SloMo-Fast: Slow-Momentum and Fast-Adaptive Teachers for Source-Free Continual Test-Time Adaptation

Md Akil Raihan Iftee[1,*], Mir Sazzat Hossain[1], Rakibul Hasan Rajib[1], Tariq Iqbal[2], Md Mofijul Islam[3,†], M Ashraful Amin[1], Amin Ahsan Ali[1] and A K M Mahbubur Rahman[1]
[1]Center for Computational & Data Sciences, Independent University, Bangladesh
[2]University of Virginia, USA, [3]Amazon GenAI, USA

Abstract

Continual Test-Time Adaptation (CTTA) is crucial for deploying models in real-world applications with unseen, evolving target domains. Existing CTTA methods, however, often rely on source data or prototypes, limiting their applicability in privacy-sensitive and resource-constrained settings. Additionally, these methods suffer from long-term forgetting, which degrades performance on previously encountered domains as target domains shift. To address these challenges, we propose SloMo-Fast, a source-free, dual-teacher CTTA framework designed for enhanced adaptability and generalization. It includes two complementary teachers: the Slow-Teacher, which exhibits slow forgetting and retains long-term knowledge of previously encountered domains to ensure robust generalization, and the Fast-Teacher rapidly adapts to new domains while accumulating and integrating knowledge across them. This framework preserves knowledge of past domains and adapts efficiently to new ones. We also introduce Cyclic Test-Time Adaptation (Cyclic-TTA), a novel CTTA benchmark that simulates recurring domain shifts. SloMo-Fast consistently outperforms state-of-the-art methods across Cyclic-TTA along with nine other CTTA settings, highlighting its ability to generalize across evolving and revisited domains.

1. Introduction

Adapting models to dynamic environments is essential for real-world deployment in areas like autonomous driving, healthcare, and robotics, where systems must operate effectively under evolving conditions without prior knowledge of these changes. Unlike Test-Time Adaptation (TTA) such as TENT [38], MEMO [50], and EATA [30], which allows models to adapt to a single, unseen domain using only unlabeled test data, Continual Test-Time Adaptation (CTTA) [21, 35, 40, 46] extends this by enabling models to adapt over time to a sequence of changing domains, crucial for real-world scenarios like autonomous driving, where environments evolve continuously with unpredictable weather and lighting conditions.

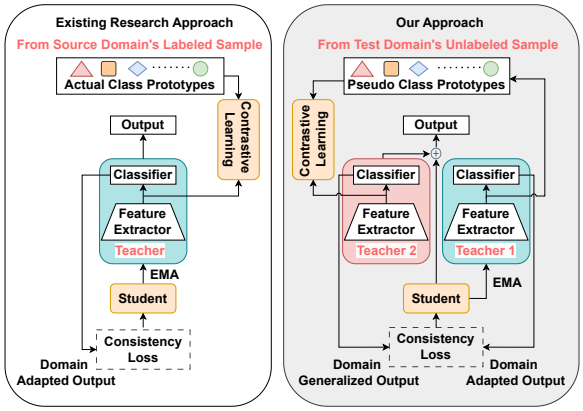

Figure 1. Overview of CTTA approaches with teacher-student models and contrastive learning. SloMo-Fast (on the right) integrates a second teacher model and dynamically generates prototypes at test time without requiring source data.

Recent advances in CTTA challenges in achieving practical, effective adaptation for real-world applications. First, models must adapt to evolving, source-free [6, 14, 34, 44] data streams in an online fashion [17, 25], requiring simultaneous prediction and continuous model updating for adaptation. Second, the dynamic nature of distribution shifts across continually changing target domains can compromise the reliability of pseudo-labels [16, 52], hinder source knowledge retention, and lead to error accumulation [40, 42] and catastrophic forgetting [22, 45]. Third, ensuring robust generalization [3, 12] is critical, as previously encountered domains may reappear during test time, necessitating long-term memory of prior knowledge.

To mitigate error accumulation and catastrophic forgetting of the source domain, CoTTA [40] uses knowledge distillation along with stochastic restoration of the source model. Other approaches reduce knowledge forgetting by enforcing alignment in the parameter-space [29, 32],

*Corresponding author: iftee1807002@gmail.com
†Work does not relate to position at Amazon.

feature-space [2, 8] or through logit-driven energy minimization strategies [5, 49]. However, achieving reliable alignment remains challenging in the absence of ground-truth labels during test time.

To better align target features with the source, it's important to reference feature class boundaries to avoid excessive drift during adaptation. Self-training or knowledge distillation-based CTTA methods like RMT [8], RoTTA [48], AR-TTA [35], and DPLOT [47] rely heavily on the quality of pseudo-labels from a teacher model. These methods often use memory banks to store source prototypes or labeled features, which help guide adaptation and preserve domain-specific characteristics. However, in real-world scenarios, accessing source data or prototypes is often restricted due to privacy concerns [13], storage limits, or transmission constraints [30, 42], making such approaches less viable in sensitive fields like healthcare, finance, autonomous driving, and surveillance.

Identifying reliable samples during test-time adaptation remains a core challenge, especially under distribution shifts. To address this, several TTA methods directly minimize the entropy score [9, 38, 43] to encourage confident predictions, while others [30, 31, 48] rely on uncertainty or entropy-based sample selection to filter out high-entropy samples that may hinder adaptation. However, these approaches often degrade under spurious correlation shifts [1], TRAP factors [16], and biasness [26]. Recent studies, such as Deyo [16] and FATA [4], have demonstrated that confidence-based metrics alone are insufficient for identifying trustworthy samples under diverse distribution shifts.

Recent methods like ROID [26] adopts diversity and certainty based weighting to retain past domain knowledge. VIDA [22] uses high- and low-rank adapters, MGG [7] refines gradients through historical memory, and TCA [28] enforces inter-class structure via topological constraints to mitigate CTTA challenges. Although effective in some continual settings, none of these approaches explicitly handle real-world CTTA scenarios involving cyclic domain arrivals, where domains may repeat over time, such as in autonomous driving or UAV applications where weather patterns can recur. Furthermore, many of these methods lack robust domain generalizability and require re-adaptation when previously seen domains return, showing limited ability to retain long-term domain-specific knowledge.

To address the challenges, we propose SloMo-Fast, a dual-teacher framework alongside a student model, that eliminates the need for source data while enhancing adaptability and generalization (Figure 2). It employs two teachers: the Fast-Teacher (T_1), which adapts quickly to new domains, and the Slow-Teacher (T_2), which adapts gradually to ensure robust generalization. Unlike prior work that relies on source-based prototypes from labeled data, SloMo-Fast introduces a new method for generating class-wise prototypes directly from unlabeled target data using high-confidence features filtered by pseudo label probability difference (PLPD). The Fast-Teacher, updated via exponential moving average (EMA) without backpropagation, produces these prototypes, which guide the Slow-Teacher through contrastive learning to support robust cross-domain generalization. While (T_1) ensures fast, low-cost adaptation, (T_2) updates only batch normalization layers, maintaining efficiency. This dual-teacher design enables effective adaptation to current domains while preserving knowledge of previously encountered ones, ensuring reliable pseudo-labels and robust generalization performance in dynamic, continually evolving real-world environments.

Our extensive experiments show that SloMo-Fast consistently outperforms state-of-the-art methods across CTTA benchmarks in 10 unique domain arrival scenarios, including our proposed benchmark Cyclic. It achieves a mean error rate of 33.8%, surpassing existing methods by at least 1.5% across various TTA settings. On five datasets, CIFAR10-C, CIFAR100-C, ImageNet-C, ImageNet-R, and ImageNet-Sketch, SloMo-Fast delivers consistently strong performance, establishing it as a state-of-the-art CTTA framework capable of adapting effectively to diverse real-world scenarios.

The key contributions of our work are as follows:

- We propose SloMo-Fast, a dual-teacher CTTA framework that eliminates the need for source data while enhancing adaptability and generalization. The Fast-Teacher (T_1) adapts quickly to new domains, while the Slow-Teacher (T_2) ensures robust generalization by adapting gradually.
- We introduce an entropy and PLPD-aware prototype prioritization approach to refine the Slow-Teacher for learning generalized representations across domains. The feature prototypes of each class are generated dynamically at test time without requiring source data.
- We propose a novel TTA setting, Cyclic-TTA, where domains can repeat over time, as a new benchmark.
- SloMo-Fast consistently outperforms state-of-the-art methods across 10 diverse CTTA scenarios across 5 datasets, demonstrating robust domain generalization.

2. Related Works

In this section, we provide summary of recent CTTA models followed by their different experimental setting such as gradual or mixed domain adaptation.

2.1. Continual Test Time Adaptation (CTTA)

Recent CTTA methods tackle challenges like sample selection, forgetting, and domain generalization. DeYo [16] introduced a confidence metric, Pseudo-Label Probability Difference (PLPD), for more reliable sample selection. DPLOT [47] enhanced adaptation by fine-tuning domain-specific blocks via Paired-View Pseudo-Labeling. CMF

[18] addressed catastrophic forgetting with Continual Momentum Filtering. BECoTTA [15] selectively updated low-rank domain experts to dynamically capture evolving domain knowledge. MGG [7] proposed a Meta Gradient Generator with a lightweight Gradient Memory Layer to refine noisy gradients during adaptation. TCA [28] enforced topological consistency by minimizing centroid distortion across class relationships. OT-VP [51] achieved source-free adaptation by learning universal visual prompts that align domains via Optimal Transport, without modifying ViT parameters.

2.2. Knowledge Distillation based CTTA

Knowledge distillation has been widely applied in CTTA to balance stability and adaptability. CoTTA [40] introduced a teacher-student framework for non-stationary environments. EcoTTA [36] improved memory efficiency using meta-networks and self-distilled regularization. RoTTA [48] added a time-aware reweighting strategy to account for sample timeliness and uncertainty. PSMT [37] selectively updated network parameters to mitigate overfitting. VIDA [22] addressed adaptability–forgetting trade-offs with high- and low-rank domain adapters. Zhu et al. [52] proposed an uncertainty-aware buffer for high-confidence samples and a graph-based constraint to preserve class relations. SoTa-DiT [23] used dual prompts in a ViT to disentangle and retain source-domain knowledge while adapting to the target domain via contrastive learning.

2.3. CTTA with Gradual/Mixed Settings

RMT [8] addressed continual and gradual shifts using a robust mean teacher with contrastive learning to align target and source domains. GTTA [27] mitigated error accumulation in long sequences by generating intermediate domains via mixup and lightweight style transfer, handling both gradual and abrupt shifts. ROID [26] proposed universal TTA for mixed settings, incorporating weight ensembling, certainty- and diversity-based weighting, and adaptive prior correction to balance generalization with domain-specific adaptation, while leveraging normalization layer adjustments (e.g., group/layer normalization). PaLM [24] introduced adaptive layer selection based on uncertainty (gradient norms from KL divergence), freezing others, and applying layer-specific learning rates for robust adaptation under continual and gradual shifts.

3. Methodology

3.1. Overview

We address the task of adapting a pre-trained model to perform effectively in a continuously evolving target domain. The initial model, f_{θ_0} with parameters θ_0, is trained on a source dataset (X^s, Y^s). Our goal is to improve the model's performance during inference in a dynamic environment where data distributions shift over time, without access to the source data. At each time step t, the model receives new target data x_t, predicts $f_{\theta_t}(x_t)$, and updates its parameters from θ_t to θ_{t+1} to better adapt to the changing distributions.

Fig. 2 provides an overview of our method, which incorporates two teacher models and a student model. All models share the same architecture, feature extractor and classifier and are initialized with θ_0, but differ in update strategies. The student model S, with weights θ_S, is updated using symmetric cross-entropy, leveraging pseudo-labels from both teacher models. The fast-teacher model, T_1, updates its weights θ_{T_1} using an EMA of the student's weights, smoothing the student's learning process. The slow-teacher model, T_2, initially updates its weights θ_{T_2} by optimizing contrastive loss, mean squared error (MSE) loss, and information maximization loss to learn domain-invariant features. Subsequently, its parameters are updated via EMA of the student model at each time step. This dual-teacher framework offers complementary supervision, enhancing adaptation and stability across shifting distributions.

3.2. Self-training with Dual Teacher

For an incoming test sample x_t at time step t, the student model S aims to minimize the discrepancy between its own predictions and the pseudo labels generated by the teacher models T_1 and T_2. Instead of standard cross-entropy for discrepancy minimization, we use symmetric cross-entropy [41], which was originally proposed to address noisy labels and has been shown to exhibit better gradient properties compared to standard cross-entropy [8]. For two distributions p and q, the symmetric cross-entropy is defined as:

$$\mathcal{L}_{SCE}(p,q) = -\sum_{c=1}^{C} p(c)\log q(c) - \sum_{c=1}^{C} q(c)\log p(c) \quad (1)$$

where C is the number of classes, $p(c)$ and $q(c)$ represents the probability of class c under distribution p and q, respectively. The training objective for the student model S, leveraging predictions from teacher models T_1 and T_2, results in the following self-training or consistency loss:

$$\begin{aligned}\mathcal{L}_{\mathcal{ST}}(x_t) =& \mathcal{L}_{SCE}(f_{\theta_S}(x_t), f_{\theta_{T_1}}(x_t)) + \\ & \mathcal{L}_{SCE}(f_{\theta_S}(x_t), f_{\theta_{T_2}}(x_t))\end{aligned} \quad (2)$$

After optimizing the student model S using $\mathcal{L}_{\mathcal{ST}}$, the teacher T_1 is updated via EMA:

$$\theta_{T_1}^{t+1} = \alpha\theta_{T_1}^{t} + (1-\alpha)\theta_S^{t+1} \quad (3)$$

Here, α is a smoothing factor.

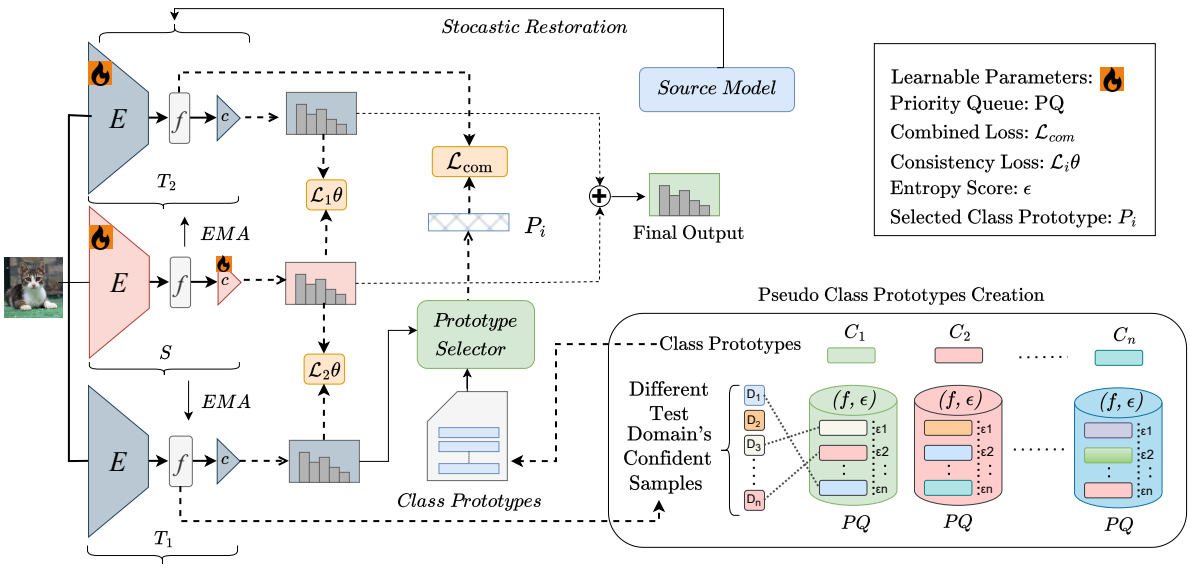

Figure 2. The SloMo-FAST framework comprises a dual-teacher and student model. The fast teacher T_1 quickly adapts to the current domain by taking the exponential moving average of the student. Confident feature vectors from T_1 are used to construct robust class prototypes via a priority queue, which refine the slow teacher T_2 through contrastive learning. This enables T_2 to learn domain-invariant representations while preserving knowledge from previous domains.

3.3. Class Specific Prototype Generation

To enable robust domain-generalized adaptation, we construct class-specific priority queues that store high-quality features from Fast-Teacher. These queues are iteratively updated during inference using a dual-criterion strategy based on entropy (confidence) and sensitivity (stability).

Pseudo Label Assignment: For each test sample x_t, the Fast-Teacher T_1 generates a pseudo label:

$$\hat{y}_{T_1} = \arg\max_c \, y_{T_1}(c), \tag{4}$$

where $y_{T_1}(c)$ denotes the softmax probability for class c.

Dual-Criterion Priority Queue Update: Each class-specific priority queue $\mathcal{Q}_c$ maintains a maximum of K elements, storing tuples $(z_t, \mathcal{H}_t)$ where z_t is the extracted feature and $\mathcal{H}_t$ is the entropy of prediction:

$$\mathcal{H}_t = -\sum_{c=1}^{C} y_{T_1}(c) \log y_{T_1}(c). \tag{5}$$

To ensure reliability, each feature is evaluated using two criteria: *(i) Confidence Criterion:* The entropy $\mathcal{H}_t$ must be less than a predefined threshold σ. *(ii)Sensitivity Criterion:* While entropy helps select confident predictions, it can mistakenly include overconfident but incorrect features, especially under domain shifts. To address this, [16] introduces a sensitivity criterion where the prediction sensitivity Δp_t must exceed a threshold δ, defined as:

$$\Delta p_t = p(\hat{y}_z \mid x_t) - \mathbb{E}_{x' \sim \mathcal{A}(x_t)}[p(\hat{y}_z \mid x')], \tag{6}$$

where $\hat{y}_z = \arg\max_c p(y = c \mid x_t, z)$ and $\mathcal{A}(x_t)$ denotes the set of augmentations of x_t. The sensitivity criterion checks if the prediction stays stable when the input is slightly changed using augmentations such as pixel, patch-shuffling, or center occlusion. If a feature is sensitive, it is likely unreliable. By keeping only stable features, the model learns more general and robust representations that work better across different domains.

A feature $(z_t, \mathcal{H}_t)$ is inserted into $\mathcal{Q}_c$ only if both criteria are met. If the queue is not full, the feature is inserted directly. If full, the new feature replaces the one with the highest entropy in the queue, but only if $\mathcal{H}_t$ is lower.

Queue Maintenance: To maintain diversity, every p time steps, the element with the *lowest* entropy is removed from each queue $\mathcal{Q}_c$. This prevents overfitting to early high-confidence predictions and allows inclusion of more representative features over time.

Reliable Feature Set: The resulting set of retained features for class c is:

$$\mathcal{S}_c = \{z_t \in \mathcal{Q}_c \mid \mathcal{H}_t \leq \sigma, \; \Delta p_t \geq \delta\}. \tag{7}$$

Only both confident and structurally stable features contribute to prototype construction for T_2.

Prototype Generation: We generate class prototypes using the reliable feature subset stored in each priority queue. Prototypes are computed as a weighted average, where the weight is the inverse of entropy, normalized for consistency. For class c, the prototype P_c is computed as:

$$P_c = \frac{1}{w} \sum_{(z, \mathcal{H}(z)) \in \mathcal{S}_c} w_z z, \tag{8}$$

where $w_z = \frac{1}{\mathcal{H}(z)}$ and $w = \sum_{(z,\mathcal{H}(z))\in\mathcal{S}_c} w_z$. This encourages confident, diverse prototypes that generalize across domains. (See the Algorithm 1 in Section B of the supplementary materials)

Contrastive Learning with Class Prototypes: To learn domain-generalized features, we apply contrastive learning with class prototypes—aligning same-class features across domains while separating different classes. During inference, we update class-specific priority queues with features from the teacher model T_1 and recompute prototypes using Equation (8).

We focus on samples where T_1 is confident but T_2 is not, using the binary variable:

$$n_i = \mathbb{1}[\mathcal{H}(y_{T_1}^{(i)}) \leq \sigma] \cdot \mathbb{1}[\mathcal{H}(y_{T_2}^{(i)}) > \sigma] \tag{9}$$

The selected features form the set:

$$\mathbb{S} = \{s_i \mid n_i = 1\}, \tag{10}$$

where s_i is the feature from T_2 for the i-th sample. For each feature in $\mathbb{S}$, we compute the cosine similarity with all class prototypes and select the nearest prototype to form a positive pair. To ensure invariance to input changes, we include the test sample's augmented view, resulting in a batch size of $3N$, where N is the size of $\mathbb{S}$. Each batch consists of original features, augmented views, and prototypes. For $i \in I := \{1, \ldots, 3N\}$, let $A(i) := I \setminus \{i\}$ and $V(i)$ represent different views of sample i. Following [8], we use a non-linear projection layer to obtain $z = \text{Proj}(s_i)$. The contrastive loss is defined as:

$$\mathcal{L}_{\mathcal{CL}} = -\sum_{i\in I}\sum_{v\in V(i)} \log\left(\frac{\exp\left(\text{sim}(z_i, z_v)/\tau\right)}{\sum_{a\in A(i)}\exp\left(\text{sim}(z_i, z_a)/\tau\right)}\right) \tag{11}$$

where τ is the temperature, and $\text{sim}(u, v) = \frac{u^T v}{\|u\|\|v\|}$ is the cosine similarity.

Feature Alignment with MSE Loss: To promote domain-generalizable representations, we apply an MSE loss that aligns features from the T_2 model with their corresponding class prototypes. Each test sample's feature z_i is matched with the prototype $P_{\hat{y}^i_{T_1}}$ based on the pseudo-label from T_1:

$$\mathcal{L}_{\text{MSE}} = \frac{1}{N}\sum_{i=1}^{N} \|z_i - P_{\hat{y}^i_{T_1}}\|^2 \tag{12}$$

where N is the number of selected test samples, and $\hat{y}^i_{T_1}$ denotes the pseudo-label from Equation (4).

Information Maximization Loss: To ensure T_2 provides strong guidance to the student model S, it must maintain both discriminability and diversity in its predictions. Following state-of-the-art unsupervised domain adaptation methods [19, 20], we use an information maximization loss, $\mathcal{L}_{\text{IM}}$, comprising two components:

$$\mathcal{L}_{\text{IM}} = -\mathbb{E}_{x_t\in X_t}\sum_{c=1}^{C} y_{T_2}(c)\log(y_{T_2}(c)) - \sum_{c=1}^{C}\bar{q}(c)\log(\bar{q}(c)), \tag{13}$$

where the first term enhances individual prediction certainty, and the second term promotes variation across class distributions.

The overall training objective for the teacher model T_2 is defined as follows:

$$\mathcal{L}_{T2} = \lambda_{\text{cl}}\mathcal{L}_{\text{CL}} + \lambda_{\text{mse}}\mathcal{L}_{\text{MSE}} + \lambda_{\text{im}}\mathcal{L}_{\text{IM}} \tag{14}$$

where λ_{cl}, λ_{mse}, and λ_{im} represent the weighting factors for the contrastive loss, the MSE loss, and the information maximization loss, respectively.

To address error accumulation from distribution shifts, we use a stochastic restoration method [40] that combines the pretrained source model's original weights with updated weights after each gradient step. This approach mitigates catastrophic forgetting by selectively restoring weights, preserving knowledge from the source model.

3.4. Prediction Ensembling

Inspired by [8], we combine the outputs of both the student and T_2 models. The student model adapts quickly to the current domain, while the T_2 model provides generalized predictions across domains. This combination leverages their complementary strengths, improving prediction robustness and accuracy in dynamic environments. For a test sample x_t, the final prediction is:

$$y_t = f_{\theta_S}(x_t) + f_{\theta_{T_2}}(x_t) \tag{15}$$

Prior Correction: Due to domain shift, the learned posterior $q(y|x)$ may diverge from the true posterior $p(y|x)$, degrading performance [26]. We apply prior correction [33] in the dual-teacher setup:

$$p(y|x) = q(y|x)\cdot\frac{p(y)}{q(y)} \tag{16}$$

where $p(y)$ is estimated by the current batch's mean softmax output $\hat{p}_t$, assuming near-uniform learned priors due to the information loss in Equation (13). To reduce sensitivity to small batch sizes, we apply adaptive smoothing [26]:

$$\bar{p}_t = \frac{\hat{p}_t + \gamma}{1 + \gamma N_c} \tag{17}$$

where γ is a smoothing factor, N_c is the no of classes.

4. Result and Discussion

This section analyzes experimental results across datasets and CTTA settings, highlighting the proposed method's performance, robustness, and comparison with baselines.

Setting	Dataset	Source	TENT-cont.	RoTTA	CoTTA	ROID	SloMo-Fast	SloMo-Fast*
Continual	CIFAR10-C	43.5	20.0	19.3	16.5	16.2	15.8±0.09	**14.8±0.07**
	CIFAR100-C	46.4	62.2	34.8	32.8	29.3	29.2±0.11	**27.9±0.12**
	ImageNet-C	82.0	82.5	78.1	76.0	54.5	54.2±0.10	**52.8±0.23**
	ImageNet-R	63.8	57.6	60.7	57.4	51.2	51.0±0.05	**50.4±0.07**
	ImageNet-Sketch	75.9	69.5	70.8	69.5	64.3	65.2±0.13	**64.1±0.21**
Mixed	CIFAR10-C	43.5	44.1	32.5	33.4	28.4	29.7±0.09	**28.0±0.06**
	CIFAR100-C	46.4	82.5	43.1	45.4	35.0	34.4±0.15	**33.5±0.02**
	ImageNet-C	82.0	86.4	78.1	79.4	**69.5**	71.3±0.11	70.8±0.27
Gradual	CIFAR10-C	43.5	26.2	11.8	10.8	10.5	10.4±0.11	**8.9±0.09**
	CIFAR100-C	46.4	75.9	33.4	27.0	24.3	24.7±0.25	**23.3±0.33**
	ImageNet-C	82.0	91.6	96.4	67.7	38.8	39.1±0.06	**37.9±0.17**
Episodic	CIFAR10-C	43.5	18.2	21.6	18.3	17.5	**17.8±0.08**	**16.7±0.15**
	CIFAR100-C	46.4	31.1	41.9	34.5	30.4	31.4± 0.31	**30.1±0.13**
	ImageNet-C	82.0	57.3	6.70	61.5	**51.6**	53.1±0.25	52.7±0.14
Cross Group (Continual)	CIFAR10-C	43.5	15.8	18.8	19.7	16.4	16.0±0.18	**14.7±0.09**
	CIFAR100-C	46.4	61.5	32.5	34.9	29.5	30.1±0.14	**27.9±0.11**
	ImageNet-C	82.0	62.2	68.6	59.2	55.7	54.3±0.22	**52.3±0.18**
Easy2Hard (Continual)	CIFAR10-C	43.5	19.6	17.8	15.7	15.9	15.5±0.18	**13.9±0.13**
	CIFAR100-C	46.4	52.8	33.0	32.2	29.3	29.2±0.12	**28.2±0.14**
	ImageNet-C	82.0	60.0	65.1	52.5	54.3	52.8±0.31	**48.6±0.13**
Hard2Easy (Continual)	CIFAR10-C	43.5	21.6	19.4	17.1	16.3	16.1±0.20	**15.3±0.04**
	CIFAR100-C	46.4	66.7	35.7	33.0	29.5	30.2± 0.11	**28.2±0.06**
	ImageNet-C	82.0	62.8	68.4	63.2	55.1	54.3±0.22	**52.8±0.17**
Mixed After Continual (Overlapping)	CIFAR10-C	43.5	21.3	19.9	16.8	16.9	16.7±0.12	**16.2**±0.15
	CIFAR100-C	46.4	63.7	35.1	33.2	29.6	30.6±0.17	**28.1**±0.10
	ImageNet-C	82.0	91.8	75.4	70.9	**52.5**	55.4±0.19	53.6±0.11
Continual After Mixed (Overlapping)	CIFAR10-C	43.5	46.0	18.9	17.8	16.7	16.6±0.06	**14.6±0.07**
	CIFAR100-C	46.4	97.1	34.5	33.3	29.4	29.1± 0.18	**26.6±0.17**
	ImageNet-C	82.0	85.6	64.2	55.3	55.1	55.6±0.20	**54.7±0.20**
Cyclic	CIFAR10-C	43.5	17.0	19.4	16.7	15.6	15.2±0.07	**14.6± 0.08**
	CIFAR100-C	46.4	34.7	37.8	33.7	28.9	28.6±0.27	**27.5±0.15**
	ImageNet-C	82.0	59.7	66.1	63.7	53.1	52.7±0.11	**51.9±0.23**
Mean Error Rates	All	57.9	54.5	42.4	40.6	35.3	34.6±0.13	**33.8±0.21**

Table 1. Average online classification error rate (%) over 5 runs for different TTA settings across multiple datasets. The table includes results for various TTA methods: TENT-cont., RoTTA, CoTTA, ROID, SloMo-Fast, and SloMo-Fast* (where all parameters of the student model are updated), evaluated in different settings. Results are shown for CIFAR10-C, CIFAR100-C, ImageNet-C, ImageNet-R, and ImageNet-Sketch datasets. Results in bold represent the best performance, while those in gray are the second best.

4.1. Implementation Details

We evaluate our approach on diverse domain shifts, including artificial corruptions and natural variations. Following [26], we use the corruption benchmark on CIFAR10-C(C10-C), CIFAR100-C(C100-C), and ImageNet-C [10] (IN-C), which apply 15 corruption types at five severity levels. Additionally, we assess our method on ImageNet-R [11] and ImageNet-Sketch [39]. We use priority queue size 10, and batch size 200 for CIFAR10-C and CIFAR100-C, and 64 for ImageNet-C, ImageNet-R, ImageNet-Sketch.

4.2. Result for Different TTA Setting

The proposed SloMo-Fast framework consistently achieves state-of-the-art performance across a variety of Test-Time Adaptation (TTA) settings (See their definitions in the supplementaray material in section D.1, demonstrating strong robustness and adaptability to different types of distribution shifts. In the Continual Setting, SloMo-Fast* achieves standout results on CIFAR10-C (14.8%) and CIFAR100-C (27.9%), SloMo-Fastalso outperforms existing best method, ROID by only updating BN parameters. It closely matches ROID on ImageNet-C (54.2% vs. 54.5%), and surpasses CoTTA on ImageNet-C (76.0%), ImageNet-R (50.4%), and ImageNet-Sketch (64.1%). In the Mixed Domain, it performs well on CIFAR10-C (28.0%) and CIFAR100-C (33.5%). Under the Gradual Setting, SloMo-Fast* significantly reduces error rates, achieving 8.9% on CIFAR10-C, 23.3% on CIFAR100-C, and 38.8% on ImageNet-C, outperforming both ROID and CoTTA. In the Episodic Setting, it achieves the lowest error rates on CIFAR10-C (16.7%) and CIFAR100-C (30.1%), and improves over CoTTA and RoTTA on ImageNet-C (51.6%). In Specialized Settings like cross-group and hard-to-easy adaptation, it also performs best achieving 14.7% and 15.3% on CIFAR10-C respectively-consistently outperforming ROID, RoTTA,

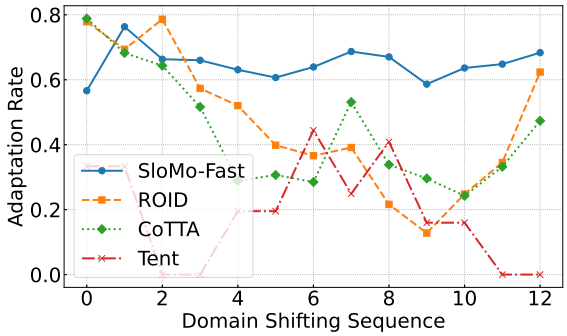

Figure 3. Adaptation rate over a cyclic domain shift. SloMo-Fast shows the best and most stable performance, benefiting from prototype memory and slow-teacher guidance.

CoTTA. Overall, these results demonstrate both SloMo-Fast* and SloMo-Fast(trains a BN parameters only) strong adaptability, making it a reliable and effective solution for real-world TTA challenges.

Mean Error Rate (%)			
Methods	CIFAR10-C	CIFAR100-C	IN-C
Tent (ICLR'21) [38]	20.0	62.2	82.5
CoTTA (CVPR'22) [40]	16.5	32.8	76.0
RoTTA (CVPR'23) [48]	19.3	34.8	78.1
ROID (WACV'24) [26]	16.2	29.3	54.5
BeCoTTA (ICML'24) [15]	16.3	35.5	60.9
ViDA (ICLR'24) [22]	20.7	28.6	61.2
SANTA (TMLR'24) [2]	16.1	30.3	60.1
OBAO (ECCV'24) [52]	15.8	29.0	59.0
PALM (AAAI'25) [24]	15.5	30.1	60.1
TCA (CVPR'25) [28]	15.1	29.7	59.3
SloMo-Fast	**14.8**	**27.9**	**52.8**

Table 2. A comparative analysis of recent advances in CTTA on CIFAR10-C, CIFAR100-C, and ImageNet-C benchmarks.

4.3. Results for Cyclic TTA Setting

In the **Cyclic Setting**, involving recurring domain shifts, SloMo-Fast* shows remarkable stability and performance. As shown in Table 1, it achieves the lowest error rates of 14.6±0.08% (CIFAR10-C), 27.5±0.15% (CIFAR100-C), and 51.9±0.23% (ImageNet-C), consistently outperforming all baselines, including its variant SloMo-Fast (e.g., 15.2±0.07% on CIFAR10-C). These results confirm its strong adaptability to dynamic distribution shifts.

4.4. Ablation Study on Adaptation Rate

To further show the effectiveness of our proposed method in terms of adaptation dynamics, we introduce a metric, Adaptation Rate, that quantifies how fast a model adapt after each domain shift. Our proposed metric is composed of three components: *(i) Time to Plateau(**TTP**)*: The number of adaptation steps required for the moving average of accuracy (computed over a window of size k) to reach 80% of its maximum value. A lower TTP indicates faster adaptation. *(ii) Average Positive Slope(**APS**)*: The mean of all positive differences between consecutive moving averages of accuracy values. Higher APS means a faster upward trend in adaptation. *(iii) Stability (**STD**)*: Stability quantifies the consistency of the adaptation process and is assessed using the standard deviation of the moving average of accuracy values. A lower STD implies greater adaptation stability.

We define the composite score for adaptation rate as:

$$\text{Adaptation Rate} = \frac{\text{APS}}{\text{TTP}} - \lambda \cdot \text{STD} \tag{18}$$

where λ is a stability regularizer. A higher score indicates a more desirable adaptation behavior: fast, strong, and stable.

Using Equation 18, we evaluate Slomo-Fast's adaptability under cyclic-TTA. As shown in Figure 3, SloMo-Fast achieves the highest adaptation rate, especially when previously seen domains reoccur. This demonstrates its fast adaptability, as it leverages stored knowledge from past domains through prototypes learned by the slow teacher.

Mean Error Rate (%)			
Components	**CIFAR10-C**	**CIFAR100-C**	**IN-C**
w/o $\mathcal{L}_{\text{MSE}}$	15.89	28.23	54.81
w/o $\mathcal{L}_{\text{IM}}$	16.17	28.35	55.42
w/o $\mathcal{L}_{\text{CL}}$	16.04	28.57	55.40
w/o PC	15.78	28.08	54.47
w/o ST	16.11	28.48	55.43
SloMo-Fast	**14.88**	**27.97**	**52.80**

Table 3. Ablation study of mean classification error rates (%) for online CTTA. The table shows the impact of removing individual components: $\mathcal{L}_{\text{MSE}}$, $\mathcal{L}_{\text{IM}}$, $\mathcal{L}_{\text{CL}}$, Prior Correction (PC), and Stochastic Restoration (ST).

Dataset \ Queue Size	5	20	25	50	100
CIFAR10-C	14.97	14.92	14.88	14.93	14.98
CIFAR100-C	28.19	28.24	28.16	28.21	28.11
ImageNet-C	54.52	53.14	56.32	59.88	63.72

Table 4. Ablation study of classification error rates (%) for CIFAR10-C and CIFAR100-C online continual test-time adaptation tasks. This table examines the impact of different queue sizes on classification error rates.

Method	**Trainable Params (% of Total)**	**Execution Time(s)**	**Error Rate(%)**
Tent	4.9%	0.01	54.5
CoTTA	100%	1.85	40.6
ROID	4.9%	0.30	35.0
SloMo-Fast	4.9%	0.24	34.6
SloMo-Fast*	51%	0.28	33.8

Table 5. Comparison of methods based on trainable parameters and average execution time during adaptation.

4.5. Ablation Study on Loss Components

Table 3 shows the effect of removing different loss components, Mean Squared Error ($\mathcal{L}_{\text{MSE}}$), Information Maxi-

mization ($\mathcal{L}_{\text{IM}}$), and Contrastive Loss ($\mathcal{L}_{\text{CL}}$), as well as optimization strategies like Prior Correction (PC) and Stochastic Restoration (ST), on CIFAR10-C and CIFAR100-C. The rise in error when a component is removed confirms its contribution. Using all components yields the best performance: 15.78% error on CIFAR10 and 28.48% on CIFAR100, emphasizing the benefit of combining diverse objectives and strategies for robust adaptation.

4.6. Ablation Study on Priority Queue Size

Table 4 explores how varying the queue size affects classification error on CIFAR10-C and CIFAR100-C. For CIFAR10-C, performance remains stable, with the best result (14.88%) at a queue size of 25. Larger sizes offer no clear gains. For CIFAR100-C, the lowest error (28.11%) occurs at size 100, but differences are minor overall. These findings suggest that the model is relatively insensitive to queue size on both datasets.

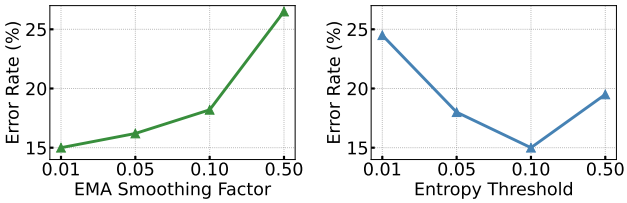

Figure 4. Comparison of loss against EMA factor and entropy.

4.7. Threshold and Hyperparameter Selection

We fixed key hyperparameters to optimal values: EMA smoothing factor = 0.01, and loss weights $\lambda_{cl} = \lambda_{im} = \lambda_{mse} = 1$, with an entropy threshold of 0.5 (Figure 4). The entropy threshold plays a key role in selecting reliable samples for prototype. As shown in Figure 4, increasing the threshold up to 0.5 improves performance by including more trustworthy samples. Beyond 0.5, performance drops due to noisy samples degrading prototype quality.

4.8. Parameter Efficiency and Adaptation Cost:

We have provided a comparative analysis (Table 5) showing that SloMo-Fast (BN-only) shares a similar ratio of trainable parameters (4.9%) with ROID and achieves comparable test-time efficiency, whereas in SloMo-Fast*, it has 51% trainable parameters, and Teacher 2 (Slow-Teacher) updates only batch normalization layers along with all trainable params in student model. Here, SloMo-Fast* remains time efficient (0.28s per batch) despite using more parameters whereas ROID (0.30s) needs larger per batch execution time because of using input data sampling and different augmentation techniques.

4.9. Qualitative Results: t-SNE Visualization

Finally, to visualize the effectiveness of our method, we provide t-SNE plots of the feature space at final stage of adaptation in Figure 5. The t-SNE visualization for **SloMo-Fast** shows that the learned representations are well-clustered and exhibit clear separation between the different classes, even under severe corruption conditions.

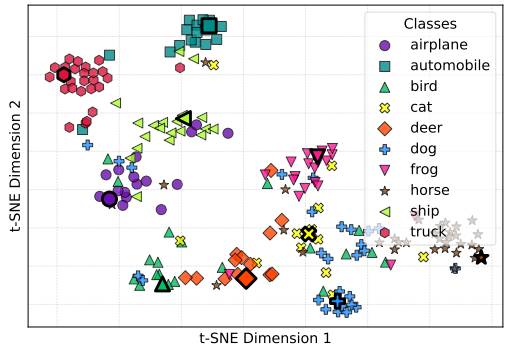

Figure 5. t-SNE visualization of feature representations and class prototypes (bigger shapes with thick black border). The visualization highlights distinct class separation, showcasing the model's ability to effectively learn discriminative feature representations.

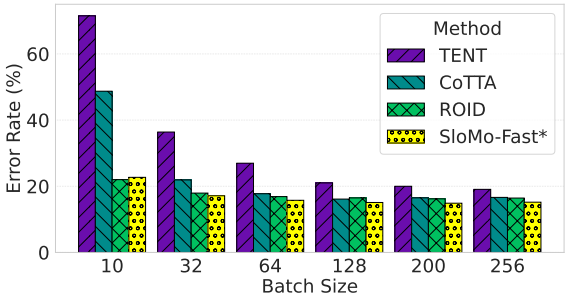

Figure 6. Impact of batch size on error rates (%) in the CIFAR10C online CTTA setup across different methods. Highlights that increasing batch size improves classification performance, with SloMo-Fast* outperforming all other methods at each batch size.

5. Conclusion

We present SloMo-Fast, a dual-teacher framework for CTTA that eliminates the need for source data while significantly enhancing adaptability, generalization, and efficiency. The Fast-Teacher (T_1) enables rapid on-the-fly adaptation to shifting domains, while the Slow-Teacher (T_2) ensures stable and robust generalization through gradual updates, guided by a entropy and PLPD-aware prototype prioritization strategy. These class prototypes are dynamically generated during inference, requiring no access to source data. To better reflect real-world deployment, we introduce Cyclic-TTA, a new CTTA setting where domain shifts may repeat over time. Extensive experiments across 10 challenging scenarios on 5 dataset including CIFAR-10C, CIFAR-100C, and ImageNet-C demonstrate that SloMo-Fast consistently outperforms state-of-the-art methods, setting new benchmarks for robustness, generalization, and practicality in privacy-sensitive.

References

[1] Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in terra incognita. In *Proceedings of the European conference on computer vision (ECCV)*, pages 456–473, 2018. 2

[2] Goirik Chakrabarty, Manogna Sreenivas, and Soma Biswas. Santa: Source anchoring network and target alignment for continual test time adaptation. *Transactions on Machine Learning Research*, 2023. 2, 7

[3] Liang Chen, Yong Zhang, Yibing Song, Ying Shan, and Lingqiao Liu. Improved test-time adaptation for domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24172–24182, 2023. 1

[4] Younggeol Cho, Youngrae Kim, Junho Yoon, Seunghoon Hong, and Dongman Lee. Feature augmentation based test-time adaptation. 2024. 2

[5] Wonjeong Choi, Do-Yeon Kim, Jungwuk Park, Jungmoon Lee, Younghyun Park, Dong-Jun Han, and Jaekyun Moon. Adaptive energy alignment for accelerating test-time adaptation. In *The Thirteenth International Conference on Learning Representations*. 2

[6] Peihua Deng, Jiehua Zhang, Xichun Sheng, Chenggang Yan, Yaoqi Sun, Ying Fu, and Liang Li. Multi-granularity class prototype topology distillation for class-incremental source-free unsupervised domain adaptation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 30566–30576, 2025. 1

[7] Qi Deng, Shuaicheng Niu, Ronghao Zhang, Yaofo Chen, Runhao Zeng, Jian Chen, and Xiping Hu. Learning to generate gradients for test-time adaptation via test-time training layers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 16235–16243, 2025. 2, 3

[8] Mario Döbler, Robert A. Marsden, and Bin Yang. Robust mean teacher for continual and gradual test-time adaptation. In *CVPR*, pages 7704–7714, 2023. 2, 3, 5, 1

[9] Jisu Han, Jaemin Na, and Wonjun Hwang. Ranked entropy minimization for continual test-time adaptation. *International conference on machine learning*, 2025. 2

[10] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019. 6

[11] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8340–8349, 2021. 6

[12] Yusuke Iwasawa and Yutaka Matsuo. Test-time classifier adjustment module for model-agnostic domain generalization. *Advances in Neural Information Processing Systems*, 34:2427–2440, 2021. 1

[13] Neerav Karani, Ertunc Erdil, Krishna Chaitanya, and Ender Konukoglu. Test-time adaptable neural networks for robust medical image segmentation. *Medical Image Analysis*, 68: 101907, 2021. 2

[14] Jogendra Nath Kundu, Naveen Venkat, R Venkatesh Babu, et al. Universal source-free domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4544–4553, 2020. 1

[15] Daeun Lee, Jaehong Yoon, and Sung Ju Hwang. BECoTTA: Input-dependent online blending of experts for continual test-time adaptation. In *ICML*, pages 27072–27093, 2024. 3, 7

[16] Jonghyun Lee, Dahuin Jung, Saehyung Lee, Junsung Park, Juhyeon Shin, Uiwon Hwang, and Sungroh Yoon. Entropy is not enough for test-time adaptation: From the perspective of disentangled factors. In *ICLR*, 2024. 1, 2, 4

[17] Jae-Hong Lee and Joon-Hyuk Chang. Stationary latent weight inference for unreliable observations from online test-time adaptation. In *Forty-first International Conference on Machine Learning*. 1

[18] Jae-Hong Lee and Joon-Hyuk Chang. Continual momentum filtering on parameter space for online test-time adaptation. In *ICLR*, 2024. 3

[19] Xinyao Li, Zhekai Du, Jingjing Li, Lei Zhu, and Ke Lu. Source-free active domain adaptation via energy-based locality preserving transfer. In *Proceedings of the 30th ACM international conference on multimedia*, pages 5802–5810, 2022. 5

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

[21] 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

[22] 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 *ICLR*, 2024. 1, 2, 3, 7

[23] Tianyi Ma and Maoying Qiao. Disentangle source and target knowledge for continual test-time adaptation. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 8024–8034. IEEE, 2025. 3

[24] Sarthak Kumar Maharana, Baoming Zhang, and Yunhui Guo. Palm: Pushing adaptive learning rate mechanisms for continual test-time adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 19378–19386, 2025. 3, 7

[25] Massimiliano Mancini, Hakan Karaoguz, Elisa Ricci, Patric Jensfelt, and Barbara Caputo. Kitting in the wild through online domain adaptation. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1103–1109. IEEE, 2018. 1

[26] 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. 2, 3, 5, 6, 7, 4

[27] Robert A. Marsden, Mario Döbler, and Bin Yang. Introducing intermediate domains for effective self-training during test-time. In *IJCNN*, pages 1–10, 2024. 3

[28] 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 Computer Vision and Pattern Recognition Conference*, pages 15319–15328, 2025. 2, 3, 7

[29] Fahim Faisal Niloy, Sk Miraj Ahmed, Dripta S Raychaudhuri, Samet Oymak, and Amit K Roy-Chowdhury. Effective restoration of source knowledge in continual test time adaptation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2091–2100, 2024. 1

[30] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, pages 16888–16905. PMLR, 2022. 1, 2

[31] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Zhiquan Wen, Yaofo Chen, Peilin Zhao, and Mingkui Tan. Towards stable test-time adaptation in dynamic wild world. *arXiv preprint arXiv:2302.12400*, 2023. 2

[32] Ori Press, Steffen Schneider, Matthias Kümmerer, and Matthias Bethge. Rdumb: A simple approach that questions our progress in continual test-time adaptation. *Advances in Neural Information Processing Systems*, 36:39915–39935, 2023. 1

[33] Amelie Royer and Christoph H Lampert. Classifier adaptation at prediction time. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1401–1409, 2015. 5

[34] Pascal Schlachter, Simon Wagner, and Bin Yang. Memory-efficient pseudo-labeling for online source-free universal domain adaptation using a gaussian mixture model. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 6425–6434. IEEE, 2025. 1

[35] Damian Sójka, Sebastian Cygert, Bartłomiej Twardowski, and Tomasz Trzciński. Ar-tta: A simple method for real-world continual test-time adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3491–3495, 2023. 1, 2

[36] Junha Song, Jungsoo Lee, In So Kweon, and Sungha Choi. EcoTTA: Memory-efficient continual test-time adaptation via self-distilled regularization. In *CVPR*, pages 11920–11929, 2023. 3

[37] Jiaxu Tian and Fan Lyu. Parameter-selective continual test-time adaptation. In *Proceedings of the Asian Conference on Computer Vision*, pages 1384–1400, 2024. 3

[38] Dequan Wang, Evan Shelhamer, Shaoteng Liu, B. Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *ICLR*, 2021. 1, 2, 7, 4

[39] Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. *Advances in Neural Information Processing Systems*, 32, 2019. 6

[40] Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *CVPR*, pages 7191–7201, 2022. 1, 3, 5, 7, 4

[41] Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. Symmetric cross entropy for robust learning with noisy labels. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 322–330, 2019. 3

[42] Yanshuo Wang, Jie Hong, Ali Cheraghian, Shafin Rahman, David Ahmedt-Aristizabal, Lars Petersson, and Mehrtash Harandi. Continual test-time domain adaptation via dynamic sample selection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1701–1710, 2024. 1, 2

[43] Xiangyu Wu, Feng Yu, Qing-Guo Chen, Yang Yang, and Jianfeng Lu. Multi-label test-time adaptation with bound entropy minimization. *International Conference on Learning Representations*, 2025. 2

[44] Gezheng Xu, Hui Guo, Li Yi, Charles Ling, Boyu Wang, and Grace Yi. Revisiting source-free domain adaptation: a new perspective via uncertainty control. In *The Thirteenth International Conference on Learning Representations*, 2024. 1

[45] Xu Yang, Yanan Gu, Kun Wei, and Cheng Deng. Exploring safety supervision for continual test-time domain adaptation. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, pages 1649–1657, 2023. 1

[46] Xu Yang, Moqi Li, Jie Yin, Kun Wei, and Cheng Deng. Navigating continual test-time adaptation with symbiosis knowledge. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pages 5326–5334, 2024. 1

[47] Yeonguk Yu, Sungho Shin, Seunghyeok Back, Minhwan Ko, Sangjun Noh, and Kyoobin Lee. Domain-specific block selection and paired-view pseudo-labeling for online test-time adaptation. In *CVPR*, pages 22723–22732, 2024. 2

[48] Longhui Yuan, Binhui Xie, and Shuang Li. Robust test-time adaptation in dynamic scenarios. In *CVPR*, pages 15922–15932, 2023. 2, 3, 7

[49] Yige Yuan, Bingbing Xu, Liang Hou, Fei Sun, Huawei Shen, and Xueqi Cheng. Tea: Test-time energy adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23901–23911, 2024. 2

[50] Marvin Zhang, Sergey Levine, and Chelsea Finn. MEMO: test time robustness via adaptation and augmentation. In *NeurIPS*, pages 38629–38642, 2022. 1

[51] Yunbei Zhang, Akshay Mehra, and Jihun Hamm. Ot-vp: Optimal transport-guided visual prompting for test-time adaptation. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1122–1132. IEEE, 2025. 3

[52] 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, 3, 7

SloMo-Fast: Slow-Momentum and Fast-Adaptive Teachers for Source-Free Continual Test-Time Adaptation

Supplementary Material

A. Architecture Evolution

In figure A.1, we have shown how our SloMo-Fast evolves from prior research works and highlights the comparison of methodology with previous knowledge distillation based techniques of TTA. CoTTA [40] employs a teacher-student framework where the student model is updated based on the pseudo-labels generated by the teacher model. The teacher model, in turn, is updated using the Exponential Moving Average (EMA) of the student parameters. While CoTTA demonstrates effective adaptation, it suffers from catastrophic forgetting and lacks the ability to retain long-term domain knowledge.

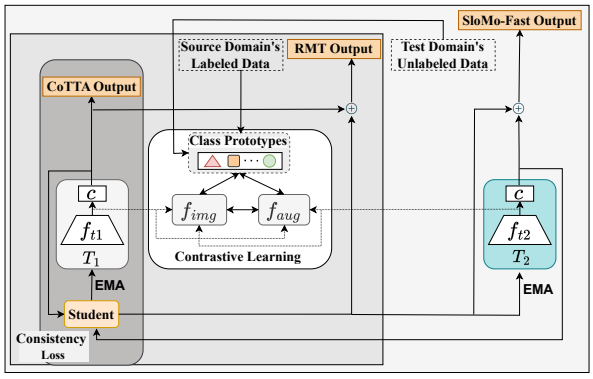

Figure A.1. Compared to CoTTA, RMT, and SloMo-Fast.

To address this limitation, RMT [8] introduces source prototypes and utilizes contrastive loss between the source class prototypes and test-time inputs. However, relying on source prototypes is often impractical in real-world scenarios due to their rarity and unavailability in many applications.

In contrast, our SloMo-Fast framework introduces a second teacher model that is more domain-generalized. Instead of using source prototypes, SloMo-Fast constructs class prototypes from confident test samples. This approach eliminates the dependence on source data while enabling long-term retention of domain knowledge, ensuring robust adaptation and generalization across dynamic and evolving domains.

B. Class-Specific Prototype Creation via Dual-Criterion Filtering

To maintain reliable and discriminative prototypes, we selectively store test features in class-specific queues based on low entropy and high sensitivity. Periodic pruning ensures the queues retain only the most confident and stable representations for each class.

Algorithm 1 Dual-Criterion Reliable Feature Selection

Input: test samples X_t, teacher model T_1, entropy threshold σ, sensitivity threshold δ, time interval p, max queue size K
Output: Updated priority queues Q_c for each class c
Initialize priority queue Q_c for each class c with size K
for each test sample x_t **do**
    Predict class c, entropy $\mathcal{H}_t$, sensitivity Δp_t, and feature z_t from $T_1(x_t)$
    if $t \mod p = 0$ **then** Remove element with $\min \mathcal{H}$ from each Q_c
    end if
    if $\mathcal{H}_t \leq \sigma$ **and** $\Delta p_t \geq \delta$ **then**
        if Q_c is full and $\mathcal{H}_t < \max \mathcal{H}$ in Q_c **then** Replace element with $\max \mathcal{H}$ by $(z_t, \mathcal{H}_t)$
        else if Q_c is not full **then** Insert $(z_t, \mathcal{H}_t)$ into Q_c
        end if
    end if
end for

C. Datasets:

- **CIFAR10-C:** Consists of 10 classes, with 1,000 samples per class for each domain, amounting to 10,000 images per domain.
- **CIFAR100-C:** Comprises 100 categories, with 100 samples per category or class for each domain, yielding a total of 10,000 images per domain.
- **ImageNet-C:** Contains 1,000 categories or classes, with 50 samples per category for each domain, resulting in 50,000 images per domain.
- **ImageNet-R:** Features 200 classes from ImageNet, with 30,000 images focusing on a variety of renditions, such as art, cartoons, and sketches.
- **ImageNet-Sketch:** Contains 1,000 classes with 50,889 images in total, created as sketch drawings corresponding to the ImageNet categories.

D. Supplementary Experimental Results

D.1. Benchmarks for Test-Time Adaptation

All evaluations are conducted in an *online test-time adaptation (TTA)* setting, where predictions are updated and evaluated immediately. We evaluate our model on benchmarks for analyzing CTTA:

- **Continual Domains:** Following [26], the model adapts sequentially across K domains $[D_1, D_2, \ldots, D_K]$ without prior knowledge of domain boundaries. For the corruption datasets, the sequence includes all 15 corruption types encountered at severity level 5.
- **Mixed Domains:** As in [26], test data from multiple domains are encountered together in a mixed manner during adaptation, with consecutive samples often coming from different domains.
- **Gradual Domains:** Although some domain shifts happen abruptly, many progress gradually over time(severity of domain shifts changes incrementally), making this setting a practical scenario for test-time adaptation.
- **Episodic Setting:** This setting considers a single domain shift, where upon encountering a new domain, the adaptation model resets to the source model and starts adaptation from the beginning.
- **Cyclic Domains:** A new benchmark where the domain sequence is repeated in cycles based on corruption subgroups (e.g., Noise, Blur, Weather, Digital, and Distortion). Subgroups include corruptions such as noise (gaussian, shot, impulse), blur (defocus, motion, glass), weather (snow, fog, frost), digital (brightness, contrast), and distortion (elastic transform, pixelate, jpeg compression).
- **Continual-Cross Group:** Domains are encountered sequentially in a continual setup, where each domain is sampled one after another from different corruption groups (e.g., Noise, Blur, Weather, Digital, Distortion) like inter group mixing.
- **Continual-Hard2Easy:** Domains are encountered sequentially, where corruptions are sorted from high error to low error based on the initial source model's performance at severity level 5.
- **Continual-Easy2Hard:** Domains are encountered sequentially, where corruptions are sorted from low error to high error based on the initial source model's performance at severity level 5.
- **Mixed after Continual TTA:** Domains are first encountered sequentially, as in the continual setting, followed by data from previously seen domains being encountered in a mixed manner.
- **Continual after Mixed TTA:** Domains are first encountered in a mixed manner, where test data from multiple domains come together randomly. After this mixed phase, the domains are encountered sequentially, as in the continual setting.

D.2. Detailed result

Across a wide range of datasets and model architectures—including convolutional networks (WRN-28, ResNet-50), hierarchical transformers (Swin-b), and plain vision transformers (ViT-b-16)—Slomo-Fast and its enhanced variant Slomo-Fast* demonstrate consistently superior performance in continual test-time adaptation. Unlike existing methods, whose effectiveness often varies significantly with the backbone or data distribution, our methods remain robust and stable across both lightweight and large-scale architectures. The improvements are especially pronounced on complex benchmarks like IN-C, IN-R, and IN-Sketch, confirming the generalizability of Slomo-Fast across both domain shifts and backbone types.

D.3. Ablation Study on Losses Applied to T2

The results of the ablation study are summarized in Tables A.11 and A.12, which evaluate the effect of different loss functions applied to the T_2 model on the CIFAR10-to-CIFAR10C and CIFAR100-to-CIFAR100C online continual test-time adaptation tasks, respectively, evaluations use the WideResNet-28 and ResNeXt-29 model under the highest corruption severity level (level 5). The classification error rates (%) are reported for 15 corruption types, along with the mean error rate as a summary.

In the CIFAR10-to-CIFAR10C task (Table A.11), the T_2 model trained with all three losses—mean squared error (MSE), information maximization (IM), and contrastive loss (CL)—achieves the lowest mean error rate of 14.88%. This indicates the strong performance of the full configuration under severe corruption scenarios. Removing the contrastive loss (✓ MSE, ✓ IM) slightly increases the mean error rate to 16.04%, suggesting that CL contributes significantly to robustness. Excluding the information maximization loss (✓ MSE, ✓ CL) results in a mean error rate of 16.17%, highlighting the importance of IM in the adaptation process. When MSE is excluded (✓ IM, ✓ CL), the mean error rate is slightly better at 15.89%, reflecting a strong interaction between IM and CL, even in the absence of MSE.

For the CIFAR100-to-CIFAR100C task (Table A.12), similar trends are observed. The T_2 model trained with all three losses achieves the lowest mean error rate of 28.00%. Removing CL (✓ MSE, ✓ IM) increases the mean error rate to 28.57%, demonstrating the importance of CL in enhancing robustness. Excluding IM (✓ MSE, ✓ CL) leads to a mean error rate of 28.35%, showing the critical role of IM in the adaptation process. Finally, removing MSE (✓ IM, ✓ CL) results in a mean error rate of 28.23%, again underscoring the synergy between IM and CL.

The results from both CIFAR10-to-CIFAR10C and CIFAR100-to-CIFAR100C tasks consistently highlight the

Table A.1. Average online classification error rate (%) over 5 runs under the continual TTA setting.

Dataset	Architecture	Source	TENT-cont.	RoTTA	CoTTA	ROID (ours)	Slomo-Fast	Slomo-Fast*
CIFAR10-C	WRN-28	43.5	20.0	19.3	16.5	16.2	15.8	**14.8**
	ResNext-29	46.4	62.2	33.3	34.8	29.3	28.7	**27.9**
IN-C	ResNet-50	82.0	62.6	65.5	66.5	54.5	53.2	**52.0**
	Swin-b	64.0	61.4	62.7	59.2	47.0	46.2	**45.1**
	ViT-b-16	60.2	54.5	58.3	77.0	45.0	44.1	**43.3**
IN-R	ResNet-50	63.8	57.5	52.9	53.0	51.2	50.3	**49.0**
	Swin-b	54.2	53.8	–	52.9	45.8	44.9	**43.8**
	ViT-b-16	56.0	53.3	54.4	56.0	44.2	43.5	**42.5**
IN-Sketch	ResNet-50	75.9	68.7	66.4	67.0	58.6	57.1	**56.0**
	Swin-b	68.4	68.7	–	69.1	58.4	57.2	**56.3**
	ViT-b-16	70.6	59.7	68.6	69.6	58.6	57.4	**56.1**

Observation: Across a wide range of datasets and model architectures—including convolutional networks (WRN-28, ResNet-50), hierarchical transformers (Swin-b), and plain vision transformers (ViT-b-16)—**Slomo-Fast** and its enhanced variant **Slomo-Fast*** consistently outperform prior methods. The strong performance holds across both simple and complex domain shifts, highlighting the robustness and generalizability of our method across architectural paradigms.

benefits of integrating all three losses in the T_2 model. This combination achieves the lowest error rates across diverse corruption types, validating the effectiveness of the proposed design for continual test-time adaptation.

D.4. Ablation Study on Prior Correction and Stochastic Restoration

The results of the ablation study are presented in Tables A.13 and A.14, which evaluate the effect of Prior Correction (PC) applied to the model output and Stochastic Restoration (ST) of the T_2 model on the CIFAR10-to-CIFAR10C and CIFAR100-to-CIFAR100C online continual test-time adaptation tasks, respectively. The evaluations are conducted using the WideResNet-28 and ResNeXt-29 model under the highest corruption severity level (level 5). Classification error rates (%) are reported for 15 corruption types, along with the mean error rate as an overall summary.

In the CIFAR10-to-CIFAR10C task (Table A.13), applying PC to the output and using Stochastic Restoration of the T_2 model achieves the lowest mean error rate of 14.88%. This result demonstrates the effectiveness of combining these techniques for robust adaptation. When Stochastic Restoration is removed, and only PC is applied to the output, the mean error rate increases to 16.11%, indicating the critical role of Stochastic Restoration in enhancing the model's robustness under severe corruptions. Conversely, removing PC while retaining Stochastic Restoration results in a mean error rate of 15.78%, suggesting that Prior Correction also significantly contributes to improved performance. These findings highlight the complementary roles of PC and ST in enhancing the adaptation capabilities of the T_2 model.

In the CIFAR100-to-CIFAR100C task (Table A.14), a similar trend is observed. Applying PC to the output alongside Stochastic Restoration of the T_2 model achieves the lowest mean error rate of 28.00%. Removing Stochastic Restoration while retaining PC increases the mean error rate to 28.48%, demonstrating the importance of Stochastic Restoration for handling severe corruptions. On the other hand, using only Stochastic Restoration without PC results in a mean error rate of 28.08%, highlighting the significant role of Prior Correction in reducing classification errors.

The results from both CIFAR10-to-CIFAR10C and CIFAR100-to-CIFAR100C tasks consistently demonstrate that the combination of Prior Correction and Stochastic Restoration leads to the most effective adaptation.

D.5. Effect of Consistency Loss

Tables A.15 and A.16 present the classification error rates (%) for the CIFAR10-to-CIFAR10C and CIFAR100-to-CIFAR100C online continual test-time adaptation tasks, respectively. These results evaluate the effect of applying a consistency loss between the student model and teacher: T_1, T_2, and T_1 with data augmentation input($T_1(aug)$). The evaluations are conducted using WideResNet-28 for CIFAR10C and ResNeXt-29 for CIFAR100C under the largest corruption severity level (level 5). Classification error rates are reported for 15 corruption types, along with the mean er-

Method	*Gaussian*	*shot*	*impulse*	*defocus*	*glass*	*motion*	*zoom*	*snow*	*frost*	*fog*	*bright.*	*contrast*	*elastic*	*pixelate*	*jpeg*	**Mean**
CIFAR10-C																
Source	72.3	65.7	72.9	46.9	54.3	34.8	42.0	25.1	41.3	26.0	9.3	46.7	26.6	58.4	30.3	43.5
TENT-cont.	25.0	20.3	29.0	13.8	31.7	16.2	14.1	18.6	17.6	17.4	10.8	15.6	24.3	19.7	25.1	20.0
RoTTA	30.3	55.5	70.0	23.8	44.1	20.7	21.0	22.7	16.0	9.4	27.7	27.0	58.6	29.2	33.4	19.3
CoTTA	24.2	21.9	26.5	12.0	27.9	12.7	10.7	15.2	14.6	12.8	7.9	11.2	18.5	14.0	18.1	16.5
ROID	23.7	18.7	26.4	11.5	28.1	12.4	10.1	14.7	14.3	12.0	7.5	9.3	19.8	14.5	20.3	16.2
SloMo-Fast	22.5	18.2	26.0	11.1	27.5	12.1	9.9	14.5	13.7	12.4	7.4	9.5	18.5	14.0	19.6	15.8
SloMo-Fast*	22.4	18.5	24.7	11.9	24.6	12.2	10.1	12.7	12.9	11.4	7.5	9.9	16.2	11.7	15.9	14.8
CIFAR100-C																
Source	73.0	68.0	39.4	29.3	54.1	30.8	28.8	39.4	35.4	30.5	9.3	55.1	37.2	74.7	41.2	46.4
TENT-cont.	37.3	35.6	41.6	37.9	51.3	48.1	48.9	59.8	65.3	73.6	74.2	85.7	89.1	91.1	93.7	62.2
RoTTA	49.1	44.9	45.5	30.2	42.7	29.5	26.1	32.2	30.7	37.5	24.7	29.1	32.6	30.4	36.7	34.8
CoTTA	40.5	38.2	39.8	27.2	38.2	28.4	26.4	33.4	32.2	40.6	25.2	27.0	32.4	28.4	33.8	32.8
ROID	36.5	31.9	33.2	24.9	34.9	26.8	24.3	28.9	28.5	31.1	22.8	24.2	30.7	26.5	34.4	29.3
SloMo-Fast	35.6	30.9	32.7	24.2	32.4	27.0	23.1	28.3	28.9	33.0	22.9	25.0	30.8	27.2	34.7	29.2
SloMo-Fast*	37.8	32.7	33.3	26.2	31.2	26.9	24.3	26.8	26.5	28.4	23.3	24.3	26.1	24.2	27.0	27.9
ImageNet-C																
Source	97.8	97.1	98.2	81.7	89.8	85.2	77.9	83.5	77.1	75.9	41.3	94.5	82.5	79.3	68.6	82.0
TENT-cont.	92.8	91.1	92.5	87.8	90.2	87.2	82.2	82.2	82.0	79.8	48.0	92.5	83.5	75.6	70.4	82.5
RoTTA	89.4	88.6	89.3	83.4	89.1	86.2	80.0	78.9	76.9	74.2	37.4	89.6	79.5	69.0	59.6	78.1
CoTTA	89.1	86.6	88.5	80.9	87.2	81.1	75.8	73.3	75.2	70.5	41.6	85.0	78.1	65.6	61.6	76.0
ROID	76.4	75.3	76.1	77.9	81.7	75.1	69.9	70.9	68.8	64.3	42.5	85.4	69.8	53.0	55.6	54.5
SloMo-Fast	68.6	65.2	64.5	68.2	66.7	57.0	49.7	51.0	56.4	43.1	33.8	57.3	43.9	41.4	45.7	54.2
SloMo-Fast*	68.5	62.6	60.3	65.6	63.4	55.7	50.4	50.4	54.5	43.7	36.3	53.5	43.0	40.7	43.0	52.8

Table A.2. Online classification error rate (%) for the corruption benchmarks at the highest severity level 5 for the Continual TTA setting. For CIFAR10-C the results are evaluated on WideResNet-28, for CIFAR100-C on ResNeXt-29, and for ImageNet-C, ResNet-50 are used. Results marked with (*) indicate that all parameters of the student model are updated; otherwise, only the Batch Normalization layers are updated.

ror rate as a summary.For CIFAR10-C, the best results are achieved by incorporating the consistency loss between the student predictions and the predictions from both T_1 and T_2. For CIFAR100-C, the best performance is obtained by using the consistency loss between the student predictions and the predictions from T_2 and T_1 with augmented samples.

D.6. CTTA Under Cyclic Domain Settings

In continual test-time adaptation, catastrophic forgetting occurs when the model forgets previously learned knowledge while adapting to new domains. To address this, we propose a second teacher model that learns more generalized knowledge compared to the primary teacher model, which is more adapted to the current domain. This helps retain critical knowledge from past domains while enabling adaptation to new ones, mitigating the risk of forgetting. To validate our approach, we conduct an ablation study in cyclic domain settings, where domains are grouped and presented in a cycle. This setup allows us to compare the effectiveness of various methods designed to tackle catastrophic forgetting. Table A.18-A.30 presents the detailed results on the newly proposed benchmark CTTA under cyclic domain settings.

The experimental results demonstrate that our method improves performance when domains repeat, indicating that it retains past knowledge to some extent while adapting to new domains. Specifically, our approach achieves lower error rates compared to state-of-the-art methods. In CIFAR10-C, our method achieves an error rate of 14.89% in Cycle 1 and 14.38% in Cycle 2, showing improvement in error rate as domains are repeated. In contrast, TENT[38], which does not specifically address continual domain adaptation, results in higher error rates, with Cycle 1 at 17.47% and Cycle 2 at 16.64%. While COTTA[40] shows some improvement initially, it does not exhibit reduction in error rates when domains are repeated. ROID[26], on the other hand, shows limited improvement under cyclic domain settings. Compared to state-of-the-art methods, our method demonstrates better retention of past knowledge, leading to more stable performance across cyclic domains. These re-

Method	*Gaussian*	*shot*	*impulse*	*defocus*	*glass*	*motion*	*zoom*	*snow*	*frost*	*fog*	*bright.*	*contrast*	*elastic*	*pixelate*	*jpeg*	Mean
CIFAR10-C																
Source	72.3	65.7	72.9	46.9	54.3	34.8	42.0	25.1	41.3	26.0	9.3	46.7	26.6	58.4	30.3	43.5
TENT-cont.	73.5	70.1	81.4	31.6	60.3	29.6	28.5	30.8	35.3	25.7	13.6	44.2	32.6	70.2	34.9	44.1
CoTTA	38.7	36.0	56.1	36.0	36.8	32.3	31.0	19.9	17.6	27.2	11.7	52.6	30.5	35.8	25.7	32.5
RoTTA	60.0	55.5	70.0	23.8	44.1	20.7	21.3	20.2	22.7	16.0	9.4	22.7	27.0	58.6	29.2	33.4
ROID	37.1	34.3	50.9	24.8	38.1	22.5	22.0	18.8	18.5	18.8	9.9	25.6	27.2	45.7	26.2	28.0
SloMo-Fast	39.1	36.8	53.8	27.5	38.6	24.7	23.5	18.0	18.1	19.2	9.2	33.3	28.9	51.9	24.9	29.7
SloMo-Fast*	33.4	32.1	53.9	26.4	35.0	22.7	23.4	17.9	17.8	19.8	11.4	30.1	25.9	46.4	23.4	28.0
CIFAR100-C																
Source	73.0	68.0	39.4	29.3	54.1	30.8	28.8	39.5	45.8	50.3	29.5	55.1	37.2	74.7	41.2	46.4
TENT-cont.	95.6	95.2	89.2	72.8	82.9	74.4	72.3	78.0	79.7	84.7	71.0	88.5	77.8	96.8	78.7	82.5
CoTTA	54.4	52.7	49.8	36.0	45.8	36.7	33.9	38.9	35.8	52.0	30.4	60.9	40.2	38.0	41.1	43.1
RoTTA	65.0	62.3	39.3	33.4	50.0	34.2	32.6	36.6	36.5	45.0	26.4	41.6	40.6	89.5	48.5	45.4
ROID	40.5	38.0	32.0	28.1	40.5	29.7	27.6	34.1	33.8	41.3	28.7	38.7	34.3	39.7	38.5	35.0
SloMo-Fast	44.6	40.9	30.2	19.6	37.5	28.4	26.3	31.4	31.6	39.5	25.8	37.9	31.3	52.5	39.1	34.4
SloMo-Fast*	41.6	39.2	29.8	28.1	36.7	29.6	27.4	31.3	31.5	37.9	27.1	34.0	32.2	42.3	34.4	33.5
Imagenet-C																
Source	97.8	97.1	98.2	81.7	89.8	85.2	77.9	83.5	77.1	75.9	41.3	94.5	82.5	79.3	68.6	82.0
TENT-cont.	99.2	98.7	99.0	90.5	95.1	90.5	84.6	86.6	84.0	86.5	46.7	98.1	86.1	77.7	72.9	86.4
CoTTA	89.1	86.6	88.5	80.9	87.2	81.1	75.8	73.3	75.2	70.5	41.6	85.0	78.1	65.6	61.6	76.0
RoTTA	89.4	88.6	89.3	83.4	89.1	86.2	80.0	78.9	76.9	74.2	37.4	89.6	79.5	69.0	59.6	78.1
ROID	76.4	75.3	76.1	77.9	81.7	75.1	69.9	70.9	68.8	64.3	42.5	85.4	69.8	53.0	55.6	69.5
SloMo-Fast	80.5	79.4	81.3	76.9	81.3	75.4	72.1	67.8	70.2	65.7	44.0	82.7	72.7	61.9	59.1	71.3
SloMo-Fast*	78.5	77.9	79.1	75.9	78.8	73.6	71.2	67.1	69.1	64.7	43.2	80.7	71.0	60.9	58.1	70.8

Table A.3. Online classification error rate (%) for the corruption benchmarks at the highest severity level 5 for the generalization experiments with mixed domains. For CIFAR10-C the results are evaluated on WideResNet-28, for CIFAR100-C on ResNeXt-29, and for ImageNet-C, ResNet-50 are used. Results marked with (*) indicate that all parameters of the student model are updated; otherwise, only the Batch Normalization layers are updated.

sults highlight the effectiveness of our approach in mitigating catastrophic forgetting and adapting to domain shifts, outperforming existing methods in terms of reduced error rates.

D.7. Catastrophic Fogetting

The figures illustrate the performance of different CTTA methods, including SloMo-Fast, on the CIFAR10-C benchmark, highlighting challenges like catastrophic forgetting and the ability to retain long-term knowledge.

In the standard CTTA setting, as shown in A.2, the SloMo-Fast method achieves consistently low error rates, with a mean error of **15.79%**, outperforming CoTTA (**16.5%**) and ROID (**16.2%**). This demonstrates SloMo-Fast's superior adaptability while avoiding performance degradation seen in other methods.

For mixed domain settings, as shown in A.3, SloMo-Fast maintains the best mean error rate of **28.0%**, compared to CoTTA (**32.5%**) and ROID (**28.0%**). This highlights SloMo-Fast's ability to handle mixed corruption scenarios effectively.

When evaluating performance in a mixed-after-continual setting, as in A.4, SloMo-Fast achieves the lowest mean error rate of **21.34%**, significantly outperforming ROID (**27.37%**) and CoTTA (**26.76%**), showcasing its resilience to catastrophic forgetting.

In the cyclic domain adaptation scenario, as shown in A.5, SloMo-Fast exhibits stable performance, maintaining an average error rate of **14.63%** across repeated domains, compared to ROID's **15.63%**. This demonstrates SloMo-Fast's ability to retain previously learned knowledge without succumbing to forgetting, a common issue in ROID and CoTTA.

Overall, the results validate SloMo-Fast as a robust solution for CTTA, capable of preserving long-term domain knowledge while achieving state-of-the-art performance.

Method	*Gaussian*	*shot*	*impulse*	*defocus*	*glass*	*motion*	*zoom*	*snow*	*frost*	*fog*	*bright.*	*contrast*	*elastic*	*pixelate*	*jpeg*	*Mixed*	*Mean*
CIFAR10-C																	
Source	72.3	65.7	72.9	46.9	54.3	34.8	42.0	25.1	41.3	26.0	9.3	46.7	26.6	58.4	30.3	43.5	43.5
Tent-cont.	24.6	19.8	28.4	13.1	31.2	16.8	14.0	18.9	18.3	16.9	11.4	17.2	25.5	19.7	25.2	39.3	21.3
CoTTA	24.0	21.8	25.7	11.7	27.5	21.6	10.2	15.0	13.9	12.5	7.5	10.8	18.1	13.5	17.8	26.7	16.8
RoTTA	30.2	25.4	34.6	18.1	33.9	14.6	10.8	16.4	14.8	14.2	7.9	12.1	20.5	16.8	19.4	29.5	19.9
Roid	23.6	18.7	26.5	11.6	28.2	12.5	9.9	14.4	13.9	11.7	7.3	9.3	19.7	14.4	20.5	27.3	16.9
SloMo-Fast	23.8	18.8	26.3	12.2	26.5	13.1	10.6	14.3	13.5	12.9	8.1	11.4	18.6	13.2	17.3	26.4	16.7
SloMo-Fast*	22.7	18.5	24.6	12.5	24.7	13.7	12.1	14.4	14.5	12.8	9.6	12.0	16.9	12.6	16.1	21.3	16.2
CIFAR100-C																	
Source	73.0	68.0	39.4	29.3	54.1	30.8	28.8	39.4	35.4	30.5	9.3	55.1	37.2	74.7	41.2	46.4	46.4
Tent-cont.	37.3	35.7	42.1	38.2	51.0	45.9	46.3	55.8	62.1	72.8	72.3	83.9	90.6	92.8	95.3	97.8	63.7
CoTTA	40.8	38.0	39.8	27.2	38.0	28.5	26.4	33.4	32.2	40.2	25.1	26.9	32.1	28.4	33.8	40.9	33.2
RoTTA	49.4	44.7	45.5	30.2	42.3	29.6	25.9	32.0	30.5	37.7	24.7	29.4	32.8	29.9	36.6	40.8	35.1
Roid	36.4	31.9	33.6	24.8	34.8	27.0	24.1	29.1	28.5	31.3	22.8	24.2	30.5	26.4	33.9	34.7	29.6
SloMo-Fast	37.1	32.8	35.0	26.2	35.2	28.0	25.1	29.6	29.0	33.2	23.8	25.2	31.1	27.1	34.8	36.3	29.5
SloMo-Fast*	37.8	32.9	33.1	26.6	31.6	27.1	24.8	26.5	26.2	28.4	23.6	24.3	26.5	24.5	27.1	28.0	28.1
Imagenet-C																	
Source	97.8	97.1	98.2	81.7	89.8	85.2	77.9	83.5	77.1	75.9	41.3	94.5	82.5	79.3	68.6	82.0	82.0
TENT-cont.	71.4	66.4	69.1	82.8	91.0	95.7	97.6	99.0	99.3	99.3	99.2	99.5	99.4	99.3	99.4	99.6	91.8
CoTTA	78.2	68.3	64.2	75.4	71.9	70.1	67.8	72.3	71.6	67.7	62.7	74.4	70.2	67.5	69.2	82.4	70.9
RoTTA	79.6	72.0	69.6	77.1	72.1	73.1	68.6	72.1	73.6	77.1	65.7	90.9	69.4	75.7	75.4	93.8	75.4
Roid	63.6	60.3	61.1	65.1	65.0	52.5	47.4	48.0	54.1	39.9	32.6	53.5	42.1	39.4	44.5	70.5	52.5
SloMo-Fast	63.8	60.8	60.7	63.3	61.8	53.6	47.0	47.8	52.6	41.1	32.3	53.7	41.3	39.2	43.2	63.0	53.7
SloMo-Fast*	65.8	62.8	62.7	65.5	63.8	55.6	48.8	49.8	55.2	42.6	33.3	55.7	43.1	40.6	45.0	65.7	53.6

Table A.4. Online classification error rate (%) for the corruption benchmarks at the highest severity level 5 for the mixed after continual domains TTA setting. For CIFAR10-C the results are evaluated on WideResNet-28, for CIFAR100-C on ResNeXt-29, and for ImageNet-C, ResNet-50 are used. Results marked with (*) indicate that all parameters of the student model are updated; otherwise, only the Batch Normalization layers are updated.

Method	*Gaussian*	*Shot*	*Impulse*	*Defocus*	*Glass*	*Motion*	*Zoom*	*Snow*	*Frost*	*Fog*	*Bright.*	*Contrast*	*Elastic*	*Pixelate*	*JPEG*	Mean
CIFAR10-C																
Source	72.3	65.7	72.9	46.9	54.3	34.8	42.0	25.1	41.3	26.0	9.3	46.7	26.6	58.4	30.3	43.5
TENT-cont.	24.7	22.2	32.4	11.6	32.1	12.9	10.9	16.0	16.1	13.0	7.6	11.1	21.97	17.2	23.6	18.2
RoTTA	30.2	27.4	37.8	13.7	35.9	14.7	12.7	17.8	19.0	15.5	8.0	19.3	23.65	21.0	27.6	21.6
CoTTA	24.0	22.9	27.7	12.2	30.2	13.4	11.7	16.8	17.0	14.4	7.9	12.5	22.50	18.7	22.6	18.3
ROID	23.6	21.7	30.6	11.0	30.4	12.9	10.5	15.2	15.2	12.7	7.6	10.4	21.06	16.2	23.6	17.5
SloMo-Fast	22.7	19.6	26.1	12.8	26.7	13.7	11.5	14.9	15.1	12.8	8.8	12.0	19.92	14.6	19.6	16.7
SloMo-Fast*	24.5	20.3	28.1	11.5	28.7	12.4	10.2	14.5	14.4	12.7	7.6	10.4	19.69	14.8	20.6	16.7
CIFAR100-C																
Source	73.0	68.0	39.4	29.3	54.1	30.8	28.8	39.4	35.4	30.5	9.3	55.1	37.2	74.7	41.2	46.4
TENT-cont.	37.2	34.8	34.4	24.9	37.3	27.5	25.1	30.3	31.9	33.6	23.9	28.1	32.8	28.3	36.8	31.1
RoTTA	49.4	47.5	48.6	29.9	47.2	32.2	30.3	39.0	44.1	44.1	28.9	62.2	40.5	38.9	45.6	41.9
CoTTA	40.8	38.3	40.3	27.8	39.7	29.7	27.7	35.4	34.4	42.8	26.0	30.1	35.5	31.5	37.7	34.5
ROID	36.4	34.1	34.1	24.5	36.3	26.9	24.9	30.1	30.4	33.4	23.4	26.2	32.1	27.9	35.7	30.4
SloMo-Fast	37.9	34.2	34.5	27.2	36.3	29.2	26.3	30.8	30.6	32.3	25.1	27.3	33.5	29.2	36.0	31.4
SloMo-Fast*	37.3	33.7	34.8	25.0	35.7	27.1	24.3	29.5	29.1	33.0	23.1	25.5	31.4	27.1	34.9	30.1
Imagenet-C																
Source	97.8	97.1	98.2	81.7	89.8	85.2	77.9	83.5	77.1	75.9	41.3	94.5	82.5	79.3	68.6	82.0
TENT-cont.	71.4	69.4	70.2	71.9	72.7	58.7	50.7	52.9	58.7	42.6	32.7	73.4	45.5	41.4	47.5	57.3
RoTTA	79.8	79.6	80.4	80.7	81.3	68.3	56.9	58.9	63.1	46.7	32.4	76.2	50.9	45.8	54.1	63.7
CoTTA	78.1	77.8	77.3	80.5	78.2	64.0	52.7	58.0	60.5	43.9	32.9	75.1	48.8	42.3	52.4	61.5
ROID	63.7	61.4	62.4	65.9	65.9	52.9	47.6	48.0	54.1	39.9	32.6	53.9	42.2	39.4	44.6	51.6
SloMo-Fast	67.8	64.8	63.5	67.5	65.7	55.6	48.6	49.7	55.5	41.6	32.5	56.4	42.6	40.0	44.8	53.1
SloMo-Fast*	68.2	62.4	60.4	65.4	63.2	55.7	50.7	50.4	54.4	43.6	36.3	53.5	43.1	40.6	42.9	52.7

Table A.5. Online classification error rate (%) for the corruption benchmarks at the highest severity level (Level 5) in the episodic TTA setting. Adaptation resets to the source model parameters for each domain shift. The results are evaluated on WideResNet-28 for CIFAR10-C, ResNeXt-29 for CIFAR100-C, and ResNet-50 for ImageNet-C. Results marked with (*) indicate that all parameters of the student model are updated; otherwise, only the Batch Normalization layers are updated.

Method	*Mixed*	*Gaussian*	*Shot*	*Impulse*	*Defocus*	*Glass*	*Motion*	*Zoom*	*Snow*	*Frost*	*Fog*	*Bright.*	*Contrast*	*Elastic*	*Pixelate*	*JPEG*	Mean
CIFAR10-C																	
Source	43.5	72.3	65.7	72.9	46.9	54.3	34.8	42.0	25.1	41.3	26.0	9.3	46.7	26.6	58.4	30.3	43.5
TENT-cont.	41.1	43.5	43.3	51.6	34.8	54.0	42.7	40.1	47.7	48.2	44.2	40.0	48.2	52.4	48.9	55.2	46.0
CoTTA	32.4	22.1	19.9	24.6	12.9	26.8	13.4	12.8	15.3	14.5	14.4	9.6	13.9	19.5	14.8	18.7	17.8
ROTTA	33.1	24.9	20.9	29.7	14.8	30.5	14.3	11.1	16.5	15.3	13.1	9.1	12.7	20.3	17.4	19.4	18.9
ROID	28.2	22.1	17.8	25.8	11.3	27.8	12.4	10.0	14.5	14.0	12.4	7.3	9.2	19.5	14.6	20.2	16.7
SloMo-Fast	29.7	21.2	18.2	26.4	11.6	27.5	12.2	9.7	14.2	13.6	12.4	7.5	10.3	18.8	13.7	19.4	16.6
SloMo-Fast*	27.6	18.2	15.9	21.7	11.1	23.0	11.5	9.5	12.7	12.0	10.8	7.2	9.3	16.0	11.3	15.5	14.6
CIFAR100-C																	
Source	46.4	73.0	68.0	39.4	29.3	54.1	30.8	28.8	39.4	35.4	30.5	9.3	55.1	37.2	74.7	41.2	46.4
TENT-cont.	83.8	97.2	97.7	97.9	97.9	98.1	97.8	98.0	97.9	98.2	98.0	97.9	98.2	98.3	98.4	98.5	97.1
CoTTA	43.0	37.6	36.5	38.6	26.7	37.3	28.1	26.8	33.4	32.3	41.2	25.8	27.8	33.4	28.8	34.6	33.3
RoTTA	45.3	38.3	36.0	36.6	27.9	40.3	30.0	27.4	33.2	31.5	37.9	27.1	29.5	34.4	37.2	39.0	34.5
ROID	35.0	33.9	31.6	32.5	24.7	34.9	26.7	24.0	29.2	28.6	31.0	22.8	24.4	30.5	26.7	33.9	29.4
SloMo-Fast	37.5	33.1	31.3	33.6	24.5	34.0	26.7	23.6	28.7	28.1	32.2	22.6	24.5	30.2	26.2	33.6	29.1
SloMo-Fast*	34.6	30.3	27.9	29.4	24.4	28.8	25.4	23.0	25.4	25.3	28.0	22.5	23.4	26.2	24.0	27.9	26.6
Imagenet-C																	
Source	82.0	97.8	97.1	98.2	81.7	89.8	85.2	77.9	83.5	77.1	75.9	41.3	94.5	82.5	79.3	68.6	82.0
TENT-cont.	87.7	89.1	88.1	87.3	88.4	90.6	86.8	80.3	86.3	87.2	81.1	68.5	93.8	84.6	83.9	86.0	85.6
CoTTA	76.0	62.4	61.1	60.9	63.8	64.5	55.7	51.4	54.0	55.3	47.4	40.1	55.7	47.1	43.3	46.1	55.3
RoTTA	78.0	80.0	74.8	74.8	84.6	75.9	68.8	58.4	60.1	61.7	53.0	36.5	69.1	52.1	48.4	50.7	64.2
ROID	69.4	67.4	61.5	62.2	69.7	65.7	57.5	49.5	52.3	58.1	43.3	33.5	58.9	44.9	41.7	45.6	55.1
SloMo-Fast	75.0	66.0	62.0	61.0	67.0	65.0	56.0	48.0	50.0	55.0	42.0	31.0	58.0	44.0	40.0	44.0	55.6
SloMo-Fast*	75.0	65.0	60.0	60.0	66.0	64.0	55.0	46.0	48.0	54.0	40.0	29.0	56.0	41.0	38.0	42.0	54.5

Table A.6. Online classification error rate (%) for the corruption benchmarks at the highest severity level 5 for the continual after mixed domains TTA setting. For CIFAR10-C the results are evaluated on WideResNet-28, for CIFAR100-C on ResNeXt-29, and for ImageNet-C, ResNet-50 are used. Results marked with (*) indicate that all parameters of the student model are updated; otherwise, only the Batch Normalization layers are updated.

Method	*Gaussian*	*Shot*	*Impulse*	*Defocus*	*Glass*	*Motion*	*Zoom*	*Snow*	*Frost*	*Fog*	*Bright.*	*Contrast*	*Elastic*	*Pixelate*	*JPEG*	**Mean**
CIFAR10-C																
Source	72.3	65.7	72.9	46.9	54.3	34.8	42.0	25.1	41.3	26.0	9.3	46.7	26.6	58.4	30.3	43.5
TENT-cont.	15.5	14.7	21.0	13.8	35.7	31.5	29.3	32.7	26.5	25.9	23.7	26.3	31.9	27.9	37.0	26.2
CoTTA	15.9	11.5	13.6	7.7	18.1	9.3	8.5	10.8	9.8	8.4	8.1	8.1	10.5	9.1	12.7	10.8
ROTTA	16.9	11.5	15.1	8.3	19.7	11.1	9.1	12.6	11.1	8.7	8.1	8.5	11.6	10.3	14.6	11.8
ROID	14.1	11.8	15.3	6.7	19.7	9.1	7.5	11.1	10.1	6.8	5.9	6.4	10.5	8.9	14.4	10.5
SloMo-Fast	14.9	11.9	15.3	6.8	19.3	9.1	7.5	10.6	9.5	7.1	6.0	6.7	10.1	8.4	13.5	10.4
SloMo-Fast*	13.3	10.3	12.2	7.0	14.6	8.1	7.2	8.5	7.9	6.7	6.4	6.5	7.9	6.9	9.4	8.9
CIFAR100-C																
Source	73.0	68.0	39.4	29.3	54.1	30.8	28.8	39.4	35.4	30.5	9.3	55.1	37.2	74.7	41.2	46.4
TENT-cont.	36.4	45.1	47.4	47.5	64.6	72.6	73.4	77.1	86.7	96.4	97.8	98.1	98.4	98.6	98.5	75.9
CoTTA	33.7	29.4	29.0	24.8	30.2	25.6	25.0	26.9	26.4	26.5	24.4	25.0	25.6	24.6	27.6	27.0
RoTTA	34.3	28.6	30.1	27.2	35.0	30.6	29.5	33.3	33.5	34.6	32.3	34.5	36.4	36.1	45.2	33.4
ROID	28.5	26.0	22.8	21.3	29.3	23.5	22.1	24.4	24.5	23.0	21.0	21.6	25.0	22.7	29.3	24.3
SloMo-Fast	29.3	26.5	24.3	22.0	29.3	23.8	22.4	24.8	24.6	23.5	21.3	21.9	24.8	22.6	29.1	24.7
SloMo-Fast*	28.2	24.9	23.3	22.4	24.8	22.8	22.2	22.9	22.6	22.3	21.9	21.9	22.5	22.1	23.8	23.3
Imagenet-C																
Source	97.8	97.1	98.2	81.7	89.8	85.2	77.9	83.5	77.1	75.9	41.3	94.5	82.5	79.3	68.6	82.0
TENT-cont.	44.8	54.1	80.1	99.0	99.5	99.5	99.6	99.6	99.6	99.7	99.7	99.8	99.8	99.7	99.8	91.6
CoTTA	44.2	53.3	58.9	65.4	68.6	69.1	71.0	72.4	74.2	73.1	71.5	73.3	74.1	73.0	74.0	67.7
RoTTA	59.4	92.0	98.2	99.1	99.3	99.5	99.6	99.8	99.8	99.8	99.8	99.8	99.8	99.9	99.8	96.4
ROID	42.5	42.7	44.0	45.7	45.5	38.6	40.0	41.1	44.5	32.9	28.5	34.5	35.2	32.1	34.7	38.8
SloMo-Fast	43.0	43.3	44.8	46.3	45.9	39.2	39.9	41.2	44.6	33.3	28.7	34.4	35.2	32.6	34.6	39.1

Table A.7. Online classification error rate (%) for the corruption benchmarks in the gradual domains TTA setting. In this setting, the severity of domain shifts changes incrementally over time, simulating a practical scenario for test-time adaptation. The results are evaluated on WideResNet-28 for CIFAR10-C, ResNeXt-29 for CIFAR100-C, and ResNet-50 for ImageNet-C. Results marked with (*) indicate that all parameters of the student model are updated; otherwise, only the Batch Normalization layers are updated.

Method	*Gaussian*	*Defocus*	*Snow*	*Brightness*	*Shot*	*Glass*	*Frost*	*Contrast*	*Impulse*	*Motion*	*Fog*	*Elastic*	*Pixelate*	*Zoom*	*JPEG*	Mean
CIFAR10-C																
Source	72.3	46.9	25.1	9.3	65.7	54.3	26.0	41.3	72.9	34.8	42.0	26.6	58.4	30.3	43.5	41.3
TENT-cont.	24.6	11.8	15.1	8.1	19.6	30.8	15.8	15.3	30.9	15.6	16.2	22.4	17.8	13.9	24.0	18.8
CoTTA	24.0	11.9	16.1	7.8	20.1	26.8	13.9	10.8	23.2	11.5	12.0	18.2	13.7	9.7	17.6	15.8
RoTTA	30.2	19.0	18.1	8.1	25.2	32.6	15.3	15.3	33.1	16.9	13.4	19.7	16.5	11.3	20.2	19.7
ROID	23.6	11.8	14.4	7.1	19.6	27.3	14.6	9.6	28.3	12.4	12.0	19.4	14.7	10.0	20.9	16.4
SloMo-Fast	22.5	18.6	26.6	11.2	27.0	12.3	10.0	14.4	14.4	12.4	7.3	10.8	18.9	14.2	20.2	16.1
SloMo-Fast*	22.5	12.1	13.9	7.8	16.5	24.4	13.4	10.7	21.7	11.8	11.6	16.6	12.0	9.4	16.3	14.7
CIFAR100-C																
Source	73.0	29.3	39.4	9.3	68.0	54.1	30.5	35.4	39.4	30.8	28.8	37.2	74.7	41.2	46.4	41.2
TENT-cont.	37.3	29.8	35.4	30.1	41.2	49.5	52.6	62.0	70.4	72.5	82.0	87.5	88.7	90.5	93.4	61.5
CoTTA	40.8	36.6	37.7	27.4	37.4	27.3	25.4	34.4	39.9	32.5	25.6	27.5	32.9	28.6	33.7	32.5
ROTTA	49.4	41.6	41.5	31.7	39.2	29.8	26.0	36.1	36.3	31.7	25.5	34.1	32.1	30.8	36.8	34.9
ROID	36.4	32.7	31.8	25.3	34.6	26.9	24.1	28.9	28.8	31.4	22.9	25.3	31.1	26.8	34.8	29.5
SloMo-Fast	37.2	25.4	29.3	23.2	33.0	35.0	29.2	26.1	34.8	27.0	33.3	31.1	27.2	24.3	34.8	30.1
SloMo-Fast*	37.8	26.7	28.4	23.7	29.1	31.7	27.5	25.1	30.3	25.6	28.7	27.3	24.4	23.1	28.4	27.9
ImageNet-C																
Source	97.8	81.7	77.9	41.3	97.1	89.8	75.9	77.1	83.5	85.2	98.2	94.5	82.5	79.3	68.6	82.0
TENT-cont.	81.4	78.9	60.5	35.4	73.5	74.2	61.9	71.8	70.5	65.7	51.5	51.6	47.2	55.7	53.6	62.2
CoTTA	84.6	83.5	63.4	34.6	73.5	74.4	57.9	69.4	64.7	59.4	44.9	45.4	40.5	49.0	43.1	59.2
ROTTA	88.0	92.5	69.7	35.4	84.5	84.8	67.4	77.2	85.3	77.3	51.6	52.6	48.9	58.9	54.7	68.6
ROID	71.7	70.3	55.0	34.1	66.8	68.1	58.5	58.3	65.5	58.2	43.9	44.9	42.0	50.2	47.5	55.7
SloMo-Fast	68.8	69.6	52.0	33.7	64.2	67.0	56.7	58.1	63.3	56.3	43.4	43.9	41.6	49.3	46.4	54.3
SloMo-Fast*	68.1	67.7	53.0	37.6	58.6	62.1	54.4	55.7	56.9	52.7	43.3	43.1	41.2	46.3	43.5	52.3

Table A.8. Online classification error rate (%) for the corruption benchmarks at the highest severity level (Level 5) in the continual-cross group TTA setting. In this setting, domains are sequentially sampled from different corruption groups (e.g., Noise, Blur). The results are evaluated on WideResNet-28 for CIFAR10-C, ResNeXt-29 for CIFAR100-C, and ResNet-50 for ImageNet-C. Results marked with (*) indicate that all parameters of the student model are updated; otherwise, only the Batch Normalization layers are updated.

CIFAR10-C

Method	*Brightness*	*Snow*	*Fog*	*Elastic*	*JPEG*	*Motion*	*Frost*	*Zoom*	*Contrast*	*Defocus*	*Glass*	*Pixelate*	*Shot*	*Gaussian*	*Impulse*	**Mean**
Source	9.3	25.1	26.0	26.6	30.3	34.8	41.3	42.0	43.5	46.7	46.9	54.3	58.4	65.7	72.3	72.9
TENT-cont.	7.6	15.4	13.1	22.2	25.5	15.5	17.2	14.8	16.4	13.9	30.9	18.4	24.6	25.3	33.6	19.6
CoTTA	8.0	16.4	13.3	21.0	20.6	12.2	14.5	9.9	10.4	9.8	25.4	13.9	18.1	19.5	22.6	15.7
RoTTA	8.0	17.8	14.7	22.3	24.9	13.2	16.4	12.9	11.0	11.9	30.2	16.2	20.9	19.6	27.3	17.8
ROID	7.6	14.7	11.9	19.5	21.0	12.5	14.4	10.1	9.4	10.3	28.2	14.2	19.0	19.9	25.5	15.9
SloMo-Fast	19.3	17.4	24.9	10.5	27.1	12.3	10.3	14.1	14.0	11.8	7.6	10.2	19.0	13.9	20.7	15.5
SloMo-Fast*	8.5	13.4	11.6	18.1	18.8	11.6	12.6	9.3	9.7	9.2	22.4	12.0	15.1	15.9	20.7	13.9

CIFAR100-C

Method	*Zoom*	*Defocus*	*Brightness*	*Motion*	*Elastic*	*Impulse*	*Snow*	*JPEG*	*Frost*	*Fog*	*Glass*	*Contrast*	*Shot*	*Gaussian*	*Pixelate*	**Mean**
Source	9.3	28.8	29.3	30.5	30.8	35.4	37.2	39.4	39.4	41.2	46.4	54.1	55.1	68.0	73.0	74.7
TENT-cont.	25.0	25.7	25.2	28.6	34.6	40.3	42.6	50.1	52.7	59.3	70.9	76.9	82.1	88.1	90.0	52.8
CoTTA	27.7	27.0	25.4	28.2	33.5	38.2	32.8	35.3	32.0	40.0	36.1	27.1	35.0	36.2	28.2	32.2
RoTTA	30.3	28.7	26.2	30.1	34.0	40.6	32.1	37.5	30.7	36.8	36.5	29.5	36.6	35.2	30.7	33.0
ROID	36.3	31.9	33.5	24.8	34.9	26.9	24.1	29.1	28.6	31.0	23.0	24.4	30.6	26.4	34.1	29.3
SloMo-Fast	35.2	32.4	33.6	25.0	33.5	27.3	24.2	29.3	29.0	33.3	23.1	25.1	30.8	27.3	33.1	29.5
SloMo-Fast*	37.8	32.6	33.3	26.0	32.3	27.1	24.0	27.1	26.9	28.9	23.0	24.2	27.0	24.6	28.4	28.2

ImageNet-C

Method	*Brightness*	*JPEG*	*Fog*	*Frost*	*Zoom*	*Pixelate*	*Defocus*	*Elastic*	*Snow*	*Motion*	*Glass*	*Contrast*	*Shot*	*Gaussian*	*Impulse*	**Mean**
Source	41.3	68.6	75.9	77.1	77.9	79.3	81.7	82.0	82.5	83.5	85.2	89.8	94.5	97.1	97.8	98.2
TENT-cont.	34.1	54.4	46.6	61.9	55.1	46.1	75.6	49.4	59.3	63.3	71.8	69.7	72.3	71.4	69.5	60.0
CoTTA	34.6	56.6	45.8	60.5	51.8	40.8	68.6	44.3	50.3	52.4	61.8	55.0	55.7	55.5	53.4	52.5
RoTTA	34.3	58.8	53.4	68.5	63.7	50.6	81.1	56.3	61.6	70.0	77.9	74.7	78.9	74.0	72.8	65.1
ROID	33.4	48.0	43.2	58.4	51.7	42.3	67.3	46.3	51.7	57.7	65.6	60.7	64.3	61.8	61.5	54.3
SloMo-Fast	32.6	46.1	41.4	55.7	49.1	41.0	66.0	43.4	49.7	55.1	65.5	57.8	63.1	63.4	62.6	52.8

Table A.9. Online classification error rate (%) for the corruption benchmarks at the highest severity level (Level 5) in the continual-easy-to-hard TTA setting. In this setting, domains are sorted sequentially from low to high error based on the initial source model's performance. The results are evaluated on WideResNet-28 for CIFAR10-C, ResNeXt-29 for CIFAR100-C, and ResNet-50 for ImageNet-C. Results marked with (*) indicate that all parameters of the student model are updated; otherwise, only the Batch Normalization layers are updated.

CIFAR10-C

Method	*Impulse*	*Gaussian*	*Shot*	*Pixelate*	*Glass*	*Defocus*	*Contrast*	*Zoom*	*Frost*	*Motion*	*Jpeg*	*Elastic*	*Fog*	*Snow*	*Brightness*	Mean
Source	72.9	72.3	65.7	58.4	54.3	46.9	46.7	43.5	42.0	41.3	34.8	30.3	26.6	26.0	25.1	9.3
TENT-cont.	32.4	23.6	22.2	19.9	31.6	16.0	17.1	15.1	20.2	19.3	24.9	26.0	21.2	21.3	13.7	21.6
CoTTA	27.9	24.2	22.6	17.6	28.1	11.7	13.5	10.9	14.8	12.3	19.1	18.6	13.0	14.6	8.1	17.1
RoTTA	37.8	27.5	24.5	21.5	33.3	14.0	15.2	12.0	15.6	13.9	20.2	19.6	13.0	14.4	8.3	19.4
ROID	30.5	21.7	18.2	14.9	26.6	11.2	9.6	10.0	14.6	11.9	21.1	19.3	12.9	14.5	7.3	16.3
SloMo-Fast	21.4	17.9	29.5	11.1	26.9	11.9	10.2	14.1	14.2	12.5	7.3	10.7	18.5	14.7	20.6	16.1
SloMo-Fast*	29.0	21.2	18.5	14.4	24.8	11.5	11.2	9.9	12.7	11.4	16.8	16.7	11.8	12.5	7.4	15.3

CIFAR100-C

Method	*Pixelate*	*Gaussian*	*Shot*	*Contrast*	*Glass*	*Fog*	*Frost*	*Jpeg*	*Snow*	*Impulse*	*Elastic*	*Motion*	*Brightness*	*Defocus*	*Zoom*	Mean
Source	74.7	73.0	68.0	55.1	54.1	46.4	41.2	39.4	39.4	37.2	35.4	30.8	30.5	29.3	28.8	9.3
TENT-cont.	28.3	37.0	37.4	42.3	49.4	55.9	61.3	70.8	76.0	84.5	88.8	89.7	91.8	92.9	94.1	66.7
CoTTA	31.5	40.0	37.6	28.8	37.6	41.8	32.7	35.8	33.5	38.6	33.6	27.6	25.0	25.9	25.6	33.0
RoTTA	39.0	49.1	42.9	44.9	41.9	39.7	31.6	37.5	30.4	39.8	33.0	28.0	24.8	26.4	25.8	35.7
ROID	34.3	31.6	33.0	23.3	34.6	26.4	23.8	29.4	28.9	32.6	23.1	26.1	31.9	28.2	34.8	29.5
SloMo-Fast	29.1	36.0	32.9	26.4	35.3	34.5	29.5	35.4	29.2	34.4	31.4	26.9	23.1	24.4	24.0	30.2
SloMo-Fast*	29.8	33.1	31.1	27.1	31.9	31.0	27.3	30.6	27.0	30.5	28.3	25.8	23.1	23.0	23.2	28.2

ImageNet-C

Method	*Impulse*	*Gaussian*	*Shot*	*Contrast*	*Glass*	*Motion*	*Snow*	*Elastic*	*Defocus*	*Pixelate*	*Zoom*	*Frost*	*Fog*	*JPEG*	*Brightness*	Mean
Source	98.2	97.8	97.1	94.5	89.8	85.2	83.5	82.5	82.0	81.7	79.3	77.9	77.1	75.9	68.6	41.3
TENT-cont.	81.9	75.8	71.8	74.1	74.9	65.8	61.2	50.1	73.9	47.6	56.2	63.9	52.0	53.5	39.0	62.8
CoTTA	84.6	83.0	80.0	78.3	78.5	67.9	60.1	51.5	73.7	45.4	54.6	58.0	47.2	47.5	37.7	63.2
RoTTA	88.5	83.7	82.1	96.2	84.7	73.3	67.6	55.9	76.7	50.0	58.6	66.0	53.4	54.4	34.4	68.4
ROID	71.7	63.6	61.0	60.5	67.5	57.1	52.8	44.7	70.8	41.9	50.2	59.0	43.2	48.1	34.3	55.1
SloMo-Fast	67.9	66.6	64.0	59.6	66.9	56.5	51.2	44.0	67.4	41.7	49.9	56.2	42.8	46.5	33.8	54.3
SloMo-Fast*	68.5	62.5	60.6	65.5	63.1	55.7	50.5	50.4	54.4	43.7	36.4	53.7	43.0	40.6	43.2	52.8

Table A.10. Online classification error rate (%) for the corruption benchmarks at the highest severity level (Level 5) in the continual-hard-to-easy TTA setting. In this setting, domains are sorted sequentially from high to low error based on the initial source model's performance. The results are evaluated on WideResNet-28 for CIFAR10-C, ResNeXt-29 for CIFAR100-C, and ResNet-50 for ImageNet-C. Results marked with (*) indicate that all parameters of the student model are updated; otherwise, only the Batch Normalization layers are updated.

Table A.11. Evaluating the effect of our proposed loss on T_2, evaluated on the CIFAR10-to-CIFAR10C online continual test-time adaptation task. Results are reported as classification error rates (%) using a WideResNet-28 model with corruption severity level 5. Mean squared error (MSE), information maximization (IM), and contrastive loss (CL).

Design Choices			Error Rate (%)														
MSE	IM	CL	Gaussian	Shot	Impulse	Defocus	Glass	Motion	Zoom	Snow	Frost	Fog	Brightness	Contrast	Elastic	Pixelate	JPEG	Mean
✓		✓	22.5	18.4	25.1	13.3	24.8	14.0	12.5	14.5	14.3	13.4	10.0	12.4	17.2	13.1	16.5	16.1
✓	✓		23.2	18.9	25.4	12.0	25.5	13.4	11.9	14.4	14.3	12.7	9.4	12.1	17.2	12.7	16.7	16.0
	✓	✓	22.6	18.5	24.6	13.0	24.6	13.6	12.0	14.3	14.1	13.1	9.6	12.1	17.2	12.6	15.9	15.8
✓	✓	✓	22.4	18.5	24.7	11.9	24.6	12.2	10.1	12.7	12.9	11.4	7.5	9.9	16.2	11.7	15.9	14.8

Table A.12. Evaluating the effect of our proposed loss on T_2, evaluated on the CIFAR100-to-CIFAR100C online continual test-time adaptation task. Results are reported as classification error rates (%) using a ResNeXt-29 model with corruption severity level 5. Mean squared error (MSE), information maximization (IM), and contrastive loss (CL).

Design Choices			Error Rate (%)														
MSE	IM	CL	Gaussian	Shot	Impulse	Defocus	Glass	Motion	Zoom	Snow	Frost	Fog	Brightness	Contrast	Elastic	Pixelate	JPEG	Mean
✓	✓		38.1	33.0	33.9	26.7	32.4	27.6	25.0	27.4	26.6	29.4	23.8	24.8	26.5	24.9	27.8	28.5
✓		✓	38.9	33.2	33.4	26.6	32.0	27.3	24.8	26.7	27.0	28.3	23.6	24.2	26.7	24.7	27.2	28.3
	✓	✓	38.0	32.6	33.1	26.8	31.5	26.9	24.8	27.0	26.9	28.2	23.7	24.6	26.6	24.8	27.2	28.2
✓	✓	✓	37.9	32.5	33.2	26.5	31.4	26.8	24.4	26.5	26.3	28.4	23.5	24.6	26.3	24.2	27.1	28.0

Table A.13. Classification error rate (%) for the CIFAR10-to-CIFAR10C online continual test-time adaptation task. Results are evaluated using the WideResNet-28 model with corruption severity level 5. Prior Correction (PC) is applied to the model output, and Stochastic Restoration (ST) is applied to the T_2 model.

Design Choices		Error Rate (%)															
PC	ST	Gaussian	Shot	Impulse	Defocus	Glass	Motion	Zoom	Snow	Frost	Fog	Brightness	Contrast	Elastic	Pixelate	JPEG	Mean
✓		22.6	17.8	23.8	13.8	24.7	14.9	12.4	14.4	14.3	13.7	10.3	12.5	17.3	12.8	15.7	16.1
	✓	22.5	18.5	24.4	12.8	24.7	13.3	11.7	14.4	14.0	13.0	9.6	11.7	17.0	12.5	15.8	15.7
✓	✓	22.4	18.5	24.7	11.9	24.6	12.2	10.1	12.7	12.9	11.4	7.5	9.9	16.2	11.7	15.9	14.8

Table A.14. Classification error rate (%) for the CIFAR100-to-CIFAR100C online continual test-time adaptation task. Results are evaluated using the ResNeXt-29 model with corruption severity level 5. Prior Correction (PC) is applied to the model output, and Stochastic Restoration (ST) is applied to the T_2 model.

Design Choices		Error Rate (%)															
PC	ST	Gaussian	Shot	Impulse	Defocus	Glass	Motion	Zoom	Snow	Frost	Fog	Brightness	Contrast	Elastic	Pixelate	JPEG	Mean
✓		37.3	32.6	33.7	27.4	32.2	27.5	25.1	27.2	26.8	29.3	24.0	24.6	27.0	24.6	27.4	28.4
	✓	37.9	32.5	33.0	26.7	31.5	27.2	24.7	26.6	26.4	28.3	23.4	24.5	26.4	24.5	27.0	28.0
✓	✓	37.9	32.5	33.2	26.5	31.4	26.8	24.4	26.5	26.3	28.4	23.5	24.6	26.3	24.2	27.1	28.0

Table A.15. Classification error rate (%) for the standard CIFAR10-to-CIFAR10C online continual test-time adaptation task. Results are evaluated on WideResNet-28 with the largest corruption severity level 5. The consistency loss calculated between student and teachers. T_1 indicates consistency loss calculated between student and teacher 1, T_2 indicates consistency loss calculated between student and teacher 2, $T_1(aug)$ indicates consistency loss calculated between student and teacher 1 where the input of teacher is augementation of input images.

Design Choices			Error Rate (%)														
T1	T_2	T1(aug)	Gaussian	Shot	Impulse	Defocus	Glass	Motion	Zoom	Snow	Frost	Fog	Brightness	Contrast	Elastic	Pixelate	JPEG	Mean
✓	✓		22.7	18.1	24.2	12.8	25.5	13.5	11.5	15.0	14.2	13.3	9.7	12.2	17.0	13.3	15.7	15.9
✓		✓	22.7	18.7	25.4	12.9	25.7	14.3	12.3	15.3	15.1	13.4	10.3	13.3	17.8	13.4	17.1	16.5
	✓	✓	22.6	18.2	24.8	13.2	25.1	14.6	12.2	14.5	14.6	13.1	10.2	12.3	17.6	12.9	16.4	16.1

Table A.16. Classification error rate (%) for the standard CIFAR100-to-CIFAR100C online continual test-time adaptation task. Results are evaluated on ResNeXt-29 with the largest corruption severity level 5. The consistency loss calculated between student and teachers. T_1 indicates consistency loss calculated between student and teacher 1, T_2 indicates consistency loss calculated between student and teacher 2, $T_1(aug)$ indicates consistency loss calculated between student and teacher 1 where the input of teacher is augementation of input images.

Design Choices			Error Rate (%)														
T1	T_2	T1(aug)	Gaussian	Shot	Impulse	Defocus	Glass	Motion	Zoom	Snow	Frost	Fog	Brightness	Contrast	Elastic	Pixelate	JPEG	Mean
✓	✓		38.2	32.9	33.9	26.3	32.0	27.0	24.7	27.3	26.6	28.9	23.7	24.0	26.5	24.3	27.1	28.2
✓		✓	38.1	33.1	33.9	27.1	32.5	27.4	25.4	27.7	27.0	29.0	24.2	25.6	27.7	25.6	28.6	28.9
	✓	✓	37.3	32.7	33.0	26.3	31.6	27.2	24.7	26.9	26.3	28.3	23.6	24.7	26.7	24.6	27.1	28.1

Method	Repetition	CIFAR100-C						CIFAR10-C						Imagenet-C					
		Noise	Blur	Weather	Digital	Distortion	Avg. Error	Noise	Blur	Weather	Digital	Distortion	Avg. Error	Noise	Blur	Weather	Digital	Distortion	Avg. Error
TENT	Cycle 1	38.28	31.14	32.93	25.04	34.09	32.29	23.66	16.95	15.22	9.07	20.09	17.47	76.35	69.32	57.09	55.31	50.69	61.75
	Cycle 2	47.88	37.12	37.93	25.18	38.95	37.41	23.66	16.95	15.22	9.07	20.09	16.64	69.31	65.14	54.19	52.27	47.47	57.68
	Avg.	43.08	34.13	35.43	25.11	36.52	34.85	23.66	16.95	15.22	9.07	20.09	17.06	72.83	67.23	55.64	53.79	49.08	59.72
COTTA	Cycle 1	36.52	29.43	30.98	23.56	32.75	30.96	23.16	14.98	15.57	10.01	20.63	17.23	82.38	76.91	59.52	56.98	52.82	65.72
	Cycle 2	44.67	34.69	35.93	23.97	36.39	34.69	23.15	15.54	15.15	9.86	20.06	16.28	78.63	72.06	55.52	53.79	47.91	61.58
	Avg.	39.60	32.06	33.46	23.77	34.57	33.70	23.15	15.26	15.36	9.94	20.35	16.75	80.5	74.48	57.52	55.38	50.36	63.65
RoTTA	Cycle 1	46.66	33.82	40.83	41.70	41.17	40.84	30.08	18.98	17.00	12.39	23.95	20.64	84.36	76.76	63.04	58.09	55.21	67.49
	Cycle 2	43.72	29.96	34.31	32.09	36.35	34.70	26.18	16.66	16.02	12.39	21.68	18.23	80.53	72.54	60.50	57.35	52.76	64.74
	Avg.	45.19	31.89	37.57	36.9	38.76	37.77	28.13	17.82	16.51	12.39	22.81	19.44	82.44	74.65	61.77	57.72	53.98	66.11
ROID	Cycle 1	33.94	27.58	30.11	24.09	31.20	29.38	22.16	15.52	13.55	8.48	18.44	16.08	65.23	61.95	51.22	46.46	44.79	53.93
	Cycle 2	32.43	28.31	29.29	23.21	30.55	28.52	22.16	15.52	13.55	8.48	18.44	15.17	61.37	59.62	50.81	45.74	44.22	52.35
	Avg.	**33.18**	27.95	29.70	23.65	30.87	28.95	22.16	15.52	13.55	**8.48**	18.44	15.63	63.3	60.78	51.02	46.1	44.5	53.14
SloMo-Fast	Cycle 1	33.29	27.02	26.96	24.84	25.79	**27.98**	20.62	15.21	13.09	10.23	14.02	**14.89**	66.33	60.70	50.20	25.01	43.55	53.22
	Cycle 2	33.29	27.02	26.96	24.84	25.79	**27.18**	20.62	15.21	13.09	10.23	14.02	**14.38**	63.58	59.46	49.30	44.88	43.25	52.09
	Avg.	33.29	**27.02**	**26.96**	24.84	**25.79**	**27.58**	**20.62**	**15.21**	**13.09**	10.23	**14.02**	**14.63**	64.95	60.08	49.75	34.95	43.4	52.66

Table A.17. Our Proposed Cyclic TTA results of SloMo-Fast compared with existing methods on CIFAR10-C and CIFAR100-C for different domain groups. Each subgroup completes a cycle of seeing different test domains twice. (Gaussian, Shot, Impulse): Noise, (Defocus, Glass, Motion, Zoom): Blur, (Snow, Frost, Fog): Weather, (Brightness, Contrast): Digital, (Elastic, Pixelate, JPEG): . SloMo-Fast achieves the best performance across both datasets.

Method	Subgroup	Cycle 1			Cycle 2		
		Domain	Error (%)	Avg	Domain	Error (%)	Avg
TENT	Noise	gaussian	23.42	23.66	gaussian	24.87	25.49
		shot	21.98		shot	24.37	
		impulse	25.58		impulse	21.74	
	Blur	defocus	11.81	16.95	defocus	11.81	16.95
		glass	29.76		glass	29.76	
		motion	14.01		motion	14.01	
		zoom	12.23		zoom	12.23	
	Weather	snow	16.34	15.22	snow	14.98	14.99
		frost	15.94		frost	15.44	
		fog	14.10		fog	14.55	
	Digital	brightness	7.91	9.07	brightness	7.67	8.78
		contrast	10.81		contrast	9.89	
	Distortion	elastic	22.11	20.09	elastic	20.55	19.47
		pixel	16.22		pixel	15.54	
		jpeg	23.77		jpeg	22.33	
		Cycle 1 Avg: 17.47%			Cycle 2 Avg: 17.14%		

Table A.18. Detailed Evaluation Results for TENT on CIFAR10-C under Cyclic Domain Settings

Method	Subgroup	Cycle 1			Cycle 2		
		Domain	Error (%)	Avg	Domain	Error (%)	Avg
TENT	Noise	gaussian	38.12	38.28	gaussian	47.32	47.88
		shot	38.45		shot	48.23	
		impulse	38.27		impulse	48.09	
	Blur	defocus	30.87	31.14	defocus	37.00	37.12
		glass	31.19		glass	36.78	
		motion	30.75		motion	37.39	
		zoom	31.27		zoom	37.50	
	Weather	snow	33.05	32.93	snow	36.88	37.93
		frost	33.21		frost	36.32	
		fog	32.55		fog	38.58	
	Digital	brightness	25.32	25.04	brightness	24.95	25.18
		contrast	24.76		contrast	25.41	
	Distortion	elastic	33.72	34.09	elastic	39.05	38.95
		pixel	34.56		pixel	39.14	
		jpeg	33.98		jpeg	38.66	
		Cycle 1 Avg: 32.29%			Cycle 2 Avg: 37.41%		

Table A.19. Detailed Evaluation Results for TENT on CIFAR100-C under Cyclic Domain Settings

Method	Subgroup	Cycle 1			Cycle 2		
		Domain	Error (%)	Avg	Domain	Error (%)	Avg
TENT	Noise	gaussian	81.38	76.35	gaussian	70.78	69.31
		shot	74.82		shot	68.50	
		impulse	72.86		impulse	68.66	
	Blur	defocus	81.66	69.32	defocus	72.56	65.14
		glass	77.04		glass	72.72	
		motion	65.18		motion	62.26	
		zoom	53.40		zoom	53.00	
	Weather	snow	62.02	57.09	snow	56.38	54.19
		frost	62.66		frost	60.58	
		fog	46.58		fog	45.62	
	Digital	brightness	34.22	55.31	brightness	33.20	52.27
		contrast	76.40		contrast	71.34	
	Distortion	elastic	52.92	50.69	elastic	47.82	47.47
		pixel	46.36		pixel	44.22	
		jpeg	52.78		jpeg	50.36	
		Cycle 1 Avg: 61.75%			Cycle 2 Avg: 57.68%		

Table A.20. Detailed Evaluation Results for TENT on Imagenet-C under Cyclic Domain Settings

Method	Subgroup	Cycle 1			Cycle 2		
		Domain	Error (%)	Avg.	Domain	Error (%)	Avg.
COTTA	Noise	gaussian	36.14	36.52	gaussian	44.23	44.67
		shot	36.84		shot	44.98	
		impulse	36.57		impulse	44.81	
	Blur	defocus	29.12	29.43	defocus	34.45	34.69
		glass	29.55		glass	34.08	
		motion	28.99		motion	34.72	
		zoom	29.36		zoom	34.51	
	Weather	snow	31.25	30.98	snow	35.45	35.93
		frost	30.84		frost	35.21	
		fog	30.85		fog	37.13	
	Digital	brightness	23.28	23.56	brightness	23.95	23.97
		contrast	23.84		contrast	24.09	
	Distortion	elastic	32.48	32.75	elastic	36.54	36.39
		pixel	32.88		pixel	36.19	
		jpeg	32.89		jpeg	36.44	
		Cycle 1 Avg: 30.96%			Cycle 2 Avg: 34.69%		

Table A.21. Detailed Evaluation Results for COTTA on CIFAR100-C under Cyclic Domain Settings

Method	Subgroup	Cycle 1			Cycle 2		
		Domain	Error (%)	Avg.	Domain	Error (%)	Avg.
COTTA	Noise	gaussian	84.54	82.38	gaussian	80.64	78.63
		shot	81.94		shot	78.30	
		impulse	80.66		impulse	76.96	
	Blur	defocus	86.00	76.91	defocus	79.48	72.06
		glass	83.74		glass	77.06	
		motion	73.82		motion	70.00	
		zoom	64.06		zoom	61.70	
	Weather	snow	65.04	59.52	snow	60.76	55.52
		frost	61.38		frost		
		fog	47.80		fog	44.42	
	Digital	brightness	34.64	56.98	brightness	34.22	53.79
		contrast	79.32		contrast	73.36	
	Distortion	elastic	55.72	52.82	elastic	50.84	47.91
		pixel	48.10		pixel	42.48	
		jpeg	54.64		jpeg	50.40	
		Cycle 1 Avg: 65.72%			Cycle 2 Avg: 61.58%		

Table A.22. Detailed Evaluation Results for COTTA on Imagenet-C under Cyclic Domain Settings

Method	Subgroup	Cycle 1			Cycle 2		
		Domain	Error (%)	Avg	Domain	Error (%)	Avg
ROID	Noise	gaussian	23.94	22.32	gaussian	20.62	22.16
		shot	22.41		shot	21.00	
		impulse	20.62		impulse	24.87	
	Blur	defocus	10.52	15.52	defocus	10.52	15.52
		glass	28.20		glass	28.20	
		motion	12.06		motion	12.06	
		zoom	10.06		zoom	10.06	
	Weather	snow	15.12	13.55	snow	14.01	13.24
		frost	14.41		frost	13.79	
		fog	12.04		fog	11.92	
	Digital	brightness	7.76	8.48	brightness	7.37	8.27
		contrast	9.61		contrast	9.17	
	Distortion	elastic	21.08	18.44	elastic	19.16	17.90
		pixel	15.22		pixel	14.51	
		jpeg	20.62		jpeg	20.02	
		Cycle 1 Avg: 16.08%			Cycle 2 Avg: 15.17%		

Table A.23. Detailed Evaluation Results for ROID on CIFAR10-C under Cyclic Domain Settings

Method	Subgroup	Cycle 1			Cycle 2		
		Domain	Error (%)	Avg	Domain	Error (%)	Avg
ROID	Noise	gaussian	32.34	32.53	gaussian	33.67	33.83
		shot	33.12		shot	34.58	
		impulse	32.12		impulse	33.25	
	Blur	defocus	27.12	26.44	defocus	28.44	28.31
		glass	25.67		glass	27.23	
		motion	26.22		motion	28.23	
		zoom	26.65		zoom	29.34	
	Weather	snow	28.77	30.11	snow	29.29	29.70
		frost	28.06		frost	28.77	
		fog	33.50		fog	31.03	
	Digital	brightness	23.64	24.09	brightness	22.46	23.21
		contrast	24.53		contrast	23.95	
	Distortion	elastic	32.08	31.20	elastic	30.86	30.55
		pixel	27.28		pixel	26.90	
		jpeg	34.23		jpeg	33.88	
		Cycle 1 Avg: 29.38%			Cycle 2 Avg: 28.52%		

Table A.24. Detailed Evaluation Results for ROID on CIFAR100-C under Cyclic Domain Settings

Method	Subgroup	Cycle 1			Cycle 2		
		Domain	Error (%)	Avg	Domain	Error (%)	Avg
ROID	Noise	gaussian	72.24	65.23	gaussian	61.84	61.37
		shot	61.54		shot	60.46	
		impulse	61.92		impulse	61.80	
	Blur	defocus	72.68	61.95	defocus	66.64	59.62
		glass	67.26		glass	66.02	
		motion	58.36		motion	57.02	
		zoom	49.50		zoom	48.82	
	Weather	snow	52.54	51.22	snow	51.38	50.81
		frost	57.96		frost	57.76	
		fog	43.16		fog	43.30	
	Digital	brightness	33.30	46.46	brightness	33.88	45.74
		contrast	59.62		contrast	57.60	
	Distortion	elastic	45.32	44.79	elastic	44.10	44.22
		pixel	42.40		pixel	42.36	
		jpeg	46.64		jpeg	46.20	
		Cycle 1 Avg: 53.93%			Cycle 2 Avg: 52.35%		

Table A.25. Detailed Evaluation Results for ROID on Imagenet-C under Cyclic Domain Settings

Method	Subgroup	Cycle 1			Cycle 2		
		Domain	Error (%)	Avg	Domain	Error (%)	Avg
RoTTA	Noise	gaussian	30.21	30.08	gaussian	25.50	26.18
		shot	25.43		shot	22.32	
		impulse	34.59		impulse	30.72	
	Blur	defocus	13.80	18.98	defocus	11.33	16.66
		glass	36.19		glass	31.81	
		motion	14.78		motion	13.49	
		zoom	11.13		zoom	10.01	
	Weather	snow	17.81	17.00	snow	16.27	16.02
		frost	17.68		frost	15.53	
		fog	15.52		fog	13.30	
	Digital	brightness	8.06	12.39	brightness	8.83	12.39
		contrast	18.35		contrast	14.32	
	Distortion	elastic	23.64	23.95	elastic	22.15	21.68
		pixel	21.65		pixel	19.46	
		jpeg	26.57		jpeg	23.44	
		Cycle 1 Avg: 20.64%			Cycle 2 Avg: 18.23%		

Table A.26. Detailed Evaluation Results for RoTTA on CIFAR10-C under Cyclic Domain Settings

Method	Subgroup	Cycle 1			Cycle 2		
		Domain	Error (%)	Avg	Domain	Error (%)	Avg
RoTTA	Noise	gaussian	49.48	46.66	gaussian	41.89	43.72
		shot	44.87		shot	39.18	
		impulse	45.62		impulse	41.29	
	Blur	defocus	29.94	33.82	defocus	25.95	29.96
		glass	47.33		glass	40.52	
		motion	30.86		motion	28.32	
		zoom	27.16		zoom	25.07	
	Weather	snow	39.00	40.83	snow	32.99	34.31
		frost	41.40		frost	33.15	
		fog	42.09		fog	36.78	
	Digital	brightness	28.95	41.70	brightness	26.63	32.09
		contrast	54.44		contrast	37.56	
	Distortion	elastic	40.58	41.17	elastic	35.83	36.35
		pixel	40.05		pixel	33.93	
		jpeg	42.89		jpeg	39.28	
		Cycle 1 Avg: 40.84%			Cycle 2 Avg: 34.70%		

Table A.27. Detailed Evaluation Results for RoTTA on CIFAR100-C under Cyclic Domain Settings

Method	Subgroup	Cycle 1			Cycle 2		
		Domain	Error (%)	Avg	Domain	Error (%)	Avg
RoTTA	Noise	gaussian	87.98	84.36	gaussian	81.94	80.53
		shot	82.74		shot	80.28	
		impulse	82.36		impulse	79.38	
	Blur	defocus	84.66	76.76	defocus	79.40	72.54
		glass	86.60		glass	81.941	
		motion	75.60		motion	71.58	
		zoom	60.16		zoom	57.22	
	Weather	snow	67.04	63.04	snow	64.90	60.50
		frost	67.48		frost	64.96	
		fog	54.60		fog	51.64	
	Digital	brightness	34.54	58.09	brightness	35.94	57.35
		contrast	81.64		contrast	78.76	
	Distortion	elastic	55.44	55.21	elastic	53.78	52.76
		pixel	52.10		pixel	49.34	
		jpeg	58.10		jpeg	55.16	
		Cycle 1 Avg:67.49%			Cycle 2 Avg: 64.74%		

Table A.28. Detailed Evaluation Results for RoTTA on Imagenet-C under Cyclic Domain Settings

Method	Subgroup	Cycle 1			Cycle 2		
		Domain	Error (%)	Avg	Domain	Error (%)	Avg
SloMo-Fast	Noise	gaussian	23.89	23.05	gaussian	19.96	20.76
		shot	18.89		shot	17.60	
		impulse	26.38		impulse	24.72	
	Blur	defocus	12.23	15.73	defocus	11.47	15.09
		glass	26.33		glass	24.83	
		motion	13.50		motion	13.34	
		zoom	10.86		zoom	10.71	
	Weather	snow	13.86	13.51	snow	13.39	13.17
		frost	13.42		frost	13.26	
		fog	13.26		fog	12.85	
	Digital	brightness	7.79	9.43	brightness	8.22	9.30
		contrast	11.08		contrast	10.38	
	Distortion	elastic	18.15	16.22	elastic	17.91	16.02
		pixel	13.21		pixel	13.04	
		jpeg	17.31		jpeg	17.10	
		Cycle 1 Avg: 15.59%			Cycle 2 Avg: 14.87%		

Table A.29. Detailed Evaluation Results for SloMo-Fast on CIFAR10-C under Cyclic Domain Settings

Method	Subgroup	Cycle 1			Cycle 2		
		Domain	Error (%)	Avg	Domain	Error (%)	Avg
SloMo-Fast	Noise	gaussian	35.59	33.71	gaussian	32.39	31.45
		shot	31.41		shot	30.76	
		impulse	34.14		impulse	31.19	
	Blur	defocus	24.68	27.64	defocus	24.63	27.13
		glass	33.52		glass	32.31	
		motion	26.84		motion	26.65	
		zoom	25.53		zoom	25.93	
	Weather	snow	28.23	29.39	snow	27.52	29.07
		frost	27.95		frost	27.65	
		fog	31.99		fog	32.04	
	Digital	brightness	23.03	24.01	brightness	23.47	24.20
		contrast	25.00		contrast	24.93	
	Distortion	elastic	29.61	29.62	elastic	29.94	29.84
		pixel	25.94		pixel	26.11	
		jpeg	33.30		jpeg	33.47	
		Cycle 1 Avg: 28.89%			Cycle 2 Avg: 28.34%		

Table A.30. Detailed Evaluation Results for SloMo-Fast on CIFAR100-C under Cyclic Domain Settings

Method	Subgroup	Cycle 1			Cycle 2		
		Domain	Error (%)	Avg	Domain	Error (%)	Avg
SloMo-Fast	Noise	gaussian	68.72	66.33	gaussian	64.82	63.58
		shot	65.56		shot	62.82	
		impulse	64.73		impulse	63.09	
	Blur	defocus	68.51	60.70	defocus	66.23	59.46
		glass	66.71		glass	65.77	
		motion	57.39		motion	56.28	
		zoom	50.19		zoom	49.57	
	Weather	snow	50.94	50.20	snow	49.47	49.30
		frost	56.59		frost	55.90	
		fog	43.08		fog	42.52	
	Digital	brightness	33.61	25.01	brightness	33.65	44.88
		contrast	56.98		contrast	56.12	
	Distortion	elastic	43.65	43.55	elastic	43.05	43.25
		pixel	41.21		pixel	40.93	
		jpeg	45.80		jpeg	45.77	
		Cycle 1 Avg: 53.22%			Cycle 2 Avg: 52.09%		

Table A.31. Detailed Evaluation Results for SloMo-Fast on Imagenet-C under Cyclic Domain Settings

Method	Subgroup	Cycle 1			Cycle 2		
		Domain	Error (%)	Avg	Domain	Error (%)	Avg
SloMo-Fast*	Noise	gaussian	21.65	20.62	gaussian	20.34	20.62
		shot	19.78		shot	21.12	
		impulse	20.43		impulse	20.40	
	Blur	defocus	15.03	14.79	defocus	14.34	15.21
		glass	14.65		glass	15.92	
		motion	17.07		motion	16.27	
		zoom	12.41		zoom	13.90	
	Weather	snow	13.72	13.21	snow	13.10	13.09
		frost	13.42		frost	13.23	
		fog	12.48		fog	12.58	
	Digital	brightness	9.33	10.24	brightness	9.17	10.23
		contrast	11.15		contrast	11.26	
	Distortion	elastic	15.85	13.96	elastic	15.64	14.02
		pixel	11.56		pixel	11.98	
		jpeg	14.46		jpeg	14.61	
		Cycle 1 Avg: 14.89%			Cycle 2 Avg: 14.38%		

Table A.32. Detailed Evaluation Results for SloMo-Fast* on CIFAR10-C under Cyclic Domain Settings

Method	Subgroup	Cycle 1			Cycle 2		
		Domain	Error (%)	Avg	Domain	Error (%)	Avg
SloMo-Fast*	Noise	gaussian	34.12	33.29	gaussian	34.82	33.51
		shot	32.54		shot	31.89	
		impulse	33.12		impulse	33.84	
	Blur	defocus	27.01	27.02	defocus	27.03	27.02
		glass	26.97		glass	27.04	
		motion	26.89		motion	27.12	
		zoom	27.22		zoom	26.89	
	Weather	snow	27.00	26.96	snow	26.98	26.96
		frost	26.90		frost	27.01	
		fog	26.98		fog	26.89	
	Digital	brightness	24.74	24.84	brightness	24.41	24.84
		contrast	25.05		contrast	25.18	
	Distortion	elastic	25.72	25.89	elastic	25.67	25.79
		pixel	24.79		pixel	24.97	
		jpeg	26.84		jpeg	26.73	
		Cycle 1 Avg: 27.98%			Cycle 2 Avg: 27.18%		

Table A.33. Detailed Evaluation Results for SloMo-Fast* on CIFAR100-C under Cyclic Domain Settings

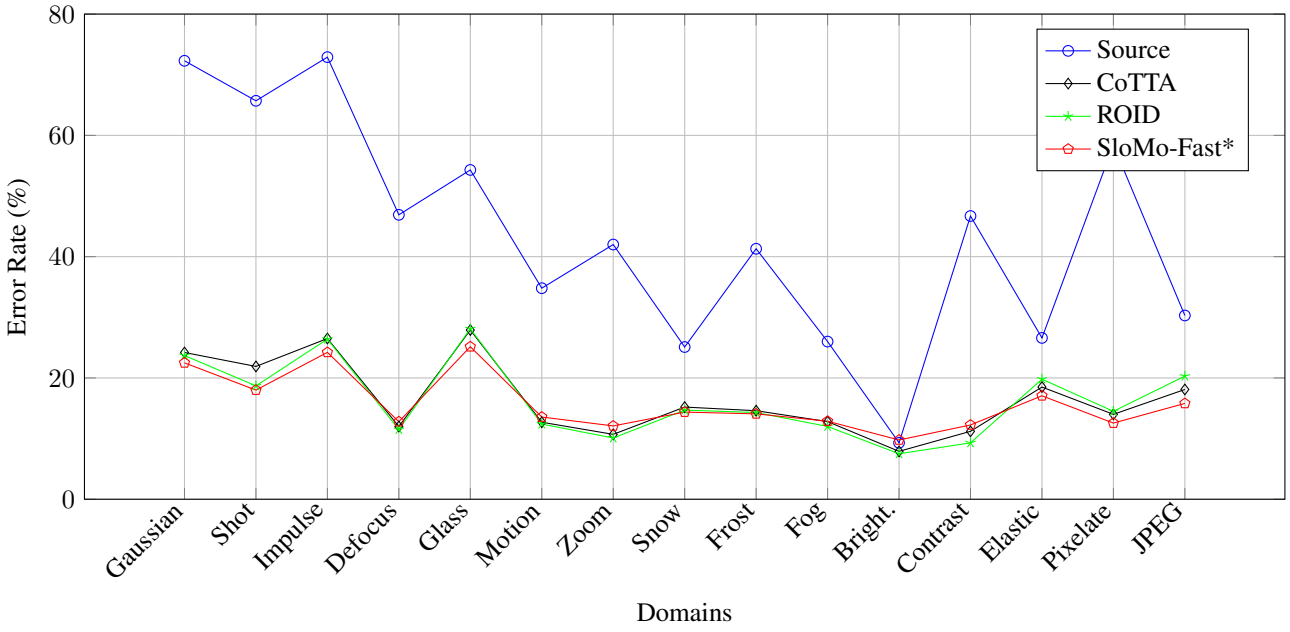

Figure A.2. CTTA Error rates (%) for Source (blue), CoTTA (black), ROID (green), and PA (red) across domains in the CIFAR10-C benchmark.

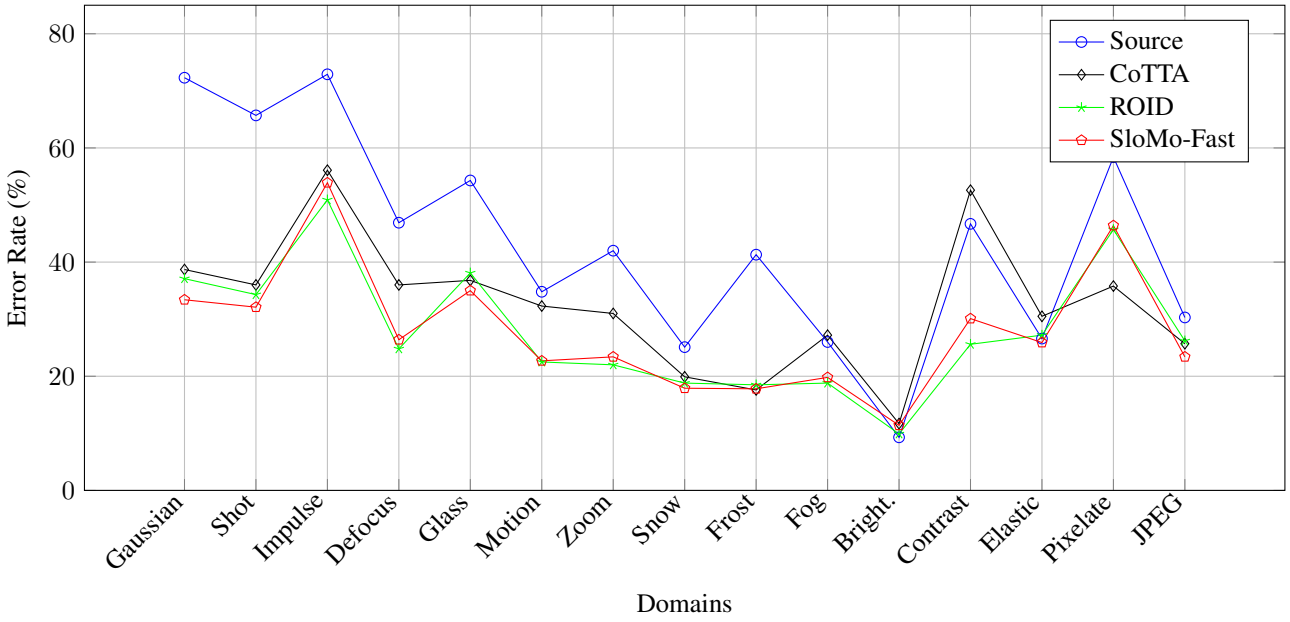

Figure A.3. Mixed TTA Error rates (%) for Source (blue), CoTTA (black), ROID (green), and PA (red) methods across domains in the CIFAR10-C benchmark for mixed domains.

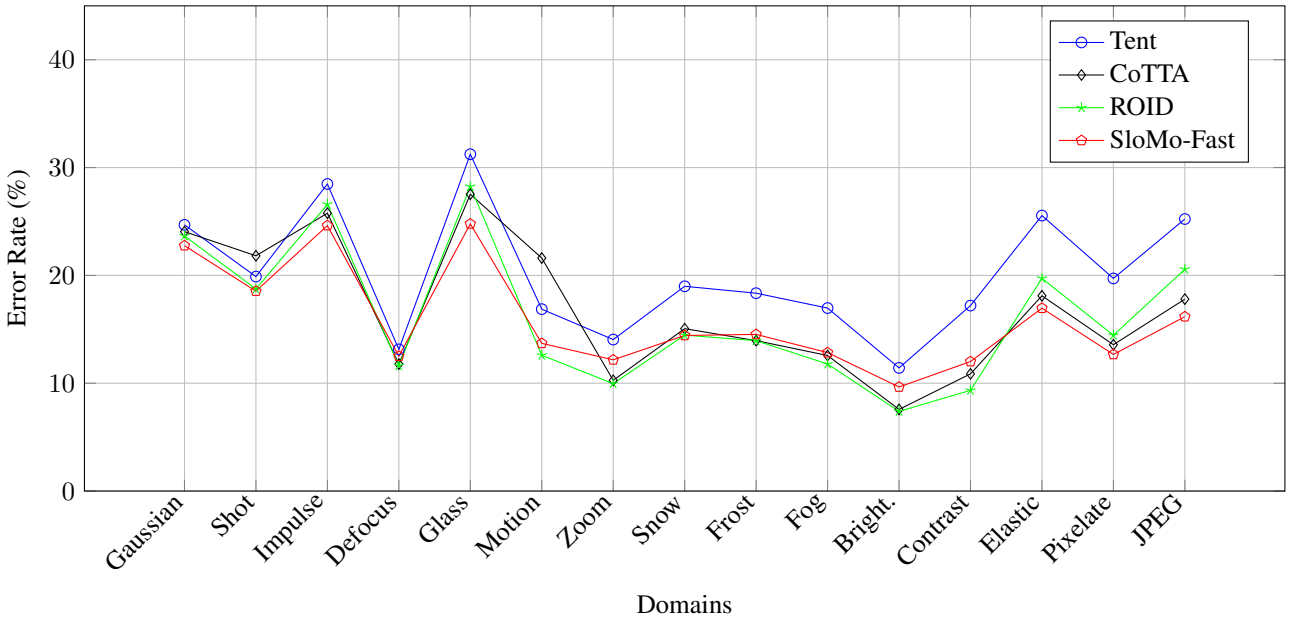

Figure A.4. Mixed after Continual TTA Error rates (%) for Tent (blue), CoTTA (black), ROID (green), and PA (red) methods across domains in the CIFAR10-C benchmark for mixed domains after continual learning.

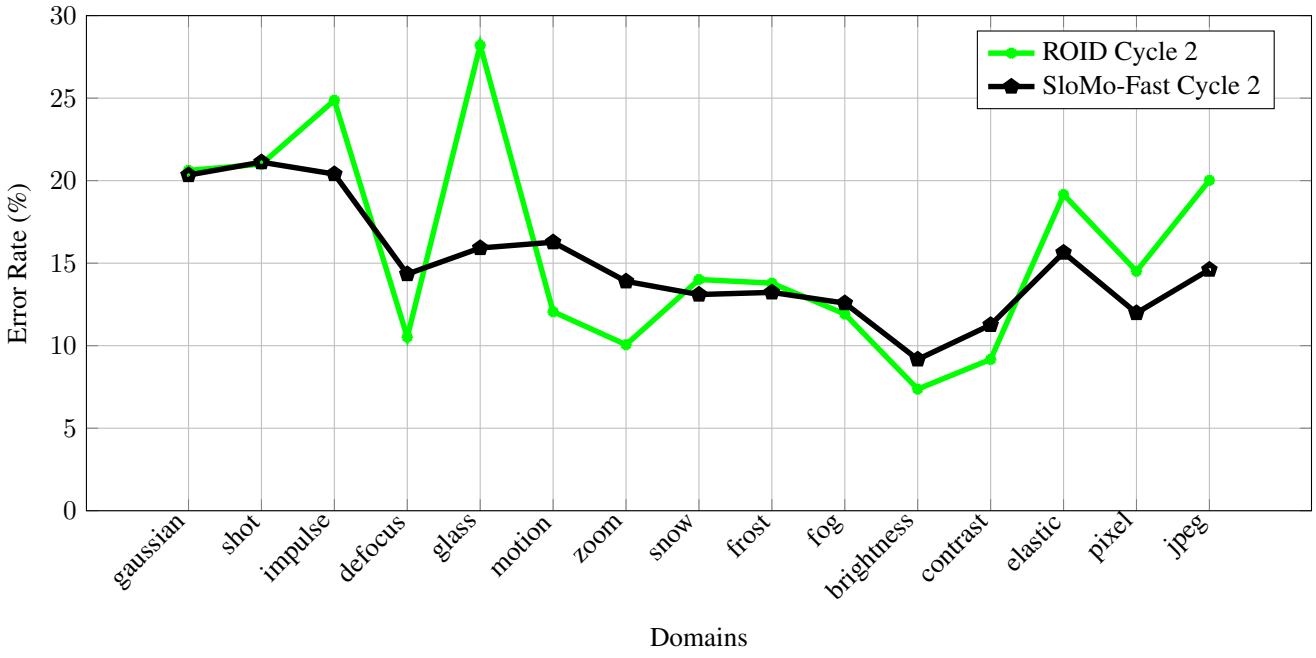

Figure A.5. Cyclic TTA Error rates (%) for ROID and PA methods across domains with subgroup boundaries (Cycle 2 only). Here, Existing best ROID is fluctuating and indicates catestrophic forgetting where SloMo-Fast is stable