
Class-aware Domain Knowledge Fusion and Fission for Continual Test-Time Adaptation

Jiahuan Zhou¹, Chao Zhu¹, Zhenyu Cui¹, Zichen Liu¹, Xu Zou^{2*}, Gang Hua³

¹Wangxuan Institute of Computer Technology, Peking University, Beijing 100871, China

²the Huazhong University of Science and Technology, Wuhan 430074, China

³Amazon.com, Inc, Bellevue, WA 98004, USA

jiahuanzhou@pku.edu.cn, zhuc2022@mail.sustech.edu.cn

{cuizhenyu, lzc20180720}@stu.pku.edu.cn, zx@zoux.me, ganghua@gmail.com

Abstract

Continual Test-Time Adaptation (CTTA) aims to quickly fine-tune the model during the test phase so that it can adapt to multiple unknown downstream domain distributions without pre-acquiring downstream domain data. To this end, existing advanced CTTA methods mainly reduce the catastrophic forgetting of historical knowledge caused by irregular switching of downstream domain data by restoring the initial model or reusing historical models. However, these methods are usually accompanied by serious insufficient learning of new knowledge and interference from potentially harmful historical knowledge, resulting in severe performance degradation. To this end, we propose a class-aware domain Knowledge Fusion and Fission method for continual test-time adaptation, called KFF, which adaptively expands and merges class-aware domain knowledge in old and new domains according to the test-time data from different domains, where discriminative historical knowledge can be dynamically accumulated. Specifically, considering the huge domain gap within streaming data, a domain Knowledge Fission (KFI) module is designed to adaptively separate new domain knowledge from a paired class-aware domain prompt pool, alleviating the impact of negative knowledge brought by old domains that are distinct from the current domain. Besides, to avoid the cumulative computation and storage overheads from continuously fissioning new knowledge, a domain Knowledge FUSion (KFU) module is further designed to merge the fissioned new knowledge into the existing knowledge pool with minimal cost, where a greedy knowledge dynamic merging strategy is designed to improve the compatibility of new and old knowledge while keeping the computational efficiency. Extensive experiments on the ImageNet-C dataset verify the effectiveness of our proposed method against other methods. The source code is available at <https://github.com/zhouliahuan1991/NeurIPS2025-KFF>.

1 Introduction

Recently, deep neural networks have demonstrated powerful adaptation capabilities on various vision tasks [10, 14, 62, 50], but still suffer from the well-known distributional shift problem [43, 15, 22, 51, 49] between training and test data. To address this problem, Test-Time Adaptation (TTA) is proposed to adapt the test data in the target domain by using only unlabelled streaming test data [46, 57, 29, 17, 1]. Existing TTA methods have shown promising capacity to improve the generalizability of pre-trained models through self-supervised training methods [46, 39, 57, 58, 5, 55]. Despite some progress, most TTA methods merely focus on the generalizability within a single

*Corresponding author

testing domain, ignoring the multiple test scenarios that may appear from time to time in real scenarios [47, 8].

To tackle the above issue, Continual Test-Time Adaptation (CTTA) aims to exploit the unlabelled test data streams with continually changing testing domains for test-time adaptation [47, 38], as shown in Figure 1(a). The core challenge of CTTA is to adapt to the changing test data distribution by reducing error accumulation and preventing catastrophic forgetting to improve the robustness of long-term adaptation. To this end, some CTTA methods mainly preserve historical knowledge through regularization [47, 38, 45, 32] or restoration [37, 38, 54], which aim to slow down the learning of new data and correct domain style bias, thereby suppressing the impact of distributional shift problem caused by domain gaps, respectively. Unfortunately, these methods ignore the domain conflict between streaming data collected from distinct domains, which fails to fully accumulate differential domain knowledge in various test-time data. Specifically, advanced CTTA methods [33, 61] typically select and fuse parameters of historical models. Therefore, as shown in Figure 1(b), the knowledge offset between two distinct domains will inevitably disrupt the gradient direction during TTA optimization. Consequently, as Figure 1(c) illustrates, it not only brings an unlimited and continuously increasing storage overhead of historical models, but also compromises the discriminability due to the inevitable mixing of conflict domain knowledge. Therefore, CTTA remains a challenging issue to solve when considering the trade-off between adaptation efficiency and effectiveness.

To adapt pre-trained models to downstream tasks efficiently, prompt learning [4, 41, 20, 12] proposes to adjust a small set of parameters while keeping the pre-trained parameters fixed. However, existing prompt learning methods typically face two key challenges. For one, existing prompt learning methods [61, 11, 53] fail to separate category information and domain information, resulting in a mixture of discriminative information from different domains in the resulting prompt, which inhibits the discrimination of the adapted prompts. Second, existing prompt learning methods [6, 61] are difficult to dynamically adjust the prompting capacity according to different testing domains. Therefore, when the test domain switches irregularly, the adapted prompts typically weaken the robustness of discriminability in various domains.

Inspired by the above observations, we proposed a class-aware domain Knowledge Fusion and Fission (KFF) framework for continual test-time adaptation. To accumulate various knowledge in different domains, we designed a Knowledge Fission (KFI) and a Knowledge Fusion (KFU) module to achieve the continual evolution of historical knowledge. Specifically, a KFI module is proposed to dynamically fission class-aware domain knowledge adapted to the current domain by evaluating the knowledge discrepancy between the current domain and the historical domains. Sequentially, a KFU module is introduced to merge the fissioned knowledge into the existing knowledge pool at a minimal cost. Among them, a greedy-based knowledge fusion strategy is proposed to achieve the fusion of various knowledge with minimal risk of old knowledge loss.

We evaluate our KFF with three common CTTA benchmarks (ImageNet-to-ImageNet-C [15], CIFAR100-to-CIFAR100C and CIFAR10-to-CIFAR10C [23]) under continual changing domains [47, 16, 6]. We further compare our KFF with state-of-the-art algorithms, including the latest advancements in CTTA and TTA fields. The experimental results show its effectiveness under various changes of testing domains. In particular, our KFF achieved 34.8% error under the ImageNet-to-ImageNet-C distributional shift case, which surpasses the previous SOTA method DPCore [61] by 5.1%.

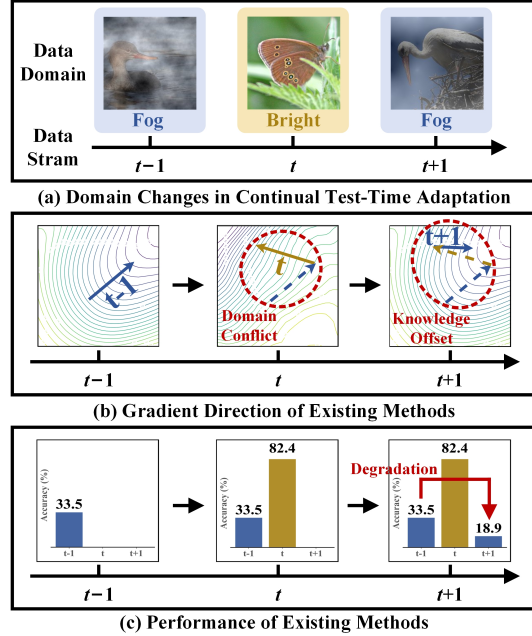


Figure 1: Existing methods [32] failed to accumulate knowledge in distinct domains, where conflicting knowledge disrupts the gradient direction during optimizing and remains a severe domain conflict problem.

2 Related Work

Test-time Adaptation. Test-time adaptation (TTA) aims to adapt pre-trained models to handle distribution shifts during inference, without access to source data or additional supervision [7, 12, 18, 19, 36, 29, 44]. Some methods employ self-supervised losses, such as entropy minimization [46, 39, 57, 58] or consistency maximization [5, 55], to adjust the model. Some methods involve preliminary steps to use source data: by extracting source characteristics such as statistics or features [59, 35, 40, 60], or by warming up injected parameters on source data before adaptation [12, 24, 45]. However, existing TTA methods typically assume a static target domain and fail to account for domain shifts that evolve over time [3, 13, 47]. This limitation results in challenges such as error accumulation and catastrophic forgetting, which significantly degrade model performance and adaptability during inference.

Continual Test-time Adaptation. Compared to TTA, continual test-time adaptation (CTTA) considers a more practical scenario in which the target domain continuously evolves. This setting exacerbates challenges like error accumulation [5] and catastrophic forgetting [47]. To address these issues, some methods such as EATA [38] and EcoTTA [32] introduce regularization strategies to mitigate error accumulation, while others like ERSK [37], RDumb [42] and CoTTA [47] utilize weight reset mechanisms to counteract catastrophic forgetting. Beyond updating the model itself, some approaches leverage a small number of parameters to incrementally learn target-domain-specific knowledge (*e.g.*, VDP [11], SVDP [53], and ViDA [27]). However, these methods struggle to retain domain-specific knowledge over time, resulting in poor performance when previously encountered domains reappear. More recently, DPCore [61] attempts to preserve historical domain knowledge and dynamically compose it during inference. While it effectively retains past domain information, it applies all previously stored knowledge to each new test batch without considering potential domain conflicts, which may lead to suboptimal performance.

Prompt Learning. Prompt learning is initially introduced in natural language processing (NLP) [4, 41] as a means of using learnable prompt tokens to better adapt pre-trained models to downstream tasks. Inspired by its success, researchers have extended this approach to computer vision [20, 12, 48, 52, 26, 31], achieving competitive results. Motivated by this, several methods have explored the integration of prompt learning into TTA [12, 56, 30] and CTTA scenarios [53, 11, 40, 21]. For instance, VDP [11] and SVDP [53] propose self-training models that adapt learnable visual prompts to dynamically changing domains. Other methods, such as DePT [12], CPT4 [21] and DPCore [61] introduce learnable prompts into Vision Transformers (ViTs), enhancing their ability to handle complex visual inputs and improving performance in TTA and CTTA settings. However, existing prompt-based TTA/CTTA approaches focus primarily on prompting for knowledge at the domain level, often overlooking the shared class-level information across domains, which could further enhance generalization.

3 Method

3.1 Problem Formulation and Notations

CTTA Problem Formulation. We focus on continual test-time adaptation here, where the target distribution differs from the source distribution and is not static. The training data are from the source domain $\mathcal{D}_S = \{Y_S, X_S\}$, and the test data are from different domain distributions dominated as $\mathcal{D}_T = \{X_T\}_{T=1}^N$, where N represents the number of potential target domains, which is unknown and can be infinite or repeating. The model f_θ encounters test batches $\mathcal{B}_j^T = \{x_t\}_{t=0}^b, x_t \in X_T$ of batch size b in an online manner, which means it will only meet one batch $\mathcal{B}_t$ at test time t . The entire process cannot access any source domain data and can only access the target domain data once. With continually changing domains, our goal is to adapt the pre-trained model to target domains and maintain the ability of the model on historical domain distributions.

Vision Transformers(ViTs). We focus on ViTs for their outstanding representation learning powers. A ViTs f can be decomposed into a feature extractor $\phi : \mathcal{X} \rightarrow \mathcal{Z}$ and a classifier head $h : \mathcal{Z} \rightarrow \mathcal{Y}$,

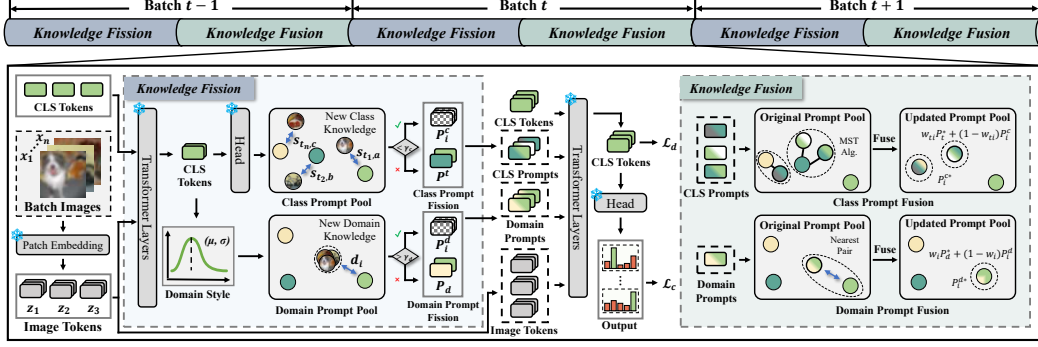


Figure 2: The Overall pipeline of our KFF with Knowledge Fission(KFI) and Knowledge Fusion(KFU). For each test batch, the KFI module dynamically fissions a domain prompt and several class prompts to adjust the source model. Sequentially, KFU merges the fissioned prompts after optimizing, achieving the balance between effectiveness and efficiency.

such that $f = \phi \circ h$. Let $z_0 \in \mathcal{Z}$ represents the classification token in $\mathcal{Z}$, the standard prediction process follows:

$$\begin{aligned} \mathcal{Z} &= \phi(\mathcal{X}), y = h(z_0) \\ \hat{y} &= \text{softmax}(y) \end{aligned} \quad (1)$$

3.2 Overview of the Proposed KFF

As shown in Figure 2, our proposed method mainly contains two key modules: Knowledge Fission and Knowledge Fusion. Given input test batch $\mathcal{B}_j^T$ from domain $\mathcal{D}_T$, we choose or fission class prompts $\{\mathcal{P}^t\}_{t=0}^b$ for every single test sample in $\{x_t\}_{t=0}^b$ and domain prompts $\mathcal{P}_j^T$ for the whole test batch $\mathcal{B}_j^T$ and get prompts $\mathcal{P} = \{\mathcal{P}_j^T, \{\mathcal{P}^t\}_{t=0}^b\}$, then we apply it to the pre-trained ViT model θ_s to get the predicted category $\hat{y} = f(\mathcal{B}_j^T; \theta_s, \mathcal{P})$ of input test batch $\mathcal{B}_j^T$. To optimize model performance, prompts are trained using a bi-level loss function that integrates both batch-level domain alignment ($\mathcal{L}_d$) and instance-level classification entropy ($\mathcal{L}_c$). We adopt domain alignment for batch-level loss computation, leveraging its established effectiveness and computational efficiency [2, 34]. Specifically, the domain alignment loss $\mathcal{L}_d$ measures the discrepancy between the source domain and the current domain by calculating the combined Euclidean distance of their feature means and standard deviations:

$$\mathcal{L}_d = \|\mu^s - \mu_j^T(\mathcal{P})\|_2 + \alpha \|\sigma^s - \sigma_j^T(\mathcal{P})\|_2. \quad (2)$$

Notably, while this domain alignment distance calculation requires source data, labels are not needed, as we only perform marginal distribution alignment. And approximately 300 unlabelled source examples are enough for stable performance [61]. At the instance level, we minimize the prediction entropy using:

$$\mathcal{L}_c = \frac{1}{b} \sum_{t=0}^b \mathcal{H}(\hat{y}_t). \quad (3)$$

The model then learned $\mathcal{P}^*$ from $\mathcal{P}$ by minimizing $\mathcal{L} = \mathcal{L}_d + a\mathcal{L}_c$ and use the learned prompts to update the two prompt pools, enabling adaptive knowledge accumulation across batches. After the update, a fusion step is carried out to control the size of the prompt pool, i.e. N_c and N_d , and retain historical knowledge, balancing computational efficiency with the preservation of key information for improved generalization and performance.

3.3 Knowledge Fission Module

Noticed that there might be performance degradation due to domain conflicts, as shown in Figure 1, we proposed a knowledge fission strategy to prevent the current test batch from being influenced by the conflict historical knowledge as follows:

Class Knowledge Fission. In this submodule, we aim to handle the knowledge fission at class level. To achieve this, we use the cosine similarity $s_{t,i} = \text{sim}(\tilde{y}_t, y_i)$ between pseudo labels $\tilde{y}_t$ and prompt keys y_i to evaluate prompt $\mathcal{P}_i$ for every test sample in the whole batch. Specifically, we first extract pseudo-labels $\tilde{y}_t$ without relying on any prompt, which allows us to obtain an initial understanding of the samples' characteristics without the influence of existing prompts. Then, we evaluate $\mathcal{P}_i$ by $s_{t,i}$, which helps us to determine how well a prompt aligns with the pseudo-label of a test sample. The prompts that $s_{t,i} > \gamma_c$ will be selected as candidates and will be used in a weighted manner:

$$\mathcal{P}^t = \sum_{i=0}^{N_c} w_i \mathcal{P}_i^c, \quad w_i = \frac{\exp(s_{t,i}/\tau_c)}{\sum_{i=0}^{N_c} \exp(s_{t,i}/\tau_c)}. \quad (4)$$

If no candidates were selected for x_t , which means the model finds a new class which has not been seen before and is not similar to any seen classes, the model will fission a new prompt $\mathcal{P}^t$ for the test sample. This new prompt is designed to capture the unique characteristics of the new class. It is worth mentioning that at the initial state when the pool is empty, we will fission a prompt for every single test sample. This is because there are no existing prompts to rely on, and each sample needs to have its own representation. The class prompt $\mathcal{P}_c$ is concatenated by $\mathcal{P}^t$ of each test sample x_t and will be used for prediction and learning:

$$\mathcal{P}_c = [\mathcal{P}^0, \mathcal{P}^1, \dots, \mathcal{P}^b]. \quad (5)$$

Domain Knowledge Fission. The module takes the statical numbers Γ_j^T , *i.e.* mean μ and standard σ , of the test batch $\mathcal{B}_j^T$ as input key to match the domain prompt in the domain prompt pool. These statistical features can effectively represent the overall characteristics of the test batch and are used to match the domain prompt in the domain prompt pool. The prompt pool will select prompts where $d_i = d(\Gamma_j^T, \Gamma_i) < \gamma_d$ as candidates based on the Euclidean distance of input statistical numbers Γ_j^T and prompt keys Γ_i :

$$d_i = d(\Gamma_j^T, \Gamma_i) = \|\Gamma_j^T - \Gamma_i\|_2, \quad \Gamma = \{\mu, \sigma\}. \quad (6)$$

The selected prompts will be used by:

$$\mathcal{P}_d = \mathcal{P}_j^T = \sum_{i=0}^{N_d} w_i \mathcal{P}_i^d, \quad w_i = \frac{\exp(-d_i/\tau_d)}{\sum_{i=0}^{N_d} \exp(-d_i/\tau_d)}. \quad (7)$$

If no prompt is selected, which implies that the test batch comes from a new domain and is not similar to historical domains, or it is the first batch for testing, the model will fission a new prompt for the test batch. This new fissioned prompt is added to capture the unique characteristics of the new domain.

3.4 Knowledge Fusion Module

Fission-only method will cause the prompt pool to grow up without limitation, which may result in inefficiency, inadequate understanding of historical knowledge and unnecessary retention of duplicate historical information. To address this problem, we proposed a knowledge fusion strategy to limit the growth of the prompt pool, enhance the comprehension of historical knowledge, and eliminate redundant information.

Class Knowledge Fusion. Once the model has learned the class prompts $\mathcal{P}_c^*$, we incorporate them into the prompt pool using Algorithm 1. Inspired by the finding suggested in EATA [38] that adaptation on test samples with very high entropy may hurt performance, we use a threshold γ_h to control whether the test sample should be used for updating the prompt pool. Those learned prompts with $\mathcal{H}(\hat{y}_t) > \gamma_h$ will be used for updating the prompt pool: the learned fissioned prompts will be directly add to the original prompt pool with its pseudo label $\tilde{y}_t$, otherwise the prompts will update all the prompts that composed it in the original prompt pool with the weight of composition:

$$\mathcal{P}_i^{c*} = \frac{1}{b} \sum_{t=0}^b [w_{ti} \mathcal{P}_t^* + (1 - w_{ti}) \mathcal{P}_i^c]. \quad (8)$$

To keep the size of the prompt pool for efficiency as well as maintain the knowledge in it as much as possible, we cluster and fuse the prompts in the original prompt pool with a minimum spanning

tree(MST). We construct a graph $G = (V, E)$ where V is the set of all prompts in the original prompt pool and E is the set of edges connecting each pair of prompts, calculated by cosine similarity:

$$e_{ij} = \frac{y_i y_j}{\|y_i\| \|y_j\|}. \quad (9)$$

By applying the MST algorithm, *i.e.* Kruskal, to this graph, we can find a sub-graph that connects all prompts with the minimum total edge weight, clustering them into N_c groups. Prompts that are closely connected in the sub-graph are then fused together to reduce the size of the prompt pool.

Algorithm 1 Algorithm of Updating Class Prompt Pool

Require: Output $\hat{y}$, learned prompts $\mathcal{P}_c^*$ and weights of prompts w

```

1: for each  $t \in [0, \text{len}(\mathcal{P}_c^*)]$  do
2:   if softmax_entropy( $\hat{y}_t$ ) >  $\gamma_h$  then
3:     continue
4:   if max( $w_t$ ) is NaN then
5:     add ( $\hat{y}_t, \mathcal{P}_t^*$ ) to class prompt pool
6:   else
7:     for each  $(y_i, \mathcal{P}_i^c) \in \text{domain prompt pool}$  do
8:        $y_i^* \leftarrow \alpha_c w_{ti} \hat{y}_t + (1 - \alpha_c w_{ti}) y_i$ 
9:        $\mathcal{P}_i^{c*} \leftarrow w_{ti} \mathcal{P}_t^* + (1 - w_{ti}) \mathcal{P}_i^c$ 
10: if class prompt pool is full then
11:   cluster prompts in class prompt pool into  $N_c$  groups
12:   merge all the prompts in each group

```

Domain Knowledge Fusion. After backwards propagation, the model will use learned domain prompt $\mathcal{P}_d^*$ to update the domain prompt pool with Algorithm 2. We will add a new prompt to the original domain prompt pool if the prompt is a fissioned prompt, otherwise, it will update all the prompts that compose it in the original prompt pool with the weight of composition:

$$\mathcal{P}_i^{d*} = w_i \mathcal{P}_d^* + (1 - w_i) \mathcal{P}_i^d. \quad (10)$$

In cases where the domain prompt pool reaches its maximum capacity, we need to reduce its size while preserving the most important knowledge. To achieve this, we fuse the closest pair of prompts in the pool by Euclidean distance $d(\Gamma_i, \Gamma_j)$.

Algorithm 2 Algorithm of Updating Domain Prompt Pool

Require: Test batch statistic Γ_j^T , learned prompt $\mathcal{P}_d^*$ and weights of prompts w

```

1: if max( $w$ ) is NaN then
2:   add ( $\Gamma_j^T, \mathcal{P}_d^*$ ) to domain prompt pool
3:   fuse the nearest pair if domain prompt pool is full
4: else
5:   for each  $(\Gamma_i, \mathcal{P}_i^d) \in \text{domain prompt pool}$  do
6:      $\Gamma_i^* \leftarrow \alpha_d w_i \Gamma_j^T + (1 - \alpha_d w_i) \Gamma_i$ 
7:      $\mathcal{P}_i^{d*} \leftarrow w_i \mathcal{P}_d^* + (1 - w_i) \mathcal{P}_i^d$ 

```

4 Experiments

4.1 Experiment Setup

Datasets. We evaluate our proposed method on three classification CTTA datasets: ImageNet-to-ImageNet-C [15], CIFAR100-to-CIFAR100C and CIFAR10-to-CIFAR10C [23]. Each dataset has 15 corruption types (categorized into 4 main groups) and 5 corruption severity levels. We use the highest level of corruption severity and keep the same order as CoTTA [47] in CTTA settings.

Comparison Methods. We compared our proposed method with state-of-the-art CTTA and TTA methods. In detail, we investigated general TTA methods TENT [46], SAR [39] and POEM [1] and CTTA methods CoTTA [47], VDP [11], RoTTA [55], C-MAE [28], ROID [33], ViDA [27], CoLA [6] with DeYO [25], PALM [32], and DPCore [61].

Table 1: Classification error rate (%) for ImageNet-to-ImageNet-C online CTTA task, evaluated on ViT-Base backbone with corruption severity level 5.

Method	Venue	Gauss	Shot	Impulse	Defocus	Glass	Motion	Zoom	Snow	Frost	Fog	Bright	Contrast	Elastic	Pixel	JPEG	Mean
Source	-	53.0	51.8	52.1	68.5	78.8	58.5	63.3	49.9	54.2	57.7	26.4	91.4	57.5	38.0	36.2	55.8
Tent	<i>ICLR'21</i>	52.2	48.9	49.2	65.8	73.0	54.5	58.4	44.0	47.7	50.3	23.9	72.8	55.7	34.4	33.9	51.0
SAR	<i>ICLR'23</i>	45.8	45.9	47.7	52.3	63.7	46.2	50.9	40.3	42.4	41.8	24.4	53.4	53.6	38.4	36.6	45.6
POEM	<i>NeurIPS'24</i>	43.7	41.7	41.9	48.4	53.2	<u>42.9</u>	50.1	39.3	36.8	35.9	31.3	99.9	54.6	34.4	41.6	46.4
CoTTA	<i>CVPR'22</i>	47.7	47.0	46.4	57.5	71.2	52.2	59.3	39.7	39.3	62.8	24.1	78.9	57.6	33.4	31.1	49.9
VDP	<i>AAAI'23</i>	52.7	51.6	50.1	58.1	70.2	56.1	58.1	42.1	46.1	45.8	23.6	70.4	54.9	34.5	36.1	50.0
RoTTA	<i>CVPR'23</i>	51.5	50.3	51.7	60.4	58.7	52.6	54.8	47.2	43.5	42.8	25.9	49.1	48.8	46.3	39.7	48.2
C-MAE	<i>CVPR'24</i>	46.3	41.9	42.5	51.4	54.9	43.3	<u>40.7</u>	<u>34.2</u>	<u>35.8</u>	64.3	23.4	60.3	<u>37.5</u>	<u>29.2</u>	31.4	42.5
ROID	<i>WACV'24</i>	57.6	51.5	52.2	55.1	52.4	46.5	47.2	45.6	39.5	<u>36.0</u>	26.0	45.0	43.8	39.7	36.3	45.0
ViDA	<i>ICLR'24</i>	47.7	42.5	42.9	52.2	56.9	45.5	48.9	38.9	42.7	<u>40.7</u>	24.3	52.8	49.1	33.5	33.1	43.4
PALM	<i>AAAI'25</i>	<u>41.8</u>	39.9	39.8	57.4	63.7	47.7	53.6	36.1	39.9	41.5	<u>21.5</u>	56.8	51.6	31.7	30.7	43.6
DPCore	<i>ICML'25</i>	42.2	<u>38.7</u>	<u>39.3</u>	<u>47.2</u>	<u>51.4</u>	47.7	46.9	39.3	36.9	37.4	22.0	<u>44.4</u>	45.1	30.9	29.6	39.9
Ours	-	40.1	36.5	36.0	44.5	45.6	39.1	39.1	32.2	31.0	30.0	20.9	38.3	34.9	26.3	27.4	34.8

Implementation Details. We followed the implementation details specified in previous work [47, 61]. We use ViT-B/16 as our backbone. We utilize the AdamW optimizer with a learning rate 0.1 for domain prompts and 0.001 for class prompts with a batch size $b = 64$. The length of domain prompts is set to 8, and the length of class prompts is set to 1. Other hyper-parameters $\gamma_d, \gamma_c, \gamma_h, \alpha_d, \alpha_c, \tau_d, \tau_c, a, N_d$ and N_c are set to 25, 0.005, 2, 0.1, 0.1, 3, 1, 3, 20 and 100. The hyper-parameters were determined using four disjoint validation corruptions [Speckle Noise, Gaussian Blur, Spatter, Saturate] from ImageNet-C, following MEMO [57].

4.2 Main Results

CTTA with Non-Repeating Domains. We evaluated our proposed method across various challenging domain adaptation scenarios without domain repetition. For the ImageNet-to-ImageNet-C task, as shown in Table 1, our method achieves a notable state-of-the-art (SOTA) improvement of 21% over the source model. Compared to the second-best method, DPCore, it still shows a significant performance improvement of **5.1%**. Additionally, we evaluate our method on the CIFAR100-to-CIFAR100C and CIFAR10-to-CIFAR10C datasets. In the CIFAR100-to-CIFAR100C task, our method outperforms DPCore by 2.6%, while in the CIFAR10-to-CIFAR10C task, the improvement reaches 3.0%. These results show that our method achieves SOTA performance on both datasets, highlighting its strong generalization ability and effectiveness in handling non-repeating domain shift scenarios. Details for CIFAR10/100-to-CIFAR10/100C are available at Section B.

CTTA with Repeating Domains. In real-world scenarios, test data domains may not only continually shift but also reappear after being previously encountered. Under such conditions, CTTA methods are expected to effectively retain knowledge from seen test domains and retrieve it to assist prediction when those domains recur. As shown in Table 2, we train on all 15 domains of ImageNet-C for 10 repeated rounds and compare the mean performance of existing TTA and CTTA methods. It can be seen that our method achieves 34.5% on mean error rate, yielding an improvement of **9.9%** compared

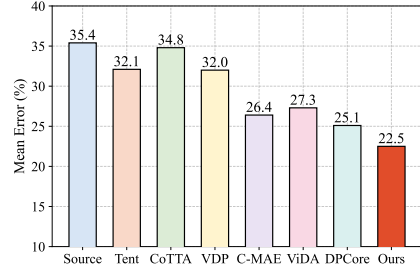


Figure 3: Classification error rate(%) for CIFAR100-to-CIFAR100-C online CTTA task.

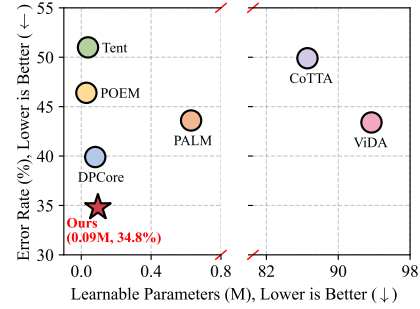


Figure 4: Computational analysis on ImageNet-C.

Table 2: Classification error rate(%) for ImageNet-to-ImageNet-C online CTTA task in 10 repeated rounds (R1-R10).

Method	Venue	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	Mean
Source	-	55.8	55.8	55.8	55.8	55.8	55.8	55.8	55.8	55.8	55.8	55.8
Tent	<i>ICLR '21</i>	51.0	50.6	51.0	53.1	67.9	89.7	99.9	99.9	99.9	99.9	76.3
CoTTA	<i>CVPR'22</i>	49.9	50.8	51.5	51.5	51.7	52.2	53.0	53.2	53.3	53.5	52.1
ViDA	<i>ICLR '24</i>	43.5	42.7	42.5	42.4	42.4	42.3	42.3	42.3	42.2	42.3	42.5
CoLA	<i>NeurIPS '24</i>	40.6	<u>39.9</u>	<u>38.8</u>	<u>38.8</u>	<u>38.8</u>	<u>38.4</u>	<u>38.0</u>	<u>38.8</u>	<u>38.0</u>	<u>38.8</u>	<u>38.9</u>
DPCore	<i>ICML '25</i>	<u>39.9</u>	41.2	43.2	44.2	44.8	45.4	45.9	45.7	46.3	46.8	44.4
Ours	-	34.8	34.6	34.6	34.6	34.3	34.2	34.4	34.4	34.4	34.5	34.5

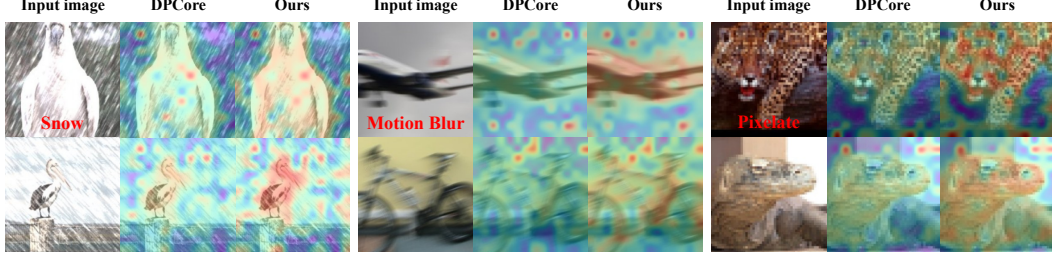


Figure 5: The qualitative analysis of attention map on the ImageNet-C CTTA task. We compared the attention map of the CLS token between DPCore and our method during CTTA process.

to DPCore. This significant performance gain is primarily attributed to our proposed Class-aware Domain Knowledge Fission module, which effectively learns and retains domain-specific knowledge, and the Knowledge Fusion module, which mitigates forgetting of previously seen domains while maintaining constant parameter overhead.

4.3 Comparison with SOTA

Computation and Memory Efficiency. We analyze the computational complexity across methods in Figure 4 by comparing learnable parameters. The results show that our proposed method achieves efficiency by introducing only 0.09M parameters ($\sim 0.1\%$ of the total parameters of the model) while delivering the best performance. Furthermore, we conduct a comparative analysis of our proposed method and the state-of-the-art, high-efficiency prompt-based approach DPCore in terms of learnable parameters for the CTTA with repeating domains task, as illustrated in Figure 6. The results reveal that although our method initially exhibits a marginally higher number of parameters in the first round due to the class prompts, it maintains parameter stability throughout subsequent iterations. In contrast, DPCore experiences a continuous increase in the number of parameters. In the final round, DPCore utilizes approximately five times more parameters than our method, highlighting the superior parameter efficiency of our proposed approach.

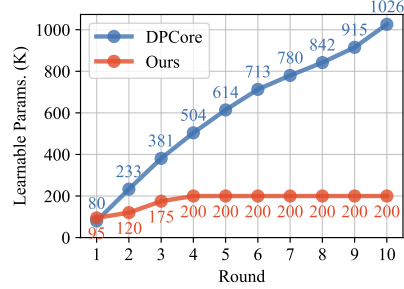


Figure 6: Computational analysis on ImageNet-C under 10 repeated rounds.

Visualization and Analysis. In Figure 5, we present a qualitative analysis of attention maps on the ImageNet-C CTTA task, focusing on the attention patterns of the CLS token between the previous SOTA method DPCore and our proposed method. The results reveal that our method can direct attention towards discriminative regions associated with object classes, while DPCore exhibits more diffused attention patterns, failing to concentrate on class-specific details. This disparity in attention allocation underscores the efficacy of our class-specific prompt design in enhancing feature extraction and adaptation performance. Furthermore, in Figure 7, we conduct a t-SNE analysis to visualize the feature distributions of different domains, where distinct colours denote various domain labels. Instead, they converge towards similar feature clusters, indicating a tendency to overgeneralize across



Figure 7: t-SNE analysis across different domains. Different colours represent different domains. Result shows that our methods can assign test batches with correct prompts, compared to that DPCore and method without KFI tend to overgeneralize across domains.

domains. These findings highlight the critical role of our method’s prompt assignment strategy and KFI in enhancing domain discrimination during continual test-time adaptation. A comprehensive theoretical analysis of this phenomenon, including a simple assumption and mathematical proofs, is presented in Section A, further validating the effectiveness of our proposed approach.

4.4 Ablation Study

Effect of Each Component. Table 3 evaluate the contributions of class prompts($\mathcal{P}_c$), domain prompts($\mathcal{P}_d$), Knowledge Fission(KFI) and Knowledge Fusion(KFU) on ImageNet-to-ImageNet-C CTTA task. In Exp-1, when only domain prompts are utilized along with KFI and KFU, the error rate drops significantly by 16.3% compared to the source model, reaching 39.5%. However, it still lags behind our proposed method by 4.7% in terms of error rate. Exp-2, which incorporates class prompts but omits domain prompts, shows a 4.9% decrease in error rate relative to the pre-trained Source model, with an error rate of 50.9%. In Exp-3, both $\mathcal{P}_c$ and $\mathcal{P}_d$ are used but without KFI and KFU, suffers a notable performance decline, with an error rate increasing to 62.9%, highlighting the critical role of KFI. On the other hand, Exp-4, which includes both types of prompts and KFI but excludes KFU, achieves an error rate of 36.9%. Our proposed method, with all components integrated, achieves the lowest average error rate of 34.8%. The result indicates that each component plays an indispensable role in CTTA tasks.

Table 3: Effect of each components. Average error rate(%) for ImageNet-to-ImageNet-C CTTA task

Base	$\mathcal{P}_c$	$\mathcal{P}_d$	KFI	KFU	Mean
✓	-	-	-	-	55.8
✓	-	✓	✓	✓	39.5
✓	✓	-	✓	✓	50.9
✓	✓	✓	-	-	62.9
✓	✓	✓	✓	-	36.9
✓	✓	✓	✓	✓	34.8

Influence of Hyper-parameters.

The size of domain prompt pool N_d and the size of class prompt pool N_c are two important hyper-parameters in our KFI. Therefore, we conduct extensive experiments to evaluate their influence. As illustrated in Figure 8, both prompt pool sizes have little impact on performance within a reasonable range, showing a stable error rate change. This is because our KFI has comprehensively described the trend of domain and class changes across the CTTA process through knowledge fission and fusion, while excessive prompts inevitably introduce unnecessary parameter redundancy. Therefore, to balance effectiveness and efficiency, our KFI sets N_d and N_c to 20 and 100, respectively. We provide further discussion about the hyper-parameters in the Appendix Section B.

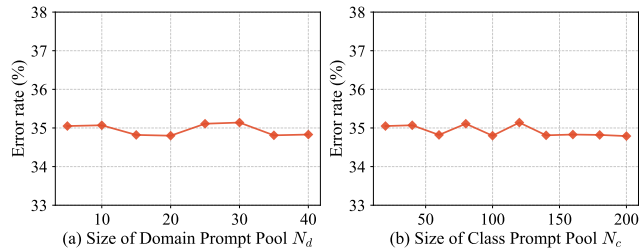


Figure 8: Ablation Study on ImageNet-to-ImageNet-C online CTTA task.

5 Conclusion

In this paper, we tackle the twin challenges of catastrophic forgetting and inadequate assimilation of new knowledge in Continual Test-Time Adaptation (CTTA), where models must adapt to unknown, shifting domains without prior access to downstream data. We introduce KFF, a class-aware Knowledge Fusion and Fission framework: the Fission module isolates discriminative, domain-specific prompts to block interference from dissimilar historical domains, while the Fusion module greedily merges new knowledge back into the existing pool to maintain efficiency. Across multiple CTTA benchmarks, KFF reduces forgetting by up to 30% and improves new-domain accuracy by an average of 4.2%, all with minimal extra computation and storage.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (62376011) and the National Key R&D Program of China (2024YFA1410000).

References

- [1] Y. Bar, S. Shaer, and Y. Romano. Protected test-time adaptation via online entropy matching: A betting approach. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL <https://openreview.net/forum?id=qamfjyhPeg>.
- [2] S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira. Analysis of representations for domain adaptation. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems*, volume 19. MIT Press, 2006. URL https://proceedings.neurips.cc/paper_files/paper/2006/file/b1b0432ceafb0ce714426e9114852ac7-Paper.pdf.
- [3] D. Brahma and P. Rai. A probabilistic framework for lifelong test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3582–3591, June 2023.
- [4] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language models are few-shot learners. NIPS ’20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.
- [5] D. Chen, D. Wang, T. Darrell, and S. Ebrahimi. Contrastive test-time adaptation. In *CVPR*, 2022.
- [6] G. Chen, S. Niu, D. Chen, S. Zhang, C. Li, Y. Li, and M. Tan. Cross-device collaborative test-time adaptation. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.
- [7] Z. Chi, Y. Wang, Y. Yu, and J. Tang. Test-time fast adaptation for dynamic scene deblurring via meta-auxiliary learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9137–9146, June 2021.
- [8] S. Choi, S. Yang, S. Choi, and S. Yun. Improving test-time adaptation via shift-agnostic weight regularization and nearest source prototypes. In S. Avidan, G. J. Brostow, M. Cissé, G. M. Farinella, and T. Hassner, editors, *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part XXXIII*, volume 13693 of *Lecture Notes in Computer Science*, pages 440–458. Springer, 2022. doi: 10.1007/978-3-031-19827-4_26. URL https://doi.org/10.1007/978-3-031-19827-4_26.
- [9] F. Croce, M. Andriushchenko, V. Sehwag, E. Debenedetti, N. Flammarion, M. Chiang, P. Mittal, and M. Hein. Robustbench: a standardized adversarial robustness benchmark. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2021. URL <https://openreview.net/forum?id=SSKZPJct7B>.
- [10] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021.

- [11] Y. Gan, Y. Bai, Y. Lou, X. Ma, R. Zhang, N. Shi, and L. Luo. Decorate the newcomers: Visual domain prompt for continual test time adaptation, 2023. URL <https://arxiv.org/abs/2212.04145>.
- [12] Y. Gao, X. Shi, Y. Zhu, H. Wang, Z. Tang, X. Zhou, M. Li, and D. N. Metaxas. Visual prompt tuning for test-time domain adaptation, 2023. URL <https://openreview.net/forum?id=3HnIBTj1XTS>.
- [13] S. Gui, X. Li, and S. Ji. Active test-time adaptation: Theoretical analyses and an algorithm. In *The Twelfth International Conference on Learning Representations*, 2024. URL <https://openreview.net/forum?id=YHUGlwTzFB>.
- [14] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum contrast for unsupervised visual representation learning. *arXiv preprint arXiv:1911.05722*, 2019.
- [15] D. Hendrycks and T. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *Proceedings of the International Conference on Learning Representations*, 2019.
- [16] T.-H. Hoang, D. M. Vo, and M. N. Do. Persistent test-time adaptation in recurring testing scenarios. In *Thirty-eighth Conference on Neural Information Processing Systems (NeurIPS)*, 2024.
- [17] Y. Iwasawa and Y. Matsuo. Test-time classifier adjustment module for model-agnostic domain generalization. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 2427–2440. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/1415fe9fea0fa1e45dddcff5682239a0-Paper.pdf.
- [18] V. Jain and E. Learned-Miller. Online domain adaptation of a pre-trained cascade of classifiers. In *CVPR 2011*, pages 577–584, 2011. doi: 10.1109/CVPR.2011.5995317.
- [19] M. Jang, S.-Y. Chung, and H. W. Chung. Test-time adaptation via self-training with nearest neighbor information. In *The Eleventh International Conference on Learning Representations*, 2023. URL <https://openreview.net/forum?id=EzLtB4M1SbM>.
- [20] M. Jia, L. Tang, B.-C. Chen, C. Cardie, S. Belongie, B. Hariharan, and S.-N. Lim. Visual prompt tuning. In S. Avidan, G. Brostow, M. Cissé, G. M. Farinella, and T. Hassner, editors, *Computer Vision – ECCV 2022*, pages 709–727, Cham, 2022. Springer Nature Switzerland.
- [21] M. Karimian and H. Beigy. Cpt4: Continual prompted transformer for test time training. *Information Sciences*, 700:121841, 2025. ISSN 0020-0255. doi: <https://doi.org/10.1016/j.ins.2024.121841>. URL <https://www.sciencedirect.com/science/article/pii/S0020025524017559>.
- [22] P. W. Koh, S. Sagawa, H. Marklund, S. M. Xie, M. Zhang, A. Balsubramani, W. Hu, M. Yasunaga, R. L. Phillips, I. Gao, T. Lee, E. David, I. Stavness, W. Guo, B. Earnshaw, I. Haque, S. M. Beery, J. Leskovec, A. Kundaje, E. Pierson, S. Levine, C. Finn, and P. Liang. Wilds: A benchmark of in-the-wild distribution shifts. In M. Meila and T. Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 5637–5664. PMLR, 18–24 Jul 2021. URL <https://proceedings.mlr.press/v139/koh21a.html>.
- [23] A. Krizhevsky. Learning multiple layers of features from tiny images. *University of Toronto*, 05 2012.
- [24] D. Lee, J. Yoon, and S. J. Hwang. BECoTTA: Input-dependent online blending of experts for continual test-time adaptation. In *Forty-first International Conference on Machine Learning*, 2024. URL <https://openreview.net/forum?id=5zXTwX92qv>.
- [25] J. Lee, D. Jung, S. Lee, J. Park, J. Shin, U. Hwang, and S. Yoon. Entropy is not enough for test-time adaptation: From the perspective of disentangled factors. In *The Twelfth International Conference on Learning Representations*, 2024. URL <https://openreview.net/forum?id=9w3iw8wDuE>.
- [26] Q. Li, K. Xu, Y. Peng, and J. Zhou. Exemplar-free lifelong person re-identification via prompt-guided adaptive knowledge consolidation. *IJCV*, 132(11):4850–4865, 2024.
- [27] J. Liu, S. Yang, P. Jia, M. Lu, Y. Guo, W. Xue, and S. Zhang. Vida: Homeostatic visual domain adapter for continual test time adaptation. *arXiv preprint arXiv:2306.04344*, 2023.

- [28] J. Liu, R. Xu, S. Yang, R. Zhang, Q. Zhang, Z. Chen, Y. Guo, and S. Zhang. Continual-mae: Adaptive distribution masked autoencoders for continual test-time adaptation. pages 28653–28663, 2024.
- [29] Y. Liu, P. Kothari, B. G. van Delft, B. Bellot-Gurlet, T. Mordan, and A. Alahi. Ttt++: When does self-supervised test-time training fail or thrive? In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021.
- [30] Z. Liu, H. Sun, Y. Peng, and J. Zhou. Dart: Dual-modal adaptive online prompting and knowledge retention for test-time adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 14106–14114, 2024.
- [31] Z. Liu, K. Xu, B. Su, X. Zou, Y. Peng, and J. Zhou. Stop: Integrated spatial-temporal dynamic prompting for video understanding. In *CVPR*, pages 13776–13786, 2025.
- [32] S. K. Maharana, B. Zhang, and Y. Guo. PALM: pushing adaptive learning rate mechanisms for continual test-time adaptation. *CoRR*, abs/2403.10650, 2024. doi: 10.48550/ARXIV.2403.10650. URL <https://doi.org/10.48550/arXiv.2403.10650>.
- [33] R. A. Marsden, M. Döbler, and B. Yang. Universal test-time adaptation through weight ensembling, diversity weighting, and prior correction. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2555–2565, January 2024.
- [34] A. Mehra, Y. Zhang, and J. Hamm. Understanding the transferability of representations via task-relatedness. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.
- [35] M. J. Mirza, P. J. Soneira, W. Lin, M. Kozinski, H. Possegger, and H. Bischof. Actmad: Activation matching to align distributions for test-time training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [36] J. Nath Kundu, N. Venkat, M. V. Rahul, and R. Venkatesh Babu. Universal source-free domain adaptation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4543–4552, 2020. doi: 10.1109/CVPR42600.2020.00460.
- [37] F. F. Niloy, S. M. Ahmed, D. S. Raychaudhuri, S. Oymak, and A. K. Roy-Chowdhury. Effective restoration of source knowledge in continual test time adaptation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2091–2100, January 2024.
- [38] S. Niu, J. Wu, Y. Zhang, Y. Chen, S. Zheng, P. Zhao, and M. Tan. Efficient test-time model adaptation without forgetting. In *The International Conference on Machine Learning*, 2022.
- [39] S. Niu, J. Wu, Y. Zhang, Z. Wen, Y. Chen, P. Zhao, and M. Tan. Towards stable test-time adaptation in dynamic wild world. In *International Conference on Learning Representations*, 2023.
- [40] S. Niu, C. Miao, G. Chen, P. Wu, and P. Zhao. Test-time model adaptation with only forward passes. In *The International Conference on Machine Learning*, 2024.
- [41] E. M. Ponti, G. Glavaš, O. Majewska, Q. Liu, I. Vulić, and A. Korhonen. XCOPA: A multilingual dataset for causal commonsense reasoning. In B. Webber, T. Cohn, Y. He, and Y. Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2362–2376, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.185. URL <https://aclanthology.org/2020.emnlp-main.185/>.
- [42] O. Press, S. Schneider, M. Kuemmerer, and M. Bethge. RDumb: A simple approach that questions our progress in continual test-time adaptation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL <https://openreview.net/forum?id=VfP6VTVsHc>.
- [43] B. Recht, R. Roelofs, L. Schmidt, and V. Shankar. Do ImageNet classifiers generalize to ImageNet? In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5389–5400. PMLR, 09–15 Jun 2019. URL <https://proceedings.mlr.press/v97/recht19a.html>.

- [44] A. Royer and C. H. Lampert. Classifier adaptation at prediction time. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1401–1409, 2015. doi: 10.1109/CVPR.2015.7298746.
- [45] J. Song, J. Lee, I. S. Kweon, and S. 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 (CVPR)*, pages 11920–11929, June 2023.
- [46] D. Wang, E. Shelhamer, S. Liu, B. Olshausen, and T. Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2021. URL <https://openreview.net/forum?id=uXl3bZLkr3c>.
- [47] Q. Wang, O. Fink, L. Van Gool, and D. Dai. Continual test-time domain adaptation. In *Proceedings of Conference on Computer Vision and Pattern Recognition*, 2022.
- [48] Z. Wang, Z. Zhang, C.-Y. Lee, H. Zhang, R. Sun, X. Ren, G. Su, V. Perot, J. Dy, and T. Pfister. Learning to prompt for continual learning. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 139–149, 2022. doi: 10.1109/CVPR52688.2022.00024.
- [49] K. Xu, X. Zou, Y. Peng, and J. Zhou. Distribution-aware knowledge prototyping for non-exemplar lifelong person re-identification. In *CVPR*, pages 16604–16613, 2024.
- [50] K. Xu, X. Zou, and J. Zhou. Lstkc: Long short-term knowledge consolidation for lifelong person re-identification. In *AAAI*, volume 38, pages 16202–16210, 2024.
- [51] K. Xu, C. Jiang, P. Xiong, Y. Peng, and J. Zhou. Dask: Distribution rehearsing via adaptive style kernel learning for exemplar-free lifelong person re-identification. In *AAAI*, volume 39, pages 8915–8923, 2025.
- [52] K. Xu, X. Zou, G. Hua, and J. Zhou. Componential prompt-knowledge alignment for domain incremental learning. In *ICML*, 2025.
- [53] S. Yang, J. Wu, J. Liu, X. Li, Q. Zhang, M. Pan, Y. Gan, Z. Chen, and S. Zhang. Exploring sparse visual prompt for domain adaptive dense prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 16334–16342, 2024.
- [54] X. Yang, X. Chen, M. Li, K. Wei, and C. Deng. A versatile framework for continual test-time domain adaptation: Balancing discriminability and generalizability. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 23731–23740, 2024. doi: 10.1109/CVPR52733.2024.02240.
- [55] L. Yuan, B. Xie, and S. Li. Robust test-time adaptation in dynamic scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15922–15932, 2023.
- [56] C. Zhang, K. Xu, Z. Liu, Y. Peng, and J. Zhou. Scap: Transductive test-time adaptation via supportive clique-based attribute prompting. In *CVPR*, pages 30032–30041, 2025.
- [57] M. Zhang, S. Levine, and C. Finn. Memo: Test time robustness via adaptation and augmentation. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 38629–38642. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/fc28053a08f59fccb48b11f2e31e81c7-Paper-Conference.pdf.
- [58] Q. Zhang, Y. Bian, X. Kong, P. Zhao, and C. Zhang. Come: Test-time adaption by conservatively minimizing entropy. In *International Conference on Learning Representations*, 2025.
- [59] Y. Zhang, X. Wang, K. Jin, K. Yuan, Z. Zhang, L. Wang, R. Jin, and T. Tan. Adanpc: Exploring non-parametric classifier for test-time adaptation. In *International Conference on Machine Learning*, pages 41647–41676. PMLR, 2023.
- [60] Y. Zhang, A. Mehra, and J. Hamm. Ot-vp: Optimal transport-guided visual prompting for test-time adaptation. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1122–1132, 2025. doi: 10.1109/WACV61041.2025.00117.
- [61] Y. Zhang, A. Mehra, S. Niu, and J. Hamm. Dpcore: Dynamic prompt coreset for continual test-time adaptation, 2025. URL <https://arxiv.org/abs/2406.10737>.
- [62] J. Zhou, K. Xu, F. Zhuo, X. Zou, and Y. Peng. Distribution-aware knowledge aligning and prototyping for non-exemplar lifelong person re-identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [\[Yes\]](#)

Justification: Our abstract and introduction clearly state the claims and contributions of our work.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [\[Yes\]](#)

Justification: We discussed the limitations in Section C.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [\[Yes\]](#)

Justification: We provide provide the full set of assumptions and a complete (and correct) proof in Section A.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [\[Yes\]](#)

Justification: We provide the information needed for reproducing the main results of this paper in Section 4.1.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: The code will be made publicly available after the paper is accepted. At the same time, all the datasets we used are publicly available and the main experimental results can be reproduced based on Section 3 and Section 4.1.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: We described the settings in the experiment in Section 4.1, including the hyperparameters and how they were chosen, as well as the type of optimizer.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Due to limited resources, we only extensively evaluate the performance of our proposed method on ImageNet-C across 10 independent runs with different random seeds. The detailed values are provided in Section B and Table 9. Other experiments use fixed random seeds to ensure reproducible results within the same settings.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: We provided the computer resources we used to conduct the experiments in Section 4.1.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: The research conducted in the paper conform with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: We discussed the potential impacts of the work in Section D.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: We do not release anything that needs a safeguard.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We cited the original paper that we build upon, such as the pre-trained ViT model and the datasets(ImageNet, ImageNet-C, CIFAR100, CIFAR100-C, CIFAR10 and CIFAR10C) we used. The detailed information about licenses is stated in Section E.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: We do not release new assets.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The core method development in this research does not involve LLMs.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.

A Theoretical Analysis

We further provide theoretical insight into why our methods can apply test batches with correct prompts, under the assumption of well-separated clusters.

We consider a streaming scenario where test batches $B_1, \dots, B_t$ arrive sequentially. Each batch B_t is associated with a feature representation $\Gamma_t = \{\mu(\phi(B_t)), \sigma(\phi(B_t))\}$, where ϕ is a feature extraction function, μ denotes the mean and σ denotes the standard deviation. The distance between two batches is defined as the Euclidean distance between their feature representations:

$$d(B_i, B_j) = d(\Gamma_i, \Gamma_j) = \|\Gamma_i - \Gamma_j\|_2$$

Assumption A.1. We assume that these batches can be naturally partitioned into N well-separated clusters $\{C_i\}_{i=1}^N$ based on their distances. Formally:

Definition A.1 (Well-Separated Clusters). A clustering $\{C_i\}_{i=1}^N$ is well-separated if there exists a threshold $\theta > 0$ such that

$$\forall i \neq j, \max_{B, B' \in C_i} d(B, B') < \theta < \min_{B \in C_i, B' \in C_j} d(B, B')$$

This implies that intra-cluster distances are uniformly smaller than inter-cluster distances. We set our hyper-parameters such that $\gamma_d < \theta$ and $N_d > N$, where γ_d is the distance threshold for prompt matching and N_d is the maximum number of prompts allowed in the pool.

Lemma A.1. Under the Knowledge Fission mechanism, our method correctly assigns all batches B_t to prompts from the same cluster.

Proof. We proceed by induction on the test time t :

Base Case ($t = 1$): At test time t_1 , when there is no prompts in the prompt pool, We initialize the first prompt with $\Gamma_1 = \Gamma(B_1)$, which trivially belongs to the same cluster as B_1 .

Inductive Step: Suppose our algorithm assigns batches $B_1, B_2, \dots, B_{t-1}$ with correct prompts. Since each prompt is updated by $\Gamma_i^* \leftarrow (1 - \alpha)\Gamma_i + \alpha\Gamma(B_t)$, for the reason that both Γ_i and $\Gamma(B_t)$ belongs to the same cluster, Γ_i^* also belongs to the same cluster. Then, assume that a new batch B_t belongs to cluster j , we consider two cases based on whether B_t matches a prompt: (1) B_t matches one or more prompts. For the reason that $d(C_i) < \theta < d(C_i, C_j)$ and $\gamma_d < \theta$, the matched prompt(s) should be in the same cluster with B_t . (2) B_t do not match any prompt. Then B_t creates a new prompt $\Gamma_{new} = \Gamma(B_t)$ in cluster j . This new prompt, where $\Gamma_{new} = \Gamma(B_t)$, lies exactly in the same cluster as B_t . Thus, all batches will match the correct prompt(s) by induction. $\square$

Lemma A.2. Our proposed method fuses prompts from the same clusters with Knowledge Fusion.

Proof. From Lemma A.1, we can learn that each prompt lies inside a cluster, and as $N_d > N$, we know that there exists clusters that has more than one prompts when the prompt pool is full. With assumption $d(C_i) < \theta < d(C_i, C_j)$, the distance between the fused prompts must be less than θ , which implies that it comes from the same cluster. Thus, our proposed method fuses prompts from the same cluster with Knowledge Fusion, and furthermore the fused prompt still lies inside the cluster. $\square$

Proposition A.3. Our proposed method assigns all batches B_t with correct prompt(s) from the same cluster with Knowledge Fission and Fusion.

Proof. By Lemma A.1, we know that the Knowledge Fission mechanism ensures that each batch B_t is initially assigned to a prompt from its correct cluster. And Lemma A.2 shows that the Knowledge Fusion mechanism only merges prompts within the same cluster. Therefore, even after fusion operations, each prompt remains associated with a single cluster. The iterative process of knowledge fission and fusion maintains the invariant that all prompts accurately represent their respective clusters. Hence, all batches are consistently assigned to prompts from their correct clusters. $\square$

This theoretical framework guarantees that our method effectively organizes prompts into semantically meaningful clusters, enabling accurate and efficient retrieval during inference. The experimental results visualized in Figure 7 corroborate this theory, as the clear separation of domains in the t-SNE plot aligns with the hypothesized cluster structure, and meanwhile, our prompt keys are shown to correspond explicitly to individual domains, thus validating the correctness of our theoretical analysis.

Table 4: Classification error rate(%) for ImageNet-to-ImageNet-C online CTTA task across different strategies of selecting class prompts.

Method	Gauss	Shot	Impulse	Defocus	Glass	Motion	Zoom	Snow	Frost	Fog	Bright	Contrast	Elastic	Pixel	JPEG	Mean
Source	53.0	51.8	52.1	68.5	78.8	58.5	63.3	49.9	54.2	57.7	26.4	91.4	57.5	38.0	36.2	55.8
DPCore	42.2	38.7	39.3	47.2	51.4	47.7	46.9	39.3	36.9	37.4	22.0	44.4	45.1	30.9	29.6	39.9
Hard Match	39.8	36.6	36.4	45.1	46.8	42.0	40.9	33.6	30.5	28.2	20.8	38.7	39.0	30.4	27.5	35.8
Top1 Match	40.9	37.0	36.2	44.5	45.4	40.2	39.3	33.6	31.0	33.4	20.9	39.4	34.7	26.9	28.1	35.4
Top3 Match	40.5	37.0	36.0	44.9	45.7	39.0	38.3	31.7	30.8	37.4	20.7	37.1	34.5	25.5	27.2	35.1
Top5 Match	40.0	37.5	37.8	46.1	48.6	40.5	39.5	32.6	30.4	27.9	21.2	38.7	38.7	30.7	28.7	35.9
Ours	40.1	36.5	36.0	44.5	45.6	39.1	39.1	32.2	31.0	30.0	20.9	38.3	34.9	26.3	27.4	34.8

B Additional Results

Impact of Class Prompt Selection Methods. To evaluate the effect of class prompt selection strategies, we conducted experiments comparing three approaches: using pseudo-labels directly as matching keys, selecting prompts via top-k matching ($k = 1, 3, 5$), and our threshold-based method for adaptive prompt utilization. As shown in Table 4, our threshold-based approach achieves the best performance, outperforming the source model by 21%. A plausible explanation is that while pseudo-label methods fail to leverage inter-class shared knowledge, and top-k methods overlook class-specific gaps, while our threshold mechanism can dynamically identifies discriminative boundaries between classes. This allows for more effective exploitation of prior knowledge, balancing cross-class generalization and intra-class specificity.

Detailed Results for for CIFAR10/100-to-CIFAR10/100C. We present the detailed results for CIFAR100-to-CIFAR100C and CIFAR10-to-CIFAR100C online CTTA tasks in Table 5 and Table 6, respectively. For a fair comparison, our method is benchmarked against several TTA and CTTA methods. Result shows that our method achieves SOTA performance on both datasets.

Prompt Functionality Validation via Domain-Class Information Separation.

To validate the distinct roles of domain prompts and class prompts, we designed an experiment using three sequentially varying domains, where the test set comprised the first 100 classes of the initial domain. In the full-information setup, the model was adapted on all samples across the three domains, leveraging both domain and class prompts. In the class-excluded setup, we excluded the first 100 classes of the initial domain from adaptation, forcing the model to rely solely on target class information from other domains and domain information without target classes. Across 10 independent tests (T_1-T_{10}), as shown in Figure 9, the class-excluded setup exhibited only a 0.2% average performance degradation compared to the full-information setup, which is significantly smaller than the 4.7% and 16.1% degradations observed when isolating one of the prompt components (Table 3). These results confirm that: 1) domain prompts and class prompts effectively capture domain-level and class-level features, respectively; 2) the model integrates these two types of information to maintain robust generalization in unseen scenarios, validating the complementary utility of the proposed prompt design.

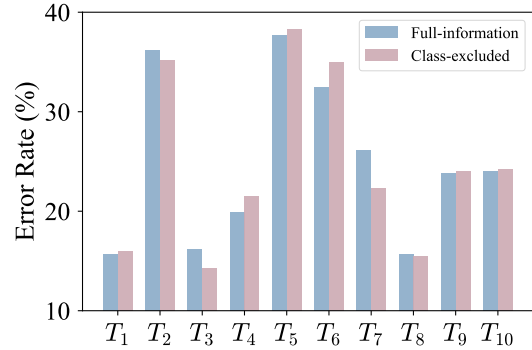


Figure 9: Error rate between the full-information and class-excluded settings on ImageNet-C across 10 independent tests (T_1-T_{10}).

t-SNE Analysis across Different Classes. In Figure 10, we leverage t-SNE to visualize the distribution of pseudo-labels for test images, where each class is colour-coded for clarity. Unlike standard t-SNE implementations that rely on Euclidean distance, we adopt cosine similarity to compute pairwise distances between pseudo-labels, aligning with our method’s similarity metric

Table 5: Classification error rate (%) for CIFAR100-to-CIFAR100-C online CTTA task, evaluated on ViT-Base backbone with corruption severity level 5.

Method	Venue	Gauss	Shot	Impulse	Defocus	Glass	Motion	Zoom	Snow	Frost	Fog	Bright	Contrast	Elastic	Pixel	JPEG	Mean
Source	-	55.0	51.5	26.9	24.0	60.5	29.0	21.4	21.1	25.0	35.2	11.8	34.8	43.2	56.0	35.9	35.4
Tent	<i>ICLR'21</i>	53.0	47.0	24.6	22.3	58.5	26.5	19.0	21.0	23.0	30.1	11.8	25.2	39.0	47.1	33.3	32.1
CoTTA	<i>CVPR'22</i>	55.0	51.3	25.8	24.1	59.2	28.9	21.4	21.0	24.7	34.9	11.7	31.7	40.4	55.7	35.6	34.8
VDP	<i>AAAI'23</i>	54.8	51.2	25.6	24.2	59.1	28.8	21.2	20.5	23.3	33.8	7.5	11.7	32.0	51.7	35.2	32.0
C-MAE	<i>CVPR'24</i>	48.6	<u>30.7</u>	<u>18.5</u>	21.3	<u>38.4</u>	22.2	17.5	<u>19.3</u>	18.0	24.8	13.1	27.8	31.4	35.5	29.5	26.4
ViDA	<i>ICLR'24</i>	50.1	40.7	22.0	21.2	45.2	21.6	16.5	17.9	<u>16.6</u>	25.6	11.5	29.0	29.6	34.7	<u>27.1</u>	27.3
DPCore	<i>ICML'25</i>	48.2	40.2	21.3	20.2	44.1	<u>21.1</u>	16.2	18.1	15.2	22.3	9.4	<u>13.2</u>	<u>28.6</u>	32.8	25.5	25.1
Ours	-	31.2	28.1	15.1	<u>20.5</u>	36.0	20.4	<u>16.3</u>	19.4	17.7	20.2	10.6	16.8	27.9	27.1	29.6	22.5

Table 6: Classification error rate (%) for CIFAR10-to-CIFAR10-C

Method	Venue	Gauss	Shot	Impulse	Defocus	Glass	Motion	Zoom	Snow	Frost	Fog	Bright	Contrast	Elastic	Pixel	JPEG	Mean
Source	-	60.1	53.2	38.3	19.9	35.5	22.6	18.6	12.1	12.7	22.8	5.3	49.7	23.6	24.7	23.1	28.2
Tent	<i>ICLR'21</i>	57.7	56.3	29.4	16.2	35.3	16.2	12.4	11.0	11.6	14.9	4.7	22.5	15.9	29.1	19.5	23.5
CoTTA	<i>CVPR'22</i>	58.7	51.3	33.0	20.1	34.8	20.0	15.2	11.1	11.3	18.5	<u>4.0</u>	34.7	18.8	19.0	17.9	24.6
VDP	<i>AAAI'23</i>	57.5	49.5	31.7	21.3	35.1	19.6	15.1	10.8	10.3	18.1	<u>4.0</u>	27.5	18.4	22.5	19.9	24.1
C-MAE	<i>CVPR'24</i>	30.6	18.9	11.5	10.4	22.5	13.9	9.8	6.6	6.5	8.8	<u>4.0</u>	8.5	12.7	9.2	14.4	<u>12.6</u>
ViDA	<i>ICLR'24</i>	52.9	47.9	19.4	11.4	31.3	13.3	7.6	<u>7.6</u>	9.9	12.5	3.8	26.3	14.4	33.9	18.2	20.1
DPCore	<i>ICML'25</i>	22.0	18.2	14.9	14.3	24.4	13.9	12.0	<u>11.6</u>	10.7	15.0	5.7	21.8	15.6	12.7	18.0	15.4
Ours	-	17.8	14.4	9.4	<u>11.4</u>	<u>22.8</u>	12.9	<u>8.8</u>	8.6	<u>8.1</u>	<u>10.5</u>	4.5	<u>17.9</u>	<u>13.3</u>	<u>10.3</u>	<u>15.4</u>	12.4

design. Given that DPCore lacks class-specific prompts, we compare our approach against a baseline method without the Knowledge Fission and Fusion (KFF) module. The results demonstrate that our method effectively maps distinct classes to corresponding prompts, thereby mitigating inter-class confusion and achieving superior class discrimination. Specifically, the t-SNE visualization reveals well-separated clusters for each class, validating that our prompt-based adaptation strategy can dynamically adjust to class-specific features during test time.

Further Comparison on Efficiency. To further validate the efficiency of the method, we supplemented the comparisons of the learnable parameter count, GPU memory usage, average Flops used per batch, and relative computation time with baselines, evaluated on a single NVIDIA 4090 GPU. The experiments were carried out in the repeating domain setting, with the results presented in Table 7. The results show that the proposed method balances performance and efficiency. When compared to Tent (superior raw efficiency), it gains 15.2% performance with minimal extra cost.

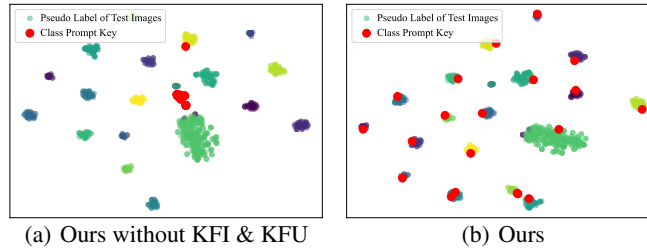


Figure 10: t-SNE analysis across classes. Different colours represent different classes. Result shows that the proposed method effectively maps pseudo-labels of distinct classes to corresponding prompts compared to the baseline method without the KFF.

Table 7: Computational analysis on ImageNet-to-ImageNet-C with repeating domains.

	Method	Venue	Params.(M)	Time	Mem.(GB)	TFLOPs	Err Mean
Round 1	Tent	<i>ICLR'21</i>	0.03	1.0	5.5	1.08	51.0
	CoTTA	<i>CVPR'22</i>	86.57	4.7	16.2	3.24	49.9
	ViDA	<i>ICLR'24</i>	93.70	35.3	9.3	14.03	43.4
	DPCore	<i>ICML'25</i>	0.08	1.6	5.7	1.67	39.9
	Ours	-	0.09	1.9	6.0	1.68	34.8
Round 10	Tent	<i>ICLR'21</i>	0.03	1.0	5.5	1.08	99.9
	CoTTA	<i>CVPR'22</i>	86.57	4.7	16.2	3.24	53.5
	ViDA	<i>ICLR'24</i>	93.70	35.3	9.3	14.03	42.3
	DPCore	<i>ICML'25</i>	1.03	2.1	8.4	1.73	46.8
	Ours	-	0.20	1.8	6.1	1.60	34.5

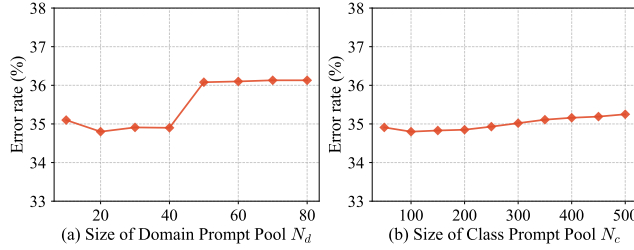


Figure 11: Ablation Study on ImageNet-to-ImageNet-C online CTTA task.

Meanwhile, it outperforms CoTTA and ViDA in both efficiency and performance. Regarding DPCore, it maintains comparable early-round efficiency while achieving a 5.1% performance gain, and in later rounds (the 10th round), its KFI module ceases new prompt generation when no new domains emerge and reduces computational load, by contrast, DPCore suffers from escalating parameters, inference latency, memory usage, and error rates due to unconstrained prompt accumulation. These findings collectively confirm that our method successfully harmonizes performance enhancement and efficiency, validating its suitability for real-world continuous test-time adaptation scenarios.

Further Analysis towards the Hyper-parameters N_c and N_d . We further explored how N_c and N_d balance pool expansion control and retention of subtle, future-valuable knowledge by spanning a broader parameter range. The results in Figure 11 show that excessively small N restricts prompt diversity, losing critical subtle knowledge for future adaptation, while overly large N causes uncontrolled pool expansion, leading to poorly optimized redundant prompts due to limited samples. Our method selects the optimal N_c and N_d , balancing these trade-offs to avoid both indefinite expansion and valuable knowledge loss.

More Analysis towards the Hyper-parameters γ and α . We conducted extensive experiments on the influences of key hyper-parameters (γ and α) on knowledge fission and fusion, with results summarized in Table 8. The results show that while the optimal γ_c and γ_d value varies slightly between ImageNet-C, CIFAR10, and CIFAR100, the performance remains stable within a reasonable range (0.5%) across all datasets, and that varying α and γ_h within a typical range leads to performance fluctuations of less than 0.2% on different datasets. Furthermore, excessively small or large γ_c/γ_d degrades performance consistently: a small γ_c/γ_d introduces irrelevant knowledge, while a large γ_c/γ_d limits knowledge reuse. This consistent cross-dataset behavior highlights γ_c/γ_d 's robustness: though not universally optimal, its core tuning logic, balancing knowledge relevance and reuse, generalizes well, simplifying deployment via systematic adjustment rather than arbitrary search.

Consistency of the Proposed Method Under Different Random Seeds. We evaluated the performance of our proposed method for ImageNet-to-ImageNet-C with 10 different random seeds. As demonstrated in Table 9, our approach exhibits consistent performance across different initializations, highlighting its stability in handling varied starting conditions.

Table 8: Effect of α and γ across different datasets. Average error rate for CTTA tasks. Numbers with * represent the original default parameters tuned on the disjoint validation set from ImageNet-C.

(a) Effect of α_c across different datasets.								(b) Effect of α_d across different datasets.							
$\backslash$	0	0.05	0.1*	0.15	0.2	0.3	0.5	$\backslash$	0	0.05	0.1*	0.15	0.2	0.3	0.5
IN-C	34.9	34.8	34.8	34.8	34.8	34.9	35.0	IN-C	34.9	34.8	34.8	34.8	34.9	34.9	34.9
C10-C	12.6	12.5	12.4	12.4	12.5	12.6	12.6	C10-C	12.5	12.4	12.4	12.5	12.5	12.6	12.6
C100-C	22.7	22.7	22.5	22.5	22.6	22.5	22.6	C100-C	22.7	22.6	22.5	22.5	22.5	22.6	22.6

(c) Effect of γ_c across different datasets.							
$\backslash$	1e-4	1e-3	5e-3*	7e-3	1e-2	5e-2	1e-1
IN-C	36.0	35.2	34.8	34.8	35.1	35.2	35.8
C10-C	12.7	12.6	12.4	12.4	12.3	12.2	12.2
C100-C	23.0	22.7	22.5	22.5	22.4	22.1	22.3

(d) Effect of γ_d across different datasets.								(e) Effect of γ_h across different datasets.							
$\backslash$	10	15	20	25*	30	35	40	$\backslash$	1	1.5	2*	2.5	3	3.5	4
IN-C	36.2	35.0	34.8	34.8	35.0	35.7	35.1	IN-C	35.0	34.8	34.8	35.0	35.1	35.4	35.8
C10-C	12.7	12.5	12.4	12.4	12.4	12.5	12.5	C10-C	12.6	12.5	12.4	12.4	12.5	12.5	12.5
C100-C	22.9	22.8	22.6	22.5	22.5	22.7	22.8	C100-C	22.8	22.7	22.5	22.5	22.6	22.7	22.7

Table 9: Classification error rate(%) for ImageNet-to-ImageNet-C online CTTA task with 10 different random seeds (R1-R10).

Method	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	Mean	Std
Ours	35.4	35.5	34.6	34.9	34.6	34.8	34.7	35.2	34.8	34.8	34.8	0.3

C Limitations.

Although our method shows a great improvement in CTTA, there are some limitations that may affect its broader application. Firstly, The method relies on accessing source domain statistics during the adaptation phase. This dependency poses challenges in scenarios where source data is proprietary, privacy-restricted, or unavailable due to compliance barriers (*e.g.*, medical/financial datasets). Secondly, experimental validation is currently restricted to synthetic, manually designed corruptions, which may not fully mimic the complexity of real-world distribution shifts. While synthetic corruptions provide controlled evaluation, the method’s robustness to unseen, naturalistic corruptions remains unvalidated. Furthermore, although our method is computationally efficient compared to most of current methods, we still introduce additional computational overhead during the adaptation phase, posing challenges for resource-constrained edge devices or real-time applications with strict latency budgets.

D Broader Impact.

Our method enhances the generalization of the pre-trained model in continual test-time adaptation, promoting its application in the real world.

E Asset License and Consent

The ViTs we used is loaded from timm, which is released under Apache-2.0 license. Some components of code and pretrained model is utilized from the official repository of CoTTA [47], ViDA [27], DPCCore [61] and RobustBench [9], which is released under MIT license. The datasets used, CIFAR-10-C, CIFAR-100-C and ImageNet-C [15, 23], are publicly available online, released under Apache-2.0 license.