Complex & Intelligent Systems (2023) 9:6567–6580
https://doi.org/10.1007/s40747-023-01094-4

ORIGINAL ARTICLE

Transferable adversarial masked self-distillation for unsupervised domain adaptation

Yuelong Xia[1,2] · Li-Jun Yun[1,2] · Chengfu Yang[1,2]

Received: 25 November 2022 / Accepted: 10 April 2023 / Published online: 24 May 2023

Abstract

Unsupervised domain adaptation (UDA) aims to transfer knowledge from a labeled source domain to a related unlabeled target domain. Most existing works focus on minimizing the domain discrepancy to learn global domain-invariant representation using CNN-based architecture while ignoring both transferable and discriminative local representation, e.g, pixel-level and patch-level representation. In this paper, we propose the Transferable Adversarial Masked Self-distillation based on Vision Transformer architecture to enhance the transferability of UDA, named TAMS. Specifically, TAMS jointly optimizes three objectives to learn both task-specific class-level global representation and domain-specific local representation. First, we introduce adversarial masked self-distillation objective to distill representation from a full image to the representation predicted from a masked image, which aims to learn task-specific global class-level representation. Second, we introduce masked image modeling objectives to learn local pixel-level representation. Third, we introduce an adversarial weighted cross-domain adaptation objective to capture discriminative potentials of patch tokens, which aims to learn both transferable and discriminative domain-specific patch-level representation. Extensive studies on four benchmarks and the experimental results show that our proposed method can achieve remarkable improvements compared to previous state-of-the-art UDA methods.

Keywords Unsupervised domain adaptation · Masked self-distillation · Masked image modeling · Adversarial weighted cross-domain adaptation

Introduction

Deep neural networks (DNNs) have shown remarkable achievements in a wide range of computer vision problems [1, 2]. However, the impressive success heavily relies on an amount of labeled training data and still suffers poor generalization performance to other emerging application domains because of the domain shift problem [3, 4]. To handle these problems, much unsupervised domain adaptation (UDA) methods [5, 6] are proposed to transfer knowledge from a labeled source domain to a different unlabeled target domain, which can be divided into main two categories: domain alignment methods [7–9] and adversarial learning methods [10, 11]. However, these methods mainly adopt CNN backbone (e.g., ResNet [12]) to learn class-level alignment, which is not robust to the generalized large-scale datasets (e.g., VisDA-2017 [13]) for satisfactory performance. For example, the ResNet-101 average results is only 52.4% [14] on VisDA-2017.

Recently, instead of CNN-based architecture, Vision Transformer (ViT) [15] is a more powerful backbone and has been used to improve transferable performance on UDA tasks. For example, Yang et al. [16] proposed a transferable vision transformer method to investigate the transferability of ViT. Sun et al. [17] proposed transformer-based self-refinement SSRT to refine the domain adaptation model. Xu et al. [18] explored cross-domain transformer for unsupervised domain adaptation. However, these ViT-based methods do not take full advantage of different complementary supervision information to improve UDA performance, e.g., global class-level representation, local pixel-level, and patch-level representation.

With the above discussions, we focus on solving UDA tasks from three aspects:

✉ Yuelong Xia
xyl@mail.ynu.edu.cn

1 School of Information Science and Technology, Yunnan Normal University, Kunming 650500, China

2 Engineering Research Center of Computer Vision and Intelligent Control Technology, Department of Education of Yunnan Province, Kunming 650500, China

First, from the global class-level representation aspect, masked image modeling [19], which trains the model to predict missing information from masked image patches, has recently revealed very strong representation learning performance. We follow ViT [15] to divide it into regular non-overlapping patches. As shown in Fig. 1, given an image, we randomly mask some proportion of image patches and replace them with a special mask embedding [MSK], which aims to learn local pixel-level representation at masked positions using missing information of contextual patches. In addition, a [CLS] token is a special symbol added to extract global information, which is helpful to classifier prediction. Specifically, we extend the masked self-distillation framework to integrate the vision transformer to UDA, thus [CLS] embedding is taken to learn discriminative and global task-specific class prediction for UDA.

Second, from the local pixel-level representation aspect, most of the existing UDA methods ignore another source of supervision obtained from the raw target domain images. We follow SimMIM [19] and simply predict the RGB pixel values for masked patches, which aims to learn local pixel-level representation at masked positions using missing information of contextual patches.

Third, from the local patch-level representation aspect, we aim to bridge the two sources of gap (i.e., cross-domain adaptation alignment) such as the local [MSK] features can improve the global [CLS] feature for better classification performance. So we introduce an adversarial weighted cross-domain adaptation objective to capture the discriminative potentials of patch tokens to learn both transferable and discriminative domain-specific patch-level representation.

Based on these three aspects, in this paper, we propose a novel UDA solution named TAMS (Transferable Adversarial Masked Self-distillation for UDA). As shown in Fig. 1, TAMS takes a vision transformer as the backbone network and utilizes a self-distillation framework for unsupervised domain adaptation. Different from existing ViT-based methods [16–18], TAMS jointly optimizes three key designs to take global and local representation into account, which effectively exploits the transferability of ViT to improve UDA performance.

In summary, the major contributions of this work are:

1. We present a novel masked self-distillation framework for UDA, called TAMS. Specifically, TAMS distills representation from a full image to the representation predicted from a masked image, which effectively introduces the masked self-distillation of ViT in transferring knowledge on the UDA task.
2. TAMS jointly optimizes three objectives to transfer knowledge for UDA task: (1) adversarial masked self-distillation objective to learn task-specific global class-level representation; (2) masked image modeling objective to learn local pixel-level representation; (3) adversarial weighted cross-domain adaptation objective to learn domain-specific patch-level representation.
3. Extensive experiments are conducted on widely tested benchmarks. TAMS achieves competitive performance compared to state-of-the-art methods including 94.18% on Office-31, 85.63% on Office-Home and 88.38% VisDA-2017.

Related work

Unsupervised domain adaptation Traditional machine learning assumes that training and test data are drawn from the same distribution. However, in practical application scenarios, the target data usually follows a different distribution from the training source data. Thus, transfer learning [20] has been used to transfer generalized knowledge across different domains based on different distributions, i.e., unsupervised domain adaptation (UDA), where no labels are available for the target domain. Existing UDA methods can be roughly divided into two categories: domain-level methods [6, 8, 21–23] and class-level methods [24–27]. Domain-level methods aim to align distribution between the source and target domain using different measures such as Maximum Mean Discrepancy (MMD) [7, 8, 28] and Correlation Alignment (CORAL) [29, 30]. Another line of effort was introduced on the fine-grained class-level label distribution alignment by adversarial learning [31], which focuses on learning domain-invariant representations by a feature extractor and a domain discriminator [32, 33]. Unlike coarse-grained domain-level alignment, class-level aligns each category distribution between the source and target domain data. Different from the above two methods, our methods adopt ViT to simultaneously take class-level, patch-level and pixel-level into account, which exploits fine-grained alignment on Transformer by adversarial cross-domain adaptation objective.

Vision transformer for UDA Many Vision Transformer (ViT) [15] variants have been applied successfully to various vision tasks such as image classification, object detection and segmentation, which models long-range dependencies among visual features by self-attention mechanism. To improve UDA performance, many ViT-based methods have been proposed. For example, Sun et al. [17] proposed transformer-based self-refinement SSRT to refine the domain adaptation model. Yang et al. [16] proposed a transferable vision transformer method to investigate the transferability of ViT. Xu et al. [18] explored cross-domain transformer for unsupervised domain adaptation. However, these methods neglect to take full advantage of different complementary supervision information, e.g., masked image modeling [19],

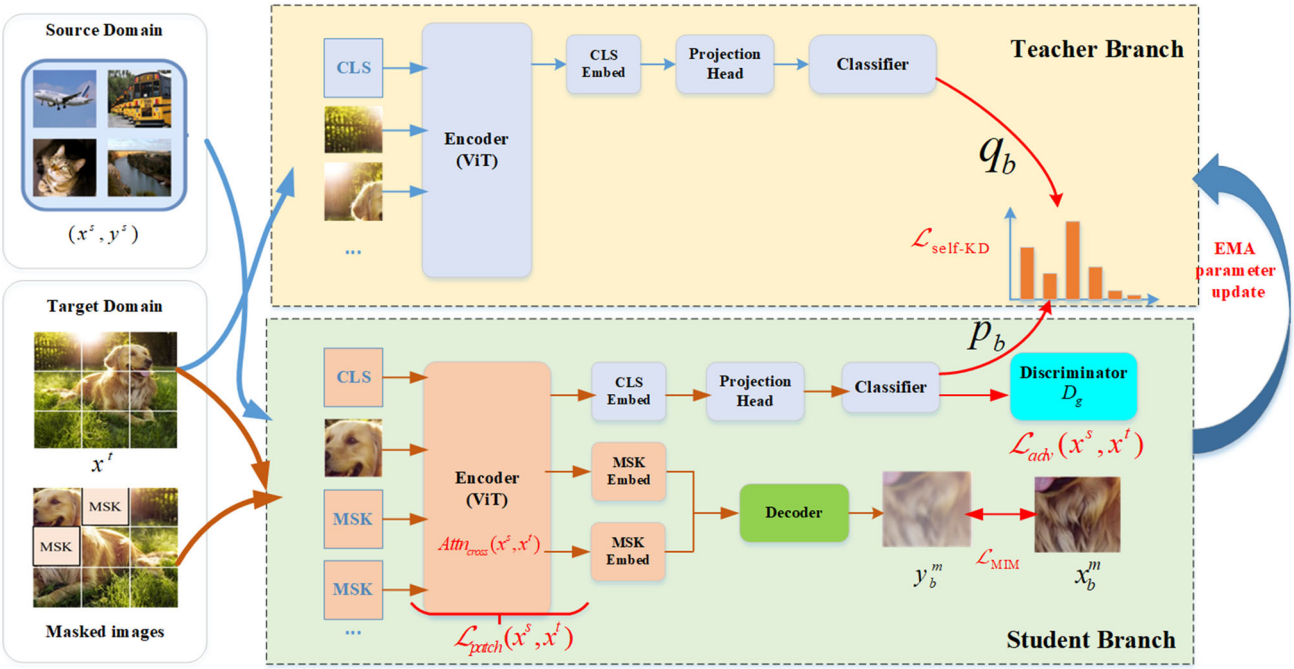


Fig. 1 The overall framework of TAMS. We introduce a masked self-distillation teacher-student network to learn the latent representation for the target domain, where the student branch consists of encoder–decoder architecture that feeds masked images, while the teacher branch contains an encoder to produce latent representation and updates weights from the student network using EMA. We randomly replace image patches with [MSK] tokens on target images, where a mask token [MSK] is taken to learn local pixel-level representation at masked positions using missing information of contextual patches while a [CLS] token is a special symbol added to extract global information. In our method, we jointly optimize three objectives to transfer knowledge for the UDA task: (1) adversarial masked self-distillation objective ("Adversarial masked self-distillation") to learn global task-specific class-level global representation; (2) masked image modeling objective ("Masked image modeling") to learn local pixel-level representation; (3) adversarial weighted cross-domain adaptation objective ("Adversarial weighted cross-domain adaptation") to learn domain-specific patch-level representation. Denote that the labeled data in the source domain are used to train the student branch, which has been omitted

which can introduce mask image modeling into UDA to improve UDA performance. We present a novel masked self-distillation framework to jointly optimize three objectives to transfer knowledge for UDA tasks.

The proposed method

We first take problem formulation in "Problem formulation", then we introduce the proposed method TAMS, where we jointly optimize three objectives to transfer knowledge for the UDA task: (1) adversarial masked self-distillation objective to learn global task-specific class-level global representation; (2) masked image modeling objective to learn local pixel-level representation; (3) adversarial cross-domain adaptation objective to learn domain-specific patch-level representation. The overall framework has been shown in Fig. 1.

Problem formulation

In UDA, there is a source domain with labeled data $\mathcal{D}_s = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$ from $\mathcal{X} \times \mathcal{Y}$ and a target domain with unlabeled data $\mathcal{D}_t = \{(x_i^t)\}_{i=1}^{n_t}$ from $\mathcal{X}$, where $\mathcal{X}$ is the input space and $\mathcal{Y}$ is the label space. We employ a ViT encoder G_f for feature learning, and a classifier G_c for classification. We can derive cross-entropy loss for the source domain:

$$\mathcal{L}_{\text{cls}}(x^s, y^s) = \frac{1}{n_s} \sum_{x_i \in \mathcal{D}_s} \mathcal{L}_{\text{ce}}(G_c(G_f(x_i^s)), y_i^s). \tag{1}$$

Adversarial masked self-distillation

Following the typical adversarial adaptation method [31], let D_g be a domain discriminator for global feature alignment, which is applied to the output of [CLS] token between the source and target domain. For adversarial domain adaptation [32, 33], G_f and D_g play a minimax game: G_f uses domain-invariant features to rival, while D_g discriminate source-domain features from target-domain. The adversarial objective can be formulated as:

$$\mathcal{L}_{\text{adv}}(x^s, x^t) = -\frac{1}{n_s + n_t} \sum_{x_i \in \mathcal{D}_s \cup \mathcal{D}_t} \mathcal{L}_{\text{ce}}(D_g(G_f(x_i^*)), y_i^d), \tag{2}$$

where $\mathcal{L}_{ce}$ is the cross-entropy loss and the superscript $*$ can be either the source domain s or the target domain t. When $y_i^d = 1$ denotes the source domain labels and $y_i^d = 0$ denotes the target domain labels.

To leverage more complimentary information, we introduce self-distillation [34, 35] to fully utilize previous knowledge to drive the model itself training. This idea also has been used to solve UDA tasks, e.g., pseudo-label learning [36] and self-ensemble learning [37]. Here, we introduce a masked self-distillation teacher–student network to produce the latent representation for the target domain, where the student branch consists of encoder–decoder architecture that feeds masked images, while the teacher branch contains an encoder to produce latent representation and updates weights $\hat{\theta}$ from the student network parameter θ using Exponential Moving Average (EMA) [38], i.e., $\hat{\theta} = \mu\hat{\theta} + (1-\mu)\theta$. Following FixMatch [39], we set μ to 0.99, and we can introduce self-distillation loss to rectify target domain labels with maximum scores above a threshold λ [37]:

$$\mathcal{L}_{\text{self-KD}}(x^t;\theta) = \frac{1}{B}\sum_{b=1}^{B}\mathbb{I}(\max q_b \geq \lambda)\mathcal{L}_{ce}(p_b, \hat{q}_b), \tag{3}$$

where p_b is the prediction of the student branch, q_b is the prediction of the teacher branch, $\hat{q}_b = \arg\max q_b$ and B is the number of batch on unlabeled target domain. In our experiments, we set λ to the 0.5. So we can derive adversarial masked self-distillation objection as

$$\mathcal{L}_{\text{mask-KD}}(x^s, x^t) = \mathcal{L}_{\text{adv}}(x^s, x^t) + \mathcal{L}_{\text{self-KD}}(x^t;\theta), \tag{4}$$

where $\mathcal{L}_{\text{mask-KD}}$ not only considers model itself complimentary information, but also can learn task-specific global class-level representation.

Masked image modeling

Although Eq. (4) can learn task-specific global class-level representation, Eq. (3) may introduce noisy pseudo-label into training. Thus we introduce another complimentary information from the raw image to learn local pixel-level representation. We follow SimMiM [19] and predict the RGB pixel values for masked patches. Specifically, given the output embedding z_b^m of the m-th [MSK] token, we first input it into a linear decode head to generate the predicted RGB values $y_b^m \in \mathcal{R}^K$ for the patch, where K denotes the number of RGB pixels per patch. Then masked image modeling objective can be formulated as

$$\mathcal{L}_{\text{MIM}}(x^t;\theta) = \frac{1}{BMK}\sum_{b=1}^{B}\sum_{m=1}^{M}\left\|y_b^m - x_b^m\right\|_1, \tag{5}$$

where M denotes the number of masked patches per image, and x_b^m denote the ground-truth RGB values. In our experiments, we also explore the influence of different masking ratios ε in "Experiments".

Adversarial weighted cross-domain adaptation

Let (H, W) denote the resolution of the original image, C is the number of channels and (P, P) denotes the resolution of each image patch. The number of patches can be computed by $N = HW/P^2$. For the self-attention of Transformer, the patches are projected into three vectors, i.e., queries $\mathbf{Q} \in \mathcal{R}^{N\times d}$, keys $\mathbf{K} \in \mathcal{R}^{N\times d}$ and values $\mathbf{V} \in \mathcal{R}^{N\times d}$, which aims to emphasize relationships among patches by computing inner product. Different from traditional self-attention, inspired by [16, 18], as shown in Fig. 2, we leverage the weighted cross-attention to learn mix-up feature representations for both source and target domains, which can be formulated as:

$$\begin{aligned}&\text{Attn}_{\text{cross}}(x^s, x^t)\\ &= \underbrace{\text{softmax}\left(\frac{Q_s K_t^T}{\sqrt{d}}\right)\times\begin{bmatrix}1\\ H(D_p(K_t^{\text{patch}}))\end{bmatrix}V_t}_{s\to t}\\ &\quad+\underbrace{\text{softmax}\left(\frac{Q_t K_s^T}{\sqrt{d}}\right)\times\begin{bmatrix}1\\ H(D_p(K_s^{\text{patch}}))\end{bmatrix}V_s}_{t\to s},\end{aligned} \tag{6}$$

where D_p is introduced for patch-level domain discriminator to match cross-domain local features. Here, instead of using masked target vector ($\mathbf{Q} \in \mathcal{R}^{N\times d}$, keys $\mathbf{K} \in \mathcal{R}^{N\times d}$ and values $\mathbf{V} \in \mathcal{R}^{N\times d}$), we use the complete target vector to prevent pixel loss and thus achieve cross attention. Using entropy $H(D_p(K^{\text{patch}}))$ to assign weights to different patches, the cross-attention in Eq. (6) not only considers semantic importance ($\text{softmax}(\frac{QK^T}{\sqrt{d}})$) but also considers the transferability of each patch token. As shown in Eq. (2), we can introduce adversarial cross-domain adaptation objective as

$$\begin{aligned}&\mathcal{L}_{\text{patch}}(x^s, x^t)\\ &= -\frac{1}{(n_s+n_t)P}\sum_{x_i\in\mathcal{D}_s\cup\mathcal{D}_t}\sum_{j=1}^{P}\mathcal{L}_{ce}(D_p(G_f(x_{ij}^*)), y_{ij}^d),\end{aligned} \tag{7}$$

where P is the number of patches. Following the adversarial learning, D_p tries to assign 1 for a source-domain patch and 0 for the target-domain patch, while G_f combats such situations. As a result, adversarial cross-domain adaptation manages to aggregate the two input images based on different weights of patches, which aligns the information from patch-level representation.

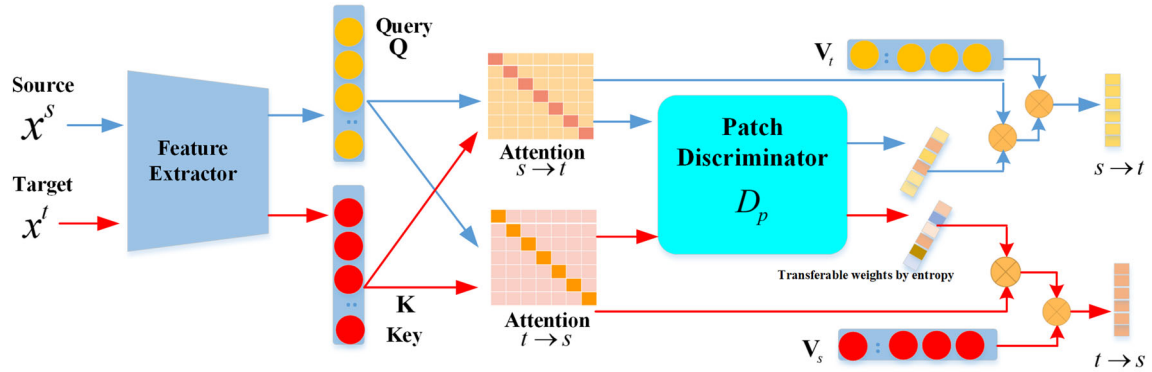


Fig. 2 The weighted cross-attention for Transformer. Using entropy $H(D_p(K^{\text{patch}}))$ to assign weights to different patches, the cross-attention mechanism (Eq. (6)) can not only capture semantic importance (softmax($\frac{QK^T}{\sqrt{d}}$)) on attention map, but also can capture the transferability of each patch token

Following the literature [16, 40, 41], we also introduce maximizing mutual information on target label distribution to avoid every target data being assigned to the same class as

$$I(p^t; x^t) = H(\bar{p}^t) - \frac{1}{n_t}\sum_{i=1}^{n_t} H(p_i^t), \tag{8}$$

where $p_i^t = \text{softmax}(G_c(G_f(x_i^t))$ and $\bar{p}^t = \mathbb{E}_{x_t}[p^t]$, by maximizing mutual information, our model can further improve the generalization to the target domain.

To summarize, the objective function of TAMS can be defined as

$$\begin{aligned}\mathcal{L}_{\text{TAMS}} = \mathcal{L}_{\text{cls}}(x^s, y^s) + \alpha\mathcal{L}_{\text{mask-KD}}(x^s, x^t) + \beta\mathcal{L}_{\text{MIM}}(x^t; \theta) \\ +\gamma\mathcal{L}_{\text{patch}}(x^s, x^t) - \delta I(p^t; x^t),\end{aligned} \tag{9}$$

where α, β, γ and δ are hyper-parameters. In our experiments, α, β and δ are set to 0.1, while γ is set to 0.01.

Experiments

In this section, we evaluate and analyze the proposed TAMS methods on benchmark datasets.

Experiment setup

Datasets To verify the effectiveness of TAMS, we conduct comprehensive experiments on benchmark datasets, including Office-31 [42], Office-Home [43] and VisDA-2017 [13]. Office-31 [42] contains 4,652 images of 31 classes from three domains: Amazon (A), DSLR (D), and Webcam (W). As shown in Table 1, 'A → W' refers that A as the source domain and W as the target domain. Office-Home [43] consists of 15,500 images of 65 classes from four domains: Artistic (Ar), Clip Art (Cl), Product (Pr), and Real-world (Rw) images. VisDA-2017 [13] is a Synthetic-to-Real dataset, which contains about 0.2 million images in 12 classes.

Baseline methods We investigate experimental comparisons against state-of-the-art UDA methods, including RevGrad [10], DDC [7], JAN [44], MinEnt [45], DAN [8], DANN [10], CDAN [46], MCD [24], SWD [47], BNM [48], DCAN [49], SHOT [14], ATDOC-NA [50], ALDA [51], TVT [16], CDTrans [18] and SSRT [17]. For a fair comparison, we use the original results on paper. We also compare our method with different backbones such as ResNet-50 and ViT.

Implementation details In our experiments, we use the ViT-small (ViT-S) and ViT-base (ViT-B) with 16 × 16 patch size [1], pre-trained on ImageNet, as the vision transformer backbones, which contain 12 transformer layers in the encoder. ViT-S (22 M parameters) is a comparable model with ResNet-50. In our experiments, the 'Baseline' indicates training ViT-S/ViT-B with the adversarial domain adaptation method (i.e., Eq. (2)). We train ViT-S/ViT-B models using a mini-batch SGD optimizer with a momentum of 0.9, which initializes the learning rate as 0 and linearly increase it to $lr = 0.03$ after 500 training steps except with $lr = 0.003$ for D → A and W → A in Office-31 dataset. For another, we set the masking ratio ε to 0.4 for masked image modeling.

Experimental results

The detailed experimental results have been shown in Tables 1, 2 and 3, we can observe that those transformer-based UDA methods perform better than ResNet-based UDA models, which shows that the transformer has more power than ResNet on UDA tasks. The detailed experimental analyses of different datasets have been shown below.

Office-31 results As shown in Table 1, the proposed TAMS outperforms the other ResNet-based UDA models and obtains better performance than the other ViT-based UDA

Table 1 Accuracy (%) on the Office-31 dataset

Backbone	Methods	A → W	D → W	W → D	A → D	D → A	W → A	Avg
AlexNet	Source-only	61.6	95.4	99.0	63.8	51.1	49.8	70.1
	DDC	61.8	95.0	98.5	64.4	52.1	52.2	70.6
	DAN	68.5	96.0	99.0	67.0	54.0	53.1	72.9
	RevGrad	73.0	96.4	99.2	72.3	53.4	51.2	74.3
	JAN	75.2	96.6	99.6	72.8	57.5	56.3	76.3
	CDAN	78.3	97.2	100.0	76.3	57.3	57.3	77.7
	PFAN	83.0	99.0	99.9	76.3	63.3	60.8	80.4
ResNet-50	Source-only	68.4	96.7	99.3	68.9	62.5	60.7	76.1
	DDC	75.6	96.0	98.2	76.5	62.2	61.5	78.3
	DAN	80.5	97.1	99.6	78.6	63.6	62.8	80.4
	RevGrad	82.0	96.9	99.1	79.7	68.2	67.4	82.2
	JAN	86.0	96.7	99.7	85.1	69.2	70.7	84.6
	CDAN	94.1	98.6	100.0	92.9	71.0	69.3	87.7
	TADA	94.3	98.7	99.8	91.6	72.9	73.0	88.4
	TAT	92.5	99.3	100.0	93.2	73.1	72.1	88.4
	SHOT	90.1	98.4	99.9	94.0	74.7	74.3	88.6
	ALDA	95.6	97.7	100.0	94.0	72.2	72.5	88.7
ViT-S	Source-only	86.9	98.6	100.0	88.6	76.0	75.9	87.7
	Baseline	91.9	99.1	100.0	89.2	78.4	77.9	89.4
	CDTrans	93.5	98.2	99.6	94.6	78.4	78.0	90.4
	TAMS	**94.02**	**99.2**	100.0	**95.1**	**78.9**	**78.6**	**90.97**
ViT-B	Source-only	91.2	99.2	100.0	90.4	81.1	80.6	90.4
	Baseline	92.5	99.2	100.0	93.6	80.7	80.7	91.1
	TVT	96.35	99.37	100.0	96.39	84.91	86.05	93.85
	CDTrans	96.7	99.0	100.0	97.0	81.1	81.9	92.6
	SSRT-B	**97.7**	99.2	100.0	**98.6**	83.5	82.2	93.5
	TAMS	96.48	**99.5**	100.0	96.99	**85.87**	**86.26**	**94.18**

The best results have been shown in bold face

methods. For example, on some transfer tasks (e.g., D → A and W → A), the proposed TAMS performs better than CDTrans [18], TVT [16] and SSRT-B [17].

Office-Home results As shown in Table 2, our method has the highest average accuracy of 85.63%. Compared with the ResNet-based models, the ViT-based models have a larger improvement. In particular, the proposed TAMS significantly improves the performance in all the transfer tasks. Compared with SSRT-B [17], our method performs better on some transfer tasks (e.g., Cl → Pr and Rw → Ar). When transferring to the CL domain, many methods have lower accuracy (e.g., CDTrans [18] and TVT [16]), our method can achieve better performance, which implicitly indicates that our method has better generalization ability on different transfer tasks.

VisDA-2017 results As shown in Table 3, our method achieves a better average accuracy of 88.4%. Compared with the other methods, our TAMS achieves the best performance on three classes including bicycle, horse, and skateboard.

Ablation study

We conduct ablation studies to verify the effects of the different loss functions, including $\mathcal{L}_{\text{cls}}(x^s, y^s)$, $\mathcal{L}_{\text{adv}}(x^s, x^t)$, $\mathcal{L}_{\text{self-KD}}(x^t; \theta)$, $\mathcal{L}_{\text{MIM}}(x^t; \theta)$, $\mathcal{L}_{\text{patch}}(x^s, x^t)$ and $I(p^t; x^t)$. We also explore the effects of the different ViT encoders, including traditional self-attention with cross-attention of Eq. (6). Comparing the second, third and fourth rows in Table 4, we can see that self-distillation loss and masked image modeling are effective for ViT backbones in UDA tasks. Comparing the fourth and fifth rows, the patch-level adversarial loss further improves the performance of the Office-31 dataset. In addition, the proposed TAMS with cross-attention in Eq. (6) achieves better performance than traditional self-attention in ViT, which demonstrates the effectiveness of the weighted cross-attention mechanism used in TAMS.

Table 2 Accuracy (%) on the Office-Home dataset

Backbone	Methods	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	Source-only	34.9	50	58	37.4	41.9	46.2	38.5	31.2	60.4	53.9	41.2	59.9	46.1
	MinEnt	51.0	71.9	77.1	61.2	69.1	70.1	59.3	48.7	77.0	70.4	53.0	81.0	65.8
	DAN	43.6	57	67.9	45.8	56.5	60.4	44	43.6	67.7	63.1	51.5	74.3	56.3
	CDAN + E	50.7	70.6	76	57.6	70	70	57.4	50.9	77.3	70.9	56.7	81.6	65.8
	SAFN	52	71.7	76.3	64.2	69.9	71.9	63.7	51.4	77.1	70.9	57.1	81.5	67.3
	BNM	56.7	77.5	81.0	67.3	76.3	77.1	65.3	55.1	82.0	73.6	57.0	84.3	71.1
	CDAN + TN	50.2	71.4	77.4	59.3	72.7	73.1	61	53.1	79.5	71.9	59	82.9	67.6
	SHOT	57.1	78.1	81.5	68	78.2	78.1	67.4	54.9	82.2	73.3	58.8	84.3	71.8
	DCAN + SCDA	60.7	76.4	82.8	69.8	77.5	78.4	68.9	59	82.7	74.9	61.8	84.5	73.1
ViT-S	Source-only	47.01	76.98	83.54	69.84	77.11	80.42	68.15	44.08	82.86	74.78	47.97	84.66	69.78
	Baseline	59.59	80.11	84.67	73.84	78.49	81.36	74.41	59.82	86.27	80.10	62.59	87.23	75.71
	CDTrans	60.6	79.5	82.4	75.6	81.0	82.3	72.5	56.7	84.4	77.0	59.1	85.5	74.7
	SSRT-S	67.03	84.21	**88.32**	**79.85**	84.28	87.58	**80.72**	66.03	88.27	82.04	**69.44**	**89.86**	80.64
	TAMS	**67.45**	**84.57**	87.49	79.63	**84.43**	**88.34**	80.23	**67.32**	**88.41**	**82.34**	69.14	89.56	**80.74**
ViT-B	Source-only	54.68	83.04	87.15	77.3	83.42	85.54	74.41	50.9	87.22	79.56	53.79	88.8	75.48
	Baseline	66.96	85.74	88.07	80.06	84.12	86.67	79.52	67.03	89.44	83.64	70.15	91.17	81.05
	CDTrans	68.8	85	86.9	81.5	87.1	87.3	79.6	63.3	88.2	82	66	90.6	80.5
	TVT	74.89	86.82	89.47	82.78	87.95	88.27	79.81	71.94	90.13	85.46	74.62	90.56	83.56
	SSRT-B	75.17	**88.98**	**91.09**	85.13	88.29	**89.95**	**85.04**	**74.23**	**91.26**	85.7	**78.58**	**91.78**	85.43
	TAMS	75.3	**88.98**	**91.09**	**85.37**	**89.75**	**89.95**	**85.04**	**74.23**	**91.26**	**86.24**	**78.58**	**91.78**	**85.63**

The best results have been shown in bold face

Table 3 Accuracy (%) on the VisDA-2017 dataset

Backbone	Methods	Plane	bcycl	Bus	Car	Horse	Knife	mcycl	Person	Plant	sktbrd	Train	Truck	Avg
ResNet-101	ResNet-101	55.1	53.3	61.9	59.1	80.6	17.9	79.7	31.2	81	26.5	73.5	8.5	52.4
	DANN	81.9	77.7	82.8	44.3	81.2	29.5	65.1	28.6	51.9	54.6	82.8	7.8	57.4
	CDAN	85.2	66.9	83	50.8	84.2	74.9	88.1	74.5	83.4	76	81.9	38	73.9
	SAFN	93.6	61.3	84.1	70.6	94.1	79	91.8	79.6	89.9	55.6	89	24.4	76.1
	MCD	87.0	60.9	83.7	64.0	88.9	79.6	84.7	76.9	88.6	40.3	83.0	25.8	71.9
	SWD	90.8	82.5	81.7	70.5	91.7	69.5	86.3	77.5	87.4	63.6	85.6	29.2	76.4
	BNM	89.6	61.5	76.9	55.0	89.3	69.1	81.3	65.5	90.0	47.3	89.1	30.1	70.4
	SHOT	94.3	88.5	80.1	57.3	93.1	94.9	80.7	80.3	91.5	89.1	86.3	58.2	82.9
ViT-B	Baseline	**99.09**	60.66	70.55	82.66	96.5	73.06	**97.14**	19.73	64.48	94.74	**97.21**	15.36	72.6
	CDTrans	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	92.92	85.58	77.51	60.48	93.6	98.17	89.35	76.4	93.56	92.02	91.69	55.73	83.92
	SSRT-B	98.93	87.6	**89.1**	**84.77**	98.34	**98.7**	96.27	81.08	94.86	97.9	94.5	43.13	**88.76**
	TAMS	98.77	**93.58**	88.12	76.34	**98.81**	97.64	96.27	79.45	91.07	**98.68**	97.03	44.79	88.38

The best results have been shown in bold face

Diversity analysis with masked image modeling

We examine the attention diversity of heads and observe whether masked image modeling (MIM) is conducive to improving the transferable performance of UDA. Following the [15], we use the average attention distance to measure whether it is local attention or global attention, which can partially reflect the receptive field size for each attention head. Figure 3 shows the average attention distance per head and layer depth using the ViT-B/16 architectures. We visualize the average attention distance between MIM and without MIM. It can be seen that: (1) Without MIM (the top row), the average attention distances of different heads in deeper layers collapse to locate within a very small distance range. This suggests that different heads learn very similar visual cues and may be wasting model capacity. (2) After using MIM (the bottom row), the attention representations become more diverse regarding the average attention distance, especially for deeper layers. As analyzed in [52], masked image modeling brings locality inductive bias to the trained model and more diversity on attention heads, which is useful to enhance the transferable performance of UDA.

Effect of different masking ratio

As shown in Fig. 4, we analyze the influence of different masking ratios ε on Office-31. When $\varepsilon = 0.4$, the model can obtain better test accuracy, which implicitly shows that it is beneficial to enhance the transferable performance of UDA by making full use of the supervision information of pixel-level in the target domain.

Effect of different masking strategies

For UDA tasks, we present a simple random masking strategy. Furthermore, we also study how different masking strategies affect the effectiveness of UDA. In our experiments, we try other masking strategies (i.e., square [53], block-wise [54], and random) with different masked patch sizes (i.e., 16 and 32). The detailed experimental results have been shown in Table 5. We first notice that the best test accuracy of our simple random masking strategy reaches 94.18%, which is + 0.12% higher than the best block-wise masking strategy. In addition, when a large masked patch size of 32 is adopted, the different masking strategies perform stably well on a small range of precision.

Effect of adversarial cross-domain adaptation

To further verify the effectiveness of adversarial cross-domain adaptation (ACA), we use t-SNE to visualize the feature representation between with ACA and without ACA, as shown in Fig. 5. Blue and red points represent the source domain and target domain samples, respectively. It can be seen that without ACA, it is not well aligned such as A $\rightarrow$ W and W $\rightarrow$ A. In addition, when using ACA, TAMS has good feature alignment, as shown in W $\rightarrow$ A, our method not only has compressed intra-class representations and separable inter-class representations but also can align feature distribution more effectively than without ACA. This shows that adversarial cross-domain adaptation can capture discriminative information and better alignment.

Table 4 Ablation study of each module

$\mathcal{L}_{\text{cls}}(x^s, y^s)$	$\mathcal{L}_{\text{adv}}(x^s, x^t)$	$\mathcal{L}_{\text{self-KD}}(x^t; \theta)$	$\mathcal{L}_{\text{MIM}}(x^t; \theta)$	$\mathcal{L}_{\text{patch}}(x^s, x^t)$	$I(p^t; x^t)$	A → W	D → W	W → D	A → D	D → A	W → A	Avg
✓						89.18	98.87	100	88.76	80.09	79.77	89.45
✓	✓					90.12	98.89	100	90.45	83.43	83.23	91.02
✓	✓	✓				93.14	98.93	100	93.32	84.63	85.13	92.53
✓	✓	✓	✓			95.34	99.03	100	94.14	84.93	85.67	93.19
✓	✓	✓	✓	✓		95.44	99.01	100	94.69	85.05	85.96	93.36
✓	✓	✓	✓	✓	✓	**96.48**	**99.5**	**100**	**96.99**	**85.87**	**86.26**	**94.18**
$\mathcal{L}_{\text{TAMS}}$ with traditional self-attention in ViT encoder						93.52	98.79	100	94.19	84.33	85.79	92.77
$\mathcal{L}_{\text{TAMS}}$ with the cross-attention mechanism in ViT encoder						95.49	99.13	100	95.82	85.28	85.98	93.62
$\mathcal{L}_{\text{TAMS}}$ with the weighted cross-attention mechanism (Eq. (6)) in ViT encoder						**96.48**	**99.5**	**100**	**96.99**	**85.87**	**86.26**	**94.18**

The best results have been shown in bold face

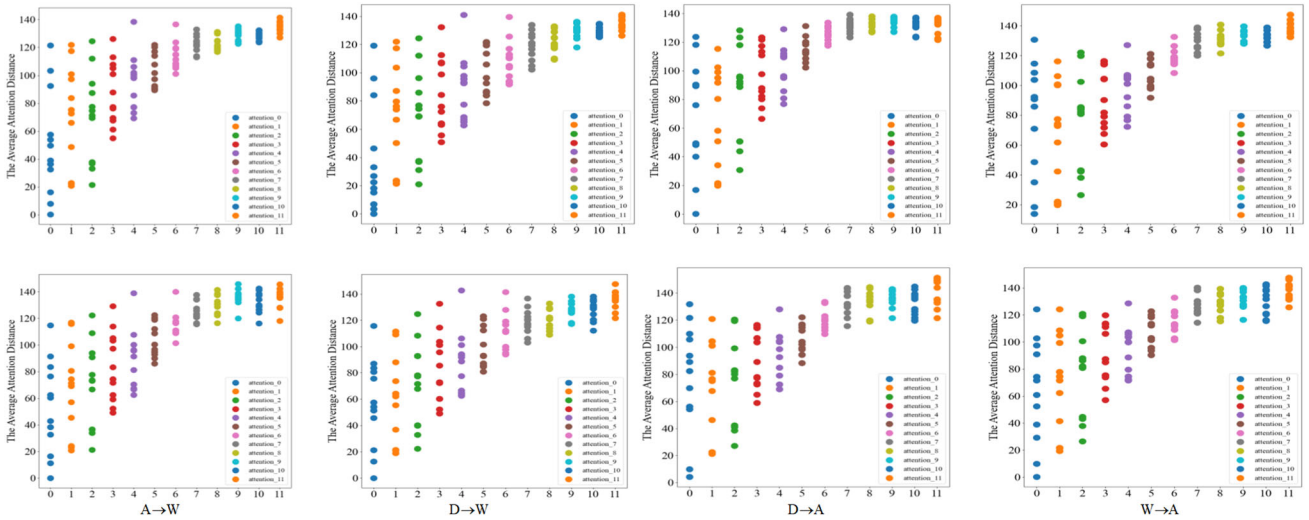


Fig. 3 The average attention distance in different attention heads (dots). Top row: TAMS does not use masked image modeling (MIM); Bottom row: TAMS uses masked image modeling (MIM)

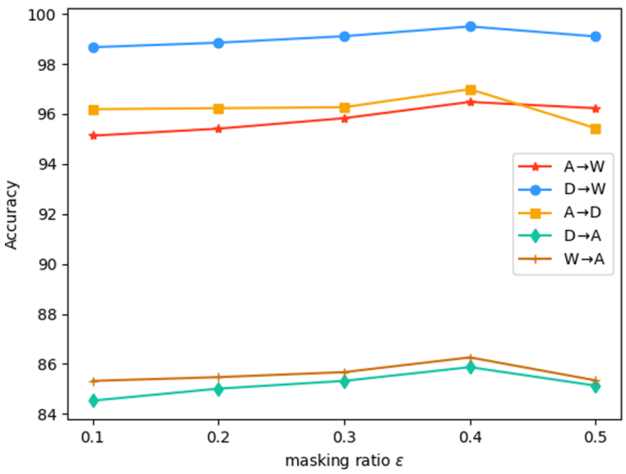


Fig. 4 The influence of different masking ratios on Office-31. When $\varepsilon = 0.4$, the model can obtain better test accuracy

Effect of masked self-distillation training

On UDA tasks, the predicted class distribution on target domain data may collapse, so we analyze whether our method is safety training. In [17], Sun et al. proposed a self-refinement strategy to avoid collapse. Our method can avoid such situations because of using the self-distillation of the teacher–student framework, where the teacher branch updates weights from the student network using the Exponential Moving Average (EMA). Following the [17], we also use the diversity curve of model predictions on the target domain to analyze whether our method is safety training. As shown in Fig. 6a, our method keeps stability training on different transfer tasks. In addition, Fig. 6b, c plot results of class-level adversarial and patch-level adversarial loss, our method can converge well on the training.

Parameter sensitivity analysis

We analyze the parameter sensitivity of TAMS by conducting experiments on Office-31 datasets. The parameters α, β, γ and δ were searched in $\{0.01, 0.1, 1\}$. For the sake of simplicity, we set the parameter $\alpha = \beta = \delta$ to take the same weights on three components and analyze the influence of different parameters γ on the performance. As shown in Fig. 7, it can

Table 5 Experiments on different masking strategies (i.e., square [53], block-wise [54], and random) with different masked patch sizes (i.e., 16 and 32), mask ratio is set to 0.4

Mask type	Masked patch size	A → W	D → W	W → D	A → D	D → A	W → A	Avg
Square [53]	16	96.33	99.39	100	96.76	85.43	86.17	94.01
	32	96.21	99.21	100	96.63	85.38	86.11	93.93
Block-wise [54]	16	96.43	99.4	100	96.78	85.56	86.17	94.06
	32	96.22	99.25	100	96.69	85.42	86.13	93.96
Random (ours)	16	96.48	99.5	100	96.99	85.87	86.26	94.18
	32	96.37	99.3	100	96.68	85.65	86.14	94.02

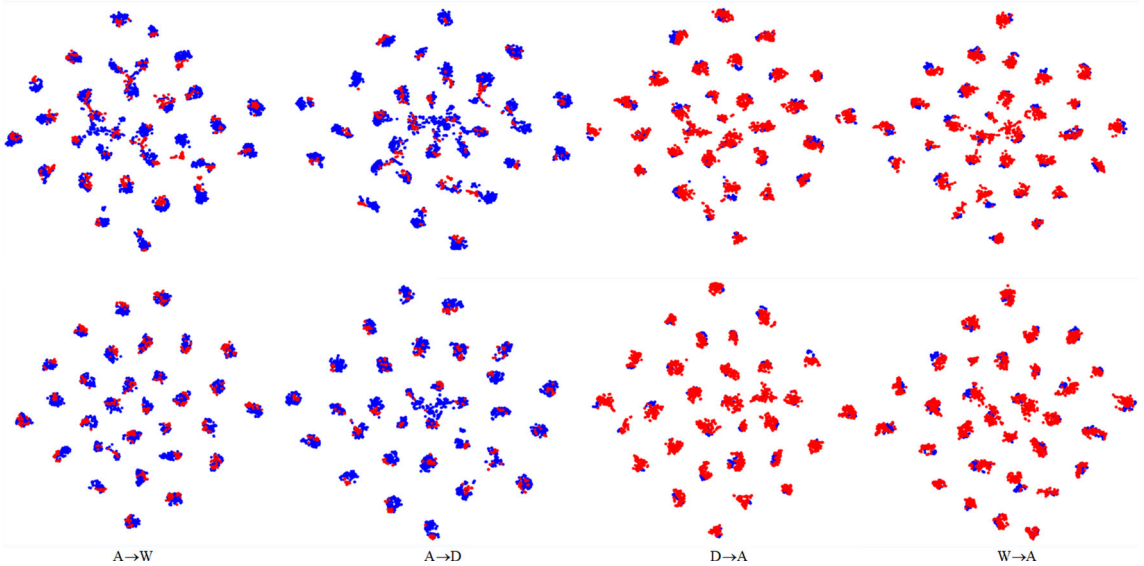


Fig. 5 t-SNE visualization of feature alignment. Blue points are source samples, and red points are target samples. Top row: TAMS does not use adversarial cross-domain adaptation (Eq. (6)); Bottom row: TAMS uses adversarial cross-domain adaptation (Eq. (6))

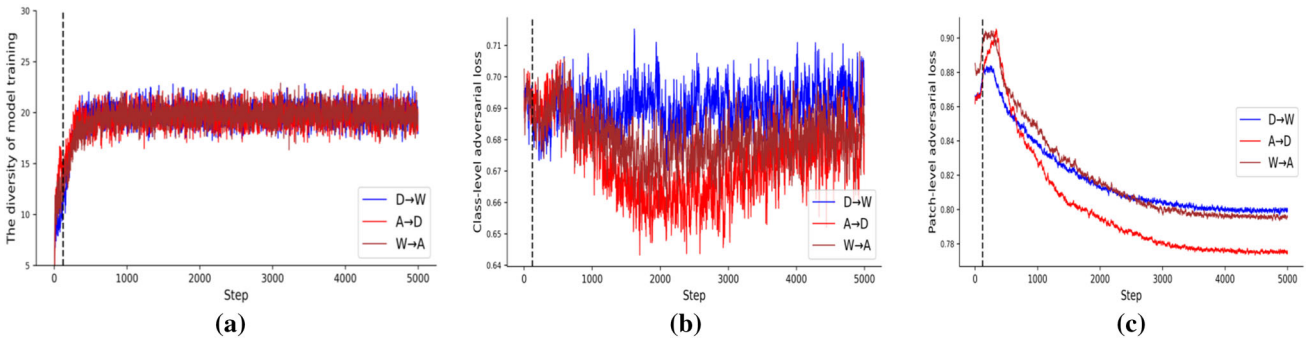


Fig. 6 Training curve on three tasks (i.e., D → W, A → D and W → A) of Office-31. **a** Plots of the diversity of model predictions on target domain data [17]; **b** class-level adversarial loss curve on Eq. (2); **c** patch-level adversarial loss curve on Eq. (7)

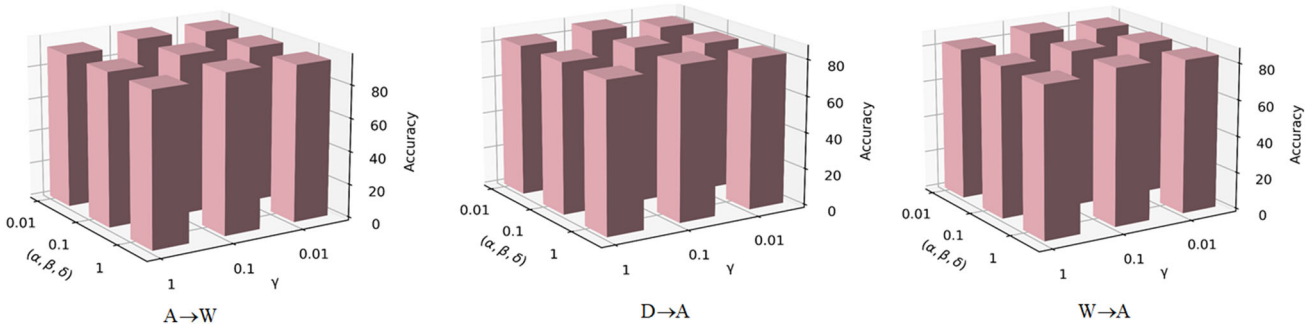
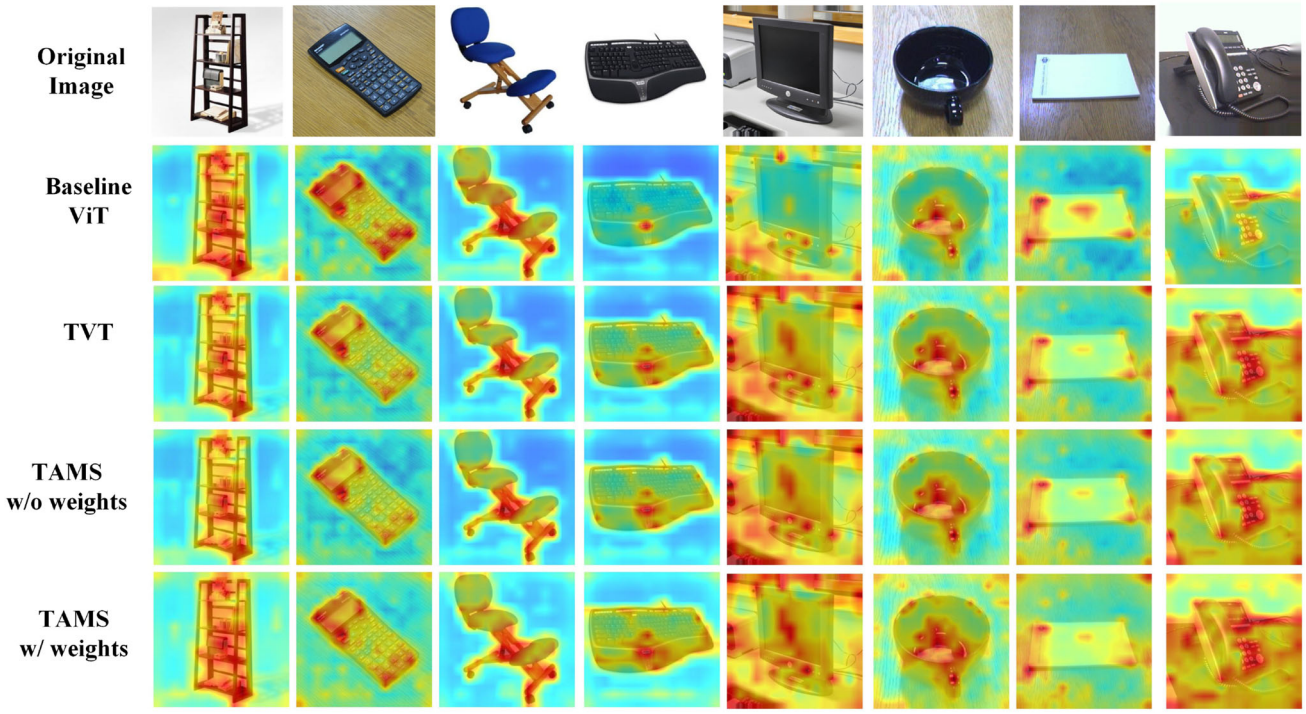
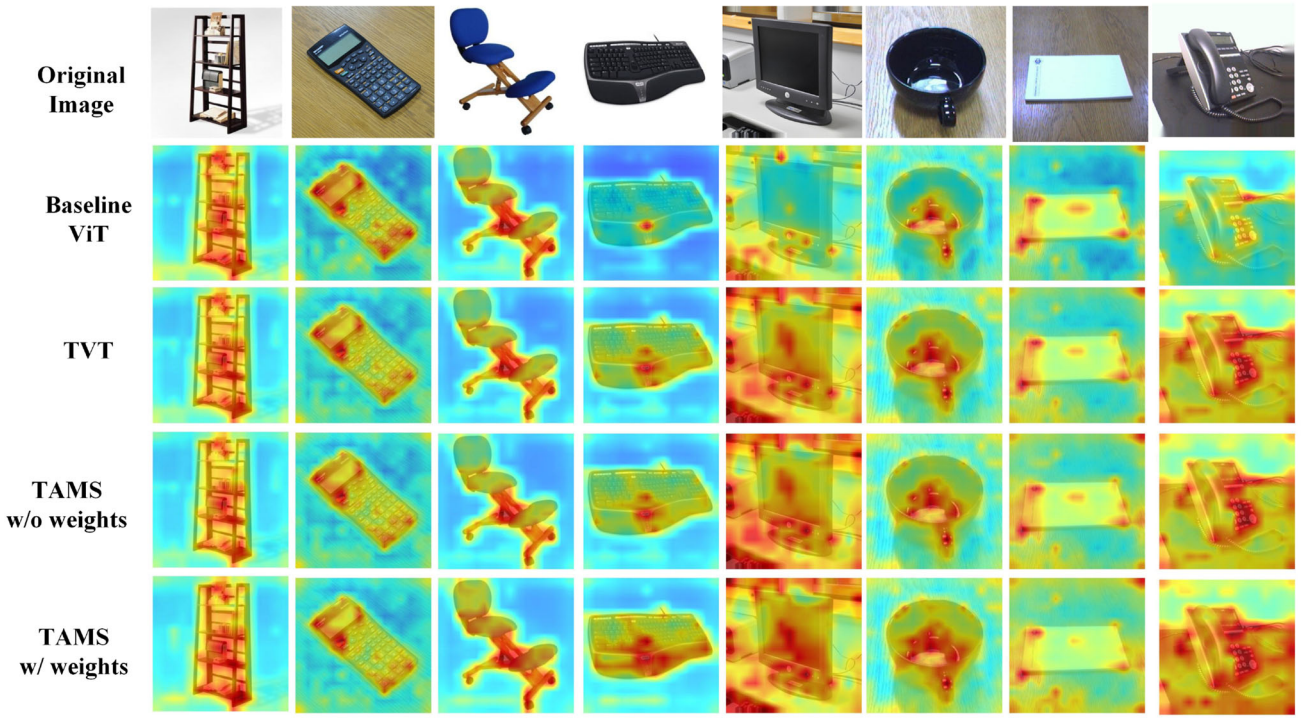


Fig. 7 Parameter sensitivity analysis. When $\gamma = 0.01$, the candidate sets for $\alpha = \beta = \delta$ can be employed in $\{0.01, 0.1, 1\}$ to obtain satisfactory performance. When $\alpha = \beta = \delta = 0.1$, $\gamma = 0.01$, three difficult tasks (i.e., A → W, D → A and W → A) can obtain the best performance

be seen that when $\gamma = 0.01$, the candidate sets for $\alpha = \beta = \delta$ can be employed in $\{0.01, 0.1, 1\}$ to obtain satisfactory performance. When $\alpha = \beta = \delta = 0.1$, $\gamma = 0.01$, three difficult tasks (i.e., A → W, D → A and W → A) can obtain the best performance. Therefore, in our experiments, α, β and δ are set to 0.1, while γ is set to 0.01. In addition, when $\gamma = 0.1$, there is no significant decrease in test accuracy, which implicitly indicates that our method is not parameter sensitive.



(a) D-A



(b) W-A

Fig. 8 Attention maps of images on Office-31 dataset. **a** D → A; **b** W → A. The hotter the color, the higher the attention

Attention visualization analysis

We visualize the attention map using Grad-CAM [55] to verify that our model can capture important local regions, where Grad-CAM uses the gradients of any target concept flowing into the final layer to produce a coarse localization map highlighting the important regions in the image for predicting the

concept. We randomly sample Office-31 images from two more difficult tasks (i.e., D → A and W → A) to visualize the attention map. As shown in Fig. 8, the proposed TAMS captures more accurate regions than the baseline. For instance, in the keyboard and cup, the TAMS method owns more hot areas on the target object than the baseline method. In addition, we also compare the difference between TAMS without weights by entropy and using weights. It can be seen that TAMS with weights can focus more attention on hot regions, which promotes the transferability of ViT in UDA tasks.

Conclusion

In this paper, we propose the transferable adversarial masked self-distillation to improve the performance of UDA, which consists of three parts including adversarial masked self-distillation, masked image modeling, and adversarial cross-domain adaptation objective. The proposed TAMS simultaneously takes class-level, pixel-level, and patch-level representations into account. Experimental results show that the proposed TAMS outperforms existing state-of-the-art ResNet-based and ViT-based methods on three benchmark datasets. In future work, we will further apply TAMS to other computer vision tasks such as object detection and semantic segmentation.

Acknowledgements This study is supported by the Yunnan Provincial Department of Education Science Research Fund, China (2023J0209); supported by the Natural Science Doctoral Research Start-Up Fund of Yunnan Normal University, China (01000205020503147); supported by the Yunnan Provincial Department of Education Science Research Fund, China (2023J0208); supported by the Key Project of Applied Basic Research Program of Yunnan Province, China (Grant no. 2018FA033).

Data Availability The authors confirm that the data supporting the findings of this study are available within the article.

Declarations

Conflict of interest The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

1. Liu Z, Lin Y, Cao Y, Hu H, Wei Y, Zhang Z, Lin S, Guo B (2021) Swin transformer: hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 10012–10022
2. Radford A, Kim JW, Hallacy C, Ramesh A, Goh G, Agarwal S, Sastry G, Askell A, Mishkin P, Clark J (2021) Learning transferable visual models from natural language supervision. In: International conference on machine learning. PMLR, pp 8748–8763
3. Csurka G (2017) Domain adaptation for visual applications: a comprehensive survey. arXiv preprint. arXiv:1702.05374
4. Wang P, Yang Y, Xia Y, Wang K, Zhang X, Wang S (2022) Information maximizing adaptation network with label distribution priors for unsupervised domain adaptation. IEEE Trans Multimed
5. Wilson G, Cook DJ (2020) A survey of unsupervised deep domain adaptation. ACM Trans Intell Syst Technol 11(5):1–46
6. Tzeng E, Hoffman J, Saenko K, Darrell T (2017) Adversarial discriminative domain adaptation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 7167–7176
7. Tzeng E, Hoffman J, Zhang N, Saenko K, Darrell T (2014) Deep domain confusion: maximizing for domain invariance. arXiv preprint. arXiv:1412.3474
8. Long M, Cao Y, Wang J, Jordan M (2015) Learning transferable features with deep adaptation networks. In: International conference on machine learning. PMLR, pp 97–105
9. Chen C, Chen Z, Jiang B, Jin X (2019) Joint domain alignment and discriminative feature learning for unsupervised deep domain adaptation. In: Proceedings of the AAAI conference on artificial intelligence, vol 33, pp 3296–3303
10. Ganin Y, Lempitsky V (2015) Unsupervised domain adaptation by backpropagation. In: International conference on machine learning. PMLR, pp 1180–1189
11. Wang J, Chen Y, Feng W, Yu H, Huang M, Yang Q (2020) Transfer learning with dynamic distribution adaptation. ACM Trans Intell Syst Technol 11(1):1–25
12. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
13. Peng X, Usman B, Kaushik N, Hoffman J, Wang D, Saenko K (2017) Visda: the visual domain adaptation challenge. arXiv preprint. arXiv:1710.06924
14. Liang J, Hu D, Feng J (2020) Do we really need to access the source data? Source hypothesis transfer for unsupervised domain adaptation. In: International conference on machine learning. PMLR, pp 6028–6039
15. Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S et al (2020) An image is worth 16x16 words: transformers for image recognition at scale. arXiv preprint. arXiv:2010.11929
16. Yang J, Liu J, Xu N, Huang J (2021) Tvt: transferable vision transformer for unsupervised domain adaptation. arXiv preprint. arXiv:2108.05988
17. Sun T, Lu C, Zhang T, Ling H (2022) Safe self-refinement for transformer-based domain adaptation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 7191–7200
18. Xu T, Chen W, Wang P, Wang F, Li H, Jin R (2021) Cdtrans: cross-domain transformer for unsupervised domain adaptation. arXiv preprint. arXiv:2109.06165
19. Xie Z, Zhang Z, Cao Y, Lin Y, Bao J, Yao Z, Dai Q, Hu H (2022) Simmim: a simple framework for masked image modeling. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 9653–9663

20. Zhang Y, Deng B, Tang H, Zhang L, Jia K (2020) Unsupervised multi-class domain adaptation: theory, algorithms, and practice. IEEE Trans Pattern Anal Mach Intell
21. Ghifary M, Kleijn WB, Zhang M, Balduzzi D, Li W (2016) Deep reconstruction-classification networks for unsupervised domain adaptation. In: European conference on computer vision. Springer, Berlin, pp 597–613
22. Bousmalis K, Silberman N, Dohan D, Erhan D, Krishnan D (2017) Unsupervised pixel-level domain adaptation with generative adversarial networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3722–3731
23. Hoffman J, Tzeng E, Park T, Zhu J-Y, Isola P, Saenko K, Efros A, Darrell T (2018) Cycada: cycle-consistent adversarial domain adaptation. In: International conference on machine learning. PMLR, pp 1989–1998
24. Saito K, Watanabe K, Ushiku Y, Harada T (2018) Maximum classifier discrepancy for unsupervised domain adaptation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3723–3732
25. Kang G, Jiang L, Yang Y, Hauptmann AG (2019) Contrastive adaptation network for unsupervised domain adaptation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 4893–4902
26. Du Z, Li J, Su H, Zhu L, Lu K (2021) Cross-domain gradient discrepancy minimization for unsupervised domain adaptation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 3937–3946
27. Li J, Li G, Shi Y, Yu Y (2021) Cross-domain adaptive clustering for semi-supervised domain adaptation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 2505–2514
28. Gretton A, Borgwardt K, Rasch M, Schölkopf B, Smola A (2006) A kernel method for the two-sample-problem. Adv Neural Inf Process Syst 19
29. Sun B, Feng J, Saenko K (2016) Return of frustratingly easy domain adaptation. In: Proceedings of the AAAI conference on artificial intelligence, vol 30
30. Sun B, Saenko K (2016) Deep coral: correlation alignment for deep domain adaptation. In: European conference on computer vision. Springer, Berlin, pp 443–450
31. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2020) Generative adversarial networks. Commun ACM 63(11):139–144
32. Luo Y, Zheng L, Guan T, Yu J, Yang Y (2019) Taking a closer look at domain shift: category-level adversaries for semantics consistent domain adaptation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 2507–2516
33. Ma H, Lin X, Wu Z, Yu Y (2021) Coarse-to-fine domain adaptive semantic segmentation with photometric alignment and category-center regularization. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 4051–4060
34. Caron M, Touvron H, Misra I, Jégou H, Mairal J, Bojanowski P, Joulin A (2021) Emerging properties in self-supervised vision transformers. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 9650–9660
35. Xia Y, Yang Y (2021) Generalization self-distillation with epoch-wise regularization. In: 2021 International joint conference on neural networks (IJCNN). IEEE, pp 1–8
36. He T, Shen L, Guo Y, Ding G, Guo Z (2022) Secret: self-consistent pseudo label refinement for unsupervised domain adaptive person re-identification. In: Proceedings of the AAAI conference on artificial intelligence, vol 36, pp 879–887
37. Li J, Savarese S, Hoi S.C (2022) Masked unsupervised self-training for zero-shot image classification. arXiv preprint. arXiv:2206.02967
38. Tarvainen A, Valpola H (2017) Mean teachers are better role models: weight-averaged consistency targets improve semi-supervised deep learning results. Adv Neural Inf Process Syst 30
39. Sohn K, Berthelot D, Carlini N, Zhang Z, Zhang H, Raffel CA, Cubuk ED, Kurakin A, Li C-L (2020) Fixmatch: simplifying semi-supervised learning with consistency and confidence. Adv Neural Inf Process Syst 33:596–608
40. Zhao H, Ma C, Chen Q, Deng Z (2021) Domain adaptation via maximizing surrogate mutual information. arXiv preprint. arXiv:2110.12184
41. Do K, Tran T, Venkatesh S (2021) Clustering by maximizing mutual information across views. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 9928–9938
42. Saenko K, Kulis B, Fritz M, Darrell T (2010) Adapting visual category models to new domains. In: European conference on computer vision. Springer, Berlin, pp 213–226
43. Venkateswara H, Eusebio, J, Chakraborty S, Panchanathan S (2017) Deep hashing network for unsupervised domain adaptation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5018–5027
44. Long M, Zhu H, Wang J, Jordan M.I (2017) Deep transfer learning with joint adaptation networks. In: International conference on machine learning. PMLR, pp 2208–2217
45. Grandvalet Y, Bengio Y (2004) Semi-supervised learning by entropy minimization. Adv Neural Inf Process Syst 17
46. Long M, Cao Z, Wang J, Jordan MI (2018) Conditional adversarial domain adaptation. Adv Neural Inf Process Syst 31
47. Lee C-Y, Batra T, Baig MH, Ulbricht D (2019) Sliced Wasserstein discrepancy for unsupervised domain adaptation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 10285–10295
48. Cui S, Wang S, Zhuo J, Li L, Huang Q, Tian Q (2020) Towards discriminability and diversity: batch nuclear-norm maximization under label insufficient situations. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 3941–3950
49. Li S, Liu C, Lin Q, Xie B, Ding Z, Huang G, Tang J (2020) Domain conditioned adaptation network. In: Proceedings of the AAAI conference on artificial intelligence, vol 34, pp 11386–11393
50. Liang J, Hu D, Feng J (2021) Domain adaptation with auxiliary target domain-oriented classifier. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 16632–16642
51. Chen M, Zhao S, Liu H, Cai D (2020) Adversarial-learned loss for domain adaptation. In: Proceedings of the AAAI conference on artificial intelligence, vol 34, pp 3521–3528
52. Xie Z, Geng Z, Hu J, Zhang Z, Hu H, Cao Y (2022) Revealing the dark secrets of masked image modeling. arXiv preprint. arXiv:2205.13543
53. Pathak D, Krahenbuhl P, Donahue J, Darrell T, Efros AA (2016) Context encoders: feature learning by inpainting. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2536–2544
54. Bao H, Dong L, Piao S, Wei F (2021) Beit: bert pre-training of image transformers. arXiv preprint. arXiv:2106.08254
55. Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D (2017) Grad-cam: visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE international conference on computer vision, pp 618–626