ADVERSARIALLY TRAINED MODELS WITH TEST-TIME COVARIATE SHIFT ADAPTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Existing defense models against adversarial examples typically provide either empirical or certified robustness. Adversarially trained models empirically demonstrate state-of-the-art defense while providing no robustness guarantees for large classifiers or higher-dimensional inputs. In contrast, a randomized smoothing framework provides state-of-the-art certification while significantly degrades the empirical performance against adversarial attacks. In this work, we propose a novel *certification through adaptation* technique that transforms an adversarially trained model into a randomized smoothing classifier during inference to provide certified robustness for ℓ_2 norm without affecting their empirical robustness against adversarial attacks. One advantage of our proposed technique is that it allows us to separately choose the appropriate noise level for certifying each test example during inference. It also leads to outperform the existing randomized smoothing models for ℓ_2 certification on CIFAR-10. Therefore, our work is a step towards bridging the gap between the empirical and certified robustness against adversarial examples by achieving both using the same classifier for the first time.

1 INTRODUCTION

Deep neural network (DNN) based models are found to be brittle to minor, adversarially-chosen perturbations for their inputs that remain undetectable to human eyes. A DNN classifier that correctly classifies an image x, can be easily fooled by an *adversarial attack* to misclassify $x + \delta$ (Szegedy et al., 2014; Goodfellow et al., 2015; Madry et al., 2018). Here, δ is a minor *adversarial perturbation* such that the change between x and $x + \delta$ remains imperceptible.

Among the existing successful defense frameworks, *adversarial training* (AT) produces the best empirical robustness against the known adversarial attacks without providing any guarantee (Madry et al., 2018; Tramèr & Boneh, 2019; Zhang et al., 2019; Rice et al., 2020; Gowal et al., 2020). It trains a DNN classifier using strong adversaries from a specific class of perturbation (e.g., a small ℓ_p-norm) to provide robustness for the same perturbation types. Several certification techniques are proposed that can be applied to adversarially trained models to certifiably verify if the prediction of a test example, x remains constant within its neighborhood (Wong & Kolter, 2018; Wang et al., 2018; Salman et al., 2019b; Dvijotham et al., 2018; Gehr et al., 2018; Sheikholeslami et al., 2021). However, these certification techniques typically do not scale for larger networks (e.g., ResNet50) and datasets (e.g., IMAGENET). Hence, currently, we cannot guarantee that a more powerful, not yet known attack can not break these adversarially trained models. In fact, several recently proposed empirical defense models are later broken by stronger adaptive adversarial attacks, indicating the importance of investigating certified defenses with suitable robustness guarantees.

In contrast to adversarial training, *randomized smoothing* provides a scalable ℓ_2-certification framework for any classification model, which is robust against large isotropic Gaussian noise (Cohen et al., 2019; Salman et al., 2019a). However, the existing randomized smoothing-based certified models produce significantly lower empirical robustness compared to the AT models. On the other hand, this technique cannot be applied for AT models as they are not robust against such large random Gaussian noises in the standard settings. Towards this, we investigate to bridge the gap between the state-of-the-art empirical and certifiable robust models against adversarial examples.

In this paper, we present a novel *certification through adaptation* framework to transform an AT model into a randomized smoothing framework during inference, providing ℓ_2 certification without

ℓ_2 Radius (CIFAR-10)	0.25	0.5	0.75	1.0	1.25	1.5	1.75	2.0	ACR
Baseline	6.96	2.04	0.09	0.0	0.0	0.0	0.0	0.0	0.026
Rand$_{\sigma=0.5}$ (Cohen et al., 2019)	51.68	40.38	30.25	20.81	13.36	7.71	3.38	0.0	0.488
(Ours) Rand$_{\sigma=0.5}$ + adaptation	62.91	52.25	40.06	25.57	17.43	10.67	5.46	1.92	0.657
SmoothAdv$_{\sigma=0.5}$ (Salman et al., 2019a)	58.82	49.68	42.68	37.55	32.64	27.52	22.42	0.0	0.918
(Ours) SmoothAdv$_{\sigma=0.5}$ + adaptation	59.89	50.4	41.76	35.5	30.92	26.1	20.25	15.05	1.008
Adv$_\infty$ (Rice et al., 2020)	35.95	29.44	23.51	0.0	0.0	0.0	0.0	0.0	0.317
(Ours) Adv$_\infty$ + adaptation	67.96	55.06	43.27	30.55	24.68	18.49	12.11	8.45	0.903
Adv$_2$ (Rice et al., 2020)	41.89	34.15	26.7	0.0	0.0	0.0	0.0	0.0	0.359
(Ours) Adv$_2$ + adaptation	**68.84**	**58.77**	**49.71**	**37.74**	**33.37**	**28.82**	**23.65**	**18.23**	**1.198**
MARCER$_{\sigma=0.5}$ (Zhai et al., 2020)	60.0	53.0	46.0	38.0	29.0	19.0	12.0	0.0	0.726
Consistancy$_{\sigma=0.5}$ (Jeong & Shin, 2020)	48.9	45.1	41.3	37.8	33.9	29.9	25.2	0.0	0.726

Table 1: CIFAR-10: Certified accuracy at various ℓ_2 radii and ACR scores. We train different models by varying the hyper-parameters for SmoothAdv, Adv$_2$ and Adv$_\infty$ (as in (Salman et al., 2019a)) and by choosing $\sigma = \{0.25, 0.5, 0.75\}$ for test-time adaptation to obtain the maximum certified radii for each test example. See Table 5 and 6 (Appendix) for detailed results on both IMAGENET and CIFAR-10 respectively. We also present the best reported results for MARCER and Consistancy at $\sigma = 0.5$, obtained from their respective papers.

any additional training or architectural modification. Our proposed certification technique consists of two steps: we first apply a covariate shift adaptation to a classifier against Gaussian noise during inference for each test example (Cariucci et al., 2017; Li et al., 2016). For our paper, we use the well-known batch normalization adaptation. This process significantly boosts the performance of the AT models against the random isotropic Gaussian noises compared to the standard non-robust models. Hence, we can now directly apply the *randomized smoothing* based certification technique to provide ℓ_2 certification in the next step. Further, the existing randomized smoothing models require selecting the noise level at training time. In contrast, our proposed framework can separately choose the appropriate noise levels for different test examples during inference (Figure 4). Furthermore, we can also evaluate the input test examples without transforming the AT models to a randomized smoothing model, ensuring that their empirical performance remains unaffected. Therefore, we are the first to provide the *test-time flexibility* to obtain empirically robust predictions as well as certify their predictions using the same classifier for high-dimensional datasets to the best of our knowledge. Hence, we improve the reliability of AT models sensitive real-world applications.

Contributions:-

1. We propose a novel *certification through adaptation* framework that can adapt an AT model during inference to provide certified robustness. Our experimental results on CIFAR-10 and IMAGENET demonstrate that the proposed certification framework can transform any AT model into a randomized smoothing classifier to provide certification for ℓ_2 norm, even when the model is learned using ℓ_∞-bounded adversaries (Table 1 & Figure 2).
2. One main advantage of our proposed framework is that it allows us to select appropriate noise levels for different test examples during inference. This leads to outperforming the existing state-of-the-art randomized smoothing models for ℓ_2 certification on CIFAR-10 using AT models (Table 1 & Figure 4). Further, we can provide certification at larger ℓ_2 radii for existing randomized smoothing models, improving their overall average certified radius (ACR).
3. Our results also indicate a strong correlation between empirical and certified robustness than previously believed (Cohen et al., 2019; Salman et al., 2019a; Tramèr & Boneh, 2019). In particular, we observe that the empirically stronger AT models lead to better ℓ_2 certification performance (Figure 5).

2 RELATED WORK

Empirical Defenses and Adversarial Training. Existing defense models against adversarial attacks can be broadly classified into *empirical* and *certified* defenses. Empirical defenses demonstrate empirical robustness against adversarial attacks (Schott et al., 2019; Moosavi Dezfooli et al., 2019; Nandy et al., 2020; Mao et al., 2021). *Adversarial training* achieves the state-of-the-art empirical defense (Madry et al., 2018). It optimizes the following loss function for a DNN classifier, f, to provide robustness within an ϵ-bounded *threat model* for an ℓ_p norm, where the perturbations, $\delta \in \Delta$ are constrained as $\Delta = \{\delta : ||\delta||_p \leq \epsilon\}$:

$$\min_{\theta} \mathbb{E}_{(x,y)}[\max_{\delta \in \Delta} \mathcal{L}(f_\theta(x+\delta), y)] \tag{1}$$

where, θ denotes the model parameters. $\mathcal{L}$ is the classification loss.

The *inner maximization* in Eq. 1 is solved by producing adversarial examples using strong iterative adversaries, e.g., *projected gradient descent (PGD)* attack (Kurakin et al., 2016; Madry et al., 2018). Wong et al. (2020) found that even a single-step *fast gradient sign method (FGSM)* attack-based AT models also achieves high empirical robustness (Goodfellow et al., 2015). Zhang et al. (2020) proposed to use the least adversaries for training. Recently Trades (Zhang et al., 2019), Adv-LLR (Qin et al., 2019) introduced additional regularizers to achieve higher empirical robustness by smoothing the loss surface. However, Rice et al. (2020) showed that the standard PGD based AT model with early-stopping criteria provides one of the best empirical defenses for a given perturbation type. Recent works also explored the importance of different hyper-parameters for adversarial training (Gowal et al., 2020; Pang et al., 2021) as well as incorporating additional data in a semi-supervised fashion (Carmon et al., 2019; Uesato et al., 2019) to further improve their empirical robustness.

Certified Defenses. Empirical defenses demonstrate robustness only against the *known* adversaries without providing any guarantees. In fact, most empirical defenses proposed in the literature were later *broken* by stronger adversaries, highlighting the importance of certified defenses to provide robustness guarantees (Athalye et al., 2018; Uesato et al., 2018; Jalal et al., 2019).

Several recent works proposed to train neural network models with provable robustness guarantees. These works include methods based on semi-definite relaxations (Raghunathan et al., 2018), linear relaxations and duality (Wong & Kolter, 2018; Wong et al., 2018), abstract interpretation (Mirman et al., 2018), and interval bound propagation (Gowal et al., 2018). Parallel to training a certified defense, several works also focus on certifying the already trained models (Tjeng et al., 2017; Gehr et al., 2018; Weng et al., 2018; Wang et al., 2018; Bunel et al., 2018). Recently Mueller et al. (2021) combined a *small* certification network with a *large*, empirically robust AT model using some selection criteria to boost overall benign accuracy along with empirical robustness for the certified framework. However, none of these techniques scale for large networks (e.g., ResNet50) or higher-dimensional datasets (e.g., IMAGENET).

Randomized Smoothing for Certification. A randomized smoothing classifier is not a neural network. It uses a neural network as its base for classification. Randomized smoothing was initially proposed as a heuristic defense (Cao & Gong, 2017; Liu et al., 2018) and later shown to be certifiable (Lecuyer et al., 2019; Li et al., 2019). Recently, Cohen et al. (2019) and Salman et al. (2019a) separately provided a tight robustness guarantee for ℓ_2-norm. Salman et al. (2019a) provides the current state-of-the-art ℓ_2 certification robustness by adversarially choosing the noise using an adaptive attack to train their base classifier. This framework is also analyzed for other ℓ_p norms using different noise distributions as well (Li et al., 2019; Lee et al., 2019; Dvijotham et al., 2020; Yang et al., 2020). Salman et al. (2020) proposed to incorporate an additional denoising module as a preprocessing unit to convert a standard DNN classifier into a randomized smoothing model to provide non-trivial certified robustness. Notably, randomized smoothing is the only scalable certification framework. Further, it also achieves superior performance for different perturbation types.

While achieving the state-of-the-art certification performance, randomized smoothing significantly degrades the empirical robustness against adversarial attacks (Lecuyer et al., 2019; Salman et al., 2019a; Cohen et al., 2019). Towards this, our proposed technique transforms an AT model into a randomized smoothing classifier without any additional training or architectural modification. Since AT models already provide the state-of-the-art empirical defense, we achieve both empirical and certified robustness against adversarial examples using the same classifier.

3 PROPOSED METHODOLOGY

In this section, we first present the background of the randomized smoothing technique and explain why it is not directly effective for AT models. Next, we present the existing test-time co-variate shift adaptation for domain adaptations and corruption robustness. Then, we present our proposed *certification through adaptation* framework that adapts a DNN model during inference to provide certified robustness without additional training or architectural modification.

3.1 BACKGROUND ON RANDOMIZED SMOOTHING

Consider a classifier f that maps inputs in $\mathbb{R}^d$ to $\mathcal{Y}$ classes. The randomized smoothing framework transforms the original base classifier f into a new, smoothed classifier g. In particular, for an

input $x \in \mathbb{R}^d$, the smoothed classifier g returns the most probable class to be predicted by the base classifier f under isotropic Gaussian noises of x. That is,

$$g(x) = arg\max_{y \in \mathcal{Y}} \mathbb{P}(f(x+\delta) == y) \qquad \text{where, } \delta \sim \mathcal{N}(0, \sigma^2 I). \tag{2}$$

The noise level, σ controls the trade-off between robustness and accuracy: Increasing σ would improve the robustness of g at higher ℓ_2 radii. However, it degrades the robustness at lower ℓ_2 radii as well as the benign accuracy.

Cohen et al. (2019) presented a tight robustness guarantee based on the Neyman-Pearson lemma for the smoothed classifier g and gave an efficient algorithm using Monte Carlo sampling for certifying of g. We can also obtain this guarantee alternatively by explicitly computing the Lipschitz constant of the smoothed classifier as shown in (Salman et al., 2019a; Yang et al., 2020). The certification procedure is as follows: Suppose a base classifier f classifies $\mathcal{N}(x, \sigma^2 I)$ to return the *"most probable"* class, c_A with probability $p_A = \mathbb{P}(f(x+\delta) == c_A)$ and the *"runner-up"* class c_B with probability $p_B = \max_{y \neq c_A} \mathbb{P}(f(x+\delta) == y)$. Then, the smooth classifier, g is certifiably robust around x within an ℓ_2 radius of R:

$$R = \frac{\sigma}{2}\Big(\Phi^{-1}(p_A) - \Phi^{-1}(p_B)\Big) \tag{3}$$

where, Φ^{-1} is the inverse of the standard Gaussian CDF.

However, computing the exact values of p_A and p_B is not possible in practice when f is a DNN. Cohen et al. (2019) addressed this problem using Monte Carlo sampling to estimate some $\underline{p_A}$ and $\overline{p_B}$ such that $\underline{p_A} \leq p_A$ and $\overline{p_B} \geq p_B$ with arbitrarily high probability. The certified radius for input x is then computed by replacing p_A and p_B with $\underline{p_A}$ and $\overline{p_B}$ respectively in Eq. 3.

As we can see in Equation 2 that the original base classifier, f needs to be robust against large Gaussian noises to provide non-trivial robustness certification results. Otherwise, it leads to lower p_A and hence a lower certification of R for the test examples. Existing randomized smoothing-based models applies custom-trained using explicit Gaussian noises to learn their original base classifier (Lecuyer et al., 2019; Cohen et al., 2019; Salman et al., 2019a; Zhai et al., 2020; Jeong & Shin, 2020). However, these models produce significantly lower empirical robustness compared to the AT models. Consequently, AT models are not robust against large Gaussian noises in the standard inference settings (see Table 2). Hence, we cannot directly use them as the base classifier for randomized smoothing.

3.2 Background on Covariate Shift Adaptation

Recent works on (Sun et al., 2017; Roy et al., 2019; Huang et al., 2018; Li et al., 2016) and corruption robustness (Schneider et al., 2020; Nado et al., 2020; Benz et al., 2021) demonstrate the importance of unsupervised covariate shift adaptation. We use adaptive batch-normalization (BN), one of the most popular and effective unsupervised covariate shift adaptation mechanisms.

A BN layer computes the mean and variance of the hidden activation maps across the channels to normalize these activations to $\mathcal{N}(0, 1)$ before feeding into the next hidden layer (Ioffe & Szegedy, 2015). It reduces the dependencies among different hidden layers, improving the training efficiency for deep architectures. Hence, most of the recent DNN architectures frequently incorporate BN layers for complex machine learning tasks. However, the distributional shifts in the test examples lead to different activation statistics compared to the training examples. Hence, impacted by the covariate shift, the statistics estimated during training fail to normalize the activation tensors to $\mathcal{N}(0, 1)$. As a result, it breaks the crucial assumption for the subsequent hidden layers to work.

More formally, let $P_T : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}^+$ as the training distribution and $P_t : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}^+$ as the test distribution; where $x \in \mathcal{X}$ are inputs and $y \in \mathcal{Y}$ are the corresponding class labels. There exists covariate shift between training and test distribution iff: $P_T(y|x)$=$P_t(y|x)$ and $P_T(x) \neq P_t(x)$ (Sugiyama & Kawanabe, 2012; Schölkopf et al., 2012). If the covariate shift only affects the first and second-order moments of the hidden layer feature activations, $f_h(x)$, we can remove it using normalization (Schneider et al., 2020):

$$P_T\Big(\frac{f_h(x) - \mathbb{E}_T[f_h(x)]}{\sqrt{\mathbb{V}_T[f_h(x)]}}\Big) P_T(x) \approx P_t\Big(\frac{f_h(x) - \mathbb{E}_t[f_h(x)]}{\sqrt{\mathbb{V}_t[f_h(x)]}}\Big) P_t(x). \tag{4}$$

Covariate shift adaptation using adaptive BN computes the BN statistics from the feature activations, μ_t, s_t^2, of the test batch. We can adapt them with the existing *training* statistics, μ_T, s_T^2, obtained using the training batches as (Cariucci et al., 2017; Li et al., 2016; Schneider et al., 2020):

$$\overline{\mu} = \rho \cdot \mu_t + (1-\rho) \cdot \mu_T \quad \overline{s} = \rho \cdot s_t + (1-\rho) \cdot s_T \tag{5}$$

where, $\rho \in [0, 1]$ is the momentum. The choice of $\rho = 0$ is equivalent to the standard inference setup with a deterministic DNN classifier in the IID settings. We should choose $\rho = 1$ when receiving larger test batches as it can provide a better estimation of the test distributions.

Assumptions for BN adaptation. It is noteworthy that these existing adaptive BN-based frameworks require a *large set of test images* from the same covariate shift to estimate the BN parameters. However, this assumption may not hold for several real-world applications, e.g., stateless web APIs. Also, these test images should be *semantically diverse*, preferably over multiple classes, to effectively estimate the test distributions. Hence, it further limits the practical usability of these frameworks for real-world applications, e.g., autonomous cars.

In contrast to these models for domain adaptation and corruption robustness, our proposed certification framework against adversarial examples does not make any such assumptions. In this case, we already know the perturbation type on which we need to adapt the model to provide the certification. Hence, we can explicitly pre-select a diverse set of clean images, $\mathbf{X}_{batch}$ and control the perturbations to adapt the models, addressing both of these limitations.

Algorithm 1: Steps for CERTIFICATION THROUGH ADAPTATION Algorithm

Input: f: classifier, x_{test}: test example, σ: desired noise-level, $\mathbf{X}_{batch}$: set of clean images (preselected from validation data or test stream).

Output: Certifiably robust ℓ_2 radius of R for x_{test}.

/* Step 1: Adapt BN parameters using $\mathbf{X}_{batch}$ with $\rho = 1$ (Eqn 5). */

$\tilde{\mathbf{X}}_{batch} = [x + \mathcal{N}(0, \sigma I) \ \forall \ x \in \mathbf{X}_{batch}]$ // perturb $\mathbf{X}_{batch}$ with desired noise.

f_{adapt} = CLONE($f.train()$) // clone f with *train-mode*.

$\_ = f_{adapt}(\tilde{\mathbf{X}}_{batch})$ // forward pass for BN parameter adaptation.

$f_{adapt}.eval()$ // fix the parameters.

/* Step 2: Certify x_{test} using Randomized Smoothing framework. */

g = GETRANDOMIZEDMODEL(f_{adapt}) // Convert f_{adapt} to randomized-smoothing classifier g (Eqn 2).

R = CERTIFY(g, x_{test}, σ) // Execute 3 for ℓ_2 certification.

return R

3.3 PROPOSED CERTIFICATION THROUGH ADAPTATION

The robustness guarantee in Eq. 3 suggests that randomized smoothing gives a framework for certifying any classifier f that is robust against large Gaussian noises. Previous works proposed customized training using explicit Gaussian noise augmentation for their training (Section 3.1). Subsequently, in Section 3.2 we note that robustness against random Gaussian noises of any classifier, f can be improved by applying covariate shift adaptation using adaptive BN technique without any additional training. However, it modifies the original base classifier f at each forward pass by recomputing the BN parameters. Since the certification guarantee in Eq. 3 is provided only for a fixed base classifier f, we cannot directly apply adaptive BN to provide ℓ_2 certification using the randomized smoothing framework. This motivates us to propose a novel certification framework that applies the covariate shift adaptation using adaptive BN as an *offline pre-processing step* to improve the robustness against random Gaussian noises, addressing the above problem.

Our proposed certification through adaptation framework consists of two steps: Given test image x_{test}, we first apply the adaptive BN technique to achieve robustness against Gaussian perturbations. Recall that adaptive BN requires a large set of diverse test images to correctly re-estimate the batch-normalization statistics. However, to provide certification for ℓ_2-norm, we only need to adapt our model against Gaussian perturbations. Hence, we can pre-select a sufficiently large set of diverse clean images, $\mathbf{X}_{batch}$ and apply Gaussian perturbations to adapt our classifier, f, as an offline pre-processing step to obtain f_{adapt}. Alternatively, when a large set of diverse test examples

are available, we can also use them for our BN adaptation. The Gaussian noise samples should be drawn from the same isotropic Gaussian distribution $\mathcal{N}(0, \sigma^2 I)$ as we need to use for the certification process. Then, we freeze the model parameters and use the adapted model, f_{adapt}, as our base classifier to certify the test example, x_{test}. Hence, the base classifier f_{adapt} remains fixed during calculating the certification radius R (Equation 3). Our proposed *certification through adaptation* technique is presented in Algorithm 1.

Advantages. The main advantage of our proposed framework is that we can adapt the classifier, f at any noise level σ as an offline pre-processing step, without any additional training (see Figure 4). As we can see in Equation 3, that we should select a large σ to certify at a bigger ℓ_2 radius of R. However, a test image that does not remain robust at higher σ produces a lower value of p_A. It leads to reducing the overall certification radius, R. Hence, providing the flexibility of choosing appropriate noise levels for different test examples allows us to improve the certification radius, R.

In contrast to our proposed framework, existing randomized smoothing frameworks cannot choose a different σ at test-time since it typically degrades their overall certification performance. Hence, they need to fix σ during training their base models or its components.

Applicability. Our proposed certification through adaptation technique can be applied to any classification model, f with batch-normalization layers. However, note that achieving high accuracy against large random Gaussian perturbations is only a necessary condition: a randomized smoothing classifier, g requires to *consistently* predict the correct class to provide higher certification guarantees at larger radii. Hence, we achieve non-trivial ℓ_2 certification guarantees at very small ℓ_2 radii for standard non-robust DNN classifiers (see Appendix B.1).

On the other hand, for existing randomized smoothing models, we achieve higher certification at larger ℓ_2 radii by adapting their base models with larger σ, improving their overall *average certified radius (ACR)* (Table 6 and 5 (Appendix)). However, we could not find any σ to obtain a significant improvement at lower ℓ_2 radii. In contrast, AT models with our proposed offline adaptation technique significantly improve their performance against large Gaussian perturbations, providing non-trivial certification robustness. Experimentally we find that our proposed technique outperforms the state-of-the-art certification models for the ℓ_2 norm.

Finally, while we focus on adaptive BN, there also exists other unsupervised covariate shift adaptation techniques such as self-supervised domain adaptation on single test examples (Sun et al., 2020), pseudo-labeling (French et al., 2017; Xie et al., 2020) etc. Wang et al. (2020) also proposed to update the normalization parameters by entropy minimization to improve the corruption robustness. Future studies may also explore these techniques for the offline pre-processing step.

4 EXPERIMENTS

Experimental setup. We use CIFAR-10 (Krizhevsky et al., 2009) and IMAGENET (Deng et al., 2009) datasets for our experiments. For CIFAR-10, we use pre-activation ResNet18 and ResNet50 for IMAGENET (He et al., 2016a;b). Our AT models are trained using *early stopping* criteria (Rice et al., 2020) as follows: For IMAGENET, we use two AT models, $\text{Adv}_\infty[\ell_\infty \leq 4/255]$ and $\text{Adv}_2[\ell_2 \leq 3]$, learned at ℓ_∞ and ℓ_2 threat models with threat boundaries of $4/255$ and 3 respectively. For CIFAR-10, we train multiple AT models with different threat boundaries. For example, we denote $\text{Adv}_\infty[\ell_\infty \leq 8/255]$ and $\text{Adv}_2[\ell_2 \leq 1]$ as the AT models for ℓ_∞ and ℓ_2 threat models, trained with threat boundaries of $8/255$ and 1, respectively. We compare with Baseline and $\text{Rand}_{\sigma=0.5}$ models. Baseline models are trained using clean images. $\text{Rand}_{\sigma=0.5}$ models are trained by augmenting random noise, sampled from isotropic Gaussian distribution, $\mathcal{N}(0, \sigma^2 I)$ with $\sigma = 0.5$. We also compare with the current state-of-the-art certification models, SmoothAdv for CIFAR-10 (Salman et al., 2019a). Please refer to Appendix A for more details. [1]

4.1 PERFORMANCE UNDER GAUSSIAN NOISE.

We first investigate the performance of different classification models under significantly larger Gaussian perturbations. It is a necessary condition to provide ℓ_2 robustness certification. In Table 2,

[1] For IMAGENET, we obtain Adv_∞ and Adv_2 from https://github.com/locuslab/robust_overfitting and Baseline and $\text{Rand}_{\sigma=0.5}$ models from https://github.com/locuslab/smoothing.

(a) IMAGENET

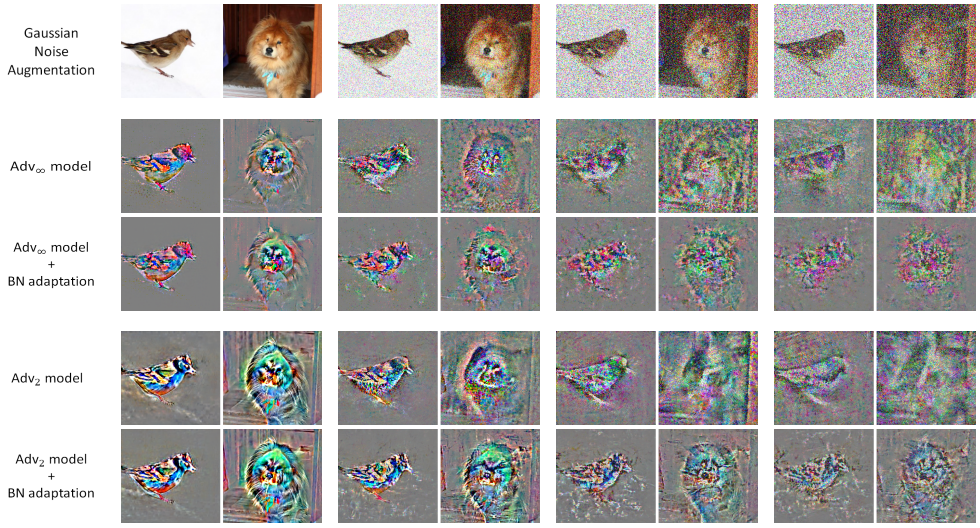
Model	$\sigma = 0$	$\sigma = 0.25$	$\sigma = 0.5$	$\sigma = 0.75$
Baseline	75.2±0.0	11.8±0.22	0.3±0.01	0.1±0.0
+ adaptive BN	74.4±0.04	31.0±0.27	7.7±0.24	2.4±0.01
$\text{Adv}_\infty[\ell_\infty \leq 4/255]$	62.8±0.0	3.9±0.03	0.4±0.01	0.2±0.01
+ adaptive BN	60.8±0.16	53.4±0.15	44.9±0.08	33.7±0.28
$\text{Adv}_2[\ell_2 \leq 3]$	59.8±0.0	9.8±0.08	0.9±0.01	0.3±0.0
+ adaptive BN	58.3±0.08	53.7±0.14	47.3±0.14	39.8±0.18
$\text{Rand}_{\sigma=0.5}$	22.0±0.0	32.8±0.11	60.9±0.04	0.9±0.06
+ adaptive BN	62.7±0.03	62.3±0.18	59.5±0.11	51.4±0.27

(b) CIFAR-10

Model	$\sigma = 0$	$\sigma = 0.25$	$\sigma = 0.5$	$\sigma = 0.75$
Baseline	95.2±0.0	10.9±0.88	10.6±0.76	10.5±1.19
+ adaptive BN	95.0±0.57	40.1±0.97	22.0±0.83	17.2±0.66
$\text{Adv}_\infty[\ell_\infty \leq 8/255]$	82.1±0.0	40.2±4.56	16.1±7.85	12.2±5.23
+ adaptive BN	81.6±0.96	74.2±0.95	62.4±0.64	51.0±1.03
$\text{Adv}_2[\ell_2 \leq 1]$	81.6±0.0	47.5±5.1	21.5±7.79	14.3±5.63
+ adaptive BN	81.8±0.7	75.8±0.43	64.9±0.73	53.5±1.71
$\text{Rand}_{\sigma=0.5}$	66.7±0.0	69.1±1.01	61.2±0.84	25.9±1.41
+ adaptive BN	74.0±2.1	73.0±2.04	66.8±2.01	56.7±0.94

Table 2: Top-1 accuracy of different classifiers under different levels of Gaussian noises augmented to the test images. We randomly shuffle test images and sample the noises and report $(mean \pm 2 \times sd)$ of five runs.

we present the performance. We observe that when the test examples are sampled from IID settings as training distributions (i.e., $\sigma = 0$ for Baseline, Adv_∞, and Adv_2 and $\sigma = 0.5$ for $\text{Rand}_{\sigma=0.5}$), these models produces the best results regardless of whether BN adaptation is applied. However, as we move away from the IID settings by increasing (or decreasing) σ, the performance of all these models significantly degrades in the standard inference setup. In contrast, covariate shift adaptation using adaptive BN improves the performance for all models. In particular, AT models achieve significantly higher performance gain using adaptive BN than the non-robust baseline models at higher noise levels. For example, at $\sigma = 0.5$, Baseline, $\text{Adv}_2[\ell_2 \leq 3]$ and $\text{Adv}_\infty[\ell_\infty \leq 4/255]$ respectively achieve top-1 accuracy of 0.3%, 0.4%, and 0.9% for IMAGENET without using BN adaptation (Table 2 (a)). However, adaptive BN for $\text{Adv}_2[\ell_2 \leq 3]$ and $\text{Adv}_\infty[\ell_\infty \leq 4/255]$ significantly improves the top-1 accuracy to 47.3% and 44.9% respectively. In contrast, the baseline model only achieves 7.7% accuracy. We observe similar results for CIFAR-10 in Table 2 (b).



(a) $\sigma = 0$ (Clean) (b) $\sigma = 0.25$ (c) $\sigma = 0.5$ (d) $\sigma = 0.75$

Figure 1: Visualizing loss-gradients produced by AT models as we apply different levels of Gaussian noises.

Loss Gradients under Gaussian Noises. To further investigate the performance of AT models, we visualize the *loss gradients* for individual pixels of an image as we increase the Gaussian noise (i.e., σ) (Figure 1). *Loss-gradients* reflect the most relevant input pixels for classification predictions. Here, we scale, translate and clip the loss-gradient values without using any sophisticated techniques (as suggested in Tsipras et al. (2019)). At $\sigma = 0$ (i.e., for clean images), the loss-gradients from AT models align properly with perceptually relevant features (as observed previously (Tsipras et al., 2019; Etmann et al., 2019)). However, as we choose higher noise using σ=0.5 and σ=0.75, the overall loss gradients become noisier. Specifically, AT models without adaptation produce sharper loss gradients (i.e., greater importance) even for background pixels. In contrast, test-time BN adaptation produces gradients for the pixels from the object of interest and suppress the gradients for background pixels (see Figure 1(c) and Figure 1(d)). Hence, they extract the required semantic information for correct classifications. It is interesting to note that Adv_2 produces significantly more human-aligned loss gradients compared to Adv_∞. This behavior is also reflected in their classifica-

tion (Table 2) and overall certification (Table 1) as we note that Adv_2 overall produces much better performance compared to Adv_∞.

4.2 CERTIFICATION USING RANDOMIZED SMOOTHING

We now present the ℓ_2 certification results using the randomized smoothing framework as the backbone, as proposed in our Algorithm 1. We certify the test images with 99.9% probability. We estimate the class label probabilities of g (in Equation 3) using Monte-Carlo sampling with $100,000$ noisy samples for each test image, as in Cohen et al. (2019); Salman et al. (2019a). We use the full test-set for CIFAR-10 and a sub-sample of 500 test images for IMAGENET (as in Cohen et al. (2019)). We provide the detailed results of *certified accuracy* along with *average certified radius (ACR)* for several models, trained using different specifications and adapting with different σ in Table 5 and 6 (Appendix).

Certifying AT models. In Figure 2, we first demonstrate that AT models can provide non-trivial ℓ_2 certified robustness using our proposed framework for both IMAGENET and CIFAR-10 datasets. Here, we use $\text{Adv}_\infty[\ell_\infty \leq 4/255]$ and $\text{Adv}_2[\ell_2 \leq 3]$ for ImageNet and $\text{Adv}_\infty[\ell_\infty \leq 8/255]$ and $\text{Adv}_2[\ell_2 \leq 1]$ for CIFAR-10 and use $\sigma = 0.5$ for adaptation and certification using Algorithm 1. We compare with the certification results of Baseline, Adv_∞ and Adv_2 models in the standard settings, without using any adaptation and certified at $\sigma = 0.25$. We can see a significant boost of ℓ_2 certification results for both Adv_∞ and Adv_2 models using our proposed framework. Further, Adv_2 models consistently achieve better performance compared to Adv_∞ in terms of certified accuracy. For CIFAR-10, both $\text{Adv}_\infty[\ell_\infty \leq 8/255]$ and $\text{Adv}_2[\ell_2 \leq 1]$ outperform the standard randomized smoothing framework i.e., $\text{Rand}_{\sigma=0.5}$, certified using $\sigma = 0.5$ (Cohen et al., 2019). For IMAGENET, $\text{Adv}_2[\ell_2 \leq 3]$ achieves better certified accuracy compared to $\text{Rand}_{\sigma=0.5}$ beyond ℓ_2-radii of 1.5.

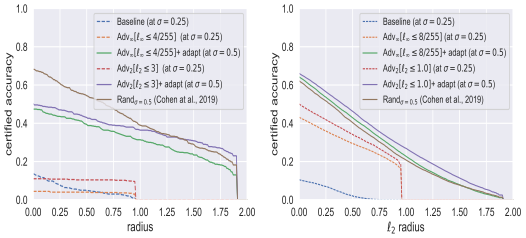
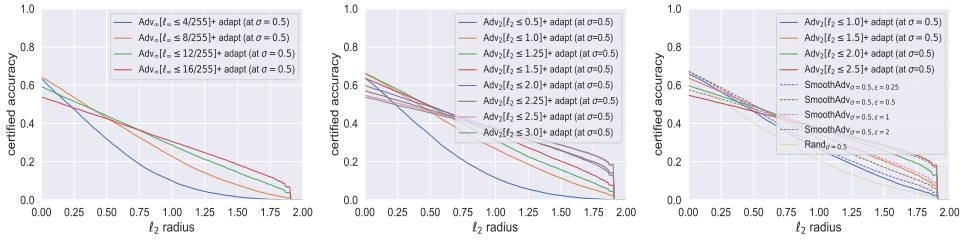


Figure 2: Certified top-1 accuracy at various ℓ_2 radii for **(Left)** IMAGENET using ResNet-50 and **(Right)** CIFAR-10 using preactivation ResNet-18.



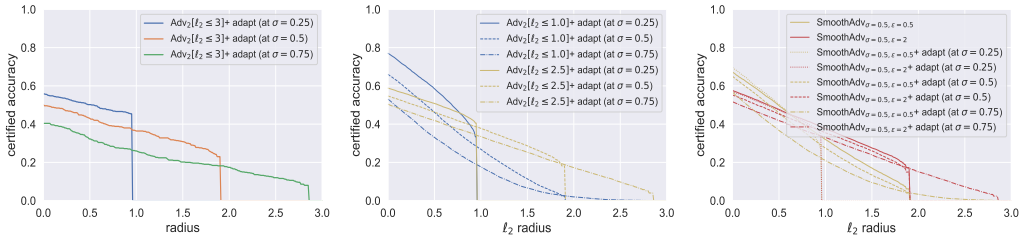
(a) (b) (c)

Figure 3: CIFAR-10: Certified top-1 accuracy achieved by (a) Adv_∞ and (b) Adv_2 models (with test-time adaptive BN at $\sigma = 0.5$), learned at different threat boundaries. (c) Comparison with the state-of-the-art SmoothAdv models (Salman et al., 2019a), trained at $\sigma = 0.5$ using preactivation ResNet-18.

Larger Threat Boundary for Better Certified Robustness. Learning AT models at a higher threat boundary improves the certification accuracy at higher ℓ_2 radii. We demonstrate this phenomena for both Adv_∞ and Adv_2 models in Figure 3(a) and 3(b) respectively for CIFAR-10.

Figure 3(c) also compares the certified accuracy of Adv_2 models with the existing state-of-the-art *SmoothAdv* models (Salman et al., 2019a). SmoothAdv utilizes adversarial training using an adaptive attack with ℓ_2 threat boundary of ϵ and Gaussian noises, $\mathcal{N}(0, \sigma^2 I)$ (See details in Appendix A). We set the noise to $\sigma = 0.5$ and vary ϵ for their training to compare with different SmoothAdv models in Figure 3(c). By adapting Adv_2 models with $\sigma = 0.5$ at test-time using our proposed Algorithm 1, we already achieve similar performance as SmoothAdv. Moreover, unlike existing frameworks,

we also provide test-time flexibility to adapt the same models using different σ, without retraining, to improve their certification, as shown below. [2]



(a) Adv_2 (IMAGENET) (b) Adv_2 (CIFAR-10) (c) SmoothAdv (CIFAR-10)

Figure 4: Certified accuracy at various ℓ_2 radii by varying σ for test-time adaptation of the same models. Choosing large σ degrades certification at lower ℓ_2 radii, while provides better performance at higher ℓ_2 radii.

Flexibility of choosing different noise-level σ for certification at test-time. Figure 4 presents the certification results as we vary $\sigma = \{0.25, 0.5, 0.75\}$ for test-time adaptation of the same models using our Algorithm 1. We note that the choice of large σ degrades certification at lower ℓ_2 radii while providing better performance for higher ℓ_2 radii. For each test example, we adapt the models with appropriate σ that provides the maximum certified radius to obtain the upper envelope of the certification accuracy curves in Figure 4. This leads to the state-of-the-art certification performance for Adv_2 models, outperforming the existing SmoothAdv models for CIFAR-10 (Table 1). Further for randomized smoothing models (Figure 4(c)), we consistently provide certification at larger ℓ_2 radii by adapting using larger σ values, improving their overall ACR scores (see Table 6 and 5 (Appendix)). Additional results using several models with different training setups are provided in Figure 7 and 8 (Appendix).

Over-fitting reduces Certification. Rice et al. (2020) demonstrate that AT models *overfit* as we train without *early stopping* criteria. It degrades their empirical robustness against adversarial attacks. In Figure 5, we compare with the certification accuracy of such *overfitted* AT models, denoted as $\text{Adv}^{overfit}$. We observe that $\text{Adv}^{overfit}$ models also degrade the certified robustness, in particular, at higher ℓ_2 radii, compared to their corresponding AT models with early stopping criteria. These results also indicate that the empirical and certified robustness are closely related: improving empirical robustness also improves the certified robustness.

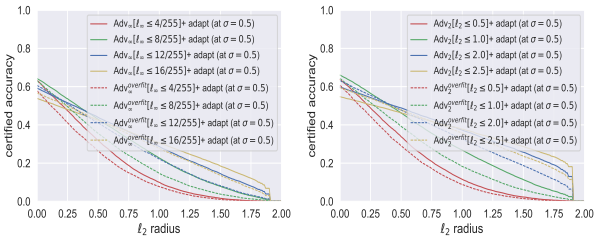


Figure 5: CIFAR-10: Comparing the certified accuracy of Adv_∞ (**Left**) and Adv_2 (**Right**) models with and without applying early-stopping criteria (denoted as $\text{Adv}^{overfit}$).

5 CONCLUSION

We propose a novel *certification through adaptation* algorithm that transforms adversarially trained models into a randomized smoothing classifier using test-time covariate shift adaptation to provide certified robustness for ℓ_2 norm. Unlike existing models using BN adaptation for different applications, our certification framework does not make any assumptions on the test examples. One main advantage of our proposed certification algorithm is to separately choose appropriate noise levels σ during inference for each test example. We achieve the state-of-the-art ℓ_2 certification using Adv_2 models for CIFAR-10. Finally, while we mainly focus on ℓ_2 certification using Gaussian noise, we can also extend this framework for other types of perturbations as long as randomized smoothing works (e.g., uniform noise for ℓ_1 norm (Yang et al., 2020)) for different applications without any additional training.

[2] Since adversarial training for IMAGENET is still at a nascent stage, we did not compare with Salman et al. (2019a) in this paper.

6 CODE OF ETHICS AND REPRODUCIBILITY

Code of Ethics. Existing defense models can only provide either empirical or certified robustness against adversarial attacks for higher dimensional input domains. In this paper, we propose a solution to provide high performance to achieve both empirical or certified robustness. It allows us to improve the reliability and trustworthiness for large AI models for sensitive real-world applications.

Reproducibility. The key results of our paper are presented using adversarially trained models. For CIFAR-10, we train the models using the codes provided in https://github.com/locuslab/robust_overfitting (Rice et al., 2020). For IMAGENET, we obtained the already trained AT models from https://github.com/locuslab/robust_overfitting (Rice et al., 2020). Please refer to Appendix A for more details. We have provided the codes for our certification algorithms in the supplementary materials for reproducing the results of our paper.

REFERENCES

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *ICML*, 2018.

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

Rudy Bunel, Ilker Turkaslan, Philip HS Torr, Pushmeet Kohli, and M Pawan Kumar. A unified view of piecewise linear neural network verification. *NeurIPS*, 2018.

Xiaoyu Cao and Neil Zhenqiang Gong. Mitigating evasion attacks to deep neural networks via region-based classification. In *ACSAC*, 2017.

Fabio Maria Cariucci, Lorenzo Porzi, Barbara Caputo, Elisa Ricci, and Samuel Rota Bulo. Autodial: Automatic domain alignment layers. In *ICCV*, 2017.

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

Jeremy M Cohen, Elan Rosenfeld, and J Zico Kolter. Certified adversarial robustness via randomized smoothing. *ICML*, 2019.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

Krishnamurthy Dvijotham, Robert Stanforth, Sven Gowal, Timothy A Mann, and Pushmeet Kohli. A dual approach to scalable verification of deep networks. In *UAI*, 2018.

Krishnamurthy (Dj) Dvijotham, Jamie Hayes, Borja Balle, Zico Kolter, Chongli Qin, Andras Gyorgy, Kai Xiao, Sven Gowal, and Pushmeet Kohli. A framework for robustness certification of smoothed classifiers using f-divergences. In *ICLR*, 2020.

Logan Engstrom, Andrew Ilyas, Hadi Salman, Shibani Santurkar, and Dimitris Tsipras. Robustness (python library), 2019. URL `https://github.com/MadryLab/robustness`.

Christian Etmann, Sebastian Lunz, Peter Maass, and Carola-Bibiane Schönlieb. On the connection between adversarial robustness and saliency map interpretability. In *ICML*, 2019.

Geoffrey French, Michal Mackiewicz, and Mark Fisher. Self-ensembling for visual domain adaptation. *arXiv*, 2017.

Timon Gehr, Matthew Mirman, Dana Drachsler-Cohen, Petar Tsankov, Swarat Chaudhuri, and Martin Vechev. Ai2: Safety and robustness certification of neural networks with abstract interpretation. In *IEEE SP*, 2018.

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

Sven Gowal, Krishnamurthy Dvijotham, Robert Stanforth, Rudy Bunel, Chongli Qin, Jonathan Uesato, Relja Arandjelovic, Timothy Mann, and Pushmeet Kohli. On the effectiveness of interval bound propagation for training verifiably robust models. *arXiv*, 2018.

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

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

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *ECCV*, 2016b.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *ICLR*, 2019.

Lei Huang, Dawei Yang, Bo Lang, and Jia Deng. Decorrelated batch normalization. In *CVPR*, 2018.

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

Ajil Jalal, Andrew Ilyas, Constantinos Daskalakis, and Alexandros G Dimakis. The robust manifold defense: Adversarial training using generative models. *arXiv*, 2019.

Jongheon Jeong and Jinwoo Shin. Consistency regularization for certified robustness of smoothed classifiers. *NeurIPS*, 2020.

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

Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236*, 2016.

Mathias Lecuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana. Certified robustness to adversarial examples with differential privacy. In *IEEE SP*, 2019.

Guang-He Lee, Yang Yuan, Shiyu Chang, and Tommi S Jaakkola. Tight certificates of adversarial robustness for randomly smoothed classifiers. *arXiv preprint arXiv:1906.04948*, 2019.

Bai Li, Changyou Chen, Wenlin Wang, and Lawrence Carin. Certified adversarial robustness with additive noise. *NeurIPS*, 2019.

Yanghao Li, Naiyan Wang, Jianping Shi, Jiaying Liu, and Xiaodi Hou. Revisiting batch normalization for practical domain adaptation. *arXiv*, 2016.

Xuanqing Liu, Minhao Cheng, Huan Zhang, and Cho-Jui Hsieh. Towards robust neural networks via random self-ensemble. In *ECCV*, 2018.

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

Chengzhi Mao, Mia Chiquier, Hao Wang, Junfeng Yang, and Carl Vondrick. Adversarial attacks are reversible with natural supervision. *arXiv preprint arXiv:2103.14222*, 2021.

Matthew Mirman, Timon Gehr, and Martin Vechev. Differentiable abstract interpretation for provably robust neural networks. In *ICML*, 2018.

Seyed Mohsen Moosavi Dezfooli, Alhussein Fawzi, Jonathan Uesato, and Pascal Frossard. Robustness via curvature regularization, and vice versa. In *CVPR*, 2019.

Mark Niklas Mueller, Mislav Balunovic, and Martin Vechev. Certify or predict: Boosting certified robustness with compositional architectures. In *ICLR*, 2021.

Zachary Nado, Shreyas Padhy, D Sculley, Alexander D'Amour, Balaji Lakshminarayanan, and Jasper Snoek. Evaluating prediction-time batch normalization for robustness under covariate shift. *arXiv*, 2020.

Jay Nandy, Wynne Hsu, and Mong-Li Lee. Approximate manifold defense against multiple adversarial perturbations. In *IJCNN*, 2020.

Tianyu Pang, Xiao Yang, Yinpeng Dong, Hang Su, and Jun Zhu. Bag of tricks for adversarial training. In *ICLR*, 2021. URL https://openreview.net/forum?id=Xb8xvrtB8Ce.

Chongli Qin et al. Adversarial robustness through local linearization. In *NeurIPS*, 2019.

Aditi Raghunathan, Jacob Steinhardt, and Percy Liang. Certified defenses against adversarial examples. *ICLR*, 2018.

Leslie Rice, Eric Wong, and Zico Kolter. Overfitting in adversarially robust deep learning. In *ICML*, 2020.

Subhankar Roy, Aliaksandr Siarohin, Enver Sangineto, Samuel Rota Bulo, Nicu Sebe, and Elisa Ricci. Unsupervised domain adaptation using feature-whitening and consensus loss. In *CVPR*, 2019.

Hadi Salman, Jerry Li, Ilya Razenshteyn, Pengchuan Zhang, Huan Zhang, Sebastien Bubeck, and Greg Yang. Provably robust deep learning via adversarially trained smoothed classifiers. In *NeurIPS*, 2019a.

Hadi Salman, Greg Yang, Huan Zhang, Cho-Jui Hsieh, and Pengchuan Zhang. A convex relaxation barrier to tight robustness verification of neural networks. In *NeurIPS*, 2019b.

Hadi Salman, Mingjie Sun, Greg Yang, Ashish Kapoor, and J Zico Kolter. Denoised smoothing: A provable defense for pretrained classifiers. *NeurIPS*, 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.

Bernhard Schölkopf, Dominik Janzing, Jonas Peters, Eleni Sgouritsa, Kun Zhang, and Joris Mooij. On causal and anticausal learning. *arXiv preprint arXiv:1206.6471*, 2012.

Lukas Schott, Jonas Rauber, Matthias Bethge, and Wieland Brendel. Towards the first adversarially robust neural network model on MNIST. In *ICLR*, 2019.

Fatemeh Sheikholeslami, Ali Lotfi, and J Zico Kolter. Provably robust classification of adversarial examples with detection. In *ICLR*, 2021.

Masashi Sugiyama and Motoaki Kawanabe. *Machine learning in non-stationary environments: Introduction to covariate shift adaptation*. MIT press, 2012.

Baochen Sun, Jiashi Feng, and Kate Saenko. Correlation alignment for unsupervised domain adaptation. In *Domain Adaptation in Computer Vision Applications*, pp. 153–171. Springer, 2017.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *ICML*, 2020.

Christian Szegedy et al. Intriguing properties of neural networks. In *ICLR*, 2014.

Vincent Tjeng, Kai Xiao, and Russ Tedrake. Evaluating robustness of neural networks with mixed integer programming. *ICLR*, 2017.

Florian Tramèr and Dan Boneh. Adversarial training and robustness for multiple perturbations. In *NeurIPS*, 2019.

Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. In *ICLR*, 2019.

Jonathan Uesato, Brendan O'Donoghue, Aaron van den Oord, and Pushmeet Kohli. Adversarial risk and the dangers of evaluating against weak attacks. *ICML*, 2018.

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

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. *arXiv*, 2020.

Shiqi Wang, Kexin Pei, Justin Whitehouse, Junfeng Yang, and Suman Jana. Efficient formal safety analysis of neural networks. In *NeurIPS*, 2018.

Lily Weng, Huan Zhang, Hongge Chen, Zhao Song, Cho-Jui Hsieh, Luca Daniel, Duane Boning, and Inderjit Dhillon. Towards fast computation of certified robustness for relu networks. In *ICML*, 2018.

Eric Wong and Zico Kolter. Provable defenses against adversarial examples via the convex outer adversarial polytope. In *ICML*, 2018.

Eric Wong, Frank R Schmidt, Jan Hendrik Metzen, and J Zico Kolter. Scaling provable adversarial defenses. *NeurIPS*, 2018.

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

Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *CVPR*, 2020.

Greg Yang, Tony Duan, Edward Hu, Hadi Salman, Ilya Razenshteyn, and Jerry Li. Randomized smoothing of all shapes and sizes. *arXiv*, 2020.

Runtian Zhai, Chen Dan, Di He, Huan Zhang, Boqing Gong, Pradeep Ravikumar, Cho-Jui Hsieh, and Liwei Wang. Macer: Attack-free and scalable robust training via maximizing certified radius. In *ICLR*, 2020.

Hongyang Zhang et al. Theoretically principled trade-off between robustness and accuracy. In *ICML*, 2019.

Jingfeng Zhang, Xilie Xu, Bo Han, Gang Niu, Lizhen Cui, Masashi Sugiyama, and Mohan Kankanhalli. Attacks which do not kill training make adversarial learning stronger. In *ICML*. PMLR, 2020.

APPENDIX ORGANIZATION

- **Section A: Experimental setup.**
- **Section B: Additional Results on Certification.**
- **Section C:Performance against different corruptions**

A EXPERIMENTAL SETUP

A.1 IMPLEMENTATION DETAILS

We present our experimental results on CIFAR-10 (Krizhevsky et al., 2009) and IMAGENET (Deng et al., 2009) datasets. The descriptions of different models and training hyper-parameters are provided in the following:

A.1.1 CIFAR-10.

We use pre-activation ResNet18 architecture (He et al., 2016b) for our experiments on CIFAR-10. We apply the SGD optimizer with a batch size of 128. We execute a total of 200 training epochs and apply a step-wise learning rate decay set initially at 0.1 and divided by 10 at 100 and 150 epochs, and weight decay 5×10^{-4}.

AT models (Madry et al., 2018; Rice et al., 2020): Unless and otherwise specified, our AT models are learned using early stopping criteria as described in (Rice et al., 2020). We learn several AT models with different threat boundaries for our experiments. We denote them by specifying their corresponding threat model and threat boundaries. For example, $\text{Adv}_2[\ell_2 \leq 1.5]$ denotes an AT model that is learned using PGD adversary with ℓ_2 threat model and a threat boundary of $\epsilon = 1.5$, along with *early-stopping criteria* (Rice et al., 2020). We also learn AT models *without* using early-stopping criteria, as in (Madry et al., 2018) for our comparison in Figure 5. These models are denoted as $\text{Adv}^{overfit}$.

We use *projected gradient descent (PGD)* adversarial attack (Madry et al., 2018) to train these AT models as follows: For Adv_∞, we use 10 iterations and an ℓ_∞ step size of $\epsilon/4$. For Adv_2, we use 10 iterations and an ℓ_2 step size of $\epsilon/8.5$. This is the same experimental setup as in (Rice et al., 2020)). We choose a small set of $1,000$ images from the CIFAR-10 test set for our validation. We apply the PGD attack with the same hyper-parameters for our validation during training. We save the best model using the *early-stopping* criteria (Rice et al., 2020).

Randomized smoothing model by Cohen et al. (2019): We also train $\text{Rand}_{\sigma=0.5}$ by training with augmented random noise, sampled from an isotropic Gaussian distribution $\mathcal{N}(0, \sigma^2 I)$ with $\sigma = 0.5$. Here, we keep the same model architecture, learning rates, batch sizes, and other hyper-parameters as used to learn the AT models.

Randomized smoothing model by Salman et al. (2019a): We also compare with the state-of-the-art certification models, called 'SmoothAdv', by Salman et al. (2019a) for our experiments on ℓ_2 certification We train the SmoothAdv models by choosing random noise vectors followed by an adaptive adversarial attack with specified ℓ_2 threat boundary of ϵ at each iteration. The noise vectors are sampled from an isotropic Gaussian distribution $\mathcal{N}(0, \sigma^2 I)$.

We note that the training hyper-parameter ϵ has the most significant impact on the certification curve for a SmoothAdv model (please refer to Table 7-15 of (Salman et al., 2019a) for more details). For our experiments, we train 4 different SmoothAdv models with $\epsilon = \{0.25, 0.5, 1, 2\}$ and $\sigma = 0.5$ using adaptive PGD attack with 10 steps. We denote them as $\text{SmoothAdv}_{\sigma=0.5,\epsilon=0.25}$, $\text{SmoothAdv}_{\sigma=0.5,\epsilon=0.5}$, $\text{SmoothAdv}_{\sigma=0.5,\epsilon=1}$ and $\text{SmoothAdv}_{\sigma=0.5,\epsilon=2}$ respectively. We use the same training set-up and other hyper-parameters as specified in their Github: https://github.com/Hadisalman/smoothing-adversarial.

A.1.2 IMAGENET.

We use ResNet50 architecture (He et al., 2016a) for IMAGENET. We obtain the Baseline and Rand$_{\sigma=0.5}$ models from (Cohen et al., 2019)[3]. These models are trained using Gaussian augmented noises, sampled from isotropic Gaussian distribution $\mathcal{N}(0, \sigma^2 I)$ with $\sigma = 0.0$ (i.e., no noise) and $\sigma = 0.5$ respectively.

The AT models i.e., Adv$_\infty[\ell_\infty \leq 4/255]$ and Adv$_2[\ell_2 \leq 3]$ are learned for ℓ_∞ and ℓ_2 threat models with threat boundary of $4/255$ and 3, respectively. We use the publicly available models provided by Rice et al. (2020) [4]. These models are fine-tuned using PGD-based adversarial training with early stopping criteria, originally provided by Engstrom et al. (2019) [5].

We resize the input images to 256×265 pixels and crop 224×224 pixels from the center. For our experiments on certification, we use a set of 500 test images by choosing at most 1 sample for each class.

A.2 CHOICE OF TEST-TIME ADAPTIVE BN HYPER-PARAMETERS

BN adaptation technique is controlled by two hyper-parameters, i.e., the *test batch-size* and *momentum* (ρ) (see Equation 5) to update the statistics of the batch-normalization layers. Assuming that the test images are obtained independently from the same test distribution, we can efficiently compute the BN statistics from these images. The hyper-parameter $\rho \in [0, 1]$ controls the tread-off between pre-computed training statistics and test statistics. We can obtain a better estimation of the test distribution from a large test batch. Hence, we can choose a higher value of ρ.

Here, we compare the top-1 test accuracy of AT models under Gaussian augmented noise with $\sigma = 0.5$ for different choices of ρ and the batch size. We skip the standard baseline models from our analysis and refer to the previous works (Schneider et al., 2020; Nado et al., 2020) that analyzed the effects of these hyper-parameters for the standard baseline DNN classifiers.

(a) IMAGENET

ρ	Adv$_\infty$	Adv$_2$
0.0 (No adaptation)	0.4$_{\pm 0.01}$	0.9$_{\pm 0.01}$
0.1	2.1$_{\pm 0.04}$	7.7$_{\pm 0.09}$
0.3	20.6$_{\pm 0.16}$	36.6$_{\pm 0.09}$
0.5	41.1$_{\pm 0.09}$	45.5$_{\pm 0.13}$
0.7	43.5$_{\pm 0.14}$	46.7$_{\pm 0.13}$
0.9	44.2$_{\pm 0.12}$	46.8$_{\pm 0.13}$
1.0 (Full adaptation)	44.8$_{\pm 0.13}$	47.2$_{\pm 0.14}$

(b) CIFAR-10

ρ	Adv$_\infty$	Adv$_2$
0.0 (No adaptation)	16.1$_{\pm 7.85}$	21.5$_{\pm 7.79}$
0.1	45.1$_{\pm 0.49}$	46.9$_{\pm 0.48}$
0.3	59.2$_{\pm 0.42}$	60.8$_{\pm 0.33}$
0.5	62.4$_{\pm 0.27}$	64.4$_{\pm 0.6}$
0.7	62.8$_{\pm 0.52}$	64.9$_{\pm 0.31}$
0.9	62.8$_{\pm 0.71}$	64.9$_{\pm 0.31}$
1.0 (Full adaptation)	62.4$_{\pm 0.64}$	64.9$_{\pm 0.73}$

Table 3: Top-1 accuracy using fixed test batch-size $= 512$ for AT models under Gaussian augmented noise with $\sigma = 0.5$ for different choices of momentum, ρ during inference. We randomly shuffle the test images to report ($mean + 2 \times sd$) of 5 different runs.

Momentum (ρ). We first investigate the effect of momentum (ρ) as we choose a large batch size of 512. In Table 3, we present the performance of AT models for different values of ρ. Recall that, $\rho = 1$ denotes *full adaptation* (Equation 5). Here, we completely ignore the training statistics and recompute the BN statistics using the test batches. In contrast, $\rho = 0$ represents *no adaptation*, i.e., the same as the standard 'deterministic' inference setup. In this case, we use the previously computed BN statistics obtained during training.

We observe that for IMAGENET (Table 3 [Left]) the performance started converging at $\rho = 0.7$. For CIFAR-10 (Table 3 [Right]), the convergence started even earlier at $\rho = 0.5$.

Batch Size. Next, we investigate the minimum size of the test batches to choose $\rho = 1$ (i.e., full-adaptation). In Table 4, we fix $\rho = 1$ and vary the test batch sizes as we evaluate these AT models. We observe that the performance of these models started improving even when we are using the

[3] https://github.com/locuslab/smoothing

[4] https://github.com/locuslab/robust_overfitting

[5] https://github.com/MadryLab/robustness

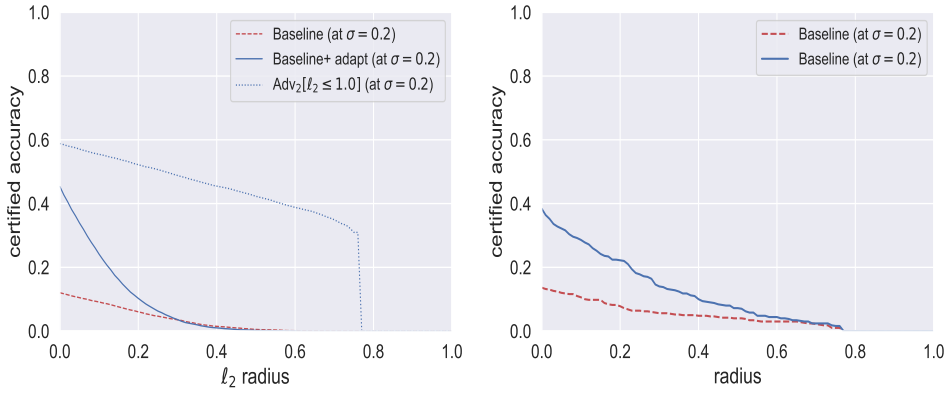
(a) IMAGENET		
Batch Size	Adv_∞	Adv_2
w/o BN adapt	$0.4_{\pm 0.01}$	$0.9_{\pm 0.01}$
8	$11.5_{\pm 0.22}$	$9.1_{\pm 0.15}$
16	$28.1_{\pm 0.22}$	$26.7_{\pm 0.14}$
32	$37.1_{\pm 0.24}$	$37.6_{\pm 0.2}$
64	$41.4_{\pm 0.26}$	$42.9_{\pm 0.12}$
128	$43.3_{\pm 0.15}$	$45.4_{\pm 0.13}$
256	$44.4_{\pm 0.21}$	$46.7_{\pm 0.07}$
512	$44.8_{\pm 0.13}$	$47.2_{\pm 0.14}$

(b) CIFAR-10		
Batch Size	Adv_∞	Adv_2
w/o BN adapt	$16.1_{\pm 7.85}$	$21.5_{\pm 7.79}$
8	$57.2_{\pm 1.23}$	$59.5_{\pm 0.38}$
16	$60.2_{\pm 0.79}$	$62.3_{\pm 0.87}$
32	$61.5_{\pm 0.46}$	$63.6_{\pm 0.55}$
64	$62.3_{\pm 0.5}$	$64.0_{\pm 0.38}$
128	$62.7_{\pm 0.68}$	$64.4_{\pm 0.53}$
256	$62.7_{\pm 0.68}$	$64.9_{\pm 0.48}$
512	$62.4_{\pm 0.64}$	$64.9_{\pm 0.73}$

Table 4: Top-1 accuracy using fixed $\rho = 1$ for AT models under Gaussian augmented noise with $\sigma = 0.5$ for different size of test batches during inference. We randomly shuffle the test images to report $(mean + 2 \times s.d.)$ of 5 different runs.

test batches of size 8. The performance further improves as we choose larger sizes of test batches. We can see that their performance started converging as we choose the test batches of size 64 for IMAGENET. On the other hand, the convergence started much earlier for CIFAR-10.

B ADDITIONAL RESULTS ON CERTIFICATION

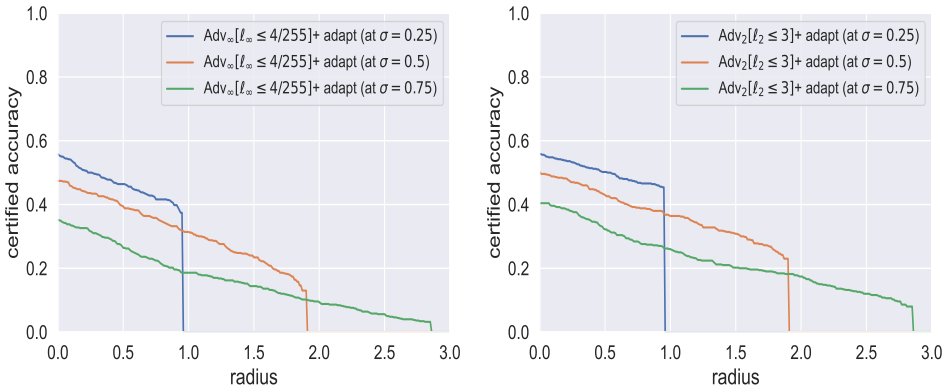


(a) CIFAR-10 (b) IMAGENET

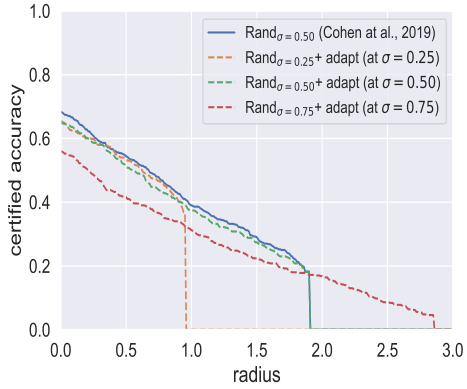
Figure 6: ℓ_2 Certification for standard non-robust classifiers. For CIFAR-10, we observe that, even after adaptation, the baseline produces lower certification compared to $\text{Adv}_2[\ell_2 \leq 1]$ model without any adaptation.

B.1 ℓ_2 CERTIFICATION FOR STANDARD NON-ROBUST CLASSIFIERS

In Figure 6, we present the ℓ_2 certification results for standard non-robust classification models using our proposed Algorithm 1. In Table 2, we note that the adaptive BN technique can also significantly improve the performance of a non-robust model at lower noise levels, σ. In particular, for CIFAR-10 dataset, Baseline models using adaptation achieve similar performance as $\text{Adv}_2[\ell_2 \leq 1]$ without BN adaptation, while produces significantly lower ℓ_2 certification robustness. This is because, Baseline models, even after adaptation cannot consistently predict the same class to provide higher certified robustness at larger ℓ_2 radii. As a result, we can only improve the certified robustness at smaller ℓ_2 radii.

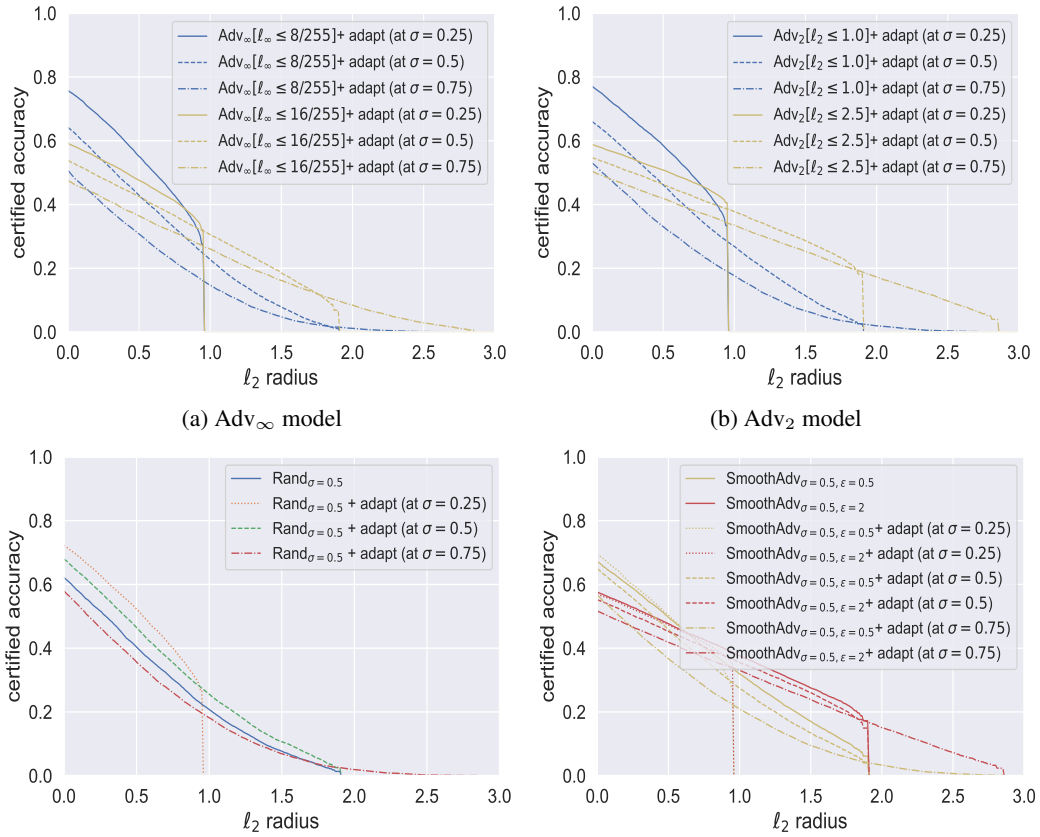


(a) Adv_∞ model (b) Adv_2 model



(c) $\text{Rand}_{\sigma=0.5}$ (Cohen et al., 2019)

Figure 7: IMAGENET: Certified top-1 accuracy at various ℓ_2 radii as we vary the noise-level, σ at test-time using proposed Algorithm 1. Adv_∞ and Adv_2 models are as defined in experimental set-up (section 4). Refer to Table 5 for complete results of all models and different settings.



(a) Adv$_\infty$ model (b) Adv$_2$ model

(c) Rand$_{\sigma=0.5}$ (Cohen et al., 2019) (d) SmoothAdv (Salman et al., 2019a)

Figure 8: CIFAR-10: Certified top-1 accuracy at various ℓ_2 radii as we vary the noise-level, σ at test-time using proposed Algorithm 1. Refer to Table 6 for complete results of all models and different settings.

IMAGENET													
Model	BN adaption	Certification	ℓ_2 Radius											
			0.25	0.5	0.75	1.0	1.25	1.5	1.75	2.0	2.25	2.5	2.75	ACR
Baseline	-	at $\sigma = 0.25$	7.8	4.8	3.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.054
Adv$_\infty[\ell_\infty \le 4/255]$	at $\sigma = 0.25$	at $\sigma = 0.25$	50.0	46.4	41.6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.445
	at $\sigma = 0.50$	at $\sigma = 0.50$	43.6	39.4	35.8	31.4	27.6	23.4	18.2	0.0	0.0	0.0	0.0	0.607
	at $\sigma = 0.75$	at $\sigma = 0.75$	31.6	26.4	22.4	18.6	16.8	14.4	11.8	9.4	7.6	5.6	3.6	0.443
Adv$_\infty[\ell_\infty \le 4/255]$ + adapt [Best Radii] (Ours)			50.0	46.4	41.6	31.4	27.6	23.4	18.2	9.4	7.6	5.6	3.6	0.759
Adv$_2[\ell_2 \le 3.00]$	at $\sigma = 0.25$	at $\sigma = 0.25$	53.2	50.2	46.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.480
	at $\sigma = 0.50$	at $\sigma = 0.50$	47.0	43.0	39.0	36.4	32.8	30.8	27.0	0.0	0.0	0.0	0.0	0.711
	at $\sigma = 0.75$	at $\sigma = 0.75$	37.8	32.2	28.4	26.0	22.4	20.2	19.0	17.4	14.2	12.0	9.6	0.639
Adv$_2[\ell_2 \le 3.00]$ + adapt [Best Radii] (Ours)			53.2	50.2	46.8	36.4	32.8	30.8	27.0	17.4	14.2	12.0	9.6	0.930
Rand$_{\sigma=0.5}$ Cohen et al. (2019)	-	at $\sigma = 0.50$	60.8	54.4	47.8	39.0	34.2	29.0	23.8	0.0	0.0	0.0	0.0	0.809
+ adaptation	at $\sigma = 0.25$	at $\sigma = 0.25$	59.8	53.6	46.6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.507
	at $\sigma = 0.50$	at $\sigma = 0.50$	58.6	51.0	43.8	37.4	32.2	27.4	22.4	0.0	0.0	0.0	0.0	0.768
	at $\sigma = 0.75$	at $\sigma = 0.75$	48.6	41.6	36.6	31.2	26.2	22.4	18.6	16.8	12.8	8.6	5.4	0.720
Rand$_{\sigma=0.5}$ + adapt [Best Radii] (Ours)			59.8	53.6	46.6	37.4	32.2	27.4	22.4	16.8	12.8	8.6	5.4	0.973

Table 5: IMAGENET: Certified top-1 accuracy at various ℓ_2 radii as we vary σ for BN adaptation and certification along with average certified radii (ACR). We use ResNet50 for IMAGENET. Each gray block is corresponding to one classification model while the rows are corresponding to its certification performances as we choose different noise levels for adaptations and certifications. The *Best Radii* are obtained by selecting the highest radius for each test example as we adapt the models with different noise levels, σ.

CIFAR-10											
Model	BN adaption	Certification	ℓ_2 Radius								
			0.25	0.5	0.75	1.0	1.25	1.5	1.75	2.0	ACR
Baseline	-	at $\sigma = 0.25$	6.96	2.04	0.09	0.0	0.0	0.0	0.0	0.0	0.026
$\text{Adv}_\infty[\ell_\infty \leq 4/255]$	at $\sigma = 0.25$	at $\sigma = 0.25$	67.96	50.46	31.96	0.0	0.0	0.0	0.0	0.0	0.485
	at $\sigma = 0.50$	at $\sigma = 0.50$	47.34	31.83	18.78	9.98	4.44	1.62	0.28	0.0	0.350
	at $\sigma = 0.75$	at $\sigma = 0.75$	26.89	15.92	8.44	4.31	2.03	0.79	0.23	0.08	0.146
$\text{Adv}_\infty[\ell_\infty \leq 8/255]$	at $\sigma = 0.25$	at $\sigma = 0.25$	66.43	55.06	42.86	0.0	0.0	0.0	0.0	0.0	0.527
	at $\sigma = 0.50$	at $\sigma = 0.50$	53.65	42.91	32.58	22.68	14.24	7.88	2.94	0.0	0.515
	at $\sigma = 0.75$	at $\sigma = 0.75$	39.96	30.76	22.01	14.64	8.84	4.81	2.29	1.15	0.352
$\text{Adv}_\infty[\ell_\infty \leq 12/255]$	at $\sigma = 0.25$	at $\sigma = 0.25$	60.52	52.42	43.27	0.0	0.0	0.0	0.0	0.0	0.499
	at $\sigma = 0.50$	at $\sigma = 0.50$	51.53	43.94	36.41	28.69	21.25	14.53	8.03	0.0	0.581
	at $\sigma = 0.75$	at $\sigma = 0.75$	42.61	35.56	28.47	22.39	16.69	11.58	7.42	4.43	0.482
$\text{Adv}_\infty[\ell_\infty \leq 16/255]$	at $\sigma = 0.25$	at $\sigma = 0.25$	53.75	47.57	41.18	0.0	0.0	0.0	0.0	0.0	0.454
	at $\sigma = 0.50$	at $\sigma = 0.50$	48.07	42.51	36.54	30.55	24.68	18.49	12.11	0.0	0.598
	at $\sigma = 0.75$	at $\sigma = 0.75$	42.05	36.42	31.24	26.05	20.74	16.15	12.01	8.45	0.557
Adv_∞ + adapt [Best Radii] (Ours)			67.96	55.06	43.27	30.55	24.68	18.49	12.11	8.45	0.903
$\text{Adv}_2[\ell_2 \leq 0.50]$	at $\sigma = 0.25$	at $\sigma = 0.25$	68.84	54.04	37.13	0.0	0.0	0.0	0.0	0.0	0.518
	at $\sigma = 0.50$	at $\sigma = 0.50$	48.81	33.82	20.95	11.5	5.64	2.29	0.62	0.0	0.382
	at $\sigma = 0.75$	at $\sigma = 0.75$	27.38	16.15	9.23	4.56	2.06	0.91	0.33	0.08	0.153
$\text{Adv}_2[\ell_2 \leq 1.00]$	at $\sigma = 0.25$	at $\sigma = 0.25$	68.02	58.54	46.98	0.0	0.0	0.0	0.0	0.0	0.551
	at $\sigma = 0.50$	at $\sigma = 0.50$	56.45	46.24	35.6	26.89	18.73	11.37	5.41	0.0	0.580
	at $\sigma = 0.75$	at $\sigma = 0.75$	43.04	33.08	24.81	17.68	11.39	6.6	3.57	1.95	0.405
$\text{Adv}_2[\ell_2 \leq 1.25]$	at $\sigma = 0.25$	at $\sigma = 0.25$	67.13	58.77	49.43	0.0	0.0	0.0	0.0	0.0	0.557
	at $\sigma = 0.50$	at $\sigma = 0.50$	57.73	48.8	39.64	31.07	22.61	15.82	8.96	0.0	0.647
	at $\sigma = 0.75$	at $\sigma = 0.75$	46.54	37.53	29.35	22.0	15.62	10.51	6.55	3.68	0.496
$\text{Adv}_2[\ell_2 \leq 1.50]$	at $\sigma = 0.25$	at $\sigma = 0.25$	64.21	57.13	49.71	0.0	0.0	0.0	0.0	0.0	0.543
	at $\sigma = 0.50$	at $\sigma = 0.50$	56.55	49.19	41.72	34.47	27.36	20.23	12.98	0.0	0.689
	at $\sigma = 0.75$	at $\sigma = 0.75$	47.73	40.89	33.78	27.22	20.78	15.19	10.51	6.77	0.585
$\text{Adv}_2[\ell_2 \leq 2.00]$	at $\sigma = 0.25$	at $\sigma = 0.25$	60.4	54.71	48.35	0.0	0.0	0.0	0.0	0.0	0.523
	at $\sigma = 0.50$	at $\sigma = 0.50$	54.27	48.89	43.1	37.34	31.52	25.74	19.14	0.0	0.731
	at $\sigma = 0.75$	at $\sigma = 0.75$	47.96	42.54	37.04	31.65	26.17	21.12	16.59	12.44	0.698
$\text{Adv}_2[\ell_2 \leq 2.25]$	at $\sigma = 0.25$	at $\sigma = 0.25$	57.08	52.5	47.11	0.0	0.0	0.0	0.0	0.0	0.504
	at $\sigma = 0.50$	at $\sigma = 0.50$	52.1	46.99	42.26	36.9	31.58	26.08	20.03	0.0	0.724
	at $\sigma = 0.75$	at $\sigma = 0.75$	46.45	41.71	36.75	31.88	26.95	22.33	17.82	13.55	0.713
$\text{Adv}_2[\ell_2 \leq 2.50]$	at $\sigma = 0.25$	at $\sigma = 0.25$	54.88	50.79	46.29	0.0	0.0	0.0	0.0	0.0	0.487
	at $\sigma = 0.50$	at $\sigma = 0.50$	50.53	46.26	41.84	37.74	33.2	28.69	23.34	0.0	0.734
	at $\sigma = 0.75$	at $\sigma = 0.75$	45.95	41.89	37.53	33.55	29.31	25.27	20.98	17.28	0.765
$\text{Adv}_2[\ell_2 \leq 3.00]$	at $\sigma = 0.25$	at $\sigma = 0.25$	53.82	49.69	45.04	0.0	0.0	0.0	0.0	0.0	0.475
	at $\sigma = 0.50$	at $\sigma = 0.50$	49.41	45.57	41.52	37.43	33.37	28.82	23.65	0.0	0.720
	at $\sigma = 0.75$	at $\sigma = 0.75$	45.37	41.54	37.75	33.49	29.35	25.62	21.83	18.23	0.771
Adv_2 + adapt [Best Radii] (Ours)			**68.84**	**58.77**	**49.71**	**37.74**	**33.37**	**28.82**	**23.65**	**18.23**	**1.198**
$\text{Rand}_{\sigma=0.5}$	-	at $\sigma = 0.50$	51.68	40.38	30.25	20.81	13.36	7.71	3.38	0.0	0.488
+ adaptation	at $\sigma = 0.25$	at $\sigma = 0.25$	62.91	52.25	40.06	0.0	0.0	0.0	0.0	0.0	0.497
	at $\sigma = 0.50$	at $\sigma = 0.50$	57.58	46.46	35.5	25.57	17.43	10.67	5.46	0.0	0.575
	at $\sigma = 0.75$	at $\sigma = 0.75$	46.4	35.63	26.06	18.17	11.61	6.86	3.64	1.92	0.427
$\text{Rand}_{\sigma=0.5}$ + adapt [Best Radii] (Ours)			62.91	52.25	40.06	25.57	17.43	10.67	5.46	1.92	0.657
$\text{SmoothAdv}_{\sigma=0.5,\epsilon=0.25}$	-	at $\sigma = 0.50$	57.8	47.63	37.41	27.88	20.33	13.53	8.03	0.0	0.609
	at $\sigma = 0.25$	at $\sigma = 0.25$	58.74	48.29	36.7	0.0	0.0	0.0	0.0	0.0	0.464
+ adaptation	at $\sigma = 0.50$	at $\sigma = 0.50$	54.0	42.91	32.63	23.6	16.08	9.93	5.5	0.0	0.535
	at $\sigma = 0.75$	at $\sigma = 0.75$	43.15	32.29	23.61	16.41	10.94	6.65	3.88	2.16	0.390
$\text{SmoothAdv}_{\sigma=0.5,\epsilon=0.50}$	-	at $\sigma = 0.50$	58.82	49.68	40.35	31.93	24.18	17.05	10.57	0.0	0.661
	at $\sigma = 0.25$	at $\sigma = 0.25$	59.89	50.4	39.99	0.0	0.0	0.0	0.0	0.0	0.483
+ adaptation	at $\sigma = 0.50$	at $\sigma = 0.50$	55.73	45.79	36.6	27.4	20.03	13.48	7.85	0.0	0.592
	at $\sigma = 0.75$	at $\sigma = 0.75$	46.25	36.72	28.2	20.9	14.73	9.46	5.78	3.32	0.470
$\text{SmoothAdv}_{\sigma=0.5,\epsilon=1.0}$	-	at $\sigma = 0.50$	56.53	49.53	41.38	34.63	27.81	21.22	14.41	0.0	0.691
	at $\sigma = 0.25$	at $\sigma = 0.25$	57.13	48.38	39.7	0.0	0.0	0.0	0.0	0.0	0.467
+ adaptation	at $\sigma = 0.50$	at $\sigma = 0.50$	53.56	45.79	37.64	30.06	23.12	17.27	11.14	0.0	0.620
	at $\sigma = 0.75$	at $\sigma = 0.75$	47.17	39.4	31.8	24.74	18.93	13.7	9.26	5.8	0.545
$\text{SmoothAdv}_{\sigma=0.5,\epsilon=2.0}$	-	at $\sigma = 0.50$	52.82	47.67	42.68	37.55	32.64	27.52	22.42	0.0	0.732
	at $\sigma = 0.25$	at $\sigma = 0.25$	52.23	47.24	41.76	0.0	0.0	0.0	0.0	0.0	0.451
+ adaptation	at $\sigma = 0.50$	at $\sigma = 0.50$	50.28	45.24	40.39	35.5	30.92	26.1	20.25	0.0	0.692
	at $\sigma = 0.75$	at $\sigma = 0.75$	46.7	41.79	37.14	33.05	28.28	23.88	19.34	15.05	0.727
$\text{SmoothAdv}_{\sigma=0.5}$ [Best Radii]			58.82	49.68	42.68	37.55	32.64	27.52	22.42	0.0	0.918
$\text{SmoothAdv}_{\sigma=0.5}$ + adapt [Best Radii] (Ours)			59.89	50.4	41.76	35.5	30.92	26.1	20.25	15.05	1.008
$\text{MARCER}_{\sigma=0.5}$ (Zhai et al., 2020)			60.0	53.0	46.0	38.0	29.0	19.0	12.0	0.0	0.726
$\text{Consistancy}_{\sigma=0.5}$ (Jeong & Shin, 2020)			48.9	45.1	41.3	37.8	33.9	29.9	25.2	0.0	0.726

Table 6: CIFAR-10: Certified top-1 accuracy at various ℓ_2 radii as we vary σ for test-time BN adaptation along with average certified radii (ACR) for individual settings. Each gray block is corresponding to one classification model while the rows are corresponding to its certification performances as we choose different noise levels for adaptations and certifications. The *Best Radii* are obtained by training different models with varying hyper-parameters and adapting them with different noise levels during inference. We also present the best reported results for MARCER (Zhai et al., 2020) and Consistancy Jeong & Shin (2020) at $\sigma = 0.5$, obtained from the respective papers.

C PERFORMANCE AGAINST DIFFERENT CORRUPTIONS

We mainly focus on ℓ_2 certification using Gaussian noise in this paper. However, we note that randomized smoothing techniques have been also applied to provide certifications for other perturbation

types as well (e.g., random uniform noise for ℓ_1 norm (Yang et al., 2020)). Consequently, we can apply our proposed Algorithm 1 to adapt an AT model for any given perturbation types without any additional training for different applications.

Further, Hendrycks & Dietterich (2019) recently introduced ImageNet-C and CIFAR10-C datasets by *algorithmically generated random corruptions* from *noise*, *blur*, *weather*, and *digital* categories with 5 different severity levels for each corruption. Several recent works demonstrated that adaptive BN techniques can significantly improve the performance of any classifier (including AT models) against different random corruptions. Further, – also demonstrated the effectiveness of AT models even without applying any adaptation. Hence, our proposed certification framework for AT models is a step forward towards further improving the reliability of sensitive real-world applications.