Removing Batch Normalization Boosts Adversarial Training

Haotao Wang[1] **Aston Zhang**[2] **Shuai Zheng**[2] **Xingjian Shi**[2] **Mu Li**[2] **Zhangyang Wang**[1]

Abstract

Adversarial training (AT) defends deep neural networks against adversarial attacks. One challenge that limits its practical application is the performance degradation on clean samples. A major bottleneck identified by previous works is the widely used batch normalization (BN), which struggles to model the different statistics of clean and adversarial training samples in AT. Although the dominant approach is to extend BN to capture this mixture of distribution, we propose to completely eliminate this bottleneck by removing all BN layers in AT. Our **no**rmalizer-**f**ree **r**obust **t**raining (NoFrost) method extends recent advances in normalizer-free networks to AT for its unexplored advantage on handling the mixture distribution challenge. We show that NoFrost achieves adversarial robustness with only a minor sacrifice on clean sample accuracy. On ImageNet with ResNet50, NoFrost achieves 74.06% clean accuracy, which drops merely 2.00% from standard training. In contrast, BN-based AT obtains 59.28% clean accuracy, suffering a significant 16.78% drop from standard training. In addition, NoFrost achieves a 23.56% adversarial robustness against PGD attack, which improves the 13.57% robustness in BN-based AT. We observe better model smoothness and larger decision margins from NoFrost, which make the models less sensitive to input perturbations and thus more robust. Moreover, when incorporating more data augmentations into NoFrost, it achieves comprehensive robustness against multiple distribution shifts. Code and pretrained models are public[1].

Work done during the first author's internship at Amazon Web Services. [1]University of Texas at Austin, Austin, USA [2]Amazon Web Services, Santa Clara, USA. Correspondence to: Haotao Wang <htwang@utexas.edu>, Aston Zhang <astonz@amazon.com>, Zhangyang Wang <atlaswang@utexas.edu>.

[1]https://github.com/amazon-research/normalizer-free-robust-training

1. Introduction

Deep neural networks (DNNs) are vulnerable to adversarial attacks (Szegedy et al., 2013), which generate adversarial images by adding slight manipulations on the original images to falsify model predictions. One of the most effective methods to defend adversarial attacks is adversarial training (AT) (Goodfellow et al., 2015; Zhang et al., 2019). It jointly fits a model on clean (original) images and adversarial images to improve the model's adversarial robustness (i.e., the accuracy on adversarial images). An improved adversarial robustness often comes at the cost of reducing accuracy on clean samples (Madry et al., 2018; Tsipras et al., 2019; Ilyas et al., 2019). However, for many real-world applications, *high clean accuracy* is a basic requirement while adversarial robustness is a favorable bonus. It is hence desirable to sustain high clean accuracy while achieving high adversarial robustness.

Previous works (Xie & Yuille, 2020; Xie et al., 2020) pointed out that the widely used batch normalization (BN) layer (Ioffe & Szegedy, 2015) contributes to the undesirable trade-off between clean accuracy and adversarial robustness. In AT, clean and adversarial images are drawn from two different distributions. It is challenging for BN to capture those two different normalization statistics. Xie & Yuille (2020) proposed mixture BN (MBN) strategy for AT, which routes clean and adversarial images through two separate BN paths. MBN is adopted as the default option for many follow-up works (Xie & Yuille, 2020; Xie et al., 2020; Merchant et al., 2020; Li et al., 2020; Wang et al., 2020b; 2021). But as pointed out by the authors, it faces practical limitations: In practice there is no oracle to tell which BN path to choose during inference for each test sample.

We explore an alternative solution. Since BN has limited capacity to estimate normalization statistics of samples from heterogeneous distributions, can we better handle this mixture distribution challenge by removing all BN layers in AT? Replacing BN with other normalization layers that do not calculate statistics across samples, such as instance normalization (IN) (Ulyanov et al., 2016), brings a small amount of benefit, but the results are still unsatisfying as shown in our experiments (Table 3). Instead, we focus on recently proposed normalizer-free networks (Brock et al., 2021a;b). Although these networks were proposed to match

state-of-the-art accuracy on standard training with improved hardware performance and memory efficiency, we leverage them for their unexplored benefit to handle data from mixture distributions.

To this end, we propose the **no**rmalizer-**f**ree **ro**bust **t**raining (NoFrost) method to improve the trade-off between clean accuracy and adversarial robustness for AT. NoFrost is based on NF-ResNet (Brock et al., 2021a), a ResNet (He et al., 2016) variant without normalization layers but achieving an comparable accuracy on ImageNet (Deng et al., 2009). Experimental results show that NoFrost achieves a better accuracy-robustness trade-off compared with previous state-of-the-art AT methods based on BN or MBN models. For example, on ImageNet with the ResNet50 backbone, NoFrost achieves 11.96% higher robustness against APGD-CE attack (Croce & Hein, 2020) and 0.42% higher clean accuracy simultaneously compared with MBNAT (Xie & Yuille, 2020). NoFrost also achieves 12.15% higher clean accuracy and 7.25% higher robustness against APGD-CE attack simultaneously than TRADES-FAT (Zhang et al., 2020). To explain the effectiveness of NoFrost, we demonstrate that NoFrost has better model smoothness (Zhang et al., 2019) and larger decision margins (Yang et al., 2020).

Moreover, we show that NoFrost can be generalized towards *comprehensive robustness* against distribution shifts beyond adversarial samples. In particular, we jointly train models on images generated by adversarial attacks and two other robust data augmentation methods, namely DeepAugment (Hendrycks et al., 2021) and texture-debiased augmentation (TDA) (Hermann et al., 2020). This extended version of NoFrost simultaneously achieves better or comparable adversarial robustness and accuracy on multiple out-of-distribution (OOD) benchmark datasets (i.e., OOD robustness), such as ImageNet-C (Hendrycks & Dietterich, 2019), Imagenet-R (Hendrycks et al., 2021), and ImageNet-Sketch (Wang et al., 2019), compared with previous state-of-the-art robust learning methods including DeepAugment and texture-debiased augmentation.

In summary, our contributions are as follows:

1. We propose NoFrost to improve the clean accuracy and adversarial robustness trade-off in AT. Our approach is simple and straightforward: just removing all BN layers to address the mixture distribution challenge.

2. To the best of our knowledge, we for the first time apply normalizer-free models to AT. We demonstrate the unexplored advantage of normalizer-free models in handling data from mixture distributions.

3. We show that NoFrost achieves substantially better accuracy-robustness trade-off on ImageNet. Using NF-ResNet50, NoFrost achieves 74.06% clean accuracy,

dropping merely 2.00% from standard training, plus 23.56% adversarial robustness against PGD attack. In comparison, BN-based AT obtains only 59.28% clean accuracy with 13.57% adversarial robustness using ResNet50.

4. We demonstrate that when combining adversarial samples with other data augmentation methods, NoFrost can simultaneously achieve adversarial robustness and out-of-distribution robustness.

2. Preliminary

Before diving deep into the hidden benefit on model robustness brought by normalizer-free training, we need to describe the concept of model robustness that includes both adversarial robustness and OOD robustness (Section 2.1). Then we will revisit the mixture distribution challenge (Section 2.2) and normalizer-free networks (Section 2.3).

2.1. Model Robustness

Model robustness refers to a model's performance under various data distribution shifts. Here we review two distribution shifts, adversarial examples and out-of-distribution examples, which are related to our work, and methods for improving model robustness.

Adversarial robustness refers to a model's performance on adversarial samples. These adversarial samples are modified from the original (clean) samples by adversarial attacks (Szegedy et al., 2013; Madry et al., 2018; Carlini & Wagner, 2017; Xiao et al., 2018; Croce & Hein, 2020) to falsify the model. To defend these attacks, one of the most effective defense methods is adversarial training (AT) (Goodfellow et al., 2015; Madry et al., 2018; Zhang et al., 2019; 2020; Xie & Yuille, 2020). AT trains a model on both clean and adversarial images. Given a pair of clean image $\boldsymbol{x}$ and its label y sampled from the data distribution $\mathcal{D}$, AT learns a robust classifier $f_{\boldsymbol{\theta}}$ with parameters $\boldsymbol{\theta}$ by

$$\min_{\boldsymbol{\theta}} \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}} (1-\lambda)\mathcal{L}(f_{\boldsymbol{\theta}}(\boldsymbol{x}),y) + \lambda\mathcal{L}(f_{\boldsymbol{\theta}}(\boldsymbol{x}^*),y), \quad (1)$$

where $\mathcal{L}(\cdot,\cdot)$ is the cross-entropy loss function. The adversarial image $\boldsymbol{x}^*$ in Equation 1 is generated from $\boldsymbol{x}$ by PGD attack (Madry et al., 2018):

$$\boldsymbol{x}^{(0)} = \text{RandomSample}(\mathcal{B}(\boldsymbol{x},\epsilon)),$$
$$\boldsymbol{x}^{(t+1)} = \Pi_{\mathcal{B}(\boldsymbol{x},\epsilon)}(\boldsymbol{x}^{(t)} + \alpha \cdot \text{sign}(\nabla_{\boldsymbol{x}^{(t)}}\mathcal{L}(f_{\boldsymbol{\theta}}(\boldsymbol{x}^{(t)}),y))),$$
$$\boldsymbol{x}^* = \boldsymbol{x}^{(T)},$$

where $\mathcal{B}(\boldsymbol{x},\epsilon)$ is the ℓ_∞ ball with radius ϵ around $\boldsymbol{x}$, the initialization $\boldsymbol{x}^{(0)}$ is randomly sampled from $\mathcal{B}(\boldsymbol{x},\epsilon)$, $\Pi_{\mathcal{B}(\boldsymbol{x},\epsilon)}$ means the nearest projection to $\mathcal{B}(\boldsymbol{x},\epsilon)$, T is the total number of iterations, and α is the step size. The hyper-parameter

λ controls the weight of the loss on adversarial samples. When $\lambda = 0$ we obtain standard training. When $\lambda = 1$ we obtain PGDAT (Madry et al., 2018). Some previous works set $\lambda = 0.5$ (Kurakin et al., 2018; Wang et al., 2020b) to trade off clean accuracy and adversarial robustness, which we denote as standard adversarial training (SAT). Other works improve the above training approach. For example, TRADES (Zhang et al., 2019) simultaneously optimizes classification error and model smoothness. FAT (Zhang et al., 2020) uses early-stopped PGD attack to generate "friendly" adversarial samples, which can also be combined with TRADES (termed as TRADES-FAT).

Out-of-distribution robustness refers to the model's performance on out-of-distribution (OOD) examples. To evaluate OOD robustness, there exists multiple benchmark datasets. ImageNet-C (Hendrycks & Dietterich, 2019) adds natural corruptions such as Gaussian noise and motion blur on the ImageNet validation set. ImageNet-Sketch (Wang et al., 2019) contains sketch-like images to evaluate cross-domain transferability. ImageNet-A (Hendrycks et al., 2019) contains naturally occurring real-world images which falsify state-of-the-art image classifiers. ImageNet-R (Hendrycks et al., 2021) contains renditions such as painting and sculpture. To improve OOD robustness, a popular approach is through data augmentations (Zhong et al., 2017; Cubuk et al., 2018; Geirhos et al., 2019; Yun et al., 2019; Wang et al., 2019; Hendrycks et al., 2021; Gong et al., 2021; Wang et al., 2021). For example, DeepAugment (Hendrycks et al., 2021) first adds random noise onto the weights of an image-to-image model (e.g., an image super-resolution model). It then feeds clean images to the noisy image-to-image model and uses the output images as augmented data. In this way, DeepAugment obtains diverse augmented images and thus achieves state-of-the-art robustness on ImageNet-C and ImageNet-R. Texture-debiased augmentation (TDA) (Hermann et al., 2020) stacks color distortion, less aggressive random crops, and other simple augmentations to debias a model towards textures and is shown to improve model generalizability (e.g., from ImageNet to ImageNet-Sketch).

2.2. The Mixture Distribution Challenge

Xie & Yuille (2020) have shown that solving Equation 1 with traditional BN-based networks leads to unsatisfying trade-off between clean accuracy and adversarial robustness. The underlying reason is that clean and adversarial images are sampled from different distributions. It is difficult for BN to estimate the correct normalization statistics of such mixture of distributions. We show the misalignment between clean and adversarial distributions in Figure 1.[2] Specifically, we train two ResNet26 models on ImageNet by only using clean

---

[2]Similar observations were first made in (Xie & Yuille, 2020). We show them here for a more self-contained introduction.
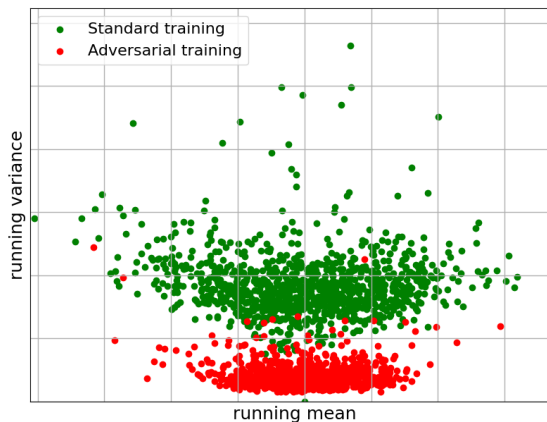

Figure 1. The mixture distribution challenge in adversarial training. Specifically, we show the channel-wise BN statistics of the 15-th layer in ResNet26 models obtained by standard training and adversarial training, respectively. Each dot represents the running mean and variance of a channel in the BN layer. We can see that clean and adversarial training samples have different feature statistics, and thus are sampled from different underlying distributions. Similar observations were first made in (Xie & Yuille, 2020).

images (i.e., $\lambda = 0$ in Equation 1) and adversarial images (i.e., $\lambda = 1$), respectively. We then plot the channel-wise BN statistics of the 15-th layer (other layers are similar) in both models. As we can see, the running means and variances of clean (the green dots) and adversarial images (the red dots) are significantly different.

To solve this mixture distribution challenge, Xie & Yuille (2020) proposed MBNAT. It uses a mixture BN (MBN) strategy to disentangle clean and adversarial statistics. Specifically, it uses two parallel BNs in each normalization layer, denoted by BN_c and BN_a. During training, clean (adversarial) images are routed to BN_c (BN_a). As a result, BN_c (BN_a) only estimates the distribution of clean (adversarial) images to avoid modeling a mixture distribution.

During inference, we should use BN_c (BN_a) for a clean (adversarial) test image. There is, however, no oracle to tell us whether a test image is clean or adversarial. As shown by Xie & Yuille (2020); Xie et al. (2020), it is difficult to choose which path in practise: If BN_c is used, the model will have a good clean accuracy while sacrificing robustness, and vice versa (see Appendix A for details).

2.3. Normalizer-Free Networks

Batch normalization (BN) is originally proposed as a regularization to enable stable training of DNNs (Ioffe & Szegedy, 2015), and is then adopted as a basic building block of DNNs. Research on normalizer-free networks aims to remove BN from DNNs for better hardware efficiency. The first attempt to train normalizer-free (NF) deep residual networks uses stable weight initialization methods (Zhang et al.,

2018; De & Smith, 2020; Bachlechner et al., 2020). For example, SkipInit initializes residual blocks in NF networks close to identity mappings, ensuring signal propagation and well-behaved gradients (De & Smith, 2020). Although these initialization methods enable stable training of NF deep residual networks, the obtained test accuracy is still lower than that of well-tuned normalized models. More recently, Brock et al. (2021a) first obtained NF networks with performance competitive with traditional BN-based ResNets (He et al., 2016) and EfficientNets (Tan & Le, 2019). The authors proposed scaled weight standardization which normalizes the weights in each layer to prevent mean shift in hidden activations and thus stabled the training. Given the weight matrix W of a convolutional or fully connected layer, the proposed scaled weight standardization takes the following form:

$$\hat{W}_{i,j} = \gamma \frac{W_{i,j} - \mu_i}{\sigma_i \sqrt{N}},$$

where μ_i and σ_i are the mean and standard deviation of the i-th row of W, γ is a fixed number, and N is the batch size. This constraint is imposed throughout training as a differentiable operation in the forward pass. Brock et al. (2021b) further proposed an adaptive gradient clipping method, which enables normalizer-free models to train with large batch sizes and strong data augmentations for better test accuracy.

3. Method

3.1. NoFrost for Adversarial Training

We adopt a simple strategy to address the mixture distribution challenge in AT. Since this challenge arises from the limited capability of BN to simultaneously encode the heterogeneous distributions of clean and adversarial samples, we simply remove all BN layers in the model. Specifically, our normalizer-free robust training (NoFrost) method, solves the AT problem in Equation 1 by using NF networks (Brock et al., 2021a) as the model f_θ. We use PGD attack (Madry et al., 2018) to generate adversarial samples in NoFrost, and set $\lambda = 0.5$ for simplicity[3]. Without BN layers, NoFrost naturally overcomes the limitation of inference-time oracle BN selection in MBNAT described in Section 2.2.

3.2. NoFrost* for Comprehensive Robustness

To achieve the more challenging *comprehensive robustness* (i.e., to be simultaneously robust against multiple adversarial attacks and naturally occurring distribution shifts), we further generalize NoFrost by combining it with other robust data augmentation methods. Different robust data augmentations have different strength and weakness: the best method

---

[3]Searching for the optimal λ value may lead to better performance, which we leave for future work.

against one type of distribution shift may not be the best against another. For example, AT uses adversarial attack as a data augmentation method that augments clean images by adding worst-case additive noises, and achieves state-of-the-art robustness against adversarial attacks; however, AT cannot achieve state-of-the-art robustness against natural distribution shifts (e.g., in ImageNet-R and ImageNet-Sketch). Similarly, DeepAugment is the state-of-the-art data augmentation method against distribution shifts caused by different renditions (e.g., art, cartoons, and graffiti in ImageNet-R), but has little benefit against adversarial attacks.

As a result, a naive way to achieve comprehensive robustness against multiple types of distribution shifts is to combine multiple robust data augmentations during training. However, jointly training on multiple different data augmentations may cause an even harder mixture distribution challenge than adversarial training that uses only one augmentation (i.e., adversarial attack).

In view of this, we extend NoFrost by incorporating more data augmentation methods to achieve comprehensive robustness. By default, we add two new robust data augmentation methods, namely DeepAugment and texture-debiased augmentation (TDA), for the extended version of NoFrost (denoted as NoFrost*). Formally, the optimization problem of NoFrost* is:

$$\min_{\theta} \mathbb{E}_{(x,y)\sim\mathcal{D}} (\mathcal{L}(f_\theta(x),y) + \\ \mathcal{L}(f_\theta(\hat{x}),y) + \mathcal{L}(f_\theta(x^*),y))/3, \tag{2}$$

where $\hat{x}$ is the augmented image generated from x using either DeepAugment or TDA (each with half probability), x^* is the adversarial image generated from x, and f_θ is a normalizer-free network. Other notations have the same meaning as in Equation 1.

4. Experiments

In this section, we first describe the general experimental settings (Section 4.1). Then we show that NoFrost outperforms previous adversarial training methods (Section 4.2) and achieves nicer properties such as better model smoothness and larger decision margins (Section 4.3). Finally, we evaluate NoFrost* for comprehensive robustness against multiple distribution shifts (Section 4.4).

4.1. Experimental Settings

Datasets, models, and metrics All methods are trained on the ImageNet (Deng et al., 2009) dataset. We use ResNet26 and ResNet50 (He et al., 2016) backbones, with different normalization strategies: BN (Ioffe & Szegedy, 2015), MBN (Xie & Yuille, 2020), and NF (Brock et al., 2021a). We evaluate *clean accuracy* on the ImageNet validation set, and use the accuracy on adversarial test images as a metric for *adver-
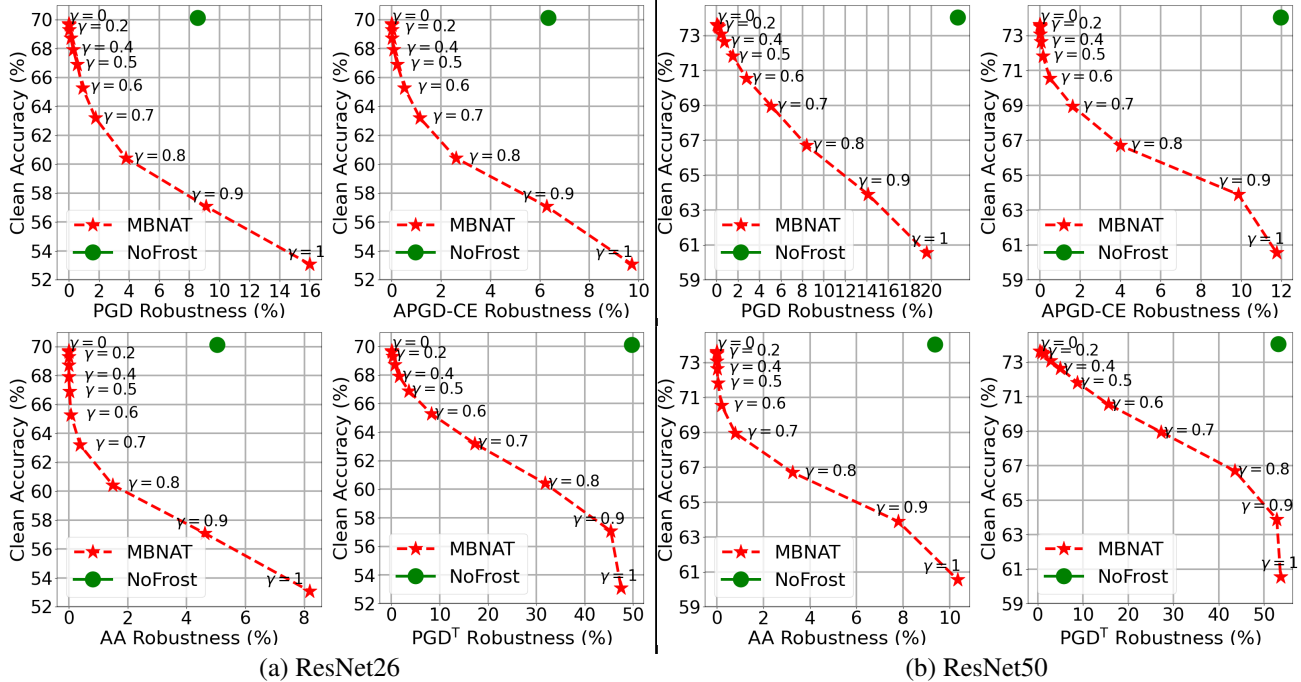
Figure 2. Trade-off between robustness and accuracy on ResNet26 (sub-figure (a)) and ResNet50 (sub-figure (b)) trained by MBNAT and NoFrost. Adversarial robustness is evaluated on PGD (the top-left panel in each sub-figure), APGD-CE (the top-right panel in each sub-figure), AutoAttack (AA, the bottom-left panel in each sub-figure) and targeted PGD (denoted as PGD^T; the bottom-right panel in each sub-figure) attacks. γ is the weight for interpolation between the two BNs in MBNAT.

sarial robustness. We generate adversarial images on the ImageNet validation set using white-box attacks (PGD (Madry et al., 2018), APGD-CE (Croce & Hein, 2020), APGD-DLR (Croce & Hein, 2020), MIA (Dong et al., 2018), CW (Carlini & Wagner, 2017)[4]), black-box attacks (RayS (Chen & Gu, 2020), and Square (Andriushchenko et al., 2020)) and the AutoAttack (AA) (Croce & Hein, 2020). We evaluate *OOD robustness* against naturally occurring distribution shifts by measuring accuracy on ImageNet-C (Hendrycks & Dietterich, 2019), ImageNet-R (Hendrycks et al., 2021), and ImageNet-Sketch (Wang et al., 2019).

General hyper-parameters For all experiments, we train on ImageNet for 90 epochs. We use the SGD optimizer with momentum 0.9. Batch size is 256. Weight decay factor is 5×10^{-5}. The initial learning rate is 0.1 and decays following a cosine annealing scheduler. All experiments are conducted with 8 NVIDIA V100 GPUs.

Implementation of adversarial training We study adversarial robustness under perturbation magnitude $\epsilon = 8$ (on scale of 0 to 255 in unsigned 8-bit pixels). We set the maximum PGD attack iteration number $T = 10$ during training for all adversarial training methods. For TRADES and TRADES-FAT, we set the loss trade-off hyper-parameter

---

[4]We use the ℓ_∞ version of CW attack following (Zhang et al., 2020).

to 1 following the original TRADES paper (Zhang et al., 2019). For FAT and TRADES-FAT, we set the PGD early-stop iteration to 1 following the original FAT paper (Zhang et al., 2020).

Details for robustness evaluation Following (Zhang et al., 2019; 2020), we set attack iterations to be 20 for all white-box attacks. For all black-box attacks, we allow 400 queries per-sample on all compared models. For AutoAttack, we use the fast version with APGD-CE and APGD-DLR attacks. For RayS, we evaluate on a subset of ImageNet validation set with 1000 images due to high computational costs.

4.2. Adversarial Training Results

We first compare NoFrost with MBNAT, the *de facto* solution for resolving the AT mixture distribution challenge (Xie & Yuille, 2020; Xie et al., 2020; Merchant et al., 2020; Li et al., 2020; Wang et al., 2020b; 2021; 2022). As discussed in Section 2.2, MBN requires an empirical weighting value γ to be set for interpolation between BN_c and BN_a during inference (see Appendix A for more details). The original MBNAT paper (Xie & Yuille, 2020) uses $\gamma = 1$ to pursue the best adversarial robustness. In another work by the same first author (Xie et al., 2020), $\gamma = 0$ is applied for the best clean accuracy. We uniformly sample γ from interval

Table 1. Adversarial robustness of ResNet26 under perturbation magnitude $\epsilon = 8$. Classification accuracy on clean images and under different adversarial attacks are reported. The best and second to the best numbers are shown in bold and underlined, respectively.

Method	Clean	White-box Attacks					Black-box Attacks		AA
		PGD	APGD-CE	APGD-DLR	MIA	CW	RayS	Square	
ST	**72.68**	0.01	0.00	0.00	0.00	0.00	18.2	27.5	0.00
SAT	52.65	10.55	5.02	5.30	8.84	9.18	30.5	44.7	3.78
TRADES	39.64	9.94	<u>6.24</u>	4.02	8.33	6.37	20.7	32.8	3.54
FAT	58.72	6.97	2.35	2.68	6.59	6.37	<u>33.6</u>	**50.8**	1.70
TRADES-FAT	55.65	<u>11.91</u>	5.79	<u>6.14</u>	<u>10.83</u>	**10.81**	31.1	46.7	<u>4.63</u>
NoFrost	<u>70.13</u>	**12.24**	**6.34**	**6.60**	**21.83**	<u>10.18</u>	**34.5**	<u>48.3</u>	**5.04**

Table 2. Adversarial robustness of ResNet50 under perturbation magnitude $\epsilon = 8$. Classification accuracy on clean images and under different adversarial attacks are reported. The best and second to the best numbers are shown in bold and underlined, respectively.

Method	Clean	White-box attacks					Black-box attacks		AA
		PGD	APGD-CE	APGD-DLR	MIA	CW	RayS	Square	
ST	**76.06**	0.04	0.00	0.00	0.00	0.00	22.5	31.4	0.00
SAT	59.28	13.57	7.80	<u>8.46</u>	10.28	11.02	27.4	40.2	6.23
TRADES	49.25	<u>14.80</u>	<u>9.20</u>	8.19	<u>12.97</u>	11.80	32.6	39.5	<u>6.66</u>
FAT	58.94	12.45	5.48	7.16	12.56	<u>12.24</u>	<u>35.9</u>	**51.4**	4.73
TRADES-FAT	60.52	11.67	4.71	5.90	11.28	10.29	34.5	<u>48.6</u>	3.87
NoFrost	<u>74.06</u>	**22.45**	**11.96**	**13.37**	**36.11**	**19.17**	**36.1**	43.1	**9.36**

$[0, 1]$ to obtain the robustness-accuracy Parato frontier of MBNAT.

Comparison results between NoFrost and MBNAT on ResNet26 and ResNet50 are shown in Figure 2. A point closer to the top-right corner represents a more desired model with higher clean accuracy and adversarial robustness. For MBNAT models, as the value of γ increases from 0 to 1, the influence of BN_a gradually outweighs that of BN_c (see Appendix A for more details). As a result, the adversarial robustness increases while the clean accuracy sharply drops. In contrast, NoFrost simultaneously achieves decent clean accuracy and adversarial robustness. In other words, NoFrost achieves a much more desired trade-off between clean accuracy and adversarial robustness compared with MBNAT. For example, on ResNet26, NoFrost achieves 70.13% accuracy and 6.34% robustness against the APGD-CE attack. To achieve comparable accuracy, MBNAT needs to set $\gamma = 0$ which leads to 0 robustness against APGD-CE attack (6.34% less than NoFrost) and 69.71% accuracy (0.42% less than NoFrost). On the other hand, to achieve comparable robustness with NoFrost, MBNAT needs to set $\gamma = 0.9$ which leads to 57.08% accuracy (13.05% less than NoFrost) and 6.27% robustness (0.07% less than NoFrost).

We further compare NoFrost with other adversarial training methods, including TRADES, FAT, and TRADES-FAT, on ImageNet. The results on ResNet26 and ResNet50 are

Table 3. Standard adversarial training (SAT) with IN-based networks yields worse robustness than NoFrost. Experiments conducted on ResNet26 with different normalizers.

	Clean	PGD
SAT w/ BN	52.65	10.55
SAT w/ IN	<u>56.78</u>	<u>11.06</u>
NoFrost	**70.13**	**12.24**

shown in Table 1 and 2, respectively. NoFrost achieves significantly higher accuracy on clean images and better or comparable robustness against different attacks, compared with all those adversarial training methods. For example, on ResNet26, NoFrost outperforms TRADES-FAT by 14.48% on clean accuracy, and 0.55% against APGD-CE attack.

Since the mixture of distribution challenge is mainly caused by the limited capability of single BN layers to encode the mixture distribution of clean and adversarial samples, another possible solution is to replace BN with instance-level normalization layers, such as instance normalization (IN) (Ulyanov et al., 2016). We denote the method of replacing BN with IN in SAT as "SAT w/ IN". The results are shown in Table 3. Both SAT w/ IN and NoFrost achieve better accuracy on clean images and robustness against PGD attack, compared with the naive BN counterpart (i.e., SAT). This is intuitive since both methods are reasonable solutions for

Table 4. Mean and standard deviation of NoFrost using ResNet26 with three different random seeds.

	Clean	PGD	APGD-CE	APGD-DLR	AA
NoFrost	70.15 ± 0.03	12.19 ± 0.10	6.34 ± 0.04	6.57 ± 0.06	5.01 ± 0.09

the mixture of distribution problem in adversarial training. However, NoFrost achieves considerably better performance than SAT w/ IN, with 13.35% higher accuracy and 1.18% higher robustness against PGD attack.

Stability analyses In Table 4, we show the stability analysis results on NoFrost over the randomness in the algorithm (e.g., random initialization, random batch sampling). Specifically, we run NoFrost on ResNet26 with three different random seeds, and report the mean (denoted as μ) and standard deviation (denoted as σ) of the testing results on those three models in the form of $\mu \pm \sigma$ in Table 4. We report accuracy on both clean and adversarial images generated by different attacks. As we can see, NoFrost has stable performance with small standard derivations on both clean accuracy and adversarial robustness.

4.3. NoFrost Leads to More Robust Model Properties

In this section, we show that NoFrost models have stronger model smoothness, larger decision margins (Wang et al., 2020c; Kim et al., 2021), and boundary thickness (Yang et al., 2020). All these have been shown to benefit model robustness (Sanyal et al., 2020; Wang et al., 2020c; Yang et al., 2020). In the following, we provide definitions for these properties and empirically show how they are influenced by removing normalization layers in adversarial training.

- Decision margin: Following (Kim et al., 2021), we define $M(\boldsymbol{x}) = \boldsymbol{p}(\boldsymbol{x})_y - \max_{i \neq y} \boldsymbol{p}(\boldsymbol{x})_i$ as the decision margin for a sample pair $(\boldsymbol{x}, y)$, where $\boldsymbol{p}(\boldsymbol{x})$ is the softmax probability of sample $\boldsymbol{x}$. $M(\boldsymbol{x}) < 0$ indicates a wrong prediction on sample $\boldsymbol{x}$.
- Boundary thickness: Following (Yang et al., 2020), the boundary thickness of date $\boldsymbol{x}$ is defined as $T(\boldsymbol{x}) = \|\boldsymbol{x} - \boldsymbol{x}^*\|_2 \int_0^1 \mathbb{I}\{\alpha < g_{ij}(t\boldsymbol{x} + (1-t)\boldsymbol{x}^*) < \beta\}dt$, where $g_{ij}(\cdot) = \boldsymbol{p}(\cdot)_i - \boldsymbol{p}(\cdot)_j$, i and j are the predicted labels of $\boldsymbol{x}$ and $\boldsymbol{x}^*$ respectively, and $\mathbb{I}\{\cdot\}$ is the indicator function. It measures the distance between two level sets $g_{ij}(\cdot) = \alpha$ and $g_{ij}(\cdot) = \beta$ along the adversarial direction. We set $\alpha = 0, \beta = 0.75$ and solve $\boldsymbol{x}^*$ via a targeted 20-step PGD attack following the original paper (Yang et al., 2020).
- Model smoothness: Following (Zhang et al., 2019; Kim et al., 2021), we use the KL divergence $D(\boldsymbol{x}) = \mathrm{KL}(\boldsymbol{p}(\boldsymbol{x})\|\boldsymbol{p}(\boldsymbol{x}^*))$ as a measurement for model smoothness for sample $\boldsymbol{x}$, where $\boldsymbol{x}^*$ is an adversarial image generated from $\boldsymbol{x}$. A smaller $D(\boldsymbol{x})$ indicates a stronger local model smoothness at $\boldsymbol{x}$.

Table 5. Decision margin, boundary thickness, and model smoothness of adversarially trained (under $\epsilon = 8$) ResNet26 models with different normalization strategies. The best and second-best values are bolded and underlined, respectively.

Normalization strategy (Method)	Decision margin $M(\boldsymbol{x})(\uparrow)$	Boundary thickness $T(\boldsymbol{x})(\uparrow)$	Model smoothness $D(\boldsymbol{x})(\downarrow)$
BN (SAT)	<u>0.3241</u>	<u>17.51</u>	4.927
MBN (MBNAT)	0.3143	13.78	**1.119**
NF (NoFrost)	**0.4700**	**31.49**	<u>2.996</u>

We measure the above metrics on the 500 validation images from the first 10 classes on ImageNet. We report their mean values over the 500 images in Table 5 and also show the distribution of those metrics using histograms in Figure 6 (in Appendix B.1). Compared with SAT, NoFrost leads to larger decision margins, thicker boundaries, and stronger model smoothness. All these three properties are beneficial for model robustness (Moosavi-Dezfooli et al., 2019; Sanyal et al., 2020; Wang et al., 2020c; Yang et al., 2020).

Another interesting observation is that, compared with SAT, MBNAT improves model smoothness while leaving the decision margin and boundary thickness almost unchanged. In contrast, NoFrost improves all three properties over SAT. This is consistent with the recent finding that different defense methods improve robustness through different underlying mechanisms (Kim et al., 2021). Our findings suggest that MBNAT improves model robustness mainly through improving model smoothness, while NoFrost improves robustness by simultaneously improving all three properties.

4.4. Comprehensive Robustness

Now we evaluate NoFrost* (subsection 3.2) for comprehensive robustness on three robustness benchmark datasets (ImageNet-C, ImageNet-R, and ImageNet-Sketch), two adversarial attacks (PGD and Square), together with clean accuracy on the ImageNet validation set. Since NoFrost* jointly fits clean, adversarial, DeepAugment, and TDA samples, we compare it with the four stand-alone methods: Standard training (training with only clean images), SAT (training with both clean and adversarial images), DeepAugment, and TDA. We also include the naive combination of the four methods (i.e., jointly training on clean, adversarial, DeepAugment, and TDA samples on a traditional BN network) as a baseline, which is denoted as "Combine". All methods are trained using the same settings in Section 4.1. Results are shown in Figure 3 and 4.

Notably, the naive combination performs the worst in most cases. This shows the inferent difficulty in fitting multiple heterogeneous augmentations under the traditional BN. In contrast, equipped with the new normalizer-free strategy,
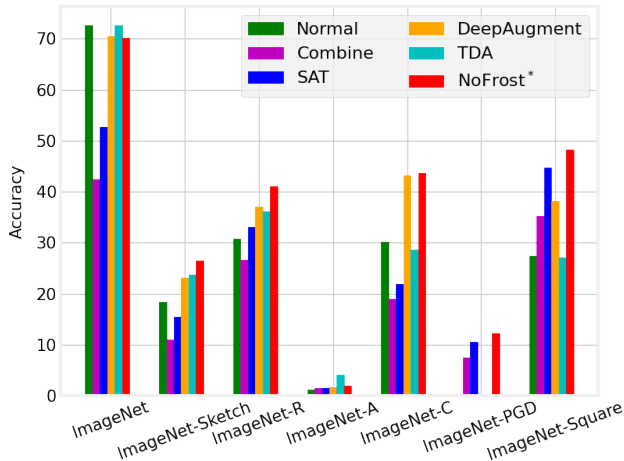
Figure 3. Model performance (accuracy in percentage) on different benchmark datasets or adversarial attacks. All methods are trained on ImageNet with ResNet26.

Figure 4. Model performance (accuracy in percentage) on different benchmark datasets or adversarial attacks. All methods are trained on ImageNet with ResNet50.

NoFrost* successfully fits all data augmentations within a single model and achieves comprehensive robustness. On ResNet26, NoFrost* achieves the best robustness on all evaluated OOD benchmark datasets and adversarial attacks. On ResNet50, although NoFrost* achieves slightly worse (3.06% less) robustness on ImageNet-C than Deep-Augment, it outperforms all baseline methods on other OOD benchmark datasets and adversarial attacks. For example, NoFrost* achieves 16.30%, 8.20%, and 2.05% higher robustness than DeepAugment on PGD attack, Square attack, and ImageNet-R, respectively.

5. Discussions

Apart from this work and the MBN papers (Xie & Yuille, 2020; Xie et al., 2020), there are other related works studying how BN affects model robustness. Benz et al. (2021a); Schneider et al. (2020) proposed to improve model robustness against natural image corruptions (e.g., random Gaussian noise and motion blurring) by unsupervised model adaptation. Specifically, they replace the BN statistic calculated on clean training images with those on unlabeled corrupted images. AdvBN (Shu et al., 2021) added adversarial perturbations on the BN statistics to increase model robustness against unseen distribution shifts such as style variations and image corruptions. Galloway et al. (2019) and Benz et al. (2021b) observed that, in *standard training*, BN grants models with better clean accuracy but harms their adversarial robustness. In contrast to their work, our paper utilizes normalizer-free networks to solve the mixture distribution challenge and improve the trade-off between clean accuracy and adversarial robustness in *adversarial training*. More related works on machine learning robustness can be found in a recent survey paper (Mohseni et al., 2021).
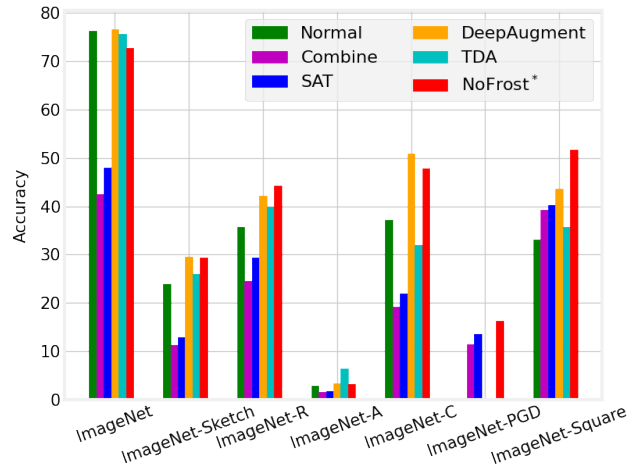
Our paper shows that removing BN can significantly boost adversarial training. Yet, some existing test-time adaptation methods utilize the existence of BN to improve model robustness (Wang et al., 2020a; Nandy et al., 2021; Awais et al., 2020; Benz et al., 2021a). Those methods are not directly applicable on normalizer-free networks, and thus NoFrost cannot be directly combined with those existing test-time adaptation methods for potentially further improved robustness. It will be our future work to study how to efficiently enable test-time adaption upon NoFrost, potentially by designing new test-time adaptation methods tailored for NF networks. On the other hand, NoFrost can potentially benefit from the future improvements in both fields of normalizer-free networks and adversarial training.

6. Conclusion

In this paper, we address the issue of significant degradation on clean accuracy in adversarial training. The proposed NoFrost method removes all BNs in AT. NoFrost achieves a significantly more favorable trade-off between clean accuracy and adversarial robustness compared with previous BN-based AT methods: It achieves decent adversarial robustness with only minor degradation on clean accuracy. It is further generalized to achieve the more challenging goal of comprehensive robustness. We hope this study could be a stepping stone towards the exploration of normalizer-free training in improving model robustness and other fields with the challenge of data heterogeneity, such as distributed learning and domain generalization.

Acknowledgement

(IOBT REIGN) and an Amazon Research Award.

References

Andriushchenko, M., Croce, F., Flammarion, N., and Hein, M. Square attack: a query-efficient black-box adversarial attack via random search. In *European Conference on Computer Vision (ECCV)*, pp. 484–501, 2020.

Awais, M., Shamshad, F., and Bae, S.-H. Towards an adversarially robust normalization approach. *arXiv preprint arXiv:2006.11007*, 2020.

Bachlechner, T., Majumder, B. P., Mao, H. H., Cottrell, G. W., and McAuley, J. ReZero is all you need: Fast convergence at large depth. *arXiv preprint arXiv:2003.04887*, 2020.

Benz, P., Zhang, C., Karjauv, A., and Kweon, I. S. Revisiting batch normalization for improving corruption robustness. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 494–503, 2021a.

Benz, P., Zhang, C., and Kweon, I. S. Batch normalization increases adversarial vulnerability and decreases adversarial transferability: A non-robust feature perspective. In *IEEE International Conference on Computer Vision (ICCV)*, pp. 7818–7827, 2021b.

Brock, A., De, S., and Smith, S. L. Characterizing signal propagation to close the performance gap in unnormalized ResNets. In *International Conference on Learning Representations (ICLR)*, 2021a.

Brock, A., De, S., Smith, S. L., and Simonyan, K. High-performance large-scale image recognition without normalization. *arXiv preprint arXiv:2102.06171*, 2021b.

Carlini, N. and Wagner, D. Towards evaluating the robustness of neural networks. In *IEEE Symposium on Security and Privacy (SP)*, pp. 39–57, 2017.

Chen, J. and Gu, Q. RayS: A ray searching method for hard-label adversarial attack. In *International Conference on Knowledge Discovery and Data Mining (KDD)*, 2020.

Croce, F. and Hein, M. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International Conference on Machine Learning (ICML)*, pp. 2206–2216, 2020.

Cubuk, E. D., Zoph, B., Mane, D., Vasudevan, V., and Le, Q. V. AutoAugment: Learning augmentation policies from data. *arXiv preprint arXiv:1805.09501*, 2018.

De, S. and Smith, S. Batch normalization biases residual blocks towards the identity function in deep networks. *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 248–255, 2009.

Dong, Y., Liao, F., Pang, T., Su, H., Zhu, J., Hu, X., and Li, J. Boosting adversarial attacks with momentum. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9185–9193, 2018.

Galloway, A., Golubeva, A., Tanay, T., Moussa, M., and Taylor, G. W. Batch normalization is a cause of adversarial vulnerability. *arXiv preprint arXiv:1905.02161*, 2019.

Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., and Brendel, W. Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations (ICLR)*, 2019.

Gong, C., Ren, T., Ye, M., and Liu, Q. MaxUp: A simple way to improve generalization of neural network training. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations (ICLR)*, 2015.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

Hendrycks, D. and Dietterich, T. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations (ICLR)*, 2019.

Hendrycks, D., Zhao, K., Basart, S., Steinhardt, J., and Song, D. Natural adversarial examples. *arXiv preprint arXiv:1907.07174*, 2019.

Hendrycks, D., Basart, S., Mu, N., Kadavath, S., Wang, F., Dorundo, E., Desai, R., Zhu, T., Parajuli, S., Guo, M., Song, D., Steinhardt, J., and Gilmer, J. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *IEEE International Conference on Computer Vision (ICCV)*, 2021.

Hermann, K., Chen, T., and Kornblith, S. The origins and prevalence of texture bias in convolutional neural networks. *Advances in Neural Information Processing Systems (NeurIPS)*, 33, 2020.

Ilyas, A., Santurkar, S., Tsipras, D., Engstrom, L., Tran, B., and Madry, A. Adversarial examples are not bugs,

they are features. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Ioffe, S. and Szegedy, C. Batch Normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning (ICML)*, pp. 448–456, 2015.

Kim, H., Lee, W., Lee, S., and Lee, J. Bridged adversarial training. *arXiv preprint arXiv:2108.11135*, 2021.

Kurakin, A., Goodfellow, I. J., and Bengio, S. Adversarial examples in the physical world. In *Artificial Intelligence Safety and Security*, pp. 99–112. 2018.

Li, Y., Yu, Q., Tan, M., Mei, J., Tang, P., Shen, W., Yuille, A., and Xie, C. Shape-texture debiased neural network training. *arXiv preprint arXiv:2010.05981*, 2020.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations (ICLR)*, 2018.

Merchant, A., Zoph, B., and Cubuk, E. D. Does data augmentation benefit from split batchnorms. *arXiv preprint arXiv:2010.07810*, 2020.

Mohseni, S., Wang, H., Yu, Z., Xiao, C., Wang, Z., and Yadawa, J. Practical machine learning safety: A survey and primer. *arXiv preprint arXiv:2106.04823*, 2021.

Moosavi-Dezfooli, S.-M., Fawzi, A., Uesato, J., and Frossard, P. Robustness via curvature regularization, and vice versa. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9078–9086, 2019.

Nandy, J., Saha, S., Hsu, W., Mong, L., and Zhu, X. X. Covariate shift adaptation for adversarially robust classifier. In *International Conference on Learning Representations Workshop (ICLRW)*, 2021.

Sanyal, A., Dokania, P. K., Kanade, V., and Torr, P. H. How benign is benign overfitting? *arXiv preprint arXiv:2007.04028*, 2020.

Schneider, S., Rusak, E., Eck, L., Bringmann, O., Brendel, W., and Bethge, M. Improving robustness against common corruptions by covariate shift adaptation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Shafahi, A., Najibi, M., Ghiasi, A., Xu, Z., Dickerson, J., Studer, C., Davis, L. S., Taylor, G., and Goldstein, T. Adversarial training for free! In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 3358–3369, 2019.

Shu, M., Wu, Z., Goldblum, M., and Goldstein, T. Encoding robustness to image style via adversarial feature perturbations. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. In *International Conference on Learning Representations (ICLR)*, 2013.

Tan, M. and Le, Q. EfficientNet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning (ICML)*, pp. 6105–6114, 2019.

Tsipras, D., Santurkar, S., Engstrom, L., Turner, A., and Madry, A. Robustness may be at odds with accuracy. In *International Conference on Learning Representations (ICLR)*, 2019.

Ulyanov, D., Vedaldi, A., and Lempitsky, V. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.

Wang, D., Shelhamer, E., Liu, S., Olshausen, B., and Darrell, T. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations (ICLR)*, 2020a.

Wang, H., Ge, S., Lipton, Z., and Xing, E. P. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 10506–10518, 2019.

Wang, H., Chen, T., Gui, S., Hu, T., Liu, J., and Wang, Z. Once-for-all adversarial training: In-situ tradeoff between robustness and accuracy for free. *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 7449–7461, 2020b.

Wang, H., Xiao, C., Kossaifi, J., Yu, Z., Anandkumar, A., and Wang, Z. Augmax: Adversarial composition of random augmentations for robust training. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Wang, H., Zhang, A., Zhu, Y., Zheng, S., Li, M., Smola, A., and Wang, Z. Partial and asymmetric contrastive learning for out-of-distribution detection in long-tailed recognition. In *International Conference on Machine Learning (ICML)*, 2022.

Wang, Y., Zou, D., Yi, J., Bailey, J., Ma, X., and Gu, Q. Improving adversarial robustness requires revisiting misclassified examples. In *International Conference on Learning Representations (ICLR)*, 2020c.

Wong, E., Rice, L., and Kolter, J. Z. Fast is better than free: Revisiting adversarial training. In *International Conference on Learning Representations*, 2020.

Xiao, C., Zhu, J.-Y., Li, B., He, W., Liu, M., and Song, D. Spatially transformed adversarial examples. *arXiv preprint arXiv:1801.02612*, 2018.

Xie, C. and Yuille, A. Intriguing properties of adversarial training. In *International Conference on Learning Representations (ICLR)*, 2020.

Xie, C., Tan, M., Gong, B., Wang, J., Yuille, A., and Le, Q. V. Adversarial examples improve image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Yang, Y., Khanna, R., Yu, Y., Gholami, A., Keutzer, K., Gonzalez, J. E., Ramchandran, K., and Mahoney, M. W. Boundary thickness and robustness in learning models. *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Yun, S., Han, D., Oh, S. J., Chun, S., Choe, J., and Yoo, Y. CutMix: Regularization strategy to train strong classifiers with localizable features. In *IEEE International Conference on Computer Vision (ICCV)*, pp. 6023–6032, 2019.

Zhang, H., Dauphin, Y. N., and Ma, T. Fixup initialization: Residual learning without normalization. In *International Conference on Learning Representations (ICLR)*, 2018.

Zhang, H., Yu, Y., Jiao, J., Xing, E. P., Ghaoui, L. E., and Jordan, M. I. Theoretically principled trade-off between robustness and accuracy. In *International Conference on Machine Learning (ICML)*, pp. 7472–7482, 2019.

Zhang, J., Xu, X., Han, B., Niu, G., Cui, L., Sugiyama, M., and Kankanhalli, M. Attacks which do not kill training make adversarial learning stronger. In *International Conference on Machine Learning (ICML)*, pp. 11278–11287, 2020.

Zhong, Z., Zheng, L., Kang, G., Li, S., and Yang, Y. Random erasing data augmentation. *arXiv preprint arXiv:1708.04896*, 2017.

A. How to Interpolate between Two BN Branches in MBNAT

We follow (Merchant et al., 2020) to interpolate between the two BN branches (i.e., the BN_c branch and BN_a branch) *during test time* for MBNAT. Specifically, given an input test image x, we first forward it through the BN_c branch (i.e., the MBN network using BN_c at each normalization layer) to get the output logits z_c, and then through the BN_a branch (i.e., the MBN network using BN_a at each normalization layer) to get the output logits z_a. We then average z_c and z_a with a weighting hyper-parameter γ, i.e., $z = (1 - \gamma)z_c + \gamma z_a$. Finally, we use the averaged logits z as the input to the softmax function to get the final prediction probabilities. As a result, when the value of γ increases from 0 to 1, the influence of BN_a gradually outweighs that of BN_c. When γ is 0 or 1, the MBN model falls back to the simple cases with only one BN (BN_c when $\gamma = 0$ or BN_a when $\gamma = 1$) at each normalization layer. This is the default interpolation method we used in our paper, which is denoted as "MBNAT (logits)" or simply "MBNAT" when used as the default.

Besides the one suggested in (Merchant et al., 2020) (i.e., MBNAT (logits)), we have also investigated other possible interpolation methods between BN_c and BN_a. For example, we can interpolate the outputs of BN_c and BN_a at each MBN layer. Specifically, if the input feature of an MBN layer is denoted as f_i, then the output feature $f_o = (1 - \gamma)BN_c(f_i) + \gamma BN_a(f_i)$, where $BN_c(\cdot)$ and $BN_a(\cdot)$ are the batch normalization operations by BN_c and BN_a, respectively. We denote this method as "MBNAT (all)". We can also conduct this output mixing on some selected MBN layers, while keeping the two parallel outputs in other MBN layers. For example, we can randomly select $p\%$ MBN layers for mixing. We denote this method as "MBNAT (random $p\%$)".

The results in Figure 5 show that "MBNAT (logits)" achieves the best robustness-accuracy trade-off curve among all compared interpolation strategies, so we use it as our default interpolation strategy for MBNAT.

B. More Experimental Results

B.1. Histograms of Decision Margin, Boundary Thickness, and Model Smoothness

We have numerically compared the average decision margins, boundary thickness and model smoothness of SAT, MBNAT and NoFrost in Table 5 (Section 4.3). Here in Figure 6, we visualize the distributions of these metrics on different models using histograms. Figure 6 is simply another way to show the results in Table 5, but gives more detailed information through histogram visualization.
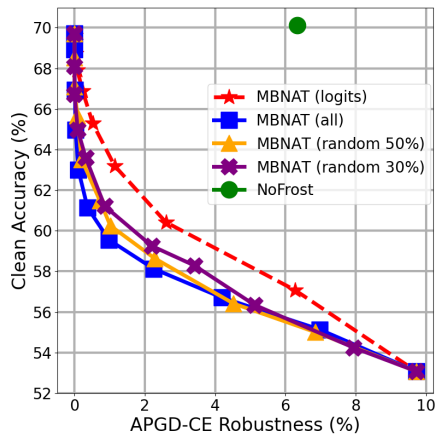

Figure 5. Trade-off between robustness and accuracy of different interpolation strategies on MBNAT with ResNet26.

B.2. Normalizer-free Networks with Standard Training is Not Robust to Adversarial Attacks

Table 6. Normalizer-free networks trained using standard training (only on clean images) is not robust against adversarial attacks. All methods are trained on ImageNet with (NF-)ResNet50.

Method	Clean	PGD	APGD-CE
ST w/ ResNet50	**76.06**	0.04	0.00
ST w/ NF-ResNet50	<u>75.02</u>	0.00	0.00
NoFrost	74.06	**22.45**	**11.96**

In this section, we show that normalizer-free networks does not naturally have satisfactory adversarial robustness after standard training (i.e., training only on clean images). Specifically, we compare the results of standard training on ResNet50 (denoted as ST w/ ResNet50), standard training on NF-ResNet50 (denoted as ST w/ NF-ResNet50), and adversarial training on NF-ResNet50 (NoFrost) in Table 6. As we can see, standard training on both ResNet50 and NF-ResNet50 have almost zero adversarial robustness. This shows that the robustness of NoFrost is not simply the result of more robust network structure, but the combination of the AT algorithm and the AT-friendly normalizer-free network structure.

B.3. Robustness under Different Perturbation Magnitudes

In the main text, we evaluated adversarial robustness using adversarial attacks with perturbation magnitude $\epsilon = 8$. In this section, we evaluate model robustness under different adversarial perturbation magnitudes. Specifically, we compare the robustness of the models in Table 1 (which are trained with $\epsilon = 8$) on targeted PGD (denoted as PGD^T) attack with ϵ ranging from 8 to 16. As shown in Figure 7, the advantage of NoFrost holds on multiple different perturbation magnitudes.
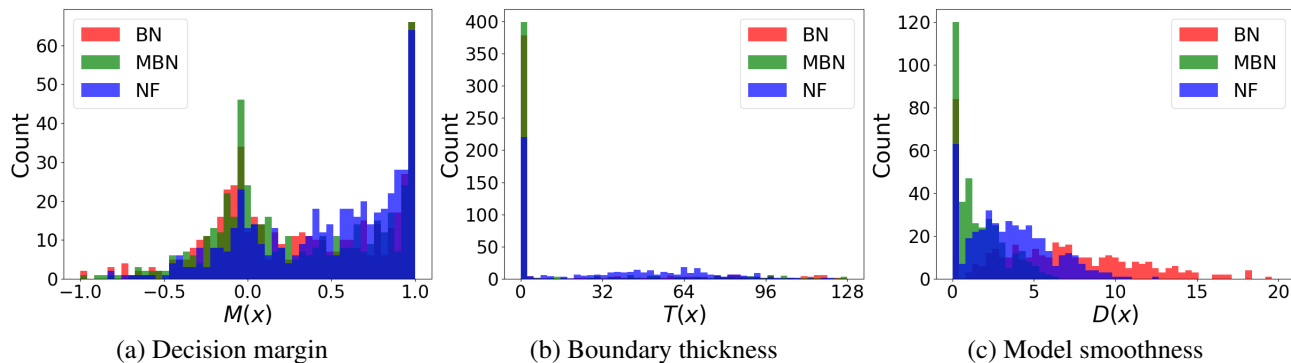
Figure 6. Histograms of decision margin, boundary thickness, and model smoothness of adversarially trained ResNet26 with different normalization strategies. All metrics are measured on the 500 validation images from the first 10 classes of ImageNet. (a) $M(\boldsymbol{x})$ as a metric for decision margin: the larger the better. (b) $T(\boldsymbol{x})$ as a metric for boundaries robustness: the larger the better. (c) $D(\boldsymbol{x})$ as as metric for model smoothness: the smaller the better.
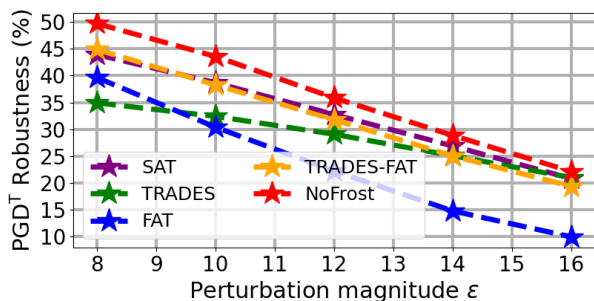

Figure 7. Adversarial robustness under different perturbation magnitudes. All methods are trained on ImageNet with ResNet26.

B.4. Adversarial Training Results with Small Perturbation Magnitudes

In the main text, we set perturbation magnitude $\epsilon = 8$ in both adversarial training and evaluation. Some previous works, such as FastAT (Wong et al., 2020) and FreeAT (Shafahi et al., 2019) conducted adversarial training on ImageNet using smaller perturbation magnitudes such as $\epsilon = 2, 4$, and also evaluation adversarial robustness using the same ϵ values. In this section, we compare NoFrost with FastAT and FreeAT using the small ϵ setting. We use PGD attack with 10 and 50 steps (denoted as PGD-10 and PGD-50 respectively) to evaluate adversarial robustness, following (Shafahi et al., 2019). The results are shown in Table 7. NoFrost largely outperforms FastAT and FreeAT under the small ϵ setting.

Table 7. Adversarial robustness of ResNet50 under perturbation magnitude $\epsilon = 2, 4$. Classification accuracy on clean images and under different adversarial attacks are reported. The best and second to the best numbers are shown in bold and underlined, respectively.

ϵ	Method	Clean	PGD-10	PGD-50
	FastAT	60.90	44.27	44.20
2	FreeAT	64.45	43.52	43.39
	NoFrost	**69.87**	**48.60**	**48.23**
	FastAT	55.45	32.10	31.67
4	FreeAT	60.21	32.77	31.88
	NoFrost	**66.05**	**36.14**	**36.05**