MedBN: Robust Test-Time Adaptation against Malicious Test Samples

Hyejin Park* Jeongyeon Hwang* Sunung Mun Sangdon Park Jungseul Ok†
Pohang University of Science and Technology (POSTECH), South Korea
{parkebbi2, jeongyeon.hwang, mtablo, sangdon, jungseul}@postech.ac.kr

Abstract

Test-time adaptation (TTA) has emerged as a promising solution to address performance decay due to unforeseen distribution shifts between training and test data. While recent TTA methods excel in adapting to test data variations, such adaptability exposes a model to vulnerability against malicious examples. Indeed, previous studies have uncovered security vulnerabilities within TTA even when a small proportion of the test batch is maliciously manipulated. In response to the emerging threat, we propose median batch normalization (MedBN), leveraging the robustness of the median for statistics estimation within the batch normalization layer during test-time inference. Our method is algorithm-agnostic, thus allowing seamless integration with existing TTA frameworks. Our experimental results on benchmark datasets, including CIFAR10-C, CIFAR100-C, and ImageNet-C, consistently demonstrate that MedBN outperforms existing approaches in maintaining robust performance across different attack scenarios, encompassing both instant and cumulative attacks. Through extensive experiments, we show that our approach sustains the performance even in the absence of attacks, achieving a practical balance between robustness and performance. Our code is available at https://github.com/ml-postech/MedBN-robust-test-time-adaptation.

1. Introduction

Deep neural networks (DNNs) have shown noticeable advances in benchmarks across diverse recognition tasks, assuming virtually no distribution shift between training and test data. However, distribution shifts are inevitable in practice mainly due to time-varying environments (e.g., lighting variations and changing weather conditions), and severely degenerate the model performance [25, 31]. It is infeasible to forecast and prepare for every potential test domain in advance. In response, test-time adaptation (TTA) has been extensively studied [6, 14, 20, 21, 40, 51, 53], where TTA aims at adapting a pre-trained model to test data, which is unlabeled and from latent domain, in an online manner.

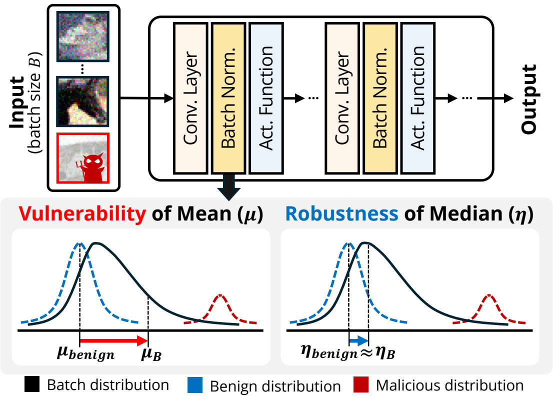


Figure 1. An illustrative example of the vulnerability of mean in a batch normalization layer to manipulation by malicious sample (left), contrasted with the robustness of median such manipulation (right), when dealing with malicious samples within the batch.

The major advantage of TTA stems from leveraging the statistics of the test batch. A prominent technique is to use test batch statistics in each batch normalization (BN) layer [37, 44] before adjusting model parameters. Hence, it is crucial to reliably estimate the test batch statistics and make necessary adjustments. Most of the recent advances have focused on robust estimations of the test batch statistics in a variety of scenarios, including continual distribution shifts [53], small test batches [30, 31], temporally correlated stream of test data [19], and out-of-distribution test data [20], where the exponential moving averaging (EMA) [20, 59] or interpolating source and test statistics [29, 34, 54] are proposed for robust statistics estimation.

Despite such efforts to build robust TTA methods, recent works [11, 54] have revealed the vulnerability of TTA methods that use the test batch statistics. By injecting small portions of malicious samples into the test batch, an adversary can easily manipulate the test batch statistics and also predictions on other (benign) samples, constituting a data poisoning attack. As we cannot presume the distribution of test samples in the real world, verifying the robustness of TTA methods against the data poisoning attack is essential since it can be considered as a worst-case study. Although the initial studies have proposed heuristics to partially address the vulnerability, it still remains a potential threat, posing a

*: Equal contribution; †: Correspondence to jungseul@postech.ac.kr

challenge even to state-of-the-art TTA methods.

This paper examines the potential vulnerabilities of existing TTA methods to data poisoning attacks through both theoretical (Section 5.2) and empirical (Section 6) investigations, including the state-of-the-art techniques [20,39,40,52]. Our theoretical analysis reveals that relying on the mean of test batch statistics creates a loophole that adversaries can exploit. This arises because the mean can be easily manipulated by even a single malicious sample, whereas the median proves to be robust against manipulation by a number of malicious samples, as illustrated in Figure 1. Furthermore, despite the integration of various modules for enhancing TTA robustness, our experiments show that state-of-the-art methods exhibit notable vulnerabilities to malicious samples.

Consequently, to address the adversarial risks in BN updates, we propose **Med**ian **B**atch **N**ormalization (MedBN) method that uses the median for estimating test batch statistics. Our approach stands out compared to existing defenses [11,54], as the model not only maintains model performance but also successfully defends against data poisoning attacks. Given the substantial vulnerability of state-of-the-art TTA methods [20, 39, 40, 52] to malicious samples, we demonstrate that integrating MedBN into each method consistently improves robustness against malicious samples.

Our main contributions are summarized as follows:

- Inspired by a theoretical analysis comparing mean and median, we propose MedBN, a simple and effective robust batch normalization method, which uses the median instead of the mean to estimate the batch statistics. We note that our method effortlessly integrates into existing TTA methods without additional training.
- Our experiments show that even sophisticated TTA methods are susceptible to data poisoning attacks, despite extensive efforts to enhance the robustness of TTA. This vulnerability arises from relying on the mean for estimation, which creates a potential loophole exploitable by adversaries.
- The robustness of the proposed MedBN is empirically justified by evaluating it over three standard benchmarks for TTA, seven TTA methods, and four different attack scenarios. Notably, MedBN outperforms comparing methods in robustness under attacks by a significant margin in all considered cases.

2. Related Works

Robust test-time adaptation methods. TTA methods have evolved to ensure robust performance under various scenarios in practice, including a single distribution shift in data distribution [52], continual distribution shifts [53], small batches of test data [30, 31], test data with temporal correlations [19], and out-of-distribution test data [20]. While significant efforts have been devoted to robustifying TTA methods, their robustness against malicious samples at test time has been relatively under-explored. Recent works [11, 54] have introduced data poisoning attack methods that generate malicious samples to sabotage TTA and demonstrated the vulnerability of a few TTA baselines [21, 33, 37, 43, 52]. In this work, we properly investigate the robustness of various state-of-the-art TTA methods against data poisoning attacks and also present a simple yet effective defense mechanism, which can be effortlessly added to most TTA methods.

Data poisoning attacks and defense mechanisms. There has been an extensive line of work on data poisoning attacks and defenses, but existing defense mechanisms are not applicable to TTA scenarios. For instance, adversarial training [18], a representative method, necessitates access to the training process, making it impractical for TTA where such access is unavailable. While some studies have proposed defense mechanisms specifically for data poisoning attacks in TTA [54], our experiments in Section 6 demonstrate that their effectiveness is limited. In contrast, our proposed method not only outperforms these defenses but also seamlessly integrates with any prior TTA methods. Additional discussion on related works is presented in Appendix D.

3. Preliminary

Let $\mathcal{X}$ be a sample space, and $\mathcal{Y}$ be a label space. Let $\mathcal{D}_{\text{src}} := \{(x_i, y_i)\}_{i\in[N_{\text{src}}]} \subseteq \mathcal{X} \times \mathcal{Y}$ be the training dataset of N_{src} labeled samples and $\mathcal{X}_{\text{test}} = \{x'_i\}_{i\in[N_{\text{test}}]} \subseteq \mathcal{X}$ be the test dataset of N_{test} unlabeled test samples. A model $f(\cdot;\theta)$ of parameters θ is pre-trained on $\mathcal{D}_{\text{src}}$, while it predicts a label $y \in \mathcal{Y}$ given a test sample $x \in \mathcal{X}_{\text{test}}$ in the presence of unknown domain shift. Depending on the context, a model f can output a distribution over labels.

TTA adjusts parameters while processing test data batch by batch where a test batch at time t is denoted by $\mathcal{B}^t \subseteq \mathcal{X}_{\text{test}}$. To address the domain shift, TTA methods that involve the adaptation of BN layers focus on adjusting BN layers, e.g., statistics and affine parameters of BN layers.

Batch normalization layers [28]. Noting that adapting parameters of BN layers is effective for TTA [20, 21, 40, 51], we describe the procedure of a BN layer converting input $z \in \mathbb{R}^{B\times C\times H\times W}$ to normalized $z' \in \mathbb{R}^{B\times C\times H\times W}$, where B, C, H, and W are the dimensions of batch, channel, height, and width, respectively. The normalization is performed channel-wisely with estimated BN statistics $(\hat{\mu}_c, \hat{\sigma}_c^2)$ and learnable affine parameters (β_c, γ_c) as follows:

$$z'_{bchw} = \gamma_c \cdot \frac{z_{bchw} - \hat{\mu}_c}{\sqrt{\hat{\sigma}_c^2 + \varepsilon}} + \beta_c \ , \qquad (1)$$

where ε is a small positive constant to avoid divided-by-zero. In the training, the BN statistics $(\hat{\mu}_c, \hat{\sigma}_c^2)$ are typically estimated by the EMA of the mean and variance of batches from source dataset $\mathcal{D}_{\text{src}}$, denoted by μ_{src} and σ^2_{src}, respectively.

Then, for every test batch $\mathcal{B}^t$, a traditional BN layer uses the same statistics μ_{src} and σ^2_{src} for $\hat{\mu}_c$ and $\hat{\sigma}^2_c$.

TTA with batch normalization. To tackle distribution shifts of test samples, a standard approach is TeBN [37] that estimates the test BN statistics (μ_c, σ^2_c) for $(\hat{\mu}_c, \hat{\sigma}^2_c)$ as follows:

$$\mu_c = \text{mean}\left\{z_{bchw}\right\}_{bhw} \ , \text{ and} \tag{2}$$

$$\sigma^2_c = \text{mean}\left\{(z_{bchw} - \mu_c)^2\right\}_{bhw} , \tag{3}$$

where z is the input to the BN layer given test batch $\mathcal{B}^t$ and we denote $\text{mean}\{z_i\}_{i\in\mathcal{I}} := \frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}} z_i$ is the average of z_i's over $i \in \mathcal{I}$. TENT [52] modulates the affine parameters (γ_c, β_c) in the BN layer (1) using TeBN by minimizing the entropy of model predictions on test samples. This simple strategy achieves excellent performance for distribution shifts and is commonly employed in TTA with adapted BN layers [20, 21, 40, 51]. However, it poses an adversarial risk because it adapts the test samples before making predictions, potentially including malicious samples. Section 4 describes our problem on TTA with malicious samples, followed by our method in Section 5. A detailed explanation of the vulnerability of TeBN is provided in Section 5.2, and a comprehensive analysis of the vulnerabilities in the state-of-the-art TTA methods with BN is presented in Section 6.

4. Problem Formulation

We have a batch $\mathcal{B}^t \subseteq \mathcal{X}_{\text{test}}$ at time t, part of which can be maliciously manipulated. We denote the malicious set by $\mathcal{B}^t_{\text{mal}}$ and the benign set by $\mathcal{B}^t_{\text{ben}}$ such that $\mathcal{B}^t = \mathcal{B}^t_{\text{mal}} \cup \mathcal{B}^t_{\text{ben}}$. We denote a tuple of labels of $\mathcal{B}^t$ as $\mathcal{Y}^t \subseteq \mathcal{Y}$ (and $\mathcal{Y}^t_{\text{ben}}$ is similarly defined). For simplicity, we denote a batch of labeled samples by $\mathcal{Z}^t$, i.e., $\mathcal{Z}^t$ is $\mathcal{B}^t$ with corresponding labels in $\mathcal{Y}^t$ (and $\mathcal{Z}^t_{\text{ben}}$ is similarly defined).

Our objective is to find a performant TTA method that is robust to malicious samples $\hat{\mathcal{B}}^t_{\text{mal}}$, which can be maliciously generated by solving the following bi-level optimization:

$$\hat{\mathcal{B}}^t_{\text{mal}} = \underset{\mathcal{B}^t_{\text{mal}}}{\arg\max}\, \mathcal{L}_{\text{attack}}(f(\cdot\,; \hat{\theta}(\mathcal{B}^t)), \mathcal{Y}^t) \ , \tag{4}$$

where $\hat{\theta}(\mathcal{B}^t)$ is updated parameters via the TTA method, i.e., $\hat{\theta}(\mathcal{B}^t) = \arg\min_\theta \mathcal{L}_{\text{TTA}}(\mathcal{B}^t; \theta)$, and $\mathcal{L}_{\text{attack}}$ is an attack objective function. For the attack objective, we consider both targeted attacks and indiscriminate attacks, as used in [54]. Solving bi-level optimization exactly is computationally expensive. However, TTA methods only perform a single-step update on θ for each $\mathcal{B}^t$, so we can approximate $\hat{\theta}$ as θ, as done in [54]. A detailed description of the attack algorithm and examples of malicious samples are presented in Appendix A and M, respectively. We confirm that TTA methods are vulnerable to these attacks. The detailed vulnerability of TTA methods can be found in Section 6. In the following, we consider two different attack types used to find $\hat{\mathcal{B}}^t_{\text{mal}}$.

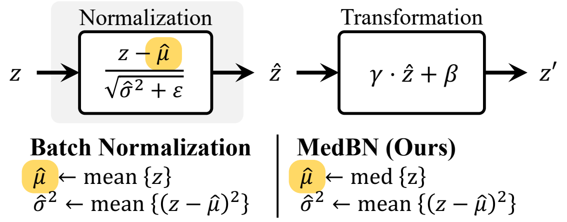

Normalization
Transformation
z
$\frac{z - \hat{\mu}}{\sqrt{\hat{\sigma}^2 + \varepsilon}}$
$\hat{z}$
$\gamma \cdot \hat{z} + \beta$
z'
Batch Normalization
$\hat{\mu} \leftarrow \text{mean}\,\{z\}$
$\hat{\sigma}^2 \leftarrow \text{mean}\,\{(z - \hat{\mu})^2\}$
MedBN (Ours)
$\hat{\mu} \leftarrow \text{med}\,\{z\}$
$\hat{\sigma}^2 \leftarrow \text{mean}\,\{(z - \hat{\mu})^2\}$

Figure 2. **An overview of MedBN.** (Top) TTA methods adapted with BN layers normalize the features (z) by estimating normalization statistics $\hat{\mu}$ and $\hat{\sigma}^2$, and optimize transformation parameters γ and β. (Bottom) In contrast to conventional BN, which computes the statistics based on the mean of inputs, our proposed MedBN utilizes the median value for estimating the statistics, $\hat{\mu}$ and $\hat{\sigma}^2$.

Targeted attack. The goal of a targeted attack is to manipulate $\mathcal{B}^t_{\text{mal}}$ fed into the TTA method such that the adapted model predicts a targeted label y^t_{target} on a targeted sample $x^t_{\text{target}} \in \mathcal{B}^t_{\text{ben}}$ as follows:

$$\hat{\mathcal{B}}^t_{\text{mal}} = \underset{\mathcal{B}^t_{\text{mal}}}{\arg\max} -\mathcal{L}_{\text{CE}}(f(x^t_{\text{target}}; \hat{\theta}(\mathcal{B}^t)), y^t_{\text{target}}) \ , \tag{5}$$

where $\mathcal{L}_{\text{CE}}$ is the cross-entropy loss.

Indiscriminate attack. The objective of an indiscriminate attack is to degrade the performance of benign samples $\mathcal{B}^t_{\text{ben}}$ by manipulating $\mathcal{B}^t_{\text{mal}}$ as follows:

$$\hat{\mathcal{B}}^t_{\text{mal}} = \underset{\mathcal{B}^t_{\text{mal}}}{\arg\max} \sum_{(x,y)\in\mathcal{Z}^t_{\text{ben}}} \mathcal{L}_{\text{CE}}(f(x; \hat{\theta}(\mathcal{B}^t)), y) \ . \tag{6}$$

Adversary's knowledge. We mainly consider a white box attack scenario where an adversary possesses knowledge of a pre-trained model, a TTA algorithm (including defense mechanism), a batch, and even the labels of samples in the batch. Our study against such a mighty adversary can be interpreted as a worst-case analysis, while we also consider more practicable (yet milder) attack scenarios with limited adversaries' knowledge and adaptive attack which obfuscates defense mechanisms in Appendix B.

5. Methodology

We propose our robust TTA method, Median Batch Normalization (MedBN), followed by its robustness analysis.

5.1. Median Batch Normalization (MedBN)

Test statistics calculated by mean can be contaminated by data poisoning attacks, as demonstrated by Theorem 1 in the following section, which in turn, disrupt the model's adaptation and lead to incorrect predictions. To mitigate the effect of malicious samples, we propose a simple approach, called Median Batch Normalization (MedBN). MedBN uses the median instead of the mean for the standardization (1) as follows, i.e., (η_c, ρ^2_c) instead of (μ_c, σ^2_c) for $(\hat{\mu}_c, \hat{\sigma}^2_c)$:

$$\eta_c = \text{med}\left\{z_{bchw}\right\}_{bhw} \ , \text{ and} \tag{7}$$

$$\rho^2_c = \text{mean}\{(z_{bchw} - \eta_c)^2\}_{bhw} \ , \tag{8}$$

where $\text{med}\{A\} := \min\{a \in A : |\{x \in A : a > x\}| \geq \frac{|A|}{2}\}$ for a set $A \subseteq \mathbb{R}$. Here, MedBN standardizes an input z using (η_c, ρ_c^2). Our method is surprisingly effective for the defense against attacks with negligible degradation of model performance. Also, its simplicity allows for easy integration within any existing TTA methods that adjust BN layers.

Note that ρ_c takes the mean of the squared deviations $(z_{bchw} - \eta_c)^2$'s, we can instead take the median of the deviations, which corresponds to the median absolute deviation (MAD), as a part of further robustifying the estimation of BN statistics. According to our study, the use of MAD shows strong defense but a substantial performance drop. Hence, we choose the mean of the squared deviations $(z_{bchw} - \eta_c)^2$'s for our method, see Appendix G for results using MAD.

5.2. Illustrative analysis: mean vs. median

The main idea of our method is to replace the use of means with that of medians when computing BN statistics. We provide an illustrative analysis comparing the robustness of using median instead of mean.

Theorem 1 *Consider a set of n numbers $\mathcal{B} = \{x_i \in \mathbb{R} : i \in [n]\}$ and $1 \leq m \leq n$ where the first m numbers are possibly manipulated by adversaries. Let $\mathcal{B}_{\text{mal}} = \{x_i : i \in [m]\}$, and $\mathcal{B}_{\text{ben}} = \mathcal{B} \setminus \mathcal{B}_{\text{mal}}$.*
(i) The mean can be arbitrarily manipulated by a single malicious sample, i.e., for any $1 \leq m \leq n$,

$$\sup_{\mathcal{B}_{\text{mal}}} |\text{mean}(\mathcal{B}_{\text{mal}} \cup \mathcal{B}_{\text{ben}}) - \text{mean}(\mathcal{B}_{\text{ben}})| = \infty \,. \tag{9}$$

(ii) The median is robust against malicious samples unless they are not the majority, i.e., for any $1 \leq m < n/2$,

$$\sup_{\mathcal{B}_{\text{mal}}} |\text{med}(\mathcal{B}_{\text{mal}} \cup \mathcal{B}_{\text{ben}}) - \text{med}(\mathcal{B}_{\text{ben}})| < \infty \,, \textit{ and} \tag{10}$$

$$\sup_{\mathcal{B}_{\text{mal}}} |\text{med}(\mathcal{B}_{\text{mal}} \cup \mathcal{B}_{\text{ben}}) - \text{mean}(\mathcal{B}_{\text{ben}})| < \infty \,. \tag{11}$$

The first part of Theorem 1 implies the risk of using mean in the presence of malicious samples. In particular, it says that just a single malicious sample can arbitrarily manipulate the estimation of mean statistics. However, as the second part of Theorem 1 suggests, such an arbitrary manipulation by malicious samples is not possible unless the attacker modifies more than half of the batch. It is noteworthy that the robustness of the median for scalars in Theorem 1 can be extended for coordinate-wise or geometric median for vectors as well. We provide this extension in Appendix H.

Proof of Theorem 1. For the first part of the vulnerability of mean (9), we consider a specific choice of $\mathcal{B}'_{\text{mal}}$ consisting of m-many $(\text{mean}(\mathcal{B}_{\text{ben}}) + L)$'s for $L \in \mathbb{R}$. Then, we have

$$\begin{aligned} &\sup_{\mathcal{B}_{\text{mal}}} |\text{mean}(\mathcal{B}_{\text{mal}} \cup \mathcal{B}_{\text{ben}}) - \text{mean}(\mathcal{B}_{\text{ben}})| \\ &\geq |\text{mean}(\mathcal{B}'_{\text{mal}} \cup \mathcal{B}_{\text{ben}}) - \text{mean}(\mathcal{B}_{\text{ben}})| = \frac{m}{n} L \,, \end{aligned} \tag{12}$$

where the last equality is from the choice of $\mathcal{B}'_{\text{mal}}$ such that

$$\begin{aligned} &n \cdot \text{mean}(\mathcal{B}'_{\text{mal}} \cup \mathcal{B}_{\text{ben}}) \\ &= (n - m) \cdot \text{mean}(\mathcal{B}_{\text{ben}}) + m \cdot \text{mean}(\mathcal{B}_{\text{ben}}) + mL \,. \end{aligned} \tag{13}$$

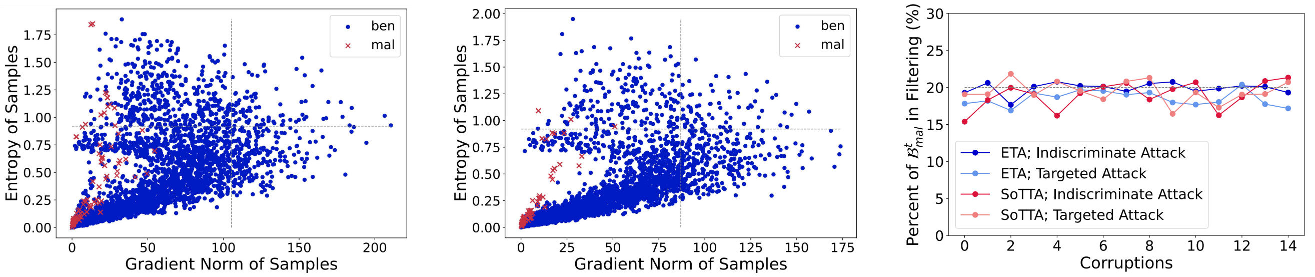
This directly leads to (9) as the choice of L is arbitrary.

For the second part on the robustness of median, we focus on (10) as the proof of (11) follows similarly. For (10), let $k = \text{med}(\mathcal{B}_{\text{mal}} \cup \mathcal{B}_{\text{ben}})$. If $k \in \mathcal{B}_{\text{ben}}$, it is trivial. If $k \in \mathcal{B}_{\text{mal}}$, given that $1 \leq m < n/2$, it follows that $\min(\mathcal{B}_{\text{ben}}) \leq k \leq \max(\mathcal{B}_{\text{ben}})$. Then, $|\text{med}(\mathcal{B}_{\text{mal}} \cup \mathcal{B}_{\text{ben}}) - \text{med}(\mathcal{B}_{\text{ben}})| \leq \max_{x,x' \in \mathcal{B}_{\text{ben}}} |x - x'| < \infty$. Therefore, this shows (10) and completes the proof of Theorem 1.

6. Vulnerability of Existing TTA Methods against Attacks

In this section, we delve into the effectiveness of TTA methods against malicious samples. For stabilizing adaptation to test data, many TTA methods propose a variety of modules, including screening out samples to remove noisy ones, optimizing model weights to resist large and noisy gradients, and employing exponential moving averages (EMA) for stable updates of batch normalization statistics. Hence, we study the influence of these TTA modules against malicious samples across three schemes: (i) filtering, (ii) sharpness-aware learning, and (iii) EMA.

Filtering scheme. Several research works [20, 39, 40] have proposed the use of filtering modules. The purpose of these modules is to eliminate noisy samples from the adaptation process, based on evaluating the entropy or softmax predictions of model outputs, e.g., screening out samples with high entropy [39, 40] or low confidence [20]. By filtering out these potentially problematic samples, the model can be more stably adapted to test data. To identify the malicious samples filtered out by the module using entropy or softmax confidence, we observe the distribution of malicious samples in the entropy-gradient space in two attack scenarios: targeted and indiscriminate attacks with 100 attack steps, a batch size of 200, and 40 malicious samples in each batch. As illustrated in Figure 3a and Figure 3b, malicious samples tend to be clustered with low entropy values, making it challenging to exclude the malicious samples. To verify this finding, we investigate the proportion of malicious samples actually filtered out by ETA [39] and SoTTA [20]. ETA filters samples with high entropy, i.e., $f(x;\theta^t) \log f(x;\theta^t)$, while SoTTA screens out samples with low softmax confidence of model outputs, i.e., $\max_{i \in [c]}(e^{f(x;\theta^t)} / \sum_{j=1}^{c} e^{f(x;\theta^t)})$, where c denotes the number of classes. As shown in Figure 3c, we observe that malicious samples still exist in the filtered batch (at least 15% of the filtered batch are malicious samples). Considering that malicious samples constitute 20% of the batch, these results demonstrate that entropy or softmax confidence-based filtering mechanisms are unable to completely remove



(a) Sample entropy and gradient norm distribution under targeted attack.

(b) Sample entropy and gradient norm distribution under indiscriminate attack.

(c) The ratio of malicious samples $\mathcal{B}^t_{mal}$ in filtering over corruptions.

Figure 3. Analysis of vulnerability of existing TTA Methods against attacks. Figure 3a and Figure 3b represent the relation between entropy and gradient norm of benign and malicious samples in targeted attack and indiscriminate attack, respectively. Figure 3c illustrates the proportion of malicious samples $\mathcal{B}_{mal}$ among the total remaining samples after filtering over the type of corruption, considering an initial condition where 20% of the samples in the batch were malicious. All experiments are performed on CIFAR10-C dataset with Gaussian noise, using a ResNet26, at the highest severity of distribution shift, i.e., level 5.

all malicious samples and allow a high percentage of malicious samples to pass through.

Sharpness-aware learning scheme. Sharpness-aware learning [40], following Sharpness-Aware Minimization (SAM) [17], focuses on the stability of model parameters by guiding them towards a flat minimum in the loss surface. This approach is based on the understanding that a flat minimum is more desirable for model robustness, especially in the presence of noisy or large gradients. However, as shown in Figure 3a and Figure 3b, the gradient norm of the malicious samples, indicated by the x-axis, is concentrated in regions with small gradients. This indicates that the SAM does not make the model to be robust against malicious samples.

Exponential moving averages (EMA) scheme. Exponential Moving Averages (EMA) scheme controls the statistics of BN layers, starting with the source statistics (μ_{src} and σ^2_{src}) from the training phase [46,50]. This differs from approaches like TeBN, which solely rely on test batch statistics. The EMA scheme is defined as follows:

$$\hat{\mu}_t = \alpha\hat{\mu}_{t-1} + (1-\alpha)\mu_t \ , \tag{14}$$

$$\hat{\sigma}^2_t = \alpha\hat{\sigma}^2_{t-1} + (1-\alpha)\sigma^2_t \ , \tag{15}$$

where $\mu_0 = \mu_{\text{src}}$, $\sigma^2_0 = \sigma^2_{\text{src}}$, and $\alpha \in [0, 1]$ is a momentum parameter. Leveraging a larger proportion ($\alpha > 0.5$) of previous statistics ($t-1$) can mitigate the influence of malicious samples but there exists potential performance degradation of the model to target distribution. Conversely, utilizing a larger proportion ($\alpha < 0.5$) of current statistics (t) allows for adaptation to the target distribution, but it compromises the robustness against malicious samples. This presents that there is a trade-off requiring strategic consideration for choosing α.

7. Experiments

In this section, we provide the results of experimental evaluations of MedBN. A detailed description of the experimental setup is presented in Section 7.1. The results on various attack scenarios for both image classification and semantic segmentation are presented in Section 7.2 and 7.3, respectively. We investigate the reasons behind the robustness of MedBN against in Section 7.4. Lastly, Section 7.5 presents an ablation study of hyper-parameters such as the number of malicious samples and the test batch size. More details of the experiments are provided in Appendix C.

7.1. Experimental setup

Datasets and model architectures. We evaluate our approach using three major benchmarks for TTA [25]: CIFAR10-C, CIFAR100-C, and ImageNet-C, which represent perturbed versions of the original CIFAR10, CIFAR100, and ImageNet datasets, respectively. We use ResNet-26 [24] for CIFAR10-C and CIFAR100-C experiments, and ResNet-50 [24] for ImageNet-C experiments. The models are pretrained on clean CIFAR10, CIFAR100, and ImageNet training sets from [13], respectively, and then evaluated on the aforementioned corrupted test sets. We additionally demonstrate the effectiveness of MedBN for various model architectures in Appendix E.

Test-time adaptation baselines. We consider seven TTA methods as baselines, that update batch statistics or the affine parameters of BN layers. Test-time normalization (TeBN) [37] updates BN statistics for each test batch. TENT [52] updates the affine parameters in BN layers using entropy minimization. Efficient anti-forgetting test-time adaptation (EATA) [39] improves a sample-efficient entropy minimization and Fisher regularizer to prevent knowledge loss from pre-trained model. ETA denotes EATA without Fisher regularization. Sharpness-aware and reliable optimization (SAR) [40] with BN layers and screening-out test-time adaptation (SoTTA) [20] leverage sample filtering and sharpness-aware minimization [17] to reduce the negative effects caused by large gradients. Source-initialized exponential moving average (sEMA) [20, 46, 50, 59] manages BN layers' statistics using EMA with the source statistics from the training phase

Table 1. Attack Success Rate (%) of the targeted and instant attack scenario. See Table 15 in Appendix J for a comprehensive comparison in ASRs over different corruptions. The rightmost column refer the error rates for TeBN without attacks. See Table 19 in Appendix L for the error rates without attacks of all methods.

Dataset	B / m	Normalization	Method							$m = 0$
			TeBN	TENT	ETA	SAR	SoTTA	sEMA	mDIA	TeBN (ER %)
CIFAR10-C	200 / 40	BatchNorm	83.91	72.36	75.07	77.42	21.47	18.18	33.91	14.92
	(20%)	MedBN (Ours)	**19.16**	**18.36**	**18.00**	**18.04**	**7.82**	**8.67**	**8.76**	15.19
CIFAR100-C	200 / 40	BatchNorm	91.78	79.29	79.96	81.64	7.60	8.71	16.62	40.08
	(20%)	MedBN (Ours)	**2.80**	**4.18**	**3.02**	**3.02**	**2.58**	**1.60**	**2.00**	40.77
ImageNet-C	200 / 20	BatchNorm	97.78	91.47	94.49	64.53	15.29	11.02	32.18	66.62
	(10%)	MedBN (Ours)	**0.36**	**0.44**	**0.44**	**0.44**	**0.80**	**0.27**	**1.07**	69.55

Table 2. Error Rate (%) of the indiscriminate and instant attack scenario. See Table 16 in Appendix J for a comprehensive comparison in ERs over different corruptions. The rightmost column refer the error rates for TeBN without attacks. See Table 19 in Appendix L for the error rates without attacks of all methods.

Dataset	B / m	Normalization	Method							$m = 0$
			TeBN	TENT	ETA	SAR	SoTTA	sEMA	mDIA	TeBN (ER %)
CIFAR10-C	200 / 40	BatchNorm	31.02	28.13	27.42	27.56	20.40	21.65	27.96	14.92
	(20%)	MedBN (Ours)	**22.34**	**20.30**	**19.81**	**19.60**	**16.49**	**17.77**	**19.06**	15.19
CIFAR100-C	200 / 40	BatchNorm	59.80	55.10	54.45	56.40	48.33	46.89	55.43	40.08
	(20%)	MedBN (Ours)	**48.55**	**46.96**	**46.59**	**48.00**	**45.38**	**43.35**	**47.84**	40.77
ImageNet-C	200 / 20	BatchNorm	81.46	72.82	74.15	77.74	66.05	73.21	77.28	66.62
	(10%)	MedBN (Ours)	**69.74**	**68.01**	**68.47**	**69.54**	**64.22**	**70.22**	**69.24**	69.55

Table 3. Attack Success Rate (%) of the targeted and cumulative attack scenario on CIFAR10-C and Error Rate (%) of the indiscriminate and cumulative attack scenario on CIFAR10-C. See Table 17 and Table 18 in Appendix K over different corruptions and other TTA benchmarks.

Objective	Dataset	B / m	Normalization	Method							$m = 0$
				TeBN	TENT	EATA	SAR	SoTTA	sEMA	mDIA	TeBN (ER %)
Targeted	CIFAR10-C	200 / 40	BatchNorm	84.04	74.18	75.73	76.80	21.16	16.13	34.09	14.92
Attack		(20%)	MedBN (Ours)	**19.20**	**18.80**	**21.02**	**8.76**	**8.13**	**8.89**	**19.06**	15.19
Indisctiminate	CIFAR10-C	200 / 40	BatchNorm	35.30	35.70	35.30	31.25	26.10	28.79	32.05	14.92
Attack		(20%)	MedBN (Ours)	**27.22**	**25.84**	**26.84**	**24.29**	**22.52**	**25.62**	**23.96**	15.19

as the initial value in (14) and (15). We use $\alpha = 0.8$ for stable update. Lastly, mitigating Distribution Invading Attacks (mDIA) [54] interpolates source statistics and test batch statistics in BN layers, except terminal BN layers.

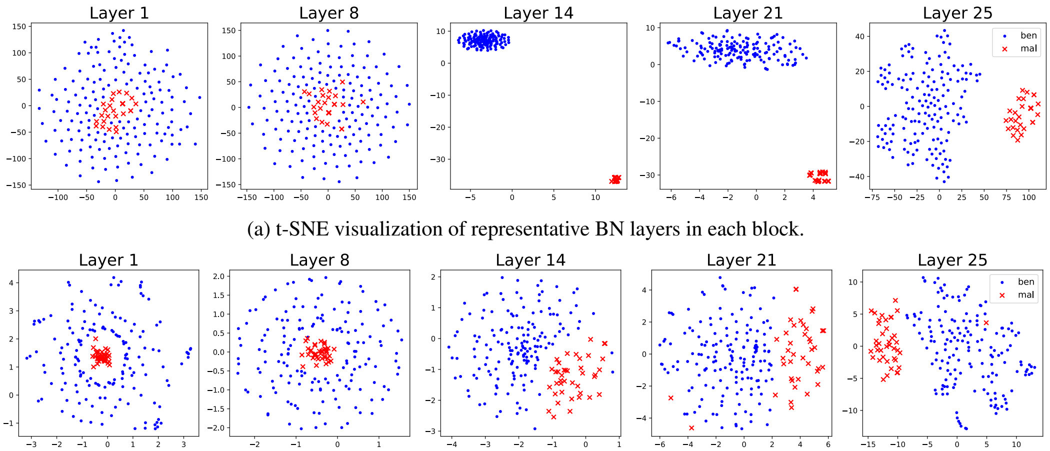
Attack scenarios. We consider four different attack scenarios over two purposes of attacks and two frequencies of attacks. In particular, *targeted* and *indiscriminate* attacks are two purposes of attacks as outlined in Section 4. For each purpose of attack, we additionally consider two types of attack: an instant attack and a cumulative attack. In *the instant attack scenario*, the attacker injects a set of malicious data into the t-th batch after adapting to the previous $(t-1)$ benign batches [54]. On the other hand, *the cumulative attack scenario* involves an attack across all batches, from the first batch up to T-th batch, where T is the total number of batches. For the number of malicious samples m per batch, we use 40, 40, and 20 for CIFAR10-C, CIFAR100-C, and ImageNet-C, respectively, out of 200 samples in a batch.

Evaluation metrics. For the evaluation of targeted attacks, we utilize the metric of Attack Success Rate (ASR), i.e., $\frac{1}{T}\sum_{t=1}^{T}\mathbb{1}(f(x^t_{\text{target}};\hat{\theta}_t) = y^t_{\text{target}})$. The performance of indiscriminate attacks is assessed through the Error Rate (ER) on benign samples after the attack, i.e., $\frac{1}{T}\sum_{t=1}^{T}\frac{1}{|\mathcal{Z}^t_{\text{ben}}|}\sum_{(x,y)\in\mathcal{Z}^t_{\text{ben}}}\mathbb{1}(f(x;\hat{\theta}_t) \neq y)$. For each purpose of attack, $f(\cdot;\hat{\theta}_t)$ is an adapted model after each attack using $\mathcal{Z}^t$. Note that in the instant attack scenario at time t, the model $f(\cdot,\hat{\theta}_{t-1})$ is updated until $(t-1)$ via TTA without any attacks. Finally, to measure the model's performance under a normal TTA setup, we use the standard TTA metric, i.e., the ER on benign samples without attacks (i.e., $m = 0$).

7.2. Main results

We demonstrate the efficacy of our method used with seven different TTA algorithms and evaluate three TTA benchmarks under four different attack scenarios.

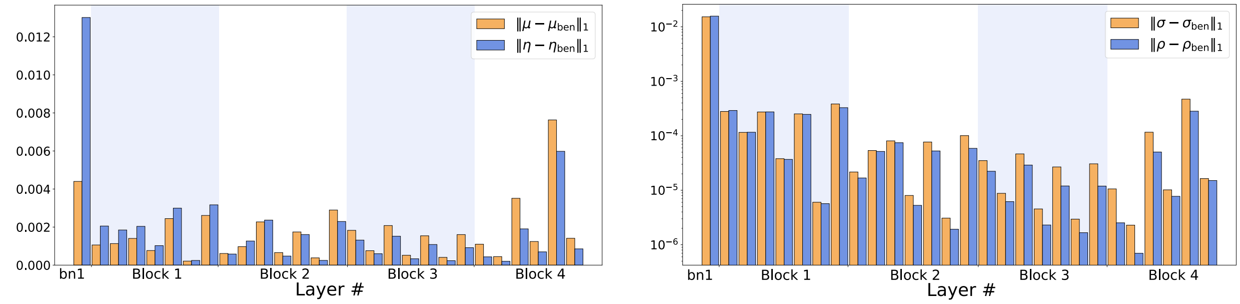
The instant attack scenario. Table 1 and Table 2 demonstrate the effectiveness of MedBN for targeted attacks and



(a) t-SNE visualization of representative BN layers in each block.

(b) t-SNE visualization of representative MedBN layers in each block.

Figure 4. t-SNE visualizations in representative layers of BN and MedBN, across ResNet26 blocks, with benign samples (blue dots) and malicious samples (red crosses).



(a) Comparison of $\|\mu - \mu_{\text{ben}}\|_1$ and $\|\eta - \eta_{\text{ben}}\|_1$ across different layers.

(b) Comparison of $\|\sigma - \sigma_{\text{ben}}\|_1$ and $\|\rho - \rho_{\text{ben}}\|_1$ across different layers.

Figure 5. L1 distance for measuring the amount of perturbation by malicious samples.

indiscriminate attacks, respectively, under the instant attack scenario. By simply integrating MedBN into TTA methods with BN layers, it demonstrates significant robustness against malicious samples, i.e., the lower attack success rates under targeted attacks and lower error rates under indiscriminate attacks over all cases, but also achieves minimal performance degradation without attacks.

In Table 1 for targeted attacks, the ASRs of all TTA methods are consistently less than 20% for CIFAR10-C, 10% for CIFAR100-C, and 1% for ImageNet-C. While SoTTA and EMA inherently possess defensive capabilities with the use of batch statistics EMA, integrating MedBN further enhances the robustness, yielding the lowest ASRs compared to other methods assessed in this study. In Table 2 for indiscriminate attacks, the results across all TTA methods indicate that MedBN shows reduced error rates as high as approximately 9% in CIFAR10-C, 11% in CIFAR100-C, and 12% in ImageNet-C. As in targeted attacks, it is noteworthy that while SoTTA and sEMA naturally provide some defense with standard BN layers, incorporating MedBN substantially enhances this protection, leading to the lowest error rates observed in all studied methods.

The cumulative attack scenario. The efficiency of MedBN is indicated in Table 3 under the cumulative attack scenario including targeted attacks and indiscriminate attacks on CIFAR10-C. The results on other datasets can be found in Appendix K. Unlike an instant attack, which involves injecting malicious data into a single batch after adapting to previous benign batches, a cumulative attack spreads across all batches. Integrating malicious samples consistently throughout the entire dataset can significantly degrade the model, depending on the attacker's goals. Particularly, cumulative attacks have a more pronounced impact in indiscriminate scenarios, where the performance reductions from earlier attacks can accumulate. Even in the cumulative attack scenario, MedBN shows lower ASRs in the targeted attack scenario and lower ERs in the indiscriminate attack scenario.

7.3. Experiments on semantic segmentation

We expand our experiments to incorporate a semantic segmentation task, examining two instant attack objectives: a targeted attack on segmentation, which aims to manipulate the prediction for a targeted pixel within an image, and an indiscriminate attack on segmentation, intending to disturb predictions on all the benign samples. Each batch comprises one malicious image and the others benign image with a batch size of 3. Table 4 shows that MedBN effectively defends against both attack scenarios while preserving the mean Intersection over Union (mIoU) on benign images. Additional experimental details are provided in Appendix C.

Table 4. Attack Sucess Rate (%) in instant targeted attack on segmentation and mIoU (%) on the benign images in instant indiscriminate attack on segmentation using TeBN to adapt the model trained on Cityscapes [12] for SYNTHIA [42].

Objective	Normalization	TeBN	mIoU (%) ($m = 0$)
Targeted Attack	BatchNorm	69.17	25.43
(ASR ↓)	MedBN (Ours)	**0.00**	24.24
Indiscriminate Attack	BatchNorm	17.11	25.43
(mIoU ↑)	MedBN (Ours)	**21.55**	24.24

7.4. Why is MedBN robust against attacks?

We investigate how MedBN counteracts the effects of malicious samples. First, we plot the t-SNE of features for each block before going through BN layers during the adaptation using TeBN on Gaussian corruptions in CIFAR10-C. The t-SNE for all BN layers can be found in Appendix I. In Figure 4a, except for the early layers, malicious samples become outliers compared to benign ones. Therefore, as demonstrated in Theorem 1, the mean is exposed to be contaminated by these malicious samples and results in the misbehavior of the model. However, when we plot the same t-SNE for MedBN layers, we observe that the malicious samples are closed from the benign samples as shown in Figure 4b, i.e., the effect of malicious samples is significantly mitigated. For the early layers that capture low-level features [3, 41, 60], the features of malicious samples are close to those of benign samples, since the malicious samples are generated by adding the imperceptible noise, making them similar to the benign samples. However, for deeper layers, the malicious samples tend to go distant from the benign samples to mislead the model. Secondly, to verify the robustness of MedBN, we measure the L_1 distance $||\mu - \mu_{\text{ben}}||_1$ and $||\eta - \eta_{\text{ben}}||_1$, $||\sigma - \sigma_{\text{ben}}||_1$ and $||\rho - \rho_{\text{ben}}||_1$. As shown in Figure 5, as the layer gets deeper, the influence of perturbation by malicious samples is smaller for MedBN statistics than BN statistics. These results align with Theorem 1 and the results of t-SNE for BN and MedBN.

7.5. Ablation studies

We perform ablation studies on four distint cases, using CIFAR10-C and CIFAR100-C datasets in targeted and indiscriminate attack scenarios, varying malicious samples and test batch size. Our focus is on evaluating the TeBN method, as it addresses the vulnerabilities related to robustly estimating BN statistics while excluding learnable parameters.

The number of malicious samples. We investigate the robustness of the MedBN against various ratios of malicious samples with batch size of 200. Across various malicious ratios, MedBN is consistently robust under targeted attacks in CIFAR10-C (Table 5). The remaining three cases with results are provided in Appendix F.

Table 5. Attack Success Rate (%) of targeted and instant attack for different numbers of malicious samples with batch size of 200.

Dataset	Normalization	# of Malicious Samples (m)				
		10	20	40	60	80
CIFAR10-C	BatchNorm	21.60	42.00	84.00	96.67	99.47
	MedBN (Ours)	**7.07**	**10.27**	**19.20**	**26.80**	**38.27**

Test batch size. We explore the effect of different batch sizes. In all cases, the ratio of malicious samples is about 20%. For targeted attacks in CIFAR10-C (Table 6), MedBN achieves significantly lower ASR than BN at all batch sizes. Note that as the batch size gets smaller, a successful attack gets more difficult because there is less malicious data. The results for the remaining three cases are included in Appendix F.

Table 6. Attack Success Rate (%) of targeted and instant attack for different batch size B with a consistent 20% of malicious samples.

Dataset	Normalization	Batch-size (B)				
		200	128	64	32	16
CIFAR10-C	BatchNorm	83.91	87.76	84.84	83.87	84.60
	MedBN (Ours)	**19.16**	**20.51**	**17.83**	**20.19**	**29.14**

8. Conclusion

We provide a comprehensive study disclosing potential threats of existing TTA methods mainly due to their vulnerable estimation of BN statistics despite the remarkable advances in TTA. Hence, we propose MedBN, an simple yet effective robust batch normalization method against malicious samples, which can be effortlessly combined with most of the existing TTA methods if BN layers are being adapted. Our comprehensive experiments demonstrate the robustness and general applicability of MedBN. In particular, we show that applying MedBN to other methods results in significant performance improvements, implying that MedBN helps attain outstanding robustness. For example, applying MedBN to SoTTA (one of the state-of-the-art) shows the best robustness across all benchmarks. We believe that our work can provide a general robust batch normalization for future work.

Acknowledgements. This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT); (No. 2019-0-01906, Artificial Intelligence Graduate School Program (POSTECH)), (No. 2021-0-00739, Development of Distributed/Cooperative AI based 5G+ Network Data Analytics Functions and Control Technology), and (No. 2021-0-02068, Artificial Intelligence Innovation Hub).

References

[1] Gilad Baruch, Moran Baruch, and Yoav Goldberg. A little is enough: Circumventing defenses for distributed learning. *Advances in Neural Information Processing Systems*, 32:8632–8642, 2019. 14

[2] Battista Biggio, Blaine Nelson, and Pavel Laskov. Poisoning attacks against support vector machines. In *International Conference on Machine Learning*, pages 1467–1474. PMLR, 2012. 13

[3] Santiago A Cadena, Marissa A Weis, Leon A Gatys, Matthias Bethge, and Alexander S Ecker. Diverse feature visualizations reveal invariances in early layers of deep neural networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 217–232, 2018. 8

[4] Han Cai, Chuang Gan, Ligeng Zhu, and Song Han. Tinytl: Reduce memory, not parameters for efficient on-device learning. *Advances in Neural Information Processing Systems*, 33:11285–11297, 2020. 13

[5] Nicholas Carlini and Andreas Terzis. Poisoning and backdooring contrastive learning. In *International Conference on Learning Representations*, 2021. 13

[6] Dian Chen, Dequan Wang, Trevor Darrell, and Sayna Ebrahimi. Contrastive test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 295–305, 2022. 1, 13

[7] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848, 2017. 13

[8] Yudong Chen, Lili Su, and Jiaming Xu. Distributed statistical machine learning in adversarial settings: Byzantine gradient descent. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 1(2):1–25, 2017. 14

[9] Sungha Choi, Sanghun Jung, Huiwon Yun, Joanne T Kim, Seungryong Kim, and Jaegul Choo. Robustnet: Improving domain generalization in urban-scene segmentation via instance selective whitening. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11580–11590, 2021. 13

[10] Sungha Choi, Seunghan Yang, Seokeon Choi, and Sungrack Yun. Improving test-time adaptation via shift-agnostic weight regularization and nearest source prototypes. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 440–458. Springer, 2022. 13

[11] Tianshuo Cong, Xinlei He, Yun Shen, and Yang Zhang. Test-time poisoning attacks against test-time adaptation models. In *2024 IEEE Symposium on Security and Privacy (SP)*, pages 72–72. IEEE Computer Society, 2023. 1, 2, 13

[12] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3213–3223, 2016. 8, 13

[13] Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Edoardo Debenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. RobustBench: a standardized adversarial robustness benchmark. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2021. 5, 14

[14] Mario Döbler, Robert A Marsden, and Bin Yang. Robust mean teacher for continual and gradual test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7704–7714, 2023. 1, 13

[15] El Mahdi El-Mhamdi, Sadegh Farhadkhani, Rachid Guerraoui, Arsany Guirguis, Lê-Nguyên Hoang, and Sébastien Rouault. Collaborative learning in the jungle (decentralized, byzantine, heterogeneous, asynchronous and nonconvex learning). *Advances in Neural Information Processing Systems*, 34:25044–25057, 2021. 14

[16] Sadegh Farhadkhani, Rachid Guerraoui, Nirupam Gupta, Rafael Pinot, and John Stephan. Byzantine machine learning made easy by resilient averaging of momentums. In *International Conference on Machine Learning*, pages 6246–6283. PMLR, 2022. 16

[17] Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations*, 2020. 5

[18] J Geiping, L Fowl, G Somepalli, M Goldblum, M Moeller, and T Goldstein. What doesn't kill you makes you robust (er): Adversarial training against poisons and backdoors. corr, 2021. 2, 13

[19] Taesik Gong, Jongheon Jeong, Taewon Kim, Yewon Kim, Jinwoo Shin, and Sung-Ju Lee. Note: Robust continual test-time adaptation against temporal correlation. *Advances in Neural Information Processing Systems*, 35:27253–27266, 2022. 1, 2

[20] Taesik Gong, Yewon Kim, Taeckyung Lee, Sorn Chottananurak, and Sung-Ju Lee. Sotta: Robust test-time

adaptation on noisy data streams. *Advances in Neural Information Processing Systems*, 36, 2023. 1, 2, 3, 4, 5, 13

[21] Sachin Goyal, Mingjie Sun, Aditi Raghunathan, and J Zico Kolter. Test time adaptation via conjugate pseudo-labels. *Advances in Neural Information Processing Systems*, 35:6204–6218, 2022. 1, 2, 3

[22] Rachid Guerraoui, Sébastien Rouault, et al. The hidden vulnerability of distributed learning in byzantium. In *International Conference on Machine Learning*, pages 3521–3530. PMLR, 2018. 14

[23] Nirupam Gupta, Shuo Liu, and Nitin Vaidya. Byzantine fault-tolerant distributed machine learning with norm-based comparative gradient elimination. In *2021 51st Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W)*, pages 175–181. IEEE, 2021. 14

[24] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 5

[25] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2018. 1, 5, 13

[26] Dan Hendrycks, Norman Mu, Ekin Dogus Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. In *International Conference on Learning Representations*, 2019. 14

[27] Junyuan Hong, Lingjuan Lyu, Jiayu Zhou, and Michael Spranger. Mecta: Memory-economic continual test-time model adaptation. In *International Conference on Learning Representations*, 2023. 13

[28] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pages 448–456. PMLR, 2015. 2, 13

[29] Juwon Kang, Nayeong Kim, Donghyeon Kwon, Jungseul Ok, and Suha Kwak. Leveraging proxy of training data for test-time adaptation. In *International Conference on Machine Learning*, pages 15737–15752. PMLR, 2023. 1

[30] Ansh Khurana, Sujoy Paul, Piyush Rai, Soma Biswas, and Gaurav Aggarwal. Sita: Single image test-time adaptation. *arXiv preprint arXiv:2112.02355*, 2021. 1, 2

[31] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, pages 5637–5664. PMLR, 2021. 1, 2

[32] Leslie Lamport, Robert Shostak, and Marshall Pease. The byzantine generals problem. In *Concurrency: the works of leslie lamport*, pages 203–226. 2019. 14

[33] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896. Atlanta, 2013. 2

[34] Hyesu Lim, Byeonggeun Kim, Jaegul Choo, and Sungha Choi. Ttn: A domain-shift aware batch normalization in test-time adaptation. In *International Conference on Learning Representations*, 2022. 1, 13

[35] Yuejiang Liu, Parth Kothari, Bastien Van Delft, Baptiste Bellot-Gurlet, Taylor Mordan, and Alexandre Alahi. Ttt++: When does self-supervised test-time training fail or thrive? *Advances in Neural Information Processing Systems*, 34:21808–21820, 2021. 13

[36] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018. 12

[37] Zachary Nado, Shreyas Padhy, D Sculley, Alexander D'Amour, Balaji Lakshminarayanan, and Jasper Snoek. Evaluating prediction-time batch normalization for robustness under covariate shift. *arXiv preprint arXiv:2006.10963*, 2020. 1, 2, 3, 5, 13

[38] Blaine Nelson, Marco Barreno, Fuching Jack Chi, Anthony D Joseph, Benjamin IP Rubinstein, Udam Saini, Charles Sutton, JD Tygar, and Kai Xia. Exploiting machine learning to subvert your spam filter. In *Proceedings of the 1st Usenix Workshop on Large-Scale Exploits and Emergent Threats*, pages 1–9, 2008. 13

[39] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *International Conference on Machine Learning*, pages 16888–16905. PMLR, 2022. 2, 4, 5, 13

[40] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Zhiquan Wen, Yaofo Chen, Peilin Zhao, and Mingkui Tan. Towards stable test-time adaptation in dynamic wild world. In *International Conference on Learning Representations*, 2022. 1, 2, 3, 4, 5, 13

[41] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. *Distill*, 2(11):e7, 2017. 8

[42] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3234–3243, 2016. 8, 13

[43] Evgenia Rusak, Steffen Schneider, George Pachitariu, Luisa Eck, Peter Gehler, Oliver Bringmann, Wieland Brendel, and Matthias Bethge. If your data distribution shifts, use self-learning. *arXiv preprint arXiv:2104.12928*, 2021. 2

[44] Steffen Schneider, Evgenia Rusak, Luisa Eck, Oliver Bringmann, Wieland Brendel, and Matthias Bethge. Improving robustness against common corruptions by covariate shift adaptation. *Advances in Neural Information Processing Systems*, 33:11539–11551, 2020. 1, 13

[45] Ali Shafahi, W Ronny Huang, Mahyar Najibi, Octavian Suciu, Christoph Studer, Tudor Dumitras, and Tom Goldstein. Poison frogs! targeted clean-label poisoning attacks on neural networks. *Advances in Neural Information Processing Systems*, 31:6106–6116, 2018. 13

[46] Saurabh Singh and Abhinav Shrivastava. Evalnorm: Estimating batch normalization statistics for evaluation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3633–3641, 2019. 5

[47] Junha Song, Jungsoo Lee, In So Kweon, and Sungha Choi. Ecotta: Memory-efficient continual test-time adaptation via self-distilled regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11920–11929, 2023. 13

[48] Jacob Steinhardt, Pang Wei W Koh, and Percy S Liang. Certified defenses for data poisoning attacks. *Advances in Neural Information Processing Systems*, 30:3520–3532, 2017. 13

[49] Lili Su and Nitin H Vaidya. Fault-tolerant multi-agent optimization: optimal iterative distributed algorithms. In *Proceedings of the 2016 ACM Symposium on Principles of Distributed Computing*, pages 425–434, 2016. 14

[50] Cecilia Summers and Michael J Dinneen. Four things everyone should know to improve batch normalization. In *International Conference on Learning Representations*, 2019. 5

[51] Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *International Conference on Machine Learning*, pages 9229–9248. PMLR, 2020. 1, 2, 3

[52] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2020. 2, 3, 5, 13

[53] Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7201–7211, 2022. 1, 2, 13

[54] Tong Wu, Feiran Jia, Xiangyu Qi, Jiachen T Wang, Vikash Sehwag, Saeed Mahloujifar, and Prateek Mittal. Uncovering adversarial risks of test-time adaptation. In *International Conference on Machine Learning*, pages 37456–37495. PMLR, 2023. 1, 2, 3, 6, 12, 13

[55] Cong Xie, Oluwasanmi Koyejo, and Indranil Gupta. Generalized byzantine-tolerant sgd. *arXiv preprint arXiv:1802.10116*, 2018. 14

[56] Cong Xie, Oluwasanmi Koyejo, and Indranil Gupta. Fall of empires: Breaking byzantine-tolerant sgd by inner product manipulation. In *Uncertainty in Artificial Intelligence*, pages 261–270. PMLR, 2020. 14

[57] Li Yang, Adnan Siraj Rakin, and Deliang Fan. Rep-net: Efficient on-device learning via feature reprogramming. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12277–12286, 2022. 13

[58] Dong Yin, Yudong Chen, Ramchandran Kannan, and Peter Bartlett. Byzantine-robust distributed learning: Towards optimal statistical rates. In *International Conference on Machine Learning*, pages 5650–5659. PMLR, 2018. 14

[59] Longhui Yuan, Binhui Xie, and Shuang Li. Robust test-time adaptation in dynamic scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15922–15932, 2023. 1, 5

[60] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 818–833. Springer, 2014. 8

[61] Marvin Zhang, Sergey Levine, and Chelsea Finn. Memo: Test time robustness via adaptation and augmentation. *Advances in Neural Information Processing Systems*, 35:38629–38642, 2022. 13