A Versatile Framework for Continual Test-Time Domain Adaptation: Balancing Discriminability and Generalizability

Xu Yang*, Xuan Chen*, Moqi Li, Kun Wei, Cheng Deng†
School of Electronic Engineering, Xidian University, Xian 710071, China
{xuyang.xd, moqili14, weikunsk, chdeng.xd}@gmail.com,
x.chen@stu.xidian.edu.cn

Abstract

Continual test-time domain adaptation (CTTA) aims to adapt the source pre-trained model to a continually changing target domain without additional data acquisition or labeling costs. This issue necessitates an initial performance enhancement within the present domain without labels while concurrently averting an excessive bias toward the current domain. Such bias exacerbates catastrophic forgetting and diminishes the generalization ability to future domains. To tackle the problem, this paper designs a versatile framework to capture high-quality supervision signals from three aspects: 1) The adaptive thresholds are employed to determine the reliability of pseudo-labels; 2) The knowledge from the source pre-trained model is utilized to adjust the unreliable one, and 3) By evaluating past supervision signals, we calculate a diversity score to ensure subsequent generalization. In this way, we form a complete supervisory signal generation framework, which can capture the current domain discriminative and reserve generalization in future domains. Finally, to avoid catastrophic forgetting, we design a weighted soft parameter alignment method to explore the knowledge from the source model. Extensive experimental results demonstrate that our method performs well on several benchmark datasets.

1. Introduction

Deep neural networks have achieved remarkable success in visual tasks when training and testing data obey the same distribution. Such networks, however, suffer from the generalization problem due to the ubiquitous domain shift [30]. For example, a classification network pre-trained in the normal, natural images may not recognize the corrupted images. Thus, domain adaptation is essential to transfer knowledge from the source domain to the target one by reducing the shift. However, the target domain labels are usually unavailable, and the problem is primarily explored at *Unsupervised Domain Adaptation* (UDA) [15, 31]. More realistically, the source data is often inaccessible during test time due to privacy or business problems, making the adaptation problem more challenging. Initial approaches attempt to employ the source model, and unlabeled target data for testing, called *Source-free / Test-Time domain Adaptation* (TTA) [3, 19, 33].

Existing TTA methods usually solve the domain shift problem by updating the adapted model parameters using the generated pseudo-labels or entropy regularization, which are effective when the target distribution is fixed but perform unstably when the distribution of the target domain is continually changing [22, 29, 36]. Such a problem has brought a novel and fully underexplored area, called Continual Test-Time Domain Adaptation (CTTA), where a source pre-trained model must adapt to a stream of continually changing target domains without using source data. Research has demonstrated that CTTA mainly faces the following problems. First, we should explore supervisory signals without labels to improve performance in the current target domain, where widespread noisy pseudo labels are a crucial factor affecting performance. Then, the adaptation process means moving the initial source parameterization to a parameterization that better models the current target distribution, which carries the risk that predictions on the source distribution become inaccurate, causing catastrophic forgetting. Finally, an excessive affinity for the existing domain will cause generalization to be lost for future domains when the current target distribution is narrow, especially under noisy pseudo labels.

Recently, some methods have been proposed to tackle such an intractable problem. CoTTA [29] adopts a weight-average teacher network to improve the quality of generated pseudo-labels and employs some source model parameters to cover the adapted model for alleviating catastrophic forgetting. Robust Mean Teacher [6] employs a multi-viewed contrastive loss to pull test features towards

[*] Equal contribution.
[†] Corresponding author.

the initial source space and learn invariances concerning the input space. However, research [20] has shown that network parameters tuned with the current domain may cause the loss of generalization and impair the performance of further domains. To this end, Gan [7] adopts the visual domain prompts to dynamically update a small portion of the input image pixels and mitigate the error accumulation problem. ViDA [18] injects visual domain adapters into the pre-trained model, which leverages high-rank and low-rank features to adapt the current domain distribution and maintain the continual domain-shared knowledge. As a simpler strategy, some methods [10, 20] only update the network's normalization parameters and freeze all others, which can retain a large amount of source pre-trained model knowledge. However, few works can fully enhance the performance of the current domain while ensuring the generalization of future domains, which requires exploring a more versatile optimization strategy.

The methods mentioned above lead to our research goal, improving the network discrimination in the current domain while ensuring the generalization of future domains. We build a versatile framework that generates high-quality supervision signals from two levels: reliability and diversity. Specifically, we calculate an independent threshold for each class through global and local strategies to divide pseudo-labels into reliable and unreliable parts. Then, we adopt the source predictions for the unreliable pseudo-labels to select potentially similar samples and calibrate the pseudo-labels for capturing diverse supervision signals. Based on the calibrated pseudo-labels, we begin by tracking the recent tendency of a model's prediction with an exponential moving average. Finally, we calculate a diversity score to ensure the generalization for future domains. In this way, we form a complete supervisory signal generation framework, which can optimize the current domain efficiently and reserve generalization in future domains.

Except for the supervising signal generation, we propose continually capturing source knowledge and calibrating the adapted model to maintain generalization and prevent catastrophic forgetting. Unlike CoTTA [29] randomly using some source model parameters to cover the adapted model, we hope that the objective function can be employed to guide the parameter transfer of the source pre-trained model and the adapted one. To this end, we introduce a soft-weighted parameter alignment that forces the adapted network to be similar to the source one. More importantly, noise signals inevitably appear in the generated pseudo-labels due to the absence of supervisory signals. The observation that the latter layers in a network are much more sensitive to label noise, while their former counterparts are quite robust [1, 35], inspires novel weighted guidance. The weights control the similarity of the adapted model to the source one with the depth of layers, allowing noise-robust former layers to be adjusted more and noise-sensitive latter ones to be adjusted less.

Research Question. Existing continual test-time adaptation implementations may face a game between discrimination in the current domain and generalization in future domains. Meanwhile, the absence of label signals makes the model performance worse when facing domain shifts. Thus, one research question is how to construct a novel pipeline that ensures generalization and improves discrimination, and the other is how to capture knowledge from the source pre-trained model.

Contributions. The highlights of the paper are threefold: 1) By the analysis and summary of the previous work, we design a versatile framework that generates high-quality supervision signals from two levels: Reliability and Diversity. The adaptive thresholds are employed to determine the reliability of pseudo labels, and the diversity score is employed to ensure the generalization for future domains. 2) We explore reliable supervision signals with a source pre-trained model to guide the test time tuning. The prior distribution for the source model is utilized to calibrate unreliable pseudo-labels, and the learnable parameters are aligned with the source parameters in a soft-weighted manner to alleviate catastrophic forgetting; 3) 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

2.1. Domain Adaptation

Domain adaptation [2, 13, 34] refers to the goal of learning a concept from labeled data in the source domain that performs well on different but related target domains. The critical domain adaptation problem lies in the misalignment between the feature and label spaces of the source and target domains [5, 28, 32]. To solve this problem, some domain adaptation methods guide the deep model to learn domain invariant representations [25] and classifiers [31]. Specifically, some works [8, 9, 26] utilize adversarial training to align feature distribution with a domain discriminator, and some works constrain the cross-domain feature space by entropy constraint [11, 23], or maximum prediction rank [4]. All the above methods need to access source and target data during the adaptation process, making learning transductive.

2.2. Test-Time Domain Adaptation

Recently, some works on test-time domain adaptation focus on a more challenging setting where only the source model and unlabeled target samples are available. Some test-time domain adaptation methods [15] utilize generative models

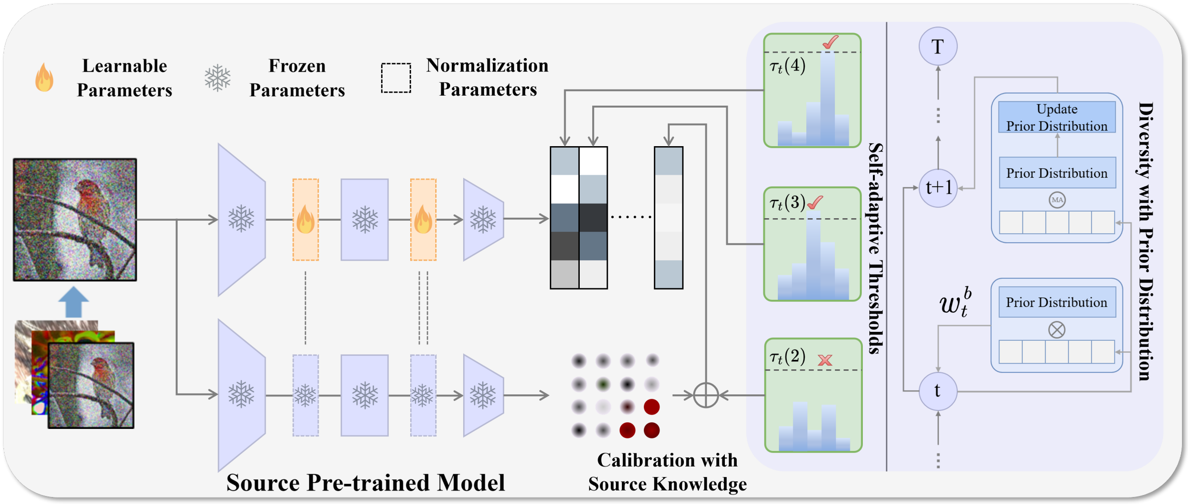

Figure 1. This is the flow of our method. We propose a novel pipeline to optimize the network's normalization parameters to ensure long-term generalization and improve instantaneous discrimination, and confidence thresholds are utilized in a self-adaptive manner to select reliable labels. Then, we explore various prior knowledge from the source pre-trained model to calibrate and enrich supervision signals. Moreover, we track the recent tendency of a model's prediction with an exponential moving average for a diversity score to ensure subsequent generalization. Finally, the learnable parameters are aligned with the source parameters in a soft-weighted manner to alleviate catastrophic forgetting.

to achieve the feature alignment between the source and target domain without additional source data. In addition, some methods achieve test-time domain adaptation by fine-tuning the source model with the help of target data and do not require explicit domain alignment. Test entropy minimization (TENT) [27] introduces entropy minimization as a test-time optimization objective, which estimates normalization statistics and optimizes channel-wise affine transformations to update online on each batch. Source HypOthesis Transfer (SHOT) [17] aims to learn the optimal target-specific feature learning module to fit the source hypothesis. EATA [21] utilizes certainty and diversity weighting for test-time adaptation and achieves competitive performance. However, the weighting scheme needs to be manually specified for each dataset.

Most test-time adaptation methods only consider the offline scenario, where the full set of test data is provided during the training process. Further, CoTTA [29] extends test-time adaptation from offline scenario to online continual scenario. It considers a more challenging but more realistic problem named *Continual Test-Time Domain Adaptation*, where a source pre-trained model needs to adapt to a stream of continually changing target test data without using any source data. RMT [6] uses symmetric cross-entropy and contrastive learning to pull the test feature space closer to the source domain. To prevent the loss of generalization and impair the performance of future domains, Gan [7] adopts the visual domain prompts to dynamically update a small portion of the input image pixels and mitigate the error accumulation problem, and ViDA [18] injects visual domain adapters into the pre-trained model to adapt the current domain distribution and maintain the continual domain-shared knowledge. In a simpler manner, NOTE [10] proposes an instance-aware batch normalization to correct normalization for out-of-distribution samples, and RoTTA [37] presents a robust batch normalization scheme to estimate the normalization statistics. Recently, ROID [20] proposes to continually weight-average the source and adapted model, and an adaptive additive prior correction scheme for diversity.

Our Study. Existing methods lack the integration of reliability optimization and diversity generalization within a unified framework, leading to performance deficiencies. Our approach initially assesses the reliability of the supervisory signals, and subsequently enhances the unreliable ones with a source pre-trained model. We assign diverse weights to different signals, ensuring comprehensive learning in the current domain while efficiently generalizing future domains.

3. Proposed Method

Following [29], we consider a continual test-time domain adaptation setting, where a pre-trained model needs to adapt to a continually changing target domain online without source data. Consider a pre-trained model $F_\theta(x)$ with parameter θ trained on the source data. Unlabeled target domain data X_t is provided sequentially, and the data distribution continually changes. At testing stage t, when the unlabeled target data $X_t = [x_t^1, ..., x_t^B]$ 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 ($\theta_t \to \theta_{t+1}$). It is worth noting that the total evaluation process is online, and the model only has access to the data X_t of the current stage t. We design a dual-stream network, which optimizes different parameters independently in each stream, to capture knowledge from continual domains. Meanwhile, we explore prior knowledge from the source pre-trained model. The framework is shown in Figure 1.

3.1. High-quality Supervision Generator

We first design a pipeline for continual test-time domain adaptation to capture discrimination and generalization. For the convenience of expression, θ_t in the following mainly refers to the parameters of the proposed network at time t, where only the batch normalization layers are tuned. The learning process can be denoted as follows.

$$\begin{aligned} p_t^b &= \texttt{Softmax}(F_{\theta_t}(x_t^b)), \\ \mathcal{L}_{ce}(X_t) &= -\frac{1}{B}\sum_{b=1}^{B} y_t^b \log p_t^b, \end{aligned} \tag{1}$$

where p_t^b represents classification result of the sample b at time t, and y_t^b is the supervision signal of the ith sample. $\mathcal{L}_{ce}$ represents the cross-entropy loss. Usually, we adopt the current network output as the supervision signals, that is, $y_t^b = p_t^b$. However, such signals have many limitations. First, network-based output results are not entirely correct, and the corresponding supervision signals contain a lot of noise. Long-term noise accumulation may cause the network to fall into a vicious cycle, causing catastrophic forgetting and collapse problems. Thus, we need to process y_t^b to improve the reliability of supervision, which will be introduced in follows.

Selection with Self-adaptive Thresholds. First, we adopt a confidence threshold to filter reliable labels. We present self-adaptive thresholding that automatically defines and adaptively adjusts the confidence threshold for each class by leveraging the current predictions during adaptation. The global threshold should represent the confidence of the model, reflecting the overall learning status. We set the global threshold τ_t as the average confidence from the model, and estimate the global confidence at each stage t. τ_t is defined and adjusted as:

$$\tau_t = \frac{1}{B}\sum_{b=1}^{B} \max(y_t^b). \tag{2}$$

Except for the global threshold, the local threshold is utilized to modulate the global threshold in a class-specific fashion to account for the intra-class diversity and the possible class adjacency. We compute the expectation of the model's predictions on each class to estimate the class-specific learning status:

$$\xi_t(c) = \frac{1}{B}\sum_{b=1}^{B} y_t^{b,c}, \tag{3}$$

where $c \in C$ is the number of classes. After integrating the global and local thresholds, we can obtain the final self-adaptive threshold of each class c.

$$\tau_t(c) = \frac{\xi_t(c)}{\max\{\xi_t(c) : c \in C\}}\tau_t. \tag{4}$$

Based on such thresholds, the samples at the current batch can be divided into two parts, the reliable part $N_{rel}(t) = \{b | b \in B, \max(y_t^b) \geq \tau_t(\arg\max y_t^b)\}$ and the unreliable one $N_{unrel}(t) = \{b | b \in B, \max(y_t^b) < \tau_t(\arg\max y_t^b)\}$.

Calibration with Source Knowledge. Second, we attempt to distill knowledge from the source pre-trained model to calibrate the unreliable signals. The source pre-trained model is fully trained with labels, so even if the domain shift causes the classification results to be biased, it is still a suitable feature extractor. In other words, the source pre-trained model can still judge samples' similarity. Based on this, We hope to extract prior distribution from the source pre-trained model to calibrate the unreliable results. Specifically, we use the features of the source pre-trained model to retrieve similar samples, and calculate the pseudo-labels of these samples . To this end, we first exploit the source pre-trained model to extract the sample features and establish a similarity matrix.

$$f_t^b = \texttt{Softmax}(F_\theta(x_t^b)), s_t^{b,d} = \text{sim}(f_t^b, f_t^d), \tag{5}$$

where f_t^b and f_t^d are the representations of the sample b and d at time t. $\text{sim}(\cdot)$ represents the consine similarity. Here, the set of the K nearest neighbors $N_{neg}^b(t), b \in N_{unrel}(t)$ are selected by $s_t^{b,d}$ for the sample, and the calibrated pseudo-labels are calculated.

$$y_t^b = \frac{1}{\sum_{d \in N_{neg}^b(t)} s_t^{b,d} + 1} \sum_{d \in N_{neg}^b(t)} s_t^{b,d} * y_t^d + y_t^b. \tag{6}$$

Diversity with Prior Distribution. Following the aforementioned steps, we acquire the refined supervisory signals. Subsequently, our objective is to learn a set of weights to assess the importance of various supervisory signals. The output results of the network may become biased or collapse to a trivial solution after a narrow distribution during test time. Therefore, we introduce a diversity criterion [20] to ensure that diverse samples are favored compared to samples 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.

$$\bar{y}_{t+1} = \alpha \bar{y}_t + \frac{1-\alpha}{B} \sum_{i=1}^{B} y_t^b, \tag{7}$$

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

$$u_t^b = 1 - \frac{\bar{y}_t^\top y_t^b}{\|y_t^b\| \|\bar{y}_t^b\|}. \tag{8}$$

u_t^b 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.

$$v_t^b = y_t^b \log y_t^b. \tag{9}$$

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.

$$w_t^b = \exp\left(\frac{u_t^b \cdot v_t^b}{\tau}\right). \tag{10}$$

After the above selection, calibration and weighting, the objective in Eq. 1 can be reconstructed as follows.

$$\mathcal{L}_{ce}(X_t) = -\frac{1}{B} \sum_{b=1}^{B} w_t^b y_t^b \log p_t^b, \tag{11}$$

3.2. Soft-weighted Parameter Alignment

The source pre-trained model, trained on labeled data, is more reliable and exhibits better generalization. Consequently, exploring knowledge from the source pre-trained model is crucial for our task. CoTTA employs a direct utilization of source parameters to encompass the adapted ones randomly, a practice that may compromise the performance of the adapted model. To address this, we construct a soft parameter alignment function and incorporate it into the loss function to optimize the network. This ensures that the network parameters are highly correlated with those of the source pre-trained model during loss optimization, rather than being overwritten afterward. The weights are derived from the fitting preference of the parameters to noisy labels (given the absence of real labels during adaptation). As a result, we anticipate that parameters susceptible to noise will exhibit greater alignment with the parameters of the source model, thus preventing excessive parameter deviation.

We hope that the objective function can be employed to directly guide the parameter transfer of the source model and the adapted one, and the Weighted Soft Parameter Alignment can be defined as follows.

$$\mathcal{L}_{pa}(\theta_t) = \sum_{l} \mathbf{1}[l \in \mathrm{BN}] \cdot \beta^l \left\| \theta_t^l - \theta^l \right\|_2^2, \tag{12}$$

where l is the layer of the network and β^l repressents the similarity strength of l-th layer. We set $\beta^l = \frac{1-e^{-10l}}{1+e^{-10l}}$, which is increased with the deeper layers.

3.3. Overall

The overall objective of our method is as follows.

$$\mathcal{L}(X_t) = \mathcal{L}_{ce}(X_t) + \lambda_1 \mathcal{L}_{pa}(\theta_t), \tag{13}$$

where λ_1 is the hyperparameter. In general, we do not directly use the results of pre-trained and adapted models as supervision signals, but apply them as prior knowledge to calibrate pseudo-labels, and design a soft-weighted parameter alignment 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 the proposed label selection and correction strategies can improve the discrimination, 2) whether our soft-weighted alignment learns meaningful results, and 3) the parameters analysis of the proposed method.

4.1. 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 [12]. Each target domain dataset contains 15 corruption types with five severity levels. Following [29], for each corruption, we use 10000 images for both CIFAR10C and CIFAR100C datasets and 5000 images for ImageNet-C.

4.2. Implementation Details

Following [29], the corrupted images are provided to the network online, which means these images can be utilized to update the model only once in the adaptation process. In addition, unlike traditional test-time adaptation methods, which adapt to each corruption type data individually, we adjust the source model to each corruption type sequentially. We evaluate the adaptation performance immediately after encountering each corruption type 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).

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 [2023.6.1]		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		20.7	17.1	20.2	12.1	24.3	**11.6**	10.9	**13.8**	**12.9**	**10.5**	8.1	**9.3**	**17.9**	**13.4**	15.3	**14.5**	**+29.0**
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 [2023.6.7]		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 [2023.6.1]		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
Ours		**16.3**	**11.1**	**9.6**	**8.4**	**14.6**	**8.6**	**5.5**	**6.3**	**5.7**	**7.1**	**3.3**	**5.4**	**10.9**	**7.7**	**12.8**	**8.9**	**+19.3**

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. Blue is the suboptimal solution.

In our experiments, we adhere to the implementation details outlined in previous works [29] to ensure consistency and comparability. For the classification CTTA, we employ ViT-base and ResNet [38] 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. The factor $\lambda_1 = 0.1$ and $K = 3$ in our experiments. We set $\beta^l = \frac{1-e^{-10l}}{1+e^{-10l}}$, where l represents the number of layers.

4.3. 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 [16, 24]; 3) **Pseudo-Label** [14] picks up the class which has the maximum predicted probability as the pseudo-labels to update the model; 4) **TENT** [27] reduces generalization error by reducing the entropy of model predictions on test data, **TENT-continual** is a continual learning version of TENT; 5) **CoTTA** [29] 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** [10] adopts an Instance-Aware Batch Normalization to correct normalization for out-of-distribution samples; 7) **RoTTA** [37] presents a robust batch normalization scheme to estimate the normalization statistics; 8) **RMT** [6] uses symmetric cross-entropy and contrastive learning to pull the test feature space closer to the source domain; 9) **ROID** [20] proposes to continually weight-average the source and adapted model, and an adaptive additive prior correction scheme; 10) **ViDA** [18] injects visual domain adapters into the pre-trained model to adapt the current domain distribution and maintain the continual domain-shared knowledge.

4.4. Performance Evaluation

CIFAR10-to-CIFAR10C. Table 8 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 the error accumulation to improve performance further. As the latest proposed methods, *NOTE* attempts to improve the performance of the model in different domains from 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* has 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 with all the previous methods, our method achieves the best results in the average error value and most of the corruption-type data under different backbones. It is worth mentioning that currently, only fine-tuning the BN layer of the Transformer network is an ideal learning strategy. Both ROID and our method achieved good performance, but our method is obviously 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

Time		*t* ⟶																
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 [2023.6.1]		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		33.5	**31.8**	**31.2**	25.9	**30.9**	**25.2**	25.9	**27.9**	**27.4**	**30.6**	25.2	**23.5**	**26.6**	**26.2**	**27.2**	**27.9**	**+18.5**
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 [2023.6.7]		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 [2023.6.1]		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
Ours		**38.2**	**31.8**	**18.2**	**20.8**	**34.3**	**20.3**	17.5	**14.9**	**16.2**	**22.9**	11.5	**27.5**	**28.2**	**32.5**	**25.3**	**24.0**	**+11.4**

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. Blue is the suboptimal solution.

Time		*t* ⟶																
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 [2023.6.7]		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 [2023.6.1]		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		**70.8**	**60.3**	**60.5**	**65.8**	**55.2**	**55.5**	**46.7**	**49.0**	**50.1**	**40.3**	34.1	**56.1**	**42.8**	**40.2**	**43.9**	**51.4**	**+30.6**
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 [2023.6.7]		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 [2023.6.1]		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
Ours		**47.5**	**42.1**	**41.6**	55.5	55.4	**44.5**	47.9	**38.8**	**37.8**	39.6	**23.6**	57.0	44.4	**33.5**	**32.3**	**42.7**	**+13.1**

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. Blue is the suboptimal solution.

Avg. Error (%)	Source	BN Adapt	TENT-continual	CoTTA	ROID	Ours
CIFAR10C	24.8	13.7	29.2	10.4	6.1	**5.7**

Table 4. Gradually changing setup results on CIFAR10-to-CIFAR10C. The severity level changes gradually between the lowest and the highest. Results are the mean over ten diverse corruption type sequences. **Bold** text indicates the best performance. Blue is the suboptimal solution. All results are evaluated on the ResNet.

error accumulation, but there is little room for improvement. *CoTTA* considers the error accumulation problem and reduces the error to 32.5%. Further, the performance of our method is better than *RMT* and *ROID* on several corruption types of data, and the average error value is reduced to 27.8%. Obviously, ViT-base is still our first choice, its overall performance is better than ResNet, and our method is still ahead of existing learning strategies.

ImageNet-to-ImageNet-C. We also make experiments on the ImageNet dataset. Following [29], 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 and reduces the av-

Time	$t \longrightarrow$															
Method	*Gaussian*	*Shot*	*Impulse*	*Defocus*	*Glass*	*Motion*	*Zoom*	*Snow*	*Frost*	*Fog*	*Brightness*	*Contrast*	*Elastic_trans*	*Pixelate*	*Jpeg*	Mean
Source	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
SST	23.3	20.4	25.0	13.8	30.5	13.9	12.8	15.5	14.6	15.4	8.0	12.4	22.4	18.2	19.4	17.7
SST+CSK	27.5	24.8	28.9	12.0	32.8	13.6	11.2	16.9	12.8	10.2	7.9	12.2	20.5	13.8	17.5	17.4
SST+DPD	25.8	22.2	27.0	11.3	29.5	13.1	10.6	15.8	12.0	10.1	7.8	12.0	19.6	13.5	15.5	16.1
SST+CSK+DPD	21.3	17.8	22.7	13.2	26.8	13.2	11.5	14.7	13.2	10.9	8.0	10.2	18.8	14.5	16.8	15.8
SST+CSK+SPA	25.5	19.1	22.2	12.1	28.3	11.9	10.9	14.7	11.9	10.6	8.5	11.6	18.5	13.2	15.9	15.6
SST+DPD+SPA	22.1	18.1	21.2	12.6	25.1	11.9	10.5	14.3	12.2	9.8	8.1	10.6	18.3	13.9	15.6	14.8
SST+CSK+DPD+SPA	20.7	17.1	20.2	12.1	24.3	11.6	10.9	13.8	12.9	10.5	8.1	9.3	17.9	13.4	15.3	14.5

Table 5. Ablation experiments of the framework for the CIFAR10-to-CIFAR10C task. ‘SST’ represents the label selection with self-adaptive thresholds, and the unreliable part is discarded directly. ‘CSK’ is the Calibration with Source Knowledge, and ‘DPD’ is the Diversity with Prior Distribution module. SPA is the Soft-weighted Parameters Alignment. All results are evaluated on the ResNet.

erage test error to 51.4% and 41.3% with ResNet and ViT networks respectively.

The improvement of the model we proposed on CIFAR100C and ImageNet is not as significant as on CIFAR10C. The main reason is that as the complexity of the category increases, the disadvantages of the limited learning ability to fine-tune normalization layers gradually become apparent. However, we still achieved extremely competitive results through label calibration. More importantly, the proposed model does not involve any parameter or data expansion. This efficient learning strategy is more in line with the practical application requirements of CTTA.

Gradually Changing Setup. Following [29], we also consider a gradually changing setup. For the standard setup, corruption types change abruptly in the highest severity. For the gradually changing setup, the corruption types change is gradual. The results shown in Table 4 represent that the proposed method achieves better performance.

4.5. Ablation Studies

We first conduct ablation experiments with the same supervision signals to prove the effectiveness of the proposed framework in Table 7. For the convenience of expression, ‘SST’ represents the label selection with self-adaptive thresholds, and the unreliable part is discarded directly. Then, such a module is combined with label calibration (Calibration with Source Knowledge, CSK) and diversity reweighting (Diversity with Prior Distribution, DPD), respectively. Ultimately, these three will form a versatile supervisory signal generator. SPA is the Soft-weighted Parameters Alignment module. The results demonstrate that the pseudo-label after selection and calibration strategies can effectively suppress noisy labels and improve performance. Moreover, diversity with prior distribution is vital for the model. Such modules work together to build high-quality supervision signals. Subsequently, we focus on validating the proposed supervision signals module. Finally, although we only fine-tuned the normalization parameters, the results show that it is still necessary to use parameter alignment. This indicates a large amount of generalization knowledge in the source pre-trained model waiting for further exploration.

5. Conclusion

This paper first proposes a versatile framework that generates high-quality supervision signals from two levels: reliability and diversity. We calculate an independent threshold for each class through global and local strategies to divide pseudo-labels into reliable and unreliable parts. Then, we propose continually capturing source knowledge using different strategies to guide the adapted model. To this end, we adopt the source predictions for the unreliable pseudo-labels to select potentially similar samples, and calibrate the pseudo-labels for capturing diverse supervision signals. Based on the calibrated pseudo-labels, we begin by tracking the recent tendency of a model’s prediction with an exponential moving average, and calculate a diversity score to ensure the generalization for future domains. Moreover, we introduce a soft-weighted parameters alignment that forces the adapted network to be similar to the source. The proposed framework can optimize the current domain efficiently and reserve generalization in future domains efficiently. Finally, we evaluate the proposed method on several benchmarks and prove its superiority.

6. Acknowledgments

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, 62171343, and 62201436), Key Research and Development Program of Shaanxi (2024GX-YBXM-127) and Fundamental Research Funds for the Central Universities (ZDRC2102).

References

[1] Yingbin Bai, Erkun Yang, Bo Han, Yanhua Yang, Jiatong Li, Yinian Mao, Gang Niu, and Tongliang Liu. Understanding and improving early stopping for learning with noisy labels. *NeurIPS*, 34:24392–24403, 2021. 2, 1

[2] Zhangjie Cao, Kaichao You, Ziyang Zhang, Jianmin Wang, and Mingsheng Long. From big to small: adaptive learning to partial-set domains. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(2):1766–1780, 2022. 2

[3] Dian Chen, Dequan Wang, Trevor Darrell, and Sayna Ebrahimi. Contrastive test-time adaptation. *arXiv preprint arXiv:2204.10377*, 2022. 1

[4] Shuhao Cui, Shuhui Wang, Junbao Zhuo, Liang Li, Qingming Huang, and Qi Tian. Towards discriminability and diversity: Batch nuclear-norm maximization under label insufficient situations. In *CVPR*, pages 3941–3950, 2020. 2

[5] Yifei Ding, Minping Jia, Jichao Zhuang, Yudong Cao, Xiaoli Zhao, and Chi-Guhn Lee. Deep imbalanced domain adaptation for transfer learning fault diagnosis of bearings under multiple working conditions. *Reliab. Eng. Syst. Saf.*, 230: 108890, 2023. 2

[6] Mario Döbler, Robert A Marsden, and Bin Yang. Robust mean teacher for continual and gradual test-time adaptation. In *CVPR*, pages 7704–7714, 2023. 1, 3, 6

[7] Yulu Gan, Xianzheng Ma, Yihang Lou, Yan Bai, Renrui Zhang, Nian Shi, and Lin Luo. Decorate the newcomers: Visual domain prompt for continual test time adaptation. *arXiv preprint arXiv:2212.04145*, 2022. 2, 3

[8] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *ICML*, pages 1180–1189. PMLR, 2015. 2

[9] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.*, 17(1):2096–2030, 2016. 2

[10] Taesik Gong, Jongheon Jeong, Taewon Kim, Yewon Kim, Jinwoo Shin, and Sung-Ju Lee. Note: Robust continual test-time adaptation against temporal correlation. In *NeurIPS*, 2022. 2, 3, 6

[11] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. *NeurIPS*, 17, 2004. 2

[12] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019. 5

[13] Junguang Jiang, Baixu Chen, Jianmin Wang, and Mingsheng Long. Decoupled adaptation for cross-domain object detection. *arXiv preprint arXiv:2110.02578*, 2021. 2

[14] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *ICMLW*, page 896, 2013. 6

[15] Rui Li, Qianfen Jiao, Wenming Cao, Hau-San Wong, and Si Wu. Model adaptation: Unsupervised domain adaptation without source data. In *CVPR*, pages 9641–9650, 2020. 1, 2

[16] Yanghao Li, Naiyan Wang, Jianping Shi, Jiaying Liu, and Xiaodi Hou. Revisiting batch normalization for practical domain adaptation. *arXiv preprint arXiv:1603.04779*, 2016. 6

[17] Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *ICML*, pages 6028–6039. PMLR, 2020. 3

[18] Jiaming Liu, Senqiao Yang, Peidong Jia, Ming Lu, Yandong Guo, Wei Xue, and Shanghang Zhang. Vida: Homeostatic visual domain adapter for continual test time adaptation. *arXiv preprint arXiv:2306.04344*, 2023. 2, 3, 6

[19] Yuang Liu, Wei Zhang, and Jun Wang. Source-free domain adaptation for semantic segmentation. In *CVPR*, pages 1215–1224, 2021. 1

[20] Robert A Marsden, Mario Döbler, and Bin Yang. Universal test-time adaptation through weight ensembling, diversity weighting, and prior correction. *arXiv preprint arXiv:2306.00650*, 2023. 2, 3, 4, 6

[21] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *ICML*, pages 16888–16905. PMLR, 2022. 3

[22] Viraj Prabhu, Shivam Khare, Deeksha Kartik, and Judy Hoffman. Sentry: Selective entropy optimization via committee consistency for unsupervised domain adaptation. In *ICCV*, pages 8558–8567, 2021. 1

[23] Kuniaki Saito, Donghyun Kim, Stan Sclaroff, Trevor Darrell, and Kate Saenko. Semi-supervised domain adaptation via minimax entropy. In *ICCV*, pages 8050–8058, 2019. 2

[24] Steffen Schneider, Evgenia Rusak, Luisa Eck, Oliver Bringmann, Wieland Brendel, and Matthias Bethge. Improving robustness against common corruptions by covariate shift adaptation. *NeurIPS*, 33:11539–11551, 2020. 6

[25] Tao Sun, Cheng Lu, and Haibin Ling. Prior knowledge guided unsupervised domain adaptation. In *ECCV*, pages 639–655. Springer, 2022. 2

[26] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *CVPR*, pages 7167–7176, 2017. 2

[27] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726*, 2020. 3, 6

[28] Mengzhu Wang, Shanshan Wang, Wei Wang, Li Shen, Xiang Zhang, Long Lan, and Zhigang Luo. Reducing bi-level feature redundancy for unsupervised domain adaptation. *Pattern Recogn.*, page 109319, 2023. 2

[29] Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. *arXiv preprint arXiv:2203.13591*, 2022. 1, 2, 3, 5, 6, 7, 8

[30] Xu Wang, Dezhong Peng, Peng Hu, Yunhong Gong, and Yong Chen. Cross-domain alignment for zero-shot sketch-based image retrieval. *IEEE Trans. Circ. Syst. Video Tech.*, 2023. 1

[31] Xu Wang, Dezhong Peng, Ming Yan, and Peng Hu. Correspondence-free domain alignment for unsupervised cross-domain image retrieval. *arXiv preprint arXiv:2302.06081*, 2023. 1, 2

[32] Binhui Xie, Longhui Yuan, Shuang Li, Chi Harold Liu, Xinjing Cheng, and Guoren Wang. Active learning for do-

main adaptation: An energy-based approach. In *AAAI*, pages 8708–8716, 2022. 2

[33] Shiqi Yang, Yaxing Wang, Joost van de Weijer, Luis Herranz, and Shangling Jui. Generalized source-free domain adaptation. In *ICCV*, pages 8978–8987, 2021. 1

[34] Xu Yang, Cheng Deng, Tongliang Liu, and Dacheng Tao. Heterogeneous graph attention network for unsupervised multiple-target domain adaptation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(4):1992–2003, 2020. 2

[35] Xu Yang, Cheng Deng, Kun Wei, and Dacheng Tao. Robust commonsense reasoning against noisy labels using adaptive correction. *IEEE Trans. Cybern.*, 2023. 2

[36] Xu Yang, Yanan Gu, Kun Wei, and Cheng Deng. Exploring safety supervision for continual test-time domain adaptation. In *IJCAI*, pages 1649–1657, 2023. 1

[37] Longhui Yuan, Binhui Xie, and Shuang Li. Robust test-time adaptation in dynamic scenarios. In *CVPR*, pages 15922–15932, 2023. 3, 6

[38] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016. 6