). Taking $\mathbf{x}^t$ as an example, we select L visual attributes $\widetilde{\mathcal{A}}^t$ from target dictionary through cosine similarity between $\mathbf{x}^t$ and $\mathcal{K}^t$. Then we select L source visual attributes $\widetilde{\mathcal{A}}^s$ for alignment. Note that the update of visual attributes mentioned previously (Section 4.1) is domain-specific, so selecting $\widetilde{\mathcal{A}}^s$ for $\mathbf{x}^t$ by cosine similarity may introduce bias due to domain shift. Therefore, based on Grad-CAM [26], VisTA proposes an attention heatmap matching mechanism for cross-domain attribute selection.

Specifically, we compute L target CAM scores for $\mathbf{x}^t$ as:

$$S^t_{\text{CAM}} = \frac{\exp\left(\cos\langle \mathrm{E_T}(\boldsymbol{a}^t_m), \mathbf{z}^t\rangle/\tau\right)}{\sum_{n=1}^{|m|}\exp\left(\cos\langle \mathrm{E_T}(\boldsymbol{a}^t_n), \mathbf{z}^t\rangle/\tau\right)}. \tag{7}$$

where $\boldsymbol{a}^t_m \in \widetilde{\mathcal{A}}^t, (m = 1, \cdots, L)$ are individual textual attributes.

Table 1: Final Accuracy (%) on Office-31. DA, CI, and RF respectively represent domain adaptation, class-incremental, and rehearsal-free. The $*$ indicates the result is cited from GROTO [2].

Method	DA	CI	RF	Backbone	A→D	A→W	D→A	D→W	W→A	W→D	Avg.
ViT-B* (ICLR21)	–	–	–	ViT	82.4	80.0	70.8	83.1	75.1	87.4	79.8
ProCA (ECCV22)	✓	✓	✗		91.1	86.3	75.6	**98.3**	77.2	**99.4**	88.0
PLDCA (TIP24)	✓	✓	✗		**91.5**	**90.7**	75.6	96.9	77.8	99.4	88.7
CLIP (ICML21)	–	–	–	CLIP	77.2	75.2	79.7	75.2	79.7	77.2	77.4
CoOp (IJCV22)	–	–	–		89.0	87.0	83.0	96.1	81.5	98.1	89.1
AttriCLIP (CVPR23)	✗	✓	✓		82.8	85.7	79.5	94.4	79.8	95.4	86.3
AD-CLIP (ICCVW23)	✓	✗	✓		85.1	85.9	83.0	95.1	81.4	93.6	87.4
DAMP (CVPR24)	✓	✗	✓		82.8	79.3	81.3	90.3	80.1	91.7	84.3
PGA (NeurIPS24)	✓	✗	✓		49.0	68.4	81.2	88.7	81.3	81.8	75.1
DAPL (TNNLS25)	✓	✗	✓		79.1	76.7	79.5	79.2	81.1	79.3	79.2
VisTA (Ours)	✓	✓	✓		87.8	88.2	**84.5**	95.2	**84.0**	96.5	**89.4**

Then we use the gradients of S^t_{CAM} with respect to the features from a layer as weights, and perform a weighted aggregation to highlight attribute-level discriminative regions. This procedure follows Grad-CAM to generate L heatmaps $\mathbf{H}^t$ for $\mathbf{x}^t$. In the same way, we also compute N source CAM scores S^s_{CAM} for all source attributes $\mathcal{A}^s$, thereby obtaining N candidate heatmaps $\mathbf{H}^s$ for $\mathbf{x}^t$.

To select $\widetilde{\mathcal{A}}^s$ from N candidates, VisTA quantify the visual attention consistency between flattened $\mathbf{H}^t$ and $\mathbf{H}^s$ using the Pearson Correlation Coefficient (PCC): $\rho = \frac{\text{cov}(\mathbf{H}^s,\mathbf{H}^t)}{\sigma_{\mathbf{H}^s}\sigma_{\mathbf{H}^t}} \in [-1,1]$, where cov represents the covariance, and σ denotes the standard deviation. A higher value of ρ highlights attributes for prioritized selection.

The visual attention consistency in images can be interpreted as the similarity of semantic concepts. As illustrated in the bottom-right panel of Figure 2, we analyze a "panda" image from $\mathcal{D}_t$. The attention heatmap $\mathbf{H}^t$ of the selected target attribute $\widetilde{\mathcal{A}}^t$ exhibits concentrated activations in the central-right region of the image, likely corresponding to the semantic concept of "Head". Notably, the selected source attribute $\widetilde{\mathcal{A}}^s$ with the highest $\rho = 0.66$ explicitly aligns with the same semantic concept of "Head".

Therefore, for L selected target attributes $\widetilde{\mathcal{A}}^t$ and total N source attributes $\mathcal{A}^s$, we compute $L \times N$ PCC ρ. Leveraging these values, we match and identify the top-L attributes $\widetilde{\mathcal{A}}^s$ as the cross-domain attribute selection result for $\mathbf{x}^t$. The procedure to employ VAC module is identical for $\mathbf{x}^s$. Importantly, PCC, defined as cosine similarity on normalized vectors, reduces the influence of global style and is thus suitable for quantifying the visual attention consistency.

Prediction Consistency. To achieve attribute alignment, we introduce a Prediction Consistency loss applied to the attributes selected by the VAC module, which enforces domain invariance for these attributes exhibiting semantic similarity.

As illustrated in the top-right panel of Figure 2, the selected attributes $\widetilde{\mathcal{A}}^s$ and $\widetilde{\mathcal{A}}^t$ are served as textual prompts for E_T to generate class-wise text embeddings $\mathbf{e}^{s,t}_k = \text{E}_\text{T}(\mathbf{P}_k(\widetilde{\mathcal{A}}^{s,t})), k = 1,\ldots,C$. This enables the computation of paired class probabilities for $\mathbf{x}$ using Equation (1). For example, given an image $\mathbf{x}^t$ is sampled from $\mathcal{D}_t$, we utilize prompts $\mathbf{P}(\widetilde{\mathcal{A}}^s)$ and $\mathbf{P}(\widetilde{\mathcal{A}}^t)$ to obtain class probability vectors p^{ts} and p^{tt}, respectively. For an image $\mathbf{x}^s$ is sampled from $\mathcal{D}_s$, analogous terms p^{ss} and p^{st} are computed. Here, the superscripts of p denote the domain of the $\mathbf{x}$ (first symbol) and the dictionary from which the prompt is selected (second symbol).

The Prediction Consistency loss is achieved by minimizing the Jensen–Shannon divergence D_JS between each pair of class probabilities:

$$\mathcal{L}_{\text{con}} = \text{D}_\text{JS}(p^{ss}, p^{st}) + \text{D}_\text{JS}(p^{tt}, p^{ts}). \tag{8}$$

In this way, VisTA effectively reduces catastrophic knowledge forgetting while mitigating the domain shift by learning class-agnostic and domain-invariant attributes. During inference, we use p^{tt} as the prediction score of target example.

4.3 Training Objective

Notably, we aim to enhance the generalization capacity of $\mathcal{A}^s$, enabling it to effectively guide predictions in $\mathcal{D}_t$. Inspired by [39], VisTA proposes a regularization loss which minimizes the distance between class-wise embeddings $\mathbf{e}^s_k$ generated by $\mathcal{A}^s$ and those derived from hand-crafted prompts ($\mathbf{w}_k$):

$$\mathcal{L}_{\text{hp}} = \sum_{k=1}^{C} |\mathbf{e}^s_k - \mathbf{w}_k|. \tag{9}$$

Finally, to promote the diversity of textual attributes, a regularization loss is applied separately to both $\mathcal{D}_s$ and $\mathcal{D}_t$ to enforce orthogonality among the attributes within $\mathcal{A}$:

$$\mathcal{L}_{\text{div}} = \frac{1}{N(N-1)} \sum_{m=1}^{N} \sum_{n=m+1}^{N} |\cos\langle \text{E}_\text{T}(\boldsymbol{a}_m), \text{E}_\text{T}(\boldsymbol{a}_n)\rangle|. \tag{10}$$

As a result, the final optimization objective of VisTA is:

$$\mathcal{L} = \mathcal{L}_{\text{sup}} + \lambda_1\mathcal{L}_{\text{con}} + \lambda_2\mathcal{L}_{\text{hp}} + \lambda_3\mathcal{L}_{\text{div}}, \tag{11}$$

where λ_1, λ_2, and λ_3 are non-negative trade-off weights, and $\mathcal{L}_{\text{sup}} = \mathcal{L}^s_{\text{sup}} + \mathcal{L}^t_{\text{sup}}$ is the classification loss.

5 Experiment

5.1 Experimental Setup

Datasets. Office-31 [24] includes 31 categories from three domains: Amazon (A), DSLR (D), Webcam (W), totaling 4,600 images. **Office-Home** [31] comprises 65 categories across four distinct domains: Art (A), Clipart (C), Product (P), and Real World (R), totaling 15,500 images. **Mini-DomainNet** is a subset of DomainNet [20] and includes 126 categories across four domains: Clipart (C), Painting (P), Real World (R), and Sketch (S).

Table 2: Final Accuracy (%) on Office-Home. DA, CI, and RF respectively represent domain adaptation, class-incremental, and rehearsal-free. The ◇ indicates the result is cited from DAPL [7], and the ∗ indicates the result is cited from GROTO [2].

Method	DA	CI	RF	Backbone	A→C	A→P	A→R	C→A	C→P	C→R	P→A	P→C	P→R	R→A	R→C	R→P	Avg.
ViT-B* (ICLR21)	–	–	–	ViT	53.2	77.7	82.1	69.1	76.6	78.7	67.8	50.8	82.1	73.0	50.2	81.8	70.3
ProCA (ECCV22)	✓	✓	✗		56.8	81.9	89.9	68.4	80.9	83.9	70.1	51.3	89.9	77.4	50.3	87.9	74.1
PLDCA (TIP24)	✓	✓	✗		58.2	83.2	89.5	76.5	83.3	85.3	74.2	54.5	87.6	80.4	55.8	90.0	76.6
CLIP◇ (ICML21)	–	–	–	CLIP	67.8	89.0	89.8	82.9	89.0	89.8	82.9	67.8	89.8	82.9	67.8	89.0	82.4
CoOp (IJCV22)	–	–	–		70.6	88.9	89.8	81.4	88.4	87.9	79.8	69.6	89.6	83.3	71.6	91.9	82.7
AttriCLIP (CVPR23)	✗	✓	✓		67.4	87.9	89.4	84.0	90.2	87.3	82.3	67.3	88.7	83.9	69.0	90.4	82.3
AD-CLIP (ICCVW23)	✓	✗	✓		70.9	91.3	89.3	82.2	91.3	90.4	80.4	71.9	90.5	83.3	71.4	92.7	83.8
DAMP (CVPR24)	✓	✗	✓		68.3	86.7	88.7	81.3	90.1	88.2	81.5	68.3	89.1	80.3	69.0	90.7	81.9
PGA (NeurIPS24)	✓	✗	✓		61.6	70.9	85.3	55.7	83.7	84.3	58.5	59.7	81.3	59.4	63.1	82.1	70.5
DAPL (TNNLS25)	✓	✗	✓		69.3	91.2	90.5	83.3	90.1	90.1	83.3	68.9	90.2	82.2	70.4	90.8	83.4
VisTA (Ours)	✓	✓	✓		**71.8**	**92.8**	**91.3**	**84.4**	**92.9**	**91.1**	**84.7**	**72.8**	**91.5**	**84.8**	**72.2**	**92.8**	**85.3**

Table 3: Final Accuracy (%) on Mini-DomainNet. DA, CI, and RF respectively represent domain adaptation, class-incremental, and rehearsal-free. The ◇ indicates the result is cited from DAPL [7].

Method	DA	CI	RF	Backbone	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.
ViT-B (ICLR21)	–	–	–	ViT	50.2	61.9	43.8	50.8	76.8	51.6	58.9	60.0	44.9	60.8	48.6	66.4	56.2
ProCA (ECCV22)	✓	✓	✗		50.7	73.3	51.0	56.2	83.0	37.0	56.8	55.2	39.7	57.7	44.3	70.8	56.3
PLDCA (TIP24)	✓	✓	✗		56.2	71.5	51.9	61.5	78.6	52.5	60.5	66.2	42.9	62.1	58.8	71.9	61.2
CLIP◇ (ICML21)	–	–	–	CLIP	80.3	90.5	77.8	82.7	90.5	77.8	82.7	80.3	77.8	82.7	80.3	90.5	82.8
CoOp (IJCV22)	–	–	–		78.5	88.5	77.8	82.5	88.8	73.3	83.0	77.8	79.2	82.8	76.5	86.2	81.2
AttriCLIP (CVPR23)	✗	✓	✓		73.8	74.7	76.7	81.8	84.0	78.2	83.0	70.5	73.3	83.8	78.0	78.2	78.0
AD-CLIP (ICCVW23)	✓	✗	✓		77.5	89.5	75.7	85.7	90.8	77.5	83.8	78.7	79.8	85.2	79.5	89.8	82.8
DAMP (CVPR24)	✓	✗	✓		77.8	85.0	76.5	82.7	87.3	79.2	79.7	74.0	78.5	84.3	74.7	82.3	80.2
PGA (NeurIPS24)	✓	✗	✓		80.9	90.2	80.3	80.7	89.3	76.2	65.8	75.6	76.0	84.4	81.8	89.6	80.9
DAPL (TNNLS25)	✓	✗	✓		81.5	90.3	78.3	85.5	91.0	78.0	85.0	81.0	77.3	85.7	80.2	90.8	83.7
VisTA (Ours)	✓	✓	✓		**84.0**	**91.5**	**81.5**	**85.7**	**91.7**	**81.7**	**85.0**	**83.0**	**81.2**	**86.0**	**82.3**	**91.2**	**85.4**

Table 4: Step-level Accuracy (%) on Office-Home and Mini-DomainNet. DA, CI, and RF respectively represent domain adaptation, class-incremental, and rehearsal-free.

Method	DA	CI	RF	Office-Home							Mini-DomainNet						
				Step 1	Step 2	Step 3	Step 4	Step 5	Step 6	Avg.	Step 1	Step 2	Step 3	Step 4	Step 5	Step 6	Avg.
ProCA (ECCV22)	✓	✓	✗	73.2	74.0	72.3	73.3	73.8	74.1	73.5	59.8	55.3	57.8	57.1	55.1	56.3	56.9
PLDCA (TIP24)	✓	✓	✗	70.8	74.4	73.7	74.8	76.1	76.6	74.4	61.0	60.9	63.7	61.4	60.4	61.2	61.4
AttriCLIP (CVPR23)	✗	✓	✓	**82.7**	84.5	81.8	82.7	83.1	82.3	82.9	74.1	77.9	80.2	79.9	80.0	78.0	78.4
AD-CLIP (ICCVW23)	✓	✗	✓	79.5	83.5	81.8	83.5	83.3	83.8	82.6	81.5	83.2	84.4	83.9	83.6	82.1	83.1
DAMP (CVPR23)	✓	✗	✓	81.0	83.3	82.1	82.4	82.0	81.9	82.1	82.5	83.5	84.2	83.2	82.3	80.2	82.6
DAPL (TNNLS25)	✓	✗	✓	81.6	83.7	81.9	82.9	83.4	83.4	82.8	**83.7**	**85.6**	86.3	85.5	84.3	83.7	84.9
VisTA (Ours)	✓	✓	✓	80.3	**84.7**	**84.2**	**85.1**	**85.1**	**85.3**	**84.1**	83.5	85.5	**87.2**	**86.6**	**86.1**	**85.4**	**85.7**

Following ProCA [16], we divide each domain of Office-31 into three disjoint subsets, each containing 10 classes in alphabetical order, and divide each domain of Office-Home into six disjoint subsets, each containing 10 classes in an order consistent with ProCA [16]. Additionally, as the first CI-UDA method to handle Mini-DomainNet, we divide each domain into six disjoint subsets, each containing 20 classes in alphabetical order. More details of datasets construction are in **Appendix**[1] **A**.

Baseline Methods. We compare VisTA with five types of methods: (1) source-only: ViT-B/16 and CoOp [49]; (2) zero-shot: CLIP; (3) existing CI-UDA methods: ProCA [16] and PLDCA [37]; (4) prompt learning method for CIL: AttriCLIP [27]; (5) prompt learning methods for UDA: AD-CLIP [27], DAMP [4], PGA [21], and DAPL [7].

Implementation Details. We use ViT-B/16 [3] as the image encoder for VisTA and all baseline methods. Details on the hyperparameters of VisTA, as well as the training procedures for VisTA and several baseline methods, are provided in **Appendix B**. We analyze the sensitivity of our method to hyperparameters in Section 5.3.

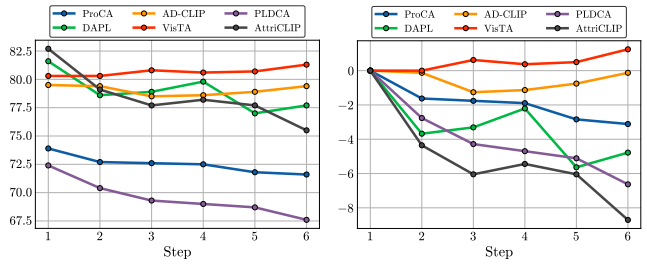
Evaluation Metrics. To comprehensively evaluate the performance of VisTA, we employ three metrics for CI-UDA:

1) **Final Accuracy**: the classification accuracy across all classes at the final time step for each adaptation task;

2) **Step-level Accuracy**: the average classification accuracy over all adaptation tasks at each time step;

3) **S-1 Accuracy**: the average classification accuracy of all adaptation tasks at each time step for classes in Step-1.

[1]All the appendices can be found at https://github.com/RyunMi/VisTA.



(a) S-1 Acc. (%) (b) Δ% (Step-1)

Figure 3: S-1 Accuracy at each step and its percentage change (Δ%) compared with Step-1 on Office-Home.

5.2 Comparisons with previous state-of-the-arts

The Final Accuracy results are summarized in Tables 1, 2, 3, while the Step-level Accuracy results are detailed in Table 4. The numbers reported in the tables are reproduced by us using the officially released code, unless otherwise specified. Additionally, the results of S-1 Accuracy are visualized in Figure 3. Due to page limitations, complete results with extended details for all metrics across three benchmarks are reported in **Appendix C**.

Comprehensive learning ability of VisTA. In Tables 1, 2, 3, each column corresponds to one specific CI-UDA task, *i.e.,* Source → Target. The column "Avg." means the average of Final Accuracy for all CI-UDA tasks.

The results of Final Accuracy demonstrate that VisTA performs favorably against other methods. VisTA achieves improvements of 0.7%, 8.7%, and 24.2% over the leading CI-UDA method PLDCA on Office-31, Office-Home, and Mini-DomainNet, respectively. It also outperforms other top competitors, surpassing CoOp by 0.3% on Office-31, AD-CLIP by 1.5% on Office-Home, and DAPL by 1.7% on Mini-DomainNet. We find that the performance advantages of VisTA scale with benchmark complexity (Office-31→Office-Home→Mini-DomainNet), as quantified by the number of time steps and underlying classes. It substantiates that the class-agnostic and domain-invariant attributes learned by VisTA effectively alleviate catastrophic forgetting and domain shift, especially in scenarios with numerous classes and long-sequence tasks.

The results further indicate that existing CI-UDA methods with ViT-B/16 underperform against source-only CoOp (on all benchmarks) and zero-shot CLIP (except on Office-31). This highlights the exceptional generalization ability of pretrained CLIP, which encodes comprehensive category knowledge via prompt learning.

Additionally, some UDA methods (*i.e.,* DAMP, PGA) and CIL method (*i.e.,* AttriCLIP) based on prompt learning obtain worse results than source-only CoOp across all three benchmarks. This suggests that an exclusive focus on mitigating either domain shift or catastrophic forgetting may reduce the generalization capability of prompt learning methods in handling CI-UDA.

Incremental learning ability of VisTA. The Step-level Accuracy and S-1 Accuracy are used to evaluate whether a method can retain knowledge of previous target classes while learning new ones. Each column in Table 4 represents the average accuracy of all adaptation tasks at a specific time step, and the column "Avg." denotes the average Step-level Accuracy. The x-axis indicates time steps and the y-axis shows the S-1 Accuracy in Figure 3(a).

It is observed that all comparison methods are affected by catastrophic forgetting, resulting in lower S-1 Accuracy at the final step compared with the first step. Some methods also exhibit a decline in Step-level Accuracy over time, indicating that they not only forget previously learned target classes but also struggle to learn new ones. In contrast, VisTA shows an upward trend in both Step-level Accuracy and S-1 Accuracy over time, achieving the best performance at both the final step and on average. The results of both metrics also show that the performance in the early adaptation phase is not state-of-the-art. We analyze that although VisTA aligns attributes across domains as much as possible, the learned $\mathcal{A}^t$ is not sufficiently refined in the early stage, while $\mathcal{A}^s$ is trained on all classes. Aligning $\mathcal{A}^s$ and $\mathcal{A}^t$ with unequal learning progress may harm performance. Only after $\mathcal{A}^t$ fully learns the attributes over time steps can the performance advantages manifest.

Moreover, Figure 3 (b) illustrates the percentage change in S-1 Accuracy at each step compared with the first step. VisTA is the only method that consistently achieves positive gains and demonstrates continuous improvement. This indicates that VisTA effectively preserves and progressively reinforces knowledge of class-incremental $\mathcal{D}_t$ when addressing CI-UDA on Office-Home.

Extension to source-free scenario. In **Appendix D**, we compare VisTA with the CI-SFUDA method GROTO [2]. These results demonstrate the capabilities of VisTA under source-free scenarios.

Table 5: Ablation analysis on Office-Home.

Method	w/o. VAC	w/o. $\mathcal{L}_{\text{con}}$	w/o. $\mathcal{L}_{\text{hp}}$	w/o. $\mathcal{L}_{\text{div}}$	VisTA
Final Acc. (%)	84.9	83.5	85.0	85.1	**85.3**
Final S-1 Acc. (%)	80.5	80.3	80.9	81.0	**81.3**

5.3 Ablation Analysis

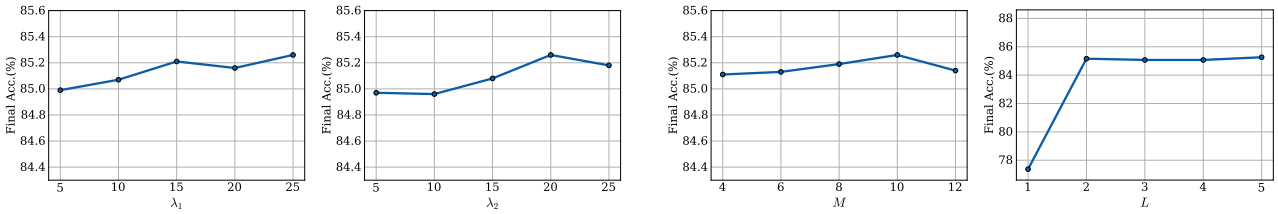
Table 5 presents Final Accuracy and S-1 Accuracy at the final step (*i.e.,* Final S-1 Accuracy) on Office-Home, obtained by removing specific modules (*i.e.,* "w/o.") while keeping other settings identical. More studies about various VLMs and computational overhead are reported in **Appendix E**.

Effect of VAC. The "w/o. VAC" is achieved by removing the VAC module, with attributes selected directly from the dictionary corresponding to the other domain using cosine similarity. This leads to a noticeable performance degradation, demonstrating that the selection enabled by the VAC module effectively mitigates bias caused by the domain shift.

Effect of $\mathcal{L}_{\text{con}}$. The "w/o. $\mathcal{L}_{\text{con}}$" indicates that $\mathcal{D}_s$ and $\mathcal{D}_t$ share the same attributes dictionary without $\mathcal{L}_{\text{con}}$. Notably, the "w/o. $\mathcal{L}_{\text{con}}$" leads to the sharpest performance decline, demonstrating that separate attribute modeling in $\mathcal{D}_s, \mathcal{D}_t$ with alignment effectively prevents conflicting knowledge acquisition in entangled attributes.

Effect of regularization terms. The "w/o. $\mathcal{L}_{\text{hp}}$," and "w/o. $\mathcal{L}_{\text{div}}$" denote the $\mathcal{L}_{\text{hp}}$ and $\mathcal{L}_{\text{div}}$ are removed from the objective (11), respectively. The observed performance decline confirms the effectiveness of both regularization terms in attribute learning.

Sensitivity to loss weights. We need to determine three loss weights of the objective (11). Empirical observations reveal λ_3 has



(a) Sensitivity to loss weight λ_1 and λ_2.

(b) Sensitivity to hyperparameters M and L of attribute dictionary.

Figure 4: Hyperparameter sensitivity analysis with respect to Final Accuracy on Office-Home.

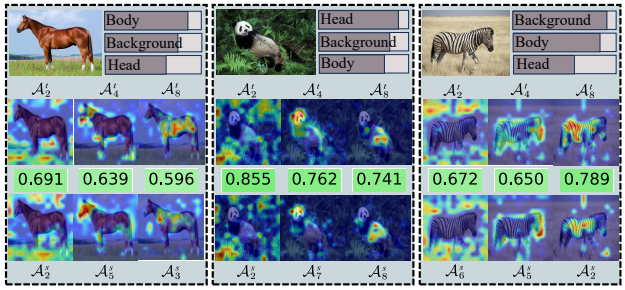
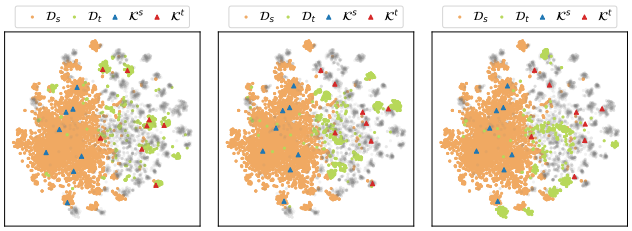


Figure 5: Grad-CAM visualization of C → R task (Mini-DomainNet) for different classes in $\mathcal{D}_t$. Attributes selected directly from $[\mathcal{K}, \mathcal{A}]^t$ are displayed in a ranked order beside the first-row images. The second and third rows show some attention heatmaps for $\mathcal{A}^t$ and $\mathcal{A}^s$, respectively. Heatmaps with the same semantic concept are grouped in columns, and their matching is guided by ρ (highlighted in green).



(a) Step 1 (b) Step 4 (c) Step 6

Figure 6: t-SNE visualization of C → P task from Office-Home at different time steps.

negligible impact, so we fix it at 1.0 and vary λ_1 and λ_2. As shown in Figure 4(a), the performance of VisTA is generally insensitive to $\lambda_1, \lambda_2 \in [5, 10, 15, 20, 25]$, with best performance at $\lambda_1 = 25, \lambda_2 = 20$.

Sensitivity to hyperparameters of attribute dictionary. We consider the following hyperparameters, such as the prompt length M, the number of attributes in the bank N, and the number of selected attributes L. To cap training and computational costs, we fix $N = 8$ and explore variations in M and L. Figure 4(b) shows that the performance of VisTA is robust to M. Moreover, when a sufficient number of attributes are selected ($L \geq 2$), VisTA also exhibits robustness to L.

Visualization of textual attributes. To verify whether the learned attributes reflect the semantic concept of images, we visualize the image contents of distinct classes corresponding to different attributes using Grad-CAM [26]. To further demonstrate the attribute matching process in VAC module, we present examples of target classes "horse," "panda," and "zebra" from Mini-DomainNet.

As shown in Figure 5, the learned $\mathcal{A}^t$ exhibit two key properties: (1) different attributes reflect distinct semantic concepts within the same image (*e.g.*, $\mathcal{A}^t_2 \rightarrow$ "Background," $\mathcal{A}^t_4 \rightarrow$ "Head," $\mathcal{A}^t_8 \rightarrow$ "Body"), and (2) the same attribute reflects identical semantic concept across different images. This demonstrates that the learned $\mathcal{A}^t$ are class-agnostic and diverse, effectively retaining knowledge to alleviate catastrophic forgetting. Unlike the learned $\mathcal{A}^t$ corresponding to $\mathcal{D}_t$, the $\mathcal{A}^s$, affected by the distribution shift, fail to learn attributes identical to $\mathcal{A}^t$ and do not exhibit property (2). However, $\mathcal{A}^s$ may still be partially similar to $\mathcal{A}^t$ in semantic concepts. Building on this similarity, VAC module selects cross-domain attributes through a ρ-guided matching mechanism to learn domain-invariant attributes that mitigate the distribution shift.

Visualization of visual attributes. To verify whether $\mathcal{K}^s$ and $\mathcal{K}^t$ can adequately cover the attributes of all examples, we conducted t-SNE visualizations for C → P task from Office-Home at steps 1, 4, and 6 as shown in Figure 6. It displays CLIP-extracted visual features from $\mathcal{D}_s$ (orange) and $\mathcal{D}_t$ (green), along with $\mathcal{K}$ obtained through K-means++ clustering. We observe that the elements within $\mathcal{K}^s$ (blue) and $\mathcal{K}^t$ (red) consistently remain diverse and distinct. Furthermore, $\mathcal{K}^t$ at the final step successfully covers the examples from other time steps (gray), demonstrating that $\mathcal{K}$ effectively captures the overall attributes of the sample from both domains.

Conclusion

In this paper, we propose to model and align attributes across domains based on CLIP to deal with the class-incremental unsupervised domain adaptation (CI-UDA), which is a rehearsal-free approach. Specifically, via CLIP, we extract the class-agnostic properties, *i.e.*, attributes. Each attribute is represented as a "key-value" pair where the key corresponds to visual prototype and the value corresponds to textual prompt. In our method, we learn to construct two dictionaries, each corresponding to a specific domain. Each dictionary consists of a group of attributes. Then we perform attribute alignment to make attribute invariant across domains via utilizing the consistency knowledge including visual attention consistency and prediction consistency. Experiments on three benchmarks verify the effectiveness of our proposed method.

Acknowledgments

This research is supported by the National Natural Science Foundation of China (NSFC) under Grants Nos. 62336003, 12371510, 92370114, and 62006119; and the National Key Research and Development Program of China (International Collaboration Special Project, No. SQ2023YFE0102775).

References

[1] Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. 2022. A Continual Learning Survey: Defying Forgetting in Classification Tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 7 (2022), 3366–3385.

[2] Peihua Deng, Jiehua Zhang, Xichun Sheng, Chenggang Yan, Yaoqi Sun, Ying Fu, and Liang Li. 2025. Multi-Granularity Class Prototype Topology Distillation for Class-Incremental Source-Free Unsupervised Domain Adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition*. 30566–30576.

[3] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations*.

[4] Zhekai Du, Xinyao Li, Fengling Li, Ke Lu, Lei Zhu, and Jingjing Li. 2024. Domain-Agnostic Mutual Prompting for Unsupervised Domain Adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition*. 23375–23384.

[5] Yu Feng, Zhen Tian, Yifan Zhu, Zongfu Han, Haoran Luo, Guangwei Zhang, and Meina Song. 2024. CP-Prompt: Composition-Based Cross-modal Prompting for Domain-Incremental Continual Learning. In *Proceedings of the 32nd ACM International Conference on Multimedia (MM '24)*. 2729–2738.

[6] Yaroslav Ganin and Victor Lempitsky. 2015. Unsupervised Domain Adaptation by Backpropagation. In *International Conference on Machine Learning*, Vol. 37. PMLR, Lille, France, 1180–1189.

[7] Chunjiang Ge, Rui Huang, Mixue Xie, Zihang Lai, Shiji Song, Shuang Li, and Gao Huang. 2025. Domain Adaptation via Prompt Learning. *IEEE Transactions on Neural Networks and Learning Systems* 36, 1 (2025), 1160–1170.

[8] Zhuo Huang, Jian Yang, and Chen Gong. 2023. They are Not Completely Useless: Towards Recycling Transferable Unlabeled Data for Class-Mismatched Semi-Supervised Learning. *IEEE Transactions on Multimedia* 25 (2023), 1844–1857.

[9] Taotao Jing, Haifeng Xia, and Zhengming Ding. 2020. Adaptively-Accumulated Knowledge Transfer for Partial Domain Adaptation. In *Proceedings of the 28th ACM International Conference on Multimedia (MM '20)*. 1606–1614.

[10] Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G. Hauptmann. 2019. Contrastive Adaptation Network for Unsupervised Domain Adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition*. 4893–4902.

[11] Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. 2023. MaPLe: Multi-Modal Prompt Learning. In *IEEE Conference on Computer Vision and Pattern Recognition*. 19113–19122.

[12] Zhengfeng Lai, Noranart Vesdapunt, Ning Zhou, Jun Wu, Cong Phuoc Huynh, Xuelu Li, Kah Kuen Fu, and Chen-Nee Chuah. 2023. PADCLIP: Pseudo-labeling with Adaptive Debiasing in CLIP for Unsupervised Domain Adaptation. In *IEEE International Conference on Computer Vision*. 16155–16165.

[13] Jingzheng Li and Hailong Sun. 2023. LiFT: Transfer Learning in Vision-Language Models for Downstream Adaptation and Generalization. In *Proceedings of the 31st ACM International Conference on Multimedia (MM '23)*. 4678–4687.

[14] Shuang Li, Chi Harold Liu, Binhui Xie, Limin Su, Zhengming Ding, and Gao Huang. 2019. Joint Adversarial Domain Adaptation. In *Proceedings of the 27th ACM International Conference on Multimedia (MM '19)*. 729–737.

[15] Xinyao Li, Yuke Li, Zhekai Du, Fengling Li, Ke Lu, and Jingjing Li. 2024. Split to Merge: Unifying Separated Modalities for Unsupervised Domain Adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition*. 23364–23374.

[16] Hongbin Lin, Yifan Zhang, Zhen Qiu, Shuaicheng Niu, Chuang Gan, Yanxia Liu, and Mingkui Tan. 2022. Prototype-Guided Continual Adaptation for Class-Incremental Unsupervised Domain Adaptation. In *European Conference on Computer Vision*. Springer Nature Switzerland, Cham, 351–368.

[17] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. 2018. Conditional Adversarial Domain Adaptation. In *Advances in Neural Information Processing Systems*, Vol. 31. Curran Associates, Inc.

[18] Marc Masana, Xialei Liu, Bartłomiej Twardowski, Mikel Menta, Andrew D. Bagdanov, and Joost van de Weijer. 2023. Class-Incremental Learning: Survey and Performance Evaluation on Image Classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 5 (2023), 5513–5533.

[19] Michael McCloskey and Neal J Cohen. 1989. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*. Vol. 24. Academic Press, 109–165.

[20] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. 2019. Moment Matching for Multi-Source Domain Adaptation. In *IEEE International Conference on Computer Vision*. 1406–1415.

[21] Hoang Phan, Lam Tran, Quyen Tran, and Trung Le. 2024. Enhancing Domain Adaptation through Prompt Gradient Alignment. In *Advances in Neural Information Processing Systems*, Vol. 37. 45518–45551.

[22] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *International Conference on Machine Learning*. PMLR, 8748–8763.

[23] Chuan-Xian Ren, Pengfei Ge, Peiyi Yang, and Shuicheng Yan. 2021. Learning Target Domain Specific Classifier for Partial Domain Adaptation. *IEEE Transactions on Neural Networks and Learning Systems* 32, 5 (2021), 1989–2001.

[24] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. 2010. Adapting Visual Category Models to New Domains. In *European Conference on Computer Vision*. Springer Berlin Heidelberg, Berlin, Heidelberg, 213–226.

[25] Kuniaki Saito, Donghyun Kim, Stan Sclaroff, Trevor Darrell, and Kate Saenko. 2019. Semi-Supervised Domain Adaptation via Minimax Entropy. In *IEEE International Conference on Computer Vision*. 8050–8058.

[26] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2017. Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization. In *IEEE International Conference on Computer Vision*. 618–626.

[27] Mainak Singha, Harsh Pal, Ankit Jha, and Biplab Banerjee. 2023. AD-CLIP: Adapting Domains in Prompt Space Using CLIP. In *IEEE International Conference on Computer Vision Workshops*. 4355–4364.

[28] Hongbo Sun, Jiahuan Zhou, Xiangteng He, Jinglin Xu, and Yuxin Peng. 2024. FineFMPL: Fine-grained Feature Mining Prompt Learning for Few-Shot Class Incremental Learning. In *International Joint Conferences on Artificial Intelligence*. 1299–1307.

[29] Jialiang Tang, Shuo Chen, Gang Niu, Hongyuan Zhu, Joey Tianyi Zhou, Chen Gong, and Masashi Sugiyama. 2025. Direct Distillation between Different Domains. In *European Conference on Computer Vision*. Springer Nature Switzerland, Cham, 154–172.

[30] Xiaoyu Tao, Xiaopeng Hong, Xinyuan Chang, Songlin Dong, Xing Wei, and Yihong Gong. 2020. Few-Shot Class-Incremental Learning. In *IEEE Conference on Computer Vision and Pattern Recognition*.

[31] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. 2017. Deep Hashing Network for Unsupervised Domain Adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition*. 5018–5027.

[32] Jindong Wang, Wenjie Feng, Yiqiang Chen, Han Yu, Meiyu Huang, and Philip S. Yu. 2018. Visual Domain Adaptation with Manifold Embedded Distribution Alignment. In *Proceedings of the 26th ACM International Conference on Multimedia (MM '18)*. 402–410.

[33] Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. 2022. Continual Test-Time Domain Adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition*. 7201–7211.

[34] Runqi Wang, Xiaoyue Duan, Guoliang Kang, Jianzhuang Liu, Shaohui Lin, Songcen Xu, Jinhu Lü, and Baochang Zhang. 2023. AttriCLIP: A Non-Incremental Learner for Incremental Knowledge Learning. In *IEEE Conference on Computer Vision and Pattern Recognition*. 3654–3663.

[35] Xuesong Wang, Yuting Ma, and Yuhu Cheng. 2018. Domain Adaptation Network Based on Autoencoder. *Chinese Journal of Electronics* 27, 6 (2018), 1258–1264.

[36] Xudong Wang, Zhirong Wu, Long Lian, and Stella X. Yu. 2022. Debiased Learning From Naturally Imbalanced Pseudo-Labels. In *IEEE Conference on Computer Vision and Pattern Recognition*. 14647–14657.

[37] Kun Wei, Xu Yang, Zhe Xu, and Cheng Deng. 2024. Class-Incremental Unsupervised Domain Adaptation via Pseudo-Label Distillation. *IEEE Transactions on Image Processing* 33 (2024), 1188–1198.

[38] Yinsong Xu, Zhuqing Jiang, Aidong Men, Yang Liu, and Qingchao Chen. 2022. Delving into the Continuous Domain Adaptation. In *Proceedings of the 30th ACM International Conference on Multimedia (MM '22)*. 6039–6049.

[39] Hantao Yao, Rui Zhang, and Changsheng Xu. 2023. Visual-Language Prompt Tuning with Knowledge-guided Context Optimization. In *IEEE Conference on Computer Vision and Pattern Recognition*. 6757–6767.

[40] Jiaping Yu, Muli Yang, Aming Wu, and Cheng Deng. 2025. Memory-Enhanced Confidence Calibration for Class-Incremental Unsupervised Domain Adaptation. *IEEE Transactions on Multimedia* 27 (2025), 610–621.

[41] Zhongqi Yue, Qianru Sun, and Hanwang Zhang. 2023. Make the U in UDA Matter: Invariant Consistency Learning for Unsupervised Domain Adaptation. In *Advances in Neural Information Processing Systems*, Vol. 36. Curran Associates, Inc., 26991–27004.

[42] Bo Zhang, Xiaoming Zhang, Yun Liu, Lei Cheng, and Zhoujun Li. 2021. Matching Distributions between Model and Data: Cross-domain Knowledge Distillation for Unsupervised Domain Adaptation. In *Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Online, 5423–5433.

[43] Jingyi Zhang, Jiaxing Huang, Sheng Jin, and Shijian Lu. 2024. Vision-Language Models for Vision Tasks: A Survey. *IEEE Transactions on Pattern Analysis and*

Machine Intelligence 46, 8 (2024), 5625–5644.

[44] Jinghua Zhang, Li Liu, Olli Silvén, Matti Pietikäinen, and Dewen Hu. 2025. Few-Shot Class-Incremental Learning for Classification and Object Detection: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 47, 4 (2025), 2924–2945.

[45] Ruru Zhang, Haihong E, and Meina Song. 2024. FSCIL-EACA: Few-Shot Class-Incremental Learning Network Based on Embedding Augmentation and Classifier Adaptation for Image Classification. *Chinese Journal of Electronics* 33, 1 (2024), 139–152.

[46] Yuchen Zhang, Tianle Liu, Mingsheng Long, and Michael Jordan. 2019. Bridging Theory and Algorithm for Domain Adaptation. In *International Conference on Machine Learning*. PMLR, 7404–7413.

[47] Da-Wei Zhou, Qi-Wei Wang, Zhi-Hong Qi, Han-Jia Ye, De-Chuan Zhan, and Ziwei Liu. 2024. Class-Incremental Learning: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 46, 12 (2024), 9851–9873.

[48] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. 2022. Conditional Prompt Learning for Vision-Language Models. In *IEEE Conference on Computer Vision and Pattern Recognition*. 16816–16825.

[49] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. 2022. Learning to Prompt for Vision-Language Models. *International Journal of Computer Vision* 130, 9 (2022), 2337–2348.