Domain-Agnostic Mutual Prompting for Unsupervised Domain Adaptation

Zhekai Du[1], Xinyao Li[1], Fengling Li[2], Ke Lu[1], Lei Zhu[3], Jingjing Li[1]
[1]University of Electronic Science and Technology of China;
[2]University of Technology Sydney; [3]Tongji University
{zhekaid, xinyao326}@std.uestc.edu.cn, lijin117@yeah.net

Abstract

Conventional Unsupervised Domain Adaptation (UDA) strives to minimize distribution discrepancy between domains, which neglects to harness rich semantics from data and struggles to handle complex domain shifts. A promising technique is to leverage the knowledge of large-scale pre-trained vision-language models for more guided adaptation. Despite some endeavors, current methods often learn textual prompts to embed domain semantics for source and target domains separately and perform classification within each domain, limiting cross-domain knowledge transfer. Moreover, prompting only the language branch lacks flexibility to adapt both modalities dynamically. To bridge this gap, we propose Domain-Agnostic Mutual Prompting (DAMP) to exploit domain-invariant semantics by mutually aligning visual and textual embeddings. Specifically, the image contextual information is utilized to prompt the language branch in a domain-agnostic and instance-conditioned way. Meanwhile, visual prompts are imposed based on the domain-agnostic textual prompt to elicit domain-invariant visual embeddings. These two branches of prompts are learned mutually with a cross-attention module and regularized with a semantic-consistency loss and an instance-discrimination contrastive loss. Experiments on three UDA benchmarks demonstrate the superiority of DAMP over state-of-the-art approaches [1].

1. Introduction

Labeling scarcity is a perennial problem in deep learning, as collecting abundant labeled data can be expensive, time-consuming, or even infeasible [31, 58]. Unsupervised Domain Adaptation (UDA) serves as a promising approach to leverage the knowledge from a well-labeled source domain to benefit the task on an unlabeled target domain, where the two domains have similar semantics but different data distributions [7, 16, 45].

[1]Code is available at: https://github.com/TL-UESTC/DAMP.

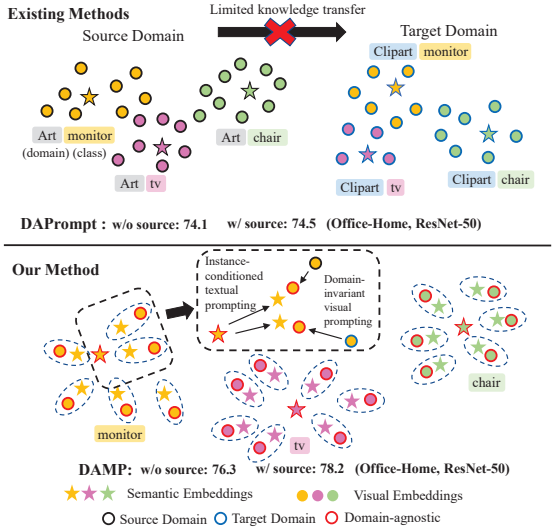


Figure 1. Top: exsiting prompt-based methods (e.g., DAPrompt [8]) only learn textual prompts to embed semantics for each domain and perform classification separately, which limits cross-domain knowledge transfer and feature alignment. Bottom: our method learns both textual and visual prompts mutually to make both modalities of embeddings domain-invariant, thus enabling better utilization of source knowledge and flexibility alignment.

Conventional UDA methods typically bridge the domain gap by minimizing the distribution discrepancy, through either moment matching [16, 21, 25, 26, 33] or adversarial learning [5, 7, 27, 36]. However, roughly aligning two domains can result in distorted semantic structure and less class discriminability in learned feature representations [2, 42]. Besides, prior works use numerical labels for training and inference, which discard rich semantics behind categories, leading to sub-optimal adaptation when handling complex categories and domain shifts.

Recently, large-scale pre-trained Vision-Language Models (VLMs) have demonstrated impressive successes in various downstream tasks [10, 13, 35, 54]. By pre-training on tremendous image-text pairs, these models learn transferable multimodal representations that align images and texts in a joint embedding space. In particular, the Contrastive Language-Image Pre-training (CLIP) model [34] encodes

rich semantic knowledge about visual concepts, presenting new opportunities to address the domain gap by leveraging the pre-trained vision and language knowledge. However, few attempts have been made to leverage VLMs for UDA since two challenges stand in the way, namely, 1) how to effectively take advantage of the rich *pre-trained knowledge* encoded in VLMs, and 2) how to transfer the *source knowledge* to the target domain for better adaptation.

Generally, there are two feasible routes for adapting large-scale pre-trained VLMs. The first is to use the zero-shot prediction capacity of VLMs to obtain pseudo-labels and fine-tune the image backbone with other UDA techniques [19]. While the *source knowledge* can be well-encoded by fine-tuning, the pseudo-labels largely rely on manually designed textual descriptions and fine-tuning the model may ruin the *pre-trained knowledge*. Another way is to freeze the pre-trained model and only tune the input data (e.g., prompt) for model adaptation, which only involves a small set of learnable parameters and retains the *pre-trained knowledge*. For instance, DAPrompt [8] proposes to embed domain semantics into domain-specific textual prompts for each domain, which are then coupled with a domain-agnostic context for domain-specific classification in the joint CLIP space. However, we argue that a large portion of *source knowledge* is prone to be encoded in source-specific prompts, which cannot be transfered to the target domain. For instance, we conduct an experiment by disabling the source supervision loss in DAPrompt, and observe marginal influence on the target performance (see Fig. 1).

In this work, we aim to learn transferable (domain-agnostic) prompts to effectively leverage both *pre-trained knowledge* and *source-knowledge* for the target domain using CLIP. However, directly learning such textual prompts in UDA can be sub-optimal, as visual embeddings from different domains typically encompass distinct, domain-biased information that conforms to different distributions within the CLIP space. This is a key motivation behind domain-specific prompting in previous methods [8, 38]. Inspired by the recent success of visual prompting [14], we propose also adapting the visual embeddings to elicit domain-invariant representations by prompting the vision backbone, based on the domain-agnostic textual prompt. Meanwhile, domain-invariant visual embeddings can still retain individual characteristics, e.g., object color and size. Such variations, even within the same category, necessitate instance-conditioned textual prompts for better alignment, as shown in Fig. 1. Given the interdependent nature of the two kinds of prompts, we build a mutual learning framework based on a cross-attention mechanism inspired by the Transformer decoder [47]. A semantic-consistency regularization and an instance-discrimination contrastive loss are further imposed to ensure that the learned prompts carry pure domain-agnostic and instance-conditioned information.

In summary, the key contributions of this work are threefold: 1) We propose a novel framework termed DAMP to learn domain-agnostic prompts for transferring pre-trained knowledge and source knowledge to the target domain using CLIP. 2) DAMP mutually aligns textual and visual embeddings by prompting both modalities to learn domain-invariant representations, which are optimized with two elaborate regularizations. 3) Extensive experiments on three UDA benchmarks validate that DAMP brings consistent and notable gains over state-of-the-art approaches.

2. Related Works

Unsupervised Domain Adaptation. To enable effective knowledge transfer, modern UDA methods typically fall into two technical routes. The first line of works aim to reduce the domain shift by aligning the feature distributions across domains. Common techniques include minimizing statistical distribution distances via moment matching [26, 33, 40] and learning domain-invariant features via adversarial alignment [7, 36, 37, 57]. More recent methods focus on disentangling domain-invariant and domain-specific factors for casual invariance [30, 55] or self-training with elaborate pseudo labels [24, 29, 49]. The second line of works resort to more large-scale networks, e.g., Vision Transformer (ViT) [4], for more transferable features. For instance, CDTrans [51] leverages the cross-attention mechanism in Transformer for cross-domain feature alignment. TVT [52] introduces the evaluated transferabilities into the Multi-head Self-Attention module to construct a transferable ViT. SSRT [41] proposes to perturb the target features to refine the ViT and designs a safe training mechanism.

Despite remarkable progresses, most existing UDA methods only operate in the vision modality, discarding the rich semantics behind features and categories, hindering effective adaptation for complex and large domain gaps.

Vision-Language Models and Prompt Learning. Recent large-scale pre-trained VLMs have shown impressive performance on various vision-and-language tasks [53, 54]. VLMs like CLIP [34] and ALIGN [13] learn joint representations of images and texts by pre-training on large amounts of image-text pairs. A key capability of VLMs is the zero-shot prediction, where the pre-trained model can be applied to downstream tasks by simply conditioning on a textual prompt like "a photo of a [CLS]". This avoids costly fine-tuning and preserves the original knowledge in VLMs. However, manually designing effective prompts can be challenging. Prompt learning has thus become a popular VLMs adaptation technique. CoOp [61] first uses learnable context tokens to prompt the language encoder of CLIP for visual classification. Later, CoCoOp [60] learns instance-conditioned prompts with a two-layer network for more generalizable textual prompts. MaPLe [17] introduces multi-modal prompts to fine-tune both modalities. Neverth-

less, these works do not consider the domain shift problem.

To leverage VLMs and prompt learning for UDA, DAPrompt [8] introduces a set of domain-specific textual tokens to encode domain semantics and perform classification with target-specific prompts. AD-CLIP [38] learns both domain- and image-specific tokens with feature statistics in the vision backbone. However, learning prompts for different domains separately may limit cross-domain knowledge transfer. Besides, prompting in a single modality cannot fully adapt the multi-modal knowledge in VLMs.

3. Proposed Method

Let $\mathcal{D}_s = \{x_i^s, y_i^s\}_{i=1}^{N_s}$ be the source domain with N_s labeled samples, where $x_s^i \sim P_s(X)$ is the input and $y_s^i \in Y$ is the label. Meanwhile, we have a target domain $\mathcal{D}_t = \{x_i^t\}_{i=1}^{N_t}$ with N_t unlabeled samples and $x_t \sim P_t(X)$. These two domains are assumed to have different distributions in the data space X, but share the same label (semantic) set Y. The goal of UDA is to learn a model $f : X \to Y$ with $\mathcal{D}_s$ and $\mathcal{D}_t$ that can perform well on the target domain.

3.1. Domain-Agnostic Prompting with CLIP

Traditional UDA methods typically implement f as a unimodal neural network and associate each category with a numerical label, which overlook the rich semantics that could inform classification. In this work, we leverage CLIP [34] to enable semantic-driven classification.

CLIP learns aligned visual and textual representations by pre-training an image encoder f_v and a text encoder f_s on a large dataset of image-text pairs. Specifically, f_v can be a ResNet [9] or ViT [4] backbone that extracts a visual embedding from an image. On the other hand, f_s uses a Transformer [47] to encode the paired textual description into a compact embedding. The two encoders are trained jointly with a contrastive loss. The aligned joint space then allows zero-shot classification for arbitrary input $\boldsymbol{x}$ [2] by comparing the visual embedding $f_v(\boldsymbol{x})$ to textual embeddings $\{f_s(\boldsymbol{t}_k)\}_{k=1}^K$ correspond to K classes in the joint space:

$$P(\boldsymbol{y} = k \mid \boldsymbol{x}) = \frac{\exp\left(cos\left(f_v(\boldsymbol{x}), f_s\left(\boldsymbol{t}_k\right)\right)/\tau\right)}{\sum_{k=1}^K \exp\left(cos\left(f_v(\boldsymbol{x}), f_t\left(\boldsymbol{t}_k\right)\right)/\tau\right)}, \tag{1}$$

where $\tau = 0.01$ is the temperature coefficient learned by CLIP, $cos(\cdot,\cdot)$ denotes the cosine similarity, and $\boldsymbol{t}_k$ is the textual prompt of the k-th class, e.g., "a photo of a [CLS]".

However, manually designed prompts can be naive and sub-optimal. A more effective way is to make $\{\boldsymbol{t}_k\}_{k=1}^K$ learnable as in CoOp [61]. In UDA, covariate shift [31] is a widely adopted assumption, which indicates that the marginal distributions differ (i.e., $P_s(X) \neq P_t(X)$) but the conditional distribution $P(Y|X)$ remains unchanged between domains. This motivates using a shared set of textual prompts to model the invariant $P(Y|X)$. In this work, we use a domain- and class-shared input prompt template:

$$\boldsymbol{t}_k := [\boldsymbol{p}_1][\boldsymbol{p}_2]\dots[\boldsymbol{p}_N][\text{CLS}_k], \tag{2}$$

where $\boldsymbol{p}_{1:N} \in \mathbb{R}^{N\times D}$ are learnable contexts with length N and dimension D, and $[\text{CLS}_k]$ is the k-th class name.

3.2. Mutual Prompt Learning with Cross-Attention

Directly learning domain-agnostic prompts as in Eq. (2) can be challenging in UDA. First, f_v is pre-trained without domain adaptation objectives, yielding domain-biased visual embeddings that conform to different distributions across domains [8]. Second, the instance diversity leads to large intra-class variation, making it difficult to align all samples to a class-level textual prompt. To address these issues, we propose to impose visual prompts on f_v to elicit more domain-agnostic visual representations. Meanwhile, we also adjust the textual prompt on f_s according to each image contextual information for better image-text paired alignment, like in the original CLIP pre-training.

Language-Guided Visual Prompting. As $\{\boldsymbol{t}_k\}_{i=1}^K$ encode domain-agnostic class semantics, we can exploit these semantics to guide the generation of visual prompts that elicit domain-invariant visual characteristics. To achieve this, we use the cross-attention [47] mechanism to pass information between the two branches, which has shown great success in modeling multimodal interactions [11, 44].

Given a textual prompt $\boldsymbol{t}_k$, the class name $[\text{CLS}_k]$ is first tokenized and embedded into $\boldsymbol{r}_k \in \mathbb{R}^{L_k\times D}$, where L_k is the name length. The text encoder f_s then extract embeddings via J Transformer encoder layers $\{\texttt{Enc}_j\}_{j=1}^J$:

$$\begin{aligned} T_j^k &= \texttt{Enc}_j([\boldsymbol{p}_{1:N}, \boldsymbol{r}_k]) \quad j = 1. \\ T_j^k &= \texttt{Enc}_j(T_{j-1}) \quad j = 2, 3, \dots, J. \end{aligned} \tag{3}$$

Here $[\cdot,\cdot]$ stands for concatenation, $T_j^k \in \mathbb{R}^{(N+L_k)\times D}$ is the extracted embeddings in layer j. CLIP only uses the embedding at the last position of layer J as the textual embedding, denoted as $\boldsymbol{s}_k$. However, we argue that embeddings at other positions also encode rich contextual information due to shared parameters among them. Therefore, we use the first N embeddings of T_J^k, denoted as $\widetilde{\boldsymbol{s}} = T_J^k[1:N] \in \mathbb{R}^{N\times D}$, to guide the generation of visual prompts, which generally encode domain- and class-agnostic semantics.

For visual prompting, there are two widely used forms in the community, i.e., the pixel-level prompts [1, 6] and token-level prompts [14, 56]. These two kinds of *pre-model* prompting poses challenges to prompt different vision backbones in a unified manner. In this work, we adopt the *post-model* prompting [35] strategy to prompt f_v in the embedding space. Specifically, we first obtain the visual embedding $\boldsymbol{v} = f_v(\boldsymbol{x}) \in \mathbb{R}^D$ for an input $\boldsymbol{x}$, and aggregate information from text contexts $\widetilde{\boldsymbol{s}}$ via a cross-attention-based

[2] We use bold font $\boldsymbol{x}$ to denote either a source or a target sample.

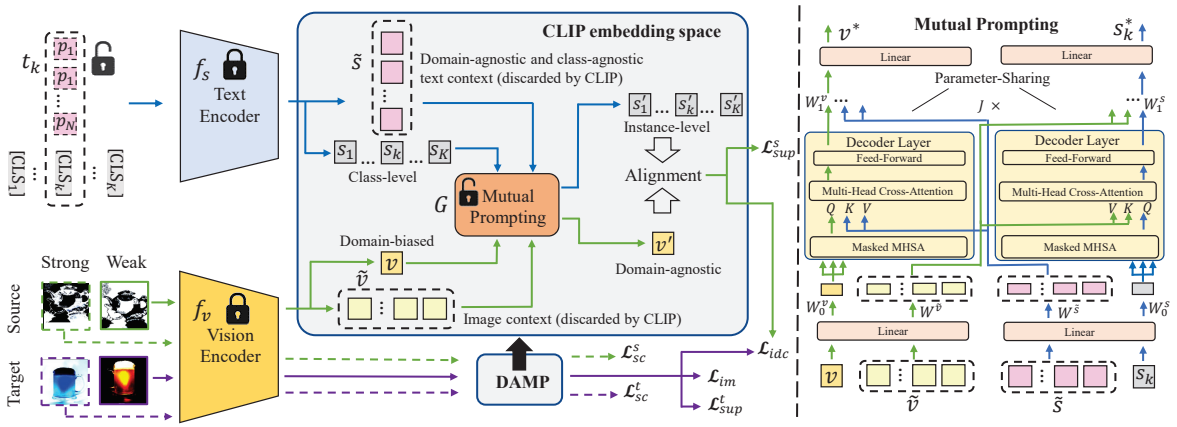


Figure 2. Overview of the proposed DAMP framework. Parameters of f_s and f_v are frozen and only $\boldsymbol{p}_{1:N}$ and G are tunable during training. The blue arrows represent text data flows, while the green and purple arrows are data flows for source and target images, respectively. We only depict the prompting process for source weakly augmentated samples. All other samples follow the same process. $\mathcal{L}^s_{sc}$ ($\mathcal{L}^t_{sc}$), $\mathcal{L}^s_{idc}$ ($\mathcal{L}^t_{idc}$), and $\mathcal{L}_{im}$ are regularizations to make the prompting domain-agnostic, instance-conditioned and semantic-compatible, respectively.

module G with L Transformer decoder layers $\{\texttt{Dec}_l\}_{l=1}^L$:

$$\begin{aligned}[W_0^v, W^{\widetilde{s}}] &= \texttt{InProj}([\boldsymbol{v},\ \widetilde{\boldsymbol{s}}]),\\ W_l^v &= \texttt{Dec}_l(W_{l-1}^v, W^{\widetilde{s}}) \quad l = 1, 2, \ldots, L,\\ \boldsymbol{v}^* &= \texttt{OutProj}(W_L^v).\end{aligned} \tag{4}$$

Here $\texttt{InProj}$ and $\texttt{OutProj}$ are two projection operations, and each token is projected independently. We then obtain the final embedding $\boldsymbol{v}'$ via a residual connection: $\boldsymbol{v}' = \boldsymbol{v} + \gamma_v \boldsymbol{v}^*$, where γ_v controls the weight. This produces visual embeddings $\boldsymbol{v}'$ guided by the domain-agnostic texts.

Vision-Guided Language Prompting. To accommodate various visual backbones, CLIP uses a modified version of ResNet to implement f_v by replacing the last Global Average Pooling (GAP) layer with an attention pooling layer. Specifically, it first transforms an image $\boldsymbol{x} \in \mathbb{R}^{H\times W\times 3}$ into a feature map $\boldsymbol{z} \in \mathbb{R}^{\hat{H}\times\hat{W}\times C}$, where $H(\hat{H}), W(\hat{W})$ and C are the height, width and the channel number. The original ResNet uses $\overline{\boldsymbol{z}} = \texttt{GAP}(\boldsymbol{z}) \in \mathbb{R}^C$ as the final visual embedding. In CLIP, $\boldsymbol{z}$ and $\overline{\boldsymbol{z}}$ are further handled by a Multi-Head Self-Attention (MHSA) layer:

$$\boldsymbol{v}, \widetilde{\boldsymbol{v}} = \texttt{MHSA}([\overline{\boldsymbol{z}}, \boldsymbol{z}]), \tag{5}$$

where $\boldsymbol{v} \in \mathbb{R}^{1\times D}$ and $\widetilde{\boldsymbol{v}} \in \mathbb{R}^{\hat{H}\hat{W}\times D}$ are the embeddings at the class token and other spatial positions, respectively, which are consistent with the ones in ViT. Generally, CLIP only uses $\boldsymbol{v}$ as the visual embedding and discards $\widetilde{\boldsymbol{v}}$. However, $\widetilde{\boldsymbol{v}}$ can also preserve useful semantical and spatial information that can be used as contextual information [35]. In this work, we leverage $\widetilde{\boldsymbol{v}}$ to adjust the textual embeddings $\{\boldsymbol{s}_k\}_{k=1}^K$ via the same *post-model* prompting strategy and G. Specifically, for the k-th class,

$$\begin{aligned}[W_0^s, W^{\widetilde{v}}] &= \texttt{InProj}([\boldsymbol{s}_k, \widetilde{\boldsymbol{v}}]),\\ W_l^s &= \texttt{Dec}_l(W_{l-1}^s, W^{\widetilde{v}}) \quad l = 1, 2, \ldots, L,\\ \boldsymbol{s}_k^* &= \texttt{OutProj}(W_L^s).\end{aligned} \tag{6}$$

The final semantic embedding $\boldsymbol{s}'_k$ is then obtained by: $\boldsymbol{s}'_k = \boldsymbol{s}_k + \gamma_s \boldsymbol{s}_k^*$, where γ_s is weight coefficient for the text modality. Note that each $\boldsymbol{s}'_k$ is updated based on a specific $\boldsymbol{x}$, making it instance-dependent, enabling better image-text alignment. These two branches of prompting are guided from each other to ensure mutual synergy. As a result, we use

$$\hat{P}(\boldsymbol{y} = k \mid \boldsymbol{x}) = \frac{\exp\left(cos(\boldsymbol{v}', \boldsymbol{s}'_k)/\tau\right)}{\sum_{k=1}^K \exp\left(cos(\boldsymbol{v}', \boldsymbol{s}'_k)/\tau\right)} \tag{7}$$

in our method for classification in the CLIP space.

3.3. Auxiliary Regularizations

While the mutual prompting framework aims to generate domain-invariant visual embeddings and instance-conditioned textual embeddings, directly optimizing with the source classification loss cannot guarantee achieving this goal. Hence, we design two auxiliary regularizations.

Instance-Discrimination Contrastive Loss. During the mutual prompting, the updated textual embeddings may still encode some domain-specific semantics from the image context, making $\{\boldsymbol{s}'_k\}_{k=1}^K$ domain-biased and less capable for the target domain. To address this problem, we design an instance-discrimination contrastive loss to prevent textual prompts from learning domain-related cues from visual contexts. Our motivation is that images from the same domain typically share the same domain-information. Therefore, maximizing the difference in $\{\boldsymbol{s}'_k\}_{k=1}^K$ among them help remove the domain-specific information.

Specifically, given a batch of source or target samples $\mathcal{B}$, denote $\{\boldsymbol{s}'_{a,k}\}_{k=1}^K$ and $\boldsymbol{v}'_a$ the textual and visual embeddings after mutual prompting for $\boldsymbol{x}_a \sim \mathcal{B}$, each $\boldsymbol{x}_a$ forms a positive pair for $\boldsymbol{v}'_a$ and $\{\boldsymbol{s}'_{a,k}\}_{k=1}^K$ and forms negative pairs for

$\{\boldsymbol{s}'_{a,k}\}_{k=1}^{K}$ and $\boldsymbol{v}'_b$ from another $\boldsymbol{x}_b$ within the same batch:

$$\begin{aligned} \mathrm{sim}(\boldsymbol{x}_a, \boldsymbol{x}_b) &= \frac{1}{K}\sum_{k=1}^{K} cos(\boldsymbol{s}'_{a,k}, \boldsymbol{v}'_b)/\tau, \\ \mathcal{L}_{idc} &= -\log \frac{\exp\left(\mathrm{sim}(\boldsymbol{x}_a, \boldsymbol{x}_a)\right)}{\mathrm{sim}(\boldsymbol{x}_a, \boldsymbol{x}_a) + \sum_{b\neq a}\mathrm{sim}(\boldsymbol{x}_a, \boldsymbol{x}_b)}. \end{aligned} \tag{8}$$

This contrastive loss forces $\{\boldsymbol{s}'_k\}_{k=1}^{K}$ to not encode domain-specific cues while retaining pure instance-specific information. Imagine that if $\{\boldsymbol{s}'_k\}_{k=1}^{K}$ contained domain-related information, they would be more similar for different images from the same domain, thus the domain-specific information can be further removed by optimizing $\mathcal{L}_{idc}$. Meanwhile, this contrastive loss can be optimized in an unsupervised way, thus providing regularizations for both domains.

Semantic-Consistency Regularization. In addition to removing domain-specific information in $\{\boldsymbol{s}'_k\}_{k=1}^{K}$, we also want to ensure the prompted visual embedding $\boldsymbol{v}'$ is domain-invariant. Inspired by FixMatch [39], we aim to exploit domain-agnostic visual characteristics with a semantic-consistency regularization. Concretely, we leverage RandAugment [3] to obtain a strongly-augmentated version of $\boldsymbol{x}$, denoted as $\mathcal{A}(\boldsymbol{x})$, and enforce it to be correctly classified. For labeled source samples $\{x_i^s, y_i^s\}_{i=1}^{N_s}$, we can directly optimize with ground-truth labels via:

$$\mathcal{L}_{sc}^{s} = -\sum_{i=1}^{N_s} \log \hat{P}(\boldsymbol{y} = y_i^s \mid \mathcal{A}(x_i^s)). \tag{9}$$

Meanwhile, we also obtain pseudo-labels $\{\hat{y_i^t}\}_{i=1}^{N_t}$ for target data $\{x_i^t\}_{i=1}^{N_t}$ and only involves confident ones for training:

$$\mathcal{L}_{sc}^{t} = -\sum_{i=1}^{N_t} \mathbb{I}\{\hat{P}(\boldsymbol{y} = \hat{y_i^t} \mid S(x_i^t)) \geq T\} \log \hat{P}(\boldsymbol{y} = \hat{y_i^t} \mid \mathcal{A}(x_i^t)), \tag{10}$$

where T is the threshold for filtering confident samples.

However, the unconfident target sampels are still not well-exploited. To make the updated target domain embeddings fit the learned semantic structure, we leverage the information maximization [12, 22] technique to regularize the unlabeled target data via an entropy-based loss:

$$\mathcal{L}_{im} = \frac{1}{N_t}\sum_{i=1}^{N_t}\sum_{k=1}^{K} p_{ti}^{c} \log p_{ti}^{c} - \sum_{k=1}^{K} \hat{p}_t^k \log \hat{p}_t^k, \tag{11}$$

where $p_{ti}^k = \hat{P}(\boldsymbol{y} = k \mid x_i^t)$ and $\hat{p}_t^k = \frac{1}{N_t}\sum_{j=1}^{N_t} p_{tj}^k$. Optimizing $\mathcal{L}_{im}$ makes predictions globally diverse and locally confident, thus avoiding category collapse and ambiguity.

3.4. Overall Training Objective

We train our DAMP with the supervised loss and above regularizations in an end-to-end manner. For the source domain, the supervised loss can be expressed by:

$$\mathcal{L}_{sup}^{s} = -\sum_{i=1}^{N_s} \log \hat{P}(\boldsymbol{y} = y_i^s \mid \mathcal{W}(x_i^s)), \tag{12}$$

where $\mathcal{W}(\cdot)$ is a weak augmentation operation. Besides, we also supervise target samples with confident pseudo-labels:

$$\mathcal{L}_{sup}^{t} = -\sum_{i=1}^{N_t} \mathbb{I}\{\hat{P}(\boldsymbol{y} = \hat{y_i^t} \mid \mathcal{W}(x_i^t)) \geq T\} \log \hat{P}(\boldsymbol{y} = \hat{y_i^t} \mid \mathcal{W}(x_i^t)). \tag{13}$$

The final training objective $\mathcal{L}_{all}$ is formulated by

$$\mathcal{L}_{all} = \mathcal{L}_{sup} + \mathcal{L}_{sc} + \lambda_c \mathcal{L}_{idc} + \lambda_i \mathcal{L}_{im}, \tag{14}$$

where λ_c and λ_i are trade-off weights, $\mathcal{L}_{sc} = \mathcal{L}_{sc}^{s} + \mathcal{L}_{sc}^{t}$ and $\mathcal{L}_{sup} = \mathcal{L}_{sup}^{s} + \mathcal{L}_{sup}^{t}$. We give equal weights to $\mathcal{L}_{sup}$ and $\mathcal{L}_{sc}$ for treating different augmentations equally. An overview of our method can be found in Fig. 2.

4. Experiments

In this section, we mainly verify the effectiveness of our method on UDA tasks. More evaluation on multi-source UDA [33] and domain generalization (DG) [20] tasks and more analytical experiments can be found in Appendix.

4.1. Experimental Setup

Datasets. We evaluate our method on three widely used UDA datasets. **Office-Home** [48] consists of images from 4 different domains: Art (Ar), Clipart (Cl), Product (Pr) and Real-World (Rw). There are 65 object categories and around 15,500 images in total. **VisDA-17** [32] contains synthetic images to real images across 12 categories. The synthetic source domain has 152,397 images generated from 3D models. The real target domain has 55,388 real images. **Mini-DomainNet** is a subset of the most challenging dataset DomainNet [33]. We use a subset with 4 domains, i.e., Clipart (Cl), Painting (Pn), Real (Rl) and Sketch (Sk), across 126 categories following previous works [38, 59].

Training Configuration. We evaluate DAMP with both ResNet-50 [9] and ViT-B/16 [4] as the visual encoder f_v. The text encoder f_s is a pretrained CLIP text encoder with depth $J = 12$. During training, we freeze these encoders and tune the input textual prompts $\boldsymbol{p}_{1:N}$ and the prompting module G. The learnable token length N is set to 32 and we use $L = 2$ Transformer decoder layers in the mutual prompting module G. For training, we use the Adam optimizer [18] with an initial learning rate of 3e-3 for all datasets, and adjust it with a cosine annealing scheduler [28]. Our model is trained for 30 epochs in total (for Mini-DomainNet, we train 500 iterations per epoch). The batch size is set to 32 for each domain. We set the confidence

Table 1. Classification accuracies (%) on **Office-Home** dataset for UDA. The best and second best results within each backbone are highlighted in bold and underline, respectively. † CDTrans uses DeiT-B [43] as the backbone. Methods within each backbone are grouped into three categories, i.e., fine-tuning, zero-shot and prompt learning (from top to bottom), respectively.

Method	f_v	Ar→Cl	Ar→Pr	Ar→Rw	Cl→Ar	Cl→Pr	Cl→Rw	Pr→Ar	Pr→Cl	Pr→Rw	Rw→Ar	Rw→Cl	Rw→Pr	**Avg.**
ResNet-50 [9]	ResNet-50	34.9	50.0	58.0	37.4	41.9	46.2	38.5	31.2	60.4	53.9	41.2	59.9	46.1
SRDC [42]		52.3	76.3	81.0	69.5	76.2	78.0	68.7	53.8	81.7	76.3	57.1	85.0	71.3
ToAlign [50]		57.9	76.9	80.8	66.7	75.6	77.0	67.8	57.0	82.5	75.1	60.0	84.9	72.0
+ FixMatch + EIDCo [58]		**63.8**	80.8	82.6	71.5	80.1	80.9	72.1	**61.3**	84.5	**78.6**	**65.8**	87.1	75.8
PADCLIP [19]		57.5	84.0	83.8	**77.8**	85.5	84.7	76.3	59.2	85.4	78.1	60.2	86.7	76.6
CLIP [34]		51.6	81.9	82.6	71.9	81.9	82.6	71.9	51.6	82.6	71.9	51.6	81.9	72.0
DAPrompt [8]		54.1	84.3	84.8	74.4	83.7	85.0	74.5	54.6	84.8	75.2	54.7	83.8	74.5
AD-CLIP [38]		55.4	85.2	85.6	76.1	85.8	86.2	**76.7**	56.1	85.4	76.8	56.1	85.5	75.9
DAMP (Ours)		59.7	**88.5**	**86.8**	76.6	**88.9**	**87.0**	76.3	59.6	**87.1**	77.0	61.0	**89.9**	**78.2**
ViT-B [4]	ViT-B/16	54.7	83.0	87.2	77.3	83.4	85.5	74.4	50.9	87.2	79.6	53.8	88.8	75.5
CDTrans † [51]		68.8	85.0	86.9	81.5	87.1	87.3	79.6	63.3	88.2	82.0	66.0	90.6	80.5
TVT-B [52]		74.9	86.8	89.5	82.8	88.0	88.3	79.8	71.9	90.1	85.5	74.6	90.6	83.6
SSRT-B [41]		75.2	89.0	91.1	85.1	88.3	90.0	85.0	74.2	91.3	85.7	78.6	91.8	85.4
+ FixMatch + EIDCo [58]		**76.9**	90.3	91.3	86.5	90.5	90.0	86.3	75.5	91.7	**88.1**	77.1	92.3	86.4
PADCLIP [19]		76.4	90.6	90.8	**86.7**	92.3	92.0	86.0	74.5	91.5	86.9	**79.1**	93.1	86.7
CLIP [34]		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
DAPrompt [8]		70.7	91.0	90.9	85.2	91.0	91.0	85.1	70.7	90.9	85.3	70.4	91.4	84.4
AD-CLIP [38]		70.9	92.5	**92.1**	85.4	92.4	**92.5**	**86.7**	74.3	**93.0**	86.9	72.6	93.8	86.1
DAMP (Ours)		75.7	**94.2**	92.0	86.3	**94.2**	91.9	86.2	**76.3**	92.4	86.1	75.6	**94.0**	**87.1**

Table 2. Per-class accuracies (%) on **VisDA-17** dataset for UDA. Marks and symbols share the same meaning in Table 1.

Method	f_v	plane	bicycle	bus	car	horse	knife	mcycl	person	plant	sktbrd	train	truck	Avg
RN-101 [9]	ResNet-101	55.1	53.3	61.9	59.1	80.6	17.9	79.7	31.2	81.0	26.5	73.5	8.5	52.4
CGDM [5]		93.4	82.7	73.2	68.4	92.9	94.5	88.7	82.1	93.4	82.5	86.8	49.2	82.3
CAN [16]		97.0	87.2	82.5	74.3	97.8	**96.2**	90.8	80.7	**96.6**	**96.3**	87.5	59.9	87.2
PADCLIP [19]		96.7	88.8	87.0	**82.8**	97.1	93.0	91.3	83.0	95.5	91.8	91.5	63.0	**88.5**
CLIP [34]		98.2	83.9	90.5	73.5	97.2	84.0	95.3	65.7	79.4	89.9	91.8	63.3	84.4
DAPrompt [8]		97.8	83.1	88.8	77.9	97.4	91.5	94.2	79.7	88.6	89.3	**92.5**	62.0	86.9
AD-CLIP [38]		**98.1**	83.6	**91.2**	76.6	**98.1**	93.4	**96.0**	81.4	86.4	91.5	92.1	64.2	87.7
DAMP (Ours)		97.3	**91.6**	89.1	76.4	97.5	94.0	92.3	**84.5**	91.2	88.1	91.2	**67.0**	88.4
ViT-B [4]	ViT-B/16	99.1	60.7	70.6	82.7	96.5	73.1	**97.1**	19.7	64.5	94.7	**97.2**	15.4	72.6
CDTrans † [51]		97.1	90.5	82.4	77.5	96.6	96.1	93.6	**88.6**	**97.9**	86.9	90.3	62.8	88.4
TVT-B [52]		92.9	85.6	77.5	60.5	93.6	98.2	89.4	76.4	93.6	92.0	91.7	55.7	83.9
SSRT-B [41]		98.9	87.6	89.1	84.8	98.3	**98.7**	96.3	81.1	94.9	**97.9**	94.5	43.1	88.8
PADCLIP [19]		98.1	**93.8**	87.1	**85.5**	98.0	96.0	94.4	86.0	94.9	93.3	93.5	**70.2**	**90.9**
CLIP [34]		99.1	91.7	93.8	76.7	98.4	91.7	95.3	82.7	86.5	96.0	94.6	60.5	88.9
DAPrompt [8]		99.2	92.5	93.3	75.4	98.6	92.8	95.2	82.5	89.3	96.5	95.1	63.5	89.5
AD-CLIP [38]		**99.6**	92.8	**94.0**	78.6	98.8	95.4	96.8	83.9	91.5	95.8	95.5	65.7	90.7
DAMP (Ours)		98.7	92.8	91.7	80.1	**98.9**	96.9	94.9	83.2	93.9	94.9	94.8	**70.2**	**90.9**

threshold $T = 0.6$ for Office-Home and 0.5 for VisDA-17 and Mini-DomainNet. For hyperparameters, we use $\lambda_c = \lambda_i = 1.0$. Due to the modality gap in CLIP [15, 23], the visual and textual embeddings need different updating magnitudes, making manually searching γ_v and γ_s challenging. Therefore, we make them learnable parameters. More implementation details about network architectures and pseudo-labels can be found in Appendix.

4.2. Comparasion with State-of-the-Arts

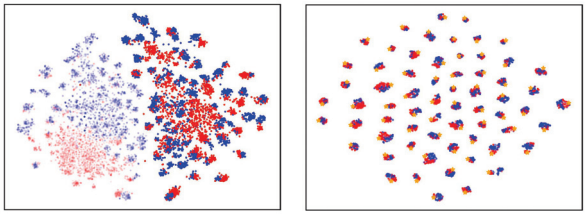
We report the results on **Office-Home** in Table 1. Our DAMP outperforms all competitors on most tasks, especially the challenging ones like Ar→Cl and Cl→Ar. Besides, DAMP brings substantial gains over strong baselines, improving the average accuracy over PADCLIP by 1.6% with ResNet-50 and 0.4% with ViT-B. Compared to prompt-based methods like DAPrompt and AD-CLIP, DAMP also shows superiority by mutually aligning both modalities. For example, it surpasses DAPrompt by 3.7% with ResNet-50 and 3.1% with ViT-B. The improvements are more significant with ResNet-50. This indicates DAMP can better exploit ViT's intrinsic transferability while effectively prompting ResNet for inspiring improvements.

For **VisDA-17** (Table 2), DAMP demonstrates competitive performance compared to other methods. It achieves 88.4% average accuracy with ResNet-101. Although it is slightly worse than the best competitor PADCLIP (88.5%), we show in Sec. 4.3 that our method involves much less learnable parameters. When using ViT-B, DAMP obtains the highest average accuracy (90.9%), comparable to PADCLIP. The results validate the consistently strong performance of DAMP across different vision backbones.

As shown in Table 3, DAMP sets new state-of-the-art on **Mini-DomainNet**. It brings significant improvements

Table 3. Classification accuracies (%) on **Mini-DomainNet** dataset for UDA. Marks and symbols share the same meaning in Table 1.

Method	f_v	Cl→Pn	Cl→Rl	Cl→Sk	Pn→Cl	Pn→Rl	Pn→Sk	Rl→Cl	Rl→Pn	Rl→Sk	Sk→Cl	Sk→Pn	Sk→Rl	Avg
ResNet-50 [9]	ResNet-50	52.1	63.0	49.4	55.9	73.0	51.1	56.8	61.0	50.0	54.0	48.9	60.3	56.3
CLIP [34]	ResNet-50	67.9	84.8	62.9	69.1	84.8	62.9	69.2	67.9	62.9	69.1	67.9	84.8	71.2
DAPrompt [8]	ResNet-50	72.4	87.6	65.9	72.7	87.6	65.6	73.2	72.4	66.2	73.8	72.9	87.8	74.8
AD-CLIP [38]	ResNet-50	71.7	88.1	66.0	73.2	86.9	65.2	73.6	73.0	68.4	72.3	74.2	**89.3**	75.2
DAMP (Ours)	ResNet-50	**76.7**	**88.5**	**71.7**	**74.2**	**88.7**	**70.8**	**74.4**	**75.7**	**70.5**	**74.9**	**76.1**	88.2	**77.5**
ViT-B [4]	ViT-B/16	63.3	79.0	56.4	62.6	83.3	55.4	62.0	70.3	53.5	63.0	63.6	75.8	65.7
CLIP [34]	ViT-B/16	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
DAPrompt [8]	ViT-B/16	83.3	92.4	81.1	86.4	92.1	81.0	86.7	83.3	80.8	86.8	83.5	91.9	85.8
AD-CLIP [38]	ViT-B/16	84.3	**93.7**	82.4	**87.5**	**93.5**	82.4	**87.3**	84.5	81.6	**87.9**	84.8	93.0	86.9
DAMP (Ours)	ViT-B/16	**86.4**	93.3	**83.5**	87.2	93.4	**84.1**	87.2	**86.5**	**82.5**	87.3	**86.6**	**93.4**	**87.6**



(a) Visual embeddings (b) Textual embeddings

Figure 3. Visualization of (a) visual embeddings and (b) textual embeddings using t-SNE [46] on task Ar → Pr (Office-Home). Light and dark colors represent embeddings before and after our mutual prompting, respectively. Red and blue points are source and target samples, respectively. Orange stars denote the class-level domain-agnostic textual embeddings $\{\boldsymbol{s}_k\}_{k=1}^{K}$.

over baseline methods like standard CLIP and other prompt learning methods like DAPrompt and AD-CLIP. For example, with ResNet-50, DAMP improves over CLIP by 6.3%, DAPrompt by 2.7%, and AD-CLIP by 2.3%. Similar gains are observed when using ViT backbone. In addition, it achieves especially large gains on challenging tasks like Cl→Pn, Cl→Sk and Sk→Pn. The consistent and substantial improvements of DAMP over strong baselines highlight the benefits of mutually aligning textual and visual prompts in a domain-agnostic and instance-conditioned manner.

4.3. Analytical Experiments

Visualization of Embeddings. Fig. 3 visualizes the visual and textual embeddings learned by DAMP. In Fig. 3a, we see that before visual prompting, the source and target domains form distinct distributions, indicating a large domain gap. After prompting, the visual features become better aligned across domains and form more clear clusters, suggesting a reduced domain gap and a discriminative semantic structure. Fig. 3b shows that instance-conditioned textual prompting increases within-class semantic diversity. This enables better pairing of text and images in both domains. Through mutual alignment of visual and textual embeddings, the two prompts make representations more domain-invariant to facilitate cross-domain knowledge transfer.

Model Capacity Analysis. Fig. 4 compares different UDA methods regarding the number of tunable parameters versus accuracy. Traditional methods like DANN and CDAN fine-tune the full model, requiring extensive parameters yet achieving relatively low accuracy. PADCLIP [19] fine-tunes the CLIP visual backbone, introducing many parameters. In contrast, our DAMP only tunes the textual prompts $\boldsymbol{p}_{1:N}$ and the prompting module G, which just adds a few parameters over DAPrompt yet achieves comparable accuracy (88.4%) to PADCLIP.

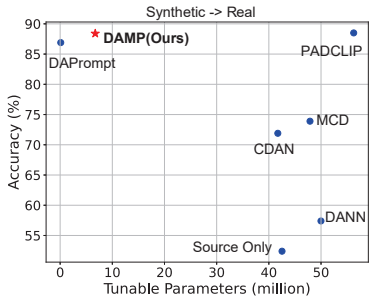
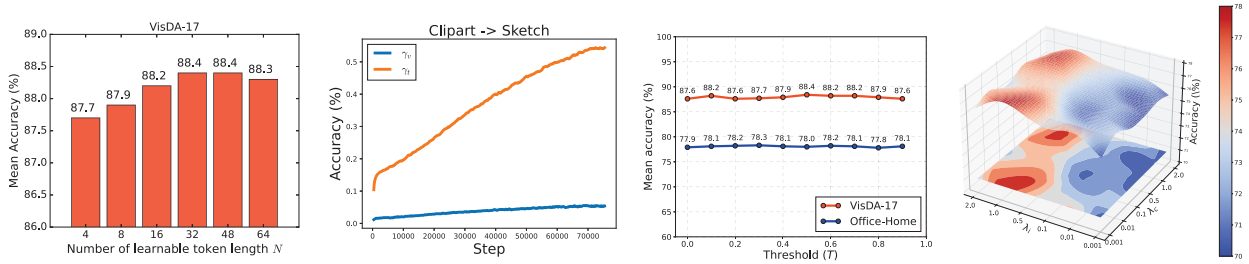


Figure 4. Comparasion between different UDA methods regarding tunable parameters and accuracies on VisDA-17 (ResNet-101). DAMP only use 11.9% parameters compared with PADCLIP.

Table 4. Comparasion between different prompting strategies on Office-Home (ViT-B/16). MP denotes mutual prompting.

Prompting Strategy	Office-Home
w/o MP (CoOp)	85.0
Independent prompting	85.4
MP w/ simple synergy	85.9
MP w/ cross-attention	**86.1**

Analysis of the Mutual Prompting Strategy. To explore the effectiveness of the cross-attention-based mutual prompting, we compare it to other prompting strategies in Table 4. For fair comparison, all regularizations are removed and only $\mathcal{L}_{sup}$ is optimized. Specifically, we evaluate an independent prompting strategy by replacing the cross-attention with self-attention in G, which results in separate prompting modules for each modality without any interaction. It shows that without mutual synergy, the improvement is limited over the baseline CoOp. We also examine a simple synergy strategy inspired by MaPLe [17], where we use a linear projection layer to obtain the prompted textual embedding $\boldsymbol{s}_k^*$ from the image context



(a) N (ResNet-101) (b) γ_v and γ_t (ResNet-50) (c) T (ResNet-50) (d) λ_c and λ_i (ResNet-50)

Figure 5. Hyperparameter analysis. (a) Performance under different learnable token length N on VisDA-17 dataset. (b) Values of learnable hyperparameters γ_v and γ_t during training on task Cl $\rightarrow$ Sk (Mini-DomainNet). (c) The influence of different choices of T on Office-Home dataset. (d) parameter sensitivities of λ_c and λ_i on task Cl $\rightarrow$ Ar (Office-Home).

Table 5. Ablation study on Office-Home (ResNet-50) and VisDA-17 (ResNet-101). ITP and VP refer to instance-level textual prompting and visual prompting, respectively.

Method	Prompting		Loss			Mean Accuracy	
	ITP	VP	$\mathcal{L}_{sc}$	$\mathcal{L}_{idc}$	$\mathcal{L}_{im}$	Office-Home	VisDA-17
Baseline (CoOp)	✗	✗	✗	✗	✗	75.3	86.1
Uni-modal Prompting	✗	✓	✗	✗	✗	75.7	86.5
	✓	✗	✗	✗	✗	75.8	87.1
Mutual Prompting	✓	✓	✗	✗	✗	76.1	87.2
	✓	✓	✓	✗	✗	76.9	87.8
	✓	✓	✓	✓	✗	77.3	88.0
	✓	✓	✓	✓	✓	78.2	88.4

embedding $\widetilde{\boldsymbol{v}}$ (and vice versa for the visual embedding). However, this uni-directional projection does not fully capture the complex interaction between the modalities in the prompting process. In contrast, our cross-attention module allows bi-directional interaction, which enables more effective fusion of information from both modalities, resulting in better domain-agnostic and instance-specific prompts.

Hyperparameter Analysis. As shown in Fig. 5a, the performance on VisDA-17 improves as N increases from 4 to 32, and then remains relatively stable with larger N. This indicates that a moderate length is enough for encoding rich semantic information. However, further increasing N does not bring substantial gains. Fig. 5b plots the values of the learnable weight coefficients γ_v and γ_s during training. We can see that γ_v converges to a relatively small value around 0.1, while γ_s converges to a larger value around 0.5. This aligns with the intuition that a smaller perturbation is needed on the visual embeddings compared to the textual embeddings, due to the inherent modality gap between vision and language in CLIP. The learnable nature of γ_v and γ_s provides the flexibility to adapt the updating magnitudes for both modalities. Fig. 5c shows that the performance is relatively stable across different choices of T from 0.5 to 1.0. Setting T too small (0.1) deteriorates the performance. This indicates that the confidence threshold is not too sensitive, while using very unconfident pseudo-labels can hurt the performance. A moderate threshold between 0.5 and 1.0 works reliably. Fig. 5d studies λ_c and λ_i on task Cl$\rightarrow$Ar, which shows DAMP is relatively robust to different choices of λ_c, and a little sensitive to the variation of λ_c. Overally, trivially setting both of them to 1.0 offers a good trade-off.

Ablation Study. We conduct an ablation study on Office-Home and VisDA-17 datasets in Table 5. We first evaluate a baseline model that directly optimizes the supervised loss $\mathcal{L}_{sup}$ on source and confident target samples to fine-tune $\boldsymbol{p}_{1:N}$ without further prompting (analogous to CoOp), which gives the lowest performance. Adding either instance-level textual prompting (ITP) or visual prompting (VP) brings gains over the baseline, showing the benefits of adapting either modality with prompting. Further prompting both modalities together with the mutual prompting framework leads to additional performance boosts, which mainly benefits from the flexibility to adapt both language and vision branches to the target domain. Finally, regularizations $\mathcal{L}_{sc}$, $\mathcal{L}_{idc}$ and $\mathcal{L}_{im}$ all contribute to the superior performance in a collaborative manner. The step-wise improvements support the rationality of our design.

5. Conclusion

We propose DAMP, a novel framework to address UDA using VLMs. DAMP mutually aligns the visual and textual modalities via prompting to elicit domain-agnostic embeddings. The prompts are optimized together through cross-attention and regularized with elaborate losses. Extensive experiments validate that DAMP brings substantial and consistent improvements over strong baselines on three benchmarks. DAMP provides an effective approach to harness both source and pre-trained VLMs knowledge for UDA.

Acknowledgement

This work was supported in part by the National Natural Science Foundation of China under Grant 62173066, 62273071, 62176042, and in part by Sichuan Science and Technology Program under Grant 2023NSFSC0483, and in part by Tencent Marketing Solution Rhino-Bird Focused Research Program.

References

[1] Hyojin Bahng, Ali Jahanian, Swami Sankaranarayanan, and Phillip Isola. Exploring visual prompts for adapting large-scale models. *arXiv preprint arXiv:2203.17274*, 2022. 3

[2] Xinyang Chen, Sinan Wang, Mingsheng Long, and Jianmin Wang. Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation. In *ICML*, pages 1081–1090. PMLR, 2019. 1

[3] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *CVPR workshops*, pages 702–703, 2020. 5

[4] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 2, 3, 5, 6, 7

[5] Zhekai Du, Jingjing Li, Hongzu Su, Lei Zhu, and Ke Lu. Cross-domain gradient discrepancy minimization for unsupervised domain adaptation. In *CVPR*, pages 3937–3946, 2021. 1, 6

[6] Yulu Gan, Yan Bai, Yihang Lou, Xianzheng Ma, Renrui Zhang, Nian Shi, and Lin Luo. Decorate the newcomers: Visual domain prompt for continual test time adaptation. In *AAAI*, pages 7595–7603, 2023. 3

[7] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *ICML*, pages 1180–1189. PMLR, 2015. 1, 2

[8] Chunjiang Ge, Rui Huang, Mixue Xie, Zihang Lai, Shiji Song, Shuang Li, and Gao Huang. Domain adaptation via prompt learning. *TNNLS*, pages 1–11, 2023. 1, 2, 3, 6, 7

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 3, 5, 6, 7

[10] Wenbin He, Suphanut Jamonnak, Liang Gou, and Liu Ren. Clip-s4: Language-guided self-supervised semantic segmentation. In *CVPR*, pages 11207–11216, 2023. 1

[11] Ronghang Hu and Amanpreet Singh. Unit: Multimodal multitask learning with a unified transformer. In *ICCV*, pages 1439–1449, 2021. 3

[12] Weihua Hu, Takeru Miyato, Seiya Tokui, Eiichi Matsumoto, and Masashi Sugiyama. Learning discrete representations via information maximizing self-augmented training. In *ICML*, pages 1558–1567. PMLR, 2017. 5

[13] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, pages 4904–4916. PMLR, 2021. 1, 2

[14] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *ECCV*, pages 709–727. Springer, 2022. 2, 3

[15] Qian Jiang, Changyou Chen, Han Zhao, Liqun Chen, Qing Ping, Son Dinh Tran, Yi Xu, Belinda Zeng, and Trishul Chilimbi. Understanding and constructing latent modality structures in multi-modal representation learning. In *CVPR*, pages 7661–7671, 2023. 6

[16] Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. Contrastive adaptation network for unsupervised domain adaptation. In *CVPR*, pages 4893–4902, 2019. 1, 6

[17] Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Maple: Multi-modal prompt learning. In *CVPR*, pages 19113–19122, 2023. 2, 7

[18] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[19] Zhengfeng Lai, Noranart Vesdapunt, Ning Zhou, Jun Wu, Cong Phuoc Huynh, Xuelu Li, Kah Kuen Fu, and Chen-Nee Chuah. Padclip: Pseudo-labeling with adaptive debiasing in clip for unsupervised domain adaptation. In *ICCV*, pages 16155–16165, 2023. 2, 6, 7

[20] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *ICCV*, pages 5542–5550, 2017. 5

[21] Jingjing Li, Erpeng Chen, Zhengming Ding, Lei Zhu, Ke Lu, and Heng Tao Shen. Maximum density divergence for domain adaptation. *TPAMI*, 43(11):3918–3930, 2020. 1

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

[23] Victor Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Y Zou. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. In *NeurIPS*, pages 17612–17625, 2022. 6

[24] Hong Liu, Jianmin Wang, and Mingsheng Long. Cycle self-training for domain adaptation. In *NeurIPS*, pages 22968–22981, 2021. 2

[25] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In *ICML*, pages 97–105. PMLR, 2015. 1

[26] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Deep transfer learning with joint adaptation networks. In *ICML*, pages 2208–2217. PMLR, 2017. 1, 2

[27] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. In *NeurIPS*, 2018. 1

[28] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 5

[29] Ke Mei, Chuang Zhu, Jiaqi Zou, and Shanghang Zhang. Instance adaptive self-training for unsupervised domain adaptation. In *ECCV*, pages 415–430. Springer, 2020. 2

[30] Raha Moraffah, Kai Shu, Adrienne Raglin, and Huan Liu. Deep causal representation learning for unsupervised domain adaptation. *arXiv preprint arXiv:1910.12417*, 2019. 2

[31] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *TKDE*, 22(10):1345–1359, 2009. 1, 3

[32] Xingchao Peng, Ben Usman, Neela Kaushik, Judy Hoffman, Dequan Wang, and Kate Saenko. Visda: The visual domain

adaptation challenge. *arXiv preprint arXiv:1710.06924*, 2017. 5

[33] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *ICCV*, pages 1406–1415, 2019. 1, 2, 5

[34] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763. PMLR, 2021. 1, 2, 3, 6, 7

[35] Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. Denseclip: Language-guided dense prediction with context-aware prompting. In *CVPR*, pages 18082–18091, 2022. 1, 3, 4

[36] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *CVPR*, pages 3723–3732, 2018. 1, 2

[37] Swami Sankaranarayanan, Yogesh Balaji, Carlos D Castillo, and Rama Chellappa. Generate to adapt: Aligning domains using generative adversarial networks. In *CVPR*, pages 8503–8512, 2018. 2

[38] Mainak Singha, Harsh Pal, Ankit Jha, and Biplab Banerjee. Ad-clip: Adapting domains in prompt space using clip. In *ICCV*, pages 4355–4364, 2023. 2, 3, 5, 6, 7

[39] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *NeurIPS*, pages 596–608, 2020. 5

[40] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *ECCV*, pages 443–450. Springer, 2016. 2

[41] Tao Sun, Cheng Lu, Tianshuo Zhang, and Haibin Ling. Safe self-refinement for transformer-based domain adaptation. In *CVPR*, pages 7191–7200, 2022. 2, 6

[42] Hui Tang, Ke Chen, and Kui Jia. Unsupervised domain adaptation via structurally regularized deep clustering. In *CVPR*, pages 8725–8735, 2020. 1, 6

[43] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, pages 10347–10357. PMLR, 2021. 6

[44] Yao-Hung Hubert Tsai, Shaojie Bai, Paul Pu Liang, J Zico Kolter, Louis-Philippe Morency, and Ruslan Salakhutdinov. Multimodal transformer for unaligned multimodal language sequences. In *ACL*, page 6558, 2019. 3

[45] Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474*, 2014. 1

[46] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *JMLR*, 9(11), 2008. 7

[47] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 2, 3

[48] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *CVPR*, pages 5018–5027, 2017. 5

[49] Xudong Wang, Zhirong Wu, Long Lian, and Stella X Yu. Debiased learning from naturally imbalanced pseudo-labels. In *CVPR*, pages 14647–14657, 2022. 2

[50] Guoqiang Wei, Cuiling Lan, Wenjun Zeng, Zhizheng Zhang, and Zhibo Chen. Toalign: task-oriented alignment for unsupervised domain adaptation. In *NeurIPS*, pages 13834–13846, 2021. 6

[51] Tongkun Xu, Weihua Chen, Pichao Wang, Fan Wang, Hao Li, and Rong Jin. Cdtrans: Cross-domain transformer for unsupervised domain adaptation. *arXiv preprint arXiv:2109.06165*, 2021. 2, 6

[52] Jinyu Yang, Jingjing Liu, Ning Xu, and Junzhou Huang. Tvt: Transferable vision transformer for unsupervised domain adaptation. In *WACV*, pages 520–530, 2023. 2, 6

[53] Yuan Yao, Ao Zhang, Zhengyan Zhang, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. Cpt: Colorful prompt tuning for pre-trained vision-language models. *arXiv preprint arXiv:2109.11797*, 2021. 2

[54] Wenwen Yu, Yuliang Liu, Wei Hua, Deqiang Jiang, Bo Ren, and Xiang Bai. Turning a clip model into a scene text detector. In *CVPR*, pages 6978–6988, 2023. 1, 2

[55] Zhongqi Yue, Hanwang Zhang, and Qianru Sun. Make the u in uda matter: Invariant consistency learning for unsupervised domain adaptation. *arXiv preprint arXiv:2309.12742*, 2023. 2

[56] Yaohua Zha, Jinpeng Wang, Tao Dai, Bin Chen, Zhi Wang, and Shu-Tao Xia. Instance-aware dynamic prompt tuning for pre-trained point cloud models. *arXiv preprint arXiv:2304.07221*, 2023. 3

[57] Yabin Zhang, Hui Tang, Kui Jia, and Mingkui Tan. Domain-symmetric networks for adversarial domain adaptation. In *CVPR*, pages 5031–5040, 2019. 2

[58] Yixin Zhang, Zilei Wang, Junjie Li, Jiafan Zhuang, and Zihan Lin. Towards effective instance discrimination contrastive loss for unsupervised domain adaptation. In *ICCV*, pages 11388–11399, 2023. 1, 6

[59] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain adaptive ensemble learning. *TIP*, 30:8008–8018, 2021. 5

[60] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *CVPR*, pages 16816–16825, 2022. 2

[61] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *IJCV*, 130(9):2337–2348, 2022. 2, 3