DAMamba: State-Space Modeling for Robust Unsupervised Domain Adaptation

Yuqi He[1]
[1] *College of Computer Science and Technology, Jilin University, Changchun, China, heyq9923@mails.jlu.edu.cn*

***Abstract*—Unsupervised domain adaptation (UDA) remains challenging due to severe distribution shifts between the labeled source and unlabeled target domains. Existing UDA approaches are based on Convolution Neural Networks (CNNs) or Vision Transformers (ViTs), which suffer from limited receptive fields or quadratic complexity issues. While recent State Space Models (SSMs) like Mamba offer linear complexity and strong long-range modeling, their recursive state updates make them vulnerable to amplified style drift and misaligned local semantics under domain shift. To address these issues, we propose DAMamba, a novel unsupervised domain adaptation framework tailored for Mamba-based visual backbones. DAMamba introduces two key modules: a State-Aware Style Memory (SASM) that models global and class-wise target statistics and performs dynamic style transfer to suppress state drift, and a Bidirectional Patch Alignment (BPA) that enforces fine-grained semantic correspondence through saliency-guided patch matching. Extensive experiments demonstrate that DAMamba significantly enhances cross-domain robustness while preserving computational efficiency.**

***Index Terms*—Unsupervised Domain Adaptation, Mamba, State Space Models, Visual Recognition**

I. Introduction

The rapid progress of deep neural networks and the availability of large-scale annotated datasets have pushed image classification to near-saturated performance on the training domain. However, when models are deployed in real-world environments, their performance often deteriorates sharply due to substantial discrepancies in style, illumination, texture, and scene composition between the training (source) and deployment (target) domains. This phenomenon, widely known as *domain shift*, highlights the persistent challenge of ensuring stable model behavior under distribution changes. Since collecting target-domain annotations is costly or impractical in many applications, *Unsupervised Domain Adaptation* (UDA) has become indispensable for improving model's adaptability from the labeled source data to the unlabeled target data.

A rich line of UDA techniques has been developed, including discrepancy minimization [1]–[8], adversarial alignment [9]–[17], and pseudo-labeling [18]–[23] methods, *etc*. Most of these approaches rely on either convolutional neural networks (CNNs) or Vision Transformers (ViTs). Although CNN-based UDA models are efficient, their inherently limited receptive fields restrict their ability to capture long-range cross-domain relationships, making them sensitive to strong style variation. ViT-based UDA models alleviate this limitation via global self-attention, but their quadratic computational complexity in token length significantly increases memory and computation costs. As modern UDA scenarios require both rich global modeling and efficient cross-domain alignment, a backbone with linear complexity, global receptive field, and strong transferability is highly desirable, yet remains underexplored in the UDA setting.

Recently, Mamba-style State Space Models (SSMs) [24]–[27] have emerged as a compelling alternative to CNNs and Transformers. By parameterizing selective state-space transitions, Mamba achieves long-range dependency modeling with linear computational complexity and superior scalability. These properties make Mamba an attractive backbone for domain adaptation. Nevertheless, directly applying Mamba to UDA is far from trivial. Unlike CNNs or ViTs, Mamba's recursive state updates introduce new forms of vulnerability: shallow-layer style noise can propagate and amplify through the recurrent dynamics, and the lack of explicit cross-domain normalization mechanisms limits its ability to adapt to large distribution gaps. As a result, *native Mamba backbones exhibit insufficient transferability under strong domain shifts*.

Motivated by these observations, we investigate how to adapt visual state-space models to the UDA setting and identify two fundamental challenges of applying Mamba to cross-domain learning. (1) *Style Drift Amplification*: domain-specific style discrepancies in low-level features accumulate along the state-space recurrence, causing state drift and corrupting semantic representations. (2) *Local Semantic Misalignment*: even if global distributions are roughly aligned, semantically similar patches across domains may remain mismatched due to local structural, texture, or foreground-background inconsistencies. Thus, a successful UDA framework for Mamba must simultaneously regulate shallow-layer style statistics and enforce fine-grained correspondence at the patch level.

To address these challenges, we propose DAMamba, a novel domain adaptation framework explicitly designed for Mamba-based visual backbones. Our DAMamba excels in strong adaptability toward target domains and meanwhile has the advantages of global receptive fields, and efficient linear complexity. Concretely, DAMamba introduces two modules integrated seamlessly into VMamba: (1) A *State-Aware Style Memory* (SASM) that constructs global and class-wise target style statistics and performs dynamic style transfer on source features, mitigating state drift without relying on any Batch Normalization. (2) A *Bidirectional Patch Alignment*(BPA) that selects salient patches and builds bi-directional soft semantic correspondences between source and target domains to eliminate local misalignment. Together, these components provide a principled solution to both shallow and high-level distribution

discrepancies while keeping Mamba's linear efficiency intact.

Our contributions are summarized as follows:

- We present DAMamba, a novel State Space Model-based framework for UDA that excels in strong adaptability across domains, and meanwhile has the advantages of global receptive fields and efficient linear complexity.
- We introduce two key modules, a State-Aware Style Memory for style regulation and a Bidirectional Patch Alignment for fine-grained semantic matching.
- Extensive experiments demonstrate that DAMamba consistently improves cross-domain robustness and achieves state-of-the-art performance across multiple UDA benchmarks while incurring negligible inference overhead.

II. Related Work

Unsupervised domain adaptation (UDA) aims to transfer knowledge from a labeled source domain to an unlabeled target domain under distribution shift. Classical UDA methods can be broadly grouped into three families. Discrepancy-based approaches minimize statistical distances such as maximum mean discrepancy [1]–[3], covariance alignment [4], [5], or category-level alignment [6]–[8] to encourage domain-invariant features. Adversarial alignment methods introduce domain discriminators [9], [10] to match feature distributions implicitly, with extensions incorporating class-conditional priors [11]–[14], or entropy reduction [15]–[17]. Self-training and information-maximization approaches generate pseudo-labels for the target domain and refine model predictions via confidence regularization [18]–[20], consistency learning [21]–[23]. Although effective, these methods are primarily designed for CNNs or ViTs and implicitly rely on normalization layers or global attention to regulate domain-specific style statistics and local feature correspondence. Unlike prior UDA approaches, DAMamba explicitly addresses the unique challenges of state-space backbones, *i.e.,* style drift amplification and local semantic misalignment, through dedicated state-aware style memory and bidirectional patch alignment mechanisms.

State Space Models (Mamba) , Mamba-style visual backbones [24] have been recently introduced into computer vision, as efficient alternatives to convolutional and transformer architectures. By parameterizing selective state transitions, Mamba captures long-range dependencies with linear-time complexity, offering improved scalability for high-resolution or long-sequence inputs. Vision variants [25]–[27] extend these ideas to 2D feature maps by arranging spatial tokens into sequences and applying recurrent state updates to aggregate global context. Despite their modeling power, SSM-based backbones lack built-in normalization or attention mechanisms that modulate domain-specific variations, making them particularly vulnerable to amplified shallow-layer style noise and unstable local correspondences under domain shift. Prior work has not explored how to adapt such recurrent architectures to UDA. In contrast, DAMamba introduces state-aware style modeling and fine-grained bidirectional patch alignment, providing the first UDA framework tailored specifically to the dynamics and vulnerabilities of Mamba-based vision models.

III. Methodology

We study standard unsupervised domain adaptation (UDA) for image classification. Given a labeled source domain $\mathcal{D}_s = \{(x_i^s, y_i^s)\}_{i=1}^{N_s}$ and an unlabeled target domain $\mathcal{D}_t = \{x_j^t\}_{j=1}^{N_t}$, our goal is to learn a classifier $f_\theta : \mathcal{X} \to \{1, \dots, K\}$ that generalizes well on the target domain.

We build on VMamba as the visual backbone and propose **DAMamba**, a UDA framework tailored for Mamba-style state-space models (SSMs). As discussed in Sec. I, directly applying VMamba to UDA faces two fundamental issues: (i) *style drift amplification* and *local semantic misalignment*. To address both, DAMamba introduces two complementary modules: **(1) State-Aware Style Memory (SASM)** for shallow-layer style regulation, and **(2) Bidirectional Patch Alignment (BPA)** for fine-grained semantic correspondence at higher-level stages. An overview of the whole framework is shown in Fig. 1.

A. State-Aware Style Memory (SASM)

SASM is inserted before the first VMamba stage to regulate shallow-layer statistics *before* recurrent state updates amplify domain-specific biases. Let the feature map entering SASM be $\mathbf{F} \in \mathbb{R}^{B\times C\times H\times W}$, where B is batch size and C is the channel dimension. For each sample b and channel c, SASM computes in-sample statistics over spatial locations:

$$\begin{aligned} \mu_{b,c} &= \frac{1}{HW}\sum_{h,w} F_{b,c,h,w}, \\ \sigma_{b,c} &= \sqrt{\frac{1}{HW}\sum_{h,w}\left(F_{b,c,h,w}-\mu_{b,c}\right)^2+\varepsilon}, \\ \hat{F}_{b,c,h,w} &= \frac{F_{b,c,h,w}-\mu_{b,c}}{\sigma_{b,c}+\varepsilon}. \end{aligned} \tag{1}$$

This *BN-free* normalization provides a stable reference for style transfer and avoids relying on BatchNorm, which is typically absent in SSM backbones.

Target Style Memory. On top of local statistics, SASM maintains a target-domain style memory at two levels: a **global** memory $(\boldsymbol{\mu}^g, \boldsymbol{\sigma}^g) \in \mathbb{R}^C$ and a **class-wise** memory $\{(\boldsymbol{\mu}^k, \boldsymbol{\sigma}^k, n_k)\}_{k=1}^K$, where n_k counts confident target samples assigned to class k. Given target logits $\mathbf{g}^t$ and probabilities $\mathbf{p}^t = \mathrm{softmax}(\mathbf{g}^t)$, the pseudo-label for sample b is $\hat{y}_b^t = \arg\max_k \mathbf{p}_{b,k}^t$ with confidence $\gamma_b^t = \max_k \mathbf{p}_{b,k}^t$. Only samples with $\gamma_b^t > \tau_p$ contribute to memory updates. We update the global memory via EMA:

$$\boldsymbol{\mu}^g \leftarrow \beta\boldsymbol{\mu}^g + (1-\beta)\boldsymbol{\mu}_{\mathrm{batch}}, \quad \boldsymbol{\sigma}^g \leftarrow \beta\boldsymbol{\sigma}^g + (1-\beta)\boldsymbol{\sigma}_{\mathrm{batch}}, \tag{2}$$

and update class-wise memories similarly using confident samples assigned to class k. For stability, when n_k is small, we fall back to global memory.

Source-to-Target Style Injection. Given source feature $\hat{\mathbf{F}}^s$ from Eq. (1) and its ground-truth label $y_b^s = k$, we select target statistics $(\bar{\boldsymbol{\mu}}_b^t, \bar{\boldsymbol{\sigma}}_b^t)$ from either class-wise or global memory depending on reliability, and inject them back to obtain style-transferred source features:

$$\tilde{\mathbf{F}}_{b,c,h,w}^{s\to t} = \hat{\mathbf{F}}_{b,c,h,w}^{s} \cdot \bar{\boldsymbol{\sigma}}_{b,c}^t + \bar{\boldsymbol{\mu}}_{b,c}^t. \tag{3}$$

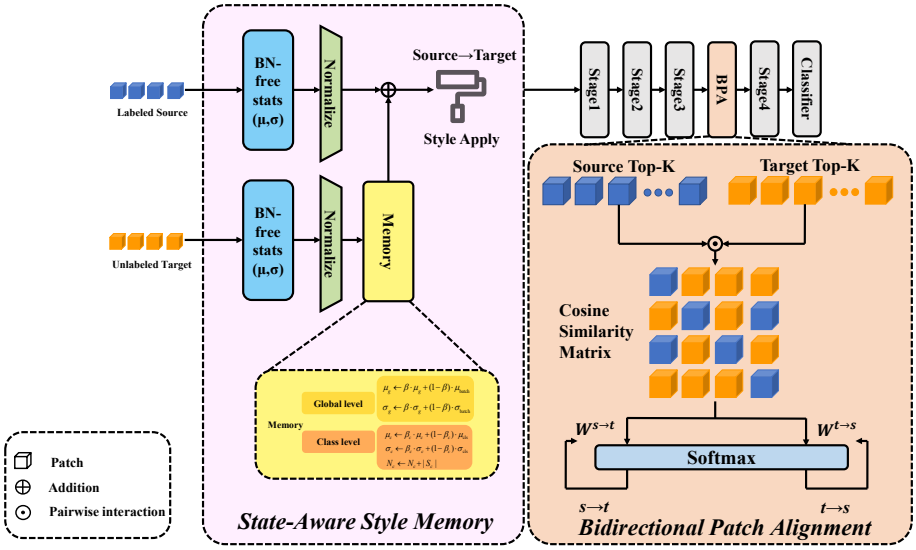

Fig. 1. Overview of the proposed DAMamba framework for unsupervised domain adaptation with Mamba-based vision models. (a) **State-Aware Style Memory (SASM)** regulates shallow-layer feature statistics by modeling target-domain styles and injecting them into source features, mitigating style drift before state-space recurrence. (b) **Bidirectional Patch Alignment (BPA)** enforces fine-grained semantic correspondence between salient source and target patches to alleviate local misalignment under domain shift. The proposed framework explicitly addresses the vulnerabilities of recurrent state space models and can be seamlessly integrated into VMamba backbones with negligible inference overhead.

This **source-to-target** adaptation mitigates style drift at shallow layers while preserving VMamba's linear-time efficiency. *Remark.* The formulation can be extended to bidirectional transfer, but in this work we instantiate the $s \to t$ direction for stability and negligible overhead.

B. Bidirectional Patch Alignment

While SASM aligns global style statistics and mitigates state drift at shallow layers, local semantic mismatches can still persist at higher levels. To address this, we therefore propose a Bidirectional Patch Alignment (BPA) built on top of the third stage of the VMamba backbone. The core of BPA is a Bi-directional Soft Matching (BSM) loss that enforces fine-grained alignment between semantic patches of the two domains. We refer to this cross-domain patch-level loss as the patch-level adaptation loss and denote it by L_{bsm}.

Saliency-driven Patch Construction. Let $F_s^{(3)}, F_t^{(3)} \in \mathbb{R}^{B\times C\times H'\times W'}$ denote the high-level features from the third stage of the VMamba backbone for source and target domains, respectively. We flatten the spatial dimensions into patch sequences $Z_d \in \mathbb{R}^{B\times L\times D}$ with $L = H'W'$ and $D = C$. For each patch feature $z_{b,\ell}^d$, we compute a saliency score $r_{b,\ell}^d = \|z_{b,\ell}^d\|_2$ and select the top $K = \lfloor \rho L \rfloor$ salient patches per image to form $P_d \in \mathbb{R}^{B\times K\times D}$, focusing subsequent alignment on semantically important regions. When class information is used, patches are further partitioned into class-wise sets and class weights are defined from joint source–target frequencies.

Bidirectional Soft Matching Loss. For a generic pair of source and target patch sets (either globally or class-wise), let $S = \{s_i\}_{i=1}^{N_s}$ and $T = \{t_j\}_{j=1}^{N_t}$, where $s_i, t_j \in \mathbb{R}^D$ are patch features. BPA first ℓ_2-normalizes patch features, $\tilde{s}_i = s_i/\|s_i\|_2$ and $\tilde{t}_j = t_j/\|t_j\|_2$, and builds a cosine-similarity matrix $M_{ij} = \tilde{s}_i^\top \tilde{t}_j$. The forward and backward correspondence weights are given by

$$W_{ij}^{s\to t} = \frac{\exp(M_{ij}/\tau)}{\sum_{j'}\exp(M_{ij'}/\tau)}, W_{ji}^{t\to s} = \frac{\exp(M_{ij}/\tau)}{\sum_{i'}\exp(M_{i'j}/\tau)}. \tag{4}$$

The forward weights yield aggregated target patches $\hat{t}_i = \sum_j W_{ij}^{s\to t} t_j$, and the backward weights produce aggregated source patches $\hat{s}_j = \sum_i W_{ji}^{t\to s} s_i$.

The bi-directional soft matching loss for a pair of patch sets is then defined as

$$L_{\text{bsm}} = 1 - \frac{1}{2}\left(\frac{1}{N_s}\sum_{i=1}^{N_s}\cos\left(s_i, \hat{t}_i\right) + \frac{1}{N_t}\sum_{j=1}^{N_t}\cos\left(t_j, \hat{s}_j\right)\right), \tag{5}$$

where $\cos(\cdot,\cdot)$ denotes cosine similarity. In the class-wise case, we compute L_{bsm} on class-specific patch sets and aggregate over all classes with non-empty source and target patches using class weights derived from joint frequencies.

The two directions $s \to t$ and $t \to s$ act as mutual constraints, which reduces the risk of aligning semantic patches to background or noisy regions in either domain. Combined with saliency-driven patch selection and class-wise weighting, BPA concentrates the adaptation on discriminative and shared structures, leading to a more stable and effective patch-level alignment.

C. Overall Objective

Given a source mini-batch (x^s, y^s) and a target mini-batch x^t, the overall training objective of DAMamba combines source classification, SASM regularization, and BPA. Let $\hat{y}^s =$

TABLE I
CLASSIFICATION ACCURACY (%) ON OFFICE-HOME FOR UDA. WE REPORT RESULTS ON 12 TRANSFER TASKS AND THEIR MEAN ACCURACY (AVG).

Method	Backbone	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
DoT [28]	ResNet	52.9	75.2	80.2	66.0	77.7	78.9	66.0	50.5	82.0	68.4	51.2	82.0	69.3
BDPN [29]	ResNet	52.6	74.1	78.5	66.1	71.2	74.4	66.4	55.4	81.5	74.6	60.3	82.3	69.8
DRLC [30]	ResNet	56.2	77.8	81.8	67.2	76.1	75.9	63.9	54.1	81.5	74.6	60.9	85.0	71.2
DiaNA [31]	ResNet	58.0	79.6	82.2	69.3	79.3	78.5	66.7	56.1	81.8	74.6	60.8	85.3	72.0
DARE-GRAM [32]	ResNet	59.3	76.2	81.4	67.4	78.1	76.3	66.7	57.4	80.5	75.9	61.3	84.7	72.2
DACDM [33]	ResNet	60.4	78.8	82.7	69.6	80.5	79.6	70.5	65.2	83.1	75.8	64.2	85.6	73.6
GSDE [34]	ResNet	57.8	80.2	81.9	71.3	78.9	80.5	67.4	57.2	84.0	76.1	62.5	85.7	73.6
DAD [35]	ResNet	62.5	78.6	83.0	70.4	79.2	79.8	70.2	58.3	83.1	76.3	63.5	88.2	74.4
TCPL [36]	ResNet	61.2	80.5	82.8	68.8	75.1	76.5	71.7	59.8	83.5	78.1	66.2	87.6	74.3
EIDCo [37]	ResNet	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
ViT-B [38]	ViT	54.7	83.0	87.2	77.3	83.4	85.6	74.4	50.9	87.2	79.6	54.8	88.8	75.5
DeiT-B [39]	DeiT	61.8	79.5	84.3	75.4	78.8	81.2	72.8	55.7	84.4	78.3	59.3	86.0	74.8
VMamba-B [25]	VMamba	63.4	79.6	84.4	75.3	80.9	81.0	71.0	57.4	84.1	78.7	60.3	87.3	75.3
DAMamba-B	VMamba	67.0	80.7	84.8	78.1	81.9	82.6	73.7	63.5	85.4	79.9	68.4	88.9	**77.9**

$f_\theta(x^s)$ be the source predictions and $p(\hat{y}_i^s = y_i^s)$ the predicted probability of the correct class for the i-th source sample. The source classification loss is the standard cross-entropy $L_{\text{cls}} = -\frac{1}{B_s}\sum_{i=1}^{B_s} \log p(\hat{y}_i^s = y_i^s)$. To improve pseudo-label quality (hence more reliable memory), we apply an information-maximization regularizer to encourage confident yet diverse predictions (e.g., entropy minimization plus batch diversity). The final objective is

$$L = L_{\text{cls}} + \lambda_{\text{im}} L_{\text{im}} + \lambda_{\text{bsm}} L_{\text{bsm}}, \tag{6}$$

where L_{im} is the information-maximization regularizer in SASM and L_{bsm} is the patch-level adaptation loss in BPA. The VMamba backbone and both adaptation modules are optimized end-to-end under (6), yielding improved robustness and generalization on the target domain with negligible additional inference cost.

IV. EXPERIMENTS

Datasets and Benchmarks. 1) *Office-31* is a classical UDA benchmark for office object recognition, containing 4,110 images from three domains: Amazon (A), DSLR (D), and Webcam (W), with 31 shared categories. Following the standard setting, we consider all six transfer tasks A→W, D→W, W→D, A→D, D→A, and W→A, where one domain is treated as labeled source and another as unlabeled target. 2) *Office-Home*is a more challenging dataset with 65 object categories and 15,500 images from four visually diverse domains: Art (Ar), Clipart (Cl), Product (Pr), and Real-World (Rw). We follow the widely-used UDA protocol and evaluate on all 12 transfer tasks (*e.g.,* Ar→Cl, Cl→Ar, *etc.*)

Implementation Details. We adopt VMamba as the backbone and use the Tiny, Small, and Base variants pre-trained on the ImageNet dataset to initialize. Our SASM is inserted before the first stage to regulate shallow-layer style statistics, while the BPA operates on high-level features at the third stage. Training images are resized to 256×256 and cropped to 224×224, using standard augmentations. Models are optimized with SGD (momentum 0.9, weight decay 5×10^{-4}) and a cosine learning rate schedule with an initial value in $1 \times 10^{-3}, 3 \times 10^{-4}$. Batch sizes follow prior work (32 for Office-31; 64 for Office-Home), and training runs for 30/50 epochs, respectively. Hyperparameters are fixed across datasets: EMA momentum $\beta = 0.9$, pseudo-label threshold $\tau_{\text{p}} = 0.9$, minimum class count $T = 50$, saliency ratio $\rho = 0.5$, and matching temperature $\tau = 0.05$. Loss weights $\lambda_{\text{im}} = 0.1$ and $\lambda_{\text{bsm}} = 1.0$.

TABLE II
CLASSIFICATION ACCURACY (%) ON OFFICE-31. WE REPORT RESULTS ON SIX TRANSFER TASKS AND THEIR AVERAGE ACCURACY (AVG).

Method	A→D	A→W	D→A	D→W	W→A	W→D	Avg
VMamba-T	86.5	82.0	64.8	96.9	64.5	99.6	82.4
DAMamba-T	92.0	89.8	71.5	98.9	70.5	**100.0**	87.1
VMamba-S	87.5	85.3	68.2	98.0	63.8	**100.0**	83.8
DAMamba-S	92.5	93.8	73.7	99.2	71.4	**100.0**	88.4
VMamba-B	89.2	88.2	70.5	98.5	71.0	**100.0**	86.2
DAMamba-B	**97.0**	**94.1**	**74.7**	**99.3**	**75.1**	**100.0**	**90.0**

A. Comparison with State-of-the-Art Methods

Results on Office-Home. Office-Home exhibits larger intra-class variation and stronger style discrepancies among the four domains (Ar, Cl, Pr, Rw), making it a challenging benchmark. As shown in Table I, DAMamba achieves clear gains over the VMamba baseline and achieves comparable results with state-of-the-art UDA methods on 12 transfer tasks, with the largest improvements observed on transfers involving the Clipart domain (*e.g.,* Ar→Cl, Pr→Cl). These improvements highlight the effectiveness of SASM in capturing diverse target styles and the ability of BPA to selectively align salient semantic patches instead of background clutter. Overall, DAMamba achieves the best mean accuracy on Office-Home, validating its robustness under complex domain shifts.

Results on Office-31. Table II summarizes the results over different VMamba baselines on Office-31. Across all six transfer tasks, DAMamba consistently outperforms the corresponding VMamba baseline. The improvement is especially pronounced on tasks involving the DSLR domain (D→W and W→D), where the limited number of DSLR images makes the model more vulnerable to overfitting on source styles. By explicitly aligning shallow-layer statistics through SASM and enforcing fine-grained patch-level alignment via BPA, DAMamba reduces state drift and local misalignment, leading to higher accuracy on both small and large domain shifts.

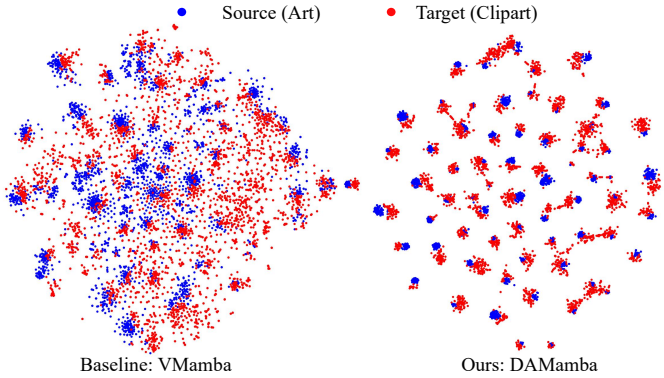

Fig. 2. Domain-wise t-SNE visualization of target features on Ar→Cl.

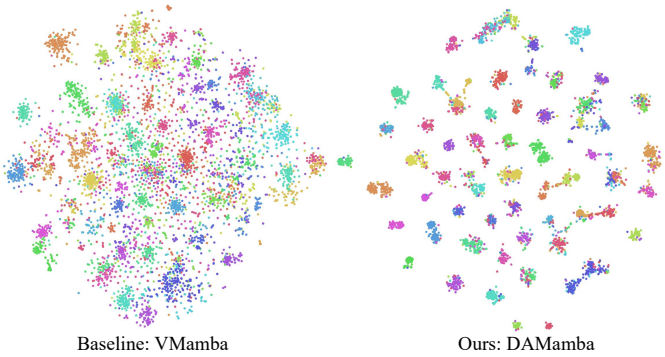

Fig. 3. Class-wise t-SNE visualization of target features on Ar→Cl.

TABLE III
ABLATION OF EACH PROPOSED MODULE WITH VMAMBA-BASE BACKBONE. TARGET-DOMAIN AVERAGE ACCURACY (%).

ID	Baseline	SASM	BPA	Office-Home
I	✓	-	-	75.3
II	✓	✓	-	77.3
III	✓	✓	✓	**77.9**

B. Ablation Studies.

To better understand the contribution of each component, we conduct detailed ablation studies on Office-Home.

Effect of Each Component. We evaluate: (i) **Baseline**: VMamba-Base with $\mathcal{L}_{cls}$ only; (ii) **Baseline + SASM**: adding SASM with $\mathcal{L}_{im}$; and (iii) **DAMamba-B**: full model with SASM + BPA optimized under Eq. (6). As summarized in Table III, introducing SASM on top of the baseline already brings noticeable gains, especially for tasks with strong style shifts, since it stabilizes state-space updates by aligning shallow-layer statistics and regularizing target predictions through L_{im}. Comparing with **Baseline + SASM**, adding BPA further shows consistent improvements across almost all transfer tasks, indicating that BPA provides additional benefits by enforcing semantic patch-level matching via the bi-directional soft matching loss L_{bsm}. Overall, combining SASM and BPA yields the best performance and consistently surpasses all partial variants, confirming that global style alignment at shallow layers and local semantic alignment at high-level stages are complementary.

Effect of Backbone Capacity. We further investigate how DAMamba behaves under different backbone capacities on Office-31. Specifically, we instantiate DAMamba on VMamba-Tiny, VMamba-Small, and VMamba-Base, and report the performance of each variant. Results are summarized in Table II. Several observations can be made. First, increasing backbone capacity generally improves the performance of DAMamba, though the gains diminish from Small to Base. Second, the relative improvement from Tiny to Small is more pronounced than that from Small to Base, suggesting that DAMamba can effectively compensate for limited capacity by providing more informative cross-domain regularization. This property is particularly attractive for resource-constrained deployment scenarios.

TABLE IV
COMPARISON OF THE COMPUTATIONAL EFFICIENCY OF DIFFERENT UDA METHODS ON OFFICE-HOME AR→CL . ALL MODELS ARE TESTED ON ONE NVIDIA GEFORCE RTX 4090 D GPU.

Model	Backbone	Params (M)	GFLOPs (G)	Time (ms)
SHOT	ResNet-50	24	4	2
DeiT-B	DeiT-B	86	18	6
VMamba-T	VMamba-T	30	5	4
DAMamba-T	VMamba-T	30	5	4

C. Visualization and Analysis

Domain-level T-SNE Visualization. Fig. 2 visualizes target-domain features for the Ar→Cl task using t-SNE, with points colored by domain. The VMamba baseline exhibits a clear separation between source (Art) and target (Clipart) samples, indicating substantial domain discrepancy in the learned representation space. In contrast, DAMamba produces a noticeably more overlapping and compact distribution across domains, suggesting improved global alignment between source and target features. This indicates improved global domain alignment, mainly attributed to SASM suppressing shallow-layer style drift before state-space recurrence.

Class-level T-SNE Visualization. Fig. 3 shows t-SNE visualizations of the same features as Fig. 2, but colored by semantic class labels. For the VMamba baseline, samples from different classes exhibit significant overlap and fragmented clusters, especially for visually similar categories under the Clipart style. In comparison, DAMamba yields more compact and well-separated class clusters. Instances belonging to the same class are grouped more tightly, while different classes become more distinguishable in the projected space. This improvement is consistent with BPA enforcing fine-grained semantic correspondence between source and target patches.

Analysis of Computational Efficiency. We evaluate computational efficiency on Office-Home Ar→Cl with input size 224×224 using a single NVIDIA RTX 4090 GPU. As shown in Table IV, DAMamba matches the VMamba baseline in terms of parameters, FLOPs, and inference latency, indicating that the proposed modules do not introduce noticeable overhead. This efficiency stems from the lightweight design of SASM and BPA: SASM involves only channel-wise statistics and memory lookup, while BPA aligns a limited number of salient patches rather than dense spatial tokens. All memory

updates and pseudo-label related operations are applied only during training, and DAMamba reduces to a standard VMamba forward pass at inference time, preserving the linear-time complexity of state space models.

V. Conclusion

We proposed **DAMamba**, an unsupervised domain adaptation framework tailored for Mamba-style visual state space models. By introducing a State-Aware Style Memory to suppress shallow-layer style drift and a Bidirectional Patch Alignment module to enforce fine-grained semantic correspondence, DAMamba effectively addresses the unique vulnerabilities of recurrent state-space backbones under domain shift. Extensive experiments demonstrate consistent improvements over strong baselines with negligible inference overhead.

References

[1] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan, "Learning transferable features with deep adaptation networks," in *ICML*, 2015, pp. 97–105.

[2] Hongliang Yan, Yukang Ding, Peihua Li, Qilong Wang, Yong Xu, and Wangmeng Zuo, "Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation," in *CVPR*, 2017, pp. 2272–2281.

[3] Djebril Mekhazni, Amran Bhuiyan, George Ekladious, and Eric Granger, "Unsupervised domain adaptation in the dissimilarity space for person re-identification," in *ECCV*, 2020, pp. 159–174.

[4] Baochen Sun and Kate Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *ECCV*, 2016, pp. 443–450.

[5] Li Ma, Melba M Crawford, Lei Zhu, and Yong Liu, "Centroid and covariance alignment-based domain adaptation for unsupervised classification of remote sensing images," *IEEE TGRS*, vol. 57, no. 4, pp. 2305–2323, 2018.

[6] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada, "Maximum classifier discrepancy for unsupervised domain adaptation," in *CVPR*, 2018, pp. 3723–3732.

[7] Zhekai Du, Jingjing Li, Hongzu Su, Lei Zhu, and Ke Lu, "Cross-domain gradient discrepancy minimization for unsupervised domain adaptation," in *CVPR*, 2021, pp. 3937–3946.

[8] Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann, "Contrastive adaptation network for unsupervised domain adaptation," in *CVPR*, 2019, pp. 4893–4902.

[9] Yaroslav Ganin and Victor Lempitsky, "Unsupervised domain adaptation by backpropagation," in *ICML*, 2015, pp. 1180–1189.

[10] Weichen Zhang, Wanli Ouyang, Wen Li, and Dong Xu, "Collaborative and adversarial network for unsupervised domain adaptation," in *CVPR*, 2018, pp. 3801–3809.

[11] Lihua Zhou, Mao Ye, Xiatian Zhu, Shuaifeng Li, and Yiguang Liu, "Class discriminative adversarial learning for unsupervised domain adaptation," in *ACM MM*, 2022, pp. 4318–4326.

[12] Lin Chen, Huaian Chen, Zhixiang Wei, Xin Jin, Xiao Tan, Yi Jin, and Enhong Chen, "Reusing the task-specific classifier as a discriminator: Discriminator-free adversarial domain adaptation," in *CVPR*, 2022, pp. 7181–7190.

[13] Xiaofeng Liu, Zhenhua Guo, Fangxu Xing, Jane You, C-C Jay Kuo, Georges El Fakhri, and Jonghye Woo, "Adversarial unsupervised domain adaptation with conditional and label shift: Infer, align and iterate," in *ICCV*, 2021, pp. 10367–10376.

[14] Weili Shi, Ronghang Zhu, and Sheng Li, "Pairwise adversarial training for unsupervised class-imbalanced domain adaptation," in *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, 2022, pp. 1598–1606.

[15] Ao Ma, Jingjing Li, Ke Lu, Lei Zhu, and Heng Tao Shen, "Adversarial entropy optimization for unsupervised domain adaptation," *IEEE TNNLS*, vol. 33, no. 11, pp. 6263–6274, 2021.

[16] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez, "Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation," in *CVPR*, 2019, pp. 2517–2526.

[17] Kuniaki Saito, Donghyun Kim, Stan Sclaroff, Trevor Darrell, and Kate Saenko, "Semi-supervised domain adaptation via minimax entropy," in *ICCV*, 2019, pp. 8050–8058.

[18] Yang Zou, Zhiding Yu, Xiaofeng Liu, BVK Kumar, and Jinsong Wang, "Confidence regularized self-training," in *ICCV*, 2019, pp. 5982–5991.

[19] Hong Liu, Jianmin Wang, and Mingsheng Long, "Cycle self-training for domain adaptation," *NeurIPS*, vol. 34, pp. 22968–22981, 2021.

[20] Ke Mei, Chuang Zhu, Jiaqi Zou, and Shanghang Zhang, "Instance adaptive self-training for unsupervised domain adaptation," in *ECCV*, 2020, pp. 415–430.

[21] Luke Melas-Kyriazi and Arjun K Manrai, "Pixmatch: Unsupervised domain adaptation via pixelwise consistency training," in *CVPR*, 2021, pp. 12435–12445.

[22] Cheng-An Hou, Yao-Hung Hubert Tsai, Yi-Ren Yeh, and Yu-Chiang Frank Wang, "Unsupervised domain adaptation with label and structural consistency," *IEEE TIP*, vol. 25, no. 12, pp. 5552–5562, 2016.

[23] Viraj Prabhu, Shivam Khare, Deeksha Kartik, and Judy Hoffman, "Sentry: Selective entropy optimization via committee consistency for unsupervised domain adaptation," in *ICCV*, 2021, pp. 8558–8567.

[24] Albert Gu and Tri Dao, "Mamba: Linear-time sequence modeling with selective state spaces," in *COLM*, 2024.

[25] Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, Jianbin Jiao, and Yunfan Liu, "Vmamba: Visual state space model," *NeurIPS*, vol. 37, pp. 103031–103063, 2024.

[26] Lianghui Zhu, Bencheng Liao, Qian Zhang, Xinlong Wang, Wenyu Liu, and Xinggang Wang, "Vision mamba: Efficient visual representation learning with bidirectional state space model," in *ICML*, 2024, pp. 62429–62442.

[27] Xiao Liu, Chenxu Zhang, Fuxiang Huang, Shuyin Xia, Guoyin Wang, and Lei Zhang, "Vision mamba: A comprehensive survey and taxonomy," *IEEE TNNLS*, 2025.

[28] Chuan-Xian Ren, Yiming Zhai, You-Wei Luo, and Hong Yan, "Towards unsupervised domain adaptation via domain-transformer," *IJCV*, vol. 132, no. 12, pp. 6163–6183, 2024.

[29] Yuhang He, Junzhe Chen, Jiehua Zhang, Wei Ke, and Yihong Gong, "A bayesian dual-pathway network for unsupervised domain adaptation," *PR*, vol. 164, pp. 111498, 2025.

[30] Shanshan Wang, Yiyang Chen, Zhenwei He, Xun Yang, Mengzhu Wang, Quanzeng You, and Xingyi Zhang, "Disentangled representation learning with causality for unsupervised domain adaptation," in *ACM MM*, 2023, pp. 2918–2926.

[31] Duojun Huang, Jichang Li, Weikai Chen, Junshi Huang, Zhenhua Chai, and Guanbin Li, "Divide and adapt: Active domain adaptation via customized learning," in *CVPR*, 2023, pp. 7651–7660.

[32] Ismail Nejjar, Qin Wang, and Olga Fink, "Dare-gram: Unsupervised domain adaptation regression by aligning inverse gram matrices," in *CVPR*, 2023, pp. 11744–11754.

[33] Yulong Zhang, Shuhao Chen, Weisen Jiang, Yu Zhang, Jiangang Lu, and James T Kwok, "Domain-guided conditional diffusion model for unsupervised domain adaptation," *Neural Networks*, vol. 184, pp. 107031, 2025.

[34] Thomas Westfechtel, Hao-Wei Yeh, Dexuan Zhang, and Tatsuya Harada, "Gradual source domain expansion for unsupervised domain adaptation," in *WACV*, 2024, pp. 1946–1955.

[35] Duo Peng, Qiuhong Ke, ArulMurugan Ambikapathi, Yasin Yazici, Yinjie Lei, and Jun Liu, "Unsupervised domain adaptation via domain-adaptive diffusion," *IEEE TIP*, vol. 33, pp. 4245–4260, 2024.

[36] Junyu Gao, Xinhong Ma, and Changsheng Xu, "Learning transferable conceptual prototypes for interpretable unsupervised domain adaptation," *IEEE TIP*, vol. 33, pp. 5284–5297, 2024.

[37] Yixin Zhang, Zilei Wang, Junjie Li, Jiafan Zhuang, and Zihan Lin, "Towards effective instance discrimination contrastive loss for unsupervised domain adaptation," in *ICCV*, 2023, pp. 11388–11399.

[38] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *ICLR*, 2021.

[39] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou, "Training data-efficient image transformers & distillation through attention," in *ICML*, 2021, pp. 10347–10357.