CAAI Transactions on Intelligence Technology

WILEY

ORIGINAL RESEARCH OPEN ACCESS

Robust Multi-Source Batch Normalisation for Test-Time Batch Adaptation

Xinlin Xiao[1] | Xiangtao Zheng[1] | Jianhua Tao[2] | Xiaoqiang Lu[1]

[1]College of Physics and Information Engineering, Fuzhou University, Fuzhou, China | [2]Beijing National Research Center for Information Science and Technology, Tsinghua University, Beijing, China

Correspondence: Xiangtao Zheng (zhengxiangtao@fzu.edu.cn)

Received: 30 January 2026 | **Revised:** 30 May 2026 | **Accepted:** 4 June 2026

Keywords: multi-source batch normalisation | source domain selection | spectral entropy weighting | test-time batch adaptation

ABSTRACT

Test-Time Batch Adaptation (TTBA) aims to adapt a pre-trained source model to small, unlabelled target batches at test time. The TTBA methods focus on adapting the pre-trained model or the target data in a one-to-one alignment paradigm. However, these one-to-one alignment paradigms assume that the source domain may share the same knowledge with the target domain. It becomes ineffective when the target domain is different from the source domain. In this paper, a multi-source batch normalisation method is introduced. Specifically, spectral similarity clustering selects a small set of representative source domains, spectral entropy-guided weighting combines source statistics based on their relevance to the target domain, and batch-wise reinitialisation with unsupervised refinement stabilises updates. Extensive experiments on multiple benchmark datasets verify the effectiveness of the proposed method, achieving superior performance over state-of-the-art TTBA methods, especially in scenarios with small target batches.

1 | Introduction

Most machine learning models are developed under the assumption that training and test samples are drawn independently from the same underlying distribution. However, in real-world deployment, this assumption is often violated because target data may differ from source data in imaging conditions, noise patterns, weather, sensor characteristics, or other environmental factors. This distributional discrepancy is commonly referred to as domain shift [1–3], and can cause substantial performance degradation when a model trained in the source domain is applied directly to the target domain. To improve robustness under domain shifts, extensive studies have explored adaptation and generalisation strategies. Domain Adaptation (DA) [4, 5] aims to reduce the discrepancy between the source and target domains by transferring knowledge from the labelled source data to the target domain. Most conventional DA methods assume that target-domain data are available during training or adaptation, commonly in an unlabelled form, so that the source-target gap can be explicitly estimated and minimised through feature alignment [6], adversarial learning [7], moment matching [8], or self-training [9, 10]. In contrast, Domain Generalisation (DG) [11] aims to learn a model from one or multiple source domains that can generalise to unseen target domains without accessing target-domain data during training. DG methods typically improve robustness by learning domain-invariant representations, simulating domain shifts [12, 13], or increasing source domain diversity [14, 15]. Nevertheless, DA usually requires target data during adaptation, while DG does not adapt the models with the available test data, which may fail to capture target-specific characteristics.

As a promising alternative, Test-Time Adaptation (TTA) adapts a pre-trained model from the source domain to unseen target

This is an open access article under the terms of the Creative Commons Attribution-NonCommercial-NoDerivs License, which permits use and distribution in any medium, provided the original work is properly cited, the use is non-commercial and no modifications or adaptations are made.

© 2026 The Author(s). *CAAI Transactions on Intelligence Technology* published by John Wiley & Sons Ltd on behalf of The Institution of Engineering and Technology and Chongqing University of Technology.

data, without accessing source domain data [16, 17]. In practical scenarios, the target data usually arrives sequentially in the form of mini-batches rather than being available all at once [18–21]. This setting, referred to as test-time batch adaptation (TTBA), requires adapting the pre-trained model on small batches of target instances [22–24]. The TTBA methods focus on constructing an adapter to align the pre-trained model with the target domain data, which can be broadly categorised into two groups: the model-level adapter methods and the data-level adapter methods. The model-level adapter methods adjust a subset of the pre-trained model layers to align with the target batch (e.g., Batch Normalisation). For example, some Batch Normalisation (BN) adapter [25–27] suggest that the BN layers represent domain-specific knowledge. To bridge the domain gap, these methods often adjust the training BN statistics with new statistics estimated over the target batch or update BN parameters by minimising an unsupervised loss computed from pseudo-labels on the target batch. By adjusting only certain layers, the model-level adapter methods offer lightweight, test-time adaptation that can improve robustness to domain shift without retraining the whole network or revisiting source data. However, continuously updating parameters based on the target batch can lead to catastrophic forgetting of previously learnt knowledge. These methods require a large number of test samples to update the parameters of certain layers, which is unsuitable for small-batch settings.

Data-level adapter methods change the target batch to align with the pre-trained model, such as the image-to-image translator [28], or the input adapter [29]. For example, Karani et al. proposed an image-to-image translator (i.e., an input adapter) for each target image, which exploited a frozen pre-trained denoising autoencoder to process the adapted input. Shu et al. [30] froze the pre-trained multi-modal model and only learnt the extra text prompt. The data-level adapter methods adjust the target data instead of the pre-trained model, which can avoid parameter drift. These methods transformed the target domain into the source-like space through input-level adjustments. However, if the target domain falls outside the representational capacity of the pre-trained model, the performance remains inherently limited.

As illustrated in Figure 1 Both TTBA methods focus on adapting either the pre-trained model or the target data in a one-to-one alignment paradigm. These one-to-one alignment paradigms assume that the source domain may share the same knowledge with the target domain shown in 1. It becomes ineffective when the target domain is different from the source domain, such as a small number of samples or different perturbations.

To address the above issues, a multi-source batch normalisation method is proposed to group multiple related models, as illustrated in Figure 2. Multiple diverse models have been reported to contain complementary information [31–33], which can rapidly converge with limited target data. Furthermore, in more general cases, the target distribution is approximated as a weighted combination of multiple models, accounting for various interferences such as Gaussian/Poisson noise, background clutter, illumination, or temperature shifts, with simulated disturbances from a few base classes (e.g., noise, blur, occlusion). Specifically, spectral similarity based clustering is used to select representative source domains from the original multiple sources, thus avoiding interference from irrelevant domains. Then, a spectral entropy-guided weighting is introduced, where the spectral entropy discrepancy between each representative source domain and the target batch is measured, and fusion weights are adaptively assigned to maximise the relevance of source statistics. Finally, to mitigate the mismatch between updated BN statistics and affine parameters, a batch-wise affine parameter re-initialisation is applied, followed by unsupervised optimization within each test batch. The proposed method remains within the TTBA paradigm, but its current corruption-benchmark implementation relies on a pool of source domain BN statistics constructed from corruption-based source variants. Therefore, the effect of the source-statistics construction protocol is discussed explicitly in the experimental section, including an additional reduced-overlap evaluation. Importantly, it should be emphasised that the proposed multi-source BN adaptation method relies strictly on offline-extracted, lightweight batch normalisation statistics and spectral features. The online test-time adaptation phase operates without any access to the raw source data, thereby ensuring privacy preservation and minimal memory overhead, which is consistent with the standard constraints of test-time adaptation. The major contributions can be summarised as follows:

1. A multi-source BN adaptation method is proposed for TTBA, where multiple source domain BN statistics are selectively integrated to improve adaptation stability under small target batches.
2. A spectral similarity based representative source selection strategy is introduced to reduce redundancy among source domains and suppress interference from irrelevant statistics.
3. A spectral entropy guided weighting mechanism together with batch-wise affine re-initialisation is developed to improve the consistency between fused normalisation statistics and BN affine parameters during online adaptation.

The rest of this paper is organised as follows. In Section 2, some related works about this paper are summarised. In Section 3, the proposed method is presented. Section 4 reports the experimental results. Section 5 draws the final conclusion.

2 | Related Work

This section briefly reviews the most relevant literature on TTBA, DA, and DG, and clarifies their relationship to the proposed multi-source BN adaptation method.

2.1 | Test-Time Batch Adaptation

The existing TTBA methods often build an adapter to align the pre-trained source model and the target domain. These TTBA methods can be divided into two broad categories: the model-level adapter methods and data-level adapter methods.

The model-level adapter methods construct adapters within the pretrained source model, which adjust its parameters (e.g., BN statistics) to align with the target batch. Nado et al. [25] proposed Predictive Batch Normalisation (PredBN), where the BN layer itself serves as an 'adapter,' replacing the training statistics with those estimated separately for each test batch. Wang et al.

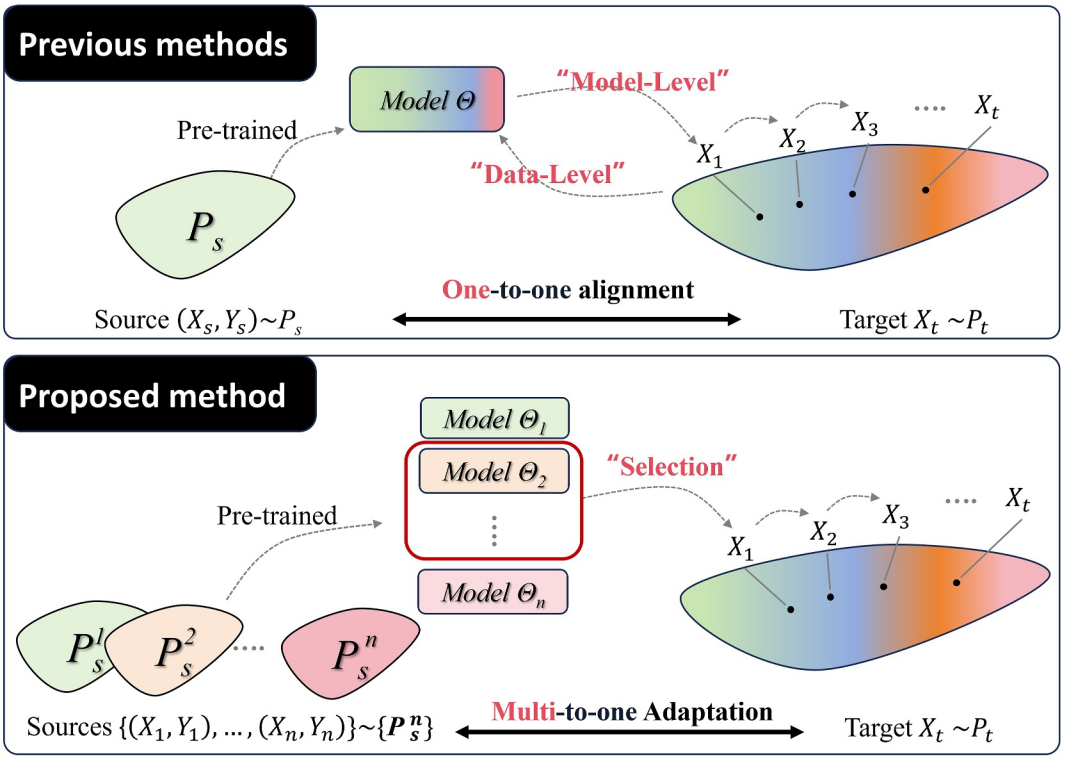


FIGURE 1 | Schematic comparison of traditional versus proposed multi-source domain adaptation method. The upper part depicts the conventional single-source approach, while the lower part illustrates the proposed multi-source BN adaptation method, which involves similarity assessment at both model and data levels and a multi-to-one adaptation process.

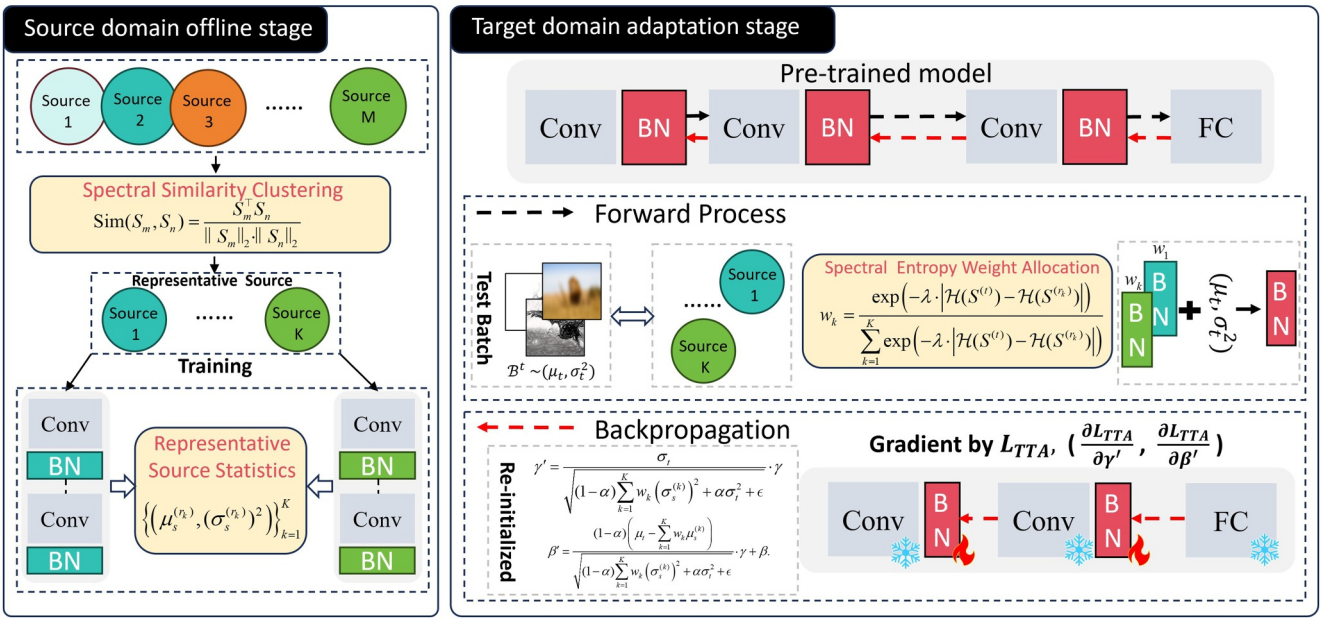


FIGURE 2 | Overall pipeline of the proposed multi-source batch normalisation test-time adaptation method. The left part illustrates the Source domain offline stage, which spectral similarity clustering is performed to select representative source domains and extract their corresponding BN statistics. The right part shows the Target domain adaptation stage, which a spectral entropy weight allocation strategy dynamically matches the test batch to the most relevant representative sources, re-initialises the BN parameters γ and β by integrating source statistics with test batch statistics, and updates them through backpropagation using the test-time adaptation loss $\mathcal{L}_{\mathrm{TTA}}$.

[34] proposed Test Entropy Minimisation (TENT), which introduces a lightweight BN-based 'adapter' by updating only the affine parameters of BN layers through entropy minimisation. Wang et al. [35] proposed Continual Test-Time Adaptation (CoTTA), which designs a teacher-student dual model 'adapter,' where the teacher weights are updated via Exponential Moving

Averaging (EMA) and stochastic weight perturbations are periodically injected for stability. Yuan et al. [36] proposed Robust Test-Time Adaptation (RoTTA), which replaces the standard BN with a robust BN 'adapter' to more accurately estimate target domain statistics under distribution shift. Jin et al. [37] proposed Robust Mean Teacher (RMT), which builds a consistency-based 'adapter' by leveraging symmetric cross-entropy as a consistency loss and integrating contrastive learning to align the target feature space with the source distribution. Niu et al. [38] constructs a sharpness-aware optimization (SAR) 'adapter' to suppress unreliable online updates during adaptation. Niu et al. [39] proposed Efficient Anti-forgetting Test-time Adaptation (EATA), which designs a sample-filtering 'adapter' by selecting informative target samples and discarding redundant or unreliable ones during online updating. Lee et al. [40] proposed DeYO, which builds a selective-update 'adapter' by identifying more reliable target samples from the perspective of disentangled adaptation factors. Gao et al. [41] proposed Unified Entropy Optimization (UniEnt), which introduces an entropy-based open-set 'adapter' by jointly optimising pseudo in-distribution and pseudo out-of-distribution samples under semantic and covariate shifts. However, continuously updating parameters based on the target batch can lead to catastrophic forgetting of previously learnt knowledge. These methods require a large number of test samples to update the parameters of certain layers, which is unsuitable for small-batch settings.

In contrast, data-level adapter methods construct an adapter outside the pre-trained source model, keeping its parameters fixed, with the adapter responsible for modifying the input target batch (e.g., style transfer) to align with the source model. This design can reduce parameter drift because the source model is largely kept fixed. For example, Pandey et al. [42] proposed Input Transformation for Test-time Prediction (ITTP), where a generative model acts as a data-level 'adapter' by projecting target features onto the source feature manifold for inference. Zhao et al. [43] proposed Training-time Fourier Calibration (TAF-Cal), which employs a Fourier-style 'adapter' based on the mean amplitude features of training data to calibrate both training and test samples. Tsai et al. [44] proposed Convolutional Visual Prompts (CVP), which design prompt-based 'adapters' in the input space and optimise them under a self-supervised contrastive objective. Termöhlen et al. [45] proposed Online Style Transfer (OST), which applies a Fourier-style transfer 'adapter' to map target inputs onto the source domain manifold. Alfarra et al. [46] proposed Anti-Adversarial perturbation (Anti-Adv), which perturbs target test batches as a defensive ''adapter' to increase the classification confidence of the source model. Gan et al. [47] proposed Visual Domain Prompting (VDP), where prompt-based 'adapters' dynamically modify parts of the input image to handle domain shifts and reduce online error accumulation. Zhang et al. [48] proposed Marginal Entropy Minimisation (MEMO), which leverages stochastic augmentation as a data-level 'adapter' by generating multiple transformed versions of each test sample and minimising the entropy of the averaged prediction. Nevertheless, data-level adapters also have limitations. Their effectiveness depends on whether the input transformation can faithfully map target samples into a source-compatible space. Some methods require additional image translators or generative models, which may introduce extra deployment costs. Moreover, if the target distribution lies outside the representational capacity of the fixed source model, modifying the input alone may be insufficient. Although model-level and data-level adapters operate on different objects, both categories often follow a one-to-one alignment paradigm between a single source model and the current target batch. This paradigm provides a limited statistical reference when the target batch is small or heavily shifted.

2.2 | Domain Adaptation

DA methods aimed to align feature distributions between source and target domains [4, 5, 49–51], and often used multiple source domains to improve generalisation [52]. Ganin et al. [53] proposed the Domain-Adversarial Neural Network (DANN), which employs a Gradient Reversal Layer (GRL) to perform adversarial learning and align feature distributions across domains. Long et al. [54] proposed Conditional Domain Adversarial Networks (CDAN), which introduce a conditioning mechanism based on task-specific classifier predictions to achieve more class-aware domain-invariant feature learning. Gretton et al. [55] proposed the Maximum Mean Discrepancy (MMD), which measures the distance between domain embeddings in a Reproducing Kernel Hilbert Space (RKHS). Sun et al. [56] proposed the Correlation Alignment (CORAL), which aligns the second-order statistics (covariances) of the source and target features. Courty et al. [57] proposed Joint Distribution Optimal Transport (JDOT), which models both feature and label distributions within an optimal transport method for more fine-grained alignment. More recently, Lü et al. [58] proposed Self-Supervised multi-Source Domain Adaptation with Label-specific Confidence (S3DA-LC), which alleviates imbalanced pseudo-label distributions and improves multi-source domain adaptation by enriching pseudo-supervised signals for target samples. Although these methods are effective, they typically assume that the data in the target domain is accessible. However, when privacy constraints or data invisibility are present, these methods often fail.

2.3 | Domain Generalisation

DG aims to learn models from one or more different but related domains so that they can perform well on unseen test domains. Balaji et al. [59] proposed Meta-Regularisation (MetaReg), which partitions the training data into meta-train and meta-test subsets to simulate domain shifts during training, allowing the model to learn more generalisable parameters. Li et al. [60] proposed domain generalisation through episodic training, which adopts a similar meta-learning strategy to improve adaptability to unseen domains. Gidaris et al. [61] proposed an unsupervised representation learning method that employs self-supervised objectives independent of downstream tasks, thus reducing the dependency on task-specific supervision. Zhang et al. [62] proposed Mixup, which generates new training samples by linearly interpolating two images and their corresponding labels to enhance the performance of the model in unseen domains. Zhou et al. [63] proposed MixStyle, which synthesises feature statistics from different instances to encourage style-invariant representations. Xu et al. [64] proposed Functional Assessment of Chronic Illness Therapy (FACT), which combines low-frequency spectral components

from multiple images to capture structural information invariant to style variations. Cheng et al. [65] proposed Disentangled Prompt Representation (DPR), which leverages vision–language model guidance to separate domain-invariant from domain-specific features. Aminbeidokhti et al. [66] proposed Rejecting Extreme Augmentations (REA), which avoids harmful augmentations while maintaining sample diversity, thus improving robustness under large domain shifts. Although these methods improve generalisation during training, the DG inherently disregards any information about the target domain. This omission constrains its adaptability, particularly when the target domain exhibits substantial distributional discrepancies from the source domains.

3 | The Proposed Method

This section details the proposed multi-source batch normalisation method for TTBA. In Section 3.1, the concept of multi-source batch normalisation is formalised. Section 3.2 introduces the source domain selection method, which is based on spectral similarity clustering. Finally, in Section 3.3, a weighted adaptation guided by spectral entropy is proposed.

3.1 | Multi-Source Batch Normalisation

Let $\mathcal{X}$ and $\mathcal{Y}$ denote the input images and the corresponding category labels, respectively. A parameterised learner $f(\cdot;\theta):\mathcal{X}\to\mathcal{Y}$ is pre-trained in the labelled source domain $\mathcal{D}_s=\{(\mathbf{x}_s^{(i)},y_s^{(i)})\}_{i=1}^{N}$, where $\mathbf{x}_s^{(i)}\in\mathcal{X}$ and $y_s^{(i)}\in\mathcal{Y}$ represent the i-th sample and its corresponding label from the source domain, and N is the total number of labelled sample-label pairs in $\mathcal{D}_s$. TTBA aims to predict the labels for a test batch $\mathcal{B}=\{\mathbf{x}_t^{(1)},...,\mathbf{x}_t^{(j)}\}$, where each sample $\mathbf{x}_t^{(j)}\in\mathcal{X}$ comes from the unlabelled target domain. During inference, TTBA receives these batches sequentially and adapts the model accordingly.

Batch Normalisation: Commonly used TTBA methods dynamically adjust BN statistics during testing, without using target labels or revisiting source data. To standardise each incoming sample, the images x are adjusted by statistics (μ,σ^2) and affine parameters (γ,β), which can be expressed as:

$$\hat{x}=\gamma\cdot\frac{x-\mu}{\sqrt{\sigma^2+\epsilon}}+\beta, \tag{1}$$

where ϵ is a small constant for numerical stability. Where ϵ is a small constant for numerical stability. BN statistics (μ,σ^2) denote the mean and variance normalisation, which standardise each incoming sample x. The affine parameters (γ,β) control the transformation with scale γ and shift β, which are optimised by back-propagation.

BN statistics have been suggested to effectively reflect the underlying distribution of unlabelled target data [34]. Generally, normalisation statistics (μ,σ^2) are approximated by target statistics (μ_t,σ_t^2) at the batch-level [67]. However, only target statistics are insufficient, especially when the test batch is small or noisy, leading to unstable or biased estimation. Therefore, both source and target statistics are integrated into the adaptation process [68]:

$$\hat{\mathbf{x}}_t^{(j)}=\gamma\cdot\frac{\mathbf{x}_t^{(j)}-\left((1-\alpha)\mu_s+\alpha\mu_t\right)}{\sqrt{(1-\alpha)\sigma_s^2+\alpha\sigma_t^2+\epsilon}}+\beta, \tag{2}$$

where the outputs $\hat{x}_t^{(j)}$ are normalised and transformed from the j-th sample $\mathbf{x}_t^{(j)}$. The source statistics (μ_s,σ_s^2) are computed via an EMA over the source data, α controls the balance between the source statistics and the target statistics.

In practice, multiple source domains are available [31], and these multiple source domains provide the ability to leverage diverse information. The M source statistics are represented as $\left\{\left(\mu_s^{(m)},\left(\sigma_s^{(m)}\right)^2\right)\right\}_{m=1}^{M}$, where $\left(\mu_s^{(m)},\left(\sigma_s^{(m)}\right)^2\right)$ denote the m-th source statistics.

3.2 | Source Domain Selection

Given the source data M, it is difficult to leverage multiple Batch Normalisation parameters. Each source domain may be different, and some source domains are irrelevant to the target domain. It is reasonable to select K representative source domains to apply BN. Based on the selected K source domains, the multi-source BN can be rewritten as follows:

$$\mathbf{x}_t^{(j)}=\gamma\cdot\frac{\mathbf{x}_t^{(j)}-\left((1-\alpha)\sum_{k=1}^{K}w_k\mu_s^{(k)}+\alpha\mu_t\right)}{\sqrt{(1-\alpha)\sum_{k=1}^{K}w_k\left(\sigma_s^{(k)}\right)^2+\alpha\sigma_t^2+\epsilon}}+\beta, \tag{3}$$

where w_k measures the weights of multiple source statistics, and $\sum_{k=1}^{K}w_k=1, w_k\geq 0$. The hyper-parameter α balances the contributions between the source and target domain statistics.

To select K source domains, spectral similarity based clustering finds the K clusters to represent the whole M source statistics.

For each source domain D_m, a spectral descriptor is constructed from the frequency-domain information of its samples. Given an image x, a two-dimensional discrete Fourier transform is first applied, and the corresponding magnitude spectrum is obtained as $A(x)=|\mathcal{F}(x)|$. The magnitude spectrum is then normalised into a spectral energy map:

$$P(x)=\frac{A(x)}{\sum_{u,v}A_{u,v}(x)+\epsilon}, \tag{4}$$

where (u,v) denotes the frequency location and ϵ is a small constant for numerical stability. Thus, $P(x)$ is a non-negative normalised spectral distribution. For colour images, the spectral energy maps are computed channel-wise and averaged across channels. The domain-level spectral descriptor of the m-th source domain is computed as:

$$S_m=\frac{1}{|D_m|}\sum_{x\in D_m}P(x). \tag{5}$$

Since S_m and S_n are matrix-form spectral descriptors, they are vectorised before computing the cosine similarity. The pairwise similarity between two source domains D_m and D_n is defined as:

$$\mathrm{Sim}(S_m, S_n) = \frac{\mathrm{vec}(S_m)^\top \mathrm{vec}(S_n)}{\|\mathrm{vec}(S_m)\|_2 \|\mathrm{vec}(S_n)\|_2}. \tag{6}$$

Here, $\mathrm{Sim}(S_m, S_n)$ is a scalar similarity value. The full similarity matrix $W \in \mathbb{R}^{M\times M}$ is then constructed by assigning each entry as:

$$W_{m,n} = \mathrm{Sim}(S_m, S_n), \quad m, n = 1, ..., M. \tag{7}$$

Then, using the standard *K-medoids* clustering algorithm, the M source domains are partitioned into K clusters:

$$\{G_1, G_2, ..., G_K\}, \quad G_k \subseteq \{1, 2, ..., M\}, \tag{8}$$

where G_k denotes the set of member indices in the k-th cluster. The centre of this cluster, denoted as c_k, was computed as the mean of all features in G_k:

$$c_k = \frac{1}{|G_k|} \sum_{m\in G_k} S_m. \tag{9}$$

The source domain closest to the cluster centre is selected as the representative domain:

$$r_k = \arg\min_{m\in G_k} \|S_m - c_k\|_F. \tag{10}$$

Through the spectral similarity based clustering process, K representative source domains are selected from the original M domains. Therefore, the final set of representative statistics is given by: $\left\{\left(\mu_s^{(r_k)}, \left(\sigma_s^{(r_k)}\right)^2\right)\right\}_{k=1}^{K}$. To ensure a fair comparison with standard TTA settings, it is strictly maintained that the proposed method does not store or revisit raw source images during the online adaptation phase. The 'multi-source' information is represented solely by a collection of BN statistics $\left\{\left(\mu_s^{(m)}, \left(\sigma_s^{(m)}\right)^2\right)\right\}_{m=1}^{M}$ and their corresponding domain-level spectral features $\{S_m\}_{m=1}^{M}$. For a typical deep network architecture, these statistical vectors introduce a negligible storage overhead (e.g., less than 0.1% of the total model size), maintaining the lightweight and privacy-preserving nature of the TTBA paradigm.

3.3 | Spectral Entropy-Guided Weighted Adaptation

Given the representative K source statistics $\left\{\left(\mu_s^{(r_k)}, \left(\sigma_s^{(r_k)}\right)^2\right)\right\}_{k=1}^{K}$, the spectral entropy discrepancy Δ_k between the k-th representative source domain and the test batch is utilised to compute the weight for each representative statistics. The each representative k-th source data spectral entropy $\mathcal{H}^{(r_k)}$ and test batch spectral entropy $\mathcal{H}^{(t)}$ are computed as:

$$\mathcal{H}^{(r_k)} = \frac{1}{\left|\mathcal{D}^{(r_k)}\right|} \sum_{i\in\mathcal{D}^{(r_k)}} P_i \log P_i, \tag{11}$$

$$\mathcal{H}^{(t)} = \frac{1}{\left|\mathcal{B}^{(t)}\right|} \sum_{j\in\mathcal{B}^{(t)}} P_j \log P_j, \tag{12}$$

where $\mathcal{D}^{(r_k)}$ denotes the k-th representative source domain, and $\mathcal{B}^{(t)}$ denotes the test batch. P_i and P_j represent the normalised spectral energy distributions of the i-th source domain sample and the j-th test sample, respectively. Δ_k denoted the absolute difference between $\mathcal{H}^{(r_k)}$ and $\mathcal{H}^{(t)}$. The adaptive weight ω_k for k-th source is assigned as:

$$\omega_k = \frac{\exp(-\tau\Delta_k)}{\sum_{k=1}^{K}\exp(-\tau\Delta_k)}, \tag{13}$$

where $\tau > 0$ is a temperature parameter controlling the sharpness of the weight distribution. Since w_k is obtained by a softmax function over the negative discrepancy $-\Delta_k$, it is monotonically decreasing with respect to Δ_k. For any two representative source domains a and b, if $\Delta_a < \Delta_b$, then $\exp(-\tau\Delta_a) > \exp(-\tau\Delta_b)$ for $\tau > 0$, which leads to $w_a > w_b$ after normalisation. Therefore, a representative source domain with a smaller spectral-entropy discrepancy to the current test batch receives a larger weight. In this sense, w_k should be interpreted as a relevance score under the proposed spectral-entropy criterion, rather than as a universal guarantee of semantic or distributional closeness.

When the BN layers incorporated both source and target domain statistics, the original affine parameters γ and β became inconsistent with the updated statistics. To mitigate this inconsistency, γ and β are re-initialised to γ' and β' for each incoming test batch, as defined in Equations (14 and 15).

$$\gamma' = \frac{\sigma_t}{\sqrt{(1-\alpha)\sum_{k=1}^{K} w_k\left(\sigma_s^{(k)}\right)^2 + \alpha\sigma_t^2 + \epsilon}} \cdot \gamma, \tag{14}$$

$$\beta' = \frac{(1-\alpha)\left(\mu_t - \sum_{k=1}^{K} w_k \mu_s^{(k)}\right)}{\sqrt{(1-\alpha)\sum_{k=1}^{K} w_k\left(\sigma_s^{(k)}\right)^2 + \alpha\sigma_t^2 + \epsilon}} \cdot \gamma + \beta. \tag{15}$$

The operation is performed after the test batch has been fed into the network and before the fine-tuning process. Subsequently, γ' and β' are further optimised within the adaptation process of each individual test batch through back-propagation, guided by an unsupervised cross-entropy loss like Equation 16:

$$\mathcal{L}_{\mathrm{TTA}} = -\sum_j \hat{y}_t^{(j)} \log f\left(\mathbf{x}_t^{(j)}; \theta\right), \tag{16}$$

where $x_t^{(j)}$ denotes the j-th target-domain sample, $f\left(x_t^{(j)}; \theta\right)$ is the softmax prediction of the model, and $\hat{y}_t^{(j)}$ is the pseudo-label generated from the model prediction. Specifically, a target sample is used for unsupervised refinement only when its maximum predicted probability is larger than a confidence threshold η. Samples with confidence lower than η are excluded from the loss to reduce noisy pseudo-label supervision. The complete procedure is summarised in Algorithm 1.

ALGORITHM 1 | Online multi-source BN adaptation for TTBA.

Input: Target batch $B_t = \left\{x_t^{(j)}\right\}_{j=1}^{J}$; x representative BN statistics $\left\{\left(\mu_s^{(k)}, \left(\sigma_s^{(k)}\right)^2\right)\right\}_{k=1}^{K}$; x representative source spectral entropy $\left\{H^{(r_k)}\right\}_{k=1}^{K}$; x pre-trained model $f(\cdot;\theta)$; x confidence threshold η; balance parameter α; temperature parameter τ.

Output: Adapted prediction for B_t.

Freeze the backbone and classifier parameters of $f(\cdot;\theta)$.

Computetarget-batch BN statistics $\left(\mu_t, \sigma_t^2\right)$.

Compute the spectral entropy $H^{(t)}$ of the incoming target batch.

for $k = 1$ to K **do**

  Compute the spectral-entropy discrepancy $\Delta_k = |H^{(r_k)} - H^{(t)}|$.

end for

Compute the source domain fusion weights:

$$\omega_k = \frac{\exp(-\tau\Delta_k)}{\sum_{l=1}^{K}\exp(-\tau\Delta_l)}.$$

Fuse representative source domain BN statistics with target-batch statistics according to Equation (3).

Re-initialise BN affine parameters (γ, β) as $\left(\gamma', \beta'\right)$ according to Equations (12) and (13).

Obtain predictions $p_t^{(j)} = f\left(x_t^{(j)};\theta\right)$ for each target sample.

Generate pseudo-labels by confidence thresholding:

$$\hat{y}_t^{(j)} = \arg\max_c p_{t,c}^{(j)}, \quad \text{if } \max_c p_{t,c}^{(j)} > \eta.$$

Exclude samples with $\max_c p_{t,c}^{(j)} \le \eta$ from the refinement loss.

Update only the BN affine parameters $\left(\gamma', \beta'\right)$ by minimising Equation (14) for one gradient step.

Predict the labels of B_t using the adapted BN affine parameters.

4 | Experiment

Datasets: To evaluate the proposed method, three commonly used benchmark datasets and their corrupted counterparts are adopted [69]. The details are summarised as follows:

1. CIFAR10/CIFAR10-C: CIFAR10 consists of 10 classes with a total of 60,000 images (50,000 training and 10,000 testing images). CIFAR10-C is the corrupted counterpart of CIFAR10, containing 15 corruption types with 5 severity levels.
2. CIFAR100/CIFAR100-C: CIFAR100 consists of 100 classes with a total of 60,000 images (50,000 training and 10,000 testing images). CIFAR100-C is the corrupted counterpart of CIFAR100, sharing the same 15 corruption types and 5 severity levels as CIFAR10-C.
3. ImageNet/ImageNet-C: ImageNet contains 1000 classes with 50,000 validation images. ImageNet-C is the corrupted version of ImageNet, including the same 15 corruption types and 5 severity levels, with 5000 images per corruption type.

In the experiments, CIFAR10, CIFAR100, and ImageNet are employed as source domains, while their corrupted counterparts CIFAR10-C, CIFAR100-C, and ImageNet-C are served as target domains. Evaluation is conducted under the highest corruption severity (level 5), and classification error rates are reported. To construct the source domain BN statistics used in the corruption-benchmark experiments, corruption-based source variants are generated from the clean source dataset following the corruption taxonomy of the corresponding benchmark. This protocol provides a structured source-statistics pool for evaluating the proposed multi-source BN adaptation method, and its implication for fairness is discussed explicitly in the manuscript.

Protocol discussion: In the current corruption-benchmark experiments, the source domain BN statistics are constructed using corruption-based source variants associated with the target benchmark. Therefore, the resulting evaluation protocol is not fully assumption-matched to standard single-source TTBA baselines. The results should thus be interpreted with this protocol difference in mind. The purpose of the current study is to evaluate whether representative source domain statistics can improve the stability of batch adaptation under corruption-oriented test-time shifts.

Implementation details: All experiments are conducted with PyTorch. The pre-trained model is obtained from the RobustBench benchmark, including ResNet-56 [70] for CIFAR10 → CIFAR10-C and ResNet-101 [70] for CIFAR100 → CIFAR100-C, DenseNet-BC [71] for ImageNet → ImageNet-C. SGD with a momentum of 0.9 is employed. τ is set to 0.05. The learning rates are set to 0.005, 0.0005, and 0.00025 for the CIFAR10, CIFAR100, and ImageNet datasets. The batch size is set to 32. The pseudo-labels used in Equation (14) are generated by confidence thresholding with $\eta = 0.9$. Low-confidence samples are excluded from the refinement loss.

4.1 | Main Results

Comparison to Previous Methods: To evaluate the performance of the proposed method, it is compared with several representative TTBA/TTA methods under a unified evaluation protocol. The compared methods are briefly introduced as follows:

1. Baseline [72]: Inference is performed using the model trained directly from the source in the target domain without adaptation.
2. BN Adapt [67]: During inference, the BN running statistics are replaced by the statistics of the current target batch.
3. TENT [34]: The model is adapted in test time by minimising the prediction entropy on unlabelled target samples.
4. CoTTA [35]: A continual adaptation method for non-stationary test streams. It employs weighted averaging and augmentation-based prediction to alleviate error accumulation while randomly restoring a small portion of source parameters to reduce catastrophic forgetting.
5. RoTTA [36]: A robust test-time adaptation method that improves target-domain normalisation and online adaptation stability under dynamic distribution shifts.

6. RMT [37]: A mean-teacher-based adaptation method that combines consistency regularisation with contrastive learning to align the target feature space with the source model.
7. SDA-TTA [73]: A diffusion-driven test-time adaptation method that introduces a synthetic intermediate domain to bridge the gap between the source and target domains.
8. MEMO [48]: The model is adapted in test time using multiple augmented views of each target sample, and the adaptation objective minimises the marginal entropy over the augmented predictions.
9. SAR [38]: A test-time adaptation method that improves adaptation stability by suppressing unreliable updates through sharpness-aware optimization.
10. EATA [39]: A selective adaptation method that filters redundant or unreliable target samples to reduce harmful online updates and catastrophic forgetting.
11. DeYO [40]: A selective test-time adaptation method that reduces harmful updates by identifying more reliable target samples from the perspective of disentangled adaptation factors.

As presented in Tables 1–3, the proposed method achieves the lowest average test error rates on CIFAR10-C, CIFAR100-C, and ImageNet-C, reaching 8.39%, 23.35%, and 48.21%, respectively. Compared with representative baselines, including TENT, CoTTA, RoTTA, RMT, MEMO, SAR, EATA, DeYO, and SDA-TTA, the proposed method shows consistent overall advantages across the three benchmarks. In particular, clear improvements are observed under several challenging corruption types, such as Gaussian noise, Impulse noise, Glass blur, and JPEG compression. Although some competing methods achieve competitive performance on specific corruption categories, the proposed method provides the best overall trade-off across diverse corruption types and datasets. These results demonstrate that the proposed multi-source BN adaptation method yields stronger robustness and more stable generalisation under test-time distribution shifts, especially in small-batch adaptation scenarios.

Effect of Batch Size: To further evaluate the adaptation performance under limited test-batch conditions, the experiments shown in Figure 3 are conducted. The results show that the proposed method consistently maintains the lowest error rates across the batch-size range from 4 to 64 on both CIFAR10 → CIFAR10-C and CIFAR100 → CIFAR100-C. In contrast, most existing methods become more unstable when the batch size is small, especially BN Adapt, TENT, CoTTA, and RMT. Although methods such as RoTTA, MEMO, SAR, EATA, and DeYO show improved robustness, noticeable fluctuations can still be observed as the batch size changes. These results indicate that the proposed method is less sensitive to insufficient target-batch information and can provide more stable adaptation in practical small-batch TTBA scenarios.

Evaluating catastrophic forgetting: Figure 4 evaluates source domain knowledge retention under sequential test-time adaptation. Starting from the source model, 2, 4, 6, 8, and 10 rounds of adaptation are performed, and the classification accuracy on the CIFAR10 test set is measured after each round. As shown in Figure 4, most compared methods suffer from different degrees of source-accuracy degradation as the number of adaptation rounds increases, indicating catastrophic forgetting. In contrast, the proposed method maintains the highest and most stable source accuracy across all iterations. This result suggests that the proposed method can better preserve source domain knowledge during online adaptation, which is mainly attributed to the preserved source-statistics references and the batch-wise re-initialisation strategy that reduces long-term error accumulation.

4.2 | Ablation and Parameter Sensitivity

In this subsection, the key parameters α, K, and w_k involved in the multi-source batch normalisation are analysed. Specifically, α controls the balance between source and target statistics, K specifies the number of representative source domains, and w_k denotes the weight assigned to each selected source domain.

Analysis of α: To validate the effectiveness of the proposed normalisation strategy in Equation (3), experiments are conducted to investigate how the hyperparameter α influences the balance between the statistics of the source and target domain, as illustrated in Figure 5. As α increases, the model gradually shifts towards the target domain and improved accuracy is observed across all datasets. Optimal performance is initially observed at $\alpha = 0.4$, where the best trade-off between source and target-domain statistics is achieved. To further refine this observation, a fine-grained search is performed around the optimal region. Specifically, the values of α in the range of $[0.36, 0.45]$ are evaluated with a step size of 0.01. The results reveal that $\alpha = 0.38$ provides slightly better and more stable performance across CIFAR10-C, CIFAR100-C, and ImageNet-C. Another notable phenomenon is that excessively large values of α cause the model to overfit to the target domain, thus diminishing the influence of source domain knowledge and ultimately degrading performance. These results collectively highlight the critical role of α in regulating the contributions of source and target domain information during adaptation, and thus demonstrate the importance of maintaining a balanced fusion between domain-specific statistics.

Analysis of Cluster K: To investigate the impact of representative source statistics, an ablation study is conducted with varying numbers of clusters K, as shown in Table 4. When only a single statistic ($K = 1$) is used, the performance is clearly limited, indicating insufficient or biased domain knowledge. Increasing K to two leads to significant improvements, demonstrating that combining multiple statistics captures richer and more complementary information. The best results are achieved when $K = 3$, where diversity and relevance are well balanced. However, increasing further K causes performance to decline as redundant or noisy information is introduced. These findings highlight the importance of appropriately selecting the number of representative statistics, with $K = 3$ emerging as the most effective choice.

Number of Source Statistics: To assess the impact of diverse source domain information on performance, four strategies for using source statistics are compared, as shown in Table 5. The BN Adapt method, which does not exploit any source domain knowledge, exhibits the weakest performance. Notably, leveraging a single-source statistic (*SSBN-One source*) yields a substantial improvement over the BN Adapt baseline, demonstrating that the underlying statistical adaptation and

TABLE 1 | Performance comparison on CIFAR10-C (error rate %, lower is better).

Time	t ——————————————————————————→															
Method	**Gaussian (%)**	**Shot (%)**	**Impulse (%)**	**Defocus (%)**	**Glass (%)**	**Motion (%)**	**Zoom (%)**	**Snow (%)**	**Frost (%)**	**Fog (%)**	**Brightness (%)**	**Contrast (%)**	**Elastic (%)**	**Pixelate (%)**	**JPEG (%)**	**Avg. (%)**
Baseline [72]	70.90	59.39	46.12	47.17	42.12	50.62	53.99	56.92	57.36	50.80	42.00	44.74	60.00	50.24	53.33	52.38
BN adapt [67]	35.25	59.79	41.13	30.69	49.49	29.06	43.35	21.67	27.98	43.23	48.65	42.98	40.11	38.25	26.48	38.54
TENT [34]	20.80	16.57	24.51	12.45	28.03	14.55	13.20	18.30	17.51	17.19	12.19	16.13	24.56	21.75	27.25	18.87
CoTTA [35]	20.33	17.35	22.45	9.64	23.46	10.11	8.33	12.91	12.04	10.50	6.58	8.75	14.36	10.13	13.34	12.15
RoTTA [36]	18.91	15.63	21.13	8.85	19.62	10.01	8.42	15.06	10.47	9.43	6.30	9.13	12.24	9.39	11.64	11.08
RMT [37]	17.06	13.79	18.78	10.34	22.32	10.62	10.25	14.97	13.97	13.25	8.26	13.52	17.60	14.70	21.41	14.59
MedBN [74]	14.73	12.18	15.64	10.87	18.26	11.33	10.94	13.42	12.88	12.07	9.76	11.54	14.27	13.06	16.38	12.95
SDA-TTA [73]	12.47	10.16	13.08	7.36	15.94	7.92	7.83	10.74	10.18	9.57	5.96	9.04	12.06	10.77	14.43	10.52
MEMO [48]	10.71	8.58	11.60	6.63	14.85	**6.98**	**6.57**	9.23	8.93	**8.46**	**5.21**	8.23	11.30	9.76	13.67	9.98
SAR [38]	14.32	11.56	14.74	8.85	18.06	10.11	9.78	12.46	11.67	10.94	7.55	10.58	14.06	11.73	16.24	12.18
EATA [39]	13.54	10.82	13.26	8.11	17.18	9.36	9.05	11.64	10.98	10.28	7.08	9.77	13.18	10.94	15.37	11.37
DeYO [40]	12.87	9.94	12.44	7.58	16.21	8.63	8.41	10.76	10.12	9.44	6.71	9.04	12.26	10.18	14.75	10.62
Proposed	**8.63**	**8.32**	**8.71**	**9.15**	**8.27**	8.27	9.17	**8.77**	**8.15**	8.66	8.15	**8.15**	**8.51**	**7.43**	**7.52**	**8.39**

Note: The best results are highlighted in bold, and the second-best results are underlined.

TABLE 2 | Performance comparison on CIFAR100-C (error rate %, lower is better).

Time	t ——————————————————————————→															
Method	**Gaussian (%)**	**Shot (%)**	**Impulse (%)**	**Defocus (%)**	**Glass (%)**	**Motion (%)**	**Zoom (%)**	**Snow (%)**	**Frost (%)**	**Fog (%)**	**Brightness (%)**	**Contrast (%)**	**Elastic (%)**	**Pixelate (%)**	**JPEG (%)**	**Avg. (%)**
Baseline [72]	56.08	58.67	73.85	66.72	45.65	66.52	59.43	56.51	69.40	73.59	72.59	54.89	60.19	66.59	73.03	63.58
BN adapt [67]	49.68	52.61	43.41	42.51	62.60	68.03	53.75	64.51	58.09	48.02	58.09	69.85	54.11	70.12	58.27	54.91
TENT [34]	42.95	37.43	50.43	27.12	59.53	30.35	28.63	39.92	38.33	36.68	25.60	35.55	50.53	45.87	57.96	41.07
CoTTA [35]	58.32	52.32	64.01	27.25	66.73	29.51	23.73	35.03	34.69	30.56	17.35	25.86	44.29	31.25	41.55	38.36
RoTTA [36]	46.63	40.60	52.07	23.56	51.82	25.40	22.51	40.66	27.53	24.98	17.32	25.69	32.31	24.88	<u>31.84</u>	31.91
RMT [37]	29.37	<u>24.77</u>	32.32	<u>18.15</u>	39.10	<u>18.62</u>	<u>17.99</u>	<u>25.79</u>	<u>23.37</u>	<u>21.13</u>	<u>14.58</u>	<u>23.31</u>	32.01	27.04	38.54	25.14
SDA-TTA [73]	<u>23.42</u>	28.91	<u>26.74</u>	25.55	<u>21.48</u>	21.36	20.47	28.02	25.63	26.58	20.11	29.05	<u>27.83</u>	<u>21.98</u>	21.71	24.59
MEMO [48]	27.75	**22.26**	30.04	**16.66**	38.41	**17.56**	**16.50**	**24.93**	**21.58**	**20.37**	**13.21**	**22.36**	29.27	25.30	35.37	<u>24.29</u>
SAR [38]	31.68	27.44	34.16	20.86	41.95	21.84	21.02	29.83	27.04	24.96	16.82	25.58	34.28	28.97	39.84	28.68
EATA [39]	30.42	26.13	33.07	19.92	40.37	20.97	20.15	28.74	26.15	24.03	16.07	24.62	33.11	28.04	38.72	27.63
DeYO [40]	28.96	25.48	31.65	19.08	39.62	19.84	18.93	27.36	24.62	22.48	15.31	23.85	31.94	27.31	37.28	26.65
Proposed	**22.23**	27.62	**25.57**	24.32	**20.18**	20.18	19.26	26.83	24.35	25.35	18.91	27.80	**26.51**	**20.71**	**20.42**	**23.35**

Note: The best results are highlighted in bold, and the second-best results are underlined.

TABLE 3 | Performance comparison on ImageNet-C (error rate %, lower is better).

Time	t ——————————————————————————→															
Method	**Gaussian (%)**	**Shot (%)**	**Impulse (%)**	**Defocus (%)**	**Glass (%)**	**Motion (%)**	**Zoom (%)**	**Snow (%)**	**Frost (%)**	**Fog (%)**	**Brightness (%)**	**Contrast (%)**	**Elastic (%)**	**Pixelate (%)**	**JPEG (%)**	**Avg. (%)**
Baseline [72]	82.41	80.95	80.12	72.84	77.63	71.49	70.88	75.26	73.74	79.15	66.92	72.38	77.91	70.27	70.65	74.84
BN adapt [67]	75.88	74.42	73.85	65.74	70.83	64.58	63.92	68.45	66.93	72.24	59.84	65.21	70.76	62.83	63.19	67.91
TENT [34]	69.42	68.05	67.48	60.38	65.27	59.12	58.47	62.34	60.95	66.31	54.66	59.83	64.57	57.12	57.48	62.10
CoTTA [35]	66.05	64.63	63.87	56.44	61.35	55.08	54.39	58.76	57.27	62.44	50.93	55.84	60.48	53.19	53.58	58.29
RoTTA [36]	63.12	61.74	60.98	53.95	58.82	52.64	51.93	56.87	55.34	60.01	48.92	53.71	57.96	51.12	51.54	55.91
RMT [37]	60.84	59.41	58.73	51.88	57.23	50.76	49.95	54.44	53.22	58.05	46.73	51.84	56.38	49.06	49.48	53.87
SDA-TTA [73]	58.12	56.94	55.87	**46.98**	53.73	46.81	45.36	51.47	49.88	55.18	42.35	48.93	53.21	45.88	46.63	50.49
MEMO [48]	61.24	59.68	58.95	47.32	57.44	**44.38**	**43.92**	54.17	49.12	58.34	41.62	51.55	56.36	48.94	46.71	51.99
SAR [38]	65.38	63.92	63.21	55.84	60.66	53.97	53.18	57.74	56.53	61.45	49.62	54.81	59.73	52.06	52.41	57.37
EATA [39]	63.77	62.35	61.66	54.46	59.21	52.58	51.86	56.38	55.22	60.04	48.43	53.52	58.34	50.73	51.17	55.98
DeYO [40]	60.21	58.74	**52.73**	50.11	56.33	48.25	47.63	53.42	48.92	53.01	44.26	49.15	54.86	47.11	45.60	51.36
Proposed	**54.90**	**53.50**	53.24	47.28	**51.20**	45.05	44.60	**48.50**	**48.15**	**52.05**	**41.10**	**46.20**	**49.95**	**43.50**	**43.98**	**48.21**

Note: The best results are highlighted in bold, and the second-best results are underlined.

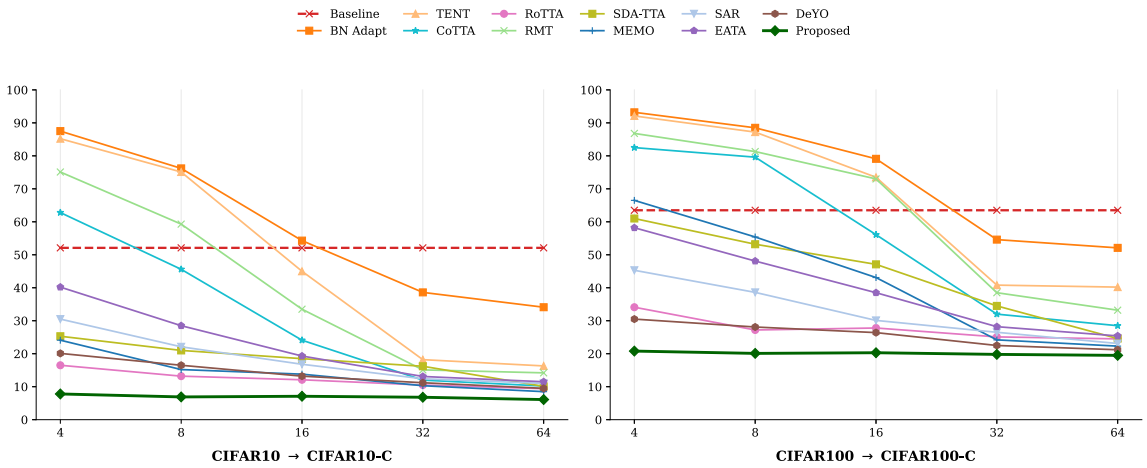


FIGURE 3 | Classification error rates (%) of different TTA methods on CIFAR10 → CIFAR10-C (left) and CIFAR100 → CIFAR100-C (right) across varying test batch sizes.

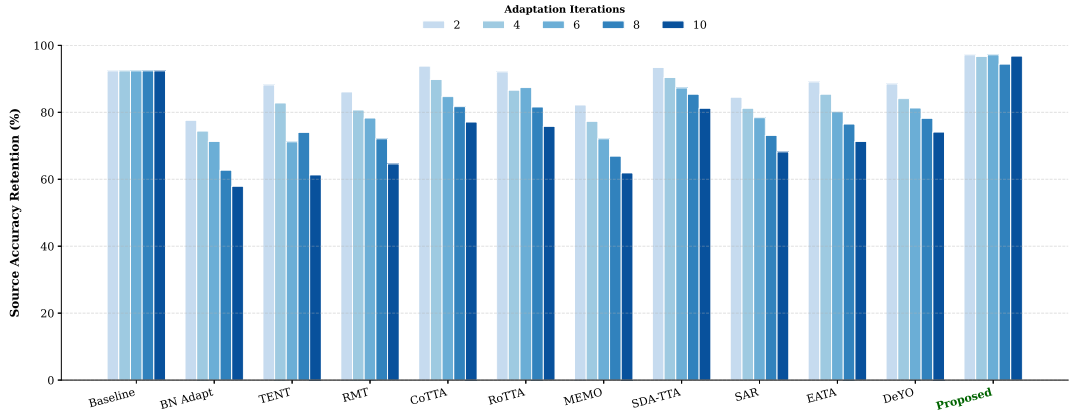


FIGURE 4 | Source accuracy retention in CIFAR10 after different adaptation iterations (2, 4, 6, 8, 10) on various TTA methods, used to evaluate catastrophic forgetting.

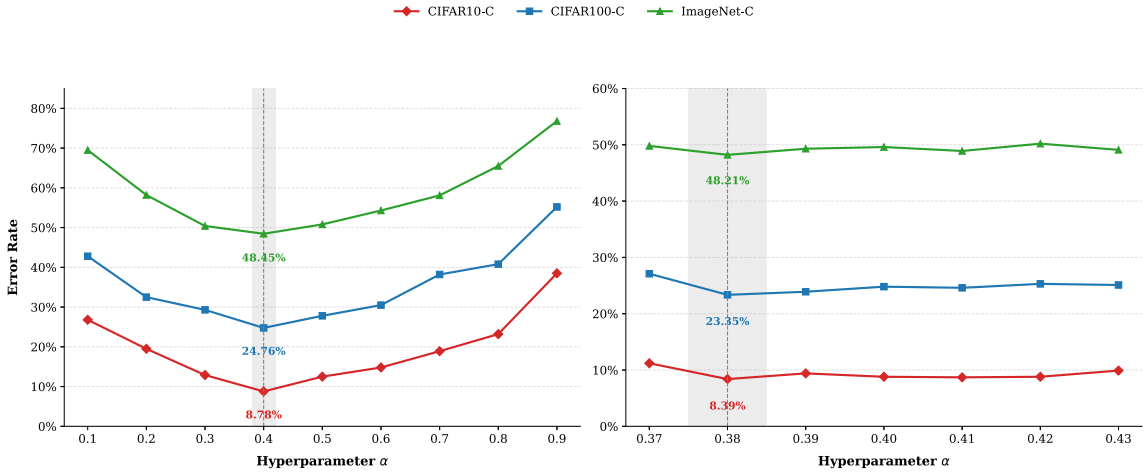


FIGURE 5 | Performance of the proposed method across different values of α on the CIFAR10-C, CIFAR100-C, and ImageNet-C datasets. The left plot shows the coarse-grained analysis with α ranging from 0.1 to 0.9, where $\alpha = 0.4$ achieves the optimal balance. The right plot presents the fine-grained analysis with α values between 0.36 and 0.45, highlighting the slightly improved performance at $\alpha = 0.38$. The red circles indicate the optimal α values for each scenario.

re-initialisation mechanisms remain highly effective even under strictly constrained single-source conditions. Furthermore, extending this to the multi-source approach (*MSBN-M source*) by combining statistics from all sources provides even greater performance gains. The proposed representative source selection strategy (*MSBN-K source*) achieves the best

results by selectively integrating only the most informative statistics and avoiding noise from irrelevant sources. These results demonstrate that targeted use of diverse, yet relevant source domain information is essential for maximising adaptation performance.

Analysis of w_k: Four weight allocation strategies for test-time adaptation are compared. The first strategy assigns equal weights to all source domains, that is, $w_k = 1/K$, serving as a naive baseline. The second strategy determines weights using cosine similarity, where weights are computed based on the similarity between the target batch's feature statistics (μ_t, σ_t) and the source domains statistics. The third strategy assigns weights based on cross-entropy between predictions from source domain models and the target batch. In the fourth strategy, spectral entropy-based weighting is applied, leveraging the inherent spectral characteristics of the data to better capture the underlying structure and adapt more effectively to domain shifts. The results, presented in Table 6, show that the equal-weight baseline performs worst due to its inability to account for domain relevance, while similarity and entropy-based schemes progressively improve performance. Notably, the proposed spectral entropy weighting achieves the lowest error rates across CIFAR10-C, CIFAR100-C, and ImageNet-C, reducing error rates by a large margin compared to both the equal-weight baseline and other weighting heuristics.

Effect of Source-Target Family Overlap: To further examine whether the proposed method depends on explicit overlap between the source-statistics pool and the target corruption family, an additional leave-one-family-out evaluation is conducted on CIFAR10-C and CIFAR100-C as shown in Table 7. The corruption types are grouped into four families: Noise, Blur, Weather, and Digital. For each target family, all source domain variants belonging to the same family are excluded from the offline source-statistics pool, and adaptation is performed using only the remaining source families. In this way, the representative source statistics used during test-time adaptation do not include source variants from the same corruption family as the target. The results show that removing the matched corruption family leads to a moderate performance drop compared with the full source pool, indicating that direct family overlap is beneficial. Nevertheless, the proposed method remains consistently stronger than the corresponding single-source and naive multi-source alternatives under reduced source-target overlap on both CIFAR10-C and CIFAR100-C. This suggests that the proposed representative-source selection and spectral entropy guided weighting mechanism do not

TABLE 4 | Results for the number of cluster K on CIFAR10-C, CIFAR100-C, and ImageNet-C (error rate %).

Number of cluster K.	CIFAR10-C (%)	CIFAR100-C (%)	ImageNet-C (%)
$K = 1$	22.14	44.91	69.84
$K = 2$	13.45	30.36	56.42
$K = 3$	**8.39**	**23.35**	**48.21**
$K = 5$	17.61	34.69	60.38
$K = 8$	16.18	32.87	57.16
$K = 11$	21.16	41.69	67.83
$K = 14$	15.21	31.14	55.92
$K = 15$	15.45	36.18	56.37

Note: The best results are highlighted in bold.

TABLE 5 | Results for the number of source statistics on CIFAR10-C, CIFAR100-C, and ImageNet-C (error rate %).

Number of source stats.	CIFAR10-C (%)	CIFAR100-C (%)	ImageNet-C (%)
BN adapt [67]	38.54	54.91	67.91
SSBN-one source	29.45	48.36	61.84
MSBN-M source	15.45	36.18	53.72
MSBN-K source	**8.39**	**23.35**	**48.21**

Note: The best results are highlighted in bold.

TABLE 6 | The weight strategy of K source domain statistics (error rate %).

Methods	CIFAR10-C (%)	CIFAR100-C (%)	ImageNet-C (%)
$w_k = 1/K$	29.64	41.71	61.13
$w_k \in \mathbb{R}^+$ with cosine similarity	26.95	47.69	63.14
$w_k \in \mathbb{R}^+$ with cross entropy	16.35	36.19	54.27
$w_k \in \mathbb{R}^+$ with spectral entropy	**8.39**	**23.35**	**48.21**

Note: The best results are highlighted in bold.

TABLE 7 | Leave-one-family-out evaluation on CIFAR10-C and CIFAR100-C.

	CIFAR10-C				CIFAR100-C			
Method	**Noise**	**Blur**	**Weather**	**Digital**	**Noise**	**Blur**	**Weather**	**Digital**
SSBN-one source	29.18	30.42	28.73	29.47	50.62	47.83	46.71	48.28
MSBN-M source	28.91	30.27	27.84	29.62	43.25	40.87	39.96	42.76
Proposed (leave-one-family-out)	12.84	11.93	10.64	11.21	28.37	26.91	25.74	27.18
Proposed (full source pool)	9.12	8.64	7.81	7.99	25.04	23.48	22.22	22.66

Note: For each target corruption family, source domain BN statistics from the same family are excluded from the source pool. Error rate (%, lower is better).

merely exploit exact corruption-family matching, but can still benefit from related source domain statistics with partial distributional mismatch.

Runtime Overhead Evaluation: As shown in Table 8, the proposed method introduces only moderate runtime overhead across CIFAR10-C, CIFAR100-C, and ImageNet-C. Compared with BN Adapt, the proposed method requires additional computation for spectral-entropy estimation, source-statistics weighting, and one-step BN affine-parameter refinement. However, the overall latency remains close to TENT and RoTTA, and is clearly lower than methods involving heavier online optimization or augmentation-based inference, such as CoTTA, RMT, and MEMO. This is because the proposed method updates only BN affine parameters and does not perform full-network adaptation. These results indicate that the proposed method maintains practical efficiency for online batch adaptation while achieving stronger robustness.

TABLE 8 | Runtime overhead comparison on three benchmarks with batch size fixed to 32.

Method	**CIFAR10-C ms/batch ↓**	**CIFAR100-C ms/batch ↓**	**ImageNet-C ms/batch ↓**
BN adapt [67]	4.1	5.8	26.4
TENT [34]	7.6	10.9	38.7
CoTTA [35]	15.8	21.5	72.3
RoTTA [36]	9.4	12.6	43.5
RMT [37]	13.7	19.2	67.8
MEMO [48]	29.5	41.8	134.6
Proposed	8.9	12.1	42.7

Note: Latency denotes the average wall-clock time for processing one incoming target batch, including forward inference and online adaptation. Lower is better.

4.3 | Evaluation on Cross-Domain Benchmarks Under the TTBA Protocol

For cross-domain evaluation under the TTBA protocol, PACS [75] and ImageNet-R [76] are adopted.

Experimental Setup: On PACS, a leave-one-domain-out setting is used, where one domain serves as the unlabelled target stream and the remaining three domains are treated as source domains, with source domain BN statistics computed and stored separately in an offline manner. On ImageNet-R, the source data are taken from the original ImageNet training set, while ImageNet-R is treated as the unlabelled target stream; since ImageNet-R does not naturally provide multiple source domains, the ImageNet source data are partitioned offline into multiple source subdomains by spectral clustering, and BN statistics are computed separately for each subdomain. In both cases, target samples are sequentially processed in mini-batches without using target labels. Classification accuracy is reported, and only a small set of representative reference methods, including Source Only, BN Adapt, TENT, and MEMO, is considered, since the goal of this experiment is to verify whether the proposed multi-source BN adaptation method remains effective under broader cross-domain shifts rather than to provide a large-scale benchmark comparison.

As shown in Table 9, the proposed method also achieves consistent improvements on PACS and ImageNet-R under the TTBA protocol. Compared with Source Only, BN Adapt, TENT, and MEMO, the proposed method obtains the best average accuracy on PACS and the highest accuracy on ImageNet-R. On PACS, the gain is observed across all four target domains, indicating that the proposed multi-source BN adaptation method can effectively exploit natural source domain diversity when multiple source domains are directly available. On ImageNet-R, the proposed method also shows clear advantages after constructing multiple source subdomains from ImageNet through offline spectral clustering. These results suggest that the proposed method is not restricted to corruption-style distribution shifts, and can also provide useful adaptation benefits under broader cross-domain style and rendition changes.

4.4 | Exploratory Evaluation Under Open-Set TTBA

To further evaluate the robustness of the proposed method under semantic distribution shifts, additional experiments are conducted under an open-set test-time adaptation (OS-TTA) setting. The target stream contains both covariate-shifted known-class samples and semantic out-of-distribution samples.

Open-Set Construction: Specifically, CIFAR10-C and CIFAR100-C are used as the known-class target data under corruption shifts, while SVHN [77] is adopted as the unknown-class source following common open-set evaluation settings. During test time, corrupted CIFAR samples and SVHN samples are mixed to form the unlabelled target stream. No labels are available during adaptation.

Compared Methods and Evaluation Metric: The proposed method is compared with representative entropy-based and open-set TTA methods, including Source Only, TENT [34], TENT + OSTTA [78], and TENT + UniEnt [41]. Following common OS-TTA protocols, each test mini-batch contains an

TABLE 9 | Results on cross-domain benchmarks under the TTBA protocol (Accuracy %, higher is better).

	PACS					
Method	**Art**	**Cartoon**	**Photo**	**Sketch**	**Avg.**	**ImageNet-R**
Source only	80.94	78.87	94.96	73.55	82.08	36.94
BN adapt [67]	81.26	79.18	95.08	73.92	82.36	37.42
TENT [34]	81.73	79.65	95.32	74.28	82.75	39.17
MEMO [48]	82.18	80.31	95.44	75.06	83.25	39.21
Proposed	**83.14**	**81.02**	**95.88**	**76.11**	**84.04**	**42.56**

Note: The best results are highlighted in bold.

TABLE 10 | Evaluation under open-set test-time adaptation (error rate %, lower is better).

Methods	**CIFAR10-C (%)**	**CIFAR100-C (%)**
Source only	18.27	46.75
TENT [34]	23.84	52.16
TENT + OSTTA [78]	<u>17.65</u>	<u>43.28</u>
TENT + UniEnt [41]	**15.42**	**39.74**
Proposed	18.03	44.85

Note: The best results are highlighted in bold, and the second-best results are underlined.

equal number of corrupted CIFAR samples and SVHN unknown samples. During evaluation, only the error rate (%) on known-class CIFAR samples is reported.

Evaluation under Open-Set Test-Time Adaptation: As shown in Table 10, conventional entropy-minimisation based methods such as TENT suffer from clear performance degradation when semantic out-of-distribution samples are introduced into the target stream. Open-set aware methods including TENT + OSTTA and TENT + UniEnt achieve improved robustness by suppressing unreliable adaptation updates caused by unknown samples. Although the proposed method is not specifically designed for open-set adaptation, competitive performance is still achieved on both CIFAR10-C and CIFAR100-C, outperforming the standard TENT baseline and remaining relatively stable under mixed semantic and covariate shifts. These results suggest that the proposed multi-source BN adaptation strategy can alleviate part of the instability introduced by open-set target distributions.

5 | Conclusion

A multi-source normalisation TTBA method is proposed to address the limitations of existing TTBA methods. The method mitigates the fragility of one-to-one alignment by using a spectral similarity clustering method to select representative source domains, a spectral entropy guided weighting method to fuse source statistics in proportion to relevance, and a batch-wise affine re-initialisation method with unsupervised refinement to stabilise updates. The method reduces dependence on target samples, remains stable under small or unstable batches, and is robust to diverse domain conditions such as noise, background clutter, and illumination or temperature shifts. Experiments on multiple benchmarks verify the method, showing consistent gains over state-of-the-art TTBA methods, with pronounced improvements in the small-batch regime.

The current empirical study is mainly focused on image classification under closed-set test-time batch adaptation settings. Although additional evaluations on cross-domain transfer and open-set test-time adaptation have been included, a substantial part of the experiments still relies on benchmark-specific domain structures and synthetic distribution shifts. Broader validation under large-scale real-world distribution changes, more challenging open-set scenarios, and other vision tasks such as semantic segmentation and object detection would further strengthen the generality of the proposed method.

Funding

This work was supported in part by the National Natural Science Foundation of China under Grant 62271484, and in part by the Fujian Provincial Natural Science Foundation of China under Grant 2026J010018.

Conflicts of Interest

Xiangtao Zheng is an Associate Editor for the journal, and was not involved in peer review process or the decision to publish this article. The authors declare that they have no conflict of interest.

Data Availability Statement

The dataset used for the research is available publicly, and the request can be made to authors for further assistance.

References

1. J. Wang, M. M. Cheng, and J. Jiang, "Domain Shift Preservation for Zero-Shot Domain Adaptation," *IEEE Transactions on Image Processing* 30 (2021): 5505–5517, https://doi.org/10.1109/tip.2021.3084354.

2. Z. Wu, G. Luo, Z. Yang, Y. Guo, K. Li, and Y. Xue, "A Comprehensive Review on Deep Learning Approaches in Wind Forecasting Applications," *CAAI Transactions on Intelligence Technology* 7, no. 2 (2022): 129–143, https://doi.org/10.1049/cit2.12076.

3. X. Huang and S. H. Choi, "Learning Feature Alignment and Dual Correlation for Few-Shot Image Classification," *CAAI Transactions on Intelligence Technology* 9, no. 2 (2024): 303–318, https://doi.org/10.1049/cit2.12273.

4. W. Wang, Z. Shen, D. Li, P. Zhong, and Y. Chen, "Probability-Based Graph Embedding Cross-Domain and Class Discriminative Feature Learning for Domain Adaptation," *IEEE Transactions on Image Processing* 32 (2023): 72–87, https://doi.org/10.1109/tip.2022.3226405.

5. T. Shermin, G. Lu, S. W. Teng, M. Murshed, and F. Sohel, "Adversarial Network With Multiple Classifiers for Open Set Domain Adaptation," *IEEE Transactions on Multimedia* 23 (2021): 2732–2744, https://doi.org/10.1109/tmm.2020.3016126.

6. J. Sun, Z. Wang, W. Wang, H. Li, and F. Sun, "Domain Adaptation With Geometrical Preservation and Distribution Alignment," *Neurocomputing* 454 (2021): 152–167, https://doi.org/10.1016/j.neucom.2021.04.098.

7. M. M. Eltoukhy, F. S. Alsubaei, M. M. Abdel-Aziz, and K. M. Hosny, "RDHNet: Reversible Data Hiding Method for Securing Colour Images Using AlexNet and Watershed Transform in a Fusion Domain," *CAAI Transactions on Intelligence Technology* 10, no. 5 (2025): 1422–1445, https://doi.org/10.1049/cit2.70038.

8. D. Tuia, J. Munoz-Mari, and G. Camps-Valls, "Kernel Manifold Alignment for Domain Adaptation," *PLoS One* 11, no. 2 (2016): 0148655, https://doi.org/10.1371/journal.pone.0148655.

9. S. Tang, W. Su, Y. Yang, L. Chen, and M. Ye, "Model Adaptation via Credible Local Context Representation," *CAAI Transactions on Intelligence Technology* 10, no. 3 (2025): 638–651, https://doi.org/10.1049/cit2.12228.

10. N. Alqahtani, A. A. Khan, R. K. Mahendran, and M. Faheem, "VSMI2-PANet: Versatile Scale-Malleable Image Integration and Patch Wise Attention Network With Transformer for Lung Tumour Segmentation Using Multi-Modal Imaging Techniques," *CAAI Transactions on Intelligence Technology* 10, no. 5 (2025): 1376–1393, https://doi.org/10.1049/cit2.70039.

11. Z. Niu, J. Yuan, X. Ma, et al., "Knowledge Distillation-Based Domain-Invariant Representation Learning for Domain Generalization," *IEEE Transactions on Multimedia* 26 (2024): 245–255, https://doi.org/10.1109/tmm.2023.3263549.

12. Y. Chen, Q. Wang, J. Yang, B. Chen, H. Xiong, and S. Du, "CSCC: Cross-Scene Crowd Counting via Learning to Diversify for Domain Generalization," *IEEE Transactions on Multimedia* 27 (2025): 3320–3330, https://doi.org/10.1109/tmm.2025.3535302.

13. T. S. Haque, A. Chakraborty, S. P. Mondal, and S. Alam, "Approach to Solve Multi-Criteria Group Decision-Making Problems by Exponential Operational Law in Generalised Spherical Fuzzy Environment," *CAAI Transactions on Intelligence Technology* 5, no. 2 (2020): 106–114, https://doi.org/10.1049/trit.2019.0078.

14. A. Gasparin, S. Lukovic, and C. Alippi, "Deep Learning for Time Series Forecasting: The Electric Load Case," *CAAI Transactions on Intelligence Technology* 7, no. 1 (2022): 1–25, https://doi.org/10.1049/cit2.12060.

15. M. Ghifary, D. Balduzzi, W. B. Kleijn, and M. Zhang, "Scatter Component Analysis: A Unified Framework for Domain Adaptation and Domain Generalization," *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39, no. 7 (2017): 1414–1430, https://doi.org/10.1109/tpami.2016.2599532.

16. Y. Su, X. Xu, T. Li, and K. Jia, "Revisiting Realistic Test-Time Training: Sequential Inference and Adaptation by Anchored Clustering Regularized Self-Training," *IEEE Transactions on Pattern Analysis and Machine Intelligence* 46, no. 8 (2024): 5524–5540, https://doi.org/10.1109/tpami.2024.3370963.

17. J. Liang, D. Hu, and J. Feng, "Do We Really Need to Access the Source Data? Source Hypothesis Transfer for Unsupervised Domain Adaptation," in *International Conference on Machine Learning* (2020), 6028–6039, https://doi.org/10.48550/arXiv.2002.08546.

18. J. MaImproved Self-Training for Test-Time Adaptation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2024), 23701–23710, https://doi.org/10.1109/CVPR52733.2024.02237.

19. M. Tan, G. Chen, J. Wu, et al., "Uncertainty-Calibrated Test-Time Model Adaptation Without Forgetting," *IEEE Transactions on Pattern Analysis and Machine Intelligence* 47, no. 8 (2025): 6274–6289, https://doi.org/10.1109/tpami.2025.3560696.

20. Y. Tang, S. Chen, Z. Kan, Y. Zhang, Q. Guo, and Z. He, "Learning Visual Conditioning Tokens to Correct Domain Shift for Fully Test-time Adaptation," *IEEE Transactions on Multimedia* (2024): 1–12, https://doi.org/10.1109/tmm.2024.3443633.

21. C. Mu, J. Peng, and Y. Tang, "Learning-Based Control for Discrete-Time Constrained Nonzero-Sum Games," *CAAI Transactions on Intelligence Technology* 6, no. 2 (2021): 203–213, https://doi.org/10.1049/cit2.12015.

22. S. M. Ayyoubzadeh, W. Liu, I. Kezele, et al., "Test-Time Adaptation for Optical Flow Estimation Using Motion Vectors," *IEEE Transactions on Image Processing* 32 (2023): 4977–4988, https://doi.org/10.1109/tip.2023.3309108.

23. Z. Wen, S. Niu, G. Li, Q. Wu, M. Tan, and Q. Wu, "Test-Time Model Adaptation for Visual Question Answering With Debiased Self-Supervisions," *IEEE Transactions on Multimedia* 26 (2024): 2137–2147, https://doi.org/10.1109/tmm.2023.3292597.

24. Y. Yu, S. Shin, S. Back, M. Ko, S. Noh, and K. Lee, "Domain-Specific Block Selection and Paired-View Pseudo-Labeling for Online Test-Time Adaptation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2024), 22723–22732.

25. Z. Nado, S. Padhy, D. Sculley, A. D'Amour, B. Lakshminarayanan, and J. Snoek, "Evaluating Prediction-Time Batch Normalization for Robustness Under Covariate Shift," in *Proceedings of ICML Workshops* (2020), https://arxiv.org/abs/2006.10963.

26. Y. Li, N. Wang, J. Shi, X. Hou, and J. Liu, "Adaptive Batch Normalization for Practical Domain Adaptation," *Pattern Recognition* 80 (2018): 109–117, https://doi.org/10.1016/j.patcog.2018.03.005.

27. H. Nam and H. E. Kim, "Batch-Instance Normalization for Adaptively Style-Invariant Neural Networks," in *Advances in Neural Information Processing Systems (NeurIPS)*, ed. S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Vol. 31 (2018), https://arxiv.org/abs/1805.07925.

28. N. Karani, E. Erdil, K. Chaitanya, and E. Konukoglu, "Test-Time Adaptable Neural Networks for Robust Medical Image Segmentation," *Medical Image Analysis* 68 (2021): 101907, https://doi.org/10.1016/j.media.2020.101907.

29. H. Li, H. Liu, D. Hu, et al., "Self-Supervised Test-Time Adaptation for Medical Image Segmentation," in *Machine Learning in Clinical Neuroimaging* (2022), 32–41, https://doi.org/10.1007/978-3-031-17899-3_4.

30. M. Shu, W. Nie, D. A. Huang, et al., "Test-Time Prompt Tuning for Zero-Shot Generalization in Vision-Language Models," *Advances in Neural Information Processing Systems* 35 (2022): 14274–14289, https://arxiv.org/abs/2209.07511.

31. X. Lu, T. Gong, and X. Zheng, "Multisource Compensation Network for Remote Sensing Cross-Domain Scene Classification," *IEEE Transactions on Geoscience and Remote Sensing* 58, no. 4 (2020): 2504–2515, https://doi.org/10.1109/tgrs.2019.2951779.

32. L. Li, J. Gou, W. Ou, W. Chen, and L. Du, "Contrastive Learning-Based Multi-Level Knowledge Distillation," *CAAI Transactions on Intelligence Technology* 10, no. 5 (2025): 1478–1488, https://doi.org/10.1049/cit2.70036.

33. C. Hua, X. Cao, J. Li, and S. Li, "Fixed-Time Zeroing Neural Dynamics for Adaptive Coordination of Multi-Agent Systems," *CAAI*

Transactions on Intelligence Technology 11, no. 1 (2026): 267–278, https://doi.org/10.1049/cit2.70089.

34. D. Wang, E. Shelhamer, S. Liu, B. Olshausen, and T. Darrell, "Tent: Fully Test-Time Adaptation by Entropy Minimization," in *International Conference on Learning Representations* (2021), https://doi.org/10.48550/arXiv.2006.10726.

35. Q. Wang, O. Fink, L. Van Gool, and D. Dai, "Continual Test-Time Domain Adaptation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2022), 7191–7201, https://doi.org/10.1109/CVPR52688.2022.00706.

36. L. Yuan, B. Xie, and S. Li, "Robust Test-Time Adaptation in Dynamic Scenarios," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2023), 15922–15932, https://doi.org/10.1109/CVPR52729.2023.01528.

37. M. Döbler, R. A. Marsden, and B. Yang, "Robust Mean Teacher for Continual and Gradual Test-Time Adaptation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2023), 7704–7714, https://doi.org/10.1109/CVPR52729.2023.00744.

38. S. Niu, J. Wu, Y. Zhang, et al., "Towards Stable Test-Time Adaptation in Dynamic Wild World," in *International Conference on Learning Representations* (2023), https://doi.org/10.48550/arXiv.2302.12400.

39. S. Niu, J. Wu, Y. Zhang, et al., "Efficient Test-Time Model Adaptation Without Forgetting," in *Machine Learning in Clinical Neuroimaging*, Vol. 162 (2022), 16888–16905, https://doi.org/10.48550/arXiv.2204.02610.

40. J. Lee, D. Jung, S. Lee, et al., "Entropy Is Not Enough for Test-Time Adaptation: From the Perspective of Disentangled Factors," in *International Conference on Learning Representations* (2024), https://doi.org/10.48550/arXiv.2403.07366.

41. Z. Gao, X. Y. Zhang, and C. L. Liu, "Unified Entropy Optimization for Open-Set Test-Time Adaptation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2024), 23975–23984, https://doi.org/10.1109/CVPR52733.2024.02263.

42. C. K. Mummadi, R. Hutmacher, K. Rambach, E. Levinkov, T. Brox, and J. H. Metzen, "Test-Time Adaptation to Distribution Shift by Confidence Maximization and Input Transformation," *arXiv preprint arXiv: 2106.14999* (2021), https://doi.org/10.48550/arXiv.2106.14999.

43. X. Zhao, C. Liu, A. Sicilia, S. J. Hwang, and Y. Fu, "Test-Time Fourier Style Calibration for Domain Generalization," in *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence* (2022), 1721–1727, https://doi.org/10.24963/ijcai.2022/240.

44. Y. Y. Tsai, C. Mao, and J. Yang, "Convolutional Visual Prompt for Robust Visual Perception," in *Advances in Neural Information Processing Systems*, Vol. 36 (2023), https://doi.org/10.5555/3666122.3667333.

45. J. A. Termöhlen, M. Klingner, L. J. Brettin, N. M. Schmidt, and T. Fingscheidt, "Continual Unsupervised Domain Adaptation for Semantic Segmentation by Online Frequency Domain Style Transfer," in *IEEE International Intelligent Transportation Systems Conference* (2021), 2881–2888, https://doi.org/10.1109/ITSC48978.2021.9564566.

46. M. Alfarra, J. C. Pérez, A. Thabet, A. Bdi, P. H. S. Torr, and B. Ghanem, "Combating Adversaries With Anti-Adversaries," in *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36, no. 6 (2022), 5992–6000, https://doi.org/10.1609/aaai.v36i6.20545.

47. Y. Gan, X. Ma, Y. Lou, et al., "Decorate the Newcomers: Visual Domain Prompt for Continual Test Time Adaptation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37, no. 6 (2023), 7595–7603, https://doi.org/10.1609/aaai.v37i6.25922.

48. M. Zhang, S. Levine, and C. Finn, "MEMO: Test Time Robustness via Adaptation and Augmentation," in *Advances in Neural Information Processing Systems*, Vol. 35 (2022), 38629–38642, https://doi.org/10.5555/3600270.3603069.

49. Y. Lu, H. Huang, X. Hu, Z. Lai, and X. Li, "Multiple Adaptation Network for Multi-Source and Multi-Target Domain Adaptation," *IEEE Transactions on Multimedia* 27 (2025): 1–15, https://doi.org/10.1109/tmm.2025.3543094.

50. Y. Zuo, H. Yao, and C. Xu, "Attention-Based Multi-Source Domain Adaptation," *IEEE Transactions on Image Processing* 30 (2021): 3793–3803, https://doi.org/10.1109/tip.2021.3065254.

51. G. Kang, L. Jiang, Y. Wei, Y. Yang, and A. Hauptmann, "Contrastive Adaptation Network for Single- and Multi-Source Domain Adaptation," *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, no. 4 (2022): 1793–1804, https://doi.org/10.1109/tpami.2020.3029948.

52. F. Qiao, L. Zhao, and X. Peng, "Learning to Learn Single Domain Generalization," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2020), 12553–12562, https://doi.org/10.1109/CVPR42600.2020.01257.

53. Y. Ganin, E. Ustinova, H. Ajakan, et al., "Domain-Adversarial Training of Neural Networks," *Journal of Machine Learning Research* 17, no. 59 (2016): 1–35, https://doi.org/10.5555/2946645.2946704.

54. M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Conditional Adversarial Domain Adaptation," in *Advances in Neural Information Processing Systems*, Vol. 31 (2018), 1647–1657, https://doi.org/10.5555/3326943.3327094.

55. A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A Kernel Two-Sample Test," *Journal of Machine Learning Research* 13, no. 1 (2012): 723–773, https://doi.org/10.5555/2188385.2188410.

56. B. Sun and K. Saenko, "Deep Coral: Correlation Alignment for Deep Domain Adaptation," in *Computer Vision – ECCV 2016 Workshops* (Springer, 2016), 443–450, https://doi.org/10.1007/978-3-319-49409-8_35.

57. N. Courty, R. Flamary, A. Habrard, and A. Rakotomamonjy, "Joint Distribution Optimal Transportation for Domain Adaptation," in *Advances in Neural Information Processing Systems*, Vol. 30 (2017), 3733–3742, https://doi.org/10.5555/3294996.3295130.

58. S. Lü, M. Kang, and X. Li, "Alleviating lmbalanced Pseudo-Label Distribution: Self-Supervised Multi-Source Domain Adaptation With Label-Specific Confidence," in *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence* (2024), 4669–4677, https://doi.org/10.24963/ijcai.2024/516.

59. Y. Balaji, S. Sankaranarayanan, and R. Chellappa, "MetaReg: Towards Domain Generalization Using Meta-Regularization," in *Advances in Neural Information Processing Systems*, Vol. 31 (2018), 998–1008, https://doi.org/10.5555/3326943.3327036.

60. H. Li, S. J. Pan, S. Wang, and A. C. Kot, "Domain Generalization With Adversarial Feature Learning," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2018), 5400–5409, https://doi.org/10.1109/CVPR.2018.00566.

61. S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised Representation Learning by Predicting Image Rotations," in *International Conference on Learning Representations* (2018), https://doi.org/10.48550/arXiv.1803.07728.

62. H. Zhang and YNDDLP, "Mixup: Beyond Empirical Risk Minimization," in *International Conference on Learning Representations* (2018), https://doi.org/10.48550/arXiv.1710.09412.

63. K. Zhou, Y. Yang, Y. Qiao, and T. Xiang, "Mixstyle Neural Networks for Domain Generalization and Adaptation," *International Journal of Computer Vision* 132, no. 3 (2024): 822–836, https://doi.org/10.1007/s11263-023-01913-8.

64. Q. Xu, R. Zhang, Y. Zhang, Y. Wang, Q. Tian, "A Fourier-Based Framework for Domain Generalization," in *IEEE/CVF Conference on*

Computer Vision and Pattern Recognition (2021), 14378–14387, https://doi.org/10.1109/CVPR46437.2021.01415.

65. D. Cheng, Z. Xu, X. Jiang, N. Wang, D. Li, and X. Gao, "Disentangled Prompt Representation for Domain Generalization," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2024), 23595–23604, https://doi.org/10.1109/CVPR52733.2024.02227.

66. M. Aminbeidokhti, F. A. G. Peña, H. R. Medeiros, T. Dubail, E. Granger, and M. Pedersoli, "Domain Generalization by Rejecting Extreme Augmentations," in *IEEE/CVF Winter Conference on Applications of Computer Vision* (2024), 2204–2214, https://doi.org/10.1109/WACV57701.2024.00221.

67. S. Schneider, E. Rusak, L. Eck, O. Bringmann, W. Brendel, and M. Bethge, "Improving Robustness Against Common Corruptions by Covariate Shift Adaptation," in *Advances in Neural Information Processing Systems*, Vol. 33 (2020), 11539–11551, https://doi.org/10.5555/3495724.3496692.

68. J. Zhang, L. Qi, Y. Shi, and Y. Gao, "DomainAdaptor: A Novel Approach to Test-Time Adaptation," in *IEEE/CVF International Conference on Computer Vision* (2023), 18925–18935, https://doi.org/10.1109/ICCV51070.2023.01739.

69. D. Hendrycks and T. Dietterich, "Benchmarking Neural Network Robustness to Common Corruptions and Perturbations," in *International Conference on Learning Representations* (2019), https://doi.org/10.48550/arXiv.1903.12261.

70. K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2016), 770–778, https://doi.org/10.1109/CVPR.2016.90.

71. G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2017), 2261–2269, https://doi.org/10.1109/CVPR.2017.243.

72. I. Gulrajani and D. Lopez-Paz, "In Search of Lost Domain Generalization," in *International Conference on Learning Representations* (2021), https://doi.org/10.48550/arXiv.2007.01434.

73. J. Guo, J. Zhao, C. Du, et al., "Everything to the Synthetic: Diffusion-Driven Test-Time Adaptation via Synthetic-Domain Alignment," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2025), 30503–30513, https://doi.org/10.1109/CVPR52734.2025.02840.

74. H. Park, J. Hwang, S. Mun, S. Park, and J. Ok, "MedBN: Robust Test-Time Adaptation Against Malicious Test Samples," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2024), 5997–6007, https://doi.org/10.1109/CVPR52733.2024.00573.

75. D. Li, Y. Yang, Y. Z. Song, and T. M Hospedales, "Deeper, Broader and Artier Domain Generalization," in *IEEE International Conference on Computer Vision* (2017), 5542–5550, https://doi.org/10.1109/ICCV.2017.591.

76. D. Hendrycks, S. Basart, N. Mu, et al., "The Many Faces of Robustness: A Critical Analysis of Out-of-Distribution Generalization," in *IEEE/CVF International Conference on Computer Vision* (2021), 8340–8349, https://doi.org/10.1109/ICCV48922.2021.00823.

77. Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading Digits in Natural Images With Unsupervised Feature Learning," in *NIPS Workshop on Deep Learning and Unsupervised Feature Learning* (2011), https://research.google/pubs/reading-digits-in-natural-images-with-unsupervised-feature-learning.

78. Y. Yang, C. Chen, and H. Huang, "Back to Source: Open-Set Continual Test-Time Adaptation via Domain Compensation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2026), 7957–7966, https://doi.org/10.48550/arXiv.2604.21772.