A CLOSER LOOK AT DUAL BATCH NORMALIZATION AND TWO-DOMAIN HYPOTHESIS IN ADVERSARIAL TRAINING WITH HYBRID SAMPLES

Anonymous authors
Paper under double-blind review

ABSTRACT

There is a growing concern about applying batch normalization (BN) in adversarial training (AT), especially when the model is trained on both *adversarial* samples and *clean* samples (termed Hybrid-AT). With the assumption that *adversarial* and *clean* samples are from two different domains, a common practice in prior works is to adopt dual BN, where BN_{adv} and BN_{clean} are used for adversarial and clean branches, respectively. A popular belief for motivating dual BN is that estimating normalization statistics of this mixture distribution is challenging and thus disentangling it for normalization achieves stronger robustness. In contrast to this belief, we reveal that what makes dual BN effective mainly lies in its two sets of affine parameters. Moreover, we demonstrate that the domain gap between adversarial and clean samples is actually not very large, which is counter-intuitive considering the significant influence of adversarial perturbation on the model. Overall, our work sheds new light on understanding the mechanism of dual BN in Hybrid-AT as well as its underlying two-domain hypothesis.

1 INTRODUCTION

Adversarial training (AT) (Madry et al., 2018) that optimizes the model on adversarial examples is a time-tested and effective technique for improving robustness against adversarial attack. Beyond classical AT (also termed Madry-AT) (Madry et al., 2018), a common AT setup is to train the model on both *adversarial* samples and *clean* samples (termed Hybrid-AT) (Goodfellow et al., 2015; Kannan et al., 2018; Xie et al., 2020a). Batch normalization (BN) (Ioffe & Szegedy, 2015) has become a de facto standard component in modern deep neural networks (DNNs) (He et al., 2016; Huang et al., 2017; Zhang et al., 2019a; 2021), however, there is a notable concern regarding how to use BN in the Hybrid-AT setup. The concern mainly stems from a two-domain hypothesis: "clean images and adversarial images are drawn from two different domains" (Xie & Yuille, 2020). Guided by this hypothesis, a technique has been proposed to disentangle the mixture distribution of the two domains by applying a separate BN for each domain (Xie & Yuille, 2020).

The above technique has been adopted in multiple works with different names, such as auxiliary BN (Xie et al., 2020a), mixture BN (Xie & Yuille, 2020), Dual BN (Jiang et al., 2020; Wang et al., 2020; 2021). Despite different names, they refer to the same practice of adopting BN_{adv} and BN_{clean} for adversarial and clean samples, respectively. To avoid confusion, we stick to using Dual BN in the remainder of this work. Despite its increasing popularity, the mechanism of how dual BN helps Hybrid-AT remains not fully clear. Towards a better understanding of the underlying mechanism, we first revisit a long-held belief motivated by the two-domain hypothesis (Xie & Yuille, 2020). Specifically, (Xie & Yuille, 2020) justifies the necessity of dual BN in hybrid AT with the following claim (quoted from the abstract of (Xie & Yuille, 2020)):

"Estimating normalization statistics of the mixture distribution is challenging" and *"disentangling the mixture distribution for normalization, i.e., applying separate BNs to clean and adversarial images for statistics estimation, achieves much stronger robustness."*

The underlying motivation for the above claim is that BN statistics calculated on clean domain is incompatible with training the model on adversarial domain, and vice versa. With the mixed distribution for calculating the normalization statistics, Hybrid-AT with single BN suffers from such

incompatibility, which is avoided by Dual BN through training the clean branch on BN_{clean} and the adversarial branch on BN_{adv}. This cross-domain BN incompatibility is also corroborated by the observation that BN_{clean} yields close-to-zero adversarial robustness. Given the two-domain hypothesis (Xie & Yuille, 2020), the incompatibility between BN_{clean} and adversarial branch might be inherently inevitable. To investigate such incompatibility, we experiment with a new variant of AT with cross-domain BN, namely training the adversarial branch with BN_{clean}. Interestingly, we find that using BN from another domain only has limited influence on the performance. This observation inspires us to challenge the original motivation for justifying the necessity of dual BN in Hybrid-AT.

Except for disentangling the mixed distribution (Xie & Yuille, 2020), dual BN also implicitly introduces two sets of affine parameters. In contrast to prior work that attributes the superiority of dual BN over single BN to disentangled normalization statistics (NS), we conjecture what makes dual BN more effective lies in two sets of affine parameters (APs) instead of disentangled NS. This conjecture is motivated by an interesting observation that swapping NS in the Hybrid-AT has little influence on the model performance and is confirmed with a novel experiment design by untwining NS and AP in dual BN to include one effect while excluding the other. Regarding why introducing two sets of AP is more critical than disentangling NS, we point out that their key difference lies in the fact that AP is learnable while NS is not. Moreover, it has been highlighted in prior work (Xie & Yuille, 2020) that the choice between BN_{adv} and BN_{clean} at test stage has a significant influence on the model performance. Prior works mainly attribute this test-time performance difference regarding the BN choice to their normalization statistics (NS) (Xie & Yuille, 2020; Xie et al., 2020a), however, we show that AP plays a major role.

Note that the motivation for introducing Dual BN is inspired by a two-domain hypothesis that "clean images and adversarial images are drawn from two different domains" (Xie & Yuille, 2020). After showing their motivation does not hold, we further revisit this two-domain hypothesis itself. Interestingly, we find that the domain gap between adversarial and clean samples is not as large as claimed in prior work (Xie & Yuille, 2020). We point out a hidden flaw when prior work visualizes the NS from two domains for highlighting a large adversarial-clean domain gap. Specifically, their visualization ignores the influence of different AP. After fixing this hidden flaw, we demonstrate that this domain gap is small. Surprisingly, under the same perturbation/noise magnitude, we show that there is no significant difference between adversarial-clean domain gap and noisy-clean counterpart, which is counter-intuitive.

Overall, considering the increasing interest in adopting dual BN in Hybrid-AT, our work comes timely by taking a closer look at dual BN in Hybrid-AT as well as its underlying two-domain hypothesis. The contributions of this work are summarized as follows.

- We are the first to attempt at training a model with cross-domain BN statistics. Given the prior motivation for justifying the necessity of dual BN in Hybrid-AT, its success is unexpected and thus inspires us to take a closer look.
- We point out that it also introduces two sets of AP. Through untwining NS and AP in dual BN, we demonstrate that what makes it effective lies in two sets of AP instead of disentangled NS (as claimed in prior work).
- We point out a hidden flaw in prior work for visualizing the NS to highlight a (large) domain gap between adversarial and clean samples. After fixing it, we reveal that the adversarial-clean domain gap is not as large as prior work suggests.

2 Problem overview and related work

2.1 Development of adversarial training

Adversarial training. Adversarial training (AT) has been the most powerful defense method against adversarial attacks, among which Madry-AT (Madry et al., 2018) is a typical method detailed as follows. Let's assume $\mathcal{D}$ is a data distribution with (x, y) pairs and $f(\cdot, \theta)$ is a model parametrized by θ. l indicates cross-entropy loss in classification. Instead of directly feeding clean samples from $\mathcal{D}$ to minimize the risk of $\mathbb{E}_{(x,y)\sim\mathcal{D}}[l(f(x, \theta), y)]$, (Madry et al., 2018) formulates a saddle problem for

finding model parameter θ by optimizing the following adversarial risk:

$$\arg\min_{\theta} \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\max_{\delta\in\mathbb{S}} l(f(x+\delta;\theta), y) \right] \tag{1}$$

where $\mathbb{S}$ denotes the allowed perturbation budget which is a typically l_p norm-bounded ϵ. We term the above adversarial training framework as Classical-AT. It adopts a two-step training procedures(inner maximization + outer minimization), and trains the robust model with only adversarial samples. Following the same procedure, (Xie & Yuille, 2020) proposes to train the robust model with both clean and adversarial samples, termed as **Hybrid-AT**. The loss of Hybrid-AT is defined as follows:

$$\mathcal{L}_{Hybrid} = \alpha l(f(x;\theta), y) + (1-\alpha) l(f(x+\delta;\theta), y) \tag{2}$$

where x and $x+\delta$ indicate clean and adversarial samples, respectively. α is a hyper-parameter for balancing the clean and adversarial branches, is set to 0.5 in this work following (Goodfellow et al., 2015; Xie & Yuille, 2020).

Development of AT. Since the advent of Mardy-AT and Hybrid-AT, numerous works have attempted to improve AT from various perspectives. From the data perspective, (Uesato et al., 2019; Carmon et al., 2019; Zhang et al., 2019b) have independently shown that unlabeled data can be used to improve the robustness. From the model perspective, AT often benefits from the increased model capacity of models (Uesato et al., 2019; Xie & Yuille, 2020). (Xie et al., 2020b; Pang et al., 2020; Gowal et al., 2020) have investigated the influence and suggested that a smooth activation function, like parametric softplus, is often but not always (Gowal et al., 2020) helpful for AT. It has been shown in (Pang et al., 2020) that the basic training settings in AT can have a significant influence on the model performance, and suggested a set of parameters for fair comparison of AT methods. If not specified, we follow their suggested parameter settings.

Experimental setups. In this work, we perform experiments on CIFAR10 with ResNet18 and follow the suggested training setups in (Pang et al., 2020) unless specified. Specifically, we train the model for 110 epochs. The learning rate is set to 0.1 and decays by a factor of 0.1 at the epoch 100 and 105. We adopt an SGD optimizer with weight decay 5×10^{-4}. For generating adversarial examples during training, we use ℓ_∞ PGD attack with 10 iterations and step size $\alpha = 2/255$. For the perturbation constraint, ϵ is set to ℓ_∞ $8/255$ (Pang et al., 2020) or $16/255$ (Xie & Yuille, 2020). Following (Pang et al., 2020), we evaluate the model robustness under PGD-10 attack (PGD attack with 10 steps) and AutoAttack (AA) (Croce & Hein, 2020).

2.2 Batch normalization in AT

Batch normalization (BN). We briefly summarize how BN works in modern networks. For a certain layer in the DNN, we denote the feature layers of a mini-batch in the DNN as $\mathcal{B} = \{x^1, ..., x^m\}$. The feature layers are normalized by mean μ and standard deviation σ as:

$$\hat{x}^i = \frac{x^i - \mu}{\sigma} \cdot \gamma + \beta \tag{3}$$

where γ and β indicates the weight and bias in BN, respectively. To be clear, we refer μ and σ as normalization statistics (NS), γ and β as affine parameters (AP). During training, NS is calculated on the current mini-batch statistics for the update of model weights. Meanwhile, a running average of NS is recorded in the whole training process, which is applied for inference after training ends.

Dual BN in AT. There is an increasing interest in investigating BN in the context of adversarial robustness (Awais et al., 2020; Cheng et al., 2020; Nandy et al., 2021; Sitawarin et al.; Gong et al., 2022). This work focuses on Hybrid-AT with dual BN (Xie & Yuille, 2020; Xie et al., 2020a) which applies BN_{clean} and BN_{adv} to clean branch and adversarial branch, respectively. Prior work (Xie et al., 2020a) shows that AEs can be used to improve recognition (accuracy) by adversarial training where AEs are normalized by an independent BN_{adv}. Moreover, (Xie & Yuille, 2020) has shown that adding clean images in adversarial training (AT) can significantly decrease robustness performance, where such negative effects can be alleviated to a large extent by simply normalizing CEs with an independent BN_{clean}. Inspired by their finding, (Jiang et al., 2020) also adopts Dual BN in adversarial contrastive learning, showing that single BN performs significantly worse than Dual BN. Beyond Dual BN, triple BN has been attempted in (Fan et al., 2021) for incorporating another adversarial branch. Recently, (Wang et al., 2021) has also combined Dual BN with instance normalization to form

Dual batch-and-instance normalization for improving robustness. Despite the above investigation, the underlying mechanism behind Dual BN remains not fully clear. We note that prior works studying dual BN highlight its role of disentangling the mixture distribution for normalization ($\text{NS}_{clean+adv}$) into independent ones (NS_{adv} for adversarial branch, and NS_{clean} for clean branch), while pays less attention to another effect of introducing two sets of learnable APs.

3 CAN BN_{clean} BE COMPATIBLE WITH ADVERSARIAL ROBUSTNESS?

Existing motivation for dual BN. It is hypothesized in (Xie & Yuille, 2020) that clean and adversarial samples are from two domains, thus Dual BN should be introduced to disentangle the mixed distributions in Hybrid-AT. The basic assumption in (Xie & Yuille, 2020) is that there is a clean-adversarial domain gap so that a separate BN is required for each branch. Compared with single BN to handle the mixed distribution, dual BN allows the two branches to be normalized by their corresponding BN only. In other words, the underlying motivation is that the BN statistics from a different domain has a negative influence on the training which can be avoided with Dual BN by only using statistics calculated on its own domain, *i.e.* BN_{clean} for the clean branch and BN_{adv} for the adversarial branch. With this motivation, BN_{clean} might be inherently not compatible with the adversarial branch for achieving robustness due to the clean-adversarial domain gap. For example, it is reported in (Jiang et al., 2020) that the model robustness with BN_{clean} adopted at test time is close to zero, which is also confirmed in our setup. Overall, BN_{clean} in dual BN seems to be incompatible with the desired high (non-zero) robustness. Here, we ask a question:

Can BN_{clean} be compatible with the adversarial branch to yield a robust model during inference?

The answer to the above question has significant implications on the existing motivation for justifying the necessity of dual BN in Hybrid-AT. If the merit of dual BN over single BN lies in the assumption that domain-specific BN statistics is better than mixed one, it is unlikely that BN_{clean} can be compatible with the adversarial branch for achieving adversarial robustness. In other words, if the answer to the above question is positive, the existing motivation for dual BN might not hold.

Cross-AT. There is a possibility that this incompatibility is caused by the fact that the adversarial branch of Hybrid-AT is trained with BN_{adv} instead of BN_{clean}. To test the above possibility, we perform a cross-AT by replacing vanilla BN with the module in Figure 1 where the adversarial branch is normalized by BN_{clean}. Specifically, the cross-AT still keeps its clean branch in the Hybrid-AT for the forward propagation to get BN_{clean} but the model weights are updated only by the adversarial branch (as shown in Figure 1). Such a setup guarantees that BN_{clean} is used and only used for the adversarial branch. We find that the cross-AT yields a robust model with BN_{clean} at test time. Given the motivation for introducing dual BN in (Xie & Yuille, 2020), the success of making BN_{clean} compatible with adversarial robustness via Cross-AT is unexpected.

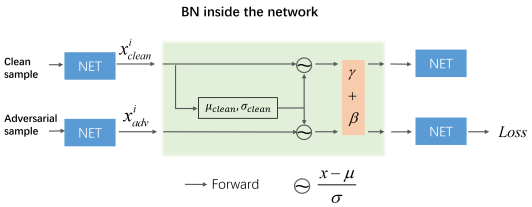
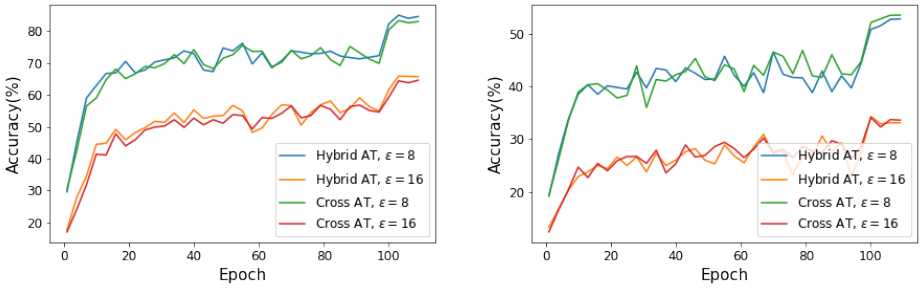


Figure 1: Cross-AT: Replacing BN_{adv} with BN_{clean} in the adversarial branch of Hyrbid-AT.

The results of Cross-AT are reported in Figure 2. We find that Cross-AT achieves comparable performance as Hybird-AT with BN_{adv} adopted during inference. Such a success inspires us to suspect that the merit of dual BN over single BN in Hybrid-AT might not lie in disentangling the mixed distribution for normalization so that each branch is normalized by the statistics calculated on its own domain (as claimed in (Xie & Yuille, 2020)).

4 A CLOSER LOOKER AT DUAL BN IN HYBRID-AT

Introducing an auxiliary BN component, Dual BN causes two changes: (i) disentangling the mixture distribution for normalization statistics (NS) and (ii) introducing two sets of affine parameters (AP). Prior works (Xie & Yuille, 2020; Xie et al., 2020a) mainly highlight the effect of disentangled NS but pay little attention to that of two sets of AP.



(a) Clean Accuracy (b) Robust Accuracy

Figure 2: Cross-AT robustness . PGD-10 is evaluated with the step size $\epsilon/4$.

Cross-Hybrid-AT. Inspired by the success of Cross-AT in Section. 3, we perform a Hybrid-AT with cross BN statistics. Different from the setup in Section. 3 that only uses a single BN, dual BN has two sets of APs. Therefore, we let each branch use its own AP but train the two branches with NS from another branch. In other words, the clean branch is trained (and tested) with NS_{adv} and the adversarial branch is trained (and tested) with NS_{clean}. We term this setup Cross-Hybrid-AT and its results are reported in Table 1. Table 1 shows that swapping NS has little influence on the performance on the two branches, *e.g.* NS_{clean} in Cross Hybrid-AT achieves similar performance with NS_{adv} in Hybrid-AT. Comparing vanilla Hybrid-AT and Cross-Hybrid-AT, we highlight that they achieve comparable performance for both adversarial branch and clean branch. The original motivation of disentangling NS is to avoid the influence of NS calculated on partial (half) samples from a different domain, however, we show that NS calculated on full samples from a different domain distribution actually has no significant influence on the performance. Motivated by this observation, in contrast to prior works that attribute the merit of dual BN over single BN in Hybrid-AT to disentangled NS, we establish the following hypothesis:

Model	NS	AP	Clean	PGD-10	AA
Hybrid-AT	NS_{adv}	AP_{adv}	61.84	31.67	22.51
	NS_{clean}	AP_{clean}	94.18	0.00	0.00
Cross Hybrid-AT	NS_{clean}	AP_{adv}	59.56	31.25	22.40
	NS_{adv}	AP_{clean}	93.86	0.00	0.00

Table 1: Cross Hybrid-AT ($\epsilon = 16/255$).

Conjecture 1. We conjecture that what makes Dual BN more effective than single BN in Hybrid-AT is mainly caused by two sets of AP instead of disentangled NS.

4.1 Untwining NS and AP in Dual BN

Training setup design. As discussed above, compared with the default Hybrid-AT baseline, Dual BN brings two effects: disentangled NSs and two sets of APs. To determine the influence of each effect on the model performance, we design two setups of experiments to include only one effect while excluding the other. In Setup1, we only include the effect of two sets of APs, by applying two different sets of APs (β_{adv}/γ_{adv} and $\beta_{clean}/\gamma_{clean}$) in the adversarial and clean branches while using the default mixture distribution for normalization. In Setup2, we only include the effect of two sets of NSs by only disentangling this mixture distribution with two different sets of NSs while making BN_{clean} and BN_{adv} share the same set of APs. The above setups of BNs are summarized in Figure 3.

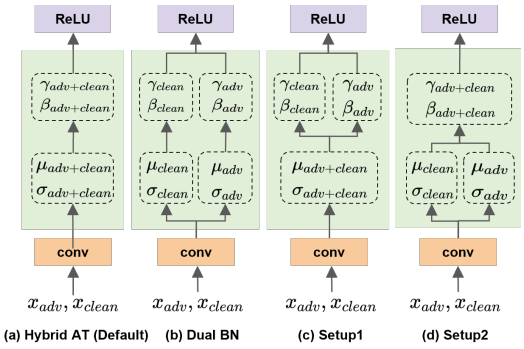


Figure 3: Illustration of different BN setups.

Results. As shown in Table 2, Dual BN (with BN_{adv} during inference) brings significant robustness improvement over the default baseline, which is consistent with findings in (Xie & Yuille, 2020). Interestingly, under the attack of PGD-10, their robustness gap is not significant, however, under AA, the Default setup achieves very low robustness (7.03% and 6.93% for $\epsilon = 8/255$ and $\epsilon = 16/255$,

Setups	NS	AP	$\epsilon = 8/255$			$\epsilon = 16/255$		
			Clean	PGD-10	AA	Clean	PGD-10	AA
Single BN	1	1	88.06	49.75	7.03	93.70	29.86	0.48
Dual BN (BN_{adv})	2	2	82.77	51.33	46.19	61.84	31.67	23.14
Dual BN (BN_{clean})	2	2	94.91	0.32	0.10	94.18	0.00	0.00
Setup1 (AP_{adv})	1	2	81.86	50.99	44.63	60.02	30.89	23.43
Setup1 (AP_{clean})	1	2	94.74	0.10	0.04	94.30	0.00	0.00
Setup2 (NS_{adv}^{adv})	2	1	85.49	49.39	42.96	55.91	21.92	10.64
Setup2 ($\text{NS}_{clean}^{clean}$)	2	1	89.22	49.48	42.96	86.35	1.08	0.00

Table 2: Test accuracy (%) . For NSs, 1 indicates mixture distribution and 2 indicates disentangled distribution for normalziaiton. For APs, 1 indicates single set of APs and 2 indicates double sets of APs.

respectively). Moreover, Setup1 (AP_{adv}) achieves comparable robustness as that of Dual BN (BN_{adv}) for $\epsilon = 8/255$ and $\epsilon = 16/255$, suggesting two sets of APs alone significantly improve the robustness. The results collaborate our conjecture that two sets of APs are a key factor in improving Hybrid-AT. The effect of two sets of NSs is more nuanced: for a small perturbation $\epsilon = 8/255$, disentangling mixture distribution is beneficial for boosting the robustness under strong AA; for a large perturbation $\epsilon = 16/255$ this benefit is less significant. This can be explained by the fact the domain distribution mismatch is much smaller for $\epsilon = 8/255$ than that for $\epsilon = 16/255$. Note that Setup2 (NS_{adv}^{adv}) and Setup2 ($\text{NS}_{clean}^{clean}$) achieve comparable robustness for $\epsilon = 8/255$, while their gap is significant for $\epsilon = 16/255$. Overall, we conclude two sets of APs are sufficient for avoiding the issue of low robustness against AA in Default Hybrid-AT, and achieve comparable robustness as Dual BN. Moreover, we experiment with using two sets of AP in either first half or second half of the model, which achieves slightly inferior robustness performance (see Table 3).

Setups	AP (first half)	AP (second half)	Clean	PGD-10	AA
Setup1 (AP_{adv})	2	2	60.02	30.89	23.43
Setup1 (AP_{clean})	2	2	94.30	0.00	0.00
Setup3 (AP_{adv})	1	2	58.14	29.45	19.80
Setup3 (AP_{clean})	1	2	93.66	0.00	0.00
Setup4 (AP_{adv})	2	1	59.91	30.62	22.10
Setup4 (AP_{clean})	2	1	94.05	0.00	0.00

Table 3: Test accuracy (%) with $\epsilon = 16/255$. With 1 set of NS, different sets of AP in the first and second half of model.

4.2 Does BN normalization statistics characterize different model performance during inference?

Even though dual BN is adopted during training, only one branch of BN parameters can be adopted during inference, and prior works (Xie & Yuille, 2020; Jiang et al., 2020) have reported a large performance gap between BN_{clean} and BN_{adv} during inference. Regarding this phenomenon, prior work claims that "BN normalization statistics (NS) characterizes different model performance" (Xie & Yuille, 2020). In the following, we will refute this claim and show that APs play a major role.

AP plays a main role in robustness evaluation. As discussed above, Dual BN not only introduces two sets of NS but also two sets of affine parameters (AP). To investigate the individual role of NS and AP during inference, we conduct a swap experiment during inference as shown in Table 4. The model with original DuN achieves 51.33% and 0.10 % with BN_{adv} (NS_{adv} and AP_{adv}) and BN_{clean} (NS_{clean} and AP_{clean}), respectively. If we keep NS_{clean} fixed and swap AP, the robustness increases from 0.32% to 17.1% with the AP_{clean} replaced by AP_{adv} . However, when we keep AP_{clean} fixed, both NS_{clean} and NS_{adv} achieve almost zero robustness. These phenomena are strong evidence demonstrating that the APs play a larger part in robustness evaluation. However, the results seem to suggest that NS is also important. For example, the robustness with the configuration of NS_{clean} and AP_{adv} is still much lower than that of NS_{adv} and AP_{adv}, for which our analysis in the following section provides an explanation (See Table 5).

Setups	NS	AP	PGD10	AA	PGD10	AA
			$\epsilon = 8/255$		$\epsilon = 16/255$	
Dual BN	NS_{adv}	AP_{adv}	51.33	46.19	31.67	22.51
	NS_{clean}	AP_{clean}	0.32	0.10	0.00	0.00
Swap	NS_{clean}	AP_{adv}	17.1	9.16	10.02	9.80
	NS_{adv}	AP_{clean}	0.00	0.00	0.45	0.00

Table 4: Swap NS_{clean} and NS_{adv} in Hybrid-AT during inference.

5 A CLOSER LOOKER AT THE TWO-DOMAIN HYPOTHESIS

A model trained on a source domain performs poorly on a new target domain when there is a domain shift (Daumé III, 2007; Sun et al., 2017). With BN as the target, it is common in the literature (Li et al., 2017; Benz et al., 2021; Schneider et al., 2020; Xie & Yuille, 2020; Xie et al., 2020a) to indicate the domain gap by the difference of NS between two domains. For example, an early work (Li et al., 2017) has shown that adapting NS from the target domain during inference can improve the performance on a new target domain without retraining the model. This test-time BN adaptation has also been adopted in (Benz et al., 2021; Schneider et al., 2020) for improving the model robustness against common corruptions by perceiving them (random noise for instance) as a new domain. With such an understanding, it is straightforward for prior works (Xie & Yuille, 2020; Xie et al., 2020a; Jiang et al., 2020) to also perceive adversarial domain as a new domain.

5.1 A HIDDEN FLAW OF VISUALIZING NS IN PRIOR WORK

To highlight an adversarial-clean domain gap, prior work (Xie & Yuille, 2020) visualizes the difference of NS in BN_{adv} and BN_{clean} (see Figure 5 of (Xie & Yuille, 2020)). We quote the following sentence from (Xie & Yuille, 2020): "*We observe that clean images and adversarial images induce significantly different running statistics, though these images share the same set of convolutional filters for feature extraction*". With our analysis in Section. 4, however, we know that the AP in BN_{clean} and BN_{adv} is different. The clean branch and adversarial branch still have different weights, *i.e.* AP, even though the same set of convolutional filters are shared. In other words, the significant difference between NS_{clean} and NS_{adv} is induced by not only the difference between image inputs (clean images *v.s.* adversarial images) but also different model (AP) weights. To summarize, the NS difference between BN_{clean} and BN_{adv} is characterized by two factors: (a) AP inconsistence and (b) different domain inputs.

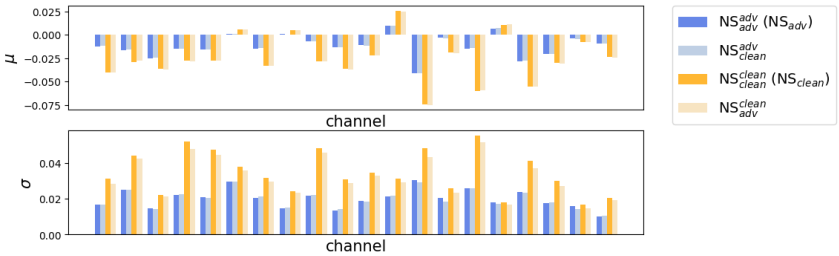


Figure 4: μ and σ visualization. Randomly choose 20 channels and display the APs in NPs.

In the default setup of dual BN, NS_{clean} is calculated on clean samples with AP_{clean}, while NS_{adv} is calculated on adversarial samples with AP_{adv}. We additionally calculate the NS on clean samples with AP_{adv} (denoted as NS_{clean}^{adv}) and calculate the NS on adversarial samples with AP_{clean} (denoted as NS_{adv}^{clean}). Following NS_{adv}^{clean} and NS_{clean}^{adv} to indicate AP choice with the superscript and indicate sample choice with the subscript, we can also denote NS_{clean} as $\text{NS}_{clean}^{clean}$ and denote NS_{adv} as NS_{adv}^{adv}. To exclude the influence of AP inconsistency, we intend to compare NS between clean and adversarial samples with the same AP. In other words, the domain gap is characterized by the difference between $\text{NS}_{clean}^{clean}$ and NS_{clean}^{adv} or that between NS_{adv}^{adv} and NS_{adv}^{clean}. Following the procedures in (Xie & Yuille, 2020), we plot different types of NS in Figure 4 by randomly sampling 20 channels of the second BN layer in the first residual block. Fig. 4 shows that there exists a gap between $\text{NS}_{clean}^{clean}$ and NS_{adv}^{adv}, which is consistent with the findings in (Xie & Yuille, 2020). Moreover, there are two other observations from Figure 4. First, if we fix the input samples and calculate NS with different AP, there exists a large gap, *i.e.* the gap between NS_{clean}^{adv} and $\text{NS}_{clean}^{clean}$, as well as the gap between NS_{adv}^{adv} and NS_{adv}^{clean}. Second, those NSs with the same APs are very close to each other, *e.g.*, NS_{adv}^{adv} and NS_{clean}^{adv} are very similar to each other, and the same applies for NS_{adv}^{clean} and $\text{NS}_{clean}^{clean}$. The visualization results highlight the significance of AP in Dual BN, and is consistent with the finding in Section 4. Without considering the influence of AP, the visualization and conclusions in (Xie & Yuille, 2020) might convey a misleading message.

An interesting phenomenon in Table 4 is that the robustness with NS_{clean} and AP_{adv} achieves much lower robustness(17.1%) than the original BN_{adv} (51.33%). This is caused by the fact that the NS_{clean} is calculated on AP_{clean} and thus there is a mismatch between NS_{clean} and AP_{adv}. Following the notations in Fig. 4, we evaluate a pretrained Hybrid-AT model by replacing its original NS_{clean} and NS_{adv} with different NS, as shown in Table 5. Table 5 shows that given AP_{adv}, NS_{clean}^{adv} achieves a robustness of 51.75%, which is comparable to 51.33% with NS_{adv}^{adv}. Moreover, given AP_{clean}, both NS_{adv}^{clean} and $\text{NS}_{clean}^{clean}$ yield almost zero robustness. We conclude that AP characterizes the large robustness gap between BN_{clean} and BN_{adv} during inference, instead of NS claimed in (Xie & Yuille, 2020). When AP is fixed, the robustness gap between the NS calculated on clean or adversarial samples is quite subtle.

Setups	NS	AP	PGD10	AA	PGD10	AA
			$\epsilon = 8/255$		$\epsilon = 16/255$	
Dual BN	NS_{adv}^{adv}	AP_{adv}	51.33	46.19	31.67	22.51
	$\text{NS}_{clean}^{clean}$	AP_{clean}	0.32	0.10	0.00	0.00
Fine-grain	NS_{clean}^{adv}	AP_{adv}	51.75	46.55	32.73	24.40
	NS_{adv}^{clean}	AP_{clean}	0.00	0.00	0.00	0.00

Table 5: Fine-grained categorization of NS in Hybrid-AT during inference, $\epsilon = 16/255$.

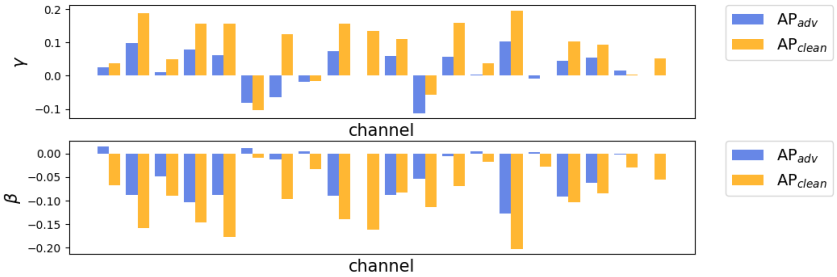


Figure 5: γ and β visualization. Randomly choose 20 channels and display the APs in BN_{clean} and BN_{adv}.

Overall, we demonstrate that the adversarial-clean domain gap is not as large as reported in prior work (Xie & Yuille, 2020). The reason can be attributed to whether the AP difference is considered. The difference between AP_{clean} and AP_{adv} is visualized in Figure 5, which shows that there is a significant difference between them. For a quantitative comparison, we measure the Wasserstein distance in different layers, as shown in Figure 6. It shows a large distance between AP_{clean} and AP_{adv} in all layers. However, with the same AP_{adv}, the gap between NS_{adv}^{adv} and NS_{clean}^{adv} stays almost zero in all layers.

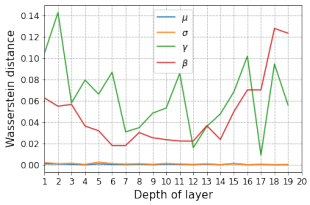


Figure 6: Layer-wise discrepancy visualization.

5.2 Comparing adversarial-clean and noisy-clean domain gap

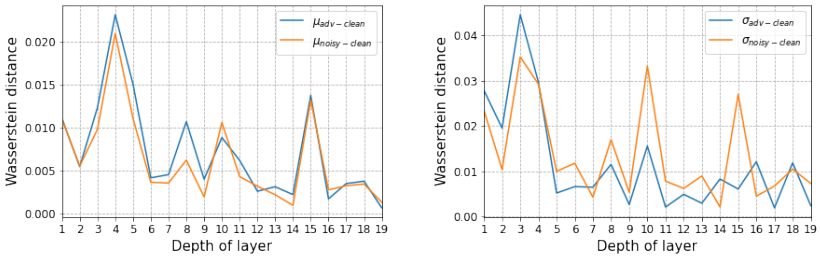
As suggested in (Benz et al., 2021; Schneider et al., 2020), noisy samples (images corrupted by random noise) can be seen as a domain different from clean samples. Adversarial perturbation is a *worst-case* noise for attacking the model. Taking a ResNet18 model trained on clean samples for example, we report the performance under adversarial perturbation and random noise (with the same magnitude) in Table 6. As expected, the model accuracy drops to zero with adversarial perturbation. Under random noise of the same magnitude, we find that the model performance only drops by a small margin. Given that the influence of adversarial perturbation on the model performance is significantly larger than that of random noise, it might be tempting to believe that the adversarial-clean domain gap is much larger than noisy-clean domain gap.

Noise/perturbation Size	0	8/255	16/255
Random noise	94.0	92.7	86.6
Adversarial perturbation	94.0	0.00	0.00

Table 6: Accuracy (%) under random noise and adversarial perturbation.

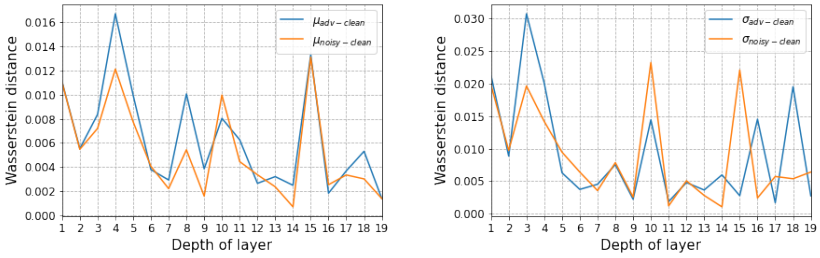
With Wasserstein distance of NS between different domains as the metric, we compare the adversarial-clean domain gap with noisy-clean counterpart on the above ResNet18 model trained on clean samples, as shown in Figure 7 (perturbation/noise magnitude $16/255$) and Figure 8 (perturbation/noise magnitude $8/255$). Interestingly, we observe that there is no significant difference between adversarial-clean domain gap and noisy-clean counterpart. In other words, the adversarial-clean domain gap is

not as large as many might believe considering the strong performance drop caused by adversarial perturbation.



(a) μ (b) σ

Figure 7: Visualization of advresarial-clean domain gap and noisy-clean domain gap (perturbation/noise magnitude is set to $16/255$.



(a) μ (b) σ

Figure 8: Visualization of advresarial-clean domain gap and noisy-clean domain gap (perturbation/noise magnitude is set to $8/255$.

Discussion on the difference between NS and AP in dual BN. Two sets of APs can be seen as disentangling APs for the two branches in Hybrid-AT. Why is disentangling APs more important than disentangling NSs for Hybrid-AT? Mathematically, both NS and AP are model weights that perform linear operation on the intermediate features, however, a key difference is that APs are learnable while NSs are not. Akin to prior work justifying the role of disentangling NSs with the two-domain hypothesis, we provide a two-task hypothesis for explaining two sets of APs. Intuitively, with the two branches in Hybrid-AT, the model weights are trained for two tasks: one for clean accuracy and the other for robustness. Intuitively, it is difficult for a single same set of parameters to realize two tasks. Disentangling APs is equivalent to making partial learnable network weights are not shared between the two tasks. Considering the adversarial-clean domain gap is similar to noisy-clean domain gap as well as strict constraint on allowable perturbation budget, future works utilizing dual BN in Hybrid AT are suggested to discard the two-domain hypothesis and embrace the two-task hypothesis.

6 Conclusion

We attempt at training the adversarial branch with BN statistics calculated on the clean branch, the success of which motivates us to doubt the motivation in prior work for justifying the necessity of dual BN in Hybrid AT. Overall, we take a closer look at dual BN and its underlying two-domain hypothesis in a prior work (Xie & Yuille, 2020), which yields two intriguing findings. First, what makes dual BN effective lies in two sets of affine parameters instead of disentangled normalization statistics. Second, the adversarial-clean domain gap is not as large as many might expect and it is similar to its noisy-counterpart under the same perturbation/noise magnitude.

REFERENCES

Muhammad Awais, Fahad Shamshad, and Sung-Ho Bae. Towards an adversarially robust normalization approach. *arXiv preprint arXiv:2006.11007*, 2020.

Philipp Benz, Chaoning Zhang, Adil Karjauv, and In So Kweon. Revisiting batch normalization for improving corruption robustness. *WACV*, 2021.

Yair Carmon, Aditi Raghunathan, Ludwig Schmidt, Percy Liang, and John C Duchi. Unlabeled data improves adversarial robustness. *NeurIPS*, 2019.

Minhao Cheng, Zhe Gan, Yu Cheng, Shuohang Wang, Cho-Jui Hsieh, and Jingjing Liu. Adversarial masking: Towards understanding robustness trade-off for generalization. 2020.

Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *ICML*, 2020.

Hal Daumé III. Frustratingly easy domain adaptation. In *Association of Computational Linguistics (ACL)*, 2007.

Lijie Fan, Sijia Liu, Pin-Yu Chen, Gaoyuan Zhang, and Chuang Gan. When does contrastive learning preserve adversarial robustness from pretraining to finetuning? *NeurIPS*, 2021.

Xinyu Gong, Wuyang Chen, Tianlong Chen, and Zhangyang Wang. Sandwich batch normalization: A drop-in replacement for feature distribution heterogeneity. In *WACV*, 2022.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *ICLR*, 2015.

Sven Gowal, Chongli Qin, Jonathan Uesato, Timothy Mann, and Pushmeet Kohli. Uncovering the limits of adversarial training against norm-bounded adversarial examples. *arXiv preprint arXiv:2010.03593*, 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *CVPR*, 2017.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.

Ziyu Jiang, Tianlong Chen, Ting Chen, and Zhangyang Wang. Robust pre-training by adversarial contrastive learning. *NeurIPS*, 2020.

Harini Kannan, Alexey Kurakin, and Ian Goodfellow. Adversarial logit pairing. *arXiv preprint arXiv:1803.06373*, 2018.

Yanghao Li, Naiyan Wang, Jianping Shi, Jiaying Liu, and Xiaodi Hou. Revisiting batch normalization for practical domain adaptation. *ICLR workshp*, 2017.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018.

Jay Nandy, Sudipan Saha, Wynne Hsu, Mong Li Lee, and Xiao Xiang Zhu. Adversarially trained models with test-time covariate shift adaptation. *arXiv preprint arXiv:2102.05096*, 2021.

Tianyu Pang, Xiao Yang, Yinpeng Dong, Hang Su, and Jun Zhu. Bag of tricks for adversarial training. *arXiv preprint arXiv:2010.00467*, 2020.

Steffen Schneider, Evgenia Rusak, Luisa Eck, Oliver Bringmann, Wieland Brendel, and Matthias Bethge. Improving robustness against common corruptions by covariate shift adaptation. *NeurIPS*, 2020.

Chawin Sitawarin, Arvind Sridhar, and David Wagner. Improving the accuracy-robustness trade-off for dual-domain adversarial training.

Baochen Sun, Jiashi Feng, and Kate Saenko. Correlation alignment for unsupervised domain adaptation. In *Domain Adaptation in Computer Vision Applications*. 2017.

Jonathan Uesato, Jean-Baptiste Alayrac, Po-Sen Huang, Robert Stanforth, Alhussein Fawzi, and Pushmeet Kohli. Are labels required for improving adversarial robustness? *NeurIPS*, 2019.

Haotao Wang, Tianlong Chen, Shupeng Gui, Ting-Kuei Hu, Ji Liu, and Zhangyang Wang. Once-for-all adversarial training: In-situ tradeoff between robustness and accuracy for free. *arXiv preprint arXiv:2010.11828*, 2020.

Haotao Wang, Chaowei Xiao, Jean Kossaifi, Zhiding Yu, Anima Anandkumar, and Zhangyang Wang. Augmax: Adversarial composition of random augmentations for robust training. *NeurIPS*, 2021.

Cihang Xie and Alan Yuille. Intriguing properties of adversarial training at scale. *ICLR*, 2020.

Cihang Xie, Mingxing Tan, Boqing Gong, Jiang Wang, Alan L Yuille, and Quoc V Le. Adversarial examples improve image recognition. In *CVPR*, 2020a.

Cihang Xie, Mingxing Tan, Boqing Gong, Alan Yuille, and Quoc V Le. Smooth adversarial training. *arXiv preprint arXiv:2006.14536*, 2020b.

Chaoning Zhang, Francois Rameau, Seokju Lee, Junsik Kim, Philipp Benz, Dawit Mureja Argaw, Jean-Charles Bazin, and In So Kweon. Revisiting residual networks with nonlinear shortcuts. In *BMVC*, 2019a.

Chaoning Zhang, Philipp Benz, Dawit Mureja Argaw, Seokju Lee, Junsik Kim, Francois Rameau, Jean-Charles Bazin, and In So Kweon. Resnet or densenet? introducing dense shortcuts to resnet. In *WACV*, 2021.

Jingfeng Zhang, Bo Han, Gang Niu, Tongliang Liu, and Masashi Sugiyama. Where is the bottleneck of adversarial learning with unlabeled data? *arXiv preprint arXiv:1911.08696*, 2019b.