Diffusion-calibrated Continual Test-time Adaptation

Xu Yang, Moqi Li, Kun Wei*

Xidian University
{xuyang.xd, moqili14, weikunsk}@gmail.com

Abstract

Continual Test-Time Domain Adaptation (CTTA) aims to adapt a pre-trained source model to a dynamically evolving target domain without requiring additional data collection or labeling efforts. A key challenge in this setting is to achieve rapid performance improvement in the current domain using unlabeled data, while avoiding impairing generalization to future domains in complex scenarios. To enhance the discriminative capability of the inference models, we propose a novel framework that integrates an external auxiliary generative model with a test-time adaptive method, leveraging cross-validation to identify reliable supervisory signals. Specifically, for each test instance, we use a diffusion module to generate a calibrated instance from the textual description of its predicted category. Based on the generated one, we design a learning strategy with the following components: (1) the calibrated instance and its category are used to form a supervisory signal; (2) the predicted category of the calibrated instance is compared with the test instance for selecting reliable signals. For these generated and selected instances, adaptive weighting is applied during optimization to stabilize the category distribution and preserve prediction diversity. Finally, based on the inverse process of diffusion, we construct a negative instance from the generated instance and introduce robust contrastive learning to calibrate model optimization further. Extensive experiments demonstrate that our method achieves state-of-the-art performance across multiple benchmarks. Ablation studies further validate the effectiveness of each proposed component.

1 Introduction

Deep neural networks have demonstrated exceptional performance in visual tasks when trained and evaluated on data following identical distributions. Nevertheless, these models face generalization challenges due to domain shift—a pervasive issue arising from discrepancies between training and testing environments (Wang et al. 2023a). For instance, a classification model trained on standard natural images often fails to recognize corrupted or distorted inputs. To address this, domain adaptation techniques aim to transfer knowledge from a labeled source domain to an unlabeled target domain by mitigating distributional differences between the two domains. The most common approach, unsupervised domain adaptation (UDA) (Li et al. 2020; Wang et al. 2023b), assumes access to both source and target data during adaptation. However, real-world scenarios frequently involve two critical constraints: 1) the target domain lacks labeled annotations, and 2) source domain data becomes inaccessible at inference time due to privacy concerns or operational limitations. These constraints define the more challenging source-free/test-time domain adaptation (TTA) paradigm (Chen et al. 2022; Yang et al. 2021; Liu, Zhang, and Wang 2021), which relies solely on the pretrained source model and unlabeled target samples for adaptation.

Existing Test-Time Adaptation (TTA) methods typically address domain shift by updating model parameters using pseudo-labels or entropy regularization. These approaches work effectively when the target distribution is fixed but exhibit unstable behavior when the target domain distribution changes continuously (Wang et al. 2022; Prabhu et al. 2021; Yang et al. 2024a, 2023). This issue has given rise to a novel and largely underexplored research area: Continual Test-Time Domain Adaptation (CTTA), where a pretrained source model must adapt to a stream of dynamically changing target domains without access to source data. Research shows that CTTA primarily faces the following key challenges. It needs to exploit supervisory signals without ground-truth labels to boost performance on the current target domain, yet widespread noisy pseudo-labels significantly hinder this process, while overfitting to the existing domain (especially under noisy pseudo-labels) will impair generalization to future domains, particularly when the current target distribution is narrow. These challenges highlight the complexity of CTTA, which requires balancing dynamic adaptation, memory retention, and robustness to noisy supervision. This problem space remains largely unexplored in the current literature.

Recent advancements in addressing the intractable problem of Continual Test-Time Domain Adaptation (CTTA) have introduced several methodological innovations (Döbler, Marsden, and Yang 2023; Gan et al. 2022; Yang et al. 2024b,c). CoTTA (Wang et al. 2022) employs a weight-averaged teacher network to enhance pseudo-label quality while preserving source-domain knowledge by partially re-

*Corresponding Author

taining original model parameters. However, empirical studies (Marsden, Döbler, and Yang 2023) reveal a critical tradeoff: domain-specific parameter adjustments may compromise generalization to subsequent domains. To mitigate this limitation, ViDA (Liu et al. 2023) introduces visual domain adapters that utilize both high-rank and low-rank feature representations to adapt to current domain distributions while maintaining cross-domain knowledge. As an alternative strategy, methods such as GongNote and Universal TTA (Marsden, Döbler, and Yang 2023) adopt a minimalist approach, updating only normalization parameters while freezing other weights, thereby retaining most of the pre-trained model's source knowledge. For uncertainty-aware adaptation, Continual-MAE (Liu et al. 2024) leverages Monte Carlo (MC) dropout to quantify pixel-level uncertainty and employs a masked autoencoder to enhance domain-invariant representations. Meanwhile, REM (Han, Na, and Hwang 2025) addresses prediction consistency by aligning probability distributions across varying difficulty levels while preserving entropy rank orders. Despite these advances, the inherent complexity of CTTA remains a significant challenge. Constructing a robust adaptive inference framework under noisy target data conditions—without access to source data—demands novel strategies that balance dynamic adaptation and long-term generalization.

To address the inference model's limited discriminative capacity in complex tasks, this paper proposes integrating an external auxiliary diffusion model with a test-time adaptive method that uses cross-validation to construct reliable supervisory signals. Specifically, we condition on the text description of each test instance's predicted category and use a diffusion module to generate a corresponding calibrated instance. Based on such an instance, the following learning strategy is employed. First, the calibrated instance and its condition category are combined to form supervisory signals. Then, the prediction of the calibrated instance is compared with that of the test instance to select reliable signals. To mitigate instability in the category distribution of generated and selected instances, adaptive weights are applied to maintain diversity and balance in model predictions. Finally, based on the inverse process of diffusion, we construct the negative instance of the generated instance and introduce a robust contrastive learning approach to calibrate model optimization further. Extensive experimental results demonstrate that our method achieves state-of-the-art performance on several datasets. The ablation experiments are conducted to verify the effectiveness of each module.

2 Related Work

Test-time domain adaptation (TTA) has evolved toward practical scenarios in which only a pre-trained source model and unlabeled target data are available. Test-Time Training (TTT) (Sun et al. 2020; Zhang et al. 2024a) pioneered the idea of optimizing during test time, but it relied on source data and training loss, thereby limiting its practicality. To address this, Tent (Wang et al. 2020; Li et al. 2024) proposed 'Fully TTA,' enabling online adaptation to unlabeled target data without source data by updating normalization layers, laying a foundational framework.

Subsequent methods have further improved TTA. Entropy minimization approaches, such as EATA (Niu et al. 2022), introduce sample filtering for uncertain predictions and regularization to enhance stability. SAR (Niu et al. 2023) observed model collapse caused by trivial solutions and proposed adaptation to flat minima along with filtering large gradient samples. DeYO (Lee et al. 2024) added criteria for sample selection through image rearrangement, and COME (Zhang et al. 2024b) put forward conservative entropy minimization to tackle overconfidence.

Most early TTA methods focused on offline scenarios, whereas Continual Test-Time Adaptation (CTTA) (Wang et al. 2022; Qiao et al. 2024; Zhao et al. 2023) extends TTA to adapt to continuous domain shifts, emphasizing stability. CoTTA (Wang et al. 2022) addresses catastrophic forgetting with consistency loss between the base model and a weight-averaged model, along with stochastic parameter restoration from the source model. PETAL (Brahma and Rai 2023) uses a probabilistic framework and parameter restoration based on the Fisher Information Matrix.

ViDA (Liu et al. 2023) balances stability and plasticity by designing adapters that separate domain-invariant and domain-specific features. Continual-MAE (Liu et al. 2024) measures pixel uncertainty via Monte Carlo dropout to identify object presence and enhances domain-invariant feature representation with a masked autoencoder. However, many state-of-the-art CTTA methods adopt teacher-student frameworks or parameter restoration, leading to high computational costs and memory overhead, which conflicts with TTA's goal of real-time adaptation.

3 Proposed Method

This work considers a continual test-time domain adaptation setting, where a pre-trained model must adapt to a continually changing target domain online without access to source data. Consider a pre-trained model $F_\theta(x)$ with parameter θ trained on the source data. Unlabeled target domain data $\mathcal{X}_t$ is provided sequentially, and the data distribution continually changes. At testing stage t, when the unlabeled target data $\mathcal{X}_t = [x_t^1, ..., x_t^B]$ is sent to the model F_{θ_t}, where B is the number of samples. The model F_{θ_t} needs to make the prediction $P_t = [p_t^1, ..., p_t^B]$ and adapts itself accordingly for the next input ($\theta_t \rightarrow \theta_{t+1}$). It is worth noting that the total evaluation process is online, and the model only has access to the data $\mathcal{X}_t$ of the current stage t. Considering the complexity of this task, this paper proposes introducing a diffusion model to construct additional supervisory signals and utilizing contrastive learning to achieve robust optimization of the model. The framework is shown in Figure 1.

Instance Generation with Diffusion Model

For the convenience of expression, θ_t in the following mainly refers to the parameters of the proposed network at time t, and the prediction process can be denoted as follows.

$$p_{x_t^b} = \texttt{Softmax}(F_{\theta_t}(x_t^b)),\ H(x_t^b) = -p_{x_t^b} \log p_{x_t^b}, \quad (1)$$

where $p_{x_t^b}$ represents the classification result of the instance b at time t, and H represents the entropy, which is often

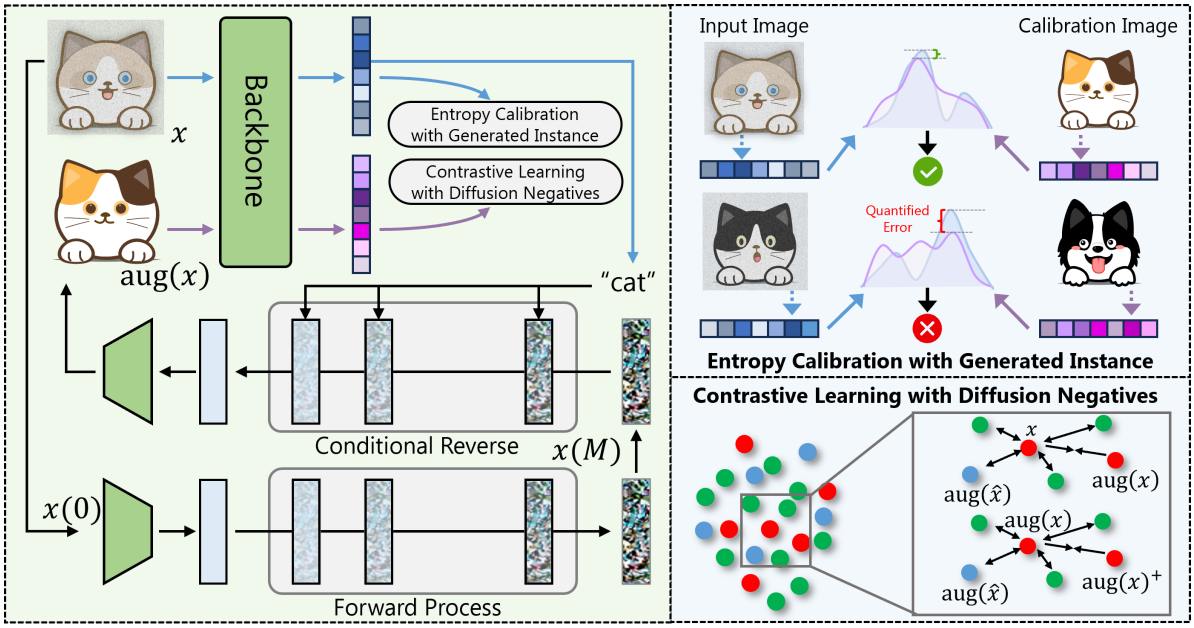

Figure 1: This is the flow of our method. We propose using a diffusion model to generate calibration instances for the test ones. Comparing the predictions for the test and calibration instances determines the reliability of the predictions for the test instances. We employ adaptive weighting to prevent model collapse caused by class imbalance. We use the inverse of the diffusion model to construct corresponding negative samples, while also implementing extended contrastive learning for different test instances.

used to supervise model optimization. However, such signals have many limitations. The output results based on the network are not entirely correct. Using them directly to supervise model optimization will generate a lot of noise. Long-term accumulation may lead the network to fall into a vicious cycle, resulting in model collapse. Existing methods employ optimization strategies, such as contrastive learning and denoising, to calibrate pseudo-labels; however, relying solely on the pre-trained model itself has little effect. To this end, this paper attempts to start with the diffusion generation model to inject reliable supervision signals into the TTA model.

Diffusion models involve a forward diffusion process and a reverse denoising process. Given the test instance x, the forward process gradually adds Gaussian noise, producing $x(m)$ as m increases from 0 until M, which can be formulated as:

$$x(m) = \sqrt{\alpha_m}x(0) + \sqrt{1-\alpha_m}\epsilon, \epsilon \sim \mathcal{N}(0,1), \quad (2)$$

where $\alpha_m = \prod_{i=1}^{m}(1-\beta_i)$ and $\{\beta_i\}_{i=0}^{M}$ denotes a fixed or learned variance schedule. In the following reverse process, noise is removed from $x(m)$ using a learned noise estimator $\epsilon_{\theta_d}(x(m), m, \mathcal{C})$ conditioned on $\mathcal{C}$.

$$x(m-1) = \sqrt{\frac{\alpha_{m-1}}{\alpha_m}}x(m) + \sigma_m\epsilon_m - \sqrt{\alpha_{m-1}}\psi\left(\alpha_m, \alpha_{m-1}, \sigma_m\right)\hat{\epsilon}_{\theta_d}\left(x(m), m, \mathcal{C}\right), \quad (3)$$

where $\psi(\alpha_m, \alpha_{m-1}, \sigma_m)$ denotes the constant schedule that depends on the fixed parameters α_m, α_{m-1}, and σ_m. $\hat{\epsilon}_{\theta_d}(x(m), m, \mathcal{C}) = \epsilon_{\theta_d}(x(m), m) + \gamma[\epsilon_{\theta_d}(x(m), m, \mathcal{C}) - \epsilon_{\theta_d}(x(m), m)]$. Here, $\epsilon_\theta(x(m), m)$ represents the diffusion model without condition, and γ and σ_m control the strength of conditional guidance and random noise ϵ_m, respectively. Then, the definitions of $\epsilon_{\theta_d}(x(m), m)$ and $\epsilon_{\theta_d}(x(m), m, \mathcal{C})$ are as follows.

$$\epsilon_\theta\left(x(m), m\right) = -\sqrt{1-\overline{\alpha}_m}\nabla_{x(m)}\log p_{\theta_d}\left(x(m)\right), \quad (4)$$

$$\epsilon_\theta\left(x(m), m, \mathcal{C}\right) = -\sqrt{1-\overline{\alpha}_m}\nabla_{x(m)}\log p_{\theta_d}\left(x(m)|\mathcal{C}\right), \quad (5)$$

where $\overline{\alpha}_m = \prod_{i=1}^{m}\alpha_i$ and p_{θ_d} denotes the data distribution parameterized by the diffusion model θ_d. ∇ is the gradient, which denotes the direction that maximizes the corresponding likelihood.

For the test instance x_t^b, we first construct the noise distribution $x_t^b(M)$ with Eq. 2. Then we generate the $\text{aug}(x_t^b)$ with Eq. 3 under the condition $\mathcal{C}(x_t^b)$. Given the absence of labels during test-time tuning, we use the predicted category of the test instance to construct the corresponding category text description ($\mathcal{C}(x_t^b) \leftarrow$ A photo of a [CLASS]), where [CLASS] represents text description of the predicted category $\arg\max(p_{x_t^b})$ of the test instance.

Entropy Calibration with Generated Instance

After obtaining the generated instance of the corresponding category, we constructed the following supervisory signals. First, considering the powerful capabilities of the diffusion model, the generated instance and the corresponding category can be directly used to optimize the model.

$$p_{\text{aug}(x_t^b)} = \texttt{Softmax}(F_{\theta_t}(\text{aug}(x_t^b))),$$
$$\mathcal{L}_{\text{CE1}}(\text{aug}(x_t^b)) = -y_{\text{aug}(x_t^b)}\log p_{\text{aug}(x_t^b)}, \quad (6)$$

where $y_{\text{aug}(x_t^b)} = \text{one\_hot}(\arg\max(p_{x_t^b}))$. Then, the prediction results of the generated instance and the original instance will form a calibration instance. Thanks to the powerful generative capabilities of the existing pre-trained diffusion model, we can easily generate instances of the corresponding category. By comparing the prediction results of the generated instances with those of the test instances, we can determine whether the prediction of the test instance is accurate. Simply put, when the prediction category of the generated instance is consistent with that of the original instance and the entropy value is close, we believe that the prediction result of the original instance is relatively reliable. Otherwise, it means that the prediction result of the original instance may be wrong or cannot be used to optimize the model.

$$\mathcal{L}_{\text{CE2}}(x_t^b) = -y_{x_t^b} \log p_{x_t^b} \quad x_b^t \in \mathcal{S}_t, \tag{7}$$

where $y_{x_t^b} = p_{x_t^b}$, and $\mathcal{S}_t = [x_t^b | \arg\max(p_{x_t^b}) = \arg\max(p_{\text{aug}(x_t^b)}), \quad |\max(p_{x_t^b}) - \max(p_{\text{aug}(x_t^b)})| < \delta]$. Based on this, we can construct a generated instance of the same prediction category for each test instance and then determine whether the prediction of the test instance is reliable by comparing the prediction results of the generated instance with those of the test instance.

Diversity with Prior Distribution. To mitigate the category imbalance caused by the above loss, since the categories of generated and selected instances are uncontrollable, we employ mean shift-based diversity weights to maintain the balance of supervision signals, thereby preventing model collapse. Our objective is to learn a set of weights to assess the importance of various supervisory signals. Therefore, we introduce a diversity criterion (Marsden, Döbler, and Yang 2023) to ensure that diverse samples are favored over those similar to the central tendency of recent model predictions. The diversity weighting is employed by tracking the recent tendency of a model's prediction with an exponential moving average.

$$\bar{y}_t = \alpha \bar{y}_{t-1} + \frac{1-\alpha}{B} \sum_{i=1}^{B} y_{\text{aug}(x_{t-1}^b)} + \frac{1-\alpha}{|\mathcal{S}_{t-1}|} \sum_{\mathcal{S}_{t-1}} y_{x_{t-1}^b}, \tag{8}$$

where $\alpha = 0.9$. To determine a diversity weight for each test sample, the cosine similarity between the current model output $y_{x_t^b}, y_{\text{aug}(x_t^b)}$ and the tendency of the recent output $\bar{y}_t$ is calculated as follows.

$$\begin{aligned} u_{x_t^b} &= 1 - \frac{\bar{y}_t^\top y_{x_t^b}}{\|y_{x_t^b}\| \|\bar{y}_t\|}, \\ u_{\text{aug}(x_t^b)} &= 1 - \frac{\bar{y}_t^\top y_{\text{aug}(x_t^b)}}{\|y_{\text{aug}(x_t^b)}\| \|\bar{y}_t\|}, \end{aligned} \tag{9}$$

u has the advantage that if the model output is uniform, uncertain predictions receive a smaller weight, mitigating errors in the model. More importantly, certainty weighting based on negative entropy is employed to avoid bias towards specific classes.

$$v_{x_t^b} = y_{x_t^b} \log y_{x_t^b}. \tag{10}$$

Since conditions [CLASS] directly generate the labels of generated instances, there is no need to consider their diversity. We normalize the certainty and diversity weights to be within the unit range, and exponentiate the product of diversity and certainty weights, scaled by a temperature τ. Thus, the weight of each sample can be obtained.

$$w_{\text{aug}(x_t^b)} = \exp\left(\frac{u_{\text{aug}(x_t^b)}}{\tau}\right), \quad w_{x_t^b} = \exp\left(\frac{u_{x_t^b} \cdot v_{x_t^b}}{\tau}\right). \tag{11}$$

After the above selection, calibration, and weighting, the objective can be reconstructed as follows.

$$\begin{aligned} \mathcal{L}_{WCE} = & \frac{1}{|\mathcal{X}_t|} \sum_{\mathcal{X}_t} w_{\text{aug}(x_t^b)} \mathcal{L}_{\text{CE1}}(\text{aug}(x_t^b)) \\ & \frac{1}{|\mathcal{S}_t|} \sum_{\mathcal{S}_t} w_{x_t^b} \mathcal{L}_{\text{CE2}}(x_t^b). \end{aligned} \tag{12}$$

Contrastive Learning with Diffusion Negatives

We design a contrastive learning framework to enhance the model's discriminative ability further. Specifically, to enhance the discrimination ability, the positive-negative pairs should provide effective contrast. Thus, negative instances should belong to distinct semantic classes, differing significantly from their corresponding positive instances while sharing similar visual traits.

For each augmented instance, we want to construct a corresponding negative instance. Specifically, we erase the class to which the extended example belongs from the test sample. The essence of this process is to minimize the probability that the instance belongs to the class to which the positive example belongs. Inspired by Eq. 3, we introduce a conditional inverse diffusion process, which uses a conditional Denoising Diffusion Model to map the real instance to a noise vector instead of a random vector.

$$\begin{aligned} \hat{x}(m) = & \sqrt{\alpha_m} \psi\left(\alpha_m, \alpha_{m-1}, 0\right) \epsilon_{\theta_d}\left(\hat{x}(m-1), m, \mathcal{C}\right) \\ & + \sqrt{\frac{\alpha_m}{\alpha_{m-1}}} \hat{x}(m-1). \end{aligned} \tag{13}$$

For the test instance x_t^b, we first construct the noise distribution $\text{aug}(\hat{x}_t^b(M))$ with Eq. 13 under the condition $\mathcal{C}(x_t^b)$, where $\text{aug}(\hat{x}_t^b(0)) = x_t^b$. Then, we preserve visual characteristics in the unconditional reverse process. Hence, we utilize unconditional diffusion model to reverse the conditional noise vector $\text{aug}(\hat{x}_t^b(M))$ into a new image $\text{aug}(\hat{x}_t^b)$, which represents negative instance that is different from category of $\text{aug}(x_t^b)$.

$$\begin{aligned} \hat{x}(m-1) = & \sqrt{\frac{\alpha_{m-1}}{\alpha_m}} \hat{x}(m) + \sigma_m \epsilon_m \\ & - \sqrt{\alpha_{m-1}} \psi\left(\alpha_m, \alpha_{m-1}, \sigma_m\right) \epsilon_{\theta_d}\left(\hat{x}(m), m\right). \end{aligned} \tag{14}$$

In this process, $\hat{x}(m)$ is formulated as Eq. 14, derived from Eq. 3 with $\gamma = 0$. To mitigate potential instance degradation caused by semantic removal and preserve visual fidelity, we introduce random noise with $\sigma_m = 0.2$.

Based on these generated instances, we construct the extended contrastive loss as follows.

$$\mathcal{L}_{CL} = -\frac{\sum_{x_b^t \in \{\mathcal{X}_t \setminus \mathcal{S}_t\}} \log \frac{\exp(z_{x_t^b} \cdot z^+_{x_t^b})}{\sum_{x_t^d \in \mathcal{X}_t} \exp(z_{x_t^b} \cdot z_{x_t^d}))}}{|\mathcal{X}_t \setminus \mathcal{S}_t|}$$
$$-\frac{\sum_{x_b^t \in \mathcal{S}_t} \log \frac{\exp(z_{x_t^b} \cdot z_{\text{aug}(x_t^b)})}{\sum_{x_t^d \in \mathcal{X}_t} \exp(z_{x_t^b} \cdot z_{x_t^d}) + \exp(z_{x_t^b} \cdot z_{\text{aug}(\hat{x}_t^b)})}}{|\mathcal{S}_t|}$$
$$-\frac{\sum_{x_b^t \in \mathcal{X}_t} \log \frac{\exp(z_{\text{aug}(x_t^b)} \cdot z^+_{\text{aug}(x_t^b)})}{\sum_{x_t^d \in \mathcal{X}_t} \exp(z_{\text{aug}(x_t^b)} \cdot z_{\text{aug}(x_t^d)}) + \exp(z_{\text{aug}(x_t^b)} \cdot z_{\text{aug}(\hat{x}_t^b)})}}{|\mathcal{X}_t|}, \tag{15}$$

where $z_{x_t^b} = F_{\theta_t}(x_t^b)$. Here, we further construct calibration negative instances to enhance the model's discriminative capabilities for reliable instances. For test instances with lower confidence, conventional expansion methods, such as cropping, represented as $+$, are utilized to optimize the distribution of both test and generated instances.

Overall

The overall objective of our method is as follows.

$$\mathcal{L}_O = \mathcal{L}_{WCE} + \lambda \mathcal{L}_{CL}, \tag{16}$$

where λ is the hyperparameter. In general, we do not directly use the results of the adapted models as supervision signals; instead, we apply the diffusion model as prior knowledge to calibrate pseudo-labels and design an adaptive weighted method to prevent excessive parameter deviation.

4 Experiments

In this section, we evaluate the effectiveness of the proposed method on three benchmark datasets in terms of 1) whether our cliabrate module learns meaningful results, 2) whether the proposed contrastive strategies can improve the discrimination, and 3) the parameter analysis of the proposed method.

Datasets

We adopt CIFAR10, CIFAR100, and ImageNet as the source domain datasets, and CIFAR10C, CIFAR100C, and ImageNet-C as the corresponding target domain datasets, respectively. The target domain datasets were created to evaluate the robustness of classification networks (Hendrycks and Dietterich 2019). Each target domain dataset contains 15 types of corruption with five levels of severity. Following (Wang et al. 2022), for each corruption, we use 10000 images for both CIFAR10C and CIFAR100C datasets and 5000 images for ImageNet-C.

Implementation Details

Following (Wang et al. 2022), the corrupted images are provided to the network online, meaning these images can be utilized to update the model only once during the adaptation process. In addition, unlike traditional test-time adaptation methods, which adapt to each data corruption type individually, we adjust the source model sequentially for each data corruption type. We evaluate the adaptation performance immediately after encountering each type of corrupted data. The total type of corruption is 15, and the corruption level is set to the highest level of 5 (except for the gradual experiments on CIFAR10-to-CIFAR10C). All experiments utilize the pre-trained Stable Diffusion 2.0 model (Rombach et al. 2022) as the DPM generator, without further optimization.

In our experiments, we adhere to the implementation details outlined in previous works (Wang et al. 2022) to ensure consistency and comparability. For the classification CTTA, we employ ViT-base and ResNet (Zagoruyko and Komodakis 2016) as the backbone. In the case of ViT-base, we resize the input images to 224x224, while maintaining the original image resolution for other backbones. For experiments involving ImageNet-to-ImageNet-C, we conduct trials under ten diverse corruption orders.

Baselines

We compare our method with several state-of-the-art continual test-time adaptation algorithms, the details of these methods are as follows: 1) **Source** directly uses the pre-trained model for adaptation without any specific method for domain adaptation; 2) **BN Stats Adapt** keeps the pre-trained model weights and uses the Batch Normalization statistics from the input data of the input batch for the prediction (Li et al. 2016; Schneider et al. 2020); 3) **Pseudo-Label** (Lee et al. 2013) picks up the class which has the maximum predicted probability as the pseudo-labels to update the model; 4) **TENT** (Wang et al. 2020) reduces generalization error by reducing the entropy of model predictions on test data, TENT-continual is a continual learning version of TENT; 5) **CoTTA** (Wang et al. 2022) reduces the error accumulation by using weight-averaged and augmentation-averaged predictions and avoids catastrophic forgetting by stochastically restoring a small part of the source pre-trained weights; 6) **NOTE** (Gong et al. 2022) adopts an Instance-Aware Batch Normalization to correct normalization for out-of-distribution samples; 7) **RoTTA** (Yuan, Xie, and Li 2023) presents a robust batch normalization scheme to estimate the normalization statistics; 8) **RMT** (Döbler, Marsden, and Yang 2023) uses symmetric cross-entropy and contrastive learning to pull the test feature space closer to the source domain; 9) **ROID** (Marsden, Döbler, and Yang 2023) proposes to continually weight-average the source and adapted model, and an adaptive additive prior correction scheme; 10) **REM** (Han, Na, and Hwang 2025) addresses prediction consistency by aligning probability distributions across varying difficulty levels while preserving entropy rank orders.

Performance Evaluation

CIFAR10-to-CIFAR10C. Table 1 shows the classification error rate for the standard CIFAR10-to-CIFAR10C task. We compare our method with the eight baseline methods. 'Gain' represents the percentage of improvement in model accuracy compared with the source method. CoTTA considers error accumulation to improve performance further. As the latest

Time		$t \longrightarrow$																
Method	Backbone	*Gaussian*	*Shot*	*Impulse*	*Defocus*	*Glass*	*Motion*	*Zoom*	*Snow*	*Frost*	*Fog*	*Brightness*	*Contrast*	*Elastic_trans*	*Pixelate*	*Jpeg*	Mean	Gain
Source	ResNet	72.3	65.7	72.9	46.9	54.3	34.8	42.0	25.1	41.3	26.0	9.3	46.7	26.6	58.5	30.3	43.5	-
BN Stats Adapt		28.1	26.1	36.3	12.8	35.3	14.2	12.1	17.3	17.4	15.3	8.4	12.6	23.8	19.7	27.3	20.4	+23.1
Pseudo-Label		26.7	22.1	32.0	13.8	32.2	15.3	12.7	17.3	17.3	16.5	10.1	13.4	22.4	18.9	25.9	19.8	+23.7
TENT-continual [ICLR'21]		24.8	20.5	28.5	14.5	31.7	16.2	15.0	19.2	17.6	17.4	11.4	16.3	24.9	21.6	26.0	20.4	+23.1
CoTTA [CVPR'22]		24.6	21.9	26.5	11.9	27.8	12.4	10.6	15.2	14.4	12.8	**7.4**	11.1	18.7	13.6	17.8	16.5	+27.0
NOTE [NeurIPS'22]		**7.3**	**7.4**	**12.5**	20.9	**13.8**	15.5	34.2	34.2	39.6	25.0	11.6	24.2	29.9	14.1	**12.7**	20.1	+23.4
RoTTA [CVPR'23]		30.3	25.4	34.6	18.3	34.0	14.7	11.0	16.4	14.6	14.0	8.0	12.4	20.3	16.8	19.4	19.3	+24.2
RMT [CVPR'23]		24.1	20.2	25.7	13.2	25.5	14.7	12.8	16.2	15.4	14.6	10.8	14.0	18.0	14.1	16.6	17.0	+26.5
ROID [WACV'24]		23.7	18.7	26.4	11.5	28.1	12.4	10.1	14.7	14.3	12.0	7.5	9.3	19.8	14.5	20.3	16.2	+27.3
Ours		18.2	14.2	18.5	**11.1**	22.5	**10.8**	**9.6**	**11.5**	**12.8**	**10.1**	8.5	**9.0**	**17.5**	**11.0**	14.8	**13.4**	+30.2
Source	ViT-base	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	-
CoTTA [CVPR'22]		58.7	51.3	33.0	20.1	34.8	20.0	15.2	11.1	11.3	18.5	4.0	34.7	18.8	19.0	17.9	24.6	+3.6
VDP [AAAI'23]		57.5	49.5	31.7	21.3	35.1	19.6	15.1	10.8	10.3	18.1	4.0	27.5	18.4	22.5	19.9	24.1	+4.1
ViDA [ICLR'24]		52.9	47.9	19.4	11.4	31.3	13.3	7.6	7.6	9.9	12.5	3.8	26.3	14.4	33.9	18.2	20.7	+7.5
ROID [WACV'24]		20.8	14.5	10.5	9.3	20.3	10.2	8.3	7.9	7.4	9.6	4.1	9.2	13.0	10.9	15.5	11.4	+16.8
REM [ICML'25]		17.3	12.5	10.3	8.4	17.7	8.4	**5.5**	6.6	5.6	7.2	3.7	6.4	11.0	**7.3**	13.0	9.4	+18.8
Ours		**16.1**	**11.0**	**9.8**	**8.4**	**15.2**	**8.0**	5.8	**6.2**	**5.3**	**7.0**	**3.2**	**5.4**	**10.6**	7.4	**12.2**	**8.7**	+19.5

Table 1: Classification error rate (%) for the standard CIFAR10-to-CIFAR10C continual test-time adaptation task. All results are evaluated with the largest corruption severity level 5 in an online fashion. Bold text indicates the best performance.

Time		$t \longrightarrow$																
Method	Backbone	*Gaussian*	*Shot*	*Impulse*	*Defocus*	*Glass*	*Motion*	*Zoom*	*Snow*	*Frost*	*Fog*	*Brightness*	*Contrast*	*Elastic_trans*	*Pixelate*	*Jpeg*	Mean	Gain
Source	ResNet	73.0	68.0	39.4	29.3	54.1	30.8	28.8	39.5	45.8	50.3	29.5	55.1	37.2	74.7	41.2	46.4	-
BN Stats Adapt		42.1	40.7	42.7	27.6	41.9	29.7	27.9	34.9	35.0	41.5	26.5	30.3	35.7	32.9	41.2	35.4	+11.0
Pseudo-Label		38.1	36.1	40.7	33.2	45.9	38.3	36.4	44.0	45.6	52.8	45.2	53.5	60.1	58.1	64.5	46.2	+0.2
TENT-continual [ICLR'21]		37.2	35.8	41.7	37.7	50.9	48.5	48.5	58.2	63.2	71.4	72.0	83.1	88.6	91.6	95.1	61.6	-15.2
CoTTA [CVPR'22]		40.1	37.7	39.7	26.8	38.0	27.9	26.5	32.9	31.7	40.4	24.6	26.8	32.5	28.1	33.8	32.5	+13.9
NOTE [NeurIPS'22]		**28.4**	32.7	36.4	44.4	42.9	42.2	65.8	61.1	70.8	51.6	34.4	45.4	62.7	39.9	36.4	43.3	+3.1
RoTTA [CVPR'23]		49.1	44.9	45.5	30.2	42.7	29.5	26.1	32.2	30.7	37.5	24.7	29.1	32.6	30.4	36.7	34.8	+11.6
RMT [CVPR'23]		40.2	36.2	36.0	27.9	33.9	28.4	26.4	28.7	28.8	31.1	25.5	27.1	28.0	26.6	29.0	30.2	+16.2
ROID [WACV'24]		36.5	31.9	33.2	24.9	34.9	26.8	**24.3**	28.9	28.5	31.1	**22.8**	24.2	30.7	26.5	34.4	29.3	+17.1
Ours		32.7	**30.1**	**29.9**	**23.6**	**28.8**	**26.1**	24.5	**26.1**	**26.5**	**30.1**	24.2	**22.1**	**25.2**	**24.8**	**26.1**	**26.7**	+19.7
Source	ViT-base	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	-
CoTTA [CVPR'22]		55.0	51.3	25.8	24.1	59.2	28.9	21.4	21.0	24.7	34.9	11.7	31.7	40.4	55.7	35.6	34.8	+0.6
VDP [AAAI'23]		54.8	51.2	25.6	24.2	59.1	28.8	21.2	20.5	23.3	33.8	**7.5**	**11.7**	32.0	51.7	35.2	32.0	+3.4
ViDA [ICLR'24]		50.1	40.7	22.0	21.2	45.2	21.6	**16.5**	17.9	16.6	25.6	11.5	29.0	29.6	34.7	27.1	27.3	+8.1
ROID [WACV'24]		45.7	32.2	20.5	22.2	37.8	24.6	17.2	16.8	15.8	23.2	10.6	28.3	29.1	33.2	26.2	25.6	+9.8
REM [ICML'25]		**29.2**	**25.5**	**17.0**	19.1	35.2	21.2	18.3	19.5	18.7	22.8	15.5	17.6	31.6	**26.2**	33.0	23.4	+12.0
Ours		30.1	25.9	18.1	**18.9**	**33.8**	**20.5**	17.5	**15.6**	**15.7**	**22.1**	10.8	24.3	**27.7**	32.1	**24.2**	**22.4**	+13.0

Table 2: Classification error rate (%) for the standard CIFAR100-to-CIFAR100C continual test-time adaptation task. All results are evaluated with the largest corruption severity level 5 in an online fashion. Bold text indicates the best performance.

proposed method, NOTE aims to enhance the model's performance across various domains compared to the distribution with BN. Although it performs well in domains such as Gaussian and shot, it performs poorly in some simple domains, such as Brightness and Contrast. ROID and REM have dramatically improved the overall performance of the model. However, the model does not perform well in some difficult domains due to the limited parameters that can be learned. Compared to all the previous methods, our method achieves the best results in terms of average error value and most types of corrupted data under different backbones. It is worth noting that, currently, fine-tuning the BN layer of the Transformer network is an ideal learning strategy. Both REM and our method achieved good performance, but our method is superior.

CIFAR100-to-CIFAR100C. Table 2 shows the classification error rate for the standard CIFAR100-to-CIFAR100C task. In the ResNet, BN Stats Adapt and NOTE do not bring error accumulation, but there is little room for improvement. CoTTA considers the error accumulation problem and reduces the error to 32.5%. Furthermore, our method outperforms REM and ROID on several types of corrupted data, with an average error reduction of 26.7%. ViT-base remains our first choice, as its overall performance is superior to that of ResNet, and our method continues to outperform existing learning strategies.

ImageNet-to-ImageNet-C. We also make experiments on the ImageNet dataset. Following (Wang et al. 2022), we conduct ImageNet-to-ImageNet-C experiments over ten diverse corruption type sequences in severity level 5. The average result of ten experiments is shown in Table 3. ImageNetC is more complex than CIFAR100C and CIFAR10C, and the overall average test error is more significant. Our method outperforms other competing methods, reducing the

Time		$t \longrightarrow$															
Method	Backbone	*Gaussian*	*Shot*	*Impulse*	*Defocus*	*Glass*	*Motion*	*Zoom*	*Snow*	*Frost*	*Fog*	*Brightness*	*Contrast*	*Elastic_trans*	*Pixelate*	*Jpeg*	Mean	Gain
Source	ResNet	97.8	97.1	98.2	81.7	89.8	85.2	78.0	83.5	77.0	75.9	41.3	94.5	82.5	79.3	68.5	82.0	-
CoTTA [CVPR'22]		84.5	82.0	80.4	81.8	79.5	69.2	58.8	60.8	61.1	48.5	36.5	67.5	47.8	41.8	45.9	63.1	+18.9
RoTTA [CVPR'23]		88.3	82.8	82.1	91.3	83.7	72.9	59.4	66.2	64.3	53.3	35.6	74.5	54.3	48.2	52.6	67.3	+14.7
RMT [CVPR'23]		79.9	76.3	73.1	75.7	72.9	64.7	56.8	56.4	58.3	49.0	40.6	58.2	47.8	43.7	44.8	59.9	+22.1
ViDA [ICLR'24]		79.3	74.7	73.1	76.9	74.5	65.0	56.4	59.8	62.6	49.6	38.2	66.8	49.6	43.1	46.2	61.2	+20.8
ROID [WACV'24]		71.7	62.2	62.2	69.6	66.5	57.1	49.3	52.3	57.4	43.5	33.4	59.1	45.4	41.8	46.2	54.5	+27.5
Ours		**68.2**	**58.1**	**60.3**	**62.2**	**53.5**	**52.7**	**46.2**	**50.1**	**49.2**	**38.9**	**33.6**	**55.2**	**42.9**	**39.7**	**42.5**	**50.2**	+31.8
Source	ViT-base	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	-
CoTTA [CVPR'22]		52.9	51.6	51.4	68.3	78.1	57.1	62.0	48.2	52.7	55.3	25.9	90.0	56.4	36.4	35.2	54.8	+1.0
VDP [AAAI'23]		52.7	51.6	50.1	58.1	70.2	56.1	58.1	42.1	46.1	45.8	23.6	70.4	54.9	34.5	36.1	50.0	+5.8
ViDA [ICLR'24]		47.7	42.5	42.9	52.2	56.9	45.5	48.9	38.9	42.7	40.7	24.3	52.8	49.1	33.5	33.1	43.4	+12.4
ROID [WACV'24]		57.6	51.5	52.2	55.1	52.4	46.5	47.2	45.6	39.5	36.0	26.0	45.0	43.8	39.7	36.3	45.0	+10.8
REM [ICML'25]		43.5	**38.1**	39.2	53.2	**49.0**	43.5	**42.8**	37.5	35.2	**35.4**	23.2	46.8	41.6	**28.9**	30.2	39.2	+16.6
Ours		**42.2**	40.7	**38.2**	**48.3**	51.2	**41.0**	43.3	**34.9**	**34.0**	36.3	**20.2**	**44.9**	**41.1**	30.2	**28.8**	**38.3**	+17.5

Table 3: Average error of standard ImageNet-to-ImageNet-C experiments over 10 diverse corruption sequences. All results are evaluated with the largest corruption severity level 5 in an online fashion. Bold text indicates the best performance.

Method	Mean	Gain
CE	51.5	-
CE + CE1	46.8	+4.7
WCE	43.2	+8.3
WCE+CL w/ $\mathcal{X}$	41.8	+9.7
WCE+CL w/ $\mathcal{X} \setminus \mathcal{S}$	42.5	+9.0
WCE+CL w/ $\mathcal{S}$	41.2	+10.3
WCE+CL	38.3	+13.2

Table 4: Ablation experiments of the supervision signals for the ImageNet-to-ImageNet-C task.

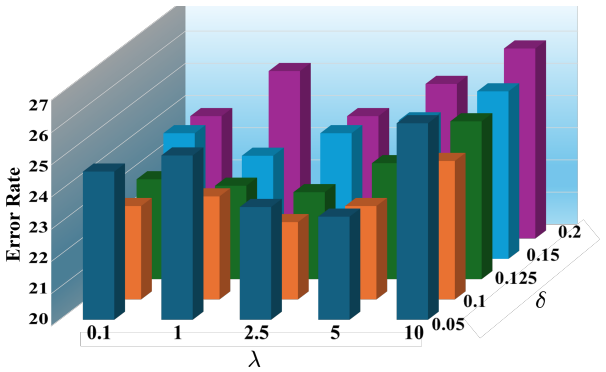

Figure 2: Parameters Analysis on CIFAR100-CIFAR100C dataset.

average test error to 50.2% and 38.3% with ResNet and ViT networks, respectively.

The improvement from our proposed model on CIFAR100C and CIFAR10C is not as significant as on ImageNet. The primary reason is that as the complexity of the category increases, the disadvantages of the limited learning ability to fine-tune normalization layers become increasingly apparent in the model. However, we still achieved extremely competitive results through diffusion augmentation calibration.

Ablation Studies

We first conduct ablation experiments using the same supervision signals to demonstrate the effectiveness of the proposed framework, as shown in Table 4. For the convenience of expression, 'CE' represents the cross-entropy loss. Then, such a module is combined with label-generated instance cross-entropy loss (CE1) and reliable instance supervision, respectively (CE2). Ultimately, these three will form a versatile supervisory signal generator (WCE). The diversity of the prior distribution is vital to the model, which was used to optimize the model throughout the process; otherwise, the model would collapse. CL is the contrastive learning module, which significantly improves model performance.

Parameters Analysis

We explored how the model varies with the relaxation factor δ and λ, and the results are shown in Figure 2. The results demonstrate that our method is not sensitive to δ in the range $[0.05, 0.1]$, and choosing appropriate parameters is crucial for effective contrastive learning.

5 Conclusion

This paper proposes integrating an external auxiliary diffusion model with a test-time adaptive method that uses cross-validation to construct reliable supervisory signals. We condition on the text description of each test instance's predicted category and use a diffusion module to generate a corresponding calibrated instance. The calibrated instance and its condition category are combined to form supervisory signals, which are compared with those of the test instance to select reliable signals. Finally, based on the inverse process of diffusion, we construct a negative instance from the generated instance and introduce a robust contrastive learning approach to calibrate model optimization further.

Acknowledgements

This work is supported in part by the National Key Research and Development Program of China (No. 2023YFC3305600), Joint Fund of Ministry of Education of China (8091B022149, 8091B02072404), National Natural Science Foundation of China (62132016, 62571412, and 62571393), Key Research and Development Program of Shaanxi (2024GX-YBXM-127) and National Key Laboratory Foundation of China (Grant No. HTKJ2024KL504011).

References

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

Chen, D.; Wang, D.; Darrell, T.; and Ebrahimi, S. 2022. Contrastive Test-Time Adaptation. *arXiv preprint arXiv:2204.10377*.

Döbler, M.; Marsden, R. A.; and Yang, B. 2023. Robust mean teacher for continual and gradual test-time adaptation. In *CVPR*, 7704–7714.

Gan, Y.; Ma, X.; Lou, Y.; Bai, Y.; Zhang, R.; Shi, N.; and Luo, L. 2022. Decorate the Newcomers: Visual Domain Prompt for Continual Test Time Adaptation. *arXiv preprint arXiv:2212.04145*.

Gong, T.; Jeong, J.; Kim, T.; Kim, Y.; Shin, J.; and Lee, S.-J. 2022. NOTE: Robust Continual Test-time Adaptation Against Temporal Correlation. In *NeurIPS*.

Han, J.; Na, J.; and Hwang, W. 2025. Ranked Entropy Minimization for Continual Test-Time Adaptation. *arXiv preprint arXiv:2505.16441*.

Hendrycks, D.; and Dietterich, T. 2019. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*.

Lee, D.-H.; et al. 2013. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *ICMLW*, volume 3, 896.

Lee, J.; Jung, D.; Lee, S.; Park, J.; Shin, J.; Hwang, U.; and Yoon, S. 2024. Entropy is not enough for test-time adaptation: From the perspective of disentangled factors. *arXiv preprint arXiv:2403.07366*.

Li, H.; Hu, P.; Zhang, Q.; Peng, X.; Liu, X.; and Yang, M. 2024. Test-time Adaptation for Cross-modal Retrieval with Query Shift. *arXiv preprint arXiv:2410.15624*.

Li, R.; Jiao, Q.; Cao, W.; Wong, H.-S.; and Wu, S. 2020. Model adaptation: Unsupervised domain adaptation without source data. In *CVPR*, 9641–9650.

Li, Y.; Wang, N.; Shi, J.; Liu, J.; and Hou, X. 2016. Revisiting batch normalization for practical domain adaptation. *arXiv preprint arXiv:1603.04779*.

Liu, J.; Xu, R.; Yang, S.; Zhang, R.; Zhang, Q.; Chen, Z.; Guo, Y.; and Zhang, S. 2024. Continual-mae: Adaptive distribution masked autoencoders for continual test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 28653–28663.

Liu, J.; Yang, S.; Jia, P.; Lu, M.; Guo, Y.; Xue, W.; and Zhang, S. 2023. ViDA: Homeostatic Visual Domain Adapter for Continual Test Time Adaptation. *arXiv preprint arXiv:2306.04344*.

Liu, Y.; Zhang, W.; and Wang, J. 2021. Source-free domain adaptation for semantic segmentation. In *CVPR*, 1215–1224.

Marsden, R. A.; Döbler, M.; and Yang, B. 2023. Universal Test-time Adaptation through Weight Ensembling, Diversity Weighting, and Prior Correction. *arXiv preprint arXiv:2306.00650*.

Niu, S.; Wu, J.; Zhang, Y.; Chen, Y.; Zheng, S.; Zhao, P.; and Tan, M. 2022. Efficient test-time model adaptation without forgetting. In *ICML*, 16888–16905. PMLR.

Niu, S.; Wu, J.; Zhang, Y.; Wen, Z.; Chen, Y.; Zhao, P.; and Tan, M. 2023. Towards stable test-time adaptation in dynamic wild world. *arXiv preprint arXiv:2302.12400*.

Prabhu, V.; Khare, S.; Kartik, D.; and Hoffman, J. 2021. Sentry: Selective entropy optimization via committee consistency for unsupervised domain adaptation. In *ICCV*, 8558–8567.

Qiao, J.; Tan, X.; Chen, C.; Qu, Y.; Peng, Y.; Xie, Y.; et al. 2024. Prompt gradient projection for continual learning. In *The Twelfth International Conference on Learning Representations*.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695.

Schneider, S.; Rusak, E.; Eck, L.; Bringmann, O.; Brendel, W.; and Bethge, M. 2020. Improving robustness against common corruptions by covariate shift adaptation. *NeurIPS*, 33: 11539–11551.

Sun, Y.; Wang, X.; Liu, Z.; Miller, J.; Efros, A.; and Hardt, M. 2020. Test-time training with self-supervision for generalization under distribution shifts. In *International conference on machine learning*, 9229–9248. PMLR.

Wang, D.; Shelhamer, E.; Liu, S.; Olshausen, B.; and Darrell, T. 2020. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726*.

Wang, Q.; Fink, O.; Van Gool, L.; and Dai, D. 2022. Continual Test-Time Domain Adaptation. *arXiv preprint arXiv:2203.13591*.

Wang, X.; Peng, D.; Hu, P.; Gong, Y.; and Chen, Y. 2023a. Cross-Domain Alignment for Zero-Shot Sketch-Based Image Retrieval. *IEEE Trans. Circ. Syst. Video Tech.*

Wang, X.; Peng, D.; Yan, M.; and Hu, P. 2023b. Correspondence-Free Domain Alignment for Unsupervised Cross-Domain Image Retrieval. *arXiv preprint arXiv:2302.06081*.

Yang, M.; Li, Y.; Zhang, C.; Hu, P.; and Peng, X. 2024a. Test-time adaptation against multi-modal reliability bias. In *The twelfth international conference on learning representations*.

Yang, S.; Wang, Y.; van de Weijer, J.; Herranz, L.; and Jui, S. 2021. Generalized source-free domain adaptation. In *ICCV*, 8978–8987.

Yang, X.; Chen, X.; Li, M.; Wei, K.; and Deng, C. 2024b. A versatile framework for continual test-time domain adaptation: Balancing discriminability and generalizability. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 23731–23740.

Yang, X.; Gu, Y.; Wei, K.; and Deng, C. 2023. Exploring safety supervision for continual test-time domain adaptation. In *IJCAI*, 1649–1657.

Yang, X.; Li, M.; Yin, J.; Wei, K.; and Deng, C. 2024c. Navigating continual test-time adaptation with symbiosis knowledge. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, 5326–5334.

Yuan, L.; Xie, B.; and Li, S. 2023. Robust test-time adaptation in dynamic scenarios. In *CVPR*, 15922–15932.

Zagoruyko, S.; and Komodakis, N. 2016. Wide residual networks. *arXiv preprint arXiv:1605.07146*.

Zhang, J.; Wang, Y.; Yang, X.; Wang, S.; Feng, Y.; Shi, Y.; Ren, R.; Zhu, E.; and Liu, X. 2024a. Test-time training on graphs with large language models (llms). In *Proceedings of the 32nd ACM International Conference on Multimedia*, 2089–2098.

Zhang, Q.; Bian, Y.; Kong, X.; Zhao, P.; and Zhang, C. 2024b. COME: Test-time adaption by conservatively minimizing entropy. *arXiv preprint arXiv:2410.10894*.

Zhao, Z.; Zhang, Z.; Tan, X.; Liu, J.; Qu, Y.; Xie, Y.; and Ma, L. 2023. Rethinking gradient projection continual learning: Stability/plasticity feature space decoupling. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 3718–3727.