Attacking Adversarial Defences by Smoothing the Loss Landscape

Panagiotis Eustratiadis [1] **Henry Gouk** [1] **Da Li** [1 2] **Timothy Hospedales** [1 2]

Abstract

This paper investigates a family of methods for defending against adversarial attacks that owe part of their success to creating a rugged loss landscape that adversaries find difficult to navigate. A common, but not universal, way to achieve this effect is via the use of stochastic neural networks. We show that this is a form of gradient obfuscation, and propose a general extension to gradient-based adversaries based on the Weierstrass transform, which smooths the surface of the loss function and provides more reliable gradient estimates. We further show that the same principle can strengthen gradient-free adversaries. We demonstrate the efficacy of our loss-smoothing method against both stochastic and non-stochastic adversarial defences that exhibit robustness due to this type of obfuscation. Furthermore, we provide analysis of how it interacts with Expectation over Transformation; a popular gradient-sampling method currently used to attack stochastic defences.

1. Introduction

The discovery of adversarial examples in deep learning (Szegedy et al., 2014), together with its growing commercial and societal importance, has led to adversarial defence emerging as an important field of machine learning research, with the purpose of creating models that are robust against adversarial perturbations. There is an interplay between adversarial attack and defence research, where stronger defences are developed, and often subsequently broken by more innovative attacks (Kurakin et al., 2018). An example of this dynamic is the discovery that many defences against gradient-based adversaries relied on masking the gradient signal from the attacker (Athalye et al., 2018a). However, as shown by (Athalye et al., 2018a), such obfuscation gives a false sense of security and is easy to circumvent. They successfully attack stochastic defences by repeatedly sampling the gradient of the loss function w.r.t. the input and averaging the samples to obtain more reliable gradient estimates. They name this technique Expectation over Transformation (EoT) (Athalye et al., 2018b). It has since been standardised that new stochastic defences (Eustratiadis et al., 2021; He et al., 2019; Jeddi et al., 2020; Yu et al., 2021) apply EoT during evaluation, to ensure that their apparent robustness does not rely on stochastic gradients.

In this paper, we reveal a form of gradient obfuscation that, to the best of our knowledge, is not yet known. So far, it is understood that stochastic neural networks (SNNs) defend effectively against adversarial attacks because having stochastic weights reduces overfitting, with similar effect to training the original neural network with Lipschitz regularisation (Liu et al., 2018), a property with strong theoretical links to adversarial robustness (Hein & Andriushchenko, 2017). We show that there is an additional reason for their robust performance. Stochastic defences, even when averaging multiple gradient samples with EoT, tend to create a rough loss landscape that white-box adversaries find difficult to navigate. A second, and perhaps more interesting finding, is that this property is not exclusive to stochastic defences; there exist non-stochastic adversarial defences that have the same effect (Alfarra et al., 2021).

We show that the aforementioned property can be attacked by an adversary. Specifically, we propose a stochastic extension to gradient-based attacks that approximates performing the Weierstrass Transform (WT) (Bilodeau, 1962; Weierstrass, 1885) on the loss function in order to smooth it before computing its gradient. Interestingly, we find that the same method can be applied in a gradient-free setting to effectively circumvent the same type of obfuscation.

We experimentally support our insights by applying our extension to Projected Gradient Descent (PGD) (Madry et al., 2018) and recent FGSM variants (Lin et al., 2020; Wang & He, 2021) as well as Zeroth Order Optimization (ZOO) (Chen et al., 2017), in the gradient-based and gradient-free settings respectively. We demonstrate the efficacy of our loss-smoothing method against both stochas-

[*]Equal contribution [1]University of Edinburgh [2]Samsung AI Center, Cambridge. Correspondence to: Panagiotis Eustratiadis <p.eustratiadis@ed.ac.uk>, Henry Gouk <henry.gouk@ed.ac.uk>, Da Li <dali.academic@gmail.com>, Timothy Hospedales <t.hospedales@ed.ac.uk>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

tic (Eustratiadis et al., 2021; He et al., 2019; Jeddi et al., 2020; Yu et al., 2021) and non-stochastic defences (Alfarra et al., 2021) that create a rough loss surface, and damage their robust performance by as much as 20%. Finally, we analyse how the WT interacts with EoT when attacking stochastic defences. We show that these two methods serve different purposes and are complementary. However, unlike an attack that applies EoT, a WT-based attack is effective against both stochastic and non-stochastic defences.

2. Background and Related Work

2.1. Dealing with Obfuscated Gradients

In their paper, (Athalye et al., 2018a) demonstrate that many existing defences create a false impression of robustness to gradient-based adversaries by masking the gradient of the loss function from the attacker. They identify three types of gradient obfuscation: shattered, stochastic, and vanishing gradients; and show that gradient-obfuscating defences are easy to circumvent and not reliable.

Stochastic gradients, that are largely relevant to our work, stem from defences where either the weights or the activations of SNNs are sampled from a distribution (Liu et al., 2018; 2019). As a result, the gradient of their loss is also a distribution. To deal with stochastic gradients, (Athalye et al., 2018a) applied EoT (Athalye et al., 2018b), a method that repeatedly samples the target model's gradient w.r.t. the input, and computes the average of these samples to obtain the "true" gradient. Following (Athalye et al., 2018a), it has become a requirement for stochastic defence research (Eustratiadis et al., 2021; He et al., 2019; Lee et al., 2021) to incorporate a series of checks that ensure new stochastic defence methods do not owe their success to gradient obfuscation.

Expectation over Transformation We now highlight a few technical details about EoT. Let h_θ be a SNN with parameters θ, and x an input image belonging to class $c \in C$. The stochastic weights or activations of h_θ cause $h_\theta(x)$ to be randomised; as a result, $\nabla_x \mathcal{L}(h_\theta(x), c)$ is a distribution of gradients. EoT is, in essence, a Monte-Carlo sampling method that estimates the true gradient ω of the loss function by averaging n gradient samples as

$$\omega = \frac{1}{n} \sum_{i=0}^{n} \nabla_x \mathcal{L}(h_\theta^i(x), c) \ . \tag{1}$$

It is important to emphasise that the WT and EoT serve different purposes. Unlike our proposed method, detailed in Section 3, EoT has no "spatial awareness" of the loss' landscape, i.e., while applying EoT results in a better estimation of the gradient at x, it is uninformative regarding the gradient at $x + \delta$. In this paper, we demonstrate that the WT and EoT are complementary, and maximally effective when used in combination.

2.2. Defences with a Noisy Loss Landscape

We consider both stochastic and non-stochastic defences that we have found to create a rough loss surface that is difficult for gradient-based adversaries to navigate. In the case of stochastic defences, we only consider related work that have applied EoT in their model evaluation.

Parametric Noise Injection (PNI) (He et al., 2019) is a defence that equips convolutional neural network layers with additive noise drawn from an isotropic normal distribution. Learn2Perturb (L2P) (Jeddi et al., 2020) extends PNI to a richer noise model. Instead of learning a scalar intensity parameter α, a noise injection module is learned that determines the strength of parameter-wise Gaussian noise injection at each layer. Similarly to L2P, the Simple and Effective SNN (SE-SNN) (Yu et al., 2021), learns a parameter-wise noise distribution motivated by the variational information bottleneck (Alemi et al., 2017), and noise is only applied to the penultimate neural network layer. Finally, Weight-Covariance Alignment (WCA) (Eustratiadis et al., 2021) extends the noise models above to include a full covariance (anisotropic) Gaussian noise model, thus generating correlated perturbations across channels. All the mentioned approaches (Eustratiadis et al., 2021; He et al., 2019; Jeddi et al., 2020; Yu et al., 2021) include some noise-promoting regulariser to prevent the noise from shrinking to zero during training, with WCA's covariance alignment regulariser being derived from an adversarial generalisation bound in contrast to the prior models' heuristics.

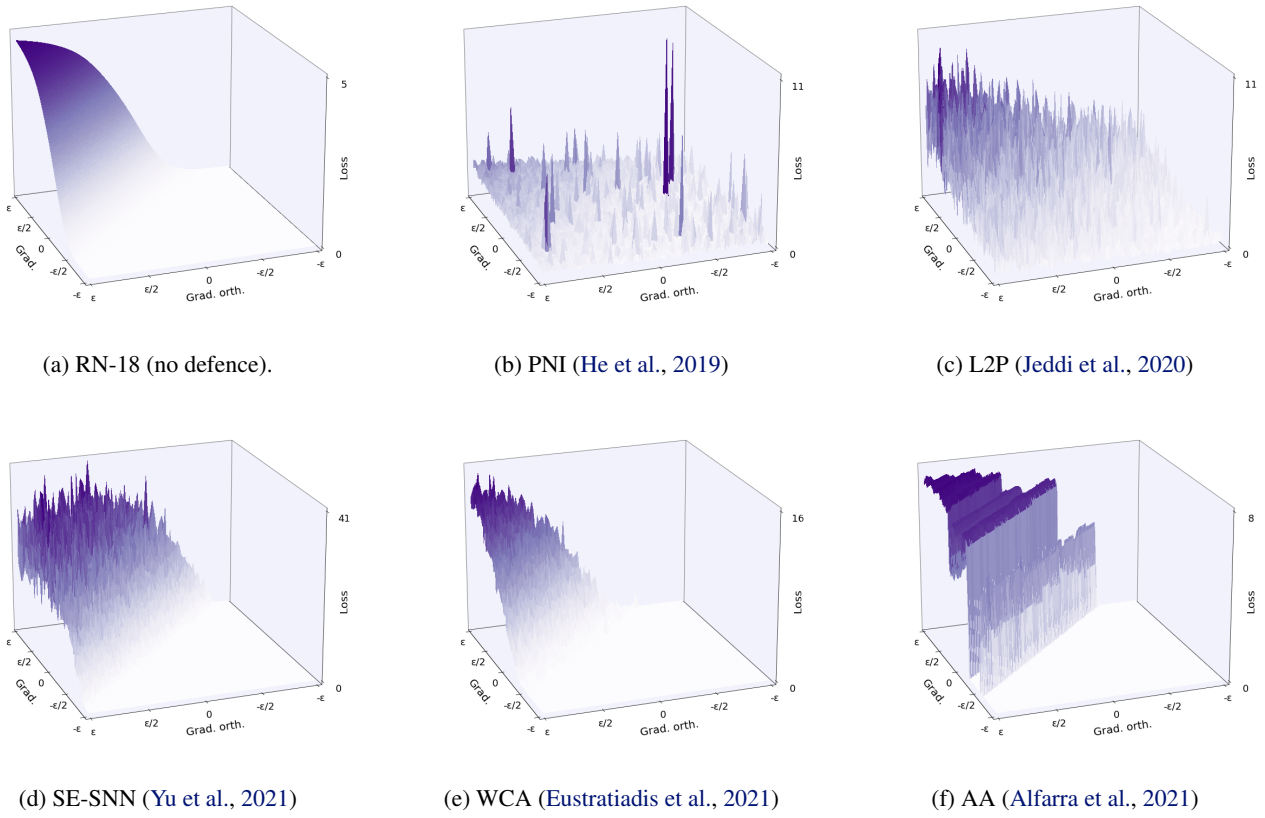
An obfuscated loss landscape is not an exclusive characteristic of SNNs. Anti-Adversaries (AA) (Alfarra et al., 2021) is a recent training-free adversarial defence that could be categorised as a "black-box" defence. It improves adversarial robustness by prepending a layer that induces discontinuity to the loss landscape.

Our observation is that all these methods defend against white-box adversarial attacks largely through inducing rough loss landscapes that gradient-based adversaries struggle to ascend. Slices through the loss landscapes of the aforementioned defences are shown in Fig. 1 and we provide further details about this figure in Appendix C.

3. Method

3.1. The Weierstrass Transform

The Weierstrass Transform (WT) (Bilodeau, 1962; Weierstrass, 1885) of a function f is defined as the convolution of f with a Gaussian kernel function k in order to obtain g, a



(a) RN-18 (no defence). (b) PNI (He et al., 2019) (c) L2P (Jeddi et al., 2020)

(d) SE-SNN (Yu et al., 2021) (e) WCA (Eustratiadis et al., 2021) (f) AA (Alfarra et al., 2021)

Figure 1. Loss landscapes of each of the adversarial defences considered in this paper. All defences use a ResNet-18 backbone and the loss surfaces are constructed on a correctly-classified test image from CIFAR-10. Sub-figure 1a shows the smooth surface of an undefended ResNet-18 for comparison. The X axis is the gradient w.r.t. the clean input image, and the Y axis is chosen to be orthogonal to X. The Z axis is the value of the loss function for each perturbation within the ϵ-ball of the input image, where $\epsilon = \frac{8}{255}$.

smoothed version of f. Formally,

$$g(x) = \int_{-\infty}^{+\infty} k(x-y)\ f(y) \cdot dy, \quad k(x) = \frac{1}{\sqrt{4\pi}}\ e^{\frac{-x^2}{4}}\ . \tag{2}$$

The conventional Weierstrass Transform (Weierstrass, 1885) is defined for functions of scalar variables and uses a Gaussian with a variance of $\sqrt{2}$. Because we are applying it to neural networks that are functions of many variables, and which may need to be smoothed to different extents, we relax these two conditions by using a multivariate Gaussian with a tuneable isotropic covariance matrix.

3.2. Using the Weierstrass Transform to Attack

Let $\mathcal{L}(h_\theta(x), c)$ be the classification loss function where x is an input image belonging to a class $c \in C$, and h_θ a function approximator with parameters θ. We can use Eq. 2 to define the smoothed loss function $\tilde{\mathcal{L}}$ as

$$\tilde{\mathcal{L}}(h_\theta(x), c) = \int_{\mathbb{R}^d} k(x-y)\ \mathcal{L}(h_\theta(y), c) \cdot dy\ , \tag{3}$$

where d is the dimensionality of x. This can also be interpreted as an expectation

$$\tilde{\mathcal{L}}(h_\theta(x), c) = \mathbb{E}_\eta[\mathcal{L}(h_\theta(x+\eta), c)], \quad \eta \sim \mathcal{N}(0, \sigma^2 I)\ . \tag{4}$$

The dimensionality of the integral in Eq. 3 corresponds to the number of input pixels; so computing it directly is computationally infeasible. However, it is possible to compute a stochastic unbiased estimate of $\tilde{\mathcal{L}}$ by using Monte-Carlo sampling,

$$\hat{\mathcal{L}}(h_\theta(x), c) = \frac{1}{m} \sum_{i=1}^{m} \mathcal{L}(h_\theta(X_i), c)\ , \tag{5}$$

where m is the number of perturbations sampled around x and

$$X_i = x + \eta_i, \quad \eta_i \sim \mathcal{N}(0, \sigma^2 I)\ . \tag{6}$$

The error introduced by this approximation of the WT is bounded (with high confidence), as shown in the following

Algorithm 1 WT-PGD

Data: x, c
Model: h_θ
Input: k, m, n, α, ϵ, σ
Output: $\tilde{x}$
$\tilde{x} \longleftarrow x + z, \quad z \sim \mathcal{U}(-\epsilon, \epsilon)$ **for** k iterations **do**
  $\tilde{X} \longleftarrow$ sample m points around $\tilde{x}$ [Eq. 6] **if** defence is stochastic **then**
    $\omega \leftarrow \frac{1}{mn} \sum_{i=0}^{m} \sum_{j=0}^{n} \nabla_x \mathcal{L}(h_\theta^j(\tilde{X}_i), c)$ [Eq. 9]
  else
    $\omega \leftarrow \frac{1}{m} \sum_{i=0}^{m} \nabla_x \mathcal{L}(h_\theta(\tilde{X}_i), c)$ [Eq. 8]
  end
  $\tilde{x} \longleftarrow \tilde{x} + \alpha\, \epsilon\, \text{sign}(\omega)$ project $\tilde{x}$ to ℓ_p-ball
end

Theorem. It can be seen that the quality of the approximation improves as the number of samples, m, is increased.

Theorem 3.1. *For a k-Lipschitz network, h_θ, applied to a fixed instance (x, c), and a loss function, $\mathcal{L}$, that is L-Lipschitz on the co-domain of h_θ, we have with probability at least $1 - \delta$ that*

$$|\hat{\mathcal{L}}(h_\theta(x), c) - \tilde{\mathcal{L}}(h_\theta(x), c)| \leq kL\sigma \sqrt{\frac{4d\ln(1/\delta)}{m}} + \frac{2kL\ln(1/\delta)}{3m} \tag{7}$$

where we assume that x is contained within the unit ball in d-dimensional Euclidean space.

The proof of Theorem 1 is provided in Appendix A.

3.3. A Stochastic WT Extension of Gradient-Based Attacks

Conceptually, any gradient-based adversary can be extended with the WT to smooth noisy loss landscapes and estimate the gradient of the loss more reliably. Algorithm 1 describes WT-PGD, our proposed method that is an extension of PGD. In addition to the standard hyperparameters of PGD, i.e., the number of iterations k, step size α, and attack strength ϵ, we add m as the number of images sampled around x, and the standard deviation σ of the zero-mean normal distribution from which the images are sampled.

The main idea is that, given enough samples in close proximity to x, we can compute the true slope of the loss function as the average slope of the surface where these samples lie. Therefore, within the context of WT-PGD, we define the true gradient ω as

$$\omega = \frac{1}{m} \sum_{i=0}^{m} \nabla_x \mathcal{L}(h_\theta(\tilde{X}_i), c) \;, \tag{8}$$

where $\tilde{X}$ denotes the set of images sampled around the perturbed image $\tilde{x}$, following Eq. 6.

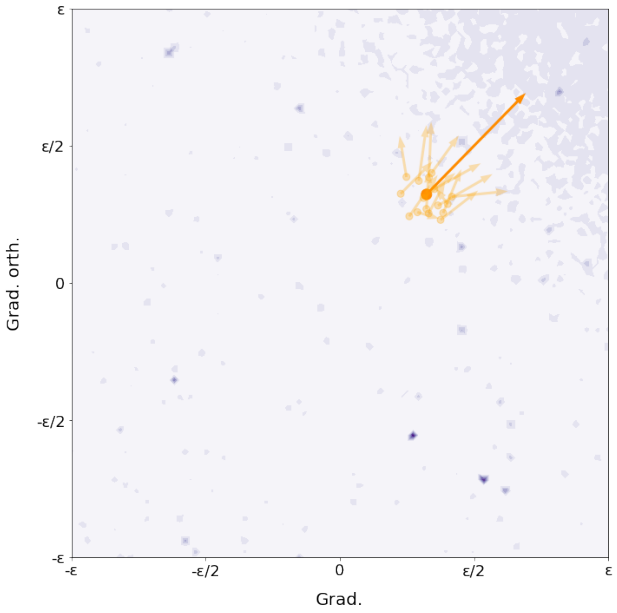


Figure 2. Illustration of the intuition behind our WT attack. This loss surface is a top-down view of PNI (He et al., 2019) from Fig. 1c. The loss landscape around x (dark orange point) is noisy and the adversary cannot find a reliable direction to follow. It therefore samples m images around x (yellow points) and follows the average gradient obtained at each of those points. Best viewed in color.

Fig. 2 illustrates the concept of this attack. While the gradient at a particular image x and samples nearby are individually noisy (random small yellow arrows), their aggregate direction (large orange arrow) ascends the loss surface.

Generalisation Properties Note that the WT only affects the gradient computation part of a gradient-based attack. In this paper we choose to illustrate the WT extension on PGD as a proof of concept, due to its convenient mathematical formulation as well as its efficacy as an attack. However, Eq. 8 can effectively replace the gradient computation step in any gradient-based adversary (Goodfellow et al., 2015; Lin et al., 2020; Wang & He, 2021).

3.3.1. Integration with EoT

When we use Eq. 5 and 6 to smooth the loss landscape of a stochastic defence, the gradient w.r.t. the input x, $\nabla_x \mathcal{L}(h_\theta(\tilde{X}), c)$, remains stochastic (Athalye et al., 2018a). It is therefore sensible to apply EoT (Athalye et al., 2018b) on the sampled $\tilde{X}$, and average over the output distribution of h_θ. Incorporating Eq. 1 into Eq. 8 we get

$$\omega = \frac{1}{mn} \sum_{i=0}^{m} \sum_{j=0}^{n} \nabla_x \mathcal{L}(h_\theta^j(\tilde{X}_i), c) \;. \tag{9}$$

Table 1. Robust accuracy % of WT-PGD on CIFAR. All defences use a RN-18 backbone.

	CIFAR-10				CIFAR-100			
Method	PGD_{10}	$WT\text{-}PGD_{10}$	PGD_{100}	$WT\text{-}PGD_{100}$	PGD_{10}	$WT\text{-}PGD_{10}$	PGD_{100}	$WT\text{-}PGD_{100}$
PNI	49.4	34.8 (-14.6)	31.4	13.7 (-17.7)	22.2	17.9 (-4.3)	10.1	9.4 (-0.7)
L2P	56.1	47.2 (-8.9)	20.5	18.2 (-2.3)	26.1	11.5 (-14.6)	18.4	10.3 (-8.1)
SE-SNN	39.8	21.3 (-18.5)	13.9	12.5 (-1.4)	18.6	8.0 (-10.6)	15.9	5.9 (-10.0)
WCA	61.7	53.3 (-8.4)	58.6	37.6 (-21.0)	41.7	27.4 (-14.3)	39.0	10.8 (-28.2)
AA	63.2	43.9 (-19.3)	43.6	25.9 (-17.7)	47.9	29.6 (-18.3)	43.6	21.2 (-22.4)

Table 2. Robust accuracy % of WT-PGD on CIFAR-100 and Imagenette (full-resolution). All defences use a WRN-34-10 backbone.

	CIFAR-100				Imagenette			
Method	PGD_{10}	$WT\text{-}PGD_{10}$	PGD_{100}	$WT\text{-}PGD_{100}$	PGD_{10}	$WT\text{-}PGD_{10}$	PGD_{100}	$WT\text{-}PGD_{100}$
PNI	51.6	32.5 (-19.1)	48.4	31.3 (-17.1)	51.8	39.6 (-12.2)	42.3	24.3 (-18.0)
L2P	45.3	32.4 (-12.9)	40.0	29.5 (-10.5)	63.4	46.9 (-16.5)	42.4	23.2 (-19.2)
SE-SNN	44.6	34.9 (-9.7)	46.0	31.0 (-15.0)	47.2	22.9 (-24.3)	41.1	21.7 (-19.4)
WCA	63.6	54.5 (-9.1)	56.7	44.5 (-12.2)	67.5	51.0 (-16.5)	50.3	35.6 (-14.7)
AA	76.1	59.2 (-16.9)	62.4	54.0 (-8.4)	69.3	44.8 (-24.5)	57.1	39.4 (-17.7)

Table 3. Robust accuracy % of SI-NI-FGSM (F1, (Lin et al., 2020)) and VMI-FGSM (F2, (Wang & He, 2021)) attacks and their respective WT extensions on CIFAR (RN-18 backbone) and Imagenette (WRN-34-10 backbone). Names are shortened for better readability.

	CIFAR-10				CIFAR-100				Imagenette			
Method	(F1)	WT-(F1)	F2	WT-(F2)	(F1)	WT-(F1)	F2	WT-(F2)	(F1)	WT-(F1)	F2	WT-(F2)
PNI	48.2	35.5 (-12.7)	38.3	27.4 (-10.9)	24.9	13.0 (-11.9)	25.7	18.6 (-7.1)	47.4	37.2 (-10.2)	42.5	33.2 (-9.3)
L2P	56.1	44.9 (-11.2)	31.7	19.2 (-12.5)	27.2	18.5 (-8.7)	30.1	21.0 (-9.1)	59.6	46.1 (-13.5)	42.4	30.5 (-11.9)
SE-SNN	40.5	31.6 (-8.9)	38.1	22.8 (-15.3)	25.3	12.2 (-13.1)	28.9	15.0 (-13.9)	44.8	33.9 (-10.9)	40.7	38.4 (-2.3)
WCA	58.5	54.0 (-4.5)	55.7	34.8 (-20.9)	45.8	30.4 (-15.4)	44.0	33.2 (-10.8)	64.0	59.0 (-5.0)	51.6	42.3 (-9.3)
AA	61.8	53.6 (-8.2)	58.0	41.4 (-16.6)	46.7	31.8 (-14.9)	41.1	23.3 (-17.8)	66.5	49.3 (-17.2)	56.9	43.0 (-13.9)

Table 4. Ablation: Effect of the WT and EoT individually against stochastic defences. The scores are the robust accuracy % on CIFAR-10.

(Attack: $WT\text{-}PGD_{10}$)	$WT_1 + EoT_1$	$WT_1 + EoT_{16}$	$WT_{16} + EoT_1$	$WT_{16} + EoT_{16}$
PNI	50.6	49.1	48.7	34.8
L2P	58.9	54.4	55.0	47.2
SE-SNN	46.6	39.5	39.7	21.3
WCA	72.0	58.4	61.1	53.3

A thorough empirical analysis of how the WT interacts with EoT is presented in Section 4.3, along with an ablation study for each individual component.

3.4. A Stochastic WT Extension of Gradient-Free Attacks

Although we primarily focus on the WT as an extension of gradient-based attacks, its potential impact when applied to gradient-free attacks cannot be ignored. In this Appendix A.1 we demonstrate WT's generality by integrating it with ZOO (Chen et al., 2017), a black-box adversary that uses gradient approximation instead of surrogate models (Chen et al., 2017; Papernot et al., 2016; 2017), assuming access only to the per-class posterior $h(x)$.

Table 5. Robust accuracy % of gradient-free WT-ZOO on CIFAR-10 (RN-18 backbone) and Imagenette (WRN-34-10 backbone).

	CIFAR-10		Imagenette	
Method	ZOO	WT-ZOO	ZOO	WT-ZOO
PNI	62.1	54.3 (-7.8)	59.2	41.0 (-18.2)
L2P	63.7	56.1 (-7.6)	65.8	54.3 (-11.5)
SE-SNN	59.4	44.3 (-15.1)	49.8	37.6 (-12.2)
WCA	70.9	64.8 (-6.1)	72.3	61.9 (-10.4)
AA	74.1	66.5 (-7.6)	77.9	60.6 (-17.3)

4. Experiments

4.1. Experimental Setup

In our experiments we consider four stochastic defences (PNI (He et al., 2019), L2P (Jeddi et al., 2020), SE-SNN (Yu et al., 2021) and WCA (Eustratiadis et al., 2021)) and one non-stochastic (AA (Alfarra et al., 2021)). For fair comparison these defences use the same backbone architecture, ResNet-18 (RN-18) and Wide ResNet-34-10 (WRN-34-10) (He et al., 2016; Zagoruyko & Komodakis, 2016) in the corresponding experiments. We evaluate their performance against the gradient-based WT-PGD$_{10}$ and WT-PGD$_{100}$, and the gradient-free WT-ZOO. The hyperparameter setting for our experiments is outlined in Appendix B.

4.2. Quantitative Evaluation

In Tables 1 and 2 we report the accuracy of our selection of adversarial defences when under our WT-PGD attack against the baselines. It is evident that WT-PGD outperforms PGD consistently across defences, benchmarks, for different attack strength and network depth. In particular, we can see that: (i) Every defence considered suffers substantially. (ii) Weaker defences are broken near completely, with L2P and SE-SNN failing on CIFAR-10; and PNI, L2P and SE-SNN failing on CIFAR-100. (iii) The stronger WCA and AA defences tend to suffer large hits, especially under WT-PGD$_{100}$. (iv) Our attack is particularly strong with high-resolution Imagenette images (average $\approx 450\times$ pixels), with most defenses suffering over 15% performance reduction.

To show the generality of our method, we apply the WT extension to the more sophisticated and recently proposed gradient-based adversaries NI-FGSM (Lin et al., 2020) and VMI-FGSM (Wang & He, 2021) that use Nesterov's acceleration and variance tuning to improve attack strength and transferability. Table 3 shows results consistent with our previous evaluation, and proves that our loss-smoothing method can effectively strengthen recent, more sophisticated attacks. Finally, in Table 5 we present our evaluation of WT-ZOO. It is evident that even though (i) the performance reduction is on average slightly lower than the gradient-based setting and (ii) WT-ZOO imposes an additional query-efficiency cost, WT-ZOO is still successful in attacking these obfuscating defences.

These empirical results support our claim that rugged loss surfaces can be attacked, and equipping adversaries with the capability to smooth the loss surface makes them significantly stronger against this type of gradient obfuscation.

4.3. Interaction between WT and EoT

In this Section we analyse how the WT interacts with EoT when attacking stochastic defences. An ablation study is presented in Table 4, where we evaluate the two methods individually and in combination when attacking PNI, L2P, SE-SNN and WCA. We start by setting the baseline to regular PGD; which is equivalent to WT-PGD with 1 WT sample (x itself) and 1 iteration of EoT. We then vary each of the two components by setting the number of WT samples and EoT iterations to 16, to keep consistent with our evaluation in Section 4.2. The last column of Table 4 is the same as the second column of Table 1.

Our ablation study shows that, while each method increases attack strength, neither is significantly better than the other in terms of individual performance. We conclude the WT and EoT are most effective when used in combination, to deal with the noisy loss landscape and the stochastic gradients respectively. Further analysis on the ablation study is provided in Appendix D.

5. Conclusions

We reveal a new form of gradient obfuscation that can be a property of stochastic, as well as non-stochastic adversarial defences. This gradient obfuscation occurs when a defence creates a noisy loss landscape to mislead gradient-based adversaries. This does not constitute an adequate defence, and can be circumvented by smoothing the surface of the loss function before following the gradient w.r.t. the input. We propose a smoothing method with which gradient-based, as well as gradient-free adversaries can be extended, utilising a Monte-Carlo variant of the Weierstrass transform. As demonstrated by our proposed algorithms, WT-PGD and WT-ZOO, this extension enables strong, successful attacks. We further illustrate the smoothing capabilities of our adversary beyond the quantitative evaluation presented in Section 4.2, by plotting the loss surfaces of the defences before and after WT smoothing (Fig. 1 and 4). We hope that highlighting this novel type of attack against this class of adversarial defences will inspire future research to avoid relying on this weak defence strategy for robustness.

The source code for WT-PGD, WT-ZOO, and our diagnostic tool for visualising a loss landscape is available on GitHub: `https://github.com/peustr/wt-pgd`.

Acknowledgements

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) Grant number EP/S000631/1; and the MOD University Defence Research Collaboration (UDRC) in Signal Processing.

References

Alemi, A. A., Fischer, I., Dillon, J. V., and Murphy, K. Deep variational information bottleneck. *ICLR*, 2017.

Alfarra, M., Pérez, J. C., Thabet, A., Bibi, A., Torr, P. H. S., and Ghanem, B. Combating adversaries with anti-adversaries. In *ICML*, 2021.

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

Athalye, A., Engstrom, L., Ilyas, A., and Kwok, K. Synthesizing robust adversarial examples. In *ICML*, 2018b.

Bilodeau, G. G. The weierstrass transform and hermite polynomials. *Duke Mathematical Journal*, 29(2):293–308, 1962.

Carmon, Y., Raghunathan, A., Schmidt, L., Duchi, J. C., and Liang, P. Unlabeled data improves adversarial robustness. In *NeurIPS*, 2019.

Chen, P., Zhang, H., Sharma, Y., Yi, J., and Hsieh, C. ZOO: zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *ACM*, 2017.

Croce, F., Andriushchenko, M., Sehwag, V., Debenedetti, E., Flammarion, N., Chiang, M., Mittal, P., and Hein, M. RobustBench: a standardized adversarial robustness benchmark. In *NeurIPS Datasets and Benchmarks Track*, 2021.

Eustratiadis, P., Gouk, H., Li, D., and Hospedales, T. M. Weight-covariance alignment for adversarially robust neural networks. In *ICML*, 2021.

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. In *ICLR*, 2015.

Gowal, S., Qin, C., Uesato, J., Mann, T. A., and Kohli, P. Uncovering the limits of adversarial training against norm-bounded adversarial examples. *CoRR*, abs/2010.03593, 2020.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *CVPR*, 2016.

He, Z., Rakin, A. S., and Fan, D. Parametric noise injection: Trainable randomness to improve deep neural network robustness against adversarial attack. In *CVPR*, 2019.

Hein, M. and Andriushchenko, M. Formal guarantees on the robustness of a classifier against adversarial manipulation. In *NIPS*, 2017.

Hendrycks, D., Lee, K., and Mazeika, M. Using pre-training can improve model robustness and uncertainty. In *ICML*, 2019.

Howard, J. Imagenette, 2019. URL https://github.com/fastai/imagenette/.

Jeddi, A., Shafiee, M. J., Karg, M., Scharfenberger, C., and Wong, A. Learn2perturb: An end-to-end feature perturbation learning to improve adversarial robustness. In *CVPR*, 2020.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. *Toronto.edu [Online]. Available: https://www.cs.toronto.edu/ kriz*, 2009.

Kurakin, A., Goodfellow, I., Bengio, S., Dong, Y., Liao, F., Liang, M., Pang, T., Zhu, J., Hu, X., Xie, C., et al. Adversarial attacks and defences competition. *arXiv preprint arXiv:1804.00097*, 2018.

Lafferty, J., Liu, H., and Wasserman, L. Concentration of measure, 2010.

Lee, S., Kim, H., and Lee, J. Graddiv: Adversarial robustness of randomized neural networks via gradient diversity regularization, 2021.

Lin, J., Song, C., He, K., Wang, L., and Hopcroft, J. E. Nesterov accelerated gradient and scale invariance for adversarial attacks. In *ICLR*, 2020.

Liu, X., Cheng, M., Zhang, H., and Hsieh, C. Towards robust neural networks via random self-ensemble. In *ECCV*, 2018.

Liu, X., Li, Y., Wu, C., and Hsieh, C. Adv-bnn: Improved adversarial defense through robust bayesian neural network. In *ICLR*, 2019.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018.

Papernot, N., McDaniel, P. D., and Goodfellow, I. J. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. *CoRR*, abs/1605.07277, 2016.

Papernot, N., McDaniel, P. D., Goodfellow, I. J., Jha, S., Celik, Z. B., and Swami, A. Practical black-box attacks against machine learning. In *AsiaCCS*, 2017.

Rade, R. and Moosavi-Dezfooli, S.-M. Helper-based adversarial training: Reducing excessive margin to achieve a better accuracy vs. robustness trade-off. In *ICML 2021 Workshop on Adversarial Machine Learning*, 2021.

Rebuffi, S., Gowal, S., Calian, D. A., Stimberg, F., Wiles, O., and Mann, T. A. Fixing data augmentation to improve adversarial robustness. *CoRR*, abs/2103.01946, 2021.

Sridhar, K., Sokolsky, O., Lee, I., and Weimer, J. Improving neural network robustness via persistency of excitation. *CoRR*, abs/2106.02078, 2021.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I. J., and Fergus, R. Intriguing properties of neural networks. In *ICLR*, 2014.

Wang, X. and He, K. Enhancing transferability of adversarial attacks through variance tuning. In *CVPR*, 2021.

Wang, Y., Zou, D., Yi, J., Bailey, J., Ma, X., and Gu, Q. Improving adversarial robustness requires revisiting misclassified examples. In *ICLR*, 2020.

Weierstrass, K. Über die analytische darstellbarkeit sogenannter willkürlicher functionen einer reellen veränderlichen. *Sitzungsberichte der Königlich Preußischen Akademie der Wissenschaften zu Berlin*, 2:633–639, 1885.

Wu, D., Xia, S., and Wang, Y. Adversarial weight perturbation helps robust generalization. In *NeurIPS*, 2020.

Yu, T., Yang, Y., Li, D., Hospedales, T., and Xiang, T. Simple and effective stochastic neural networks. In *AAAI*, 2021.

Zagoruyko, S. and Komodakis, N. Wide residual networks. In *BMVC*, 2016.

Zhang, J., Zhu, J., Niu, G., Han, B., Sugiyama, M., and Kankanhalli, M. S. Geometry-aware instance-reweighted adversarial training. In *ICLR*, 2021.

A. Proof of Theorem 1

Proof. The proof is based on using a Bernstein inequality. Let Z_1, ..., Z_m be independent random variables taking positive values in $[a, b]$, and let $S = \frac{1}{m}\sum_i^m Z_i$. From (Lafferty et al., 2010), Bernstein's inequality tells that

$$P(|S - \mathbb{E}[S]| > t) \leq 2\text{exp}\left(\frac{-mt^2}{2\text{Var}[S] + \frac{2}{3}rt}\right), \tag{10}$$

where $r = b - a$. By setting $\delta = P(|S - \mathbb{E}[S]| > t)$ this can be rearranged to show that, with probability at least $1 - \delta$,

$$|S - \mathbb{E}[S]| \leq \sqrt{\frac{2\text{Var}[S]\ln(1/\delta)}{m}} + \frac{2r\ln(1/\delta)}{3m}. \tag{11}$$

The result follows from using $Z_i = \mathcal{L}(h_\theta(X_i), c)$ and upper bounding $\text{Var}[S]$ and r. Because h_θ is k-Lipschitz and $\mathcal{L}$ is L-Lipschitz on the co-domain of h_θ, we can say that $\mathcal{L}(h_\theta(\cdot), \cdot)$ is kL-Lipschitz. From this Lipschitz property, we know that $b \leq a + kL$, and therefore $r \leq kL$.

Denote by X_i' and S' random variables that follow the same distribution as X_i and S, respectively. The bound for the variance arises from

$$\begin{aligned}
&\text{Var}[S] && (12)\\
&= \mathbb{E}_S[(\mathbb{E}_{S'}[S'] - S)^2] && (13)\\
&\leq \mathbb{E}_{X_i}\mathbb{E}_{X_i'}\left[\left(\frac{1}{m}\sum_{i=1}^m(\mathcal{L}(h_\theta(X_i'), c) - \mathcal{L}(h_\theta(X_i), c))\right)^2\right] && (14)\\
&\leq \mathbb{E}_{X_i}\mathbb{E}_{X_i'}\left[\|X_i' - X_i\|_2^2 k^2 L^2\right] && (15)\\
&= 2k^2L^2d\sigma^2, && (16)
\end{aligned}$$

where the first inequality is due to Jensen's inequality, and the second is from the Lipschitz property of the model. The final equality arises because $X' - X \sim \mathcal{N}(0, 2\sigma^2 I)$, and the expected value of the squared Euclidean norm of a sample from a Gaussian distribution is the trace of the covariance matrix. □

A.1. A Stochastic WT Extension of Gradient-Free Attacks

Given an input image x and a pixel coordinate ρ, ZOO iteratively constructs a perturbation δ on x_ρ as

$$\delta(x, c) = \begin{cases} -\alpha\hat{g}_\rho(x, c) & \hat{h}_\rho \leq 0 \\ -\alpha\frac{\hat{g}_\rho(x,c)}{\hat{h}_\rho(x,c)} & \text{otherwise} \end{cases}, \tag{17}$$

Algorithm 2 WT-ZOO (Newton's Coordinate Descent)

Data: x^d, c
Model: h
Input: k, m, n, α, ϵ, σ
Output: $\tilde{x}$
for k iterations **do**
  Randomly pick coordinates $\vec{\rho} \in \{1, \ldots, d\}$ $\tilde{X} \longleftarrow$ sample m points around $\tilde{x}$ [Eq. 6] **if** defence is stochastic **then**
    $\delta^* \leftarrow \frac{1}{mn}\sum_{i=0}^m\sum_{j=0}^n \delta_j(X_i, c)$ [Eq. 20]
  else
    $\delta^* \leftarrow \frac{1}{m}\sum_{i=0}^m \delta(X_i, c)$ [Eq. 19]
  end
  $\tilde{x} \longleftarrow \tilde{x} + \delta^*$ project $\tilde{x}$ to ℓ_p-ball
end

where α denotes the learning rate. $\hat{g}_i$ and $\hat{h}_i$ are the first- and second-order approximate gradients of a hinge-like loss function

$$f(x, c_0) = \max\{\log h(x)_{c_0} - \max_{c \neq c_0} \log h(x)_c, -\kappa\} \ , \tag{18}$$

where $\kappa \geq 0$. Algorithm 2 details WT-ZOO. Note that the principle behind the WT extension remains the same as in the white-box setting. Adapting Eq. 8 and 9 with ZOO's gradient approximation (Eq. 17) we respectively get

$$\delta^* = \frac{1}{m} \sum_{i=0}^{m} \delta(X_i, c) \ , \tag{19}$$

and for stochastic defences

$$\delta^* = \frac{1}{mn} \sum_{i=0}^{m} \sum_{j=0}^{n} \delta_j(X_i, c) \ . \tag{20}$$

As ZOO estimates gradients with finite difference it is susceptible to being mislead by a rough loss surface (Fig. 1). Smoothing the loss estimates at each point improves the quality of approximate gradient estimation for the ZOO attacker.

B. Experimental Setup: Hyperparameters

For WT-PGD, we set an attack strength of $\epsilon = 8/255$ and a step size of $\alpha = 0.01$, as is standard practice. For WT-ZOO we set $k = 100$ and $\alpha = 0.01$. The number of WT samples and EoT iterations in our main experiments are both set to $m = n = 16$. We justify this hyperparameter choice in the analysis of Appendix D. Finally, selecting an appropriate value for σ is important. If the value of σ is too high, then the WT samples will be too far from x, lying on points too dissimilar to x to provide an informative gradient signal. If the value of σ is too low, the sampled points will be too close to x, and there will be no smoothing effect. We found that $\sigma = 0.05$ is a suitable value for normalized images, and use it across all experiments. In terms of benchmark datasets, we consider CIFAR-10, CIFAR-100 (Krizhevsky et al., 2009) and Imagenette (Howard, 2019) with full-resolution images. We deem this selection of datasets suitable because they offer a diversity in complexity, number of classes, and image resolution.

It should be mentioned that in the case of AA we do not apply EoT, as it is not a stochastic defence and therefore does not produce stochastic gradients. In addition, all stochastic models evaluated in this paper are retrained, following the instructions in the original published material, when available. As a result, the accuracy scores may not exactly reflect the scores from the original papers.

C. Visualising the Loss Landscapes

In this Section, we describe a diagnostic method that we use to visually identify whether an adversarial defence produces a noisy loss landscape, and to generate the visualisations in Fig. 1 and 4.

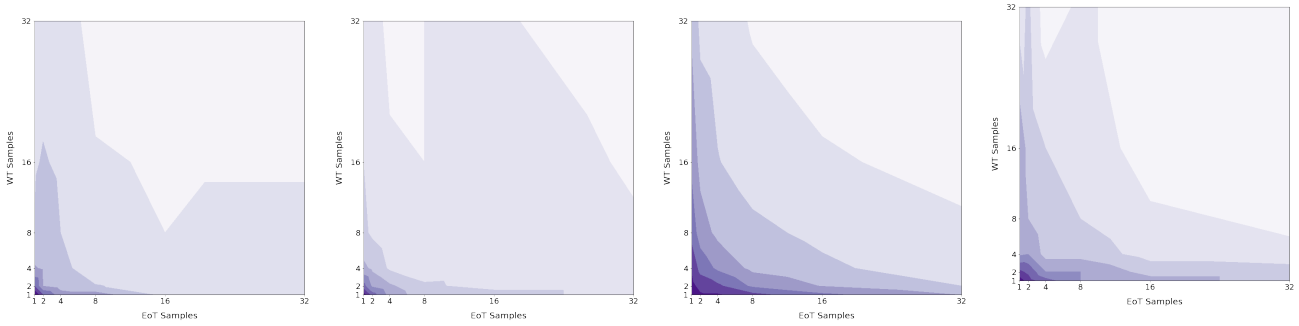
Given an unperturbed input image x that the target model h_θ classifies correctly as class c, we compute the gradient of the loss w.r.t. x as $g_1 = \nabla_x \mathcal{L}(h_\theta(x, c))$. We then arbitrarily choose a dimension g_2, such that $g_1 \perp g_2$. Finally, we create evenly-spaced query images (and potential adversarial examples) $\tilde{x_i}$ in the ϵ-ball of x as

$$\tilde{x_i} = x + \epsilon_1 \operatorname{sign}(g_1) + \epsilon_2 \operatorname{sign}(g_2) \ , \tag{21}$$

where $\epsilon_1, \epsilon_2 \in [-\frac{8}{255}, \frac{8}{255}]$, and project their calculated loss values $\mathcal{L}(h_\theta(\tilde{x_i}, c))$ to the g_1 and g_2 axes.

Fig. 1 shows the above 2D slice through the loss landscapes of PNI, L2P, SE-SNN, WCA, and AA defences. In Fig. 4 we show the corresponding smoothed loss landscapes, when under attack by WT-PGD, side-by-side for easier means of visual comparison. Further, Appendix E includes the loss surfaces of the highest scoring non-stochastic adversarial defences listed in RobustBench (Croce et al., 2021), to give the reader a frame of reference of how non-rugged loss landscapes should look like in state-of-the-art defences.

In the case of AA, recall that when an input x is queried, it returns $(\mathcal{L}(h_\theta(x+\gamma), c), \nabla_x \mathcal{L}(h_\theta(x+\gamma), c))$, where γ is chosen anti-adversarially, with respect to the current prediction $h_\theta(x)$. This has the effect of sharpening the loss landscape w.r.t. x (compare Fig. 1f and Fig. 1a), making correctly classified queries become less adversarial. This also has the effect of making adversarial (incorrectly classified) queries become *more* adversarial. Any successful attack on a vanilla ResNet will



(a) PNI (He et al., 2019) (b) L2P (Jeddi et al., 2020) (c) SE-SNN (Yu et al., 2021) (d) WCA (Eustratiadis et al., 2021)

Figure 3. Analysis of the interaction between WT and EoT on stochastic defences. WT and EoT are complementary. Neither can achieve peak performance alone, and best performance requires combining them (lighter color = lower accuracy).

also be successful against AA, but the gradient information of unsuccessful queries is obfuscated to make successful attacks harder to find. By stochastically smoothing this sharpened loss landscape (recall Fig. 2), our WT attack reveals some of the previously hidden true gradients (see smoothed Fig. 4j and 4k compared to the sharper Fig. 1f).

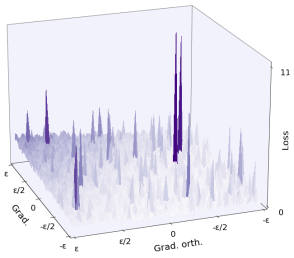
D. Ablation Study: Selection of m and n

We also conduct an experiment using a grid of EoT and WT samples from $\{1, 2, 4, 8, 16, 32\}$. Fig. 3 presents an overhead plot of the resulting network accuracy as a function of number of samples for each of EoT and WT. Darker colors indicate higher accuracy, starting from the point (1, 1), i.e., 1 iteration of EoT and 1 WT sample (the input image itself). We see that: (i) After (16, 16) the performance of the attack quickly saturates across all defences. This justifies our use of $m = n = 16$ samples in the main experiment. (ii) Even at the limit of 32 samples, neither attack method on its own performs as well as their combination. This shows that simply increasing the number of EoT samples can not replicate the effect of WT (and vice-versa).

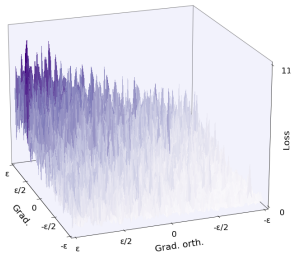
E. Strong Defences with Smooth Loss Landscapes

In the main paper, we see the effect of our attack on gradient-obfuscating adversarial defences that construct a noisy loss landscape to confuse the adversary. To further support future adversarial defence research, in this Section we want to inform the reader about how the loss landscapes of non-obfuscating defences should look like.

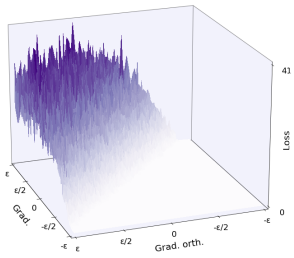
To that end, we choose the 9 highest-scoring adversarial defences from the ℓ_∞ CIFAR-10 leaderboard of the widely used RobustBench (Croce et al., 2021) and visualise their loss landscapes in Fig. 5. The visualisation method is the same that produced Fig. 1 of the main paper; except that none of the defences are stochastic and therefore EoT is not used to obtain better gradient estimates.



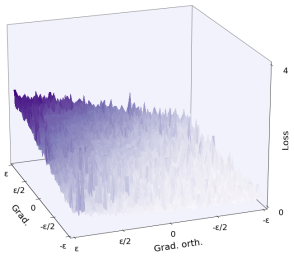
(a) PNI (He et al., 2019)



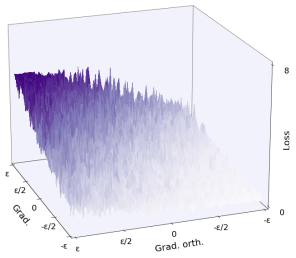
(b) L2P (Jeddi et al., 2020)



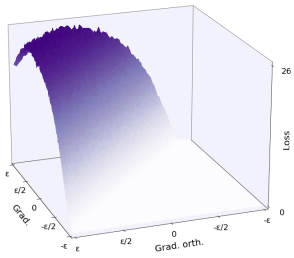
(c) SE-SNN (Yu et al., 2021)



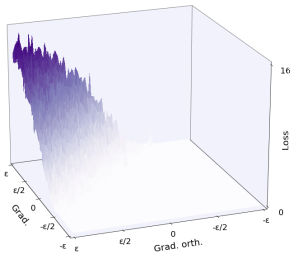
(d) PNI + WT-PGD



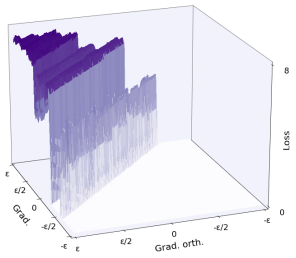
(e) L2P + WT-PGD



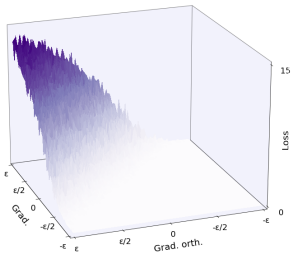
(f) SE-SNN + WT-PGD



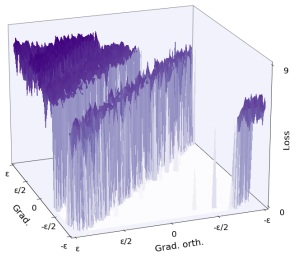
(g) WCA (Eustratiadis et al., 2021)



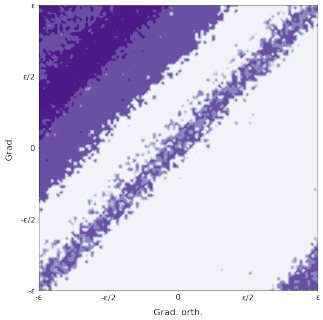
(h) AA (Alfarra et al., 2021)



(i) WCA + WT-PGD

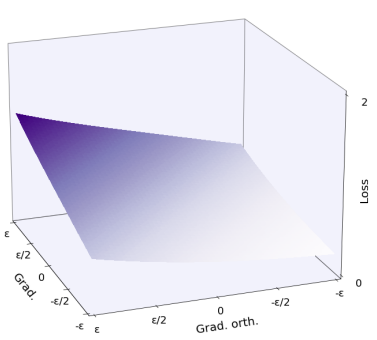


(j) AA + WT-PGD

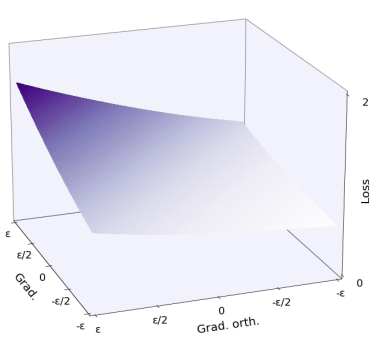


(k) AA + WT-PGD (top-down)

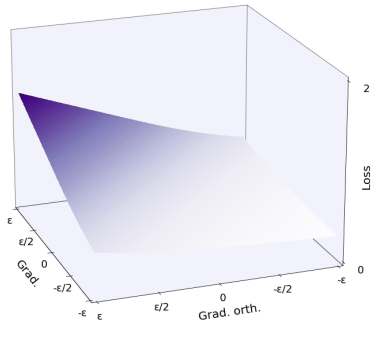
Figure 4. Loss landscapes of PNI, L2P, SE-SNN, WCA, and AA when under attack by WT-PGD. For AA, we show the surface plot and a 2D contour plot (top-down view) for better interpretability. The WT has smoothed the landscapes compared to those shown in Fig. 1.



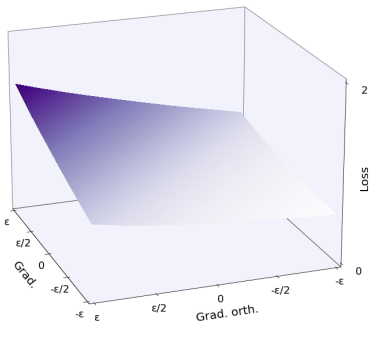
(a) Rebuffi et al. (Rebuffi et al., 2021)



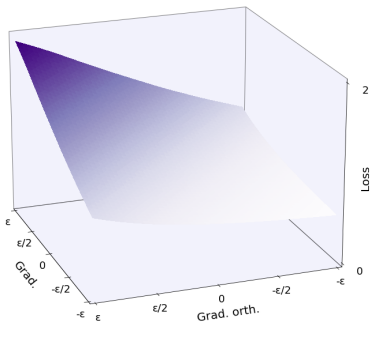
(b) Gowal et al. (Gowal et al., 2020)



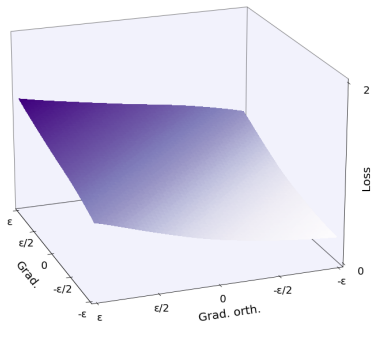
(c) Rade et al. (Rade & Moosavi-Dezfooli, 2021)



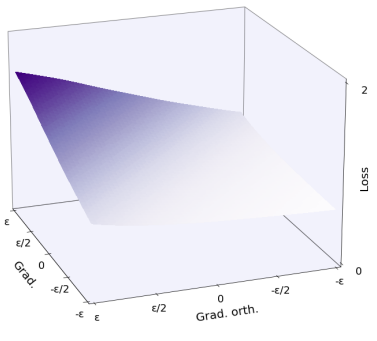
(d) Sridhar et al. (Sridhar et al., 2021)



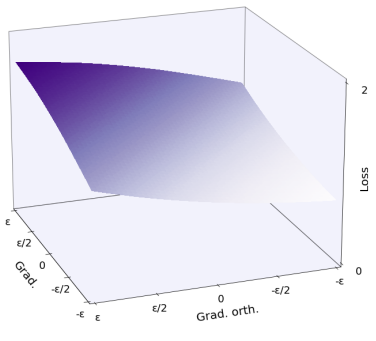
(e) Wu et al. (Wu et al., 2020)



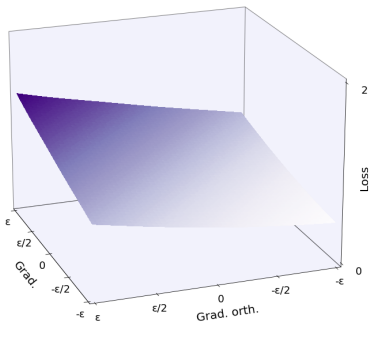
(f) Zhang et al. (Zhang et al., 2021)



(g) Carmon et al. (Carmon et al., 2019)



(h) Wang et al. (Wang et al., 2020)



(i) Hendrycks et al. (Hendrycks et al., 2019)

Figure 5. Landscapes of non-obfuscating adversarial defences that score competitively on RobustBench (Croce et al., 2021).