ACM IWSPA 2022
This is a self-archived pre-print version of this article.
The final publication is available at ACM via
https://doi.org/10.1145/3510548.3519370.

An Empirical Evaluation of Adversarial Examples Defences, Combinations and Robustness Scores[*]

Aleksandar Jankovic
a.jankovic1993@gmail.com
Vienna University of Technology
Vienna, Austria

Rudolf Mayer
rmayer@sba-research.org
SBA Research & Vienna University of Technology
Vienna, Austria

ABSTRACT

Over the past few years, deep learning has been dominating the field of machine learning in applications such as speech, image, and text recognition, which lead to an increased use of deep learning techniques in safety-critical tasks. However, Neural Networks are vulnerable to adversarial examples, i.e. well-crafted small perturbations of the input that aim to disturb the prediction correctness. Therefore, robustness and security of deep learning models has become a major concern, indirectly also affecting safety.

In this paper, we therefore evaluate several state-of-the-art white- and black-box adversarial attacks against Convolutional Neural Networks for image recognition, for various attack targets. Further, defences such as adversarial training and pre-processors are evaluated. Moreover, we investigate whether combinations of them can improve these defences. Finally, we examine whether attack-agnostic robustness scores such as CLEVER are able to correctly estimate the robustness against our large range of attack.

Our results indicate that pre-processors are very effective against attacks with adversarial examples that are very close to the original images, that combinations can improve the defence strength, and that CLEVER is insufficient as the sole indicator of robustness.

CCS CONCEPTS

• Security and privacy; **• Computing methodologies → Supervised learning**;

KEYWORDS

Adversarial Examples, Defences, Robustness Scores

ACM Reference Format:
Aleksandar Jankovic and Rudolf Mayer. 2022. An Empirical Evaluation of Adversarial Examples Defences, Combinations and Robustness Scores. In *Proceedings of the 2022 ACM International Workshop on Security and Privacy Analytics (IWSPA '22), April 24–27, 2022, Baltimore, MD, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3510548.3519370

1 INTRODUCTION

With increases in performance of machine learning (ML), primarily fuelled by deep learning (DL), in modalities such as speech, image, and text, and their practical significance, these techniques are also increasingly employed in critical settings. For example, convolutional neural networks (CNNs) are used e.g. to recognise road signs. As ML is employed in an increasing number of systems involved in potentially autonomous decision making, the robustness and security of ML models becomes of concern: autonomous vehicles failing to recognise a STOP sign could result in major incidents, and ML system thus indirectly also affect safety. In recent years, several methods to successfully attack ML processes were demonstrated.

In this paper, we address evasion attacks such as adversarial examples. Here, the attacker manipulates the test data (at prediction time) by applying (minor) perturbations, with the goal of maximising the error. In many cases, these modifications are so subtle that a human observer does not notice them, but the model still makes mistakes. Even worse, the attacked model usually reports high confidence for the wrong prediction. It has been shown that many DL techniques are vulnerable to evasion attacks, sometimes even if the adversary has no access to the underlying model.

On the other hand, a number of defences have been proposed to mitigate the success of evasion attacks. Some techniques are based on pre-processing the inputs, with the aim to remove or disturb the perturbations, while aim to train the learned model to be aware of potential attack patterns. Only few mechanisms can defend against multiple attack types, which leads to the idea to combine them.

To measure the robustness of a model against adversarial attacks is of great interest, as that allows to estimate potential risks upfront. One such metric, CLEVER, is attack-agnostic, i.e. it should generalise to a range of attacks and predict the robustness. However, there is still a lack of an in-depth evaluation of this correlation.

Our contributions in this paper are as follows.

- We provide an in-depth evaluation of several attacks, on various datasets and against different CNN architectures
- We evaluate multiple attack target settings: untargeted, least likely class, and "next" class (i.e. next higher index)
- We compare existing defences, and combine them in various ways to achieve a better protection of the models
- We measure the robustness estimation of the CLEVER metric

Threat model. We define the attacker's goal, knowledge, and capabilities mainly based on the discussion by Biggio and Roli [1].

The **attacker's goal** is defined in the following terms.
Security violation: the attacker aims to cause either an integrity violation, i.e. to evade detection without compromising normal system operation, or an availability violation, i.e. to compromise the normal system functionalities available to legitimate users.

[*]SBA Research (SBA-K1) is a COMET Centre within the framework of COMET – Competence Centers for Excellent Technologies Programme and funded by BMK, BMDW, and the federal state of Vienna; COMET is managed by FFG
This work is supported by BMK and FFG under Grant No. 873979 (project PRIMAL).

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

IWSPA '22, April 24–27, 2022, Baltimore, MD, USA

© 2022 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 978-1-4503-9230-3/22/04…$15.00
https://doi.org/10.1145/3510548.3519370

Attack specificity: targeted attacks aim to cause the model to misclassify a specific set of samples (e.g. a given user), while untargeted attacks aim to misclassify any sample. We consider both variants. *Error specificity*: we consider the attacker both aiming to misclassify a sample to a specific class, and to any of the non-true classes.

The **attacker's knowledge** of the targeted system can include knowledge on e.g. the training data, the feature representation, the learning algorithm along with the objective function, and the learned parameters. We can distinguish the following cases. In a *white-box setting*, the attacker is assumed to know everything, while a *black-box setting* means that attacker knows nothing about the targeted system; *grey box* is any setting in between. We focus primarily on white-box attacks, for two reasons. First, white box is harder to defend against – if CNNs can be protected against an attacker with complete knowledge, then they can likely be protected in the same way and (at least) to the same degree when the attacker knows less. Secondly, Papernot et al. [14] showed that one can train a substitute model given black-box access to a target model, and transfer an attack on this back to the target model. Thus, they transform an originally black- into a white(r)-box setting.

Attacker's Capability. In terms of the influence on the data, adversarial examples are an exploratory attack, i.e. the attacker can only manipulate *test data* (also known as evasion attack). In contrast, in a causative attack, the attacker can manipulate both training and test data (commonly known as poisoning attacks). We require no specific data manipulation constraints, except the resulting inputs to be still valid.

This paper is organised as follows. Section 2 discusses related work. Section 3 describes our evaluation setup, and Section 4 discusses our results. We conclude and discuss future work in Section 5.

2 RELATED WORK

Adversarial Example Generation. Szegedy et al. [19] first noticed the existence of adversarial examples in image classification, showing that state-of-the-art neural networks (NN) are surprisingly vulnerable. Several further methods to craft adversarial examples were subsequently proposed. Many methods assume white-box access, including to the architecture and learned parameters.

The Fast Gradient Sign Method (FGSM) [8] is a popular attack, as it is capable of crafting adversarial examples with relatively small perturbation fast. It is optimised for the L_∞ norm as distance metric to quantify the similarity between original and adversarial samples. Intuitively, for each pixel, FSGM uses the gradient of the loss function to determine in which direction the pixel's intensity should be changed (increased or decreased) to minimise the loss. A parameter ϵ determines the strength of the change.

FGSM was designed to be rather fast than optimal. Since then, many improvements were introduced, to defeat proposed defences. As such, the Basic Iterative Method (BIM) [10], replaces the single step of size ϵ in the direction of the gradient sign with multiple smaller steps α. Additionally, the result is clipped by the same ϵ.

Projected Gradient Descent (PGD) [11] is another iterative method where the perturbation is projected on an l_p-ball of specified radius after each iteration. This is done in addition to clipping, to ensure that the samples lie in the permitted data range. PGD is formulated as a constraint optimisation problem to find a perturbation that maximises the loss function used to train the CNN model, while the perturbation stays inside the L_p ball of the original sample.

Instead of solving the constrained optimisation problem of PGD, the Shadow attack [7] optimises a range of components, which (i) force the perturbation to have a small total variation to appear smooth and natural, (ii) limit the perturbation globally by constraining the change in the mean of each colour channel to suppress extreme changes, and (iii) promote perturbations that assume similar values in each colour channel, which results in making the pixels darker/lighter without changing the colour balance of the image. The penalties minimise the perception of perturbations, while at the same time, they allow perturbations that are very large in L_p-norm.

Carlini and Wagner (CW) attacks [3] find an adversarial transformation for an input that minimises the difference between the original and perturbed image, but changes the classification of that input, while the result is still a valid image. This is difficult to solve directly, and thus the problem is reformulated as a heuristic optimisation. Empirically, an objective function was found, which is adjusted for the specific metric (L_1, L_2 or L_∞). For the latter, the distance metric is not fully differentiable and standard gradient descent does not perform well for it, which is resolved by using an iterative attack. CW incurs large computational overhead.

While the above-mentioned attacks create a perturbation specific to one single input, a universal perturbation ([12]) is able to fool a model on most inputs, always with the same perturbation. These input-agnostic perturbations are generally larger, but still remain quasi-imperceptible. The perturbation is desired to be small in terms of the L_p norm, with $p \in [1, \infty)$). One parameter controls the magnitude of the perturbation, and a second parameter specifies the desired fooling rate for all images. Universal perturbations often generalise well across different models and thus result in universal perturbations that are both image- and model-agnostic.

HopSkipJump [4] is a black-box attack based on a novel estimate of the gradient direction using binary information at the decision boundary. The algorithm is iterative, and can perform untargeted and targeted attacks. HopSkipJump requires fewer queries than other decision-based attacks, e.g. the Boundary Attack [2].

Adversarial Defences. Adversarial examples are hard to defend against. A theoretical solution to the process of generating adversarial examples is difficult to construct, as many attacks are non-linear and non-convex optimisation problems. Thus, it is equally hard to derive any theoretical conclusions that a given defence improves robustness against adversarial examples. Further, if a defence mechanism makes a considerable modification to the model or input, it may affect the ability to correctly predict legitimate, unmodified inputs. Robustness could thus be at the cost of the effectiveness. Finally, most current defence strategies are defending against a specific attack, but might be still vulnerable to other types of attacks.

Current defences can be categorised as follows [13]. **Detectors** attempt to detect the perturbations added to inputs. For example, the Fast Generalized Subset Scan [18] adapts the subset scanning methods from the anomalous pattern detection. **Preprocessors** modify the input to the model, to remove or at least disturb the perturbation. **Trainers** re-train the model so that it is more robust to adversarial samples, by adding adversarial examples crafted by the defender to the training set. **Transformers** also perform model (re-)training, but often also introduce a change to the model architecture, e.g. via Defensive Distillation [15].

We focus primarily on preprocessors and adversarial training. They are well suited for combination, and currently state-of-the-art. At the moment of publication, the authors of [15] believed distillation would counter all attacks, mainly because it was believed that the reason adversarial examples exist is due to “blind spots“ (as Szegedy et al. [19] call them) in highly non-linear neural networks. However, the CW attacks [3] showed that this is incorrect – they can achieve a 100 % success rate also on distilled models.

Preprocessors. Spatial Smoothing is part of the approach called Feature Squeezing by Xu et al. [23]. The core assumption is that the input space in image recognition tasks are often very large, and thus provide a lot of freedom to craft adversarial examples. The goal is to limit this by “squeezing“ out unnecessary input features. After the original input is preprocessed, both the original and preprocessed inputs are given to the model to classify. If the predictions differ significantly, the input is regarded as an adversarial example. Local smoothing methods make use of the nearby pixels to smooth each pixel, e.g. via median smoothing (or blur or filter), where the centre pixel of a sliding window is replaced with the median value of is neighbours. The size of the window can be defined from 1 pixel to up to the image size, whereas the shape is square. Median smoothing is particularly effective at removing sparsely occurring black and white pixels in an image (known as "salt-and-pepper noise"), whilst preserving edges of objects well. [23] empirically shows that it performs especially well against attacks based on the L_0 norm.

JPEG (re-)compression [6] is inspired by the fact that most image classification datasets are JPG compressed. When a JPEG image is transformed into an adversarial example, it may no longer be in JPEG space – thus, re-compression might revert the perturbation. This has been shown to be true for small perturbations, but not if the adversarial perturbations are larger – a result not promising, as even larger perturbations are still barely visible to humans. However, this defence can be easily combined with others. We speculate that such a combination may increase the strength of the defence.

Guo et al. [9] proposed Total Variance Minimisation (TVM), a compressed sensing approach [1] that combines pixel dropout with total variation minimisation [16]. The technique randomly selects a small set of pixels and reconstructs an image that is consistent with the selected pixels. As adversarial perturbations are usually small and localised, the reconstructed image is not adversarial anymore.

Wang et al. [20] argue that a strong input-transformation defence should be non-differentiable and randomised. TVM fulfils both properties – it is difficult to differentiate because it involves a complex minimisation of a function that is inherently random, and it randomly selects the pixels used for reconstruction. Randomness is important as it implies that the adversary has to find a perturbation that changes the prediction for the entire dataset, which is harder than attacking a single image as shown in [12].

The authors note that (i) TVM has an advantage over adversarial training, as the latter are differentiable, and (ii) adversarial training is based on the specific adversarial attack(s) selected, whereas TVM generalises well across different adversarial attacks.

Adversarial Training. This approach generates adversarial examples and includes them into the training set, so that the classifier learns the adversarial patterns. One important aspect is the choice of the attack; initially FGSM was preferred, mainly because of its speed and the fact that many adversarial attacks are extensions and generalisations of FGSM. However, Madry et al. [11] showed empirically that using FGSM does not increase robustness for large ϵ. They assume this is due to the model easily overfitting the generated adversarial examples, as the adversary produces a very restricted set of these. Moreover, it does not exhibit any kind of robustness against PGD – which they propose to use instead. This defence has remained very robust since, but comes at a large computational overhead, often by an order of magnitude, as adversarial examples are generated in each training step and for all samples in the batch.

Using PGD for more complex architectures is thus prohibitive, as training the network itself is already computational expensive. Shafahi et al. [17] thus proposed recycling the gradient computed when updating model parameters to eliminate the overhead of generating adversarial examples. Their so-called "free" adversarial training algorithm achieves robustness comparable to [11].

Wong et al. [22] showed empirically that adversarial training using FGSM with random initialisation is as robust as using PGD. Adversarial training with FGSM with random initialisation combined with techniques for efficient training is significantly faster than [17]; they thus call their method "fast is better than free".

Adversarial Robustness Metrics. CLEVER [21] (**C**ross **L**ipschitz **E**xtreme **V**alue for n**E**twork **R**obustness) is a robustness metric that is computationally feasible for large neural networks. CLEVER can be seen as an attack-agnostic, efficient estimator of the lower bound for the minimum distortion. The authors showed that the CLEVER score corresponds to the practical robustness indication of several state-of-the-art architectures, even when a defence mechanism is deployed. The score requires the Lipschitz constant L_q, which can be computed through sampling a set of points in a ball around a sample and taking the maximum value of the gradient. A significant amount of samples might be needed to obtain a good estimate. However, extreme value theory ensures that the maximum value of random variables can only follow one of the three extreme value distributions, which is useful to estimate the maximal gradient with only a tractable number of samples.

3 EVALUATION SETUP

Based on the datasets used in related work on adversarial attacks and defences, we have selected CIFAR-10 and ImageNet as two datasets that are (i) commonly used and (ii) more complex.

CIFAR-10 [2] consists of 6,000 examples for each of its 10 classes. The images are split into 50,000 training and 10,000 test images. Each image has dimensions of 32 × 32 pixels. To be comparable to current state-of-the-art, we use the ”PreAct“ version of ResNet18 [22] (”*PA-ResNet18* “) and an adversarially trained version of it (”*AdvTrn-PA-ResNet18* “). We use “faster is better than free“ adversarial training [17], which uses PGD to generate adversarial examples.

ImageNet[3] is a large image dataset organised along the “synonym set“ (“synset“) concepts of the WordNet hierarchy. We use the

[1] A signal processing technique for efficiently acquiring and reconstructing a signal

[2] https://www.cs.toronto.edu/~kriz/cifar.html

[3] https://image-net.org/

subset from the ImageNet Large Scale Visual Recognition Challenge (ILSVRC), containing 1.28 million training images. We use three state-of-the-art pre-trained CNN models: *MobileNetV2*, *ResNet-50*, and *InceptionV3*. These models vary in complexity (24, 3.4, and 23 million learnable parameters) and effectiveness. They were selected to evaluate how different architectures affect attacks and defences.

For both datasets, we randomly select 1,000 images from the test set for generating adversarial examples and attacking the models.

Due to their diversity, strength, and importance, we use the following adversarial attacks: Auto-PGD [5], Carlini and Wagner L^2 and L^∞ [3], FGSM [8], HopSkipJump [4], Shadow attack [7] and Universal Perturbations [12].

We calculate the success of an attack by the percentage of the successfully attacked inputs. An image is successfully attacked if (i) in the targeted setting the classifier predicts the targeted class, or (ii) in the untargeted setting if the classifier predicts any class other than the original. In line with literature, we evaluate attacks only on the portion of the test set that was correctly predicted by the model. We also calculate the (dis)-similarities of original to adversarial images, by measuring their L^2 and L^∞ pixel-wise distances.

As defences, we use the “fast is better than free“ adversarial training [22], and three pre-processing defensive techniques: spatial smoothing [23], JPEG compression [6], and Total Variance Minimisation [9]. These defence techniques are commonly used in the research as benchmarks for how strong the adversarial attacks are, as well as for evaluating other adversarial defences. We compute the change in the success rates of the attacks on the classifiers that use the defences, and the change in effectiveness of these model on their initial (classification) task. Additionally, we combine defences and evaluate if this improves the robustness of the classifier. We exhaust all different combinations. We use the implementation of attacks and defences from the Adversarial Robustness Toolbox [13].

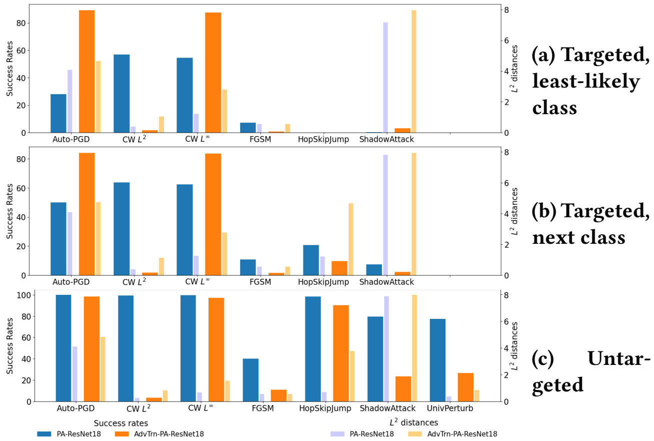
4 EVALUATION

In terms of runtime, on CIFAR-10, all attacks needed approximately 1 second per image, except CW L^2, which took 15 fold of that. On Imagenet, again 1 second per image was needed, except for HopSkipJump with 3, CW L^∞ with 19, Universal Perturbations with 134, and CW L^2 with 234 seconds.

4.1 Baseline without pre-processing defence

CIFAR-10. PA-ResNet18 performs slightly better on the original (clean) data, but is more affected by adversarial examples than *AdvTrn-PA-ResNet18* in all settings, as shown in Figure 1. This is in line with [22] reporting a drop of ≈ 30% in attack success when defending. The exceptions are Auto-PGD and CW L^∞ in the targeted settings, where the success rates *increase* against the defended model; [22] does not evaluate these two attacks, but our observation indicates that against them, this defence might not be robust. These attacks require, however, a perturbation larger than most other attacks to be succesful, thus the success comes at the cost for the attacker; only ShadowAttack has a higher L^2 distance.

In general, across all attacks and all settings, attacks had to modify images much more for them to be adversarial against adversarially re-trained than against the original model. As expected, the untargeted setting is the easiest for most attacks.



(a) Targeted, least-likely class

(b) Targeted, next class

(c) Untargeted

Figure 1: CIFAR 10: Attack success and L^2 distances

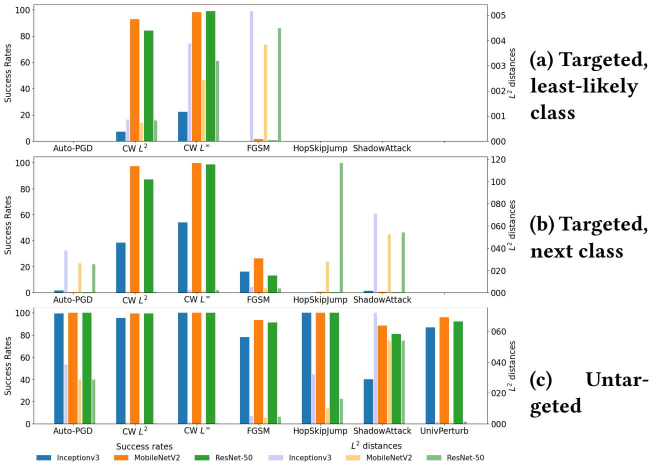
CW L^2 attack has very low success rates in both targeted settings against *AdvTrn-PA-ResNet18*, while it has about 60 % success rate against the original model. HopSkipJump has managed to bypass the adversarial training defense in the *untargeted* settings, scoring 90.12% success rates, but with a high L^2 distance.

Rgarding thet L^2 (dis)-similarities, CW L^2 and L^∞ introduced only small changes against the original model, but achieve almost perfect success rates in all target settings. Auto-PGD modified the images three times more just to achieve a similar success rate in the target settings. On the other hand, the Shadow attack, despite causing the highest modifications, achieved low success rates in the targeted settings, even on the original model.

ImageNet. Figure 2 shows that in all settings, the evaluated attacks exhibit (slightly) lower success against *InceptionV3* than against *MobileNetV2* and *ResNet-50*. We speculate that this is due to *InceptionV3* being a more complex model with more parameters. While this behavior is less apparent in the other two, it is quite obvious in the *targeted least likely class* setting with CW attacks. With relatively similar L^2 distances between adversarial and original images, both CW L^2 and L^∞ attacks have scored 80+ % success rates against *MobileNetV2* and *ResNet-50*, whereas *InceptionV3* remained fairly robust with very low success rates of 7.27% and 22.2% respectively. Interestingly, the other attacks in the *targeted least likely class* setting have been unsuccessful, which highlights the strength of the CW attack.

Another obervation is that it is hard to successfully attack a specific class such that the pertubations are minimal, which holds also for the *targeted next class* setting (albeit with slighty higher success rates than the leasty-likely class). We speculate that the reason for this could be the fact that in the 1,000 classes in ImageNet, some are semantically closer to each other, e.g. ”pan“ and ”wok pan“ – attacks that have a similar class as next class would be expected to be easier targets than the likely more distant least likely class.

All untargeted attacks have high success against all models; in particular, Auto-PGD, CW, HopSkipJump, and Universal Perturbations score 80+ % success rates. The Shadow attack had to generated much larger perturbations on *InceptionV3* compared to the other



(a) Targeted, least-likely class

(b) Targeted, next class

(c) Untargeted

Figure 2: ImageNet: Attack success and L^2 distances

models, but still scored a much lower success rate (40% vs. >80%), which indicates that *InceptionV3* is more robust against this attack.

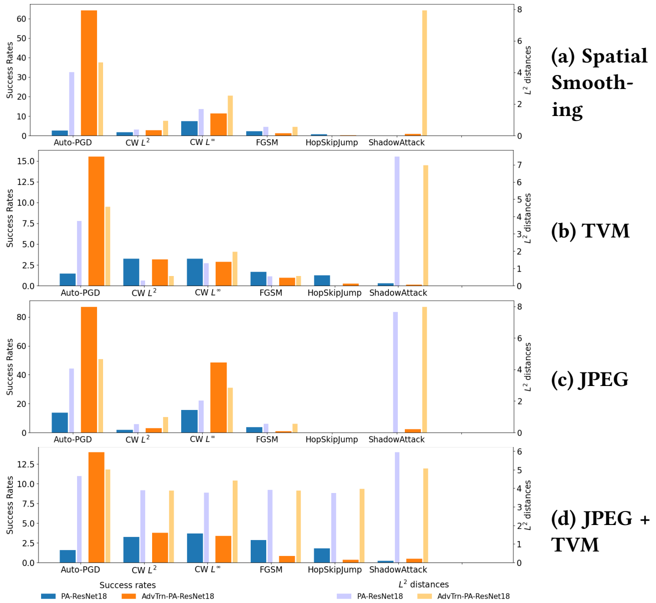
The L^2 distances are low for CW, FGSM and Universal Perturbations. The Shadow attack is the weakest, as it scored the lowest success rates while generating the largest perturbation.

4.2 Pre-processing against targeted attacks

CIFAR-10. The trends and observations when targeting the *next* class are very similar as for the least likely case, and we focus on that setting thus. The perhaps most important observation is that all defense scenarios reduce the success of all attacks, except Auto-PGD on *AdvTrn-PA-ResNet18* (cf. Figure 3). The reductions are similar to adversarially training alone, except for Auto-PGD and CW L^∞; those successfully broke adversarial training, but the pre-processing defenses have much better success. For example, Auto-PGD and CW L^∞ have scored only about 2% success rate against *PA-ResNet18* and Total Variance Minimization (TVM).

The combination of adversarial training and pre-processing has reduced success rates of most attacks to single digits. Most interestingly, CW L^∞ was reduced significantly, from 87.7% to e.g. 3.7% with all pre-processing defenses combined. For Auto-PGD, however, only TVM managed to reduce the success, from 89.4% to 15.5%.

One can observe that when two or more pre-processing defenses are combined (e.g. JPEG and TVM in Figure 3d), the L^2 metric differences became larger than when pre-processing with only one defense. This, however does not translate to higher success rates of the attacks; thus, the combinations are beneficial for the defender. Overall, most combinations perform similar.



(a) Spatial Smoothing

(b) TVM

(c) JPEG

(d) JPEG + TVM

Figure 3: CIFAR-10, least-likely class against against defences: Attack success and L^2 distances

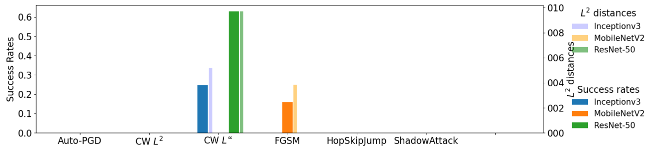


Figure 4: ImageNet, targeted least-likely class, Spatial Smoothing defence: Attack success and L^2 distances

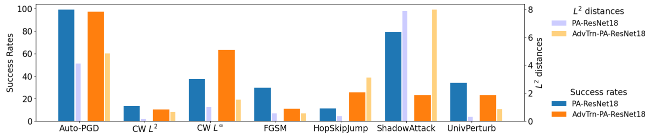


Figure 5: CIFAR-10, untargeted, JPEG Compression defence: Attack success and L^2 distances,

ImageNet. The Auto-PGD, FGSM, HopSkipJump, and Shadow attacks had already almost no success, thus virtually no further reduction was possible. The thus most interesting case for the least-likely target class is CW, which had high success rates against *MobileNetV2* and *ResNet-50* (cf. Figure 2). Pre-processing works very well, and reduces the attack to almost zero success. Overall, all defences are rather equal, and as example, we show Spatial Smoothing (SS) in Figure 4. Only CW L^∞ against JPEG Compression scored marginally sucessful, with 16.6% on *ResNet-50*, 6.2% on *MobileNetV2* and 2.22% on *InceptionV3*.

Pre-processing in the next-class setting is only slightly worse than for the least-likely class setting. As above, we speculate that this correlates with the similarities of some classes in ImageNet.

4.3 Pre-processing against untargeted attacks

CIFAR-10. As the *untargeted* setting allows for *any* misclassification, higher success rates are achieved, as we saw also in Section 4.1.

Unlike the targeted setting, the combination of adversarial training and TVM fails in at least 20% of adversarial images. For example, Auto-PGD has a success rate of 4.6% for *targeted next class*, but 74.2% in the *untargeted* setting. However, compared to the baseline results, we notice an overall great reduction in success on *PA-ResNet18*.

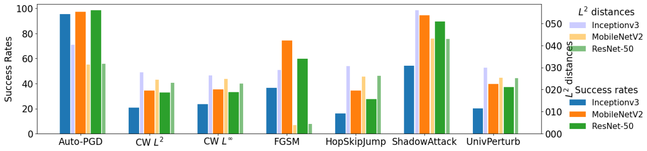


Figure 6: ImageNet, untargeted, Spatial Smoothing + JPEG Compression defence: Attack success and L^2 distances

Adversarial training, however, did not lead to a significant further sucess reduction – in average only by 10-20%. FGSM, e.g., is still successful with 30.8% against *AdvTrn-PA-ResNet18* and TVM. ShadowAttack remained strong against pre-processing. The highest success rate against *PA-ResNet18* is 79.3 %, whereas with the adversarially trained model it is 23.1 %, both with JPEG Compression. A noteworthy improvement of the defense was recorded with the combination of Spatial Smoothing and JPEG Compression, which managed to reduce the success rate of the ShadowAttack on the *AdvTrn-PA-ResNet18* model to 3.6 %.

ImageNet. Also here, the *untargeted* setting is the most difficult to defend against. Spatial Smoothing has the most success against CW attacks, HopSkipJump and Universal Perturbations. The adversarial images generated with these attacks also have the lowest L^2 distances, which indicates that the fewer modification the attack introduces, the easier it is for a pre-processing defense to correct the classification. On the other hand, Spatial Smoothing had less success against FGSM, and almost no efect against Auto-PGD and Shadow Attack. The adversarial images generated with the latter two have the highest L^2 distance, supporting our assumption.

JPEG Compression only managed to reduce the success rates on average to 50 %, which means that still half of the attacks succeed. Besides, the trends for which attacks were stronger or weaker against this defense are similar as in the case of Spatial Smoothing.

Unlike SS, TVM was able to reduce the success rate of Auto-PGD, but was in turn less successful against HopSkipJump and Universal Perturbations, and equally well against CW.

The combination of SS and TVM defendes less than the defenses individually, though it increases the L^2 distances significantly. This is most pronounced for CW attacks, which in other scenarios had very low L^2 distances. Other combinations also notably increase the L^2 distances, and are equal or even further reduce the attack success than individual defences; a good example is the combination of SS and JPEG Compression shown in Figure 6.

To summarise, pre-processing defenses indeed reduce the success rates of most attacks significantly, however, they are less effective against untargeted attacks. Further, attacks that generate more distant adversarial images are more robust against pre-processing.

4.4 CLEVER scores

CIFAR-10. Table 1 shows the CLEVER scores against CW L^2 as a representative example. On the baseline case (no defences) and the least-likely class setting, *AdvTrn-PA-ResNet18* scored a higher CLEVER score w.r.t L^2 norm than *PA-ResNet18*, which is in line with success rates of the CW L^2 as discussed in Figure 2. However, the score of (the undefendend) *PA-ResNet18* is one of the highest overall, which would indicate that the model would be more robust than most others. However, CW L^2 has a high success rate ($\approx$ 60%) against *PA-ResNet18*. The same observations hold also for the *targeted least likely* and *untargeted* setting.

Table 1: CLEVER scores on CIFAR-10 against C&W L^2

		CLEVER norm L^2			CLEVER norm L^∞		
		least likely	next class	untargeted	least likely	next class	untargeted
PA-ResNet18	No defences	0.11729	0.12792	0.11187	0.00252	0.00286	0.00295
	SS	0.02599	0.02626	0.0236	0.00071	0.00072	0.00072
	JPEG	0.04642	0.06952	0.04384	0.00149	0.00186	0.0012
	TVM	0.00399	0.01235	0.01937	0.0000	0.00013	0
	SS + JPEG	0.00558	0.00574	0.00609	0.0002	0.00018	0.00023
	SS + TVM	0.00593	0.0219	0.00337	0.00027	0.00101	0.00013
	SS + JPEG + TVM	0.0118	0.01104	0.00387	0.00043	0.00043	0.00015
	JPEG + TVM	0.05579	0.04603	0.03179	0.00136	0.001	0.00135
AdvTrn-PA-ResNet18	No defences	0.15272	0.05773	0.00132	0.00338	0.00304	5e-05
	SS	0.02314	0.00647	0.01898	0.00072	0.00017	0.00058
	JPEG	0.04138	0.02351	0.0007	0.00306	0.0008	3e-05
	TVM	0.01287	0.01305	0.01225	0.00012	0.00027	0.00042
	SS + JPEG	0.00552	0.02028	0.0082	0.00021	0.0008	0.00032
	SS + TVM	0.0267	0.01433	0.00719	0.00079	0.00078	0.00033
	SS + JPEG + TVM	0.00338	0.00204	0.02905	9e-05	8e-05	0.00116
	JPEG + TVM	0.05983	0.0345	0.02796	0.002	0.00178	0.00097

Table 2: CLEVER scores on ImageNet against FGSM

		CLEVER norm L^2			CLEVER norm L^∞		
		least likely	next class	untargeted	least likely	next class	untargeted
InceptionV3	No defences	0.16298	0.23112	2.0	0.00036	0.00791	0.00668
	SS	0.24742	0.00494	2.0	0.00112	0.00136	0.00366
	JPEG	0.68033	0.78041	2.0	0.002	0.00229	0.00629
	TVM	0.03564	0.02002	0.03752	0.0001	0.00007	0.0001
	SS + JPEG	0.79638	0.80107	1.86768	0.00144	0.00229	0.00235
	SS + TVM	0.02222	0.01779	0.02942	0.0001	0.00014	8.00E-05
	SS + JPEG + TVM	0.03715	0.04149	0.03496	0.00012	0.00011	0.0001
	JPEG + TVM	0.0289	0.02801	0.02677	0.0001	0.00012	7e-05
MobileNetV2	No defences	0.18426	0.14277	1.01925	0.00061	0.00062	0.00331
	SS	0.25983	0.26468	2.0	0.00118	0.00112	0.00272
	JPEG	0.47149	0.58493	1.24635	0.0027	0.00263	0.00483
	TVM	0.03391	0.00014	0.02647	0.0001	0.0001	0.0001
	SS + JPEG	0.64274	0.33064	0.90888	0.00117	0.00128	0.00174
	SS + TVM	0.04123	0.03478	0.03833	0.00014	0.0001	0.0002
	JPEG + TVM	0.02359	0.01683	0.02974	7e-05	7e-05	0.00012
ResNet-50	No defences	2.0	0.65432	2.0	0.011	0.00336	0.04222
	SS	2.0	0.38601	2.0	0.00449	0.00184	0.00917
	JPEG	2.0	1.72062	2.0	0.00606	0.00336	0.02721
	TVM	0.02947	0.02442	0.02637	0.00014	0.00015	0.00012
	SS + JPEG	2.0	1.01856	2.0	0.00734	0.0032	0.01041
	SS + TVM	0.03284	0.01449	0.02144	0.00013	6e-05	7e-05
	SS + JPEG + TVM	0.01557	0.01153	0.00678	9e-05	3e-05	0.00011
	JPEG + TVM	0.01015	0.00939	0.01425	8e-05	6e-05	7e-05

With pre-processing defences, CW L^2 attacks have very similar success rates in all three settings, and thus their CLEVER scores should be also similar. This is indeed true for several cases, e.g. TVM for *targeted next class* has very similar scores for both models, as has JPEG Compression combined with TVM in the *targeted least-likely class* setting. However, this does not hold for several other cases, such as JPEG Compression in the *untargeted* setting.

When we compare the CLEVER scores for the CW L^2 attack with CLEVER scores for other attacks, we observe similar trends.

ImageNet. As a representative example, we show CLEVER against FGSM in Table 2; other attacks have very similar behaviour. We can notice that some of the scores correlate positively and some rather negatively with the actual success rates of the attacks. For example, for most individual pre-processing defences there is an increase in the CLEVER score. However, this is inverse for the combinations of defences, with the exception of SS and JPEG compression. It is also noteworthy that the *untargeted* setting yields higher CLEVER scores, which is not expected, as the success rates have been far better in that setting than in the targeted settings.

Concluding for both datasets, this means that while *some* CLEVER scores are in line with the success rates and observations made in the previous sections, others are not. CLEVER scores can not fully explain (predict) the effectiveness of adversarial attacks, and CLEVER needs to be complemented by other scores to estimate the effect of attacks and defences. Finally, with the pre-processing defences, the models are empirically more robust, and one would expect that the CLEVER scores would reflect that; however, our experiments indicate the opposite: the scores are slightly worse with pre-processing defences.

5 CONCLUSIONS AND FUTURE WORK

In this paper, we evaluated several adversarial attacks and defences on ImageNet and CIFAR-10 under three different target settings. We additionally combined defences to potentially increase their success. Further, we analysed the correlation of the CLEVER robustness metric and the actual, measured attack success.

In the targeted setting, we observed that models on both datasets are relatively robust against adversarial attacks. The only attacks that managed to target specific classes with higher success rates were CW on both datasets, and Auto-PGD on CIFAR-10. In the *untargeted* setting *AdvTrn-PA-ResNet18* was vulnerable against Auto-PGD, CW L^{∞}, and HopSkipJump, however, it was stronger against other attacks; all other models were vulnerable.

Pre-processing defences in the targeted setting have almost completely removed the success of the attacks. When comparing the two targeted settings, we have to stress that the *targeted least likely* and *targeted next class* settings showed very different success rates on ImageNet, while they were very similar on CIFAR-10.

In the *untargeted*, we have noticed large reductions of success rates with different pre-processing defences on both datasets, with very similar trends. Only, Auto-PGD and Shadow attacks were very strong against all pre-processors. We argue these defences provide better protection against attacks that generate adversarial images that are **closer** to the original images. An example is CW L^2, which has been most reduced by the pre-processors, albeit being an otherwise very powerful attack.

Regarding specific architectures, we can see that *InceptionV3* was the most robust in many settings without any defence. On ImageNet, the success of pre-processors in the *targeted next class* compared to the the least-likely setting might correlate with semantic similarities between the (randomly) chosen classes.

In general, adversarial training and pre-processors combine well, and some of the combinations, like adversarial training, SS, and JPEG Compression are noteworthy candidates to generalise well.

Future work will focus on extending these experiments to more datasets, models, and robustness scores, and include further defences for combination.

REFERENCES

[1] Battista Biggio and Fabio Roli. 2018. Wild patterns: Ten years after the rise of adversarial machine learning. *Pattern Recognition* 84 (2018).

[2] Wieland Brendel, Jonas Rauber, and Matthias Bethge. 2018. Decision-Based Adversarial Attacks: Reliable Attacks Against Black-Box Machine Learning Models. In *Int. Conf. on Learning Representations (ICLR)*. Vancouver, BC, Canada.

[3] Nicholas Carlini and David Wagner. 2017. Towards Evaluating the Robustness of Neural Networks. In *IEEE Symposium on Security and Privacy (SP)*. IEEE, San Jose, CA, USA.

[4] Jianbo Chen, Michael I. Jordan, and Martin J. Wainwright. 2020. HopSkipJumpAttack: A Query-Efficient Decision-Based Attack. In *IEEE Symposium on Security and Privacy (SP)*. IEEE, San Francisco, CA, USA.

[5] Francesco Croce and Matthias Hein. 2020. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *Int. Conf. on Machine Learning (ICML)*. PMLR.

[6] Gintare Karolina Dziugaite, Zoubin Ghahramani, and Daniel M. Roy. 2016. A study of the effect of JPG compression on adversarial images. arXiv:1608.00853.

[7] Amin Ghiasi, Ali Shafahi, and Tom Goldstein. 2020. Breaking Certified Defenses: Semantic Adversarial Examples with Spoofed robustness Certificates. In *Int. Conf. on Learning Representations (ICLR)*. Addis Ababa, Ethiopia.

[8] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and Harnessing Adversarial Examples. In *Int. Conf. on Learning Representations (ICLR)*. San Diego, CA, USA.

[9] Chuan Guo, Mayank Rana, Moustapha Cissé, and Laurens van der Maaten. 2018. Countering Adversarial Images using Input Transformations. In *Int. Conf. on Learning Representations (ICLR)*. Vancouver, BC, Canada.

[10] Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. 2017. Adversarial examples in the physical world. In *Int. Conf. on Learning Representations (Workshop Track Proceedings) (ICLR)*. Toulon, France.

[11] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2018. Towards Deep Learning Models Resistant to Adversarial Attacks. In *Int. Conf. on Learning Representations (ICLR)*. Vancouver, BC, Canada.

[12] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. 2017. Universal Adversarial Perturbations. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Honolulu, HI.

[13] Maria-Irina Nicolae, Mathieu Sinn, Minh Ngoc Tran, Beat Buesser, Ambrish Rawat, Martin Wistuba, Valentina Zantedeschi, Nathalie Baracaldo, Bryant Chen, Heiko Ludwig, Ian M. Molloy, and Ben Edwards. 2019. Adversarial Robustness Toolbox v1.0.0. arXiv:1807.01069 [cs, stat].

[14] Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow. 2016. Transferability in Machine Learning: from Phenomena to Black-Box Attacks using Adversarial Samples. arXiv:1605.07277.

[15] Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. 2016. Distillation as a Defense to Adversarial Perturbations Against Deep Neural Networks. In *IEEE Symposium on Security and Privacy (SP)*. IEEE, San Jose, CA.

[16] Leonid I. Rudin, Stanley Osher, and Emad Fatemi. 1992. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena* 60, 1-4 (1992).

[17] Ali Shafahi, Mahyar Najibi, Mohammad Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. 2019. Adversarial training for free!. In *Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc.

[18] Skyler Speakman, Srihari Sridharan, Sekou Remy, Komminist Weldemariam, and Edward McFowland. 2018. Subset Scanning Over Neural Network Activations. arXiv:1810.08676.

[19] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. 2014. Intriguing properties of neural networks. arXiv:1312.6199.

[20] Qinglong Wang, Wenbo Guo, Kaixuan Zhang, Alexander G. Ororbia II, Xinyu Xing, Xue Liu, and C. Lee Giles. 2017. Learning Adversary-Resistant Deep Neural Networks. arXiv:1612.01401.

[21] Tsui-Wei Weng, Huan Zhang, Pin-Yu Chen, Jinfeng Yi, Dong Su, Yupeng Gao, Cho-Jui Hsieh, and Luca Daniel. 2018. Evaluating the Robustness of Neural Networks: An Extreme Value Theory Approach. In *Int. Conf. on Learning Representations (ICLR)*. Vancouver, BC, Canada.

[22] Eric Wong, Leslie Rice, and J. Zico Kolter. 2020. Fast is better than free: Revisiting adversarial training. In *Int. Conf. on Learning Representations (ICLR)*. Addis Ababa, Ethiopia.

[23] Weilin Xu, David Evans, and Yanjun Qi. 2018. Feature Squeezing: Detecting Adversarial Examples in Deep Neural Networks. In *Network and Distributed System Security Symposium (NDSS)*. Internet Society, San Diego, CA.