DTS-META: DUAL-TIMESCALE META-LEARNING FOR CONTINUAL TEST-TIME ADAPTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Continual test-time adaptation (CTTA) adapts a source-pretrained model to an unlabeled, non-stationary target stream without source data or domain boundaries. Existing feature-subspace methods identify where to adapt, but coordinating persistent task-related adjustments with rapid environment-dependent corrections remains insufficiently explored. We observe that class-sensitive subspaces remain relatively stable across batches, whereas domain-sensitive subspaces exhibit greater temporal variation and environmental sensitivity. We therefore propose DTS-META, a dual-timescale feature adapter with slow class-sensitive and fast domain-sensitive pathways. The slow pathway learns persistent modulation along directions maintained through smoothed classifier-sensitivity statistics to preserve task-relevant feature structure. The fast pathway responds to the current environment via a more frequently refreshed prototype-residual covariance basis, with a MetaNet mapping projected batch statistics and teacher uncertainty to bounded scale and bias. We train the MetaNet through episodic meta-learning, encouraging updates learned from the current batch to remain effective across input variations. Across three classification benchmarks and continual semantic segmentation on CarlaTTA, DTS-META consistently outperforms existing baselines, demonstrating the benefits of timescale coordination in non-stationary adaptation.

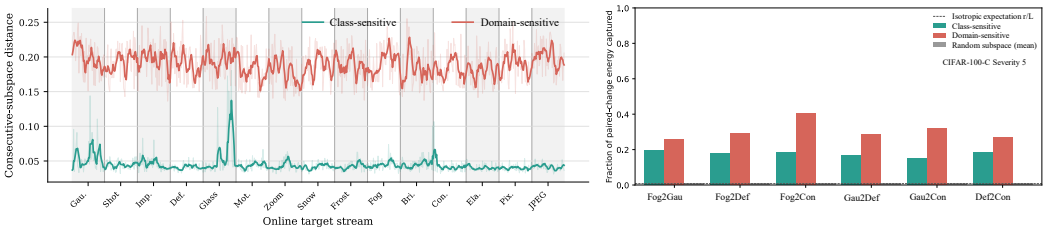
1 INTRODUCTION

At deployment, the target distribution may change continuously. In the source-free continual test-time adaptation (CTTA) setting considered here, a source-pretrained model adapts online to an unlabeled, non-stationary target stream without access to source data, target labels, or domain-boundary annotations (Sun et al., 2020; Wang et al., 2021; 2022; Yuan et al., 2023; Liang et al., 2025). Online adaptation must respond to current distribution shifts while limiting error accumulation and long-term forgetting. When the prediction task remains unchanged, the model needs to retain the feature structure used to distinguish categories, even as observation conditions such as noise, illumination, and weather change. This creates two adaptation demands: maintaining task-relevant structure over the stream and adjusting to the environment represented by the current batch, which raises the question of how to coordinate them during online adaptation.

Existing CTTA methods stabilize online adaptation using EMA teachers (Wang et al., 2022; Döbler et al., 2023), stochastic parameter restoration (Wang et al., 2022), reliability filtering (Niu et al., 2022; 2023), source anchoring (Chakrabarty et al., 2023), or online data buffering (Zhu et al., 2024). A parallel line of work designs adaptation modules using visual prompts (Gan et al., 2023), domain adapters (Liu et al., 2024), low-rank experts (Lee et al., 2024), or feature subspaces (Lai et al., 2026). However, coordinating persistent class-sensitive adjustments with rapid environment-dependent corrections at the feature level across distinct timescales remains insufficiently explored. Updates driven by batch-level distribution shifts may interfere with feature structures needed for classification, motivating separate adaptation mechanisms for components with different temporal dynamics.

To understand how these adaptation demands manifest in feature space, we examine two subspaces and observe distinct temporal dynamics and environmental sensitivities. Class-sensitive directions are estimated using AGOP, which captures sensitivity of classifier scores to feature changes (Beaglehole et al., 2024; Lai et al., 2026). We construct the domain-sensitive subspace from the covariance of prototype residuals, computed by subtracting the source class means indexed by teacher pseudo-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) Drift across the stream

(b) Sensitivity to environment shifts

Figure 1: **Temporal dynamics and environment sensitivity of two feature subspaces.** (a) Distance between consecutive subspace estimates along the target stream. The solid curves show a five-batch moving average of the per-batch distances; smaller distances indicate more stable estimates. (b) Paired environment-change energy captured by equal-rank subspaces (higher means greater sensitivity). Diagnostic protocols are detailed in Appendices B.2 and B.3.

labels from the current features. These residuals characterize current observations relative to class-specific references while suppressing the influence of varying batch class composition. Figure 1(a) shows that the class-sensitive subspace remains relatively stable across consecutive batches, whereas the domain-sensitive subspace exhibits larger changes. Furthermore, paired observations of the same images under different corruptions show that the domain-sensitive subspace captures more feature-change energy (Figure 1(b)). These observations indicate that class-sensitive and domain-sensitive subspaces respond differently to environmental changes and evolve at different rates, motivating adaptation at two timescales: persistent adjustments along class-sensitive directions and rapidly responsive corrections along domain-sensitive directions.

Motivated by these observations, we propose DTS-META, a dual-timescale feature adapter. A slow pathway accumulates smoothed AGOP statistics, refreshes its basis less frequently, and learns persistent modulation along class-sensitive directions. A fast pathway tracks prototype-residual covariance and refreshes its basis more frequently to respond to distribution shifts. The two pathways act sequentially on shared backbone features, combining continuity across the stream with batch responsiveness. The fast subspace specifies candidate correction directions, but their modulation must reflect the current batch. We therefore use a MetaNet to map projected batch residual statistics, teacher confidence, and entropy to bounded scale and bias. To learn a modulation rule that remains effective across views after an unsupervised update, we train the MetaNet through a first-order, label-free support–query objective (Finn et al., 2017; Nichol et al., 2018; Antoniou & Storkey, 2019). The current batch serves as the support view for a temporary MetaNet update, which is subsequently evaluated on a label-preserving augmented query view using the same detached teacher predictions. The outer objective combines both views to update the persistent parameters, while the first-order formulation avoids second-order differentiation through the inner update.

Our contributions are threefold:

- We propose a dual-timescale feature adapter for CTTA that coordinates persistent task-related adjustments with batch-dependent residual corrections through a slow class-sensitive pathway and a fast domain-sensitive pathway.
- We formulate a label-free support–query optimization scheme for the MetaNet, which evaluates temporary updates from the support view on an augmented query view and updates persistent parameters via first-order meta-gradients.
- We validate DTS-META across three classification benchmarks (CIFAR-10-C, CIFAR-100-C, ImageNet-C under standard, gradual, recurring, and long-horizon streams) and CarlaTTA semantic segmentation, demonstrating consistent improvements over baselines.

2 RELATED WORK

Continual test-time adaptation. CTTA methods address error accumulation and forgetting using EMA teachers with parameter restoration (Wang et al., 2022; Brahma & Rai, 2023), reliability-aware

filtering (Niu et al., 2022; 2023), source anchoring (Döbler et al., 2023; Chakrabarty et al., 2023), or memory buffers and prompt coresets (Zhu et al., 2024; Zhang et al., 2025). These mechanisms provide continuity across the stream while supporting adaptation to incoming observations. Our work examines how to coordinate these demands within a feature adapter, assigning persistent task-related adjustments and batch-dependent residual corrections to pathways with different statistics and update schedules.

Architecture-aware and subspace adaptation. Parameter-efficient CTTA reduces trainable state through visual prompts (Gan et al., 2023), domain adapters (Liu et al., 2024), or routed low-rank experts (Lee et al., 2024). In visual transfer learning, adaptation modules also reflect the backbone architecture: Visual Prompt Tuning introduces learnable prompt tokens into frozen vision Transformers (Jia et al., 2022), while AdaptFormer adds trainable bottleneck branches parallel to their feed-forward networks (Chen et al., 2022b). These designs link adaptation to backbone structure. At the feature level, GOLD uses AGOP statistics to track classifier-sensitive directions (Beaglehole et al., 2024; Lai et al., 2026). Our adapter is likewise informed by the model's decision structure and source feature references: a slow class-sensitive pathway maintains persistent modulation, and a fast domain-sensitive pathway generates batch-dependent corrections, coordinating adaptation directions and timescales.

Conditional parameter generation and episodic meta-learning. Hypernetworks generate model parameters from input descriptors (Ha et al., 2017), and input routing can select low-rank experts in CTTA (Lee et al., 2024). Gradient-based meta-learning optimizes nested support–query adaptation (Finn et al., 2017; Nichol et al., 2018), while AAL constructs unsupervised episodes using random labels and data augmentation (Antoniou & Storkey, 2019). In DTS-META, conditional generation and episodic optimization serve complementary roles. The MetaNet maps current batch statistics to bounded scale and bias in the fast subspace. A first-order support–query objective uses teacher predictions to evaluate support-derived MetaNet updates on an augmented query view, encouraging the modulation rule to remain effective across views.

3 PROBLEM SETUP AND PRELIMINARIES

3.1 CTTA SETUP AND TEACHER PREDICTIONS

Let $\mathcal{X} \subseteq \mathbb{R}^d$ denote the input space, $\mathcal{Y} = [C]$ the label space, and L the feature dimension, where $[n] = \{1, \dots, n\}$. Given a feature extractor $g_{\boldsymbol{\omega}} : \mathcal{X} \to \mathbb{R}^L$, the prediction model is defined as

$$f_{\boldsymbol{\theta}} = h_{\boldsymbol{\psi}} \circ g_{\boldsymbol{\omega}} : \mathcal{X} \to \mathbb{R}^C, \tag{1}$$

where the classifier $h_{\boldsymbol{\psi}} : \mathbb{R}^L \to \mathbb{R}^C$ outputs logits, with parameters $\boldsymbol{\theta} = (\boldsymbol{\omega}^\top, \boldsymbol{\psi}^\top)^\top$. Source pretraining on $\mathcal{D}_{\text{src}} = \{(\boldsymbol{x}_j^{\text{src}}, y_j^{\text{src}})\}_{j=1}^{N_{\text{src}}}$ yields the initial parameters $\boldsymbol{\theta}_0$.

During deployment, the model processes an unlabeled, non-stationary target stream without access to source examples, target labels, or domain-boundary annotations. At time t, the student receives a batch $\mathcal{B}_t = (\boldsymbol{x}_{t,i})_{i=1}^B$. Let $\boldsymbol{\theta}_t$ denote its parameters at the start of this batch. The extracted features are $\boldsymbol{f}_{t,i} = g_{\boldsymbol{\omega}_t}(\boldsymbol{x}_{t,i}) \in \mathbb{R}^L$, which we collect into $\mathbf{F}_t = [\boldsymbol{f}_{t,1}, \dots, \boldsymbol{f}_{t,B}]^\top \in \mathbb{R}^{B \times L}$.

To provide supervision on the unlabeled stream, an exponential moving average (EMA) teacher is maintained, initialized from the source model. Let $\bar{\boldsymbol{\theta}}_{t-1}$ denote the teacher parameters available before processing batch t. For each incoming sample, the teacher produces

$$\boldsymbol{q}_{t,i} = \mathrm{softmax}\Big(f_{\bar{\boldsymbol{\theta}}_{t-1}}(\boldsymbol{x}_{t,i})\Big) \in \mathbb{R}^C, \qquad \hat{y}_{t,i} = \arg\max_{c \in [C]} q_{t,i,c}, \tag{2}$$

where $\boldsymbol{q}_{t,i}$ is the teacher's predictive distribution and $\hat{y}_{t,i}$ is the corresponding hard pseudo-label. We stack the distributions into $\mathbf{Q}_t = [\boldsymbol{q}_{t,1}, \dots, \boldsymbol{q}_{t,B}]^\top \in \mathbb{R}^{B \times C}$. The probability distributions provide soft targets for student adaptation, while the pseudo-labels associate target features with the class-specific references introduced below.

3.2 CLASSIFIER SENSITIVITY AND CLASS PROTOTYPES

Class-sensitive directions. The average gradient outer product (AGOP) summarizes the sensitivity of classifier scores to feature changes(Beaglehole et al., 2024). Following its construction in

GOLD (Lai et al., 2026), consider n selected features $\{\boldsymbol{f}_i\}_{i=1}^n$ and their top-logit gradients:

$$\boldsymbol{g}_i = \nabla_{\boldsymbol{f}_i} \max_{c\in[C]} [h_{\boldsymbol{\psi}}(\boldsymbol{f}_i)]_c \in \mathbb{R}^L, \qquad \mathbf{G} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{g}_i \boldsymbol{g}_i^\top \in \mathbb{S}_+^L, \tag{3}$$

where $\mathbb{S}_+^L$ denotes the symmetric positive semidefinite $L \times L$ matrices. The dominant eigenvectors identify directions with large average squared sensitivity of the selected classifier scores. For an affine classifier $h_{\boldsymbol{\psi}}(\boldsymbol{f}) = \mathbf{W}\boldsymbol{f} + \boldsymbol{b}^{\text{cls}}$, $\mathbf{W} \in \mathbb{R}^{C\times L}$ and $\boldsymbol{b}^{\text{cls}} \in \mathbb{R}^C$; each gradient is a selected classifier weight row transposed. This connects AGOP to the model's decision structure. For any such statistic $\mathbf{M} \in \mathbb{S}_+^L$, we write $\mathbf{V} = \text{TopEig}_r(\mathbf{M}) \in \mathbb{R}^{L\times r}$, with $r \in [L]$ and $\mathbf{V}^\top\mathbf{V} = \mathbf{I}_r$; its columns are eigenvectors associated with the r largest eigenvalues. These columns span $\mathcal{S} = \text{span}(\mathbf{V})$; a feature has coordinates $\mathbf{V}^\top\boldsymbol{f} \in \mathbb{R}^r$, or $\mathbf{FV}$ in batch form. Here $\mathbf{I}_r$ is the $r \times r$ identity matrix.

Class-specific feature references. For each class $c \in [C]$, an offline source prototype is computed as

$$\boldsymbol{p}_c = \frac{1}{|\mathcal{I}_c|}\sum_{j\in\mathcal{I}_c} g_{\boldsymbol{\omega}_0}(\boldsymbol{x}_j^{\text{src}}) \in \mathbb{R}^L, \qquad \mathcal{I}_c = \{j \in [N_{\text{src}}] : y_j^{\text{src}} = c\}. \tag{4}$$

For $\mathcal{I}_c = \varnothing$, we set $\boldsymbol{p}_c = \mathbf{0}$. The matrix $\mathbf{P} = [\boldsymbol{p}_1, \ldots, \boldsymbol{p}_C]^\top \in \mathbb{R}^{C\times L}$ is retained and fixed during classification adaptation (Chakrabarty et al., 2023; Lai et al., 2026). The residual $\boldsymbol{f}_{t,i} - \boldsymbol{p}_{\hat{y}_{t,i}}$ measures deviation from a class-specific reference, which can reflect changing observation conditions. These two feature references underpin the adapter introduced next: classifier sensitivity determines the directions of persistent modulation, while prototype residuals characterize current observations for batch-conditioned correction.

4 DTS-META

4.1 DUAL-TIMESCALE OVERVIEW

Building on the feature references in Section 3, DTS-META combines persistent modulation with batch-conditioned correction. The slow pathway maintains a trainable scale along class-sensitive directions. The fast pathway uses prototype residuals and a MetaNet to generate feature corrections conditioned on each batch. We then train this generation rule through a support–query objective that evaluates a temporary MetaNet update on an augmented view of the same batch. As illustrated in Figure 2, the two pathways act sequentially on the shared student features $\mathbf{F}_t$. We apply feature extractors and classifiers row-wise to batches; the adapted logits $\mathbf{O}_t \in \mathbb{R}^{B\times C}$ stack the per-example vectors $\boldsymbol{o}_{t,i} \in \mathbb{R}^C$. The slow modulation is applied first, followed by the fast modulation:

$$\mathbf{F}_t \longrightarrow \mathbf{F}_t^{\text{slow}} \longrightarrow \mathbf{F}_t^{\text{adapt}} \longrightarrow \mathbf{O}_t = h_{\boldsymbol{\psi}_t}(\mathbf{F}_t^{\text{adapt}}). \tag{5}$$

The pathways separate the statistics and modulation mechanisms used for persistent and batch-dependent adjustments; their bases are not constrained to be orthogonal.

4.2 SLOW CLASS-SENSITIVE PATHWAY

The slow pathway provides persistent task-related modulation by accumulating classifier sensitivity across batches. The AGOP construction in Section 3.2 identifies directions that affect classifier scores, and a trainable scale retains adjustments along these directions throughout the stream. Let $\mathbf{V}_t^{\text{slow}} \in \mathbb{R}^{L\times r_s}$ be its orthonormal basis, with $r_s \in [L]$, and $\boldsymbol{s}_t^{\text{slow}} \in \mathbb{R}^{r_s}$ a persistent trainable scale. It transforms the student features as:

$$\mathbf{F}_t^{\text{slow}} = \mathbf{F}_t + \left[(\mathbf{F}_t\mathbf{V}_t^{\text{slow}}) \odot (\mathbf{1}_B(\boldsymbol{s}_t^{\text{slow}})^\top)\right](\mathbf{V}_t^{\text{slow}})^\top, \tag{6}$$

where $\mathbf{1}_B$ denotes the length-B all-ones vector and $\odot$ denotes element-wise multiplication. A zero scale yields identity modulation. Following GOLD, a confidence-filtered batch AGOP $\widehat{\mathbf{M}}_t^{\text{slow}}$ updates the long-term statistic:

$$\mathbf{M}_t^{\text{slow}} = (1-\alpha_s)\mathbf{M}_{t-1}^{\text{slow}} + \alpha_s\widehat{\mathbf{M}}_t^{\text{slow}}. \tag{7}$$

Here $\alpha_s \in (0, 1]$ is the EMA coefficient. Every T_s batches, we extract its leading r_s eigenvectors and use the resulting basis for subsequent batches. The scale $\boldsymbol{s}_t^{\text{slow}}$ persists across batches and is optimized at each online step through $\mathcal{L}_t^{\text{outer}}$. Thus, the slow pathway combines a gradually updated subspace with modulation parameters that retain learned adjustments across the stream.

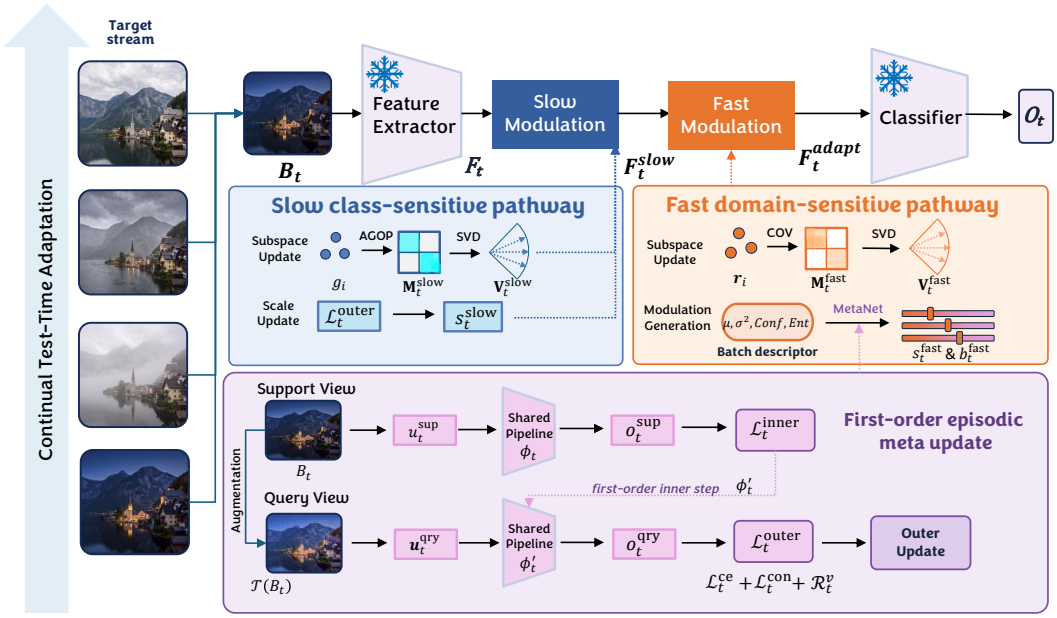


Figure 2: **Overview of DTS-META.** Top: student features pass through slow and fast modulation before classification. Middle: the slow pathway maintains an AGOP-based subspace and persistent scale; the fast pathway tracks prototype-residual covariance and maps batch statistics to scale and bias through the MetaNet. Bottom: a support-view inner step temporarily updates only the MetaNet, and the augmented query view evaluates the resulting modulation. Both views contribute to the outer update; the feature extractor and classifier remain fixed during the inner step.

4.3 FAST DOMAIN-SENSITIVE PATHWAY

The fast pathway applies a correction to the slow-modulated features, with coefficients conditioned on current observations. To identify candidate correction directions, we first characterize these observations relative to the source prototypes. Since raw feature covariance can reflect differences between class means, we subtract the prototype selected by each example's detached teacher pseudo-label. For $B \geq 2$, we estimate the covariance of the resulting residuals:

$$\boldsymbol{r}_{t,i} = \boldsymbol{f}_{t,i} - \boldsymbol{p}_{\hat{y}_{t,i}}, \qquad \mathbf{C}_t^{\text{fast}} = \frac{1}{B-1}\sum_{i=1}^{B}(\boldsymbol{r}_{t,i} - \bar{\boldsymbol{r}}_t)(\boldsymbol{r}_{t,i} - \bar{\boldsymbol{r}}_t)^\top \in \mathbb{S}_+^L, \tag{8}$$

where $\bar{\boldsymbol{r}}_t = B^{-1}\sum_{i=1}^{B}\boldsymbol{r}_{t,i}$ is the batch residual mean. Starting from $\mathbf{M}_0^{\text{fast}} = \mathbf{I}_L$, the fast statistic is updated every T_f batches, with $\alpha_f \in (0, 1]$:

$$\mathbf{M}_t^{\text{fast}} = (1-\alpha_f)\mathbf{M}_{t-1}^{\text{fast}} + \alpha_f \mathbf{C}_t^{\text{fast}}. \tag{9}$$

At each refresh, we eigendecompose the updated statistic and retain its leading r_f eigenvectors as the fast basis $\mathbf{V}_t^{\text{fast}} \in \mathbb{R}^{L\times r_f}$, with $r_f \in [L]$, for the current batch. The statistic and basis are retained between refreshes. In our configurations, the fast statistic uses a larger EMA coefficient ($\alpha_f > \alpha_s$) and a more frequent refresh ($T_f \ll T_s$), allowing it to respond rapidly to abrupt distribution changes.

The fast basis specifies candidate correction directions, while a batch descriptor determines how to modulate them. We estimate the basis from raw feature residuals and compute the descriptor from residuals after slow modulation, so that the generated correction responds to the feature state following the persistent adjustment. Let $\boldsymbol{\mu}_t, \boldsymbol{v}_t \in \mathbb{R}^{r_f}$ be the coordinate-wise mean and population variance of $\boldsymbol{a}_{t,i} = (\mathbf{V}_t^{\text{fast}})^\top(\boldsymbol{f}_{t,i}^{\text{slow}} - \boldsymbol{p}_{\hat{y}_{t,i}}) \in \mathbb{R}^{r_f}$. The detached batch descriptor is:

$$\boldsymbol{u}_t = \left[\hat{\boldsymbol{\mu}}_t;\ \hat{\boldsymbol{v}}_t;\ \bar{c}_t;\ \hat{H}_t\right] \in \mathbb{R}^{2r_f+2}, \tag{10}$$

where $\hat{\boldsymbol{\mu}}_t$ and $\hat{\boldsymbol{v}}_t$ denote the normalized residual mean and log-transformed variance, respectively; $\bar{c}_t$ is the mean teacher confidence, and $\hat{H}_t$ is the normalized mean teacher entropy. The exact

transformations are defined in Appendix A.2. The residual moments summarize displacement and dispersion along the fast directions, while the teacher statistics characterize prediction uncertainty. To translate these batch-level statistics into adaptive feature corrections, we use a MetaNet $\mathcal{H}_{\boldsymbol{\phi}} : \mathbb{R}^{2r_f+2} \to \mathbb{R}^{2r_f}$ with parameters $\boldsymbol{\phi} \in \mathbb{R}^{d_\phi}$. It maps $\boldsymbol{u}_t$ to raw scale and bias vectors in the fast subspace:

$$(\tilde{\boldsymbol{s}}_t^{\text{fast}}, \tilde{\boldsymbol{b}}_t^{\text{fast}}) = \mathcal{H}_{\boldsymbol{\phi}}(\boldsymbol{u}_t). \tag{11}$$

We bound the generated scale and bias using $\boldsymbol{s}_t^{\text{fast}} = \gamma_s \tanh(\tilde{\boldsymbol{s}}_t^{\text{fast}})$ and $\boldsymbol{b}_t^{\text{fast}} = \gamma_b \tanh(\tilde{\boldsymbol{b}}_t^{\text{fast}})$, where $\gamma_s, \gamma_b > 0$ and $\tanh$ acts element-wise. We then apply this batch-conditioned modulation to the slow-adapted features:

$$\mathbf{F}_t^{\text{adapt}} = \mathbf{F}_t^{\text{slow}} + \left[(\mathbf{F}_t^{\text{slow}} \mathbf{V}_t^{\text{fast}}) \odot (\mathbf{1}_B(\boldsymbol{s}_t^{\text{fast}})^\top) + \mathbf{1}_B(\boldsymbol{b}_t^{\text{fast}})^\top\right] (\mathbf{V}_t^{\text{fast}})^\top. \tag{12}$$

4.4 First-Order Episodic Online Optimization

Batch-conditioned generation makes fast modulation responsive to current observations. To train the modulation rule to remain effective after adapting to those observations, we evaluate a support-derived MetaNet update on a label-preserving query view. Each classification batch forms a label-free support–query episode (Antoniou & Storkey, 2019) with $\mathcal{B}_t^{\text{sup}} = \mathcal{B}_t$ and $\mathcal{B}_t^{\text{qry}} = \mathcal{T}(\mathcal{B}_t)$, where $\mathcal{T}$ is a stochastic label-preserving transformation. We compute descriptors $\boldsymbol{u}_t^{\text{sup}}$ and $\boldsymbol{u}_t^{\text{qry}}$ from the support and query features, respectively, using Eq. (10). The support modulation is generated by $\mathcal{H}_{\boldsymbol{\phi}_t}(\boldsymbol{u}_t^{\text{sup}})$, whereas the query modulation is generated by $\mathcal{H}_{\boldsymbol{\phi}'_t}(\boldsymbol{u}_t^{\text{qry}})$ using temporary parameters adapted on the support view. The two views preserve sample correspondence and are matched to the same detached teacher logits $\mathbf{O}_t^{\text{ema}} = f_{\bar{\boldsymbol{\theta}}_{t-1}}(\mathcal{B}_t)$ computed from the original batch. This keeps the target fixed during the inner update and preserves the correspondence between each query sample and its original view. Let ℓ_{sce} denote symmetric soft-distribution consistency and $\mathcal{R}_t^v$ the raw-output regularizer for $v \in \{\text{sup}, \text{qry}\}$; their definitions are given in Appendix A.2.

Inner update on the support view. Using the persistent MetaNet parameters $\boldsymbol{\phi}_t$, we formulate the support objective as

$$\mathcal{L}_t^{\text{inner}} = \frac{1}{B} \sum_{i=1}^{B} \ell_{\text{sce}}\left(\boldsymbol{o}_{t,i}^{\text{sup}}, \boldsymbol{o}_{t,i}^{\text{ema}}\right) + \lambda_{\text{reg}} \mathcal{R}_t^{\text{sup}}. \tag{13}$$

We obtain temporary parameters $\boldsymbol{\phi}'_t$ by taking one gradient-descent step from $\boldsymbol{\phi}_t$ on this objective with step size η_{in}. Only the MetaNet parameters are temporarily updated in the inner step. When differentiating the query objective, we treat the inner gradient as constant, omitting second-order derivatives while retaining the gradient path to $\boldsymbol{\phi}_t$ for computational efficiency (Nichol et al., 2018). The inner loss is used only to construct $\boldsymbol{\phi}'_t$ and is not added separately to the outer objective.

Outer objective on the support and query views. We use a joint teacher-consistency loss over the support and query predictions, obtained with $\boldsymbol{\phi}_t$ and $\boldsymbol{\phi}'_t$, respectively:

$$\mathcal{L}_t^{\text{ce}} = \frac{1}{2B} \sum_{i=1}^{B} \left[\ell_{\text{sce}}\left(\boldsymbol{o}_{t,i}^{\text{sup}}, \boldsymbol{o}_{t,i}^{\text{ema}}\right) + \ell_{\text{sce}}\left(\boldsymbol{o}_{t,i}^{\text{qry}}, \boldsymbol{o}_{t,i}^{\text{ema}}\right)\right]. \tag{14}$$

To anchor adapted features to class-specific references, we use a prototype-guided contrastive loss (Döbler et al., 2023; Chakrabarty et al., 2023; Lai et al., 2026). Each sample forms a triplet containing its adapted support and query features, along with the source prototype most similar to its support feature:

$$\mathcal{L}_t^{\text{con}} = -\frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} \frac{1}{|\mathcal{P}(a)|} \sum_{p \in \mathcal{P}(a)} \log \frac{\exp(\boldsymbol{z}_a^\top \boldsymbol{z}_p / \kappa)}{\sum_{n \in \mathcal{A} \setminus \{a\}} \exp(\boldsymbol{z}_a^\top \boldsymbol{z}_n / \kappa)}, \tag{15}$$

where $\boldsymbol{z}_a$ is a normalized projection, $\mathcal{A}$ indexes all triplet entries in the batch, $\mathcal{P}(a)$ contains the other two entries of the same triplet, and κ is the temperature. Projection and indexing details are given in Appendix A.2. Together with teacher consistency and modulation regularization, this gives the outer objective

$$\mathcal{L}_t^{\text{outer}} = \lambda_{\text{ce}} \mathcal{L}_t^{\text{ce}} + \lambda_{\text{con}} \mathcal{L}_t^{\text{con}} + \frac{\lambda_{\text{reg}}}{2} \left(\mathcal{R}_t^{\text{sup}} + \mathcal{R}_t^{\text{qry}}\right). \tag{16}$$

Both views contribute to the outer objective and update the persistent parameters. Through the first-order gradient path to ϕ_t, the query term supplies feedback on the effectiveness of the temporary support update. The regularizer discourages large raw MetaNet outputs, complementing the explicit bounds imposed by $\tanh$.

Let ϑ_t collect the slow scale s_t^{slow}, a small set of batch-normalization affine parameters, the MetaNet parameters ϕ_t, and projection-head parameters ξ_t. For the current batch, we use the support prediction generated with persistent MetaNet parameters ϕ_t and combine it with the teacher logits:

$$\mathbf{O}_t^{\text{final}} = \mathbf{O}_t^{\text{sup}} + \mathbf{O}_t^{\text{ema}}. \tag{17}$$

We then update ϑ_t using the gradient of $\mathcal{L}_t^{\text{outer}}$, followed by the EMA teacher update.

5 Experiments

5.1 Experimental Setup

We evaluate image classification on CIFAR-10-C, CIFAR-100-C, and ImageNet-C (Krizhevsky & Hinton, 2009; Deng et al., 2009; Hendrycks & Dietterich, 2019). The primary stream contains all 15 corruption types at severity 5, traversed sequentially in a fixed order without inter-corruption resets or domain-boundary notifications. We additionally consider recurring domains under the Continual Dynamic Change (CDC) protocol introduced by DPCore (Zhang et al., 2025), gradual severity changes on CIFAR-100-C, and ten consecutive passes over the CIFAR-100-C stream without reset. We report online mean classification error across all stream batches (lower is better).

We also evaluate continual semantic segmentation on CarlaTTA (Marsden et al., 2024), a synthetic driving benchmark built with the CARLA simulator (Dosovitskiy et al., 2017). CarlaTTA contains five temporally ordered sequences with changes in illumination, weather, scene content, and label distribution. Each frame is processed once in temporal order. We report sequence-level mIoU over 14 semantic classes, with higher values indicating better performance.

For classification, we use the same source architectures and checkpoints as GOLD (Lai et al., 2026): WideResNet-28 (Zagoruyko & Komodakis, 2016), ResNeXt-29 (Xie et al., 2017), and ResNet-50 (He et al., 2016). We compare DTS-Meta with TENT (Wang et al., 2021), AdaContrast (Chen et al., 2022a), CoTTA (Wang et al., 2022), RMT (Döbler et al., 2023), SAR (Niu et al., 2023), EATA (Niu et al., 2022), SANTA (Chakrabarty et al., 2023), OBAO (Zhu et al., 2024), and GOLD (Lai et al., 2026). For segmentation, we compare MEMO (Zhang et al., 2022), TENT, CoTTA, and GOLD using a common DeepLabV2–ResNet-101 checkpoint. All methods receive the same ordered unlabeled stream and are evaluated without domain-boundary information or test-label-based model selection. Dataset construction, stream schedules, and implementation details are provided in Appendices A.4–A.8. Additional investigations covering batch-size sensitivity A.5, hyperparameter sweeps, wall-clock efficiency, and qualitative visualizations are reported in Appendix B.

5.2 Main Classification Results

Table 1 reports the standard CTTA results. DTS-Meta records the lowest online mean error on all three benchmarks: 14.45% on CIFAR-10-C, 27.87% on CIFAR-100-C, and 58.75% on ImageNet-C. Relative to GOLD, these results correspond to absolute reductions of 0.59, 0.72, and 0.65 percentage points, respectively. The improvement is distributed across corruption types: DTS-Meta obtains the lowest error on 13 of 15 CIFAR-10-C corruptions and 11 of 15 CIFAR-100-C corruptions, while achieving the lowest overall error rate on ImageNet-C. The gains across diverse corruption types suggest that the improvement is not limited to a small subset of shifts.

5.3 Continual Dynamic Change

CDC uses Dirichlet partitioning with $\delta = 1$ to generate recurring domains with irregular transition times. Table 2 shows that DTS-Meta achieves the lowest error on all three benchmarks, including a reduction from 28.61% for GOLD to 27.65% on CIFAR-100-C. These results highlight the advantage of DTS-Meta under frequent and irregular domain shifts, supporting its effectiveness in rapidly changing environments.

Table 1: Online classification error rates (%, lower is better) under the standard CTTA protocol. All methods are evaluated online at severity 5. For DTS-META and GOLD, the Mean column reports the mean ± std across five independent seeds. **Bold** indicates the best result, and underline indicates the second best.

	Method	Gau.	Shot	Imp.	Def.	Glass	Mot.	Zoom	Snow	Frost	Fog	Bri.	Con.	Ela.	Pix.	JPEG	Mean
CIFAR-10-C	TENT	24.67	19.90	28.50	13.09	30.65	16.04	14.71	20.55	20.38	18.99	13.05	16.14	23.06	18.62	24.12	20.16
	Ada	29.24	22.68	30.28	13.81	32.79	14.28	11.81	16.46	15.08	14.12	8.07	10.00	21.64	17.71	19.83	18.52
	CoTTA	24.09	21.86	25.90	11.74	27.67	12.71	10.44	15.06	13.94	12.60	**7.61**	10.70	18.40	13.67	17.89	16.29
	RMT	23.78	20.08	25.06	13.10	25.10	14.54	12.46	15.91	15.50	14.51	10.65	14.28	17.47	13.65	16.45	16.84
	SAR	28.30	25.97	35.75	12.69	34.79	13.87	11.98	17.48	17.59	14.85	8.15	12.97	23.53	19.52	27.20	20.30
	EATA	24.28	19.13	27.57	12.38	29.15	14.12	11.81	16.51	14.87	14.86	9.28	12.90	20.69	15.73	20.98	17.62
	SANTA	23.81	19.73	27.86	11.62	27.87	12.50	**10.01**	14.03	13.87	12.08	7.76	10.92	19.19	13.50	19.57	16.29
	OBAO	23.55	19.11	25.02	10.98	24.70	12.64	10.79	14.10	13.56	12.21	8.63	10.76	16.23	12.27	15.47	15.33
	GOLD	22.38	19.10	24.87	11.32	24.44	12.55	10.74	13.53	13.42	11.55	8.65	10.54	15.85	11.86	14.95	15.04$_{\pm 0.10}$
	DTS-META	**20.57**	**17.67**	**24.00**	**10.59**	**23.87**	**12.02**	10.14	**13.25**	**13.17**	**11.26**	8.28	**9.89**	**15.60**	**11.57**	**14.89**	**14.45**$_{\pm 0.09}$
CIFAR-100-C	TENT	37.32	35.56	41.91	38.33	50.56	46.33	46.96	56.60	62.43	71.67	72.81	83.38	88.85	90.96	94.08	61.18
	Ada	42.65	36.89	38.52	27.72	40.36	29.65	27.64	32.95	30.71	38.17	26.27	28.41	34.70	33.56	36.24	33.63
	CoTTA	40.93	38.13	39.69	28.33	37.52	29.91	28.24	34.62	33.80	41.01	27.85	30.79	35.14	31.49	36.40	34.26
	RMT	40.22	36.42	36.79	28.23	33.93	28.43	26.31	28.74	28.61	30.63	25.56	27.20	27.95	26.60	28.93	30.30
	SAR	40.45	34.81	37.10	25.68	37.29	28.00	25.50	31.85	30.86	35.80	25.12	27.81	31.80	29.00	37.16	31.88
	EATA	37.04	33.04	36.72	27.70	37.71	29.73	26.63	32.69	32.01	35.84	26.59	28.99	33.40	29.55	37.45	32.34
	SANTA	**36.33**	**32.78**	**35.18**	25.87	35.49	27.86	24.90	29.49	29.69	32.63	23.61	26.38	31.35	27.68	35.47	30.31
	OBAO	38.28	34.99	36.43	26.08	33.39	27.05	25.10	28.16	28.51	30.67	23.72	25.50	27.12	25.28	29.13	29.29
	GOLD	37.03	34.02	35.36	25.98	32.24	26.70	24.88	27.28	27.08	29.60	23.67	25.25	26.77	24.73	**28.37**	28.59$_{\pm 0.07}$
	DTS-META	36.45	33.20	35.62	**24.63**	**31.71**	**25.93**	**23.36**	**25.96**	**26.62**	**28.89**	**22.75**	**23.78**	**26.51**	**24.00**	28.64	**27.87**$_{\pm 0.06}$
ImageNet-C	TENT	81.44	74.78	72.90	77.26	74.14	66.36	55.66	61.62	63.08	51.32	38.06	71.92	51.18	47.52	53.30	62.70
	Ada	83.06	81.00	79.26	82.68	78.12	72.06	62.82	62.78	64.10	52.70	38.44	66.32	54.32	49.42	52.90	65.33
	CoTTA	84.68	81.62	80.44	81.68	79.54	68.38	57.92	60.38	60.80	48.66	**36.46**	66.52	46.92	**41.42**	45.60	62.73
	RMT	80.32	76.32	74.59	77.02	74.33	66.10	57.82	56.95	59.25	48.26	39.05	60.75	47.20	42.84	**43.62**	60.29
	SAR	82.26	74.18	71.64	77.54	73.64	66.08	56.36	61.68	62.98	51.60	37.50	69.66	49.90	45.80	51.28	62.14
	EATA	75.76	**66.30**	**65.52**	74.54	**69.36**	64.32	57.20	63.14	62.84	53.42	40.90	64.82	50.68	47.12	50.28	60.41
	SANTA	**74.02**	72.44	72.78	74.96	74.10	64.94	54.52	55.84	63.74	46.78	37.26	69.16	51.38	44.90	48.46	60.35
	OBAO	79.86	75.78	74.48	75.72	74.14	65.24	56.64	55.86	**58.24**	48.10	39.38	**60.10**	**46.42**	42.08	44.36	59.76
	GOLD	77.50	72.32	71.50	**73.34**	71.64	62.58	54.86	57.08	60.66	48.46	37.84	62.26	47.78	45.06	48.10	59.40$_{\pm 0.17}$
	DTS-META	75.60	72.48	72.28	73.90	72.78	**62.30**	**53.64**	**55.60**	59.32	**46.00**	36.54	61.48	47.62	44.34	47.48	**58.75**$_{\pm 0.11}$

Table 2: Classification error (%, lower is better) under CDC with $\delta = 1$. Detailed results are in Appendix A.7.

Method	CIFAR-10-C	CIFAR-100-C	ImageNet-C
TENT	19.08	54.01	61.89
Ada	19.75	34.47	64.40
CoTTA	16.47	32.74	59.60
RMT	16.34	30.34	58.73
SAR	20.46	31.85	61.70
EATA	17.80	32.97	59.66
SANTA	16.42	30.83	60.37
OBAO	15.07	29.33	58.90
GOLD	14.99$_{\pm 0.14}$	28.61$_{\pm 0.16}$	58.86$_{\pm 0.13}$
DTS-META	**14.29**$_{\pm 0.10}$	**27.65**$_{\pm 0.13}$	**58.28**$_{\pm 0.15}$

Table 3: Component ablation of DTS-META under standard CTTA. Error (%, lower is better).

Variant	C10-C	C100-C	IN-C
w/o Adapter	15.18	28.73	59.65
DTS-META-Slow	14.98	28.47	59.32
Dual-Shared	14.88	28.26	59.14
DTS-META-Forward	14.72	28.08	58.98
DTS-META-Full	**14.45**	**27.87**	**58.75**

w/o Adapter removes both modulation pathways while retaining the common online adaptation framework; Slow retains only the slow pathway; Dual-Shared adds a fast pathway with stream-shared scale and bias; Forward generates fast parameters from each batch; Full adds support–query episodic optimization.

5.4 GRADUAL CORRUPTION SHIFTS

On the gradual CIFAR-100-C stream (Appendix Table 9), DTS-META reduces mean error from 25.60% for GOLD to 24.84%, an absolute improvement of 0.76 percentage points. The severity-5 error also decreases by 0.60 percentage points, indicating that the gain is retained at peak corruption severity.

5.5 LONG-HORIZON ADAPTATION

Table 4 shows that, over ten consecutive CIFAR-100-C rounds without reset, DTS-META obtains the lowest error in every round. Its ten-round mean error is 24.55%, compared with 25.33% for GOLD, and the gap remains visible in the final round. The improvement is therefore not lost as the corruption stream is repeated.

Table 4: Classification error (%, lower is better) over ten consecutive rounds of CIFAR-100-C without model reset. **Bold** and underline denote the best and second-best results in each column.

Method	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	Mean
TENT	60.08	96.72	97.78	98.10	97.97	98.00	97.92	98.41	98.95	99.00	94.29
Ada	33.63	33.88	35.13	36.39	37.71	39.03	40.26	41.39	42.41	43.38	38.32
CoTTA	32.76	31.20	31.20	31.48	31.97	32.49	32.99	33.45	33.96	34.38	32.59
RMT	30.32	27.40	26.77	26.51	26.44	26.37	26.35	26.32	26.32	26.31	26.91
SAR	31.90	32.70	34.56	36.66	38.53	39.98	41.72	43.96	46.79	50.38	39.72
EATA	32.34	34.12	35.00	35.92	36.65	37.04	37.66	37.96	38.57	39.28	36.45
SANTA	30.31	30.31	30.51	30.57	30.65	30.70	30.71	30.73	30.85	30.73	30.61
OBAO	29.36	26.57	25.70	25.25	25.05	24.87	24.76	24.71	24.71	24.71	25.57
GOLD	28.57	26.00	25.34	25.03	24.86	24.78	24.72	24.71	24.65	24.67	25.33
DTS-META	**28.01**	**25.55**	**24.62**	**24.21**	**23.98**	**23.89**	**23.84**	**23.80**	**23.79**	**23.77**	**24.55**

5.6 SEMANTIC SEGMENTATION

Table 5 shows that DTS-META obtains the highest average mIoU, 55.3%, compared with 54.3% for GOLD. It achieves the best or tied-best result on four of the five CarlaTTA sequences. The largest gain occurs under dynamic weather, where mIoU increases from 48.9% to 52.9%. These results demonstrate that our dual-timescale adaptation framework and residual modulation readily generalize to benefit dense prediction tasks. Qualitative results are provided in Appendix Figure 5.

Table 5: Continual semantic segmentation on the five CarlaTTA sequences. Results are mIoU (%, higher is better) over each complete sequence.

Method	Day2night	Clear2fog	Clear2rain	Dynamic	Highway	Average
MEMO	61.0	55.1	71.6	36.7	23.6	49.6
TENT	61.5	56.0	70.9	50.8	32.8	54.4
CoTTA	61.4	56.8	70.7	46.3	**33.8**	53.8
GOLD	61.6	**57.0**	70.9	48.9	33.2	54.3
DTS-META	**62.0**	**57.0**	**71.7**	**52.9**	32.9	**55.3**

5.7 ABLATION STUDY

Table 3 includes a w/o Adapter baseline and four internal variants of DTS-META. The baseline removes both modulation pathways while retaining the common online adaptation framework. Adding the slow pathway reduces error across all three benchmarks. Starting from the slow-only control, Dual-Shared adds a domain-sensitive fast pathway with stream-shared scale and bias. This reduces error on all three benchmarks, supporting its complementary role. DTS-META-Forward replaces the stream-shared fast parameters with batch-conditioned scale and bias generated by the MetaNet. The resulting error reduction supports the effectiveness of this design.

DTS-META-Full adds episodic meta-learning while retaining the same MetaNet architecture as Forward. It achieves the lowest error on all three benchmarks, suggesting that episodic meta-learning improves the model's adaptability to changing target distributions. These comparisons examine the fast pathway, conditional generation, and episodic optimization within a common adaptation framework. Exact configurations are provided in Appendix B.7. The direct timescale-control experiment in Appendix B.1 keeps the complete model and training setup fixed while varying only the two refresh periods, providing a focused test of timescale coordination.

6 CONCLUSION

We presented DTS-META, a dual-timescale CTTA approach combining persistent class-sensitive modulation with batch-conditioned domain-sensitive corrections. A MetaNet generates fast modulation parameters, while episodic meta-learning improves the learned modulation rule. Experiments across multiple classification stream protocols and a semantic segmentation extension demonstrate the effectiveness of the approach. These results highlight the value of coordinating adaptation directions and update timescales in non-stationary environments.

AI USE STATEMENT

In this work, we used generative AI tools for obtaining feedback and suggestions to improve manuscript clarity and writing. We have not used generative AI tools for scientific ideation, theoretical derivations, or data analysis, and other required disclosure tasks are not applicable. Additionally, we used generative AI tools for text polishing and grammar correction. We have reviewed all AI-assisted work; the authors critically evaluated all suggestions, manually refined the text, and verified technical accuracy. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

While Section 4 outlines the core model components and optimization objectives, Appendix A provides comprehensive specifications for reproduction, including source prototype computation, subspace tracking, batch descriptors, the complete online algorithm, and hyperparameter configurations. Appendix A also details the datasets, pretrained backbones, stream construction protocols, and evaluation metrics for both classification and semantic segmentation. Additional diagnostic experiments, hyperparameter sweeps, and ablation configurations are documented in Appendix B.

REFERENCES

Antreas Antoniou and Amos Storkey. Assume, augment and learn: Unsupervised few-shot meta-learning via random labels and data augmentation. *arXiv preprint arXiv:1902.09884*, 2019.

Daniel Beaglehole, Peter Súkeník, Marco Mondelli, and Mikhail Belkin. Average gradient outer product as a mechanism for deep neural collapse. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Dhanajit Brahma and Piyush Rai. A probabilistic framework for lifelong test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3582–3591, 2023.

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.

Dian Chen, Dequan Wang, Trevor Darrell, and Sayna Ebrahimi. Contrastive test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022a.

Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4): 834–848, 2018.

Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo. AdaptFormer: Adapting vision transformers for scalable visual recognition. In *Advances in Neural Information Processing Systems*, volume 35, 2022b.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

Mario Döbler, Robert A. Marsden, and Bin Yang. Robust mean teacher for continual and gradual test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. CARLA: An open urban driving simulator. In *Conference on Robot Learning*, volume 78 of *Proceedings of Machine Learning Research*, pp. 1–16, 2017.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1126–1135, 2017.

Yulu Gan, Yan Bai, Yihang Lou, Xianzheng Ma, Renrui Zhang, Nian Shi, and Lin Luo. Decorate the newcomers: Visual domain prompt for continual test time adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023.

David Ha, Andrew M. Dai, and Quoc V. Le. Hypernetworks. In *International Conference on Learning Representations*, 2017.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019.

Dan Hendrycks, Norman Mu, Ekin D. Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. AugMix: A simple data processing method to improve robustness and uncertainty. In *International Conference on Learning Representations*, 2020.

Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European Conference on Computer Vision*, 2022.

Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Guannan Lai, Da-Wei Zhou, Zhenguo Li, and Han-Jia Ye. The golden subspace: Where efficiency meets generalization in continual test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Daeun Lee, Jaehong Yoon, and Sung Ju Hwang. BECoTTA: Input-dependent online blending of experts for continual test-time adaptation. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 27072–27093, 2024.

Jian Liang, Ran He, and Tieniu Tan. A comprehensive survey on test-time adaptation under distribution shifts. *International Journal of Computer Vision*, 2025.

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

Robert A. Marsden, Mario Döbler, and Bin Yang. Introducing intermediate domains for effective self-training during test-time. In *International Joint Conference on Neural Networks*, 2024.

Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*, 2018.

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*, volume 162 of *Proceedings of Machine Learning Research*, 2022.

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

Ori Press, Steffen Schneider, Matthias Kümmerer, and Matthias Bethge. RDumb: A simple approach that questions our progress in continual test-time adaptation. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

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

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9229–9248, 2020.

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

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

Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

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

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *British Machine Vision Conference*, 2016.

Marvin Zhang, Sergey Levine, and Chelsea Finn. MEMO: Test time robustness via adaptation and augmentation. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Yunbei Zhang, Akshay Mehra, Shuaicheng Niu, and Jihun Hamm. DPCore: Dynamic prompt coreset for continual test-time adaptation. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 75757–75778, 2025.

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*, 2024.

APPENDIX

A REPRODUCIBILITY DETAILS

In the classification procedure, teacher predictions, pseudo-labels, prototype references, subspace statistics and bases, and batch descriptors are treated as constants when differentiating the adaptation losses.

A.1 SOURCE PROTOTYPES AND SLOW-PATHWAY DETAILS

The online teacher is initialized from the source model and updated after each persistent student step using an exponential moving average (Wang et al., 2022; Brahma & Rai, 2023):

$$\bar{\boldsymbol{\theta}}_t = m\bar{\boldsymbol{\theta}}_{t-1} + (1-m)\boldsymbol{\theta}_{t+1}, \tag{18}$$

where $m \in [0, 1)$ denotes the teacher momentum and $\boldsymbol{\theta}_{t+1}$ is the student after the persistent update for batch t. Before adapting to $\mathcal{B}_t$, we compute the teacher probabilities and pseudo-labels as:

$$\mathbf{Q}_t = \mathrm{softmax}\left(f_{\bar{\boldsymbol{\theta}}_{t-1}}(\mathcal{B}_t)\right), \qquad \hat{y}_{t,i} = \arg\max_c q_{t,i,c}. \tag{19}$$

Source prototypes are computed using Eq. (4) and retained as the fixed matrix $\mathbf{P}$ after source examples are discarded. They provide class-specific references for the fast residuals.

Let $\mathbf{W}_0 \in \mathbb{R}^{C\times L}$ denote the source classifier weight. Following GOLD, we initialize the slow AGOP statistic and its rank-r_s basis as:

$$\mathbf{M}_0^{\text{slow}} = \mathbf{W}_0^\top \mathbf{W}_0, \qquad \mathbf{V}_1^{\text{slow}} = \text{TopEig}_{r_s}(\mathbf{M}_0^{\text{slow}}). \tag{20}$$

At target batch t, the slow statistic is updated only from examples whose adapted-student confidence meets the threshold $\tau_s \in [0,1]$:

$$\mathcal{M}_t = \left\{ i : \max_c \text{softmax}(\mathbf{O}_t^{\text{sup}})_{i,c} \geq \tau_s \right\}. \tag{21}$$

For each selected example, we compute the gradient of the largest current-student logit with respect to its raw feature, treating the feature as an independent variable:

$$\boldsymbol{g}_{t,i} = \nabla_{\boldsymbol{f}_{t,i}} \max_c h_{\boldsymbol{\psi}_t}(\boldsymbol{f}_{t,i})_c, \tag{22}$$

$$\widehat{\mathbf{M}}_t^{\text{slow}} = \frac{1}{|\mathcal{M}_t|} \sum_{i\in\mathcal{M}_t} \boldsymbol{g}_{t,i}\boldsymbol{g}_{t,i}^\top. \tag{23}$$

If fewer than two examples pass the confidence threshold, the slow-statistic update is skipped. Otherwise, $\widehat{\mathbf{M}}_t^{\text{slow}}$ is incorporated using Eq. (7). The slow basis is refreshed every T_s batches. The AGOP construction is excluded from differentiation of the outer objective.

A.2 Descriptor and Episodic-Optimization Details

For classification, each view $v \in \{\text{sup}, \text{qry}\}$ uses a batch descriptor computed from slow-adapted prototype residuals projected onto the current fast basis:

$$\boldsymbol{d}_{t,i}^v = \boldsymbol{f}_{t,i}^{\text{slow},v} - \boldsymbol{p}_{\hat{y}_{t,i}}, \qquad \boldsymbol{a}_{t,i}^v = (\mathbf{V}_t^{\text{fast}})^\top \boldsymbol{d}_{t,i}^v \in \mathbb{R}^{r_f}. \tag{24}$$

The support and query descriptors use their respective features but share the same pseudo-labels, prototypes, and fast basis. Let $\boldsymbol{\mu}_t^v$ and $\boldsymbol{v}_t^v$ denote the coordinate-wise mean and population variance of $\boldsymbol{a}_{t,i}^v$. The transformed statistics in Eq. (10) are

$$\hat{\boldsymbol{\mu}}_t^v = \boldsymbol{\mu}_t^v \oslash \sqrt{\max(\boldsymbol{v}_t^v, \epsilon \mathbf{1}_{r_f})}, \qquad \hat{\boldsymbol{v}}_t^v = \log(\mathbf{1}_{r_f} + \boldsymbol{v}_t^v), \qquad \hat{H}_t = \bar{H}_t / \log C, \tag{25}$$

where $\bar{c}_t = B^{-1}\sum_i \max_c q_{t,i,c}$ and $\bar{H}_t = -B^{-1}\sum_{i,c} q_{t,i,c}\log q_{t,i,c}$ are the mean teacher confidence and entropy. Both views share these teacher statistics. Here $\epsilon > 0$ stabilizes the denominator; $\oslash$, max, the square root, and the logarithm operate element-wise. We concatenate $[\hat{\boldsymbol{\mu}}_t^v; \hat{\boldsymbol{v}}_t^v; \bar{c}_t; \hat{H}_t]$ to form the batch descriptor. For a single-example classification batch, the variance is set to zero and the stabilized denominator in Eq. (25) is used. Segmentation computes residual statistics over selected pixels within each crop; its descriptor, loss, and update procedure are detailed in Appendix A.8. Following the symmetric teacher–student consistency used in RMT (Döbler et al., 2023), we define, for student logits $\boldsymbol{o}$ and teacher logits $\bar{\boldsymbol{o}}$:

$$\ell_{\text{sce}}(\boldsymbol{o}, \bar{\boldsymbol{o}}) = -\frac{1}{2}\left[\text{softmax}(\bar{\boldsymbol{o}})^\top \log \text{softmax}(\boldsymbol{o}) + \text{softmax}(\boldsymbol{o})^\top \log \text{softmax}(\bar{\boldsymbol{o}})\right]. \tag{26}$$

For $v \in \{\text{sup}, \text{qry}\}$, we regularize the raw MetaNet outputs:

$$\mathcal{R}_t^v = \|\tilde{\boldsymbol{s}}_{t,v}^{\text{fast}}\|_2^2 / r_f + \|\tilde{\boldsymbol{b}}_{t,v}^{\text{fast}}\|_2^2 / r_f. \tag{27}$$

The bias term is omitted when fast bias is disabled. This penalty discourages large raw outputs, while the $\tanh$ parameterization in Eq. (11) explicitly bounds the generated coefficients. The two-view teacher consistency term is given in Eq. (14).

For each classification episode, we generate the query view using the CoTTA-style augmentation pipeline (Wang et al., 2022), including color jitter, random affine transformations, Gaussian blur, cropping, horizontal flipping, and additive Gaussian noise. These transformations are assumed to preserve the class label. Query predictions are matched to teacher logits computed from the corresponding original samples. The deployed prediction is emitted before the persistent optimizer and EMA-teacher updates.

Prototype-guided contrastive loss. The contrastive term in Eq. (16) anchors the two adapted views to a source prototype selected by feature similarity (Döbler et al., 2023; Chakrabarty et al., 2023; Lai et al., 2026). Here $\boldsymbol{f}_{t,i}^{v}$ denotes the feature after both pathways for view $v \in \{\mathrm{sup}, \mathrm{qry}\}$. For $k(i) = \arg\max_c \cos(\boldsymbol{f}_{t,i}^{\mathrm{sup}}, \boldsymbol{p}_c)$, let $\boldsymbol{z}_{i,0} = \Pi(\boldsymbol{p}_{k(i)})$, $\boldsymbol{z}_{i,1} = \Pi(\boldsymbol{f}_{t,i}^{\mathrm{sup}})$, and $\boldsymbol{z}_{i,2} = \Pi(\boldsymbol{f}_{t,i}^{\mathrm{qry}})$ denote normalized projections. With $\mathcal{A} = \{(i,v) : i \in [B], v \in \{0,1,2\}\}$ and $\mathcal{P}(i,v) = \{(i,u) \in \mathcal{A} : u \neq v\}$, these projections define the contrastive loss in Eq. (15). Here $\Pi_{\boldsymbol{\xi}} : \mathbb{R}^L \to \mathbb{R}^{d_{\mathrm{proj}}}$ denotes the projection head followed by normalization, parameterized by $\boldsymbol{\xi}$, and κ is the temperature. We abbreviate $\Pi_{\boldsymbol{\xi}}$ as Π above.

A.3 Parameter Semantics, Search Ranges, and Online Algorithm

Table 6 summarizes the principal classification hyperparameters. The implementation caps r_s and r_f by the feature dimension L. The base Adam learning rate is η_{out}. The MetaNet parameter group uses learning rate $\rho_\phi \eta_{\mathrm{out}}$, while the persistent slow scale uses the base rate.

The remaining settings are shared. We use teacher momentum $m = 0.999$, slow confidence threshold $\tau_s = 0.8$, slow-basis refresh period $T_s = 50$, contrastive temperature $\kappa = 0.1$, and a 128-dimensional projection head. The loss weight is $\lambda_{\mathrm{ce}} = 1$. Each batch receives one Adam step with $\boldsymbol{\beta} = (0.9, 0.999)$ and zero weight decay. Fast bias and first-order episodic adaptation are enabled; thus deployed logits are the sum of adapted-student and EMA-teacher logits.

Section B.5 evaluates the sensitivity to r_f, η_{in}, γ_s, γ_b, τ_s and the MetaNet hidden width d_h. For the inherited slow pathway, we refer to the broader hyperparameter study reported by GOLD (Lai et al., 2026); the dataset-specific values used here are listed in Table 6.

Coordinate alignment at basis refresh. To maintain consistent coordinates for the MetaNet's projected inputs and generated coefficients, we align each refreshed fast basis $\widetilde{\mathbf{V}}$ to its previous basis $\mathbf{V}_{\mathrm{prev}}$ using orthogonal Procrustes alignment:

$$\widetilde{\mathbf{V}}^\top \mathbf{V}_{\mathrm{prev}} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{R}^\top, \qquad \mathrm{Align}(\widetilde{\mathbf{V}}, \mathbf{V}_{\mathrm{prev}}) = \widetilde{\mathbf{V}}\mathbf{U}\mathbf{R}^\top, \tag{28}$$

where the first equality is a singular value decomposition. This rotation minimizes the Frobenius distance to the previous basis while preserving the newly estimated subspace and orthonormality, removing arbitrary sign, ordering, and within-subspace rotation changes. The descriptor and fast modulation use the same aligned basis. Initialization uses the raw fast eigenbasis. The aligned columns span the leading eigenspace but need not individually be eigenvectors.

Algorithm 1 summarizes the classification implementation. The segmentation variant uses crop-level support–query episodes, spatial residual modulation, and predictions from the updated EMA teacher; its implementation is detailed in Appendix A.8.

Table 6: Meaning and values of the principal DTS-META hyperparameters in the classification experiments. C10-C, C100-C, and IN-C denote CIFAR-10-C, CIFAR-100-C, and ImageNet-C, respectively.

Symbol	Meaning	C10-C	C100-C	IN-C
B	Incoming online batch size	200	200	64
η_{out}	Persistent Adam learning rate	10^{-3}	10^{-3}	10^{-2}
r_s	Rank of the class-sensitive slow basis	64	64	256
r_f	Rank of the fast domain-sensitive basis	64	64	128
α_s	EMA coefficient for the slow AGOP statistic	0.08	0.04	0.02
α_f	EMA coefficient for the fast residual covariance	0.4	0.4	0.2
T_s	Slow-basis refresh period (batches)	50	50	50
T_f	Fast-statistic and eigenspace refresh period (batches)	5	5	5
d_h	Hidden width of the two-layer MetaNet	256	512	512
η_{in}	Temporary MetaNet inner-update rate	10^{-4}	5×10^{-4}	10^{-4}
γ_s	Bound on generated fast scale coefficients	0.2	0.2	0.2
γ_b	Bound on generated fast bias coefficients	0.02	0.01	0.01
λ_{reg}	Weight on raw MetaNet-output regularization	10^{-6}	10^{-5}	10^{-5}
λ_{con}	Weight on prototype-anchored contrastive loss	1.0	1.0	2.0
ρ_ϕ	MetaNet outer-LR multiplier	2.0	1.0	0.5
τ_s	Confidence threshold for slow AGOP selection	0.8	0.8	0.8

Algorithm 1 Online DTS-Meta adaptation for one target stream

Require: Source model $f_{\boldsymbol{\theta}_0} = h_{\boldsymbol{\psi}_0} \circ g_{\boldsymbol{\omega}_0}$, source prototypes $\mathbf{P}$, initial slow scale $\boldsymbol{s}^{\text{slow}}$ and MetaNet parameters $\boldsymbol{\phi}$, unlabeled batches $\{\mathcal{B}_t\}$

$\bar{\boldsymbol{\theta}} \leftarrow \boldsymbol{\theta}_0$; $\mathbf{M}^{\text{slow}} \leftarrow \mathbf{W}_0^\top \mathbf{W}_0$; $\mathbf{V}^{\text{slow}} \leftarrow \text{TopEig}_{r_s}(\mathbf{M}^{\text{slow}})$
$\mathbf{M}^{\text{fast}} \leftarrow \mathbf{I}_L$; $\mathbf{V}^{\text{fast}} \leftarrow \text{TopEig}_{r_f}(\mathbf{M}^{\text{fast}})$
for $t = 1, 2, \ldots$ **do**
  $\mathbf{F} \leftarrow g_{\boldsymbol{\omega}}(\mathcal{B}_t)$; $\mathbf{O}^{\text{ema}} \leftarrow \text{stopgrad}(f_{\bar{\boldsymbol{\theta}}}(\mathcal{B}_t))$; $\hat{y}_i \leftarrow \arg\max_c \mathbf{O}^{\text{ema}}_{i,c}$
  if $(t-1) \bmod T_f = 0$ and $B \geq 2$ **then**
    Form $\mathbf{C}^{\text{fast}}$ from centered residuals $\boldsymbol{f}_i - \boldsymbol{p}_{\hat{y}_i}$ using Eq. (8)
    $\mathbf{M}^{\text{fast}} \leftarrow (1-\alpha_f)\mathbf{M}^{\text{fast}} + \alpha_f \mathbf{C}^{\text{fast}}$
    $\mathbf{V}^{\text{fast}} \leftarrow \text{Align}(\text{TopEig}_{r_f}(\mathbf{M}^{\text{fast}}), \mathbf{V}^{\text{fast}})$ ▷ Eq. (28)
  end if
  $\mathbf{F}^{\text{slow}} \leftarrow \textsc{Slow}(\mathbf{F}, \mathbf{V}^{\text{slow}}, \boldsymbol{s}^{\text{slow}})$; $\boldsymbol{u}^{\text{sup}} \leftarrow \textsc{Desc}(\mathbf{F}^{\text{slow}}, \mathbf{O}^{\text{ema}}, \mathbf{P}, \mathbf{V}^{\text{fast}})$
  $(\tilde{\boldsymbol{s}}, \tilde{\boldsymbol{b}}) \leftarrow \mathcal{H}_{\boldsymbol{\phi}}(\boldsymbol{u}^{\text{sup}})$; $\mathbf{F}^{\text{sup}} \leftarrow \textsc{Fast}(\mathbf{F}^{\text{slow}}, \mathbf{V}^{\text{fast}}, \gamma_s \tanh \tilde{\boldsymbol{s}}, \gamma_b \tanh \tilde{\boldsymbol{b}})$; $\mathbf{O}^{\text{sup}} \leftarrow h_{\boldsymbol{\psi}}(\mathbf{F}^{\text{sup}})$
  $\mathcal{L}^{\text{inner}} \leftarrow \ell_{\text{sce}}(\mathbf{O}^{\text{sup}}, \mathbf{O}^{\text{ema}}) + \lambda_{\text{reg}} \mathcal{R}^{\text{sup}}$; $\boldsymbol{\phi}' \leftarrow \boldsymbol{\phi} - \eta_{\text{in}} \text{stopgrad}(\nabla_{\boldsymbol{\phi}} \mathcal{L}^{\text{inner}})$
  $\mathcal{B}_t^{\text{qry}} \leftarrow \mathcal{T}(\mathcal{B}_t)$; compute $\mathbf{O}^{\text{qry}}$ from the query descriptor and $\mathcal{H}_{\boldsymbol{\phi}'}$ using the same $\mathbf{O}^{\text{ema}}$
  Compute confidence-filtered $\widehat{\mathbf{M}}^{\text{slow}}$ from $(\mathbf{F}, \mathbf{O}^{\text{sup}})$ using Eq. (23); skip if fewer than two samples pass
  if a valid $\widehat{\mathbf{M}}^{\text{slow}}$ is available **then**
    $\mathbf{M}^{\text{slow}} \leftarrow (1-\alpha_s)\mathbf{M}^{\text{slow}} + \alpha_s \widehat{\mathbf{M}}^{\text{slow}}$
  end if
  if $t \bmod T_s = 0$ **then**
    $\mathbf{V}^{\text{slow}} \leftarrow \text{TopEig}_{r_s}(\mathbf{M}^{\text{slow}})$
  end if
  Construct $\mathcal{L}^{\text{outer}}$ from support/query consistency, prototype contrast, and modulation regularization using Eq. (16)
  Emit $\mathbf{O}^{\text{final}} \leftarrow \mathbf{O}^{\text{sup}} + \mathbf{O}^{\text{ema}}$ before changing persistent parameters
  $\boldsymbol{\vartheta} \leftarrow \text{AdamStep}(\boldsymbol{\vartheta}, \nabla_{\boldsymbol{\vartheta}} \mathcal{L}^{\text{outer}}, \eta_{\text{out}})$, with MetaNet step size $\rho_\phi \eta_{\text{out}}$
  $\bar{\boldsymbol{\theta}} \leftarrow m\bar{\boldsymbol{\theta}} + (1-m)\boldsymbol{\theta}$ ▷ teacher update is last
end for

A.4 Classification Datasets, Models, and Standard Streams

Datasets and streams. CIFAR-10-C and CIFAR-100-C are corruption benchmarks derived from the corresponding CIFAR test sets, while ImageNet-C is derived from the ImageNet validation set (Krizhevsky & Hinton, 2009; Deng et al., 2009; Hendrycks & Dietterich, 2019). Following the standard CTTA protocol, we traverse Gaussian noise, shot noise, impulse noise, defocus blur, glass blur, motion blur, zoom blur, snow, frost, fog, brightness, contrast, elastic transform, pixelation, and JPEG compression at severity 5. We evaluate all 10,000 examples per corruption on CIFAR-C and a 5,000-image subset per corruption on ImageNet-C, following the evaluation setup of GOLD (Lai et al., 2026). The model processes the complete sequence once without shuffling, replay, domain-boundary information, or reset. Target labels are used only for evaluation. For each dataset and seed, every method receives the same ordered mini-batches.

Models and implementation. We use the same source architectures and checkpoints as GOLD (Lai et al., 2026): WideResNet-28 for CIFAR-10-C (Zagoruyko & Komodakis, 2016), an AugMix-trained ResNeXt-29 for CIFAR-100-C (Xie et al., 2017; Hendrycks et al., 2020), and ResNet-50 for ImageNet-C (He et al., 2016). The corresponding batch sizes are 200, 200, and 64. All methods begin from the same dataset-specific checkpoint and process the same stream without test-label-based model selection.

A.5 Performance Across Test-Time Batch Sizes on ImageNet-C

In practical deployment, the throughput of target data may vary. To evaluate model robustness across different operational regimes, Table 7 reports classification error rates under varying test-time batch sizes $B \in \{4, 16, 32, 64\}$ on ImageNet-C under the standard corruption stream.

Table 7: Classification error rates (%, lower is better) on ImageNet-C across varying test-time batch sizes. All methods are evaluated under the standard severity-5 corruption stream. **Bold** indicates the best result, and underline indicates the second best.

Method	$B = 4$	$B = 16$	$B = 32$	$B = 64$
TENT (Wang et al., 2021)	99.0	82.5	64.5	62.6
AdaContrast (Chen et al., 2022a)	98.4	81.9	69.9	65.5
CoTTA (Wang et al., 2022)	97.5	82.0	64.8	62.7
RMT (Döbler et al., 2023)	99.4	83.4	68.7	60.2
GOLD (Lai et al., 2026)	97.4	69.8	62.6	59.4
DTS-META (Ours)	**97.2**	**68.2**	**62.0**	**58.7**

Effect of batch size. Table 7 shows that DTS-META achieves the lowest reported error among the compared methods at each evaluated batch size. At $B = 16$, its error is 68.2%, compared with 69.8% for GOLD. At $B = 4$, DTS-META has an error rate of 97.2%, indicating substantial performance degradation. Like the compared methods, it benefits from larger batches and records the lowest error at every evaluated batch size.

A.6 CDC STREAM CONSTRUCTION

We construct CDC streams using the dynamic-domain protocol of DPCore (Zhang et al., 2025). Let $D = 15$ be the number of corruption domains, N_d the number of mini-batches in domain d, and $K = 3$ the number of temporal slots. For each domain, we independently draw

$$\boldsymbol{\pi}_d \sim \mathrm{Dirichlet}(\delta \mathbf{1}_K), \qquad \delta = 1. \tag{29}$$

Table 8: Per-corruption classification error (%, lower is better) under CDC with Dirichlet concentration $\delta = 1$. All methods use the same realized stream for each seed and are evaluated online at severity 5. For DTS-META and GOLD , the Mean column reports the mean $\pm$ std across five independent seeds. **Bold** and underline denote the best and second-best results, respectively.

	Method	**Gau.**	**Shot**	**Imp.**	**Def.**	**Glass**	**Mot.**	**Zoom**	**Snow**	**Frost**	**Fog**	**Bri.**	**Con.**	**Ela.**	**Pix.**	**JPEG**	**Mean**
CIFAR-10-C	TENT	23.02	21.62	30.86	14.10	30.85	15.28	14.11	16.54	15.76	16.96	11.25	14.16	21.59	17.34	22.78	19.08
	Ada	28.19	26.93	30.78	13.12	32.02	15.61	12.53	16.92	16.67	15.33	8.59	11.50	23.37	22.56	22.19	19.75
	CoTTA	22.05	21.28	24.17	10.83	26.26	12.43	10.54	15.82	14.86	12.37	**8.08**	11.88	20.22	16.29	19.96	16.47
	RMT	20.32	19.15	22.77	11.88	23.12	13.40	12.00	15.94	15.36	14.03	10.83	13.35	19.00	15.31	18.60	16.34
	SAR	28.49	26.21	36.48	12.74	34.91	14.16	11.99	17.63	17.62	14.92	8.38	13.04	23.55	19.35	27.36	20.46
	EATA	22.54	20.38	28.83	13.49	27.65	14.22	12.96	15.76	14.63	16.03	10.02	12.73	20.63	15.98	21.08	17.80
	SANTA	21.49	19.17	27.45	11.03	28.2	12.62	9.78	14.5	13.74	12.85	8.24	11.71	19.59	15.37	20.54	16.42
	OBAO	19.32	18.44	22.81	**10.21**	23.15	12.23	**9.74**	14.41	13.75	12.10	8.63	11.40	17.63	14.06	18.24	15.07
	GOLD	18.67	18.24	22.15	10.39	**22.74**	12.43	10.22	14.52	14.26	11.8	8.91	11.07	**17.58**	14.06	17.75	$14.99_{\pm 0.14}$
	DTS-META	**16.40**	**15.54**	**21.77**	10.22	23.38	**11.91**	10.12	**13.42**	**12.91**	**11.29**	8.71	**10.69**	17.99	**12.67**	**17.29**	$\mathbf{14.29}_{\pm 0.10}$
CIFAR-100-C	TENT	44.16	42.01	65.08	59.69	55.1	49.31	70.68	45.13	40.68	70.13	70.8	53.1	50.47	43.88	49.86	54.01
	Ada	41.48	39.81	39.65	27.81	39.47	30.12	27.73	33.02	31.61	39.4	26.86	30.71	34.62	37.58	37.15	34.47
	CoTTA	38.53	37.47	38.2	25.74	36.61	28.5	25.73	34.01	33.1	40.39	24.97	28.17	33.97	29.98	35.79	32.74
	RMT	34.33	33.53	33.76	26.79	32.32	28.48	26.01	30.98	30.34	31.53	26.13	28.27	30.71	28.99	32.93	30.34
	SAR	36.43	34.62	38.45	26.33	35.86	28.00	26.18	31.93	30.03	36.64	26.07	28.36	32.4	29.38	37.04	31.85
	EATA	35.61	34.49	38.37	28.76	37.05	29.97	28.11	32.92	30.82	37.9	28.96	30.98	33.5	30.06	37.02	32.97
	SANTA	34.72	32.9	37.17	25.85	35.45	28.25	24.91	30.46	29.8	34.92	24.53	27.95	31.87	28.3	35.36	30.83
	OBAO	34.18	32.59	33.57	24.92	32.47	26.62	24.43	29.47	29.5	30.86	24.04	26.61	30.15	27.7	32.77	29.33
	GOLD	32.00	31.76	**32.60**	24.59	31.80	26.41	**24.12**	28.84	28.42	29.72	24.10	26.45	**29.69**	26.93	31.66	$28.61_{\pm 0.16}$
	DTS-META	**30.01**	**29.42**	32.89	**24.03**	**31.72**	**24.88**	24.20	**27.17**	**26.88**	**29.40**	**22.99**	**25.38**	29.98	**25.10**	**30.72**	$\mathbf{27.65}_{\pm 0.13}$
ImageNet-C	TENT	74.76	74.04	72.20	74.02	72.68	65.84	56.32	61.78	61.34	52.34	39.34	71.10	50.86	47.90	53.88	61.89
	Ada	76.50	75.56	72.14	72.54	73.54	71.30	59.66	63.28	63.76	56.80	41.96	75.16	56.66	51.50	55.60	64.40
	CoTTA	76.08	77.10	**66.50**	68.10	71.18	63.86	**49.30**	60.64	60.82	45.66	36.74	68.84	50.66	45.74	52.76	59.60
	RMT	73.72	74.88	68.44	67.96	72.56	61.40	50.54	56.86	59.88	45.74	**35.50**	67.68	49.06	45.40	51.30	58.73
	SAR	74.56	74.74	72.22	74.34	72.14	65.80	56.04	61.74	61.12	52.12	38.42	70.96	50.04	47.58	53.72	61.70
	EATA	68.78	**67.16**	66.66	68.66	**67.48**	63.74	55.50	61.10	60.32	54.24	43.54	67.26	50.36	47.62	52.50	59.66
	SANTA	70.20	69.46	70.46	72.24	73.84	63.76	54.50	59.88	62.08	49.12	39.46	70.26	51.24	47.30	51.76	60.37
	OBAO	73.92	75.26	68.08	**67.68**	71.56	61.76	49.76	58.06	60.96	**44.32**	36.06	69.12	49.66	**45.20**	52.14	58.90
	GOLD	70.40	70.46	67.64	68.76	70.78	61.44	53.88	58.62	59.66	49.84	40.92	**63.70**	48.82	46.98	50.98	$58.86_{\pm 0.13}$
	DTS-META	**68.70**	72.77	68.56	70.76	71.57	**60.96**	53.35	**55.45**	**58.29**	46.24	37.77	65.27	**48.55**	46.22	**49.80**	$\mathbf{58.28}_{\pm 0.15}$

We partition the ordered mini-batches of domain d into K contiguous segments whose sizes are proportional to $\boldsymbol{\pi}_d$. Within each temporal slot, the available domain segments are randomly permuted and concatenated. Concatenating the three slots produces the complete stream. Every mini-batch is evaluated exactly once, but domains may recur with unequal exposure lengths and irregular transition times. Domain identities and boundaries are not provided to the methods, and model state is not reset. All methods use the same realized schedule for a given seed.

Table 8 provides the complete per-corruption results; the realized stream is shared across all methods for each seed, and the seed list and stream statistics are provided with the reproducibility materials.

A.7 Detailed CDC Classification Results

Table 8 expands the aggregate CDC results in Table 2 into the 15 corruption categories used in the standard stream. For each seed, all methods process the same realized CDC schedule. The Mean column averages the 15 corruption-wise error rates.

A.8 Additional Evaluation Protocols

Gradual corruption shifts. For CIFAR-100-C, we preserve the standard order of the 15 corruption types and vary severity within each corruption according to:

$$1 \to 2 \to 3 \to 4 \to 5 \to 4 \to 3 \to 2 \to 1.$$

The resulting stream contains 135 consecutive corruption–severity domains. It is processed once with batch size 200, without model reset or domain-boundary information. We report the mean error over the complete stream and over the 15 peak-severity domains (Wang et al., 2022; Döbler et al., 2023).

Table 9: Gradual-shift classification error (%, lower is better) on CIFAR-100-C. Mean and Sev.-5 average all 135 domains and the 15 peak-severity domains, respectively.

Method	Mean	Sev.-5
TENT	78.16	79.20
Ada	32.91	35.71
CoTTA	26.77	29.74
RMT	26.74	27.26
SAR	28.71	31.93
EATA	31.96	34.34
SANTA	25.89	29.02
OBAO	25.71	26.44
GOLD	<u>25.60</u>	<u>26.20</u>
DTS-Meta	**24.84**	**25.60**

Long-horizon adaptation. We repeat the complete severity-5 CIFAR-100-C stream for ten consecutive rounds using batch size 200. Model and optimizer states persist across corruption and round boundaries, and no boundary information is provided. Table 4 reports the error for each round and the mean over all ten rounds. This setting measures how performance changes as online updates accumulate over repeated exposure to the same corruption sequence (Zhu et al., 2024; Song et al., 2023; Press et al., 2023).

Continual semantic segmentation. CarlaTTA (Marsden et al., 2024), built with CARLA (Dosovitskiy et al., 2017), contains day-to-night, clear-to-fog, clear-to-rain, dynamic-weather, and urban-to-highway sequences. Each sequence starts from the same clear-weather DeepLabV2–ResNet-101 checkpoint (Chen et al., 2018) and is processed in temporal order without target labels. We resize the shorter side to 1024 pixels, process one frame at a time, and report sequence-level mIoU over 14 classes using the accumulated confusion matrix.

The segmentation variant retains dual-timescale subspace updates and MetaNet-generated residual modulation. The slow subspace constrains the fast correction directions through orthogonal projection, without a separate trainable slow scaling branch. Source pseudo-labels provide fixed prototypes

and feature variances for fast-statistic initialization, while confident teacher pixels maintain online target prototypes. The MetaNet uses sampled pixel residual statistics and teacher uncertainty to generate bounded scale and bias relative to mixed source–target prototypes.

Each frame yields two random crops for a support–query episode. The student receives augmentation, while teacher targets are computed from the corresponding crops before this augmentation. A first-order support step temporarily updates the MetaNet, and the query crop evaluates the adapted parameters. One persistent SGD step combines teacher consistency, prototype contrast, and modulation regularization, followed by the EMA-teacher update. The updated teacher predicts the complete frame, with adaptation state retained across frames within each sequence.

B Additional Experimental Analyses

B.1 Timescale Coordination Under CDC

The following controlled experiment isolates the effect of refresh schedules. We retain the complete DTS-META architecture, episodic training, source checkpoint, batch order, and all optimization settings, and vary only the slow and fast refresh periods (T_s, T_f). Results are reported on ImageNet-C under the recurring CDC stream with Dirichlet concentration $\delta = 1$; lower error is better.

Table 10: **Ablation on timescale coordination under CDC.** Classification error rates (%, lower is better) on ImageNet-C under recurring Dirichlet domain changes ($\delta = 1$). All variants retain the full dual-pathway architecture and episodic training, varying only the refresh schedules (T_s, T_f).

Configuration	T_s	T_f	Error (%)
Both Frequent	5	5	58.56
Both Infrequent	50	50	58.63
Inverted Timescales	5	50	58.76
DTS-META (Default)	**50**	**5**	**58.28**

The default schedule gives the lowest error among the four otherwise matched configurations. Using equal basis-refresh periods, or reversing their relative frequencies, yields higher error than the default schedule in this experiment.

B.2 Subspace Construction and Temporal Distance

Figure 1(a) measures the change in each subspace between consecutive batches. The diagnostic uses the CIFAR-100-C stream and the same student feature trajectory for both statistics. In every batch, both statistics use the same teacher-selected samples and the same rank $r = 8$; we do not apply EMA smoothing, and decompose each batch statistic directly. The plotted solid curves are five-batch moving averages of the resulting per-batch distances. Thus, the comparison of consecutive subspaces keeps the sample set, rank, and decomposition frequency fixed, differing only in the statistic used to construct the basis.

Two subspace statistics. The statistic used to estimate the class-sensitive subspace is the average outer product of top-logit feature gradients:

$$\boldsymbol{g}_{t,i} = \nabla_{\boldsymbol{f}_{t,i}} \max_{c\in[C]} [h_{\boldsymbol{\psi}_t}(\boldsymbol{f}_{t,i})]_c, \qquad \widehat{\mathbf{M}}_t^{\mathrm{cls}} = \frac{1}{n_t} \sum_{i=1}^{n_t} \boldsymbol{g}_{t,i} \boldsymbol{g}_{t,i}^\top. \tag{30}$$

Its leading eigenvectors identify class-sensitive directions. For the domain-sensitive statistic, we subtract the source prototype associated with each teacher pseudo-label and compute the covariance:

$$\boldsymbol{r}_{t,i} = \boldsymbol{f}_{t,i} - \boldsymbol{p}_{\hat{y}_{t,i}}, \qquad \widehat{\mathbf{M}}_t^{\mathrm{res}} = \frac{1}{n_t - 1} \sum_{i=1}^{n_t} (\boldsymbol{r}_{t,i} - \bar{\boldsymbol{r}}_t)(\boldsymbol{r}_{t,i} - \bar{\boldsymbol{r}}_t)^\top, \tag{31}$$

where $\bar{\boldsymbol{r}}_t = n_t^{-1} \sum_i \boldsymbol{r}_{t,i}$ is the batch residual mean and $\boldsymbol{p}_c$ is defined in Eq. (4). Its leading eigenvectors describe variation around the class-specific references. The superscripts cls and res denote

the AGOP-based class-sensitive and residual-based domain-sensitive subspaces, respectively. For $k \in \{\text{cls}, \text{res}\}$, we directly decompose the batch statistic and extract a rank-r basis:

$$\mathbf{M}_t^k = \widehat{\mathbf{M}}_t^k, \qquad \mathbf{V}_t^k = \text{TopEig}_r(\mathbf{M}_t^k), \qquad (\mathbf{V}_t^k)^\top \mathbf{V}_t^k = \mathbf{I}_r. \tag{32}$$

Here $\mathbf{V}_t^k \in \mathbb{R}^{L \times r}$ contains the leading eigenvectors; the same rank is used for the two diagnostic subspaces.

Consecutive-subspace distance. For each type of subspace, we compare its bases at batches $t-1$ and t:

$$d_t^k = \frac{\|\mathbf{V}_t^k (\mathbf{V}_t^k)^\top - \mathbf{V}_{t-1}^k (\mathbf{V}_{t-1}^k)^\top\|_F}{\sqrt{2r}} = \sqrt{1 - \frac{\|(\mathbf{V}_{t-1}^k)^\top \mathbf{V}_t^k\|_F^2}{r}} \in [0, 1]. \tag{33}$$

Here $\|\cdot\|_F$ is the Frobenius norm. A distance of zero means that the consecutive estimates span the same subspace; larger values indicate less overlap, reaching one for orthogonal subspaces. Eigenvector sign changes and rotations of the basis within a subspace do not affect the distance. Each curve therefore tracks the temporal variation of one subspace type. Figure 1(a) plots the distances along the stream, and Figure 3 aligns them at domain transitions, with zero marking the first batch of the new domain. This alignment compares local temporal variation around a boundary; the paired-image experiment below separately examines sensitivity to changes in observation conditions. Following domain transitions, the class-sensitive subspace remains relatively stable, while the domain-sensitive subspace exhibits larger temporal changes.

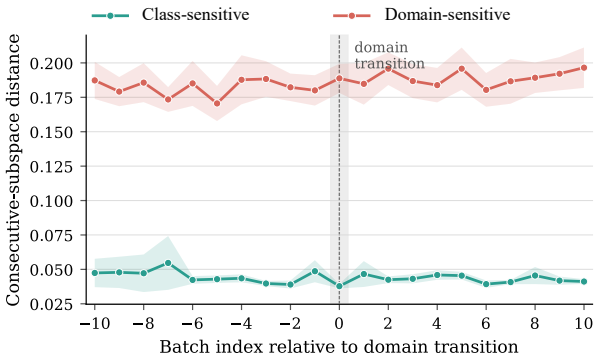


Figure 3: **Subspace dynamics around domain transitions.** Consecutive-subspace distances from Eq. (33) are aligned by the transition batch (dashed line, index zero). The class-sensitive subspace varies less than the domain-sensitive subspace throughout the displayed window.

B.3 Paired-Image Environment Sensitivity

Purpose and pairing. To isolate environmental changes from image content, we pair observations at matching indices across the 10,000-image severity-5 blocks of CIFAR-C. Each pair $(\boldsymbol{x}_i^a, \boldsymbol{x}_i^b)$ shares the original image and category while changing the corruption from a to b. This complements the temporal diagnostic by directly measuring sensitivity to observation conditions. Subspace fitting and evaluation use no target labels.

Fixed model and independent subspace fitting. We use source-pretrained AugMix ResNeXt on CIFAR-100-C ($L = 1024$), frozen in evaluation mode with fixed batch-normalization statistics and no adapter. The same model supplies pseudo-labels. A seed-1 permutation assigns 2,000 original-image indices to fitting and the next 4,000 to evaluation; this disjoint split is shared across corruptions. From the fitting images in severity-5 fog, we estimate the raw AGOP and centered prototype-residual covariance in Eqs. (30)–(31), using cached source class means without confidence filtering or temporal smoothing. Both orthonormal bases, $\mathbf{V}^{\text{cls}}$ and $\mathbf{V}^{\text{res}}$, have rank 8 and remain fixed across evaluation pairs. Equal ranks control projection capacity; disjoint indices prevent image overlap between fitting and evaluation.

Paired-change energy. For each of the $\binom{15}{2} = 105$ unordered corruption pairs, we extract features for the same $N = 4000$ held-out images with batch size 200 and compute

$$\Delta \boldsymbol{f}_i^{a,b} = g_{\boldsymbol{\omega}_0}(\boldsymbol{x}_i^b) - g_{\boldsymbol{\omega}_0}(\boldsymbol{x}_i^a), \qquad E_k(a,b) = \frac{\sum_{i=1}^N \|(\mathbf{V}^k)^\top \Delta \boldsymbol{f}_i^{a,b}\|_2^2}{\sum_{i=1}^N \|\Delta \boldsymbol{f}_i^{a,b}\|_2^2}, \quad k \in \{\text{cls}, \text{res}\}. \tag{34}$$

Here $E_k(a,b) \in [0,1]$ is the fraction of total paired-change energy captured by subspace k. We project the raw feature difference, without prototype subtraction at evaluation time. The ratio is computed after summing energies over images, rather than by averaging per-image ratios. We also evaluate 20 random orthonormal rank-8 bases, whose isotropic expected fraction is $r/L = 8/1024$. For uncertainty estimates, we jointly resample image indices for both subspaces 1,000 times and recompute the difference of their energy fractions. The 2.5th and 97.5th percentiles give a paired-image bootstrap interval conditional on the fixed model and fitted bases.

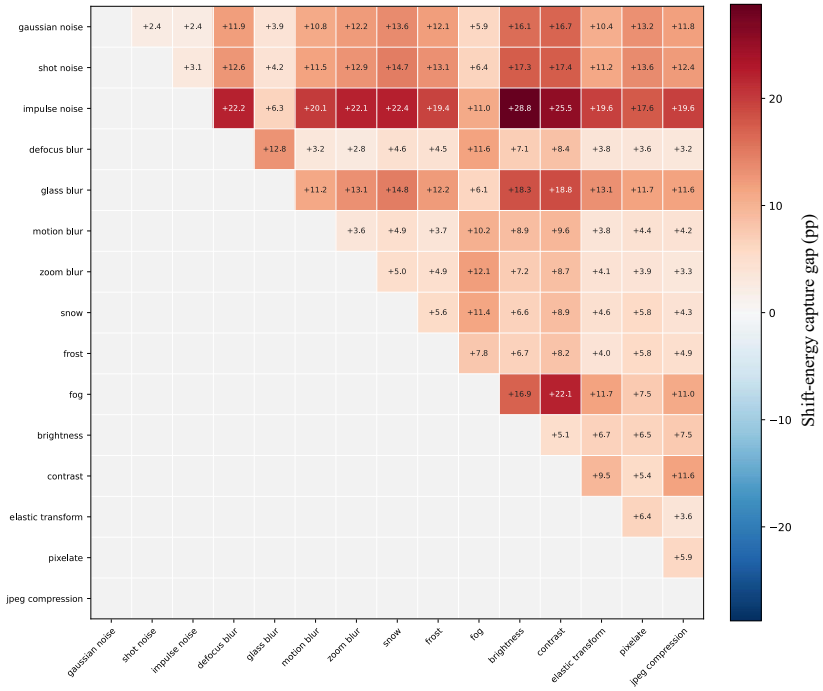


Figure 4: **Paired environment sensitivity across all 15 corruptions.** The vertical and horizontal axes identify the two corruption types a and b applied to the same images. Each upper-triangular cell reports $100[E_{\text{res}}(a,b) - E_{\text{cls}}(a,b)]$ in percentage points on CIFAR-100-C. The color bar ("Shift-energy capture gap (pp)") shows this difference in captured fractions of paired feature-change energy; positive values indicate greater capture by the domain-sensitive subspace than by the class-sensitive subspace. All corruptions use severity 5. Gray cells are omitted symmetric entries and self-pairs, not zero effects.

Results and interpretation. Figure 4 extends the selected comparisons in Figure 1(b) to all corruption pairs. The domain-sensitive subspace captures more paired-change energy in all 105 pairs, with differences ranging from 2.40 to 28.78 percentage points (mean 9.99). The domain-sensitive subspace captures more environment-change energy, indicating greater sensitivity to environmental variation. In contrast, the temporal diagnostic shows that the class-sensitive subspace remains relatively stable. These complementary properties motivate separate pathways for persistent task-related modulation and environment-dependent correction.

B.4 Qualitative Segmentation Results

Figure 5 compares five methods on CarlaTTA frames under dynamic weather, highway, rain, fog, and night conditions. The examples accompany the mIoU results in Table 5. Dashed boxes highlight regions in the DTS-META predictions.

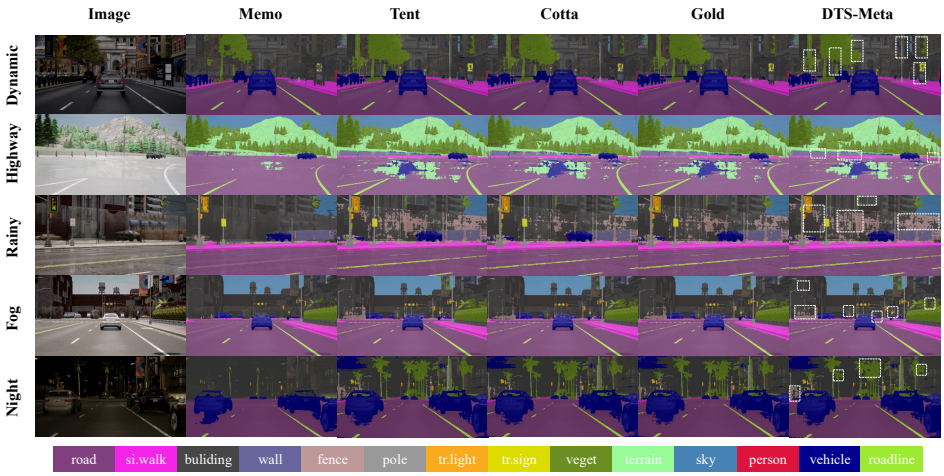


Figure 5: Qualitative semantic segmentation results on representative CarlaTTA frames. Columns show the input image and predictions from MEMO, TENT, CoTTA, GOLD, and DTS-META; rows cover dynamic weather, highway, rain, fog, and night conditions. White dashed boxes mark representative regions in the DTS-META outputs. Colors follow the class legend at the bottom.

B.5 Hyperparameter Sensitivity

We vary six hyperparameters on CIFAR-100-C under standard CTTA, changing one at a time while fixing the source checkpoint, batch order, and remaining settings. Figure 6 shows the tested values and marks the defaults with stars. Mean error varies by only 0.26 percentage points across all six sweeps.

The fast rank has no monotonic effect on accuracy (Figure 6(a)). Rank 128 gives the lowest error, but ranks 16 and 64 perform similarly. We use $r_f = 64$ to limit the projection and MetaNet dimensions while retaining near-best accuracy. The inner-step size also has a small effect across the tested range (Figure 6(b)); the default $\eta_{\text{in}} = 5 \times 10^{-4}$ gives the lowest error.

Moderate changes to the scale and bias bounds have little effect (Figure 6(c–d)). The defaults $\gamma_s = 0.2$ and $\gamma_b = 0.01$ perform best in their respective sweeps, while the largest bounds slightly increase error. Larger bounds did not reduce the observed error in these runs.

Performance also varies little with the slow AGOP confidence threshold, with $\tau_s = 0.8$ giving the lowest error (Figure 6(e)). Increasing the MetaNet hidden width from 32 to 512 steadily lowers error over the tested values (Figure 6(f)), suggesting that additional capacity helps map batch descriptors to modulation coefficients. These trends describe the plotted runs; small differences alone do not establish statistical significance.

B.6 Runtime Comparison

We compare per-batch adaptation and prediction time on a single RTX 4090 D using the standard CTTA streams. All methods use full precision and one adaptation step per batch, with batch sizes of 200 for CIFAR-C and 64 for ImageNet-C. We measure GPU-synchronized wall-clock time, including online optimization and subspace updates. Figure 7 presents the runtime comparison across methods.

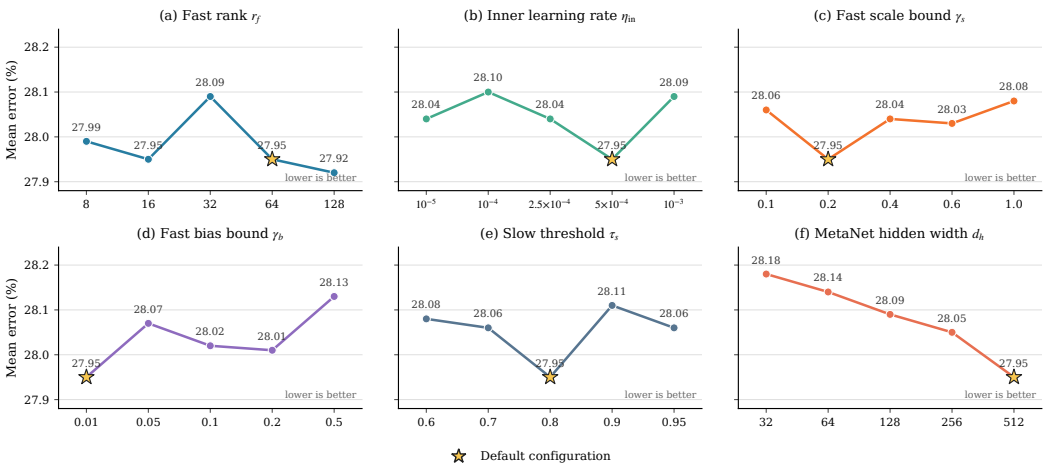


Figure 6: Hyperparameter sensitivity on CIFAR-100-C under standard CTTA: (a) fast rank, (b) inner-step size, (c) scale bound, (d) bias bound, (e) slow AGOP confidence threshold, and (f) MetaNet hidden width. Each panel varies one factor and reports mean error (%, lower is better); stars mark the default settings. All panels share a narrow vertical range to show small differences, and all results are point estimates.

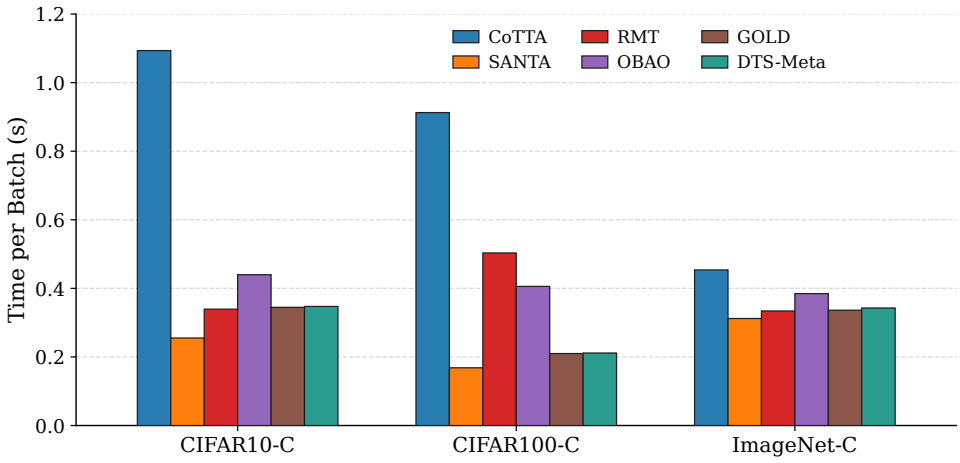


Figure 7: Mean adaptation and prediction time per batch on the standard CTTA streams, measured on an RTX 4090 D. All methods use one FP32 adaptation step per batch. Lower is better.

DTS-META takes approximately 0.35, 0.21, and 0.34 seconds per batch on CIFAR-10-C, CIFAR-100-C, and ImageNet-C, respectively. Its measured runtime is within 2% of GOLD on all three benchmarks, despite the additional fast pathway and episodic MetaNet update. On ImageNet-C, the other methods have similar runtimes. Under this setup, the proposed adapter improves accuracy with little additional runtime.

B.7 DETAILED ABLATION CONFIGURATIONS

Table 3 compares five configurations: w/o Adapter, Slow, Dual-Shared, Forward, and Full. These add slow modulation, the fast pathway, batch conditioning, and episodic optimization in turn. All use the same source initialization protocol, backbone, stream order, evaluation timing, and common optimizer settings.

w/o Adapter: common online framework without feature modulation. This baseline removes both modulation pathways and passes raw student features directly to the classifier. It retains the student–teacher framework, both input views, consistency and contrastive losses, and logit fusion, but excludes the modulation parameters, MetaNet, inner update, and modulation regularizer.

DTS-META-Slow: slow-only control under our objective. Adding the slow class-sensitive pathway gives

$$\mathbf{F}_t^{\mathrm{Slow}} = \mathbf{F}_t + \left[(\mathbf{F}_t \mathbf{V}_t^{\mathrm{slow}}) \odot \left(\mathbf{1}_B(\boldsymbol{s}_t^{\mathrm{slow}})^\top\right)\right] (\mathbf{V}_t^{\mathrm{slow}})^\top. \tag{35}$$

The AGOP basis and persistent scale follow Section 4.2. The scale is optimized under the reduced objective,

$$\mathcal{L}_t^{\mathrm{Slow}} = \mathcal{L}_t^{\mathrm{outer}}\big|_{\Delta_t^{\mathrm{fast}}=0}, \qquad \boldsymbol{s}_{t+1}^{\mathrm{slow}} = \boldsymbol{s}_t^{\mathrm{slow}} - \eta_{\mathrm{slow}} \nabla_{\boldsymbol{s}_t^{\mathrm{slow}}} \mathcal{L}_t^{\mathrm{Slow}}, \tag{36}$$

where Δ_t^{fast} denotes the fast residual correction. Fast modulation and its regularizer are omitted, and both views use the same persistent parameters. Comparing this variant with Dual-Shared tests the addition of the domain-sensitive pathway.

Dual-Shared: fast subspace with shared parameters. This variant adds the fast domain-sensitive basis with scale and bias vectors learned across the stream:

$$\begin{aligned}\mathbf{F}_t^{\mathrm{DS}} = \mathbf{F}_t^{\mathrm{slow}} + \big[&(\mathbf{F}_t^{\mathrm{slow}} \mathbf{V}_t^{\mathrm{fast}}) \odot \left(\mathbf{1}_B(\boldsymbol{s}^{\mathrm{fast}})^\top\right) \\ &+ \mathbf{1}_B(\boldsymbol{b}^{\mathrm{fast}})^\top\big] (\mathbf{V}_t^{\mathrm{fast}})^\top.\end{aligned} \tag{37}$$

The vectors $\boldsymbol{s}^{\mathrm{fast}}$ and $\boldsymbol{b}^{\mathrm{fast}}$ are updated directly by the outer objective. No MetaNet or inner update is used.

DTS-META-Forward: conditional generation without an inner step. Forward generates the fast parameters from the current batch descriptor:

$$\left(\tilde{\boldsymbol{s}}_t^{\mathrm{fast}}, \tilde{\boldsymbol{b}}_t^{\mathrm{fast}}\right) = \mathcal{H}_{\boldsymbol{\phi}_t}(\boldsymbol{u}_t), \tag{38}$$

where $\boldsymbol{u}_t$ is defined in Eq. (10). The raw outputs are bounded as in Eq. (11). Both views use the persistent MetaNet parameters $\boldsymbol{\phi}_t$, without an inner update. The fast basis and rank match Dual-Shared, though adding the MetaNet also increases capacity; this comparison does not isolate batch conditioning alone.

DTS-META-Full: episodic meta-adaptation. Full adds a temporary MetaNet update on the support view:

$$\boldsymbol{\phi}_t' = \boldsymbol{\phi}_t - \eta_{\mathrm{in}} \,\mathrm{stopgrad}\big(\nabla_{\boldsymbol{\phi}_t} \mathcal{L}_t^{\mathrm{inner}}(\boldsymbol{\phi}_t)\big). \tag{39}$$

The query view uses $\boldsymbol{\phi}_t'$, and the joint support–query objective updates the persistent parameters:

$$\boldsymbol{\vartheta}_{t+1} = \boldsymbol{\vartheta}_t - \eta_{\mathrm{out}} \nabla_{\boldsymbol{\vartheta}_t} \mathcal{L}_t^{\mathrm{outer}}(\boldsymbol{\phi}_t, \boldsymbol{\phi}_t'), \tag{40}$$

where $\mathcal{L}_t^{\mathrm{outer}}$ is defined in Eq. (16). Forward and Full use the same MetaNet architecture, so their comparison tests the episodic update. All variants with a fast pathway include bias; its individual contribution is not tested here.