Dance Across Shifts: Forward-Facilitation Continual Test-Time Adaptation through Dynamic Style Bridging

Supplementary Material

In the supplementary material, we provide more analysis and experiments to enhance the understanding of our work. In Sec. A, we provide comprehensive information about different CTTA baselines and implementations involved in our main paper. Sec. B provides a detailed illustration of symbols used in paradigm comparison. Sec. C shows details and compatibility of our framework across self-training objectives. In Sec. D, we provide additional experimental results to further validate the generalizability of our method. Finally, we present visualizations of synthetic images in Sec. E.

A. Details of Comparison Methods

In this section, we provide details of the CTTA methods included in our comparisons of the main paper.

TENT[1] [16] makes all affine parameters of normalization layers trainable through the entropy minimization loss. We follow all hyperparameters that are set in TENT unless it does not provide.

CoTTA[2] [18] represents the first method to perform TTA on continually changing domains. For its augmentation-based consistency maximization and stochastic parameter restoration, we maintain identical hyperparameters as specified in CoTTA. The trainable parameters encompass all parameters within ViT-Base.

EATA[3] [12] selectively performs test-time optimization with relatively reliable and non-redundant samples. For its thresholds of entropy filter and cosine similarities, we configure the same hyperparameters. Similar to TENT, the trainable parameters comprise all affine parameters of normalization layers.

RMT[4] [1] adopts a mean-teacher framework with a symmetric cross-entropy loss, combined with contrastive learning to align with source representative features. For its model optimization and updating process, we configure the same hyperparameters. The trainable parameters are all the parameters in ViT-Base.

CMAE[5] [8] propose adaptive distribution masked autoencoders to enhance the extraction of target domain knowledge while mitigating the accumulation of distribution shifts. We follow the same uncertain calculation method and masking strategy.

OBAO[6] [23] dynamically identifies and aggregates high-confidence target samples during test time, combined with a class relation preservation constraint to organize these samples. For its dynamical buffer, we follow the same hyperparameters.

DPCore[7] [22] utilizes a prompt coreset which dynamically manages domain knowledge to match the statistical prior in source domain. The trainable parameters are the introduced prompts. We follow all hyperparameters unless it does not provide.

REM[8] [7] proposes ranked entropy minimization to mitigate the stability problem. A progressive masking strategy is introduced to explicitly structure the prediction difficulty. We follow the same masking strategy to build the mask chain.

DDA[9] [4] projects all test inputs toward the source domain with a generative diffusion model. It deals with distribution shifts at the input level. We follow all hyperparameters and use the same diffusion model.

SDA[10] [6] is similar to DDA in terms of overall process. However, it further fine-tunes the source model in the synthesis domain. We utilize the same stable diffusion model [14] of our method to generate training samples, and further refine them via an unconditional diffusion model.

A.1. Experimental protocols

The pretrained model is ViT-B/16 [2], trained on the ImageNet-1K training set at a resolution of 224×224, with weights directly obtained from the `timm` repository. The same data preprocessing configuration is adopted to maintain consistency. For the CIFAR-based experiments, we further fine-tune the model on the respective source domains to obtain task-specific source models. To ensure fairness, all methods are initialized with the same pre-trained weights for each dataset, and the batch size is fixed to 50. We utilize official implementations of the method where available. Note that some methods only provide implementations on convolutional neural networks, at which point we prioritize the reproduction with the help of existing methods [7, 9]. If not, we reproduce it ourselves using the hyperparameters reported in the original paper. Optimization is performed using Adam with $(\beta_1, \beta_2) = (0.9, 0.999)$ and the learning

[1] https://github.com/DequanWang/tent
[2] https://github.com/qinenergy/cotta
[3] https://github.com/mr-eggplant/EATA
[4] https://github.com/mariodoebler/test-time-adaptation
[5] https://github.com/RanXu2000/continual-mae
[6] https://github.com/z1358/OBAO
[7] https://github.com/yunbeizhang/DPCore
[8] https://github.com/pilsHan/rem
[9] https://github.com/shiyegao/DDA
[10] https://github.com/SHI-Labs/Diffusion-Driven-Test-Time-Adaptation-via-Synthetic-Domain-Alignment

Table 6. Detailed explanation of graphical symbols used in paradigm comparison.

Sub-figure	Symbol/Element	Technical Interpretation
(b)	Network icon	Parameter regularization (constraints or recovery on source weights)
	Star ($\star$)	Source-domain class prototypes (feature centroids) used for representation alignment
	$\{\mu^s, \sigma^s\}$	Source feature statistics (mean and variance) / BN statistics
	Align arrow	Backward alignment that constrains the target model using source priors
(c)	$x^t \rightarrow x^K$	Projecting a target input back to a static synthetic domain
	Denoise arrow	The process of removing "domain shift" as noise
(d)	$\{x^K, y^K\}$	Explicit synthetic image-label pairs (Knowledge Base)
	Co-evolve arrow	Actively injecting target styles into synthetic knowledge (Bridging)

rate is chosen from $\{1e-3, 1e-4, 1e-5\}$ to achieve the proper magnitude.

A.2. More Implementation Details

Our approach follows the experimental protocols outlined in Sec. A.1 to ensure consistency and comparability. For the construction of our synthetic knowledge base, we utilize Stable Diffusion 1.5 [14] with 50 denoising steps, generating 2000 synthetic images per CTTA task. Importantly, our approach does not require any interaction with the diffusion model during test time. Following the prior method [23], we sample a batch from the knowledge base with the same size as the target batch B_t at each time step for loss computation. An Adam optimizer with a learning rate of 1e-5 is used to optimize the model. For the teacher-student framework involved in the self-training loss, since our forward-facilitation paradigm can provide the necessary supervision information for model adaptation, we use a larger momentum of 0.9 to update the teacher model through the exponential moving average. All experiments of our method in the main paper during test time are conducted using an NVIDIA RTX4090 GPU.

B. Detailed Illustration of Symbols in Fig. 1

Fig. 1 in the main paper illustrates the conceptual distinctions between the prevailing backward-alignment paradigm and our proposed forward-facilitation paradigm. To ensure a precise understanding of the visual schematic, we provide a detailed explanation of the key graphical symbols appearing in sub-figures (b)-(d). We summarize these elements in Tab. 6. The provided definitions bridge the conceptual illustration with the methodological details in the main text, enabling readers to interpret the figure accurately and without ambiguity.

C. Details of Self-Training Loss

Here, we provide the detailed formulation of the self-training loss employed in our online adaptation process.

Table 7. Compatibility with different self-training objectives. We report the average error rate (%, lower is better) of the baseline and our method. Consistent improvements are yielded in both settings.

Self-training	Baseline	Ours	$\Delta \downarrow$
Entropy minimization	49.8	44.8	-5.0
Teacher-student	50.0	**44.1**	**-5.9**

Following the standard teacher-student self-training practice [1, 11, 23], we enforce prediction consistency between the teacher and student models on the unlabeled target batch B_t using a symmetric cross-entropy objective:

$$\mathcal{L}_{ST} = -\sum_{c=1}^{C} q_c \log p_c - \sum_{c=1}^{C} p_c \log q_c, \tag{1}$$

where q and p denote the softmax predictions of the teacher and student models, respectively, and C represents the number of classes. Both the teacher model and the student model are initialized with the source model f_θ. Subsequently, the teacher model is continuously updated by the exponential moving average of the student model.

During adaptation, $\mathcal{L}_{ST}$ is applied exclusively to the unlabeled target data to maintain temporal stability, while the synthetic samples are supervised by our proposed semantic learning objectives. Since the self-training mechanism follows a well-established paradigm and is not the conceptual focus of our contribution, we omit detailed discussion in the main paper and provide its formulation here for reference and reproducibility.

Compatibility across self-training objectives. While we adopt the teacher-student self-training in our main implementation due to its stability, the core contribution of our work, the forward-facilitation via dynamic style bridging, is conceptually orthogonal to the specific choice of the auxiliary self-training objective on the target data. To empirically validate this, we conduct additional experiments replacing

Table 8. Classification error rate (%, lower is better) for the standard CTTA task on ImageNet-to-ImageNetC. We follow the experimental protocol established by DPCore. Bold text indicates the best.

	Time	$t \longrightarrow$															
Method	Venue	*Gaussian*	*shot*	*impulse*	*defocus*	*glass*	*motion*	*zoom*	*snow*	*frost*	*fog*	*brightness*	*contrast*	*elastic*	*pixelate*	*jpeg*	Mean↓
Source	ICLR'21	53.0	51.8	52.1	68.5	78.8	58.5	63.3	49.9	54.2	57.7	26.4	91.4	57.5	38.0	36.2	55.8
TENT	ICLR'21	52.2	48.9	49.2	65.8	73.0	54.5	58.4	44.0	47.7	50.3	23.9	72.8	55.7	34.4	33.9	51.0
CoTTA	CVPR'22	52.9	51.6	51.4	68.3	78.1	67.1	62.0	48.2	52.7	55.3	25.9	90.0	56.4	36.4	35.2	54.8
VDP	AAAI'23	52.7	51.6	50.1	58.1	70.2	56.1	58.1	42.1	46.1	45.8	23.6	70.4	54.9	34.5	36.1	50.0
EcoTTA	CVPR'23	48.1	45.6	46.3	56.5	67.1	50.4	57.1	41.3	44.5	43.8	24.1	71.6	54.8	34.1	34.8	48.0
CMAE	CVPR'24	46.3	41.9	42.5	51.4	54.9	43.3	40.7	34.2	35.8	64.3	23.4	60.3	37.5	29.2	31.4	42.5
DPCore	ICML'25	42.2	38.7	39.3	47.2	51.4	47.7	46.9	39.3	36.9	37.4	22.0	44.4	45.1	30.9	29.6	39.9
PAID	NeurIPS'25	48.8	43.7	44.4	49.4	49.6	47.3	44.2	37.5	39.4	42.1	25.2	50.0	39.3	35.5	36.5	42.2
Ours	Proposed	40.0	38.0	38.3	41.9	52.4	35.2	43.8	31.5	34.7	28.1	24.6	33.2	37.5	31.6	30.6	**36.1**

Table 9. Comparison results of standard CIFAR100-to-CIFAR100C and CIFAR10-to-CIFAR10C CTTA tasks. We report the mean classification error rate (%, lower is better) across all 15 corrupted domains. All results are evaluated with the largest corruption severity level 5 in an online manner. We follow the experimental protocol established by DPCore. Bold text indicates the best.

		Time	$t \longrightarrow$															
	Method	Venue	*Gaussian*	*shot*	*impulse*	*defocus*	*glass*	*motion*	*zoom*	*snow*	*frost*	*fog*	*brightness*	*contrast*	*elastic*	*pixelate*	*jpeg*	Mean↓
CIFAR100C	Source	ICLR'21	55.0	51.5	26.9	24.0	60.5	29.0	21.4	21.1	25.0	35.2	11.8	34.8	43.2	56.0	35.9	35.4
	TENT	ICLR'21	53.0	47.0	24.6	22.3	58.5	26.5	19.0	21.0	23.0	30.1	11.8	25.2	39.0	47.1	33.3	32.1
	CoTTA	CVPR'22	55.0	51.3	25.8	24.1	59.2	28.9	21.4	21.0	24.7	34.9	11.7	31.7	40.4	55.7	35.6	34.8
	VDP	AAAI'23	54.8	51.2	25.6	24.2	59.1	28.8	21.2	20.5	23.3	33.8	7.5	11.7	32.0	51.7	35.2	32.0
	ViDA	ICLR'24	50.1	40.7	22.0	21.2	45.2	21.6	16.5	17.9	16.6	25.6	11.5	29.0	29.6	34.7	27.1	27.3
	CMAE	CVPR'24	48.6	30.7	18.5	21.3	38.4	22.2	17.5	19.3	18.0	24.8	13.1	27.8	31.4	35.5	29.5	26.4
	DPCore	ICML'25	48.2	40.2	21.3	20.2	44.1	21.1	16.2	18.1	15.2	22.3	9.4	13.2	28.6	32.8	25.5	25.1
	PAID	NeurIPS'25	40.7	31.9	20.4	19.8	35.9	23.0	16.3	20.5	18.2	25.3	12.6	19.8	29.4	28.2	31.3	24.9
	Ours	Proposed	35.2	29.8	19.4	19.1	36.0	19.5	17.2	15.8	16.7	18.6	11.6	13.0	27.6	25.0	27.9	**22.2**
CIFAR10C	Source	ICLR'21	60.1	53.2	38.3	19.9	35.5	22.6	18.6	12.1	12.7	22.8	5.3	49.7	23.6	24.7	23.1	28.2
	TENT	ICLR'21	57.7	56.3	29.4	16.2	35.3	16.2	12.4	11.0	11.6	14.9	4.7	22.5	15.9	29.1	19.5	23.5
	CoTTA	CVPR'22	58.7	51.3	33.0	20.1	34.8	20.0	15.2	11.1	11.3	18.5	4.0	34.7	18.8	19.0	17.9	24.6
	VDP	AAAI'23	57.5	49.5	31.7	21.3	35.1	19.6	15.1	10.8	10.3	18.1	4.0	27.5	18.4	22.5	19.9	24.1
	ViDA	ICLR'24	52.9	47.9	19.4	11.4	31.3	13.3	7.6	7.6	9.9	12.5	3.8	26.3	14.4	33.9	18.2	20.7
	CMAE	CVPR'24	30.6	18.9	11.5	10.4	22.5	13.9	9.8	6.6	6.5	8.8	4.0	8.5	12.7	9.2	14.4	12.6
	DPCore	ICML'25	22.0	18.2	14.9	14.3	24.4	13.9	12.0	11.6	10.7	15.0	5.7	21.8	15.6	12.7	18.0	15.4
	PAID	NeurIPS'25	22.9	11.8	9.9	9.1	16.7	10.8	7.4	7.4	6.6	11.4	4.5	9.3	12.8	9.4	14.5	11.0
	Ours	Proposed	15.0	10.1	7.7	6.8	12.6	5.8	4.7	4.8	4.5	4.5	2.9	3.7	8.4	5.2	9.0	**7.0**

the mean-teacher loss with entropy minimization [12], another widely adopted objective [7, 12, 16] that enforces confidence regularization during adaptation. In this case, our approach updates only the learnable parameters within the normalization layer, rather than the entire model.

As shown in Tab. 7, our framework consistently yields significant improvements over the baseline regardless of the self-training objective employed. Specifically, when integrated with entropy minimization [12], our method reduces the average error rate from 49.8% to 44.8%. This demonstrates that our proposed synthetic supervision serves as a universal complement to existing self-training techniques, offering robust guidance irrespective of the specific regularization applied to the target data.

Table 10. Semantic segmentation results (mIoU in %) on the Cityscapes-to-ACDC CTTA task. The four test conditions are repeated three times. All results are evaluated based on the Segformer-B5 architecture. Bold text indicates the best performance.

Time	$t \longrightarrow$															
Round	Round 1					Round 2					Round 3					All
Condition	*Fog*	*Night*	*Rain*	*Snow*	*Mean*	*Fog*	*Night*	*Rain*	*Snow*	*Mean*	*Fog*	*Night*	*Rain*	*Snow*	*Mean*	Mean↑
Source	69.1	40.3	59.7	57.8	56.7	69.1	40.3	59.7	57.8	56.7	69.1	40.3	59.7	57.8	56.7	56.7
TENT [16]	69.0	40.2	60.1	57.3	56.7	68.3	39.0	60.1	56.3	55.9	67.5	37.8	59.6	55.0	55.0	55.7
CoTTA [18]	70.9	41.2	62.4	59.7	58.6	70.9	41.1	62.6	59.7	58.6	70.9	41.0	62.7	59.7	58.6	58.6
VDP [3]	70.5	41.1	62.1	59.5	58.3	70.4	41.1	62.2	59.4	58.2	70.4	41.0	62.2	59.4	58.2	58.2
SAR [13]	69.0	40.2	60.1	57.3	56.7	69.0	40.3	60.0	67.8	59.3	67.5	37.8	59.6	55.0	55.0	57.0
ECoTTA [15]	68.5	35.8	62.1	57.4	56.0	68.3	35.5	62.3	57.4	55.9	68.1	35.3	62.3	57.3	55.8	55.8
SVDP [20]	72.1	44.0	65.2	63.0	61.1	72.2	44.5	65.9	63.5	61.5	72.1	44.2	65.6	63.6	61.4	61.3
OBAO [23]	71.2	42.3	65.0	62.0	60.1	72.6	43.2	66.3	63.2	61.3	72.8	43.8	66.5	63.2	61.6	61.0
Ours	71.6	43.7	66.6	64.7	**61.7**	71.2	44.6	67.1	64.4	**61.8**	72.0	44.9	67.7	63.9	**62.1**	**61.9**

Table 11. Statistical reliability analysis. Average online classification error rate (%) and standard deviation over 5 runs.

Benchmark	**Mean Error Rate (%)**
ImageNetC	44.15 ± 0.16
CIFAR100C	29.89 ± 0.13
CIFAR10C	9.16 ± 0.09

D. Additional Experimental Results

D.1. Statistical Reliability

To evaluate the statistical reliability of our method, we repeat the online adaptation process across five independent runs using different random seeds. As shown in Tab. 11, our method achieves consistent performance across runs, with exceptionally low standard deviations indicating strong stability under stochastic factors. We attribute this robustness to the co-evolving nature of our synthetic knowledge base. This continual evolution ensures a steady stream of reliable, context-aware supervision, effectively suppressing noise accumulation and guaranteeing stable adaptation over time.

D.2. Scaling to Different Model Weights

To further validate the generalizability and robustness of our proposed framework, we conduct comprehensive evaluations under different pre-trained weight configurations. Following DPCore [22], we utilize model weights pre-trained on ImageNet-21k and subsequently fine-tuned on ImageNet-1k. This configuration represents a more sophisticated initialization strategy that leverages broader visual knowledge from the extended ImageNet-21k corpus. We also add comparative methods such as VDP [3], ECoTTA [15], ViDA [9], and PAID [17]. For the CIFAR-based experiments, we utilize pre-trained source models provided by prior works [8, 9, 22] to ensure consistency and comparability.

The experimental results are presented in Tabs. 8 and 9, where our method employs identical hyperparameters without additional tuning. Remarkably, in the ImageNet-to-ImageNetC task, our method consistently outperforms all previous approaches, significantly reducing the average classification error rate across 15 domains to 36.1%, representing a relative improvement of 9.5% over the current SOTA method DPCore. This substantial improvement demonstrates the effectiveness of our forward-facilitation paradigm in handling diverse and challenging domain shifts. Furthermore, our approach shows particularly strong performance in difficult scenarios such as motion blur and fog noise, where traditional backward-alignment methods often struggle due to the significant stylistic variations these corruptions introduce. These results underscore the practical significance of our framework for real-world deployment scenarios, where models should adapt to unpredictable and diverse environmental conditions while maintaining robust performance.

D.3. Experiments on Segmentation CTTA

We extend our evaluation to dense prediction settings using the challenging continual test-time semantic segmentation task Cityscapes-to-ACDC. Following [18, 23], we employ Segformer-B5 [19] trained on Cityscapes as our segmentation model, and use down-sampled images from ACDC with a resolution of 960×540 as network inputs. Dense prediction tasks require pixel-level semantic annotations, which are not directly obtainable from standard text-to-image models. Therefore, we utilize a small number of synthetic samples from UrbanSyn [5] as the knowledge base. An Adam optimizer with a learning rate of 1e-4 is adopted, and the batch size is set to 1.

The experimental results are summarized in Tab. 10. We evaluate the performance across three rounds and report the

Table 12. Average online classification error rate (%) over 5 runs in the mixed domains TTA setting.

Avg. Error (%)	Source	TENT [16]	EATA [12]	CoTTA[18]	SAR [13]	RoTTA [21]	ROID [10]	Ours
ImageNetC	60.3	55.0	51.8	89.3	52.3	58.2	50.7	**44.6**±0.09

Table 13. Average error rate (%) across various challenging settings on ImageNet-to-ImageNetC.

Method	Class Imbalance	CDC	Varying Batch Size				
			64	8	4	2	1
DPCore	43.9	42.1	39.9	43.1	45.5	78.4	82.8
Ours	**37.6**	**36.7**	**36.1**	**37.1**	**37.8**	**40.1**	**43.4**

average mIoU metric for each domain and round, providing a comprehensive view of the adaptation effectiveness for repeated domain sequences. The proposed method demonstrates consistent improvement in mIoU across cycles (61.7 $\rightarrow$ 61.8 $\rightarrow$ 62.1), showcasing its capability for long-term adaptation to dynamic environments within dense prediction tasks.

D.4. Robustness to Mixed Domain Shifts

In real-world scenarios, distribution shifts may not always occur in a temporally continual or gradual manner. To assess the robustness of our framework under varying temporal dynamics, we conduct experiments in a mixed-domain setting. Following the protocol in [1, 10], test samples from all 15 ImageNetC corruptions are randomly shuffled to simulate a rapidly changing and unpredictable environment.

Since our style bridging mechanism operates at the instance level rather than relying on batch-level or temporal coherence, it naturally adapts to this setting. As shown in Tab. 12, our method achieves a remarkably low error rate of 44.6%, significantly outperforming the state-of-the-art ROID (50.7%). Notably, methods reliant on temporal continuity (*e.g.*, CoTTA) suffer catastrophic degradation due to negative transfer. In contrast, our performance remains stable compared with the ordered setting, confirming that the style bridging mechanism produces on-the-fly supervision independent of temporal history and ensures robust adaptation regardless of domain sequence.

D.5. More Challenging Settings

We add experiments under *class imbalance*, *CDC shifts*, and *varying batch sizes*. We strictly follow the data stream generation protocol and weight configuration of DPCore [22] to ensure fair comparison with SOTA. As shown in Tab. 13, our method is less batch-hungry and remains stable across all these challenging settings, consistently outperforming the SOTA method in different scenarios.

E. Visualization of Synthetic Data

We present visualizations of synthetic images for several ImageNet classes in Fig. 6. All images are randomly sampled rather than human-picked and are employed in our experiments. Comparison with real source-domain images reveals consistent patterns across these diverse generators. We observe that while the diffusion-based models generally yield higher fidelity than BigGAN, the synthetic samples across all three generators share a common advantage: they typically exhibit cleaner backgrounds and more prominent foreground subjects compared to the real source domain. This observation strongly validates the motivation behind our "semantically pure knowledge base", confirming its ability to provide explicit and unambiguous supervision signals. Nonetheless, we also acknowledge the inherent limitations of synthetic data. Compared to the real source domain, the generated samples exhibit a deficiency in diversity, particularly regarding variations in pose and appearance. Moreover, we observe strong model-intrinsic generative biases in certain categories, such as "koala bear" and "peacock", where the models tend to overfit to specific textures or canonical compositions.

Crucially, these observations further underscore the necessity of our dynamic style bridging mechanism. They illustrate that naively utilizing these biased samples as static anchors without proper adaptation would inevitably introduce noise and mislead the model to some extent. The consistent superior performance of our framework across multiple generative models demonstrates its ability to *disentangle* reliable semantic content from biased synthetic styles. By effectively bridging the gap between generative bias and target data, our approach circumvents reliance on any particular generative prior, ensuring robust and generalizable adaptation.

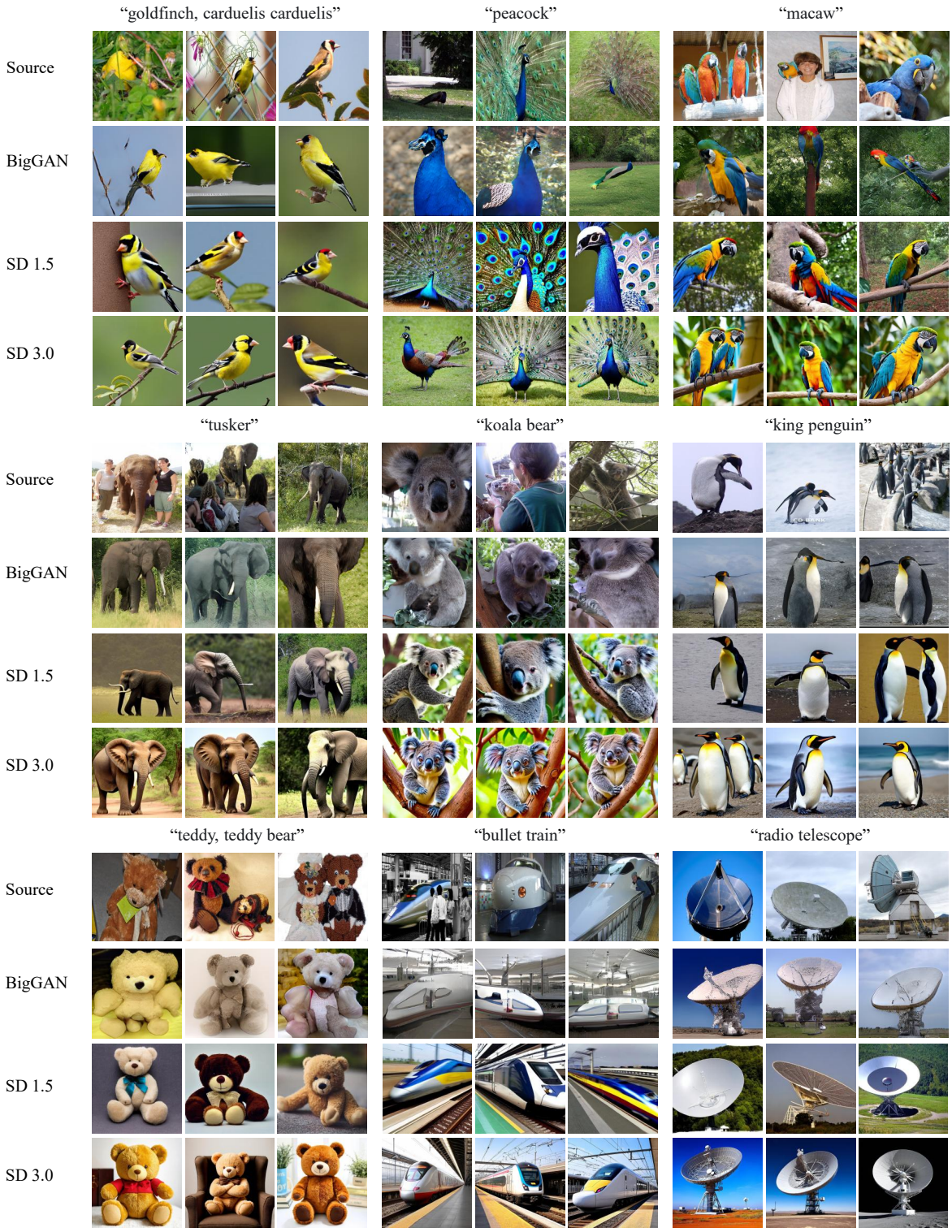

Figure 6. Visualization of synthetic images. We show source images (top row) followed by synthetic samples generated by BigGAN, Stable Diffusion 1.5, and Stable Diffusion 3.0. Each column group contains three randomly selected samples from the same class. Compared to source data, synthetic samples exhibit higher semantic purity (cleaner backgrounds and more salient objects) but also reveal generative bias such as texture overfitting. These observations highlight the necessity of our dynamic style bridging mechanism, which leverages reliable semantics while actively mitigating the inherent generative bias.

References

[1] Mario Döbler, Robert A. Marsden, and Bin Yang. Robust mean teacher for continual and gradual test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7704–7714, 2023. 1, 2, 5

[2] 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. In *International Conference on Learning Representations*, 2021. 1

[3] 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 *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 7595–7603, 2023. 4

[4] Jin Gao, Jialing Zhang, Xihui Liu, Trevor Darrell, Evan Shelhamer, and Dequan Wang. Back to the source: Diffusion-driven adaptation to test-time corruption. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11786–11796, 2023. 1

[5] Jose L Gómez, Manuel Silva, Antonio Seoane, Agnès Borrás, Mario Noriega, Germán Ros, Jose A Iglesias-Guitian, and Antonio M López. All for one, and one for all: Urbansyn dataset, the third musketeer of synthetic driving scenes. *Neurocomputing*, 637:130038, 2025. 4

[6] Jiayi Guo, Junhao Zhao, Chaoqun Du, Yulin Wang, Chunjiang Ge, Zanlin Ni, Shiji Song, Humphrey Shi, and Gao Huang. Everything to the synthetic: Diffusion-driven test-time adaptation via synthetic-domain alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 30503–30513, 2025. 1

[7] Jisu Han, Jaemin Na, and Wonjun Hwang. Ranked entropy minimization for continual test-time adaptation. In *International Conference on Machine Learning*, 2025. 1, 3

[8] Jiaming Liu, Ran Xu, Senqiao Yang, Renrui Zhang, Qizhe Zhang, Zehui Chen, Yandong Guo, and Shanghang Zhang. Continual-mae: Adaptive distribution masked autoencoders for continual test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28653–28663, 2024. 1, 4

[9] Jiaming Liu, Senqiao Yang, Peidong Jia, Renrui Zhang, Ming Lu, Yandong Guo, Wei Xue, and Shanghang Zhang. ViDA: Homeostatic visual domain adapter for continual test time adaptation. In *International Conference on Learning Representations*, 2024. 1, 4

[10] Robert A Marsden, Mario Döbler, and Bin Yang. Universal test-time adaptation through weight ensembling, diversity weighting, and prior correction. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2555–2565, 2024. 5

[11] Chenggong Ni, Fan Lyu, Jiayao Tan, Fuyuan Hu, Rui Yao, and Tao Zhou. Maintaining consistent inter-class topology in continual test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15319–15328, 2025. 2

[12] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *International Conference on Machine Learning*, pages 16888–16905. PMLR, 2022. 1, 3, 5

[13] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Zhiquan Wen, Yaofo Chen, Peilin Zhao, and Mingkui Tan. Towards stable test-time adaptation in dynamic wild world. In *International Conference on Learning Representations*, 2023. 4, 5

[14] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 1, 2

[15] Junha Song, Jungsoo Lee, In So Kweon, and Sungha Choi. Ecotta: Memory-efficient continual test-time adaptation via self-distilled regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11920–11929, 2023. 4

[16] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2021. 1, 3, 4, 5

[17] Kunyu Wang, Xueyang Fu, Yuanfei Bao, Chengjie Ge, Chengzhi Cao, Wei Zhai, and Zheng-Jun Zha. Paid: Pairwise angular-invariant decomposition for continual test-time adaptation. *arXiv preprint arXiv:2506.02453*, 2025. 4

[18] Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7201–7211, 2022. 1, 4, 5

[19] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. In *Advances in Neural Information Processing Systems*, 2021. 4

[20] Senqiao Yang, Jiarui Wu, Jiaming Liu, Xiaoqi Li, Qizhe Zhang, Mingjie Pan, Yulu Gan, Zehui Chen, and Shanghang Zhang. Exploring sparse visual prompt for domain adaptive dense prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 16334–16342, 2024. 4

[21] Longhui Yuan, Binhui Xie, and Shuang Li. Robust test-time adaptation in dynamic scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15922–15932, 2023. 5

[22] Yunbei Zhang, Akshay Mehra, Shuaicheng Niu, and Jihun Hamm. Dpcore: Dynamic prompt coreset for continual test-time adaptation. In *International Conference on Machine Learning*. PMLR, 2025. 1, 4, 5

[23] Zhilin Zhu, Xiaopeng Hong, Zhiheng Ma, Weijun Zhuang, Yaohui Ma, Yong Dai, and Yaowei Wang. Reshaping the online data buffering and organizing mechanism for continual test-time adaptation. In *European Conference on Computer Vision*, pages 415–433. Springer, 2024. 1, 2, 4