# SwiTTA: Switching Domain Experts and Aggregating Contextual Features Towards Realistic Test-Time Adaptation

**Anonymous Authors**[1]

## Abstract

The adaptability of test-time adaptation is influenced by multiple real-world factors, including continual domain shifts and temporally correlated/imbalanced distributions. To address this, we propose a general **SwiTTA** framework for both CNNs and ViTs, featuring two key components: (1) a *domain router with multiple domain experts* performing online domain identification via feature statistics analysis, and (2) *CFA* - a temporal correlation handler employing contextual feature aggregation through sliding window averaging. Extensive experiments demonstrate that SwiTTA achieves state-of-the-art performance across diverse realistic scenarios, outperforming existing methods by significant margins.

## 1. Introduction

Test-Time Adaptation (TTA) (Iwasawa & Matsuo, 2021; Bateson et al., 2022; Gandelsman et al., 2022; Niu et al., 2022; Zhou et al., 2023; Guo et al., 2024) has emerged as a promising solution, typically addressing domain shifts through normalization statistics calibration (Ioffe & Szegedy, 2015; Nado et al., 2020; Schneider et al., 2020; Wang et al., 2021) or transformer feature alignment (Lian et al., 2022; Kojima et al., 2022) to update the model parameters (BN/LN layers (Kojima et al., 2022; Yang et al., 2024), prompt tuning (Gan et al., 2023), and LoRA fine-tuning (Liu et al., 2024b). Early approaches (Nado et al., 2020; Schneider et al., 2020; Wang et al., 2021) assumed i.i.d. test data, while recent extensions handle more complex scenarios: continual shifts (Wang et al., 2022), mixed domains (Marsden et al., 2024; Tomar et al., 2024; Niu et al., 2023), temporal correlations (Boudiaf et al., 2022; Gong et al., 2022; Yuan et al., 2023), class imbalances (Niu et al., 2023; Su et al., 2024), and unified benchmark for above settings (Du et al., 2024). However, above methods are mainly designed for single scenario or single network structure (CNN or ViT).

In this work, we propose a general and versatile SwiTTA framework across different network structures and realistic TTA scenarios. Our primary insight lies in dynamically managing multiple domain experts and switching between them based on the domain router's decisions. The domain router is a general-purpose module that can be implemented using various approaches, such as training it by simulating data from different domains. However, such methods may constrain the generalization capability of the domain router and can potentially lead to the leakage of test data information. One of our key findings is that the domain of data can be effectively represented by the statistical properties of features, enabling the domain router to perform online clustering in an unsupervised manner, dynamically initializing new experts when detecting novel domains. As illustrated in Fig. 1, compared to established paradigms for single or continual domain adaptation without using multiple domain experts, the SwiTTA framework employed the domain router with multiple domain experts, achieving a significant performance improvement.

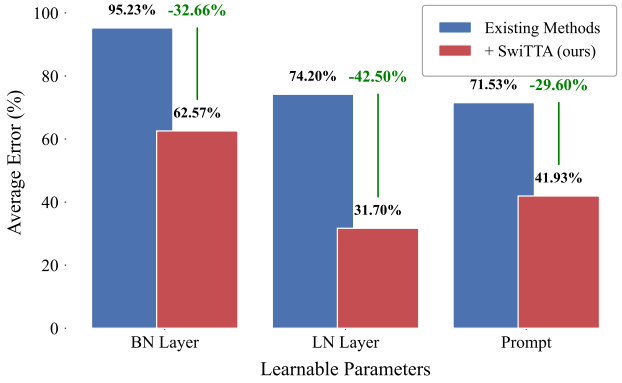


*Figure 1.* The SwiTTA framework, as a general-purpose method, achieves significant improvements across various adaptation paradigms on ImageNet-C under realistic TTA settings.

Additionally, to address potential temporal correlations in class distributions, we exploit it by aggregating the features of previous samples, resulting in an effective and efficient approach named CFA (Contextual Feature Aggregation), which does not require any modifications to model parameters. Specifically, by maintaining a context window (size $C$) of historical features, CFA predicts the current sample by averaging the features of the current and previous samples $(0, \cdots, C-1)$ and selecting the prediction with the highest

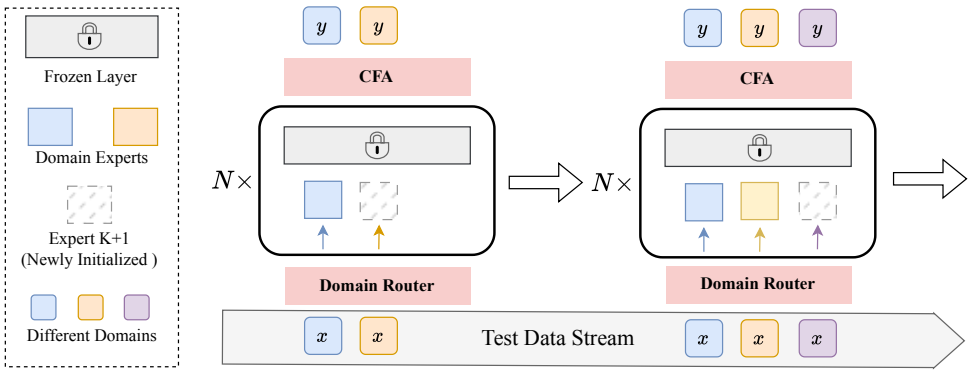


*Figure 2.* The overall architecture of the SwiTTA framework. The input sample is first processed through the Domain Router to select an appropriate domain expert or initialize a new one. Following feature extraction, the Contextual Feature Aggregation (CFA) module addresses potentially temporal class dependencies to generate final prediction labels.

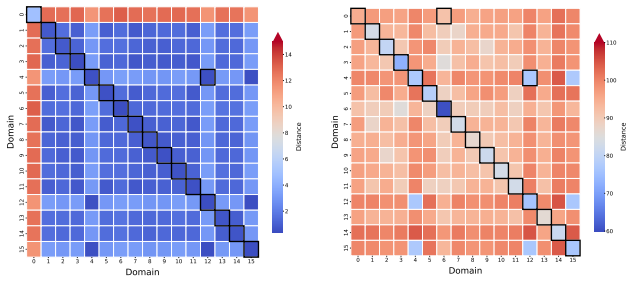


*Figure 3.* Intra/inter-domain distance matrix of feature statistics across 16 domains on ImageNet-C (index 0: source domain, left: ResNet-50, right: ViT-B/16). Diagonal elements (intra-domain) and minimum-distance domains are highlighted in bold.

confidence. CFA enables efficient parallel computation via matrix multiplication, introducing virtually no additional time or memory overhead.

## 2. Method

Our framework addresses dynamic test-time adaptation under multi-domain and class shifts through two synergistic components. The **Domain Router** dynamically evaluates input distribution characteristics to activate corresponding domain experts or initialize new domain-specific experts when encountering unseen patterns. Complementary to this, the **Contextual Feature Aggregation (CFA)** module explicitly aggregates contextual information from previous samples, effectively handling correlated class distributions.

### 2.1. Domain Router and Expert Adaptation

As discussed above, we have only defined the functionality of the Domain Router while maintaining implementation flexibility. For instance, practitioners could train a lightweight classifier for domain identification, though this would require additional annotated training data. Our fundamental contribution stems from the empirical observation that domain shifts can be reliably detected through feature distribution statistics without requiring supervised training. To validate this hypothesis, we initially computed per-domain feature statistics and implemented domain prediction through Wasserstein distance measurements between sample features and domain-specific statistics. As demonstrated in Tab. 1 and Fig. 3, our experimental results confirm three critical insights: (1) Feature statistics contain sufficient discriminative information for unsupervised domain identification; (2) Wasserstein distance serves as an effective metric for quantifying inter-domain relationships; (3) Our methodology demonstrates architecture-agnostic efficacy across both convolutional and transformer-based frameworks.

*Table 1.* Domain identification accuracy of feature statistics.

| Model | Acc (%) |
|---|---|
| ResNet-50 | 87.13 |
| ViT-B/16 | 86.46 |

**Domain Router Implementation.** Building upon these foundations, we propose a feature statistics-driven domain routing mechanism. Through an *auxiliary forward pass*, we extract feature statistics for each domain and maintain their corresponding domain statistics using an EMA updating strategy. Domain assignment is determined through Wasserstein distance of feature statistics between sample and existing domains, formally expressed as:

$$k^* = \begin{cases} \underset{1\le k\le K}{\operatorname{argmin}} D(x,k), & \text{if } \min\limits_k D(x,k) \le D(x,0) \\ K+1 \text{ (new domain)}, & \text{if } \min\limits_k D(x,k) > D(x,0), \end{cases} \tag{1}$$

where $D(x,0)$ represents the distance to the source domain and $K$ indicates the number of currently available domain experts. We specifically employ the closed-form Wasserstein distance between Gaussian distributions:

$$\begin{aligned} D(x,k) &= W_2^2(\mathcal{N}(\mu_x,\sigma_x),\mathcal{N}(\mu_k,\sigma_k)) \\ &= |\mu_x-\mu_k|^2 + |\sigma_x-\sigma_k|^2, \end{aligned} \tag{2}$$

where $(\mu_x,\sigma_x)$ and $(\mu_k,\sigma_k)$ are the mean and standard deviation of sample $x$ and domain $k$, respectively.

*Table 2.* **Average error (%) on ImageNet-C within the UniTTA benchmark.** $(\{i, n, 1\}, \{1, u\})$ denotes temporal correlation and imbalance settings, where $\{i, n, 1\}$ represent i.i.d., temporal correlated and continual, and $\{1, u\}$ represent balance and imbalance, respectively. The standard deviation of SwiTTA is shown in the last row.

| Class setting | i.i.d. and balanced (i,1) | | temporal correlated and balanced (n,1) | | | | | temporal correlated and imbalanced (n,u) | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Domain setting | (1,1) | (i,1) | (1,1) | (i,1) | (i,u) | (n,1) | (n,u) | (1,1) | (i,1) | (i,u) | (n,1) | (n,u) | Avg. |
| ResNet-50 | | | | | | | | | | | | | |
| ROID (Marsden et al., 2024) | **59.76** | 83.07 | 99.46 | 99.80 | 99.72 | 99.78 | 99.73 | 94.00 | 99.81 | 99.18 | 99.47 | 99.62 | 94.45 |
| TENT (Wang et al., 2021) | 61.98 | 84.07 | 94.16 | 97.75 | 96.74 | 98.21 | 97.26 | 83.01 | 96.46 | 91.58 | 92.38 | 95.93 | 90.79 |
| TRIBE (Su et al., 2024) | 79.65 | 84.16 | 91.95 | 96.22 | 93.43 | 97.41 | 95.79 | 67.22 | 93.54 | 89.29 | 91.94 | 95.23 | 89.65 |
| BN (Nado et al., 2020) | 68.54 | 82.22 | 89.98 | 92.93 | 92.71 | 92.94 | 93.07 | 82.74 | 89.85 | 89.24 | 89.16 | 89.17 | 87.71 |
| SAR (Niu et al., 2023) | 65.28 | 81.32 | 89.33 | 93.93 | 93.17 | 93.72 | 93.32 | 81.89 | 90.58 | 89.31 | 88.62 | 89.17 | 87.47 |
| CoTTA (Wang et al., 2022) | 62.04 | 79.33 | 90.18 | 94.17 | 93.21 | 94.17 | 93.51 | 80.68 | 89.98 | 88.14 | 88.02 | 89.01 | 86.87 |
| NOTE (Gong et al., 2022) | 81.90 | 83.91 | 83.05 | 85.37 | 83.49 | 86.64 | 84.42 | 80.77 | 82.81 | 81.64 | 82.44 | 81.99 | 83.20 |
| UnMIX-TNS (Tomar et al., 2024) | 77.19 | 85.62 | 77.47 | 84.30 | 83.54 | 84.38 | 83.25 | 79.27 | 83.61 | 84.48 | 84.34 | 82.04 | 82.46 |
| TEST | 81.98 | 81.90 | 82.09 | 82.02 | 82.43 | 82.17 | 81.74 | 81.08 | 81.49 | 81.59 | 81.40 | 81.27 | 81.77 |
| RoTTA (Yuan et al., 2023) | 67.77 | 79.91 | 71.72 | 80.54 | 79.65 | 80.30 | 79.63 | 68.74 | 78.26 | 77.94 | 79.78 | 78.36 | 76.88 |
| LAME (Boudiaf et al., 2022) | 82.76 | 82.12 | 74.77 | 73.44 | 74.90 | 74.27 | 74.10 | 75.30 | 75.06 | 75.45 | 75.04 | 74.59 | 75.98 |
| UniTTA (Du et al., 2024) | 78.07 | 78.00 | 70.25 | 66.83 | 66.42 | 68.29 | 68.05 | 72.02 | 65.68 | 66.87 | 68.48 | 67.58 | 69.71 |
| SwiTTA | 68.88 (±0.05) | **78.00** (±0.20) | **65.17** (±13.74) | **58.87** (±0.16) | **58.53** (±0.56) | **62.96** (±0.70) | **60.50** (±0.25) | **59.59** (±0.41) | **60.94** (±0.11) | **60.64** (±0.44) | **65.07** (±0.34) | **62.57** (±0.32) | **63.48** (-6.23) |
| ViT-B/16 | | | | | | | | | | | | | |
| TEST | 49.02 | 48.96 | 49.19 | 49.10 | 49.48 | 49.10 | 49.25 | 47.84 | 47.81 | 48.19 | 47.89 | 48.08 | 48.66 |
| FOA (Niu et al., 2024) | 42.14 | 44.07 | 43.24 | 43.80 | 44.15 | 44.12 | 43.59 | 41.78 | 41.74 | 41.58 | 42.30 | 42.12 | 42.88 |
| TENT (Wang et al., 2021) | 41.44 | 41.77 | 41.61 | 41.82 | 41.64 | 41.93 | 41.67 | 39.84 | 40.10 | 39.95 | 40.18 | 39.76 | 40.98 |
| SAR (Niu et al., 2023) | 38.72 | 39.12 | 38.40 | 39.25 | 39.10 | 39.24 | 39.28 | 35.90 | 37.01 | 36.83 | 37.08 | 36.71 | 38.05 |
| ViDA (Liu et al., 2024b) | 41.09 | 40.41 | 41.45 | 40.58 | 40.75 | 40.18 | 40.17 | 39.36 | 38.10 | 37.51 | 38.63 | 37.79 | 39.67 |
| LAME (Boudiaf et al., 2022) | 46.33 | 45.32 | 29.08 | 27.47 | 28.80 | 27.64 | 28.01 | 30.12 | 29.36 | 30.42 | 28.77 | 29.18 | 31.71 |
| SwiTTA | **37.87** (±1.45) | **30.96** (±3.00) | **17.73** (±2.63) | **12.52** (±0.24) | **12.76** (±0.80) | **13.18** (±1.81) | **18.15** (±0.32) | **22.10** (±1.11) | **16.24** (±1.90) | **16.37** (±1.58) | **18.56** (±2.11) | **16.35** (±4.49) | **19.40** (-12.31) |

**Expert Adaptation Strategy.** As illustrated in Fig. 1, our framework maintains modular independence between the domain router and expert adaptation components. For domain expert implementation, we adopt established paradigms effective for both single-target domain and continual domain shifts. Expert updating follows architecture-specific conventions: For CNN architectures, we maintain Batch Normalization statistics through EMA updates, while for ViTs, we formulate an optimization objective minimizing Wasserstein distance between source domain statistics and current sample statistics. This divergence metric guides stochastic gradient-based updates for Layer Normalization parameters and prompt embeddings. Comprehensive update rules and implementation specifics are detailed in App. B.1.

### 2.2. Contextual Feature Aggregation (CFA)

To address potentially temporal correlations in streaming data, we propose CFA which enhances predictions by adaptively aggregating historical context. Specifically, given a context window size $C$ and feature of current sample $f_t$, let $\mathbf{F} = [f_{t-C+1}, ..., f_t]^\top \in \mathbb{R}^{C\times d}$ represent the context window of features from preceding samples. The correlated prediction is computed through parallelizable operations:

$$\mathbf{M} = \begin{bmatrix} 1/C & 1/C & \cdots & 1/C \\ 0 & 1/(C-1) & \cdots & 1/(C-1) \\ \vdots & \vdots & \ddots & 1/2 \\ 0 & 0 & \cdots & 1 \end{bmatrix}, \quad (3)$$

$$\hat{\mathbf{Y}} = \text{softmax}\left(\frac{1}{C}\mathbf{M}\mathbf{F}\mathbf{W}^\top + \mathbf{1}\mathbf{b}^\top\right), \quad (4)$$

where $\mathbf{W}$ and $\mathbf{b}$ are the weights and bias of the linear classifier and $\mathbf{1}$ is a vector of ones. The CFA simultaneously computes predictions under varying context window size and selects the optimal window yielding the maximum prediction confidence:

$$y_t = \text{argmax}\left(\max_{0\le c\le C-1} \hat{\mathbf{Y}}[c,:]\right). \quad (5)$$

As evident from Eq. (4), the linear formulation enables efficient parallel computation at the logit level. Furthermore, CFA supports batch-wise parallel inference through matrix extension of $M$. See App. B.2 for implementation details.

## 3. Experiments

In this section, we present the results on unified realistic test-time adaptation benchmark – UniTTA (Du et al., 2024). For detailed experimental setup, please refer to App. C.

### 3.1. Main results

We first compare the robustness of different methods across various models and settings for classification tasks. Notably, multiple approaches exhibit inferior performance relative to the vanilla R50 backbone baseline, underscoring the necessity for comprehensive benchmarking. As demonstrated in Tab. 2 and App. F, our method surpasses existing approaches on average across all datasets, demonstrating consistent superiority particularly in realistic application scenarios. Notably, for the ViT-B/16 architecture, our approach achieves the highest performance across all settings.

Furthermore, Tab. 3 presents a comprehensive evaluation

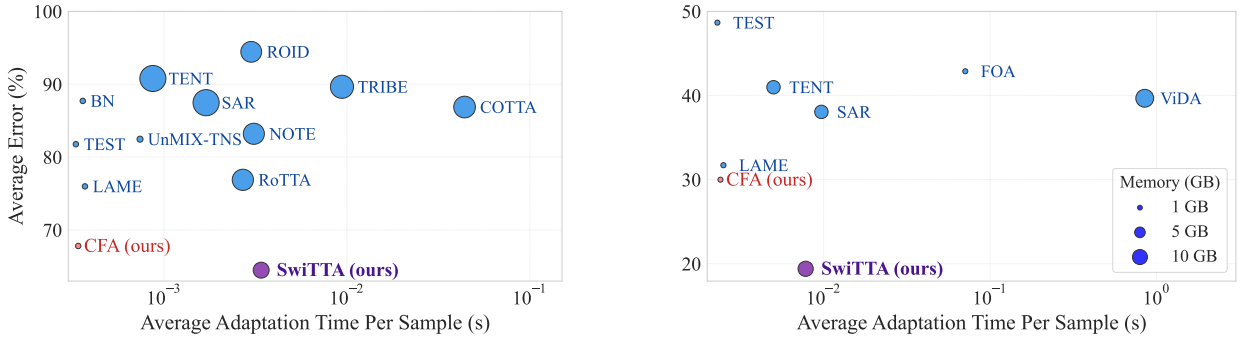


*Figure 4.* Three-way comparative analysis of (a) adaptation time, (b) GPU memory consumption, and (c) classification accuracy across different methods (left: ResNet-50, right: ViT-B/16).

*Table 3.* **Average semantic segmentation mIoU (%) on Cityscapes-to-ACDC within the UniTTA benchmark.**

| Domain setting | (i,1) | (n,1) | (1,1) | (i,u) | (n,u) | (1,u) | Time (s/sample) | Parameter | Avg. |
|---|---|---|---|---|---|---|---|---|---|
| TEST | 56.72 | 56.59 | 56.71 | 56.83 | 56.83 | 56.83 | 3.75 | – | 56.75 |
| TENT (Wang et al., 2021) | 54.68 | 54.54 | 54.65 | 55.86 | 55.84 | 53.52 | 5.04 | 0.09 M | 54.85 |
| CoTTA (Wang et al., 2022) | 59.13 | 58.86 | 59.11 | 59.44 | 59.48 | 59.83 | 7.35 | 84.61 M | 59.31 |
| BECoTTA (Lee et al., 2024) | 59.34 | 59.20 | 59.18 | 59.45 | 59.42 | 59.36 | 5.66 | 0.09 M | 59.33 |
| SwiTTA | **60.48** | **60.32** | **60.60** | **60.66** | **60.62** | **60.52** | **3.86** | **0.09 M** | **60.53** (+1.20) |

*Table 4.* Ablation study of different components. The average of 12 settings are reported on CIFAR10-C, CIFAR100-C, and ImageNet-C (ResNet-50 and ViT-B/16).

| | C10-C | C100-C | IN-C (R50) | IN-C (ViT) |
|---|---|---|---|---|
| TEST | 42.12 | 46.20 | 81.77 | 48.66 |
| CFA | 31.61 | 28.58 | 67.80 | 29.99 |
| Domain Router | 26.56 | 41.27 | 76.75 | 31.89 |
| SwiTTA | **16.61** | **23.77** | **63.48** | **19.40** |

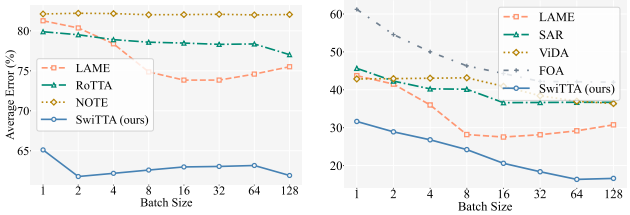


*Figure 5.* Sensitive analysis of **batch size** with ResNet-50 (left) and ViT-B/16 (right).

on semantic segmentation tasks. Compared to BECoTTA, which also employs multi-domain experts, our method delivers superior performance under identical parameters across all settings while achieving lower inference time.

**Time Memory Efficiency.** We present comparative analysis of our method against existing approaches across three key dimensions - adaptation time, GPU memory consumption, and accuracy - in Fig. 4. Among lightweight methods (e.g., BN and LAME), CFA achieves optimal performance while maintaining inference speed comparable to vanilla test-time processing. Furthermore, the integration of CFA with our proposed domain router enables SwiTTA to achieve the SOTA performance while attaining an enhanced balance between memory and efficiency.

### 3.2. Analysis

For all experiments in this section, unless specified, the default test dataset is ImageNet-C, and the default test setting is correlated. and imbalanced for both domain and class.

**Ablation Study.** This section presents the overall results, while comprehensive results are available in App. F. The results demonstrate that both core components independently enhance model performance, with their synergistic combination yielding superior outcomes.

**Hyperparameter Sensitivity.** Fig. 5 reports the performance comparison of several competitive methods under varying batch sizes. Our approach demonstrates superior performance across different configurations even when batch size is set to 1, highlighting its practical reliability.

## 4. Conclusion

In this work, we introduce a general-purpose SwiTTA framework that simultaneously addresses domain and class distribution shifts and temporal correlations. SwiTTA dynamically switches between domain experts or initializes new domain-specific experts when encountering new domain patterns. Moreover, the SwiTTA integrates with a CFA module to handle temporal class correlations. Empirical evidence from the UniTTA benchmark demonstrates that SwiTTA framework outperforms existing methods in various realistic TTA scenarios.

## Impact Statements

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

## References


Bateson, M., Lombaert, H., and Ben Ayed, I. Test-time adaptation with shape moments for image segmentation. In *ICMICCAI*, 2022.

Boudiaf, M., Mueller, R., Ben Ayed, I., and Bertinetto, L. Parameter-free online test-time adaptation. In *CVPR*, 2022.

Brahma, D. and Rai, P. A probabilistic framework for lifelong test-time adaptation. In *CVPR*, 2023.

Choi, S., Yang, S., Choi, S., and Yun, S. Improving test-time adaptation via shift-agnostic weight regularization and nearest source prototypes. In *ECCV*, 2022.

Du, C., Wang, Y., Guo, J., Han, Y., Zhou, J., and Huang, G. UniTTA: Unified Benchmark and Versatile Framework Towards Realistic Test-Time Adaptation. *arXiv preprint arXiv:2407.20080*, 2024.

Gan, Y., Bai, Y., Lou, Y., Ma, X., Zhang, R., Shi, N., and Luo, L. Decorate the newcomers: Visual domain prompt for continual test time adaptation. In *AAAI*, 2023.

Gandelsman, Y., Sun, Y., Chen, X., and Efros, A. Test-time training with masked autoencoders. In *NeurIPS*, 2022.

Gong, T., Jeong, J., Kim, T., Kim, Y., Shin, J., and Lee, S.-J. Note: Robust continual test-time adaptation against temporal correlation. In *NeurIPS*, 2022.

Grandvalet, Y. and Bengio, Y. Semi-supervised learning by entropy minimization. In *NeurIPS*, 2004.

Guo, J., Zhao, J., Ge, C., Du, C., Ni, Z., Song, S., Shi, H., and Huang, G. Everything to the synthetic: Diffusion-driven test-time adaptation via synthetic-domain alignment. *arXiv preprint arXiv:2406.04295*, 2024.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *CVPR*, 2016.

Hendrycks, D. and Dietterich, T. Benchmarking neural network robustness to common corruptions and perturbations. In *ICLR*, 2019.

Huang, G. and Du, C. The high separation probability assumption for semi-supervised learning. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 52 (12):7561–7573, 2022.

Ioffe, S. and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.

Iwasawa, Y. and Matsuo, Y. Test-time classifier adjustment module for model-agnostic domain generalization. In *NeurIPS*, 2021.

Jung, S., Lee, J., Kim, N., Shaban, A., Boots, B., and Choo, J. Cafa: Class-aware feature alignment for test-time adaptation. In *ICCV*, 2023.

Kojima, T., Matsuo, Y., and Iwasawa, Y. Robustifying vision transformer without retraining from scratch by test-time class-conditional feature alignment. *IJCAI*, 2022.

Lee, D., Yoon, J., and Hwang, S. J. Becotta: Input-dependent online blending of experts for continual test-time adaptation. *ICML*, 2024.

Lian, D., Zhou, D., Feng, J., and Wang, X. Scaling & shifting your features: A new baseline for efficient model tuning. *NeurIPS*, 35:109–123, 2022.

Lim, H., Kim, B., Choo, J., and Choi, S. Ttn: A domain-shift aware batch normalization in test-time adaptation. In *ICLR*, 2023.

Liu, J., Xu, R., Yang, S., Zhang, R., Zhang, Q., Chen, Z., Guo, Y., and Zhang, S. Continual-mae: Adaptive distribution masked autoencoders for continual test-time adaptation. In *CVPR*, pp. 28653–28663, 2024a.

Liu, J., Yang, S., Jia, P., Lu, M., Guo, Y., Xue, W., and Zhang, S. Vida: Homeostatic visual domain adapter for continual test time adaptation. *ICLR24*, 2024b.

Marsden, R. A., Döbler, M., and Yang, B. Universal test-time adaptation through weight ensembling, diversity weighting, and prior correction. In *WACV*, 2024.

Mirza, M. J., Micorek, J., Possegger, H., and Bischof, H. The norm must go on: Dynamic unsupervised domain adaptation by normalization. In *CVPR*, 2022.

Nado, Z., Padhy, S., Sculley, D., D'Amour, A., Lakshminarayanan, B., and Snoek, J. Evaluating prediction-time batch normalization for robustness under covariate shift. *CoRR*, 2020.

Niu, S., Wu, J., Zhang, Y., Chen, Y., Zheng, S., Zhao, P., and Tan, M. Efficient test-time model adaptation without forgetting. In *ICML*, 2022. URL https://proceedings.mlr.press/v162/niu22a.html.

Niu, S., Wu, J., Zhang, Y., Wen, Z., Chen, Y., Zhao, P., and Tan, M. Towards stable test-time adaptation in dynamic wild world. In *ICLR*, 2023.

Niu, S., Miao, C., Chen, G., Wu, P., and Zhao, P. Test-time model adaptation with only forward passes. In *The International Conference on Machine Learning*, 2024.

Schneider, S., Rusak, E., Eck, L., Bringmann, O., Brendel, W., and Bethge, M. Improving robustness against common corruptions by covariate shift adaptation. In *NeurIPS*, 2020.

Su, Y., Xu, X., and Jia, K. Towards real-world test-time adaptation: Tri-net self-training with balanced normalization. In *AAAI*, 2024.

Sun, Y., Wang, X., Liu, Z., Miller, J., Efros, A., and Hardt, M. Test-time training with self-supervision for generalization under distribution shifts. In *ICML*, 2020.

Tomar, D., Vray, G., Thiran, J.-P., and Bozorgtabar, B. Unmixing test-time normalization statistics: Combatting label temporal correlation. In *ICLR*, 2024.

Wang, D., Shelhamer, E., Liu, S., Olshausen, B., and Darrell, T. Tent: Fully test-time adaptation by entropy minimization. In *ICLR*, 2021. URL https://openreview.net/forum?id=uXl3bZLkr3c.

Wang, Q., Fink, O., Van Gool, L., and Dai, D. Continual test-time domain adaptation. In *CVPR*, 2022.

Xie, S., Girshick, R., Dollár, P., Tu, Z., and He, K. Aggregated residual transformations for deep neural networks. In *CVPR*, 2017.

Yang, X., Chen, X., Li, M., Wei, K., and Deng, C. A versatile framework for continual test-time domain adaptation: Balancing discriminability and generalizability. In *CVPR*, pp. 23731–23740, June 2024.

Yuan, L., Xie, B., and Li, S. Robust test-time adaptation in dynamic scenarios. In *CVPR*, 2023.

Zagoruyko, S. and Komodakis, N. Wide residual networks. In *British Machine Vision Conference*, 2016.

Zhang, Y., Mehra, A., and Hamm, J. Dynamic domains, dynamic solutions: Dpcore for continual test-time adaptation. *arXiv preprint arXiv:2406.10737*, 2024.

Zhou, Z., Guo, L.-Z., Jia, L.-H., Zhang, D., and Li, Y.-F. ODS: Test-time adaptation in the presence of open-world data shift. In *ICML*, 2023.

Zou, Y., Zhang, Z., Li, C.-L., Zhang, H., Pfister, T., and Huang, J.-B. Learning instance-specific adaptation for cross-domain segmentation. In *ECCV*, 2022.

# A. Related Work

**Test-Time Adaptation (TTA)** addresses distributional shifts in test data without requiring additional data acquisition or labeling. Sun et al.(Sun et al., 2020) propose an on-the-fly adaptation method using an auxiliary self-supervised task. Subsequent TTA algorithms (Nado et al., 2020; Schneider et al., 2020; Wang et al., 2021) leverage batches of test samples to recalibrate Batch Normalization (BN) layers(Ioffe & Szegedy, 2015) using test data. TENT (Wang et al., 2021) refines this approach by adapting a pre-trained model through entropy minimization (Grandvalet & Bengio, 2004), updating a few trainable parameters in BN layers.

**Realistic Test-Time Adaptation.** Recent advances in Test-Time Adaptation (TTA) research have focused on more practical scenarios that account for realistic distribution shifts in test data. These investigations primarily address two critical challenges: domain distribution shift (Wang et al., 2022; Brahma & Rai, 2023; Niu et al., 2023; Zhang et al., 2024; Lee et al., 2024) and temporal correlation (Boudiaf et al., 2022; Gong et al., 2022), with some works exploring their combined effects (Yuan et al., 2023; Marsden et al., 2024; Su et al., 2024; Tomar et al., 2024). Current methodological approaches can be categorized into three main paradigms: (1) self-training frameworks (Wang et al., 2022; Yuan et al., 2023; Brahma & Rai, 2023; Liu et al., 2024a) that incorporate semi-supervised learning techniques (Huang & Du, 2022), (2) parameter-free methods (Boudiaf et al., 2022) leveraging Laplacian regularization, and (3) Batch Normalization (BN) recalibration strategies (Gong et al., 2022; Mirza et al., 2022; Zou et al., 2022; Yuan et al., 2023; Tomar et al., 2024; Sun et al., 2020). The most closely related approach to our method is BECoTTA (Lee et al., 2024). However, its domain router employs a classifier pre-trained on simulated data, which exhibits limited generalization capability.

# B. Implementation Details

## B.1. Domain Experts Update Rules

For CNN, we implement the Domain Expert using Balanced BN following TRIBE (Su et al., 2024). For ViT, we employ LN layers as domain experts and utilize feature alignment with entropy minimization loss (Wang et al., 2021) for stochastic gradient descent optimization (Kojima et al., 2022; Lian et al., 2022; Niu et al., 2024). The optimization objective aligns sample feature distributions with source domain distributions. Specifically, given a sample processed by the domain router with predicted domain $d$, the feature alignment loss is computed as:

$$\begin{aligned}\mathcal{L}(x) = D(s,d) &= W_2^2(\mathcal{N}(\mu_s,\sigma_s),\mathcal{N}(\mu_d,\sigma_d)) \\ &= |\mu_s-\mu_d|^2 + |\sigma_s-\sigma_d|^2\end{aligned} \tag{6}$$

with domain statistics updated via:

$$\begin{aligned}\mu_d &= \alpha\mu_d + (1-\alpha)\mu_x \\ \sigma_d &= \alpha\sigma_d + (1-\alpha)\sigma_x,\end{aligned} \tag{7}$$

where $\mu_d, \sigma_d$ are maintained via EMA with momentum coefficient $\alpha = 0.99$, and $\mu_x, \sigma_x$ denote instance statistics. The source domain statistics $\mu_s, \sigma_s$ are precomputed before deployment, which aligns with contemporary domain adaptation methods (Lim et al., 2023; Jung et al., 2023; Choi et al., 2022; Niu et al., 2022; 2024).

The entropy minimization loss and feature alignment loss are weighted by coefficients 3 and 60 respectively, optimized using SGD with learning rate $5 \times 10^{-3}$. Domain statistics in the router follow the same update rules as Eq. (7) but with momentum coefficient $\alpha = 0.95$.

## B.2. Batch-wise Parallel Implementation of CFA

Given a context window size $C$, batch size $B$, and historical logits $\mathbf{L}_{hist} \in \mathbb{R}^{K\times d}$, we construct adaptive averaging masks $\mathbf{M}^{(c)}{}_{c=1}^{C}$ where each $\mathbf{M}^{(c)} \in \mathbb{R}^{(C+B)\times(C+B)}$ contains normalized averaging weights:

$$\mathbf{M}^{(c)}_{i,j} = \begin{cases} \frac{1}{\min(c,i+1)} & \text{if } j \in [\max(0, i-c+1), i] \\ 0 & \text{otherwise} \end{cases}. \tag{8}$$

For incoming logits $\mathbf{L}_t \in \mathbb{R}^{B\times K}$, we concatenate logits $\mathbf{L}_{cat} = [\mathbf{L}_{hist}; \mathbf{L}_t]$ and compute context-aware predictions through parallelizable masked projections:

$$\mathbf{H} = \left[\mathbf{M}^{(1)}\mathbf{L}_{cat}, ..., \mathbf{M}^{(C)}\mathbf{L}_{cat}\right] \in \mathbb{R}^{(C+B)\times C\times K}, \tag{9}$$

$$\hat{\mathbf{Y}} = \text{softmax}\left(\mathbf{H}\right) \in \mathbb{R}^{(C+B)\times C\times K}, \tag{10}$$

For each sample $x_i$ in current batch, we select the optimal context length $c_i$ that maximizes prediction confidence:

$$c_i = \underset{1\le c\le C}{\arg\max} \max_{1\le k\le K} \hat{\mathbf{Y}}[i, c, k]. \tag{11}$$

The final prediction combines probabilities from optimal contexts:

$$y_i = \underset{1\le k\le K}{\arg\max}\, \hat{\mathbf{Y}}[i, c_i, k] \tag{12}$$

This implementation maintains an efficient $O(C)$ memory buffer while enabling robust temporal adaptation through confidence-based context selection.

## C. Experimental Setup

For classification tasks, we conduct experiments on three test-time adaptation benchmarks: CIFAR10-C (Hendrycks & Dietterich, 2019), CIFAR100-C (Hendrycks & Dietterich, 2019), and ImageNet-C (Hendrycks & Dietterich, 2019). Each dataset contains 15 distinct corruption types with 5 severity levels. All evaluations are performed under the most severe corruption level (level 5). Following established protocols (Wang et al., 2021; 2022; Yuan et al., 2023; Su et al., 2024), we utilize standard pre-trained architectures: WideResNet-28 (Zagoruyko & Komodakis, 2016) for CIFAR10-C, ResNeXt-29 (Xie et al., 2017) for CIFAR100-C, and ResNet-50 (He et al., 2016) for ImageNet-C. A consistent batch size of 64 is maintained across all datasets.

For semantic segmentation tasks, we adopt the Cityscapes-ACDC configuration described in (Wang et al., 2022; Lee et al., 2024). Specifically, we employ SegFormer-B5 as the pre-trained model with the Adam optimizer configured with a learning rate of 6e-5. In segmentation experiments, we exclusively utilize the domain router component without the CFA module, and only apply the feature alignment loss during adaptation.

In our SwiTTA framework implementation, we maintains a context window size of 16. Detailed parameter update rules for domain experts and the domain router are provided in App. B.1.

All comparative methods are implemented using their original optimizer configurations, learning rate schedules, and hyperparameters as specified in respective publications. All experiments are conducted on a single NVIDIA GeForce RTX 3090 GPU.

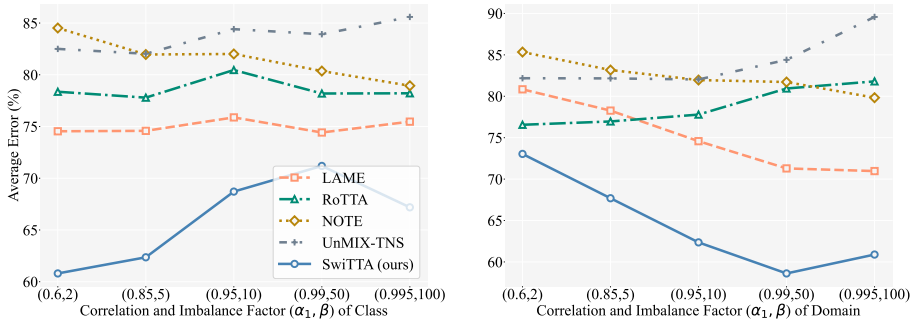


*Figure 6.* Average error (%) on ImageNet-C under various correlation and imbalance factors. Each experimental series maintains constant domain or class factors while modulating the complementary dimension.

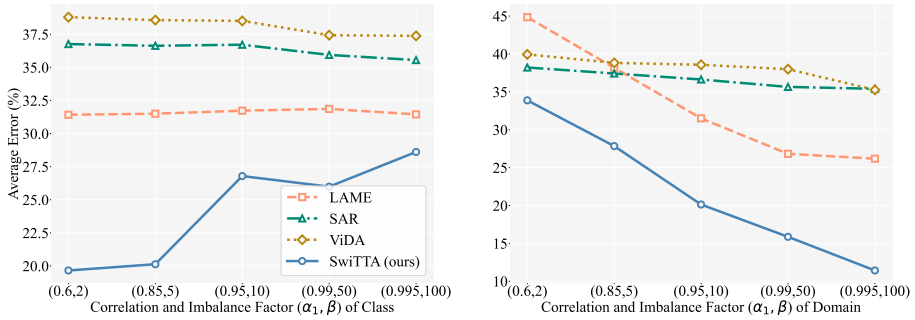


*Figure 7.* Average classification error (%) on ImageNet-C under varying correlation and imbalance factors. Each experimental series maintains constant domain or class factors while modulating the complementary dimension.

## D. More Results on UniTTA benchmark

We conduct experiments under various correlation and imbalance factors on ImageNet-C. Additional experiments are conducted under varying correlation and imbalance factors as shown in Figs. 6 and 7, where both domain and class distributions exhibit temporal correlated characteristics and imbalance patterns. Our experimental results demonstrate consistent robustness of the proposed method across diverse correlation coefficients and imbalance ratios.

Moreover, our scalability analysis reveals stable performance patterns across different data volumes. As depicted in Fig. 8, all compared methods exhibit convergence within hundreds of optimization steps, confirming the stability of our benchmark's distribution design.

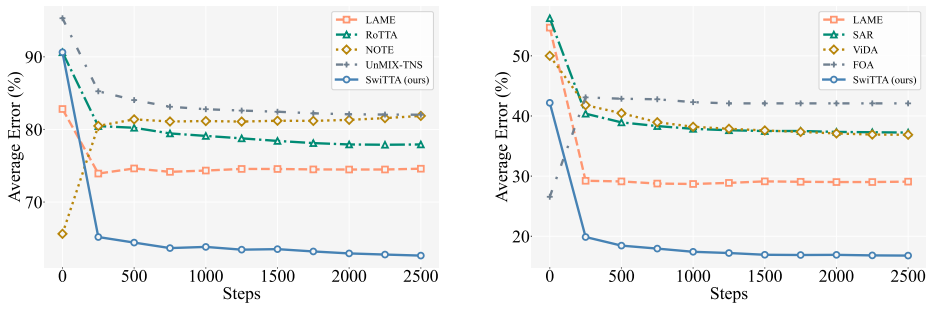


*Figure 8.* Sensitivity analysis of test data scaling effects on ImageNet-C using ResNet-50 (left) and ViT-B/16 (right). All evaluations employ correlated. imbalanced distributions for both domains and classes.

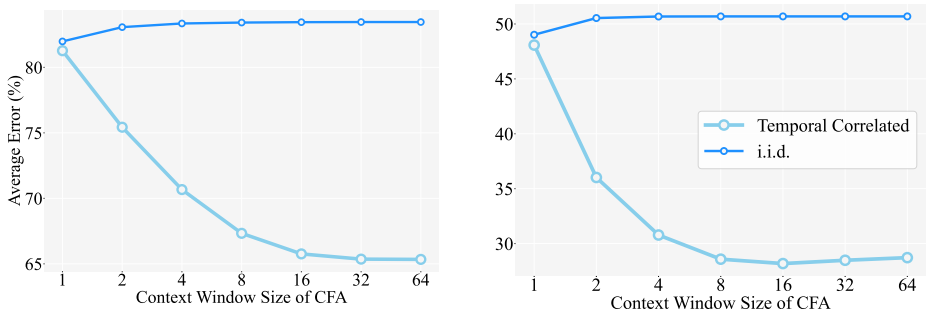


*Figure 9.* Sensitive analysis of the **context window size** with ResNet-50 (left) and ViT-B/16 (right).

## E. Extended Analysis

Unless explicitly stated otherwise, all experiments in this section utilize ImageNet-C as the default test dataset under temporal correlated and imbalanced conditions for both domain and class distributions.

### E.1. Hyperparameter Sensitivity

Fig. 9 illustrates the impact of context window size in both i.i.d and correlated settings. Under correlated settings, CFA demonstrates significant performance improvement with increasing context window size until reaching stability between 16 and 32, achieved through effective aggregation of contextual information. This phenomenon relates to the Mean Sojourn Time in Markov chains, which represents the average duration a process remains in a particular state before transitioning. In discrete-time Markov chains, the sojourn time follows a geometric distribution with expectation $\frac{1}{1-p}$, where $p$ denotes the staying probability in the current state. Our experiments empirically set $p = 0.95$ for correlated setting, yielding an optimal theoretical context window size of 20 (through $1/(1 - 0.95)$), which aligns with experimental observations. For i.i.d settings, CFA maintains robust performance across varying window sizes, demonstrating its capability to enhance performance of correlated setting while maintaining comparable performance to i.i.d settings.

*Table 5.* Average error (%) within the UniTTA benchmark and Number of domain experts initialized by domain router on CIFAR10-C. "w/ clean data" refers to the test data obtained by randomly replacing one domain with the original test data. Clean data does not significantly impact the number of domain experts The results include the count of all experts, even if only one sample is assigned to it.

| Class setting | i.i.d. and balanced (i,1) | | correlated and balanced (n,1) | | | | | correlated and imbalanced (n,u) | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Domain setting | (1,1) | (i,1) | (1,1) | (i,1) | (i,u) | (n,1) | (n,u) | (1,1) | (i,1) | (i,u) | (n,1) | (n,u) | |
| Corresponding setting | CoTTA | ROID | RoTTA | – | – | – | – | TRIBE | – | – | – | – | Avg. |
| Domain Router | 21.24 | 27.85 | 21.71 | 28.48 | 24.66 | 28.77 | 25.56 | 23.75 | 30.93 | 27.66 | 30.65 | 27.47 | 26.56 |
| w/ clean data | 20.94 | 27.90 | 21.83 | 28.65 | 25.60 | 28.37 | 25.73 | 22.62 | 30.03 | 26.46 | 30.44 | 27.09 | 26.30 |
| Domain Router | 204 | 19 | 182 | 20 | 24 | 20 | 27 | 218 | 14 | 18 | 13 | 22 | |
| w/ clean data | 191 | 20 | 159 | 19 | 28 | 19 | 23 | 211 | 19 | 30 | 16 | 24 | |

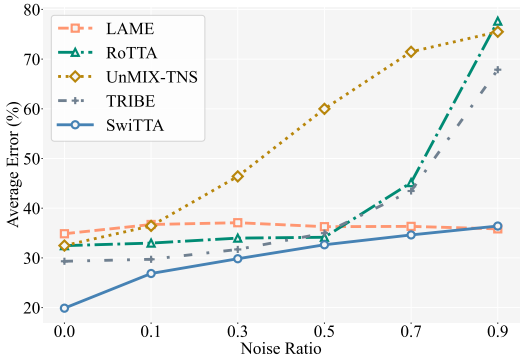


*Figure 10.* Robustness analysis of noisy data streams. The X-axis represents the proportion of data in CIFAR10-C that is randomly replaced with pure Gaussian noise.

## E.2. Impact of Clean and Noisy Data

We rigorously evaluate our framework's robustness through controlled contamination experiments. As presented in Tab. 5, introducing clean data minimally impacts domain expert initialization, demonstrating remarkable noise resilience. Further analysis in Fig. 10 confirms superior performance under increasing Gaussian noise contamination.

In Fig. 11, we further evaluate the anti-forgetting capabilities of different methods by re-testing on the source domain after completing target domain adaptation. Our approach maintains superior performance in target domain adaptation without significant degradation in catastrophic forgetting resistance.

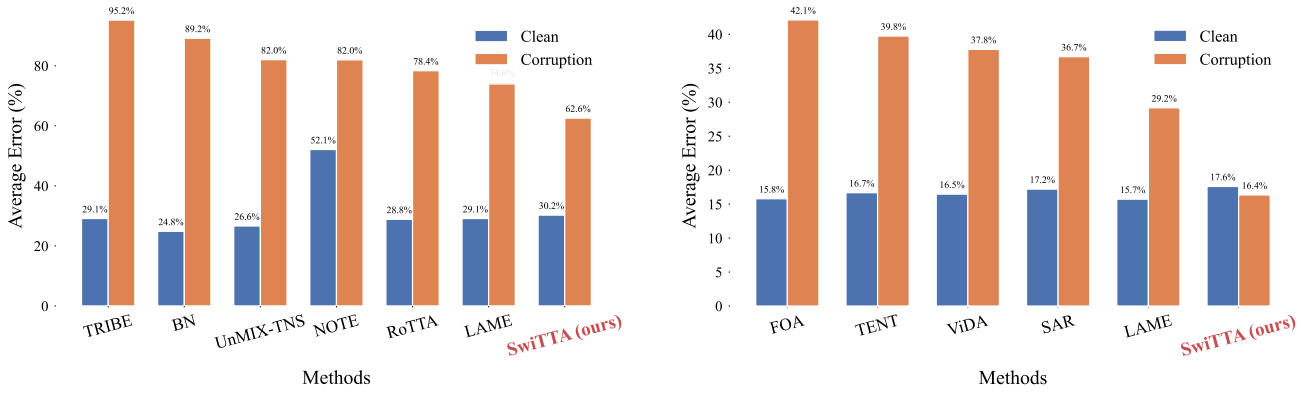


*Figure 11.* Evaluation of forgetting resistance across different methods (Left: ResNet-50; Right: ViT-B/16)

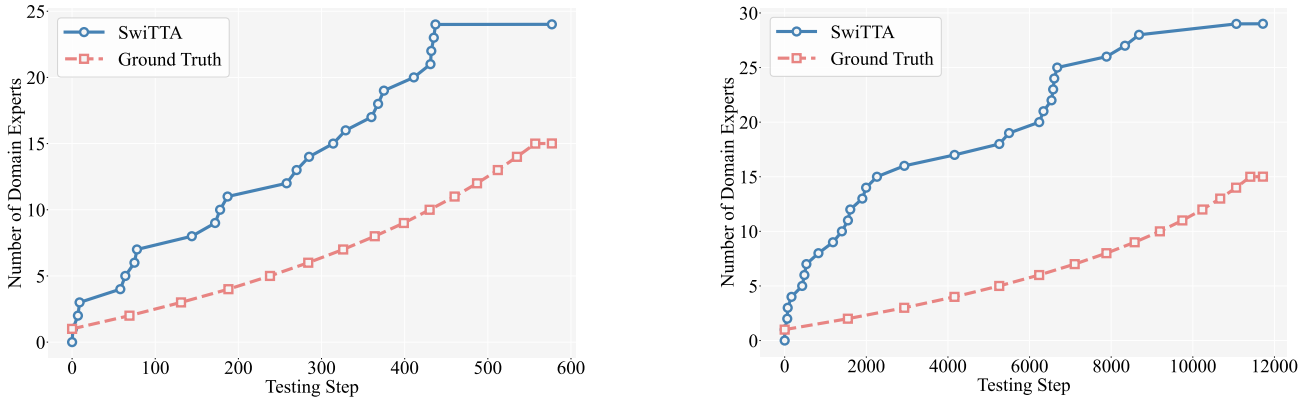


*Figure 12.* Visualization of domain router. The domain router dynamically predicts the domain of each sample based on the domain-wise statistics. Only domains with more than 100 samples are counted.

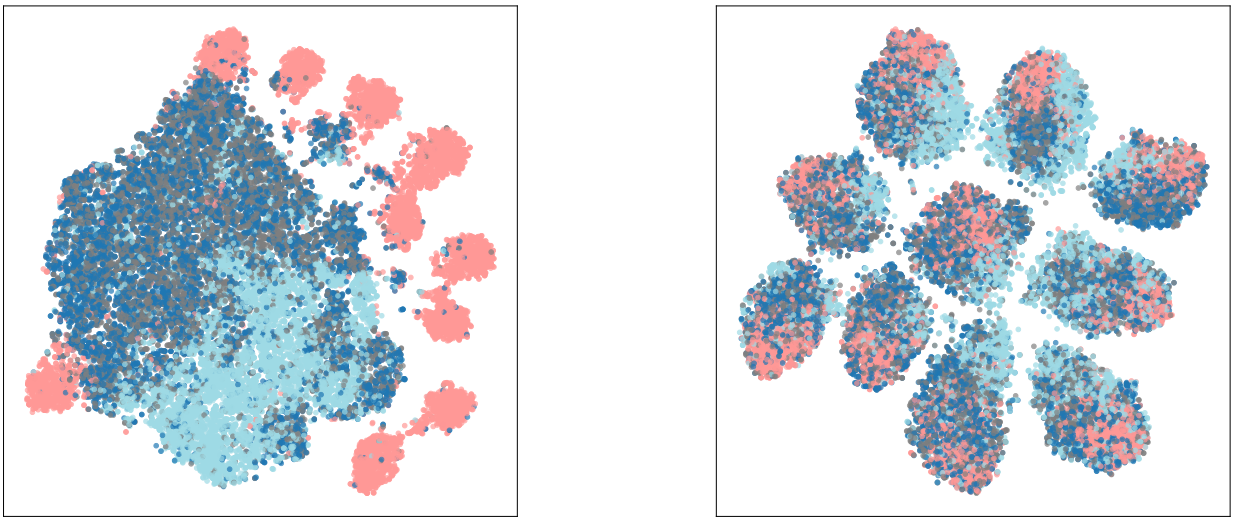

*Figure 13.* The t-SNE visualizations of vanilla TEST (Left) and our SwiTTA framework (Right). Pink points are source test data and others of different colors are target data (CIFAR10-C). 3 target domains are randomly selected.

### E.3. Visualization

We also visualize the domain router in Fig. 12. The process demonstrates that the domain router effectively captures the domain information and dynamically expands domains, which is crucial for accurate domain prediction.

We provide a t-SNE visualization of the feature distribution in Fig. 13. Each class is represented as a distinct cluster, with different domains distinguished by various colors. Our results demonstrate that SwiTTA enhances the network's ability to achieve highly discriminable representations.

# F. Results on All 24 Settings of UniTTA benchmark

*Table 6.* **Average error (%) on ImageNet-C within the UniTTA benchmark.** $(\{i, n, 1\}, \{1, u\})$ denotes correlation and imbalance settings, where $\{i, n, 1\}$ represent i.i.d., correlated and continual, respectively, and $\{1, u\}$ represent balance and imbalance, respectively.

| Class setting | correlated and balanced (n,1) | | | | | | correlated and imbalanced (n,u) | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Domain setting | (1,1) | (i,1) | (i,u) | (n,1) | (n,u) | (1,u) | (1,1) | (i,1) | (i,u) | (n,1) | (n,u) | (1,u) |
| ResNet-50 | | | | | | | | | | | | |
| ROID (Marsden et al., 2024) | 99.46 | 99.80 | 99.72 | 99.78 | 99.73 | 98.90 | 94.00 | 99.81 | 99.18 | 99.47 | 99.62 | 88.75 |
| TENT (Wang et al., 2021) | 94.16 | 97.75 | 96.74 | 98.21 | 97.26 | 91.90 | 83.01 | 96.46 | 91.58 | 92.38 | 95.93 | 82.36 |
| TRIBE (Su et al., 2024) | 91.95 | 96.22 | 93.43 | 97.41 | 95.79 | 88.49 | 67.22 | 93.54 | 89.29 | 91.94 | 95.23 | 67.71 |
| BN (Nado et al., 2020) | 89.98 | 92.93 | 92.71 | 92.94 | 93.07 | 90.34 | 82.74 | 89.85 | 89.24 | 89.16 | 89.17 | 83.23 |
| UnMIX-TNS (Tomar et al., 2024) | 77.47 | 84.30 | 83.54 | 84.38 | 83.25 | 79.04 | 79.27 | 83.61 | 84.48 | 84.34 | 82.04 | 83.05 |
| NOTE (Gong et al., 2022) | 83.05 | 85.37 | 83.49 | 86.64 | 84.42 | 82.27 | 80.77 | 82.81 | 81.64 | 82.44 | 81.99 | 81.01 |
| SAR (Niu et al., 2023) | 89.33 | 93.93 | 93.17 | 93.72 | 93.32 | 89.77 | 81.89 | 90.58 | 89.31 | 88.62 | 89.17 | 82.34 |
| TEST | 82.09 | 82.02 | 82.43 | 82.17 | 81.80 | 82.80 | 81.08 | 81.49 | 81.59 | 81.40 | 81.27 | 81.19 |
| Robust BN (Yuan et al., 2023) | 75.11 | 87.49 | 86.52 | 88.72 | 87.68 | 76.01 | 70.48 | 85.84 | 84.67 | 87.65 | 86.21 | 71.83 |
| CoTTA (Wang et al., 2022) | 90.18 | 94.17 | 93.21 | 94.17 | 93.51 | 90.02 | 80.68 | 89.98 | 88.14 | 88.02 | 89.01 | 81.56 |
| Balanced BN (Su et al., 2024) | 72.24 | 84.59 | 83.96 | 85.41 | 84.79 | 73.21 | 68.51 | 83.18 | 82.93 | 84.38 | 83.20 | 70.18 |
| LAME (Boudiaf et al., 2022) | 74.77 | 73.44 | 74.90 | 74.27 | 74.10 | 75.81 | 75.30 | 75.06 | 75.45 | 75.04 | 74.59 | 75.73 |
| RoTTA (Yuan et al., 2023) | 68.90 | 80.77 | 80.38 | 81.05 | 80.25 | 71.34 | 68.74 | 78.62 | 79.23 | 79.18 | 77.91 | 73.03 |
| CFA | 70.15 | 58.97 | 60.08 | 65.60 | 62.68 | 70.70 | 71.46 | 62.21 | 61.77 | 68.49 | 65.77 | 72.01 |
| Domain Router | 74.75 | 79.35 | 79.05 | 79.86 | 79.87 | 75.27 | 70.56 | 77.71 | 78.43 | 78.21 | 77.69 | 76.00 |
| SwiTTA | **65.17** | **58.87** | **58.53** | **62.96** | **60.50** | **60.04** | **59.59** | **60.94** | **60.64** | **65.07** | **62.57** | **60.78** |
| ViT-B/16 | | | | | | | | | | | | |
| TEST | 49.19 | 49.10 | 49.48 | 49.10 | 49.25 | 49.22 | 47.84 | 47.81 | 48.19 | 47.89 | 48.08 | 48.00 |
| FOA (Niu et al., 2024) | 43.24 | 43.80 | 44.15 | 44.12 | 43.59 | 42.76 | 41.78 | 41.74 | 41.58 | 42.30 | 42.12 | 40.32 |
| TENT (Wang et al., 2021) | 41.61 | 41.82 | 41.64 | 41.93 | 41.67 | 41.15 | 39.84 | 40.10 | 39.95 | 40.18 | 39.76 | 39.47 |
| VIDA (Liu et al., 2024b) | 41.45 | 40.58 | 40.75 | 40.18 | 40.17 | 41.45 | 39.36 | 38.10 | 37.51 | 38.63 | 37.79 | 38.65 |
| LAME (Boudiaf et al., 2022) | 29.08 | 27.47 | 28.80 | 27.64 | 28.01 | 29.03 | 30.12 | 29.36 | 30.42 | 28.77 | 29.18 | 30.31 |
| SAR (Niu et al., 2023) | 38.40 | 39.25 | 39.10 | 39.24 | 39.28 | 38.74 | 35.90 | 37.01 | 36.83 | 37.08 | 36.71 | 35.57 |
| CFA | 28.66 | 23.59 | 24.14 | 25.90 | 25.06 | 28.51 | 30.49 | 26.64 | 27.10 | 28.92 | 28.18 | 30.41 |
| Domain Router | 38.78 | 31.28 | 31.71 | 32.14 | 31.25 | 35.78 | 33.52 | 29.51 | 29.91 | 29.88 | 29.64 | 32.82 |
| SwiTTA | **17.73** | **12.52** | **12.76** | **13.18** | **18.15** | **18.93** | **22.10** | **16.24** | **16.37** | **18.56** | **16.35** | **20.10** |

*Table 7.* **Average error (%) on ImageNet-C within the UniTTA benchmark.** Continuation of the previous table. "Avg." represents the average error rate across 24 settings.

| Class setting | i.i.d. and balanced (i,1) | | | | | | i.i.d. and imbalanced (i,u) | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Domain setting | (1,1) | (i,1) | (i,u) | (n,1) | (n,u) | (1,u) | (1,1) | (i,1) | (i,u) | (n,1) | (n,u) | (1,u) | Avg. |
| ResNet-50 | | | | | | | | | | | | | |
| ROID (Marsden et al., 2024) | **59.76** | 83.07 | 97.30 | 80.65 | 83.49 | **61.59** | **60.12** | 79.95 | 82.57 | 76.69 | 80.83 | 62.92 | 86.97 |
| TENT (Wang et al., 2021) | 61.98 | 84.07 | 86.81 | 79.07 | 80.25 | 62.07 | 60.66 | 86.36 | 82.38 | 79.25 | 79.65 | **61.94** | 84.26 |
| TRIBE (Su et al., 2024) | 79.65 | 84.16 | 82.89 | 84.64 | 82.91 | 78.45 | 63.68 | 84.01 | 84.34 | 87.68 | 82.61 | 62.37 | 84.40 |
| BN (Nado et al., 2020) | 68.54 | 82.22 | 81.26 | 80.61 | 81.39 | 68.81 | 67.30 | 82.33 | 81.60 | 80.35 | 80.89 | 67.72 | 83.27 |
| UnMIX-TNS (Tomar et al., 2024) | 77.19 | 85.62 | 83.41 | 86.51 | 86.11 | 79.01 | 78.76 | 84.15 | 84.55 | 84.55 | 84.04 | 81.86 | 82.69 |
| NOTE (Gong et al., 2022) | 81.90 | 83.91 | 83.04 | 84.20 | 83.36 | 82.08 | 80.74 | 81.93 | 81.86 | 82.17 | 81.53 | 81.52 | 82.67 |
| SAR (Niu et al., 2023) | 65.28 | 81.32 | 79.61 | 79.96 | 80.82 | 65.81 | 64.83 | 80.77 | 80.58 | 78.81 | 80.12 | 65.98 | 82.46 |
| TEST | 81.98 | 81.90 | 82.36 | 82.32 | 82.48 | 82.48 | 81.42 | 81.28 | 81.51 | 81.31 | 81.35 | 81.51 | 81.80 |
| Robust BN (Yuan et al., 2023) | 69.33 | 84.78 | 83.61 | 86.17 | 85.61 | 69.91 | 68.19 | 84.73 | 84.08 | 86.21 | 84.86 | 68.97 | 81.03 |
| CoTTA (Wang et al., 2022) | 62.04 | 79.33 | 76.56 | 78.05 | 78.51 | 62.45 | 60.42 | 77.41 | 77.55 | **74.76** | 76.77 | 63.25 | 80.82 |
| Balanced BN (Su et al., 2024) | 68.70 | 83.05 | 81.89 | 83.98 | 83.47 | 69.42 | 67.46 | 82.53 | 82.37 | 83.63 | 82.76 | 68.68 | 78.94 |
| LAME (Boudiaf et al., 2022) | 82.76 | 82.12 | 82.67 | 82.64 | 82.79 | 83.28 | 82.26 | 81.48 | 81.78 | 81.63 | 81.67 | 82.27 | 78.58 |
| RoTTA (Yuan et al., 2023) | 64.30 | 80.19 | 78.01 | 80.79 | 80.99 | 67.48 | 67.45 | 78.81 | 78.98 | 79.68 | 79.90 | 71.40 | 76.14 |
| CFA | 83.46 | 82.92 | 83.09 | 83.71 | 83.74 | 83.71 | 82.96 | 82.42 | 82.50 | 83.11 | 82.71 | 83.13 | 74.47 |
| Domain Router | 68.15 | **77.39** | **76.39** | **77.48** | **78.31** | 68.55 | 66.70 | **76.33** | **76.30** | 76.43 | **76.50** | 67.53 | 75.53 |
| SwiTTA | 68.88 | 78.01 | 77.25 | 77.99 | 78.86 | 69.38 | 67.53 | 77.15 | 77.42 | 77.22 | 77.90 | 68.23 | **67.98** (-8.16) |
| ViT-B/16 | | | | | | | | | | | | | |
| TEST | 49.02 | 48.96 | 49.35 | 48.91 | 49.37 | 49.35 | 47.72 | 47.73 | 48.10 | 47.71 | 48.10 | 48.10 | 48.57 |
| FOA (Niu et al., 2024) | 42.14 | 44.07 | 43.32 | 43.83 | 43.19 | 42.06 | 40.96 | 42.00 | 41.36 | 41.90 | 41.46 | 40.72 | 42.44 |
| TENT (Wang et al., 2021) | 41.44 | 41.77 | 41.63 | 41.76 | 41.69 | 41.35 | 39.71 | 40.05 | 39.89 | 40.06 | 39.91 | 39.58 | 40.75 |
| VIDA (Liu et al., 2024b) | 41.09 | 40.41 | 40.13 | 40.43 | 39.96 | 41.17 | 39.38 | 38.75 | 38.43 | 38.51 | 38.45 | 38.97 | 39.60 |
| LAME (Boudiaf et al., 2022) | 46.33 | 45.32 | 45.66 | 45.54 | 45.77 | 46.72 | 44.93 | 43.97 | 44.17 | 44.27 | 44.36 | 45.23 | 37.10 |
| SAR (Niu et al., 2023) | 38.72 | 39.12 | 39.00 | 39.08 | 39.09 | 38.74 | 35.81 | 36.86 | 36.66 | 36.79 | 36.62 | 36.75 | 37.76 |
| CFA | 45.58 | 45.56 | 45.83 | 45.46 | 45.75 | 45.74 | 44.12 | 44.19 | 44.42 | 44.09 | 44.27 | 44.32 | 35.76 |
| Domain Router | **34.29** | **30.79** | **31.17** | **31.10** | **31.00** | 38.79 | **32.80** | 32.36 | 29.95 | 32.62 | **29.75** | 35.71 | 32.36 |
| SwiTTA | 37.87 | 30.96 | 31.42 | 31.35 | 31.23 | **34.96** | 33.42 | **29.85** | **29.92** | **30.14** | 30.17 | **34.12** | **24.52** (-13.24) |

*Table 8.* **Average error (%) on CIFAR10-C within the UniTTA benchmark.** $(\{i, n, 1\}, \{1, u\})$ denotes correlation and imbalance settings, where $\{i, n, 1\}$ represent i.i.d., correlated and continual, respectively, and $\{1, u\}$ represent balance and imbalance, respectively.

| Class setting | correlated and balanced (n,1) | | | | | | correlated and imbalanced (n,u) | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Domain setting | (1,1) | (i,1) | (i,u) | (n,1) | (n,u) | (1,u) | (1,1) | (i,1) | (i,u) | (n,1) | (n,u) | (1,u) |
| TENT (Wang et al., 2021) | 69.00 | 84.00 | 78.90 | 78.98 | 67.25 | 56.21 | 39.13 | 63.82 | 54.19 | 60.77 | 55.36 | 36.97 |
| CoTTA (Wang et al., 2022) | 52.83 | 63.88 | 61.29 | 62.49 | 59.21 | 51.69 | 38.51 | 53.20 | 51.53 | 50.21 | 49.15 | 38.45 |
| BN (Nado et al., 2020) | 49.22 | 57.05 | 54.45 | 56.42 | 54.56 | 48.35 | 40.87 | 51.29 | 48.13 | 49.76 | 47.63 | 39.18 |
| TEST | 43.84 | 43.59 | 40.26 | 43.63 | 40.62 | 39.50 | 42.57 | 42.60 | 39.06 | 42.94 | 39.23 | 38.86 |
| ROID (Marsden et al., 2024) | 43.46 | 57.20 | 53.28 | 53.97 | 52.49 | 43.27 | 40.31 | 54.07 | 49.55 | 50.42 | 48.80 | 39.28 |
| LAME (Boudiaf et al., 2022) | 41.51 | 40.52 | 36.95 | 40.47 | 37.03 | 36.66 | 41.30 | 40.39 | 36.71 | 40.81 | 36.77 | 37.30 |
| Robust BN (Yuan et al., 2023) | 22.99 | 35.71 | 31.40 | 36.19 | 32.07 | 21.78 | 26.80 | 38.53 | 35.23 | 39.10 | 35.03 | 25.57 |
| UnMIX-TNS (Tomar et al., 2024) | 24.22 | 32.71 | 28.52 | 32.80 | 28.60 | 24.70 | 30.59 | 35.89 | 32.83 | 36.04 | 32.07 | 30.41 |
| Balanced BN (Su et al., 2024) | 21.09 | 34.02 | 29.59 | 34.42 | 30.01 | 19.68 | 23.31 | 35.08 | 31.42 | 35.28 | 31.20 | 22.35 |
| RoTTA (Yuan et al., 2023) | 19.03 | 36.55 | 30.87 | 35.82 | 31.14 | 19.96 | 23.05 | 36.64 | 32.32 | 36.23 | 31.97 | 24.48 |
| NOTE (Gong et al., 2022) | 26.50 | 28.79 | 24.94 | 28.39 | 24.87 | 24.20 | 29.74 | 30.92 | 27.74 | 30.56 | 26.62 | 26.33 |
| TRIBE (Su et al., 2024) | 18.31 | 32.08 | 27.70 | 32.49 | 27.85 | 17.64 | 19.43 | 32.87 | 28.92 | 33.07 | 28.33 | 19.14 |
| CFA | 35.07 | 23.00 | 21.45 | 29.49 | 24.44 | 29.78 | 35.64 | 26.01 | 24.08 | 32.40 | 27.14 | 31.28 |
| Domain Router | 21.71 | 28.48 | 24.66 | 28.77 | 25.56 | 19.91 | 23.75 | 30.93 | 27.66 | 30.65 | 27.47 | 22.20 |
| SwiTTA | **10.72** | **12.20** | **10.41** | **14.63** | **11.50** | **9.50** | **15.11** | **17.95** | **15.90** | **20.42** | **16.83** | **14.28** |

*Table 9.* **Average error (%) on CIFAR10-C within the UniTTA benchmark.** Continuation of the previous table. "Avg." represents the average error rate across 24 settings.

| Class setting | i.i.d. and balanced (i,1) | | | | | | i.i.d. and imbalanced (i,u) | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Domain setting | (1,1) | (i,1) | (i,u) | (n,1) | (n,u) | (1,u) | (1,1) | (i,1) | (i,u) | (n,1) | (n,u) | (1,u) | Avg. |
| TENT (Wang et al., 2021) | 29.37 | 57.43 | 42.58 | 53.83 | 41.53 | 21.48 | 23.38 | 54.62 | 38.61 | 49.99 | 35.99 | 27.03 | 51.23 |
| CoTTA (Wang et al., 2022) | **16.84** | 32.75 | 30.49 | 27.59 | 26.97 | **15.72** | 30.27 | 45.05 | 41.70 | 39.38 | 40.43 | 29.91 | 42.06 |
| BN (Nado et al., 2020) | 20.99 | 34.07 | 29.83 | 32.25 | 29.49 | 18.79 | 33.88 | 44.68 | 38.24 | 43.05 | 40.95 | 31.64 | 41.45 |
| TEST | 43.57 | 43.53 | 40.43 | 43.82 | 40.32 | 40.35 | 42.64 | 42.84 | 39.09 | 43.11 | 38.81 | 39.09 | 41.38 |
| ROID (Marsden et al., 2024) | 16.99 | 31.02 | 27.50 | 28.39 | 26.64 | 15.73 | 33.49 | 50.43 | 45.13 | 48.76 | 46.10 | 33.77 | 41.25 |
| LAME (Boudiaf et al., 2022) | 45.15 | 44.69 | 41.54 | 45.30 | 41.45 | 41.83 | 42.05 | 41.81 | 38.26 | 42.41 | 37.54 | 38.35 | 40.49 |
| Robust BN (Yuan et al., 2023) | 20.93 | 33.88 | 29.46 | 34.54 | 29.98 | 19.31 | 31.92 | 44.16 | 37.62 | 44.85 | 40.63 | 29.16 | 32.37 |
| UnMIX-TNS (Tomar et al., 2024) | 24.61 | 32.86 | 28.70 | 33.03 | 28.72 | 24.85 | 31.55 | 39.81 | 33.52 | 40.11 | 34.87 | 31.55 | 31.40 |
| Balanced BN (Su et al., 2024) | 21.17 | 33.93 | 29.59 | 34.59 | 30.00 | 19.57 | 24.10 | 39.99 | 32.75 | 40.39 | 35.09 | 23.10 | 29.66 |
| RoTTA (Yuan et al., 2023) | 17.81 | 33.24 | 28.93 | 33.83 | 29.58 | 18.69 | 23.83 | 39.68 | 32.51 | 40.00 | 35.46 | 25.50 | 29.88 |
| NOTE (Gong et al., 2022) | 25.47 | **27.05** | **24.26** | **27.36** | **24.30** | 23.62 | 29.95 | **31.83** | **26.62** | **31.87** | **27.43** | 26.87 | 27.34 |
| TRIBE (Su et al., 2024) | 18.27 | 32.01 | 27.82 | 32.51 | 28.09 | 17.26 | **19.22** | 33.98 | 27.73 | 33.95 | 27.97 | **19.45** | 26.50 |
| CFA | 51.59 | 49.05 | 46.90 | 51.30 | 47.68 | 48.91 | 47.77 | 44.24 | 43.07 | 47.04 | 42.07 | 44.33 | 37.66 |
| Domain Router | 21.24 | 27.85 | 24.18 | 28.01 | 24.92 | 19.32 | 23.95 | 35.05 | 28.56 | 35.16 | 30.94 | 22.22 | 26.38 |
| SwiTTA | 25.09 | 30.80 | 27.37 | 31.36 | 28.26 | 23.34 | 25.59 | 35.60 | 29.96 | 36.56 | 32.05 | 24.24 | **21.65** (-4.85) |

*Table 10.* **Average error (%) on CIFAR100-C within the UniTTA benchmark.** ($\{i, n, 1\}, \{1, u\}$) denotes correlation and imbalance settings, where $\{i, n, 1\}$ represent i.i.d., correlated and continual, respectively, and $\{1, u\}$ represent balance and imbalance, respectively.

| Class setting | correlated and balanced (n,1) | | | | | | correlated and imbalanced (n,u) | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Domain setting | (1,1) | (i,1) | (i,u) | (n,1) | (n,u) | (1,u) | (1,1) | (i,1) | (i,u) | (n,1) | (n,u) | (1,u) |
| TENT (Wang et al., 2021) | 96.57 | 97.04 | 96.90 | 93.90 | 90.27 | 95.36 | 88.96 | 92.05 | 90.39 | 89.50 | 81.20 | 82.54 |
| BN (Nado et al., 2020) | 76.47 | 79.61 | 78.93 | 78.95 | 79.09 | 76.03 | 64.50 | 70.41 | 69.40 | 69.90 | 68.92 | 63.47 |
| CoTTA (Wang et al., 2022) | 77.80 | 80.49 | 80.14 | 78.32 | 78.43 | 76.46 | 64.00 | 69.51 | 68.90 | 68.95 | 68.63 | 63.57 |
| NOTE (Gong et al., 2022) | 53.91 | 54.59 | 52.96 | 55.00 | 53.30 | 53.41 | 53.89 | 54.58 | 53.06 | 54.70 | 53.33 | 53.05 |
| ROID (Marsden et al., 2024) | 70.82 | 78.22 | 77.10 | 75.84 | 75.45 | 70.16 | 55.59 | 64.37 | 63.86 | 63.12 | 63.04 | 54.54 |
| RoTTA (Yuan et al., 2023) | 38.28 | 54.74 | 51.75 | 53.93 | 53.37 | 39.12 | 42.32 | 54.43 | 52.42 | 54.26 | 52.75 | 46.61 |
| TEST | 46.51 | 46.89 | 44.72 | 47.22 | 45.83 | 44.42 | 46.75 | 47.00 | 44.84 | 47.19 | 44.56 | 44.50 |
| Robust BN (Yuan et al., 2023) | 41.01 | 50.67 | 48.64 | 51.26 | 50.09 | 40.20 | 39.16 | 49.01 | 47.16 | 50.04 | 47.96 | 38.37 |
| UnMIX-TNS (Tomar et al., 2024) | 39.01 | 47.15 | 44.94 | 47.00 | 45.77 | 39.90 | 42.03 | 47.40 | 45.28 | 47.48 | 45.31 | 43.09 |
| Balanced BN (Su et al., 2024) | 36.33 | 46.80 | 44.58 | 47.21 | 46.59 | 35.73 | 37.03 | 46.95 | 45.12 | 47.17 | 45.30 | 37.60 |
| TRIBE (Su et al., 2024) | 34.22 | 48.63 | 44.90 | 47.46 | 44.78 | 33.91 | 33.85 | 46.37 | 43.12 | 46.40 | 42.74 | 35.04 |
| LAME (Boudiaf et al., 2022) | 33.67 | 32.85 | 30.02 | 33.65 | 30.94 | 30.93 | 37.45 | 36.19 | 34.13 | 35.88 | 33.32 | 34.40 |
| CFA | 27.83 | 17.37 | 16.32 | 22.84 | 19.27 | 24.51 | 32.13 | 24.95 | 23.76 | 28.38 | 24.71 | 30.64 |
| Domain Router | 36.43 | 43.42 | 41.63 | 43.59 | 45.00 | 35.98 | 36.27 | 43.83 | 42.38 | 43.59 | 42.67 | 36.18 |
| SwiTTA | **14.93** | **16.49** | **15.50** | **19.17** | **18.88** | **15.10** | **21.17** | **23.13** | **22.51** | **24.10** | **22.92** | **21.34** |

*Table 11.* **Average error (%) on CIFAR100-C within the UniTTA benchmark.** Continuation of the previous table. "Avg." represents the average error rate across 24 settings.

| Class setting | i.i.d. and balanced (i,1) | | | | | | i.i.d. and imbalanced (i,u) | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Domain setting | (1,1) | (i,1) | (i,u) | (n,1) | (n,u) | (1,u) | (1,1) | (i,1) | (i,u) | (n,1) | (n,u) | (1,u) | |
| Corresponding setting | CoTTA | ROID | – | – | – | – | – | – | – | – | – | – | Avg. |
| TENT (Wang et al., 2021) | 82.17 | 90.26 | 86.43 | 87.68 | 80.42 | 67.10 | 59.02 | 82.09 | 61.09 | 79.29 | 63.18 | 43.83 | 82.39 |
| BN (Nado et al., 2020) | 36.27 | 46.64 | 44.51 | 44.83 | 44.07 | 35.46 | 39.09 | 49.14 | 46.68 | 47.62 | 46.58 | 37.71 | 58.10 |
| CoTTA (Wang et al., 2022) | 32.88 | 42.41 | 41.86 | 40.99 | 42.31 | 32.69 | 36.18 | 45.35 | 45.18 | 43.59 | 45.60 | 36.15 | 56.68 |
| NOTE (Gong et al., 2022) | 52.66 | 54.30 | 52.83 | 54.20 | 52.76 | 52.19 | 54.21 | 54.62 | 52.97 | 54.50 | 52.80 | 52.75 | 53.61 |
| ROID (Marsden et al., 2024) | **29.98** | **37.00** | **36.08** | **36.74** | **36.49** | **29.97** | **32.31** | **38.54** | **38.09** | **38.02** | **37.95** | **31.77** | 51.46 |
| RoTTA (Yuan et al., 2023) | 33.77 | 46.56 | 45.64 | 46.92 | 46.95 | 35.66 | 39.60 | 50.68 | 50.34 | 50.92 | 50.51 | 42.53 | 47.25 |
| TEST | 46.42 | 46.42 | 44.43 | 46.50 | 44.53 | 44.49 | 46.91 | 46.99 | 44.81 | 47.17 | 44.89 | 44.81 | 45.78 |
| Robust BN (Yuan et al., 2023) | 35.60 | 45.92 | 43.91 | 46.30 | 44.47 | 35.05 | 38.09 | 48.57 | 45.94 | 48.96 | 47.00 | 37.34 | 44.61 |
| UnMIX-TNS (Tomar et al., 2024) | 39.01 | 46.39 | 44.39 | 46.41 | 44.60 | 39.74 | 42.70 | 47.95 | 45.74 | 48.07 | 46.08 | 43.41 | 44.53 |
| Balanced BN (Su et al., 2024) | 35.88 | 45.97 | 43.94 | 46.16 | 44.35 | 35.66 | 37.54 | 47.21 | 45.07 | 47.36 | 45.44 | 38.03 | 42.88 |
| TRIBE (Su et al., 2024) | 33.08 | 45.67 | 42.75 | 45.93 | 42.72 | 33.28 | 33.69 | 44.28 | 41.83 | 44.14 | 41.29 | 34.92 | 41.04 |
| LAME (Boudiaf et al., 2022) | 48.33 | 47.44 | 45.47 | 47.74 | 45.73 | 46.38 | 47.40 | 46.53 | 44.84 | 46.87 | 44.49 | 45.19 | 39.99 |
| CFA | 53.77 | 53.03 | 51.22 | 53.64 | 51.65 | 51.94 | 53.88 | 53.36 | 51.67 | 53.89 | 51.75 | 51.85 | 38.98 |
| Domain Router | 35.03 | 41.37 | 40.49 | 41.52 | 40.55 | 34.65 | 36.02 | 42.97 | 41.65 | 42.76 | 41.73 | 36.00 | 40.24 |
| SwiTTA | 41.35 | 46.93 | 46.38 | 46.75 | 46.14 | 40.86 | 42.19 | 48.07 | 47.16 | 48.11 | 47.23 | 42.18 | **32.44** (-7.55) |