Beyond Entropy: Style Transfer Guided Single Image Continual Test-Time Adaptation

Younggeol Cho[1,2*], Youngrae Kim[1,3*], and Dongman Lee[1]

[1]Korea Advanced Institute of Science and Technology (KAIST)
[2]Seoul National University
[3]University of Southern California
{rangewing, dlee}@kaist.ac.kr, youngrae@use.edu

Abstract. Continual test-time adaptation methods are designed to facilitate the continual adaptation of models to dynamically changing real-world environments. Concurrently, real-world vision applications, such as semantic segmentation, necessitate the utilization of high-resolution images to achieve optimal performance, which limits the batch size during test time. However, the instability caused by batch normalization layers and entropy loss when using small batch sizes as well as a single image significantly destabilizes many existing methods in real-world continual TTA scenarios. To overcome these challenges, we present BESTTA, a novel single image continual test-time adaptation method guided by style transfer, which enables stable and efficient adaptation to the target environment by transferring the style of the input image to the source style. To implement the proposed method, we devise BeIN, a simple yet powerful normalization method, along with the style-guided losses. We demonstrate that BESTTA effectively adapts to the continually changing target environment, leveraging only a single image on both semantic segmentation and image classification tasks. Remarkably, despite training only two parameters in a BeIN layer consuming the least memory, BESTTA outperforms existing state-of-the-art methods in terms of performance.

1 Introduction

Deep learning models have significantly improved the performance of computer vision applications [6, 14]. However, in real-world scenarios, the models suffer from performance degradation due to domain shifts between training and target data. Unsupervised domain adaptation (UDA) [10, 24, 25, 30, 33, 37, 39] and test-time training (TTT) [23, 38] techniques have been proposed to address the performance gap. These methods assume that the models are adapted to the unseen domain with the huge amount of source data at test time. However it is impractical to utilize these methods in the real world due to the limited computational resources. Fully test-time adaptation (TTA) methods [2, 29, 41] have recently emerged to enable online adaptation of a trained model to the target

*These authors contributed equally.

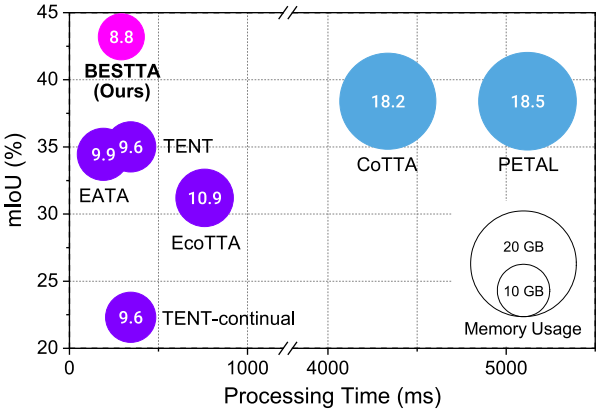

Fig. 1: Semantic segmentation performances on the Cityscapes-to-ACDC single image continual test-time adaptation task, evaluated on DeepLabV3Plus-ResNet50 [3]. Processing time includes inference and adaptation time per image. The circle radius and values indicate the peak GPU memory usage. The violet circles are batch normalization-based and the blue circles are pseudo-label-based methods. BESTTA significantly outperforms the state-of-the-art continual test-time adaptation methods [1, 28, 36, 41, 43] in terms of mIoU while consuming the least GPU memory.

environment without source data or labels. These TTA methods have limitations in that they can only adapt to a single target domain, resulting in overfitting to the target domain and forgetting the prebuilt knowledge. Consequently, recent studies [1, 28, 36, 43] have proposed continual test-time adaptation methods to adapt a model for continually changing target domains over a long period, addressing the issues of catastrophic forgetting and error accumulation found in the previous TTA methods.

Despite the advancements in TTA methods, they often overlook practicality by using infeasible batch sizes of 64 or 128 in real-world applications. Most real-world downstream vision tasks, such as semantic segmentation, require high-resolution images for optimal performance [42, 48]. This need for high resolution limits the batch size due to the limited computational resources available in edge environments where most TTA methods are deployed. However, TTA methods that consider limited batch sizes have not yet been sufficiently explored, especially when using a *single image* input, as shown in Table 1.

There are two main reasons why existing TTA methods cannot handle the single-image setting. Firstly, small mini-batch severely degrades the performances of most existing TTA methods based on batch normalization (BN) [13, 27, 28, 35, 36, 41, 44]. These methods utilize BN to align the distributions between training and target data. However, inaccurate mini-batch statistics, resulting from the small number of samples from the target domain, cause substantial performance degradation when using a small batch size. This issue is further exacerbated when dealing with a single image [22, 29]. Secondly, entropy loss, which is em-

Table 1: The settings of single image continual test-time adaptation and related adaptation areas. X^s and Y^s denote the source image and label sets, respectively, X^t denotes the target image set, $X^t_{\text{batch}} = \{x^t_i, \cdots, x^t_{i+k}\}$ denotes the batch of k target images, and x^t_i denotes the single target image at time i.

Settings	Data		Learning		Continually	Single
	Source	Target	Train stage	Test stage	Changing Domain	Image
Unsupervised Domain Adaptation	X^s, Y^s	X^t	✓	-	✗	✗
Test-Time Training	X^s, Y^s	X^t_{batch}	✓	✓	✗	✗
Fully Test-Time Adaptation	-	X^t_{batch}	-	✓	✗	✗
Continual Test-Time Adaptation	-	X^t_{batch}	-	✓	✓	✗
Single Image Continual Test-Time Adaptation	-	x^t_i	-	✓	✓	✓

ployed by the majority of existing TTA methods [13, 19, 31, 36, 41], becomes significantly unstable when only a single image is utilized, due to the large and noisy gradients from the unreliable prediction [29]. Although EATA [28] and SAR [29] mitigated this by filtering out the unreliable samples, these methods still rely solely on the entropy loss, preventing optimal performance in the single-image setting. Consistency loss using pseudo-labels [1, 43] would overcome this problem, but it is inefficient in terms of computational complexity and memory consumption because it requires tens of inferences for acquiring pseudo-labels.

In our paper, we present **BESTTA** (Beyond Entropy: Style Transfer guided single-image continual Test-Time Adaptation), a continual TTA method that achieves stable and efficient adaptation when only a single image is available for update. We formulate the TTA problem as a style transfer from the target style to the source style, which allows us to drive other losses not just the entropy loss. We introduce the style and content losses tailored to the TTA problem, which ensures effective and stable adaptation. Motivated by the normalization-based style transfer methods [7, 16], we implement the style transfer process with a single normalization layer, named BeIN. Given that effective style transfer requires the reliable statistics of the target domain, which is difficult to obtain in the single-image setting, we add two learnable parameters to BeIN that estimate the reliable statistics by learning the effective style transfer with our proposed losses.

Our contributions are summarized as follows:

- We propose BESTTA, a noble style transfer guided single-image continual test-time adaptation method that enables stable and efficient adaptation. We formulate the test-time adaptation problem as a style transfer and propose novel style and content losses for stable single-image continual test-time adaptation.
- We propose a simple but powerful normalization method, namely BeIN. BeIN provides stable style transfer by learning to estimate the target statistics from input and source statistics.
- We demonstrate that the proposed BESTTA effectively adapts to continually changing target domains with a single image on semantic segmentation and image classification tasks. Remarkably, despite training only two parameters,

BESTTA outperforms the existing state-of-the-art methods with the least memory consumption, as shown in Fig. 1.

2 Related Work

2.1 Test-Time Adaptation

In order to enable model adaptation in source-free, unlabeled, and online settings, various test-time domain adaptation methodologies [2, 19, 22, 29, 29, 31, 41, 46] have been introduced, designing unsupervised losses for this purpose. TENT [41] was the first to highlight the effectiveness of updating batch normalization based on entropy loss in a test-time adaptation task, inspiring subsequent studies to target batch normalization updates [13, 28, 36]. However, these methods require a large batch size for proper batch normalization statistics and show instability at small batch sizes [22,29]. While some methods have aimed to facilitate TTA with small batch sizes [19,22,29,31], they often require prior knowledge of domain changes or auxiliary pretraining , making them impractical.

To address this, continual TTA methods [1,13,28,36,43] have emerged, aiming to prevent catastrophic forgetting and error accumulation caused by continued exposure to the inaccurate learning signal of unsupervised loss. To mitigate these issues, techniques such as stochastic parameter restoration [1, 43] and weight regularization losses [28] have been proposed. However, these methods also lack of considerations for a small batch sizes during adaptation.

2.2 Style Transfer

Neural style transfer methods have been proposed to transfer the style of the image while preserving the content [9, 11, 12, 20, 21, 40, 47]. Among them, some studies proposed normalization techniques to effectively transfer the image [7,16]. Using a similar method, Fahes *et al.* [8] transferred the input styles in a dataset to the target style and trained subsequent neural networks for downstream tasks. Motivated by these methods, we propose to transfer the target feature to the source domain, updating the normalization layer in the cTTA setting.

3 Method

Fig. 2 illustrates the overview of the proposed method, BESTTA. Motivated by style transfer methods [7, 16] that use a normalization layer, we formulate the TTA problem as a style transfer problem in Sec 3.1. We propose a single normalization layer, BeIN, which stabilizes the target input statistics in single image cTTA in Sec 3.2, with the proposed losses in Sec 3.3.

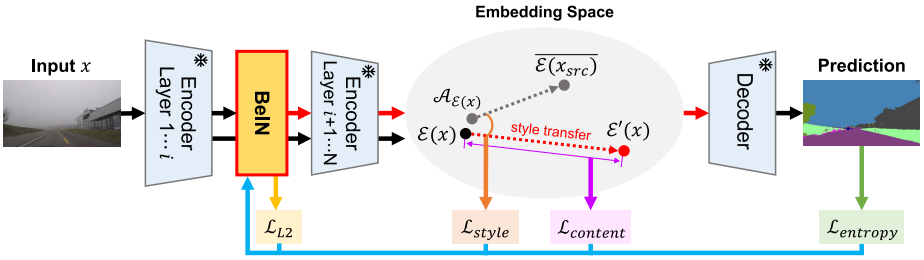

Fig. 2: Overview of BESTTA. We inject a single normalization layer, BeIN, to the pretrained model, which facilitates the stable single image continual test-time adaptation via style transfer. $\mathcal{E}(x)$ denotes the input embedding, $\mathcal{E}'(x)$ denotes the adapted input embedding, $\overline{\mathcal{E}(x_{src})}$ denotes the mean source embedding, and $\mathcal{A}_{\mathcal{E}(x)}$ denotes the exponential moving average of the target embeddings. To transfer the style of the input image to the source style, we adopt the directional style loss $\mathcal{L}_{style}$. To avoid the distortion of the content in the input image, we take the content loss $\mathcal{L}_{content}$. And to further improve the performance of the model, we use the entropy loss, i.e., $\mathcal{L}_{entropy}$. Finally, to avoid catastrophic forgetting and error accumulation, we use the L2 loss $\mathcal{L}_{L2}$. During adaptation, only two normalization parameters in BeIN are trained.

3.1 Problem Definition

Stable adaptation on a single image. Batch normalization (BN) [17] utilizes mini-batch statistics, mean and standard deviation of the source data to normalize data on the same distribution for each channel:

$$\mathrm{BN}(X) = \alpha \cdot \frac{X - \mu_s}{\sigma_s} + \beta \tag{1}$$

where $X \in \mathbb{R}^{B \times C \times H \times W}$ denotes the mini-batch of input features with C channels, $\alpha, \beta \in \mathbb{R}^C$ denote learnable affine parameters optimized during pretraining, and the mean $\mu_s \in \mathbb{R}^C$ and the standard deviation $\sigma_s \in \mathbb{R}^C$ are obtained by exponentially averaging the features during pretraining.

TENT [41] introduces the concept of updating affine parameters of BN layers. TENT replaces the source statistics (μ_s, σ_s) to a target statistics (μ_X, σ_X) of a target input feature X to address the distribution shift, and trains the learnable affine parameters α and β during test-time:

$$\mathrm{BN}_{\mathrm{TENT}}(X) = \alpha \cdot \frac{X - \mu_X}{\sigma_X} + \beta \tag{2}$$

The parameters are updated with the entropy loss $H(\hat{y}) = -\Sigma_c p(\hat{y}) \log p(\hat{y})$, where $\hat{y}$ is the prediction of the target input. This approach and related studies [28, 36] have demonstrated significant promise in TTA with the large batch size.

However, the normalization-based methods suffer significant performance drop when using small mini-batches, because they depend on the assumption

that the true mean μ_t and variance σ_t^2 of the target distribution can be estimated by the sample statistics. By the central limit theorem, the sample mean μ_X follows the Gaussian distribution $N(\mu_t, \frac{\sigma_t^2}{n})$ for sufficient large samples of size n. The sample variance follows the Chi-square distribution such that $(n-1)\frac{\sigma_X^2}{\sigma_t^2} \sim \chi_{n-1}^2$, and the variance of this distribution $Var(\sigma_X^2) = \frac{2\sigma_t^4}{n-1}$, where the data are normally distributed. Since the variances of the distributions of both sample statistics significantly increase when n is small, estimating the true mean and variance based on the sample mean and variance becomes difficult, especially for a single image. Therefore, estimating the true mean and variance accurately is required to ensure reliable adaptation with a single image. Also, the entropy loss utilized in these methods is unstable because the instability of BN persists due to its linearity. Furthermore, the entropy loss often results in model collapse, causing biased prediction [29].

Therefore, when dealing with a single image, a solution is required that (1) stabilizes the input statistics in the normalization layers and (2) uses stable losses more than minimizing entropy.

Test-time adaptation as a style transfer. In the style transfer domain, there have been several methods that utilize normalization layers for style transfer [7, 16]. Dumoulin *et al.* [7] proposed a conditional instance normalization (CIN) that trains the learnable parameters α^s and β^s to transfer the style of the encoded input x to the style s:

$$\text{CIN}(x; s) = \alpha^s \cdot \frac{x - \mu_x}{\sigma_x} + \beta^s \tag{3}$$

Similarly, AdaIN [16] uses the target style directly in its instance normalization to transfer the style of encoded input x to the style of encoded target style input y:

$$\text{AdaIN}(x, y) = \sigma_y \cdot \frac{x - \mu_x}{\sigma_x} + \mu_y \tag{4}$$

where σ_y and μ_y denote the standard deviation and mean of the encoded target style input y. Despite their simplicity, they have shown promising results in style transfer. However, it is important to note that all inputs utilized in the aforementioned methods are in-distribution, which means that the models are trained on both source and target data, therefore the statistics of the encoded input image (μ_x, σ_x) are reliable. In contrast, cTTA setting involves inputs from an arbitrary target domain that are out-of-distribution, resulting in unreliable and unstable statistics. Motivated by this, we formulate the TTA as a style transfer problem that transfers the target style to the source style:

$$\text{TTA}(x) = \sigma_s \cdot \frac{x - \mu_t}{\sigma_t} + \mu_s \tag{5}$$

where the source and the target domain follow the distributions $N(\mu_s, \sigma_s^2)$ $N(\mu_t, \sigma_t^2)$, respectively.

3.2 BeIN: BESTTA Instance Normalization

We propose a BESTTA Instance Normalization (BeIN) layer that transfers the style of a target input to the source style, enabling seamless operation of the latter parts of a model. For stability, we estimate the target statistics (μ_t, σ_t) using an anchor point and learnable parameters γ_σ and γ_μ. We use the source style $(\overline{\mu_s}, \overline{\sigma_s})$ as the anchor point because it is fixed and therefore stable. The source style contains the mean $\overline{\mu_s}$ and the standard deviation $\overline{\sigma_s}$ of the source features, which are small and can be easily obtained from the training phase, before the deployment of our method. BeIN is formulated as:

$$\text{BeIN}(x) = \overline{\sigma_s} \cdot \frac{x - \hat{\mu}_t}{\hat{\sigma}_t} + \overline{\mu_s} \tag{6}$$

where $\hat{\mu}_t$ and $\hat{\sigma}_t$ denote the estimated target mean and standard deviation, respectively. We estimate the true target statistics by combining the target input statistics and the source style. We estimate $\hat{\sigma}_t$ as a weighted harmonic mean of the source standard deviation and the target input standard deviation with a learnable parameter γ_σ:

$$\hat{\sigma}_t = \frac{\overline{\sigma_s} \cdot \sigma_x}{\rho\overline{\sigma_s} + (1-\rho)\sigma_x + \gamma_\sigma} \tag{7}$$

where ρ is the hyperparameter that adjusts the ratio of using the source statistics. For the mean, we estimate it by the weighted sum of the source mean and the target input mean with a learnable parameter γ_μ:

$$\hat{\mu}_t = \rho\frac{\hat{\sigma}_t}{\sigma_x} \cdot \mu_x + (1-\rho)\frac{\hat{\sigma}_t}{\overline{\sigma_s}} \cdot \overline{\mu_s} + \gamma_\mu \tag{8}$$

The means are scaled with the standard variations to be aligned. We insert BeIN between the layers of the encoder as depicted in Fig. 2. By training only a two parameters in the embedded layer, the latter part of the model seamlessly operates with the preserved prebuilt knowledge, leading to efficient learning and modularization of the whole model.

3.3 Style-guided losses

Conventional style losses [7, 11, 16, 20] can be used to guide the learnable parameters in BeIN to transfer the style of the input features to the source style. These style losses align the distribution of the transferred image to the target style. However, they require heavy computations such as additional encoding or decoding processes or computing the gram matrix. Therefore, considering the efficiency requirement of TTA methods, adopting the conventional style losses is impractical. To address this problem, we propose novel directional style loss and content loss that are computed in the embedding space of encoder during the inference, satisfying efficiency without heavy operations.

Directional style loss. Conventional style transfer methods utilize the similarity between an encoded input and an encoded style input. The straightforward

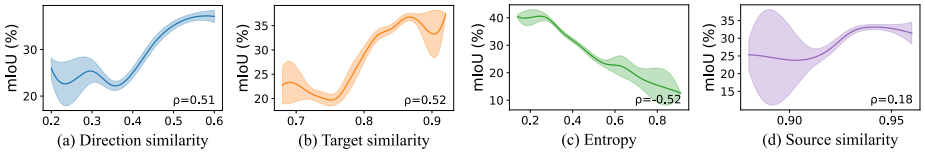

Fig. 3: Correlations between performance and style transfer related metrics. We find that direction similarity and target similarity have high correlation ($\rho > 0.5$) with the performance, whereas source similarity [8, 32] is uncorrelated ($\rho = 0.18$). We use the method of Schneider *et al.* [35] to measure the similarity between the adapted and unadapted embeddings. All results are evaluated on the ACDC dataset [34] using DeepLabV3Plus-ResNet50 [4] pretrained on the Cityscapes dataset [5].

solution to our method is to measure the source similarity $\cos(\overline{\mathcal{E}(x_{src})}, \mathcal{E}'(x))$ [8, 32], that is, the cosine similarity between the adapted target embedding $\mathcal{E}'(x)$ and the mean source embedding $\overline{\mathcal{E}(x_{src})}$. However, as shown in Fig. 3d, we empirically find that the source similarity is not correlated with the performance, thus not beneficial to improve the adaptation. Therefore, motivated by StyleGAN-NADA [9], we devise the directional style loss. We find that the similarity between the adaptation direction $(\overline{\mathcal{E}(x_{src})} - \mathcal{A}_{\mathcal{E}(x)})$ and the direction from the target to the source $(\mathcal{E}'(x) - \mathcal{E}(x))$ (see Fig. 2) has high correlation with the adaptation performance as illustrated in Fig. 3a. Therefore, we formulate our directional style loss as follows:

$$\mathcal{L}_{style} = 1 - \cos((\overline{\mathcal{E}(x_{src})} - \mathcal{A}_{\mathcal{E}(x)}), (\mathcal{E}'(x) - \mathcal{E}(x))) \tag{9}$$

where $\mathcal{E}(x)$ is the unadapted target embedding, $\mathcal{A}_{\mathcal{E}}(x)$ is the exponential moving average of target embeddings.

Content loss. Style transfer without consideration about the content of the transferred feature leads to distortion of the contents [11, 16, 20, 47]. Similar to these findings, we find that the target similarity, that is, the cosine similarity between the transferred feature $\mathcal{E}'(x)$ and the input feature $\mathcal{E}(x)$ has high correlation with the performance, as shown in Fig. 3b. Therefore, we devise the content loss as follows:

$$\mathcal{L}_{content} = 1 - \cos(\mathcal{E}(x), \mathcal{E}'(x)) \tag{10}$$

L2 regularization. In the continual TTA setting, where long-term adaptation is necessary, it is crucial to prevent catastrophic forgetting and error accumulation to ensure optimal performance. Therefore, we employ an L2 norm to regularize the learnable parameters to prevent overfitting to the current target domain. We formulate the L2 regularization loss as follows:

$$\mathcal{L}_{L2} = \|\gamma_\mu\|_2 + \|\gamma_\sigma\|_2 \tag{11}$$

Entropy loss. We incorporate the entropy loss introduced by Wang *et al.* [41], as BeIN stabilize the normalization. We verify that entropy has strong correlation

with the performance on semantic segmentation task as shown in Fig. 3c. The entropy loss is as follows:

$$\mathcal{L}_{entropy} = -\Sigma p(\hat{y}) \log p(\hat{y}) \tag{12}$$

where p denotes probability, $\hat{y}$ denotes prediction.

Total loss. We train the learnable parameters in our proposed BeIN layer with a combination of the proposed losses. The total loss $\mathcal{L}$ is as follows:

$$\mathcal{L} = \lambda_1 \cdot \mathcal{L}_{style} + \lambda_2 \cdot \mathcal{L}_{content} + \lambda_3 \cdot \mathcal{L}_{entropy} + \lambda_4 \cdot \mathcal{L}_{L2} \tag{13}$$

where λ_1, λ_2, λ_3, and λ_4 are the weights for each loss.

4 Experiments

We conduct two experiments on our proposed method in terms of semantic segmentation and image classification. We use the following methods as baselines:

- Fully TTA: BN Stats Adapt [27] and TENT [41]
- Continual TTA: TENT-continual [41], EATA [28], CoTTA [43], EcoTTA [36], and PETAL [1]

4.1 Experiments on Semantic Segmentation

We evaluate our proposed method and other baselines in two different settings on semantic segmentation: Cityscapes-to-ACDC continual TTA and Cityscapes-to-Cityscapes-C gradual TTA. All semantic segmentation results are evaluated in mean intersection over union (mIoU).

Experimental setup. Following the setting of CoTTA [43], we conduct experiments in the continually changing target environment. We adopt the Cityscapes dataset [5] as the source dataset and the Adverse Conditions (ACDC) dataset [34] as the target dataset. The ACDC dataset includes four different adverse weather conditions (i.e., fog, night, rain, and snow) captured in the real-world. We conduct all experiments without the domain label except for TENT [41]. We repeat the same sequence of four weather conditions 10 times (i.e., fog $\rightarrow$ night $\rightarrow$ rain $\rightarrow$ snow $\rightarrow$ fog $\rightarrow \cdots$, 40 conditions in total).

We also conduct experiments in the gradually changing target environment to simulate more realistic environments. We adopt the Cityscapes dataset as the source dataset and the Cityscapes-C dataset [15, 26] as the target dataset. The Cityscapes-C dataset is the corrupted version of the Cityscapes dataset that includes 5 severity levels and 15 types of corruptions. Following EcoTTA [36], we utilize the four most realistic types of corruptions (i.e., brightness, fog, frost, snow) in our experiment. The model faces varying levels of corruption severity for a specific weather type, progressing from 1 to 5 and then from 5 to 1. Once the severity reaches the lowest level, the corruption type is changed (e.g., brightness $1 \rightarrow 2 \rightarrow \cdots \rightarrow 5 \rightarrow 4 \cdots \rightarrow 1 \rightarrow$ fog $1 \rightarrow 2 \rightarrow \cdots$, 36 conditions in total).

Table 2: Semantic segmentation results (mIoU in %) on Cityscapes-to-ACDC single image continual test-time adaptation task. We compare ours with other state-of-the-art TTA methods in terms of peak GPU memory usage (GB) and time consumption (ms) for each iteration, and mean intersection over union (mIoU). All results are evaluated using the DeepLabV3Plus-ResNet50. * denotes the requirement about when the domain shift occurs. The **best** and second best results are highlighted.

Time			$t \longrightarrow$																
Method	Memory (GB)	Time (ms)	Round 1				Round 4				Round 7				Round 10				Mean
			Fog	Night	Rain	Snow	Fog	Night	Rain	Snow	Fog	Night	Rain	Snow	Fog	Night	Rain	Snow	
Source	1.76	135.9	44.3	22.0	41.5	38.9	44.3	22.0	41.5	38.9	44.3	22.0	41.5	38.9	44.3	22.0	41.5	38.9	36.6
BN Stats Adapt [27]	2.01	192.7	36.8	23.5	38.2	36.3	36.8	23.5	38.2	36.3	36.8	23.5	38.2	36.3	36.8	23.5	38.2	36.3	33.7
TENT* [41]	9.58	343.8	38.2	22.9	41.1	37.8	38.2	22.9	41.1	37.8	38.2	22.9	41.1	37.8	38.2	22.9	41.1	37.8	35.0
TENT-continual [41]	9.58	343.8	37.7	23.5	39.9	37.5	31.4	17.3	30.7	26.8	19.8	11.2	18.8	17.3	13.4	8.7	12.4	11.8	22.3
EATA [28]	9.90	190.6	36.8	23.4	38.3	36.5	37.6	23.5	39.0	37.1	38.1	23.6	39.5	37.6	38.5	23.6	39.9	38.0	34.4
CoTTA [43]	18.20	4337	46.3	22.0	44.2	40.4	48.1	21.0	45.3	40.2	48.1	20.4	45.3	39.7	48.0	20.0	45.1	39.5	38.4
EcoTTA [36]	10.90	759.3	33.4	20.8	35.6	32.9	34.2	21.2	36.1	33.2	34.7	21.3	36.3	33.2	34.9	21.4	36.5	33.2	31.2
PETAL [1]	18.51	5120	44.7	22.1	42.9	40.1	47.3	22.4	44.4	40.8	47.1	22.1	44.7	40.6	46.9	22.0	44.6	40.5	38.5
BESTTA (Ours)	8.77	291.8	**47.8**	**24.3**	**47.2**	**43.8**	**54.5**	**26.1**	**48.4**	**45.2**	**54.5**	**26.1**	**48.4**	**45.3**	**54.5**	**26.1**	**48.4**	**45.3**	**43.2**

Table 3: Semantic segmentation results (mIoU in %) on the Cityscapes-to-Cityscapes-C gradual test-time adaptation task. All experiments are evaluated using DeepLabV3Plus-ResNet50.

Time	$t \longrightarrow$				Mean
Method	Bright.	Fog	Frost	Snow	
Source	67.5	59.3	24.8	13.9	41.4
TENT-continual [41]	61.9	45.2	23.0	15.3	36.3
PETAL [1]	64.7	35.8	5.7	0.4	26.7
BESTTA (Ours)	**68.2**	**60.5**	**31.6**	**20.0**	**45.0**

Implementation Details. We utilize ResNet50-DeepLabV3+ [3] pretrained on the Cityscapes dataset. We set the batch size to 1 with an image of size 1920×1080 for the continually changing setting and a 2048×1024 size for the gradually changing setting, respectively. We collect the source style $(\overline{\mu_s}, \overline{\sigma_s})$ and the mean source embedding $\overline{\mathcal{E}(x_{src})}$ by inferencing the source data before deployment. We insert the proposed BeIN layers between the third and fourth layers in the backbone. The λ_1, λ_2, λ_3, and λ_4 are set to 0.3, 1.0, 0.3, and 0.04, respectively. For optimization, we employ the SGD optimizer with a learning rate of 0.001 for the adaptation phase. Note that the model was pretrained with a learning rate of 0.1. The experiments are conducted using an NVIDIA RTX3090 GPU.

Quantitative Results. As shown in Table 2, we compare our approach with the baselines on the cTTA setting. Our proposed BESTTA achieves the highest mIoU across all weather types and rounds. In particular, our method significantly outperforms the second-best baseline, PETAL [1], with a mean mIoU gap of 4.7%, despite using 5.7% of the processing time per image compared to the second-best model. Ours also prevent catastrophic forgetting and error accumulation in the long-term adaptation, as ours shows consistently high performance. In contranst, the BN-based methods [28, 36, 41] show severe performance degradation. Notably, even we reinitialize the TENT reinitialize whenever each domain

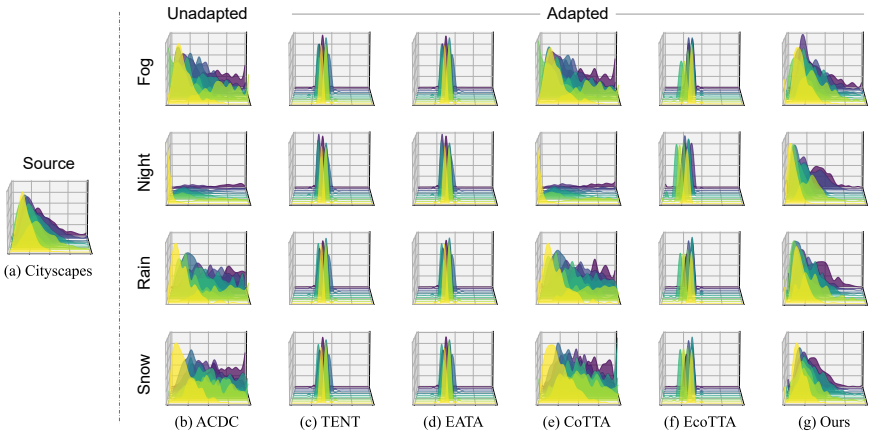

Fig. 4: Embedding feature distributions of adapted embeddings in our method and the baselines. Our method successfully aligns the target distribution (ACDC) to the source distribution (Cityscapes). Our method successfully aligns the embedding distribution with the source distribution, whereas the BN-based methods (TENT [41], EATA [28], EcoTTA [36]) exhibit instability as their embedding distributions collapse. We use DeepLabV3Plus-ResNet50 [3] pretrained on Cityscapes [5] in this experiment.

change occurs, the performance is lower than the source model. This indicates that the unstable mini-batch statistics and entropy loss hinder the adaptation in single image cTTA.

The experimental results for the gradually changing setting are presented in Table 3. Similar to the cTTA setting, our method significantly outperforms the baselines. All the baselines perform worse than the source model, indicating that they fail to deal with the single image cTTA setting.

Qualitative Results. Fig. 4 provides the distributions of the adapted features of ours and the baselines. Comparing Fig. 4a and Fig. 4b, the feature distribution of the target domain that is obtained by the source model is diversified. The BN-based methods, such as TENT, EATA, and EcoTTA, exhibit instability as their embedding distributions collapse to a single point. Although CoTTA does not show distribution collapse, its distribution is closer to the unadapted distribution than the source distribution, indicating that the model is not well adapted to the target domain. In contrast, our adapted features are aligned with the source features from the source model, and the features for each degradation are similar to each other. This demonstrates that our BESTTA effectively and stably adapts the model to the target domain by normalizing the target distribution to the source distribution.

Fig. 5 shows predictions of ours and the baselines on the ACDC dataset. Compared to the other state-of-the-art methods, our results are clearer and more accurate. In particular, at night, the baselines show globally noisy predictions while ours show clear results; in fog and snow, our method perceives the sky

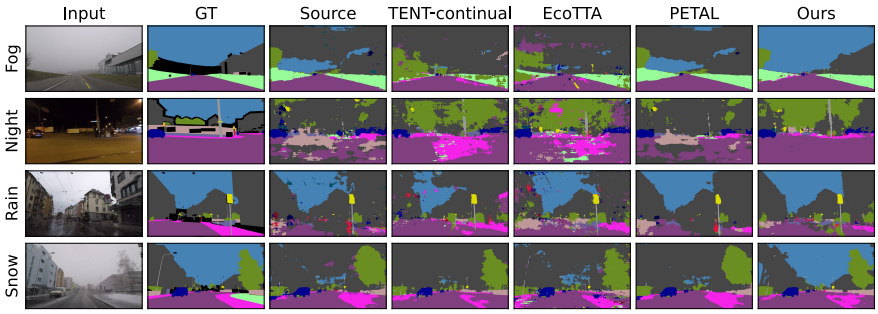

Fig. 5: Qualitative comparison of semantic segmentation on Cityscapes-to-ACDC task. All experiments are evaluated using DeepLabV3Plus-ResNet50. The results of other methods are presented in the supplementary material.

Table 4: Ablation study on our losses. Performances (mIoU in %) are evaluated on Cityscapes-to-ACDC single image continual test-time adaptation task for each different combination of losses, using DeepLabV3Plus-ResNet50 [3]. The results are separated by the number of losses used. Using all of our proposed losses considerably improves the performance. Because the content loss and the L2 regularization are only meaningful with other losses, their individual results are excluded.

$\mathcal{L}_{entropy}$	$\mathcal{L}_{style}$	$\mathcal{L}_{content}$	$\mathcal{L}_{L2}$	Round 1	Round 4	Round 7	Round 10	Mean
✓	✗	✗	✗	21.2	5.3	4.5	4.2	6.7
✗	✓	✗	✗	31.0	25.4	24.4	24.0	25.6
✓	✓	✗	✗	37.1	32.3	30.2	28.9	31.8
✓	✗	✓	✗	21.8	14.0	13.6	13.4	14.7
✓	✗	✗	✓	13.8	5.0	5.0	5.0	5.9
✗	✓	✓	✗	39.0	37.3	37.1	36.9	37.4
✗	✓	✗	✓	38.3	38.6	38.5	38.6	38.5
✓	✓	✓	✗	40.1	39.2	37.9	37.3	38.6
✓	✓	✗	✓	28.8	26.9	26.9	26.9	27.1
✓	✗	✓	✓	37.2	41.1	41.1	41.1	40.7
✗	✓	✓	✓	40.4	40.7	40.7	40.8	40.7
✓	✓	✓	✓	**40.8**	**43.5**	**43.6**	**43.6**	**43.2**

as sky while others predict it as buildings. These observations demonstrate the effectiveness of our method in a real-world environment. Further results are presented in the supplementary material.

Effectiveness of losses. We perform an ablation study on our losses, and the results are provided in Table 4. Updating the parameters in BeIN only with entropy losses yields the lowest performance. Conversely, including style loss significantly improves performance, demonstrating the effectiveness of style transfer in our method. The addition of content loss further improves performance by preserving the content of the input image, allowing only feature styles to be transferred. However, it still exhibits error accumulation, as evidenced by a gradual performance degradation. L2 loss effectively prevents this phenomenon by maintaining stable performance across rounds.

Table 5: Ablation study on the position of BeIN. Performance (mIoU in %) are evaluated on Cityscapes-to-ACDC task, using DeepLabV3Plus-ResNet50.

Layer1	*Layer2*	*Layer3*	*Layer4*	*mIoU*
✓	✗	✗	✗	41.7
✗	✓	✗	✗	40.9
✗	✗	✓	✗	**43.2**
✗	✗	✗	✓	36.0
✓	✗	✓	✗	43.0
✗	✓	✓	✗	41.7
✗	✗	✓	✓	41.4

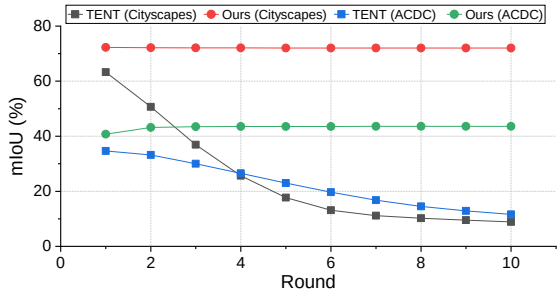

Fig. 6: Robustness to catastrophic forgetting. We evaluated the adapted models after each round on the source dataset (Cityscapes). All experiments are evaluated on Cityscapes-to-ACDC in semantic segmentation using DeepLabV3Plus-ResNet50. In comparison to TENT, ours does not show catastrophic forgetting and error accumulation.

Selection of layer to insert BeIN layer. Table 5 shows the effectiveness of the selection of layers, where the BeIN layer is inserted to transfer the features. Adaptation of features from layer third achieves the highest performance. However, the addition of other layers leads to a performance degradation compared to using only the third layer.

Prevention of catastrophic forgetting. As shown in Fig. 6, we evaluated the performance of adapted models after each round on the source dataset, to assess the robustness to the forgetting. TENT exhibits an error accumulation, reflected in the decrease in performance on the ACDC with each round. In addition, TENT experiences catastrophic forgetting of source knowledge, as evidenced by a rapid decline in performance on Cityscapes (the source dataset). In contrast, our method demonstrates resilience against forgetting source-trained knowledge, with remarkably improved performance across all rounds.

4.2 Experiments on Image Classification

Experimental Setup. We conduct experiments to verify the effectiveness of our method in image classification. Following CoTTA [43], we pretrain the

Table 6: Image classification results on CIFAR-10-C. All results are evaluated using WideResNet-28 pre-trained on CIFAR-10. We use the error rate (%) as the metric. The **lowest** and second lowest error rates are highlighted.

Time	$t \longrightarrow$															Mean
Method	gauss.	shot	impulse	defocus	glass	motion	zoom	snow	frost	fog	bright.	contra.	elastic.	pixel.	jpeg	
Source	88.9	89.2	89.6	77.5	83.2	75.6	77.2	75.2	70.0	68.4	66.9	49.9	81.0	81.2	83.2	77.1
BN Stats Adapt [27]	89.5	89.4	89.7	90.1	89.5	89.5	89.5	89.6	89.6	89.2	89.8	90.1	89.5	89.4	89.6	89.6
TENT-continual [41]	90.0	90.0	90.0	90.0	90.0	90.0	90.0	90.0	90.0	90.0	90.0	90.0	90.0	90.0	90.0	90.0
CoTTA [43]	54.0	64.0	78.0	86.8	88.0	89.9	89.7	87.8	87.6	89.7	88.7	90.0	89.4	53.9	86.7	81.6
PETAL [1]	**51.1**	52.0	77.7	73.4	62.8	79.7	84.1	48.3	67.4	78.9	19.0	81.5	56.5	57.4	69.8	64.0
BESTTA (Ours)	55.2	**50.5**	**65.6**	**38.1**	**46.1**	**29.3**	**32.6**	**23.4**	**33.1**	**21.8**	**9.1**	**38.5**	**24.5**	**44.3**	**28.9**	**36.1**

WideResNet-28 [45] on CIFAR-10 [18] and adapt the network to CIFAR-10-C [15] in the single image cTTA setting.

Results. As provided in Table 6, our approach outperforms other methods in mean accuracy over different corruption types. TENT shows performance that is almost equal to random performance. The second best method, PETAL, is better than the source model, but it is far from satisfactory performance. In contrast, our method exhibits anti-forgetting and maintains high performance over time.

5 Conclusion

In this paper, we propose BESTTA, a style transfer guided continual test-time adaptation (cTTA) method for stable and efficient adaptation, especially in the single image cTTA setting. To stabilize the adaptation, we devise a stable normalization layer, coined BeIN, that incorporates learnable parameters and source statistics. We propose style-guided losses to guide our BeIN to effectively transfer the style of the input image to the source style. Our approach achieves state-of-the-art performance and memory efficiency in the single image cTTA setting. We plan to explore a method to automatically select the best layer to insert our BeIN layer.

Acknowledgment

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.under grant RS-2019-II191126, Self-learning based Autonomic IoT Edge Computing).

References

1. Brahma, D., Rai, P.: A probabilistic framework for lifelong test-time adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3582–3591 (June 2023)
2. Chen, D., Wang, D., Darrell, T., Ebrahimi, S.: Contrastive test-time adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 295–305 (2022)
3. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European conference on computer vision (ECCV). pp. 801–818 (2018)
4. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European Conference on Computer Vision (ECCV) (September 2018)
5. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3213–3223 (2016)
6. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
7. Dumoulin, V., Shlens, J., Kudlur, M.: A learned representation for artistic style. In: International Conference on Learning Representations (2016)
8. Fahes, M., Vu, T.H., Bursuc, A., Pérez, P., de Charette, R.: Poda: Prompt-driven zero-shot domain adaptation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 18623–18633 (2023)
9. Gal, R., Patashnik, O., Maron, H., Bermano, A.H., Chechik, G., Cohen-Or, D.: Stylegan-nada: Clip-guided domain adaptation of image generators. ACM Transactions on Graphics (TOG) **41**(4), 1–13 (2022)
10. Ganin, Y., Lempitsky, V.: Unsupervised domain adaptation by backpropagation. In: International conference on machine learning. pp. 1180–1189. PMLR (2015)
11. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2414–2423 (2016)
12. Gatys, L.A., Ecker, A.S., Bethge, M., Hertzmann, A., Shechtman, E.: Controlling perceptual factors in neural style transfer. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3985–3993 (2017)
13. Gong, T., Jeong, J., Kim, T., Kim, Y., Shin, J., Lee, S.J.: Note: Robust continual test-time adaptation against temporal correlation. Advances in Neural Information Processing Systems **35**, 27253–27266 (2022)
14. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
15. Hendrycks, D., Dietterich, T.: Benchmarking neural network robustness to common corruptions and perturbations. In: International Conference on Learning Representations (2018)
16. Huang, X., Belongie, S.: Arbitrary style transfer in real-time with adaptive instance normalization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1501–1510 (2017)

17. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: International conference on machine learning. pp. 448–456. pmlr (2015)
18. Krizhevsky, A.: Learning Multiple Layers of Features from Tiny Images. Master's thesis, University of Toronto (2009)
19. Lee, J., Das, D., Choo, J., Choi, S.: Towards open-set test-time adaptation utilizing the wisdom of crowds in entropy minimization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 16380–16389 (2023)
20. Li, Y., Wang, N., Liu, J., Hou, X.: Demystifying neural style transfer. In: Proceedings of the 26th International Joint Conference on Artificial Intelligence. pp. 2230–2236 (2017)
21. Li, Y., Fang, C., Yang, J., Wang, Z., Lu, X., Yang, M.H.: Universal style transfer via feature transforms. Advances in neural information processing systems **30** (2017)
22. Lim, H., Kim, B., Choo, J., Choi, S.: Ttn: A domain-shift aware batch normalization in test-time adaptation. In: The Eleventh International Conference on Learning Representations (2022)
23. Liu, Y., Kothari, P., van Delft, B., Bellot-Gurlet, B., Mordan, T., Alahi, A.: Ttt++: When does self-supervised test-time training fail or thrive? In: Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., Vaughan, J.W. (eds.) Advances in Neural Information Processing Systems. vol. 34, pp. 21808–21820. Curran Associates, Inc. (2021), https://proceedings.neurips.cc/paper_files/paper/2021/file/b618c3210e934362ac261db280128c22-Paper.pdf
24. Long, M., Cao, Y., Wang, J., Jordan, M.: Learning transferable features with deep adaptation networks. In: International conference on machine learning. pp. 97–105. PMLR (2015)
25. Long, M., Zhu, H., Wang, J., Jordan, M.I.: Unsupervised domain adaptation with residual transfer networks. Advances in neural information processing systems **29** (2016)
26. Michaelis, C., Mitzkus, B., Geirhos, R., Rusak, E., Bringmann, O., Ecker, A.S., Bethge, M., Brendel, W.: Benchmarking robustness in object detection: Autonomous driving when winter is coming. arXiv preprint arXiv:1907.07484 (2019)
27. Nado, Z., Padhy, S., Sculley, D., D'Amour, A., Lakshminarayanan, B., Snoek, J.: Evaluating prediction-time batch normalization for robustness under covariate shift (2021)
28. Niu, S., Wu, J., Zhang, Y., Chen, Y., Zheng, S., Zhao, P., Tan, M.: Efficient test-time model adaptation without forgetting. In: International conference on machine learning. pp. 16888–16905. PMLR (2022)
29. Niu, S., Wu, J., Zhang, Y., Wen, Z., Chen, Y., Zhao, P., Tan, M.: Towards stable test-time adaptation in dynamic wild world. In: The Eleventh International Conference on Learning Representations (2023), https://openreview.net/forum?id=g2YraF75Tj
30. Pan, S.J., Tsang, I.W., Kwok, J.T., Yang, Q.: Domain adaptation via transfer component analysis. IEEE transactions on neural networks **22**(2), 199–210 (2010)
31. Park, S., Yang, S., Choo, J., Yun, S.: Label shift adapter for test-time adaptation under covariate and label shifts. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 16421–16431 (2023)
32. Patashnik, O., Wu, Z., Shechtman, E., Cohen-Or, D., Lischinski, D.: Styleclip: Text-driven manipulation of stylegan imagery. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2085–2094 (2021)
33. Patel, V.M., Gopalan, R., Li, R., Chellappa, R.: Visual domain adaptation: A survey of recent advances. IEEE signal processing magazine **32**(3), 53–69 (2015)

34. Sakaridis, C., Dai, D., Van Gool, L.: Acdc: The adverse conditions dataset with correspondences for semantic driving scene understanding. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10765–10775 (2021)
35. Schneider, S., Rusak, E., Eck, L., Bringmann, O., Brendel, W., Bethge, M.: Improving robustness against common corruptions by covariate shift adaptation. Advances in neural information processing systems **33**, 11539–11551 (2020)
36. Song, J., Lee, J., Kweon, I.S., Choi, S.: Ecotta: Memory-efficient continual test-time adaptation via self-distilled regularization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11920–11929 (June 2023)
37. Sun, Y., Tzeng, E., Darrell, T., Efros, A.A.: Unsupervised domain adaptation through self-supervision. arXiv preprint arXiv:1909.11825 (2019)
38. Sun, Y., Wang, X., Liu, Z., Miller, J., Efros, A., Hardt, M.: Test-time training with self-supervision for generalization under distribution shifts. In: International conference on machine learning. pp. 9229–9248. PMLR (2020)
39. Tsai, Y.H., Hung, W.C., Schulter, S., Sohn, K., Yang, M.H., Chandraker, M.: Learning to adapt structured output space for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7472–7481 (2018)
40. Ulyanov, D., Lebedev, V., Vedaldi, A., Lempitsky, V.: Texture networks: Feed-forward synthesis of textures and stylized images. arXiv preprint arXiv:1603.03417 (2016)
41. Wang, D., Shelhamer, E., Liu, S., Olshausen, B., Darrell, T.: Tent: Fully test-time adaptation by entropy minimization. In: International Conference on Learning Representations (2021), `https://openreview.net/forum?id=uXl3bZLkr3c`
42. Wang, L., Li, D., Zhu, Y., Tian, L., Shan, Y.: Dual super-resolution learning for semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3774–3783 (2020)
43. Wang, Q., Fink, O., Van Gool, L., Dai, D.: Continual test-time domain adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 7201–7211 (June 2022)
44. Wang, W., Zhong, Z., Wang, W., Chen, X., Ling, C., Wang, B., Sebe, N.: Dynamically instance-guided adaptation: A backward-free approach for test-time domain adaptive semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 24090–24099 (2023)
45. Zagoruyko, S., Komodakis, N.: Wide residual networks. In: Procedings of the British Machine Vision Conference 2016. British Machine Vision Association (2016)
46. Zhang, M., Levine, S., Finn, C.: Memo: Test time robustness via adaptation and augmentation. Advances in Neural Information Processing Systems **35**, 38629–38642 (2022)
47. Zhang, Y., Zhang, Y., Cai, W.: Separating style and content for generalized style transfer. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 8447–8455 (2018)
48. Zhao, H., Qi, X., Shen, X., Shi, J., Jia, J.: Icnet for real-time semantic segmentation on high-resolution images. In: Proceedings of the European conference on computer vision (ECCV). pp. 405–420 (2018)