Towards Understanding Dual BN In Hybrid Adversarial Training

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 not very large, which is counter-intuitive considering the significant influence of adversarial perturbation on the model. We further propose a two-task hypothesis for a better understanding and improvement of Hybrid-AT. Overall, our work sheds new light on understanding the mechanism of Dual BN in Hybrid-AT and its underlying justification.

1 Introduction

Adversarial training (AT) (Ganin et al., 2016; Madry et al., 2018; Shafahi et al., 2019; Andriushchenko & Flammarion, 2020; Bai et al., 2021) that optimizes the model on adversarial examples is a time-tested and effective technique for improving robustness against adversarial attack (Qiu et al., 2019; Xu & Yang, 2020; Dong et al., 2018; Zhang et al., 2021b). 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 & Yuille, 2020; Xie et al., 2020a). Batch normalization (BN) (Ioffe & Szegedy, 2015; Santurkar et al., 2018; Bjorck et al., 2018; Li et al., 2017) 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; 2021a), however, there is a notable concern regarding how to use BN in the Hybrid-AT setup. This concern mainly stems from Xie & Yuille (2020); Xie et al. (2020a), which claim the adversarial and clean samples are from two different domains, and thus a separate BN should be used for each domain. This technique applying different BN for different domains has been adopted in multiple works with different names, e.g., Dual BN (Jiang et al., 2020; Wang et al., 2020; 2021) and mixture BN (Xie & Yuille, 2020). With different names, however, they refer to the same practice of adopting BN_{adv} and BN_{clean} for adversarial and clean samples, respectively. To avoid confusion, we use Dual BN for the remainder of this work.

Despite the increasing popularity of Dual BN, the mechanism of how Dual BN helps Hybrid-AT remains not fully clear. Towards a better understanding of this mechanism, we revisit a long-held belief in Xie & Yuille (2020); Xie et al. (2020a). Specifically, it 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 above claim (Xie & Yuille, 2020) emphasizes the necessity of disentangling the normalization statistics (NS) in Hybrid-AT. The underlying motivation for the above claim is that BN statistics calculated on clean domain are incompatible with training the model on adversarial domain, and vice versa. Therefore, Hybrid-AT with single BN suffers from such incompatibility with BN statistics calculated from the mixed distribution, while Dual BN can avoid the incompatibility through training the clean and adversarial samples with two BN branches separately. As a preliminary investigation, our work experiments with a new variant of AT with Cross-BN, namely training the adversarial samples with BN_{clean} and vice versa. Interestingly, we find that using BN from another domain only has limited influence on the performance. This observation inspires us to have a closer look at how Dual BN works in Hybrid-AT. Through untwining normalization statistics (NS) and affine parameters (AP) in Dual BN to include one effect while excluding the other, we demonstrate that disentangled AP plays the main role in the merit of Dual BN in Hybrid-AT. This finding refutes the prior claim emphasizing the role of disentangled NS in Dual BN (Xie & Yuille, 2020; Xie et al., 2020a), and also inspires us to investigate whether the motivation for Dual BN holds, i.e., the two-domain hypothesis in Xie & Yuille (2020); Xie et al. (2020a).

As the motivation for adopting Dual BN, the two-domain hypothesis assumes that *"clean images and adversarial images are drawn from two different domains"* (quoted from Xie & Yuille (2020)). This hypothesis is verified in Xie & Yuille (2020) mainly by the visualization of NS, which highlights a large adversarial-clean domain gap. However, we point out that their visualization has a hidden flaw, which makes their claim regarding the domain gap between adversarial and clean samples deserve a closer look. Specifically, the visualization in Xie & Yuille (2020) ignores the influence of different AP when calculating NS. After fixing this hidden flaw, we demonstrate that the adversarial-clean domain gap is not as large as claimed in prior work. Interestingly, under the same perturbation/noise magnitude, we show that there is no significant difference between adversarial-clean domain gap and noisy-clean counterpart.

Inspired by the above findings, we propose a two-task hypothesis to replace the two-domain hypothesis in Xie & Yuille (2020); Xie et al. (2020a) for justification on how Dual BN works in Hybrid-AT. Specifically, we claim that there are two tasks in Hybrid-AT: one task for clean accuracy and the other for robustness. Therefore, it is difficult for one set of model parameters to achieve both goals. With the two-task hypothesis, we can generalize Hybrid-AT with Dual BN to various model designs. As a toy example, we show that a simple Dual Linear model performs similarly with Dual BN. Moreover, to tackle the problem that only one branch of Dual BN can be adopted during inference, the two-task hypothesis provides more possibilities to adopt only one set of model parameters while reducing the task discrepancy during the training of Hybrid-AT.

The model robustness under PGD-10 attack (PGD attack with 10 steps) (Madry et al., 2018) and AutoAttack (AA) (Croce & Hein, 2020) are evaluated in our analysis as the basic experimental settings, with more details and more specific setup discussed in the context. 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 hypothesis for justification. The main findings of our investigation are summarized as follows:

- We refute prior claims by showing that disentangling normalization statistics (NS) plays little role in explaining the merit of Dual BN over single BN in Hybrid-AT. By contrast, introducing two sets of affine parameters (AP) is the key factor.

- After pointing out a hidden flaw of NS visualization in prior work, we refute the two-domain hypothesis in prior work by demonstrating the adversarial-clean domain gap is not that large. This also corroborates our above finding regarding NS disentanglement.

- For justifying the role of two APs, we propose a two-task perspective on Hybrid-AT by perceiving the two APs as a way to mitigate the two-task conflict. This perspective also shows the possibility of alternative solutions beyond dual BN, for which we show the effectiveness of a regularization loss.

2 Problem overview and related work

2.1 Development of adversarial training

Adversarial training. Adversarial training (AT) (Ganin et al., 2016; Madry et al., 2018; Shafahi et al., 2019; Andriushchenko & Flammarion, 2020; Bai et al., 2021) 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 procedure (inner maximization + outer minimization), and trains the robust model with only adversarial samples. Following the same procedure, Xie & Yuille (2020); Xie et al. (2020a) propose 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 Classical-AT (Madry et al., 2018) and Hybrid-AT (Xie & Yuille, 2020; Xie et al., 2020a), 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. (2019c) 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. Another branch of studies aims to improve the training efficiency of adversarial training based on PGD attack, termed as FAST AT (de Jorge et al., 2022; Jia et al., 2022b; Park & Lee, 2021; Wong et al., 2020; Andriushchenko & Flammarion, 2020; Jia et al., 2022a). Specifically, FGSM attack is adopted in Wong et al. (2020); Andriushchenko & Flammarion (2020); de Jorge et al. (2022) to replace PGD attack during training, which achieves promising robustness with catastrophic overfitting problem tackled. 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 in Pang et al. (2020).

Experimental setups. In this work, we perform experiments on CIFAR10 (Krizhevsky et al., 2009; Andriushchenko & Flammarion, 2020; Zhang et al., 2022) with ResNet18 (Andriushchenko & Flammarion, 2020; Targ et al., 2016; Wu et al., 2019; Li et al., 2016; Zhang et al., 2022) 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 indicate 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 adversarial samples can be used to improve recognition (accuracy) by adversarial training where adversarial samples 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 clean samples 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. Wang et al. (2021) has also combined Dual BN with Instance Normalization to form Dual batch-and-Instance Normalization for improving robustness. A drawback of applying Dual BN in Hybrid-AT lies in the unknown source of samples during inference, which makes it difficult to choose the test BN. Prior work (Xie & Yuille, 2020) interprets the necessity of Dual BN from the perspective of an inherent large adversarial-clean domain gap, which implicitly suggests disentangling NS (via Dual BN) might be the only solution. Our work revisits how Dual BN works in Hybrid-AT and finally proposes a new interpretation from a new two-task perspective, which encourages new directions of overcoming the two-task conflict in Hybrid-AT with appropriate regularization instead of Dual BN.

3 On the BN induced misalignment

In Hybrid-AT, the model is trained with two branches: a clean branch and an adversarial branch. These two branches share all model weights but are found to require independent BN modules, i.e., Dual BN (Xie & Yuille, 2020; Xie et al., 2020a). At test time, only a single branch can be used by choosing either BN$_{adv}$ or BN$_{clean}$. The adversarial branch (with BN$_{adv}$) is adopted in Xie & Yuille (2020) for prioritizing high model robustness, while BN$_{clean}$ is adopted in Xie et al. (2020a) for only considering clean accuracy.

Table 1: Test accuracy (%) of Hybrid-AT with Dual BN. BN$_{clean}$ leads to almost zero robustness under both perturbation budgets (ϵ): 8/255 and 16/255.

ϵ	Setups	Clean	PGD-10	AA
8/255	Dual BN (BN$_{adv}$)	82.77	51.33	46.19
	Dual BN (BN$_{clean}$)	94.91	0.32	0.10
16/255	Dual BN (BN$_{adv}$)	61.84	31.67	23.14
	Dual BN (BN$_{clean}$)	94.18	0.00	0.00

However, swapping the BN during inference, i.e., adopting BN$_{clean}$ for robustness and BN$_{adv}$ for clean accuracy, leads to a significant performance drop. As shown in Table 1, BN$_{clean}$ leads to almost zero robustness during inference. This interesting phenomenon inspires us to investigate the following question: *will BN$_{clean}$ achieve robustness if it is trained with the adversarial branch, and vice versa?* For facilitating discussion of the above misalignment, we introduce a new term **Cross-BN** which refers to adopting BN$_{clean}$ for the adversarial branch or BN$_{adv}$ for the clean branch. With a similar terminology rule, BN$_{clean}$ for the clean branch or BN$_{adv}$ for the adversarial branch is termed as **Self-BN**.

Cross-AT: a preliminary investigation. Before investigating Hybrid-AT with Cross-BN, we first investigate a setting where *only* adversarial samples are used for model training. Note that it is adversarial branch, and the baseline model with a Self-BN adopts BN$_{adv}$. Cross-AT is conducted by replacing the default BN$_{adv}$ with a Cross-BN, i.e., BN$_{clean}$ (see Figure 1). Specifically, the adversarial samples are normalized by the BN statistics calculated by clean samples. It should be noted that in Cross-AT, the clean samples are used only for forward propagation to get the BN statistics, and the model weights are updated only by the ad-

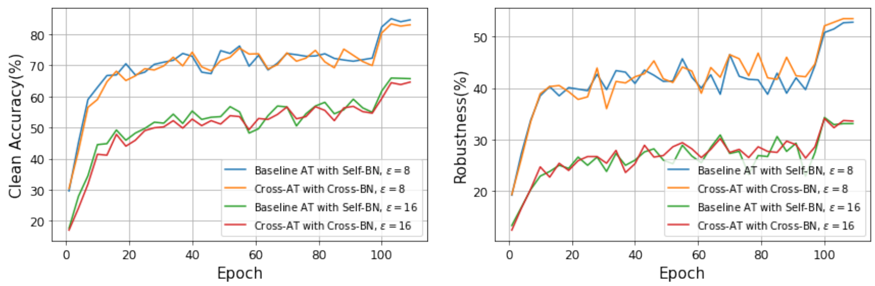

Figure 2: Clean accuracy and robustness (PGD10 Accuracy) of Cross-AT during training. In Cross-AT, the *adversarial* samples are normalized by the BN statistics calculated by *clean* samples. Interestingly, Cross-AT yield comparable robustness to original Self-BN(BN$_{adv}$).

Table 2: Test accuracy of Cross-Hybrid-AT ($\epsilon = 16/255$). In Cross-Hybrid-AT, the adversarial branch is normalized by BN$_{clean}$, and the clean branch is normalized by BN$_{adv}$. Experimental results show that Cross-Hybrid-AT achieves comparable results to Hybrid-AT with vanilla Dual BN.

Model	Training	Test	Clean	PGD-10	AA
Hybrid-AT	Dual BN	BN$_{adv}$	61.84	31.67	22.51
Cross-Hybrid-AT	Dual BN	BN$_{clean}$	59.56	31.25	22.40
Hybrid-AT	Single BN		93.70	29.86	0.48

versarial branch. Interestingly, although the adversarial branch is normalized by BN$clean$, Figure 2 shows that Cross-AT achieves comparable performance as the baseline model with Self-BN(BN$_{adv}$).

Cross-Hybrid-AT: Hybrid-AT with Cross-BN. Here, for the Dual BN in Hybrid-AT, we replace the default Self-BN with Cross-BN and term it Cross-Hybrid-AT. In Cross-Hybrid-AT, the adversarial branch is normalized by BN$_{clean}$, and the clean branch is normalized by BN$_{adv}$. As shown in Table 2, BN$_{clean}$ in Cross-Hybrid-AT achieves comparable results to BN$_{adv}$ in Hybrid-AT. The finding in Cross-Hybrid-AT is consistent with that in Cross-AT, which indicates that Cross-BN achieves comparable results to Self-BN.

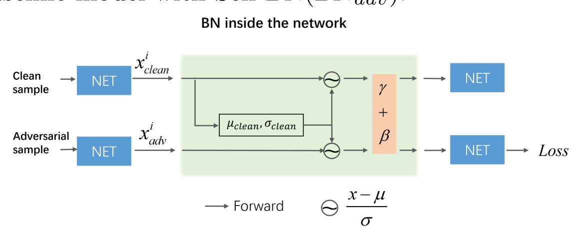

Figure 1: Cross-AT: Replacing BN$_{adv}$ with BN$_{clean}$ in the adversarial branch. The adversarial samples are normalized by the BN statistics calculated by clean samples.

Implication of the above results. As discussed above, training the model with Cross-BN leads to a comparable performance as with Self-BN in Hybrid-AT. However, this finding appears counter-intuitive considering the results of Hybrid-AT with Single BN. As shown in Table 2, Single BN leads to almost zero robustness (0.48%) under AA attack. Note that a single BN is calculated by a mixture of clean and adversarial samples. If calculating BN statistics on either clean examples or adversarial examples can lead to a high robustness, how come training on BN calculated on hybrid samples leads to an AA robustness close to zero? This motivates us to investigate how Dual BN works in Hybrid-AT.

4 Understanding how Dual BN works in Hybrid-AT

Towards understanding how Dual BN works in Hybrid-AT, we first revisit the existing claims in Xie & Yuille (2020); Xie et al. (2020a) which emphasize the importance of NS. However, these claims in Xie & Yuille (2020); Xie et al. (2020a) cannot explain our findings in Sec 3. We further investigate how Dual BN works through untwining NS and AP in Dual BN to include one effect while excluding the other, and propose a new explanation.

4.1 Revisiting existing explanation and our conjecture

Existing explanation. It is highlighted in Xie & Yuille (2020); Xie et al. (2020a) that disentangling NS is the key to justifying the merit of Dual BN in Hybrid-AT.

The underlying reasoning for disentangling NS is that BN statistics calculated on clean domain are incompatible with training the model on adversarial domain, and vice versa. Therefore, Hybrid-AT with single BN suffers from such incompatibility with the mixed distribution for calculating the normalization statistics. Meanwhile, it is claimed in Xie & Yuille (2020) that this incompatibility can be avoided by Dual BN through training the clean branch on BN$_{clean}$ and the adversarial branch on BN$_{adv}$. In Section 3, we conduct a preliminary investigation of this incompatibility and find that using BN from another domain only has limited influence on the performance. This finding conflicts with the claims in Xie & Yuille (2020); Xie et al. (2020a) that emphasize the importance of NS, and inspires us to investigate how Dual BN works in Hybrid-AT.

On top of the single BN as a default case, Dual BN introduces an auxiliary BN component and 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. Intuitively, disentangling NS avoids the influence of NS calculated on partial (half) samples from a different branch. However, we show that NS calculated on full samples from a cross-branch BN leads to comparable performance with that using the default BN (see Section 3). 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:

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.

4.2 Conjecture verification and additional investigation

Untwining NS and AP in Dual BN. As discussed above, compared with Hybrid-AT with Single-BN, 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 and we discuss the experimental results in Table 3 as follows.

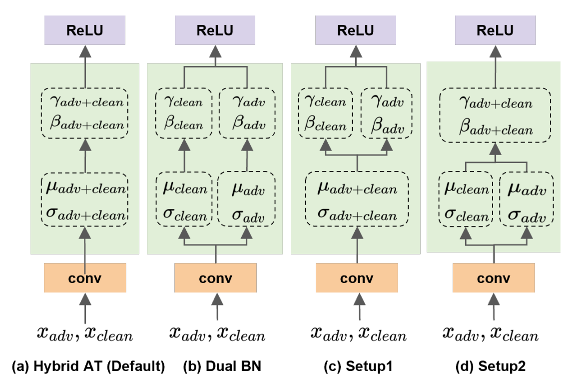

Figure 3: Illustration of different BN setups for untwining NS and AP in Dual BN of Hybrid-AT.

Two sets of AP characterize Dual BN. As shown in Table 3, Dual BN (with BN$_{adv}$ during inference) brings significant robustness improvement over the Single BN 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 Single BN achieves very low robustness (7.03% and 0.48% for $\epsilon = 8/255$ and 16/255, respectively). Moreover, Setup1 (AP$_{adv}$) achieves comparable robustness as that of Dual BN (BN$_{adv}$) for $\epsilon = 8/255$ and 16/255, suggesting two sets of APs alone achieve similar performance as Dual BN for yielding higher robustness (AP$_{adv}$) than single BN setting. The results collaborate with 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

Table 3: Test accuracy (%) of untwining NS and AP in Dual BN. For NSs, 1 indicates mixed distribution and 2 indicates disentangled distribution for normalization. For APs, 1 indicates single set and 2 indicates double sets of APs. The subscripts of AP_{adv} and AP_{clean} indicate the input data type used during training. Setup1 with two sets of APs achieves comparable results with Dual BN.

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})	2	1	85.49	49.39	42.96	55.91	21.92	10.64
Setup2 (NS_{clean})	2	1	89.22	49.48	42.95	86.35	1.08	0.00

fact that training under $\epsilon = 16/255$ is much harder than $\epsilon = 8/255$. Note that Setup2 (NS_{adv}) and Setup2 (NS_{clean}) achieve comparable robustness for $\epsilon = 8/255$, while they both fail to achieve high robustness for $\epsilon = 16/255$ (especially the AA results). Overall, we conclude two sets of APs are sufficient for avoiding the issue of low robustness against AA in Single BN setting of Hybrid-AT, and achieve comparable performance as Dual BN.

Further investigation beyond BN. Inspired by above finding that two sets of AP can achieve comparable results to Dual BN, we further investigate whether this holds in cases beyond BN where disentangling NS is not applicable. For example, layer normalization (LN) adopts sample-wise NS, and therefore it is not applicable to disentangle distribution-wise NS between two domains. We experiment with dual AP on ResNet with LN and the results are reported in Table 4. We observe that with Dual AP, LN performs similarly with BN in either setup (b) and (c) in Figure 3 (see the BN results of Dual BN and Setup1 in Table 3). We also investigate other normalization methods Table 4, e.g., group normaliztion and instance normalization, which show the same trend with LN and BN.

Table 4: Effect of dual AP on various types of normalizations ($\epsilon = 16/255$), where LN, GN and IN denote Layer Normalization, Group Normalization and Instance Normalization, respectively.

Norm	Setups	Branch	Clean	PGD10	AA
LN	Single AP	/	75.12	18.81	11.80
	Dual AP	AP_{adv}	62.56	26.98	16.90
	Dual AP	AP_{clean}	88.41	0.00	0.00
GN	Single AP	/	81.85	21.94	14.50
	Dual AP	AP_{adv}	70.27	29.36	18.30
	Dual AP	AP_{clean}	91.82	0.00	0.00
IN	Single AP	/	92.55	23.06	1.20
	Dual AP	AP_{adv}	52.29	25.27	16.10
	Dual AP	AP_{clean}	92.35	0.00	0.00

5 On the domain gap between clean and adversarial samples

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 the adversarial domain as a new domain. Prior work (Xie & Yuille, 2020) attributes the success of Hybrid-AT with Dual BN to disentangled NS. This claim is motivated by a two-domain hypothesis that clean and adversarial samples are from two different domains (Xie & Yuille, 2020; Xie et al., 2020a). As discussed in Section 4, we have pointed out that disentangled NS is not what makes Dual BN necessary. This finding further inspires us to investigate whether the underlying (two-domain) hypothesis for disentangling NS in Xie & Yuille (2020); Xie et al. (2020a) holds.

Revisiting the domain gap claimed in prior works. To highlight the two-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

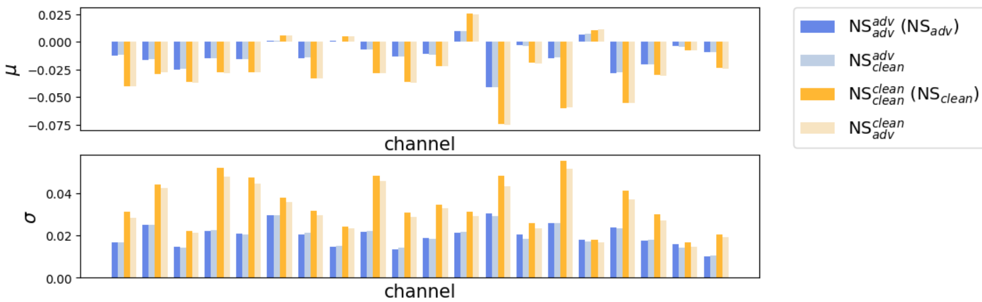

Figure 4: Visualization of normalization statistics (NS) by randomly choosing 20 channels and displaying the NS calculated with different APs. The superscript and subscript of NS refer to the AP and input images when calculating NS, respectively. For example, NS^{adv}_{clean} is computed on clean samples with AP_{adv}. NSs calculated by the same AP are close to each other, such as NS^{adv}_{adv} and NS^{adv}_{clean} calculated by AP_{adv}, so is similar $\text{NS}^{clean}_{clean}$ and NS^{clean}_{adv} calculated by AP_{clean}.

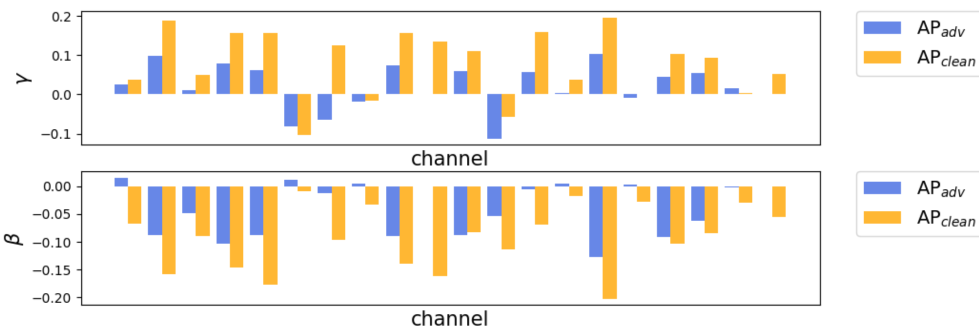

Figure 5: Visualization of affine parameters (AP). Randomly chose 20 channels for visualizing AP_{clean} and AP_{adv}. There exists a gap between AP_{clean} and AP_{adv}.

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, we know that the AP in BN_{clean} and BN_{adv} are 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 inconsistency and (b) different domain inputs. We discuss the influence of these two factors on the NS difference as follows.

5.1 A hidden flaw leads to a misleading visualization

A hidden flaw in prior visualization of NS. 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}. In order to analyze the influence of different AP and domain inputs on the NS, we additionally calculate the NS on clean samples with AP_{adv} (denoted as NS^{adv}_{clean}) and calculate the NS on adversarial samples with AP_{clean} (denoted as NS^{clean}_{adv}). These two NS are termed **re-calibrated NS** since the AP and inputs are from different branches. Following NS^{clean}_{adv} and NS^{adv}_{clean} to indicate AP choice with the superscript and indicate sample choice with the subscript, we can also denote vanilla NS_{clean} as $\text{NS}^{clean}_{clean}$ and denote NS_{adv} as NS^{adv}_{adv}. Both $\text{NS}^{clean}_{clean}$ and NS^{adv}_{adv} are termed as **vanilla NS** for differentiation. To exclude the influence of AP inconsistency, we intend to compare NS between clean and adversarial samples with the same AP (the superscript in NS). 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}

Table 5: Evaluation results of re-calibrated NS, $\epsilon = 16/255$. During inference, **re-calibrated** NS achieves comparable performance to the default setting.

Setups	NS	AP	$\epsilon = 8/255$		$\epsilon = 16/255$	
			PGD10	AA	PGD10	AA
Default	NS^{adv}_{adv}	AP_{adv}	51.33	46.19	31.67	22.51
	NS^{clean}_{clean}	AP_{clean}	0.32	0.10	0.00	0.00
Swap	NS^{clean}_{clean}	AP_{adv}	17.1	9.16	10.02	9.80
	NS^{adv}_{adv}	AP_{clean}	0.00	0.00	0.45	0.00
Re-calibration	NS^{adv}_{clean}	AP_{adv}	51.75	46.55	32.73	24.40
	NS^{clean}_{adv}	AP_{clean}	0.00	0.00	0.00	0.00

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 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^{adv}_{clean} and NS^{clean}_{clean}, as well as the gap between NS^{adv}_{adv} and NS^{clean}_{adv}. Second, those NSs with the same APs are very close to each other: NS^{adv}_{adv} and NS^{adv}_{clean} are very similar to each other, and the same applies for NS^{clean}_{adv} and 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. Specifically, the domain gap between clean samples and adversarial samples is not that large. The seemingly large domain gap is actually caused by the AP discrepancy in the dual BN setup. We report the visualization results of AP in Figure 5 for comparison, which shows a significant gap between AP_{clean} and AP_{adv}. For a quantitative comparison, we measure the Wasserstein distance between clean and adversarial branches in different layers in Figure 6. As shown in Figure 6, the Wasserstein distance of NS between clean and adversarial branches is much smaller than the difference of AP for a certain layer. This finding is consistent with that in Figure 4 and Figure 5.

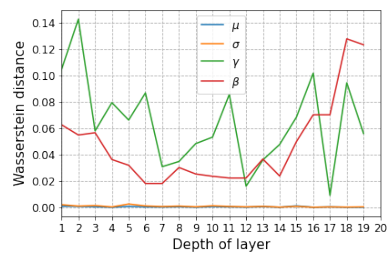

Figure 6: Layer-wise discrepancy visualization. For all layers, there exists a large distance (higher than zero) between AP_{clean} and AP_{adv}, see γ and β in the figure. However, with the same AP_{adv}, the gap between NS^{adv}_{adv} and NS^{adv}_{clean} stays almost zero in all layers, see μ and σ in the figure.

Inference evaluation with re-calibrated NS. We further evaluate the clean accuracy and robustness with re-calibrated NS during inference. Table 5 shows that given AP_{adv}, re-calibrated NS^{adv}_{clean} achieves a robustness of 51.75%, which is comparable to 51.33% with NS^{adv}_{adv}. Note that the only difference between NS^{adv}_{clean} and NS^{adv}_{adv} is that they are calculated by clean and adversarial samples, respectively. Moreover, given AP_{clean}, both NS^{clean}_{adv} and NS^{clean}_{clean} yield zero robustness. The results of swapping NS^{clean}_{clean} and NS^{adv}_{adv} when AP is fixed is also given in Table 5 for comparison. We conclude that AP characterizes the large robustness gap between BN_{clean} and BN_{adv} during inference, instead of NS as claimed in Xie & Yuille (2020). When AP is fixed, the robustness gap between the NS calculated on clean or adversarial samples is limited to be moderate.

5.2 Adversarial-clean domain gap *v.s.* 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

Table 6: Test accuracy (%) under random noise and adversarial perturbation during inference.

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

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.

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. The perturbation and noise magnitude are set to 16/255. Interestingly, we observe that there is no significant difference between the 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.

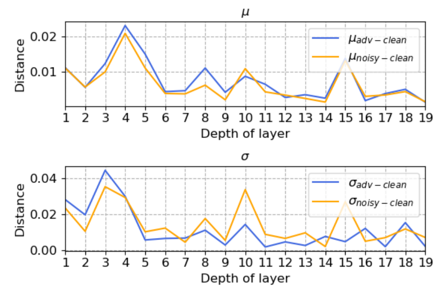

Figure 7: Visualization of adversarial-clean domain gap and noisy-clean domain gap (perturbation/noise magnitude is set to 16/255).

5.3 Interpreting Hybrid-AT from a two-task perspective

From two-domain to two-task. Considering the adversarial-clean domain gap is similar to noisy-clean domain gap as well as a strict constraint on allowable perturbation budget, future works investigating Hybrid-AT are suggested to discard the two-domain hypothesis. Akin to prior work justifying the role of disentangling NSs with the two-domain hypothesis, we provide a two-task hypothesis for justifying how Dual BN works in Hybrid-AT with a focus on the importance of disentangled 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. A common approach for handling two tasks with a shared backbone is to make the top layers unshared. Here, we experiment with a shared encoder of single BN but with dual linear classifiers. The results in Table 7 show that this setup results in similar behavior as Dual BN. Such a phenomenon corroborates that disentangling APs is equivalent to making partial learnable network weights not shared between the two tasks.

Table 7: Test accuracy of Hybrid-AT with Dual linear, $\epsilon = 16/255$. Hybrid-AT with Dual Linear achieves a similar trend and comparable results with Dual BN.

Setups	Branch	Clean	PGD10	AA
Dual BN	BN_{adv}	61.84	31.67	22.51
	BN_{clean}	94.18	0.00	0.00
Dual Linear	Linear_{adv}	60.72	28.84	16.50
	Linear_{clean}	91.43	2.21	1.30

Two sets of APs can be a double-edged sword. During inference, whether the test sample is clean or adversarial is unknown and only a single BN can be adopted. Prioritizing the robustness, prior work (Xie & Yuille, 2020) adopts BN_{adv} at test time at the cost of clean accuracy drop. Considering this, two sets of APs can be a double-edged sword. Our new perspective on Hybrid-AT enables alternative solutions to mitigate the two-task conflict without resorting to two sets of APs. Here, we experiment with including an additional regularization loss, which is introduced to minimize the gap between two tasks. As a concrete example, we add a KL loss on the basic loss of Hybrid-AT in Eq 2. The extra loss is designed to explicitly minimize the discrepancy between the outputs of adversarial and clean branches. Interestingly, this simple change improves the AA result of Single BN significantly from 0.48% to 23.60%. Compared to BN_{adv} (the default branch during inference), Hybrid-AT with KL loss achieves superior performance on both clean accuracy and robustness. Interestingly, the Hybrid-AT with KL loss reminds us of another AT framework termed Trades-AT (Zhang et al., 2019b), which is also trained on hybrid samples and has a KL loss. This might provide an explanation for the effectiveness of Trades-AT (Zhang et al., 2019b) by analyzing

Table 8: Test accuracy of Hybrid-AT (Single BN) with KL loss, $\epsilon = 16/255$.

Setups	Clean	PGD10	AA
Single BN	93.70	29.86	0.48
Single BN (with KL loss)	68.86	33.61	23.60
Dual BN (BN_{adv})	61.84	31.67	22.51
Dual BN (BN_{clean})	94.18	0.00	0.00

the KL term. Admittedly, KL loss on the output is just a naive attempt, but its promising result invites future works to explore other solutions.

6 Conclusion

We experiment with Cross-AT and demonstrate the compatibility of BN statistics of clean samples with the adversarial branch, which inspires us to doubt the motivation in prior work for justifying the necessity of Dual BN in Hybrid AT. We take a closer look at Dual BN and its underlying hypothesis, 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. In addition, we propose a new interpretation of Hybrid-AT with Dual BN from the two-task perspective. This work mainly focuses on providing a new understanding of Dual BN in AT. Understanding BN in other setups can be interesting future directions.

References

Maksym Andriushchenko and Nicolas Flammarion. Understanding and improving fast adversarial training. *NeurIPS*, 2020.

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

Tao Bai, Jinqi Luo, Jun Zhao, Bihan Wen, and Qian Wang. Recent advances in adversarial training for adversarial robustness. *arXiv preprint arXiv:2102.01356*, 2021.

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

Nils Bjorck, Carla P Gomes, Bart Selman, and Kilian Q Weinberger. Understanding batch normalization. *Advances in neural information processing systems*, 31, 2018.

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.

Pau de Jorge, Adel Bibi, Riccardo Volpi, Amartya Sanyal, Philip HS Torr, Grégory Rogez, and Puneet K Dokania. Make some noise: Reliable and efficient single-step adversarial training. *arXiv preprint arXiv:2202.01181*, 2022.

Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *CVPR*, 2018.

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.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 2016.

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.

Xiaojun Jia, Yong Zhang, Xingxing Wei, Baoyuan Wu, Ke Ma, Jue Wang, and Xiaochun Cao. Prior-guided adversarial initialization for fast adversarial training. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part IV*, pp. 567–584. Springer, 2022a.

Xiaojun Jia, Yong Zhang, Baoyuan Wu, Jue Wang, and Xiaochun Cao. Boosting fast adversarial training with learnable adversarial initialization. *IEEE Transactions on Image Processing*, 2022b.

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.

Alex Krizhevsky et al. Learning multiple layers of features from tiny images. ., 2009.

Sihan Li, Jiantao Jiao, Yanjun Han, and Tsachy Weissman. Demystifying resnet. *arXiv preprint arXiv:1611.01186*, 2016.

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.

Geon Yeong Park and Sang Wan Lee. Reliably fast adversarial training via latent adversarial perturbation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7758–7767, 2021.

Shilin Qiu, Qihe Liu, Shijie Zhou, and Chunjiang Wu. Review of artificial intelligence adversarial attack and defense technologies. *Applied Sciences*, 2019.

Shibani Santurkar, Dimitris Tsipras, Andrew Ilyas, and Aleksander Madry. How does batch normalization help optimization? In *NeurIPS*, 2018.

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

Ali Shafahi, Mahyar Najibi, Mohammad Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. Adversarial training for free! In *NeurIPS*, 2019.

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.

Sasha Targ, Diogo Almeida, and Kevin Lyman. Resnet in resnet: Generalizing residual architectures. *arXiv preprint arXiv:1603.08029*, 2016.

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.

Eric Wong, Leslie Rice, and J Zico Kolter. Fast is better than free: Revisiting adversarial training. *ICLR*, 2020.

Zifeng Wu, Chunhua Shen, and Anton Van Den Hengel. Wider or deeper: Revisiting the resnet model for visual recognition. *Pattern Recognition*, 2019.

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.

Cong Xu and Min Yang. Adversarial momentum-contrastive pre-training. *arXiv preprint arXiv:2012.13154*, 2020.

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*, 2021a.

Chaoning Zhang, Philipp Benz, Chenguo Lin, Adil Karjauv, Jing Wu, and In So Kweon. A survey on universal adversarial attack. *IJCAI*, 2021b.

Chaoning Zhang, Kang Zhang, Chenshuang Zhang, Axi Niu, Jiu Feng, Chang D Yoo, and In So Kweon. Decoupled adversarial contrastive learning for self-supervised adversarial robustness. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXX*, pp. 725–742. Springer, 2022.

Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P Xing, Laurent El Ghaoui, and Michael I Jordan. Theoretically principled trade-off between robustness and accuracy. In *ICML*, 2019b.

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*, 2019c.